PIC16F913/914/916/917/946
Data Sheet

28/40/44/64-Pin Flash-Based, 8-Bit CMOS Microcontrollers with LCD Driver and nanoWatt Technology

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.

Trademarks

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2002

Microchip received ISO/TS-16949:2002 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.

PIC16F913/914/916/917/946

28/40/44/64-Pin Flash-Based, 8-Bit CMOS Microcontrollers with LCD Driver and nanoWatt Technology

High-Performance RISC CPU:

- Only 35 instructions to learn:
  - All single-cycle instructions except branches
- Operating speed:
  - DC – 20 MHz oscillator/clock input
  - DC – 200 ns instruction cycle
- Program Memory Read (PMR) capability
- Interrupt capability
- 8-level deep hardware stack
- Direct, Indirect and Relative Addressing modes

Special Microcontroller Features:

- Precision Internal Oscillator:
  - Factory calibrated to ±1%, typical
  - Software selectable frequency range of 8 MHz to 125 kHz
  - Software tunable
  - Two-Speed Start-up mode
  - External Oscillator fail detect for critical applications
  - Clock mode switching during operation for power savings
- Software selectable 31 kHz internal oscillator
- Power-Saving Sleep mode
- Wide operating voltage range (2.0V-5.5V)
- Industrial and Extended temperature range
- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Brown-out Reset (BOR) with software control option
- Enhanced Low-Current Watchdog Timer (WDT) with on-chip oscillator (software selectable nominal 268 seconds with full prescaler) with software enable
- Multiplexed Master Clear with pull-up/input pin
- Programmable code protection
- High-Endurance Flash/EEPROM cell:
  - 100,000 write Flash endurance
  - 1,000,000 write EEPROM endurance
  - Flash/Data EEPROM retention: > 40 years

Low-Power Features:

- Standby Current:
  - <100 nA @ 2.0V, typical
- Operating Current:
  - 11 μA @ 32 kHz, 2.0V, typical
  - 220 μA @ 4 MHz, 2.0V, typical
- Watchdog Timer Current:
  - 1 μA @ 2.0V, typical

Peripheral Features:

- Liquid Crystal Display module:
  - Up to 60/96/168 pixel drive capability on 28/40/64-pin devices, respectively
  - Four commons
- Up to 24/35/53 I/O pins and 1 input-only pin:
  - High-current source/sink for direct LED drive
  - Interrupt-on-change pin
  - Individually programmable weak pull-ups
- In-Circuit Serial Programming™ (ICSP™) via two pins
- Analog comparator module with:
  - Two analog comparators
  - Programmable on-chip voltage reference (CV_{REF}) module (% of V_{DD})
  - Comparator inputs and outputs externally accessible
- A/D Converter:
  - 10-bit resolution and up to 8 channels
- Timer0: 8-bit timer/counter with 8-bit programmable prescaler
- Enhanced Timer1:
  - 16-bit timer/counter with prescaler
  - External Timer1 Gate (count enable)
  - Option to use OSC1 and OSC2 as Timer1 oscillator if INTOSCIO or LP mode is selected
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Addressable Universal Synchronous Asynchronous Receiver Transmitter (AUSART)
- Up to 2 Capture, Compare, PWM modules:
  - 16-bit Capture, max. resolution 12.5 ns
  - 16-bit Compare, max. resolution 200 ns
  - 10-bit PWM, max. frequency 20 kHz
- Synchronous Serial Port (SSP) with I^2C™

Device	Program Memory	Data Memory		I/O	10-bit A/D (ch)	LCD (segment drivers)	CCP	Timers 8/16-bit
	Flash (words/bytes)	SRAM (bytes)	EEPROM (bytes)					
PIC16F913	4K/7K	256	256	24	5	16(1)	1	2/1
PIC16F914	4K/7K	256	256	35	8	24	2	2/1
PIC16F916	8K/14K	352	256	24	5	16(1)	1	2/1
PIC16F917	8K/14K	352	256	35	8	24	2	2/1
PIC16F946	8K/14K	336	256	53	8	42	2	2/1

Note 1: COM3 and SEG15 share the same physical pin on the PIC16F913/916, therefore SEG15 is not available when using 1/4 multiplex displays.

Pin Diagrams – PIC16F914/917, 40-Pin

40-pin PDIP

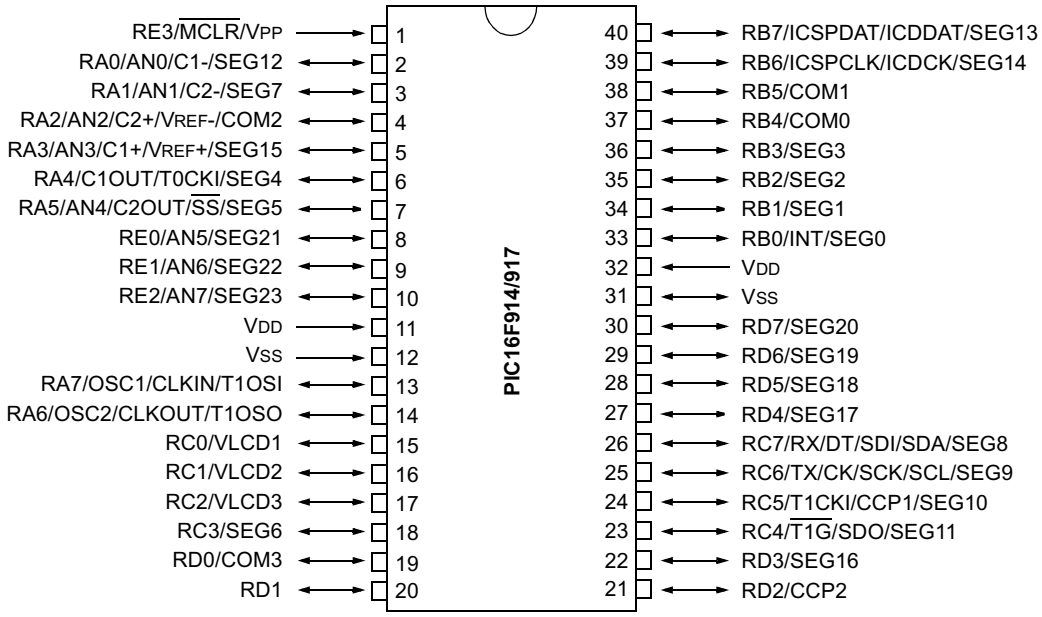


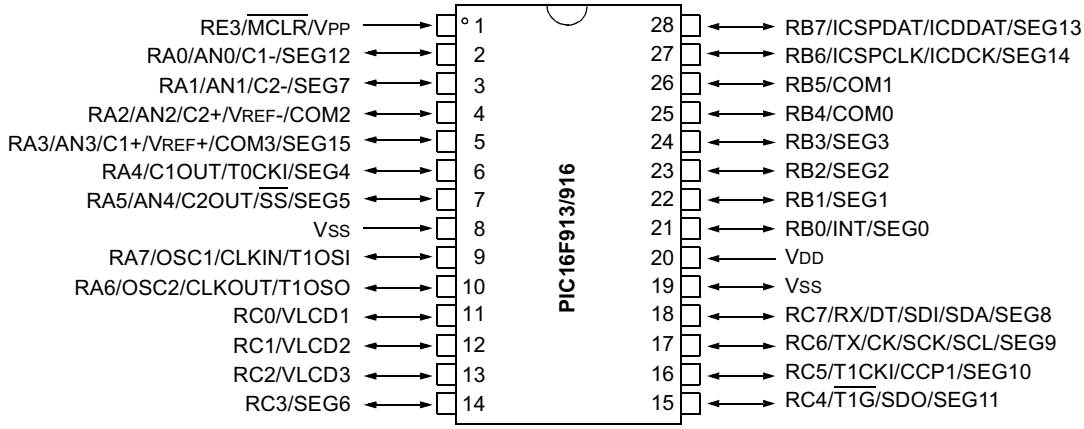
TABLE 1: PIC16F914/917 40-PIN SUMMARY

I/O	Pin	A/D	LCD	Comparators	Timers	CCP	AUSART	SSP	Interrupt	Pull-Up	Basic
RA0	2	AN0	SEG12	C1-	—	—	—	—	—	—	—
RA1	3	AN1	SEG7	C2-	—	—	—	—	—	—	—
RA2	4	AN2/VREF-	COM2	C2+	—	—	—	—	—	—	—
RA3	5	AN3/VREF+	SEG15	C1+	—	—	—	—	—	—	—
RA4	6		SEG4	C1OUT	T0CKI	—	—	—	—	—	—
RA5	7	AN4	SEG5	C2OUT	—	—	—	$\overline{SS}$	—	—	—
RA6	14	—	—	—	T1OSO	—	—	—	—	—	OSC2/CLKOUT
RA7	13	—	—	—	T1OSI	—	—	—	—	—	OSC1/CLKIN
RB0	33	—	SEG0	—	—	—	—	—	INT	Y	—
RB1	34	—	SEG1	—	—	—	—	—	—	Y	—
RB2	35	—	SEG2	—	—	—	—	—	—	Y	—
RB3	36	—	SEG3	—	—	—	—	—	—	Y	—
RB4	37	—	COM0	—	—	—	—	—	IOC	Y	—
RB5	38	—	COM1	—	—	—	—	—	IOC	Y	—
RB6	39	—	SEG14	—	—	—	—	—	IOC	Y	ICSPCLK/ICDCK
RB7	40	—	SEG13	—	—	—	—	—	IOC	Y	ICSPDAT/ICDDAT
RC0	15	—	VLCD1	—	—	—	—	—	—	—	—
RC1	16	—	VLCD2	—	—	—	—	—	—	—	—
RC2	17	—	VLCD3	—	—	—	—	—	—	—	—
RC3	18	—	SEG6	—	—	—	—	—	—	—	—
RC4	23	—	SEG11	—	$\overline{T1G}$	—	—	SDO	—	—	—
RC5	24	—	SEG10	—	T1CKI	CCP1	—	—	—	—	—
RC6	25	—	SEG9	—	—	—	TX/CK	SCK/SCL	—	—	—
RC7	26	—	SEG8	—	—	—	RX/DT	SDI/SDA	—	—	—
RD0	19	—	COM3	—	—	—	—	—	—	—	—
RD1	20	—	—	—	—	—	—	—	—	—	—
RD2	21	—	—	—	—	CCP2	—	—	—	—	—
RD3	22	—	SEG16	—	—	—	—	—	—	—	—
RD4	27	—	SEG17	—	—	—	—	—	—	—	—
RD5	28	—	SEG18	—	—	—	—	—	—	—	—
RD6	29	—	SEG19	—	—	—	—	—	—	—	—
RD7	30	—	SEG20	—	—	—	—	—	—	—	—
RE0	8	AN5	SEG21	—	—	—	—	—	—	—	—
RE1	9	AN6	SEG22	—	—	—	—	—	—	—	—
RE2	10	AN7	SEG23	—	—	—	—	—	—	—	—
RE3	1	—	—	—	—	—	—	—	—	Y(1)	$\overline{MCLR}$/VPP
—	11	—	—	—	—	—	—	—	—	—	VDD
—	32	—	—	—	—	—	—	—	—	—	VDD
—	12	—	—	—	—	—	—	—	—	—	VSS
—	31	—	—	—	—	—	—	—	—	—	VSS

Note 1: Pull-up enabled only with external $\overline{MCLR}$ configuration.

Pin Diagrams – PIC16F913/916, 28-Pin

28-pin PDIP, SOIC, SSOP



28-pin QFN

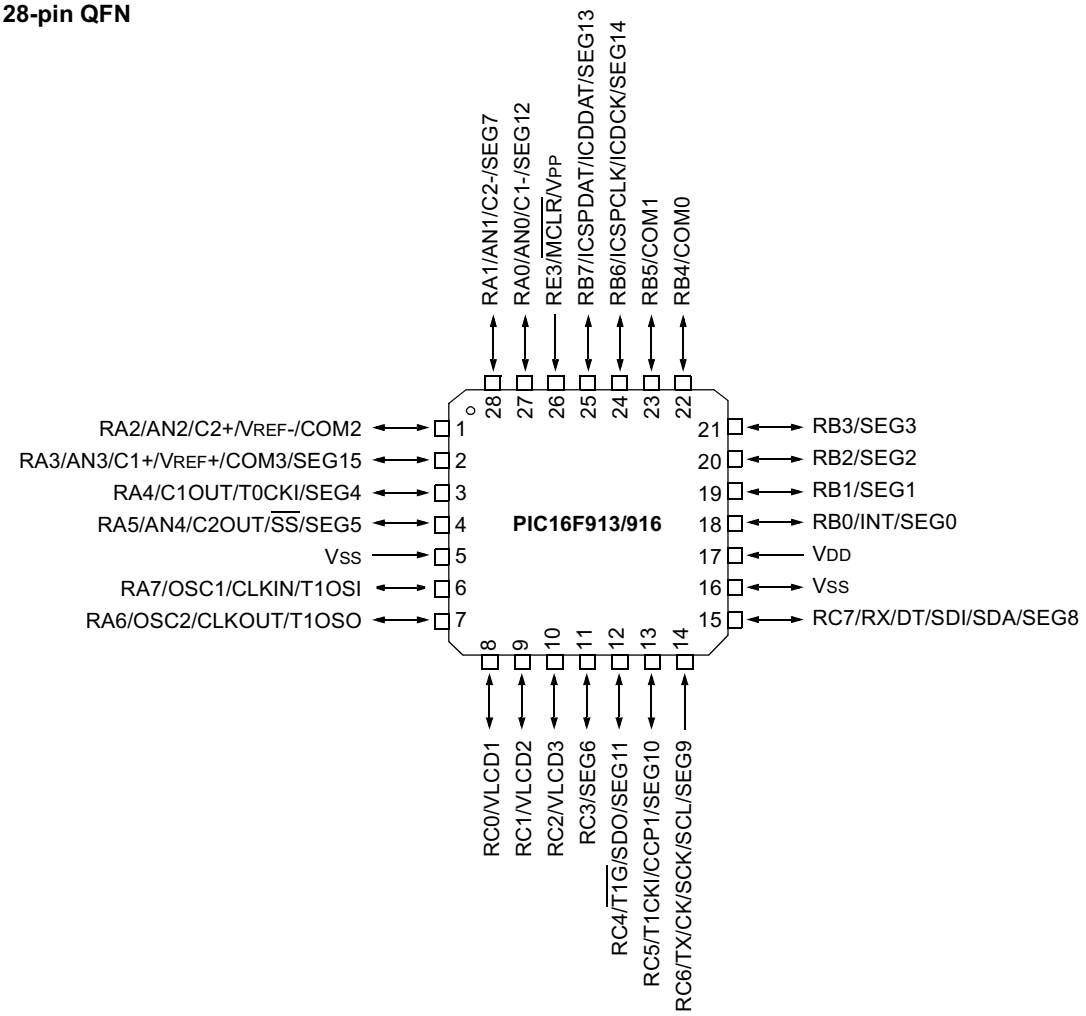


TABLE 2: PIC16F913/916 28-PIN (PDIP, SOIC, SSOP) SUMMARY

I/O	Pin	A/D	LCD	Comparators	Timers	CCP	AUSART	SSP	Interrupt	Pull-Up	Basic
RA0	2	AN0	SEG12	C1-	—	—	—	—	—	—	—
RA1	3	AN1	SEG7	C2-	—	—	—	—	—	—	—
RA2	4	AN2/VREF-	COM2	C2+	—	—	—	—	—	—	—
RA3	5	AN3/VREF+	SEG15/ COM3	C1+	—	—	—	—	—	—	—
RA4	6	—	SEG4	C1OUT	T0CKI	—	—	—	—	—	—
RA5	7	—	SEG5	C2OUT	—	—	—	$\overline{SS}$	—	—	—
RA6	10	—	—	—	T1OSO	—	—	—	—	—	OSC2/CLKOUT
RA7	9	—	—	—	T1OSI	—	—	—	—	—	OSC1/CLKIN
RB0	21	—	SEG0	—	—	—	—	—	INT	Y	—
RB1	22	—	SEG1	—	—	—	—	—	—	Y	—
RB2	23	—	SEG2	—	—	—	—	—	—	Y	—
RB3	24	—	SEG3	—	—	—	—	—	—	Y	—
RB4	25	—	COM0	—	—	—	—	—	IOC	Y	—
RB5	26	—	COM1	—	—	—	—	—	IOC	Y	—
RB6	27	—	SEG14	—	—	—	—	—	IOC	Y	ICSPCLK/ICDCK
RB7	28	—	SEG13	—	—	—	—	—	IOC	Y	ICSPDAT/ICDDAT
RC0	11	—	VLCD1	—	—	—	—	—	—	—	—
RC1	12	—	VLCD2	—	—	—	—	—	—	—	—
RC2	13	—	VLCD3	—	—	—	—	—	—	—	—
RC3	14	—	SEG6	—	—	—	—	—	—	—	—
RC4	15	—	SEG11	—	$\overline{T1G}$	—	—	SDO	—	—	—
RC5	16	—	SEG10	—	T1CKI	CCP1	—	—	—	—	—
RC6	17	—	SEG9	—	—	—	TX/CK	SCK/SCL	—	—	—
RC7	18	—	SEG8	—	—	—	RX/DT	SDI/SDA	—	—	—
RE3	1	—	—	—	—	—	—	—	—	Y(1)	$\overline{MCLR}$/VPP
—	20	—	—	—	—	—	—	—	—	—	VDD
—	8	—	—	—	—	—	—	—	—	—	VSS
—	19	—	—	—	—	—	—	—	—	—	VSS

Note 1: Pull-up enabled only with external $\overline{MCLR}$ configuration.

TABLE 3: PIC16F913/916 28-PIN (QFN) SUMMARY

I/O	Pin	A/D	LCD	Comparators	Timers	CCP	AUSART	SSP	Interrupt	Pull-Up	Basic
RA0	27	AN0	SEG12	C1-	—	—	—	—	—	—	—
RA1	28	AN1	SEG7	C2-	—	—	—	—	—	—	—
RA2	1	AN2/VREF-	COM2	C2+	—	—	—	—	—	—	—
RA3	2	AN3/VREF+	SEG15/ COM3	C1+	—	—	—	—	—	—	—
RA4	3	—	SEG4	C1OUT	T0CKI	—	—	—	—	—	—
RA5	4	AN4	SEG5	C2OUT	—	—	—	$\overline{SS}$	—	—	—
RA6	7	—	—	—	T1OSO	—	—	—	—	—	OSC2/CLKOUT
RA7	6	—	—	—	T1OSI	—	—	—	—	—	OSC1/CLKIN
RB0	18	—	SEG0	—	—	—	—	—	INT	Y	—
RB1	19	—	SEG1	—	—	—	—	—	—	Y	—
RB2	20	—	SEG2	—	—	—	—	—	—	Y	—
RB3	21	—	SEG3	—	—	—	—	—	—	Y	—
RB4	22	—	COM0	—	—	—	—	—	IOC	Y	—
RB5	23	—	COM1	—	—	—	—	—	IOC	Y	—
RB6	24	—	SEG14	—	—	—	—	—	IOC	Y	ICSPCLK/ICDCK
RB7	25	—	SEG13	—	—	—	—	—	IOC	Y	ICSPDAT/ICDDAT
RC0	8	—	VLCD1	—	—	—	—	—	—	—	—
RC1	9	—	VLCD2	—	—	—	—	—	—	—	—
RC2	10	—	VLCD3	—	—	—	—	—	—	—	—
RC3	11	—	SEG6	—	—	—	—	—	—	—	—
RC4	12	—	SEG11	—	$\overline{T1G}$	—	—	SDO	—	—	—
RC5	13	—	SEG10	—	T1CKI	CCP1	—	—	—	—	—
RC6	14	—	SEG9	—	—	—	TX/CK	SCK/SCL	—	—	—
RC7	15	—	SEG8	—	—	—	RX/DT	SDI/SDA	—	—	—
RE3	26	—	—	—	—	—	—	—	—	Y(1)	$\overline{MCLR}$/VPP
—	17	—	—	—	—	—	—	—	—	—	VDD
—	5	—	—	—	—	—	—	—	—	—	VSS
—	16	—	—	—	—	—	—	—	—	—	VSS

Note 1: Pull-up enabled only with external $\overline{MCLR}$ configuration.

Pin Diagrams – PIC16F914/917, 44-Pin

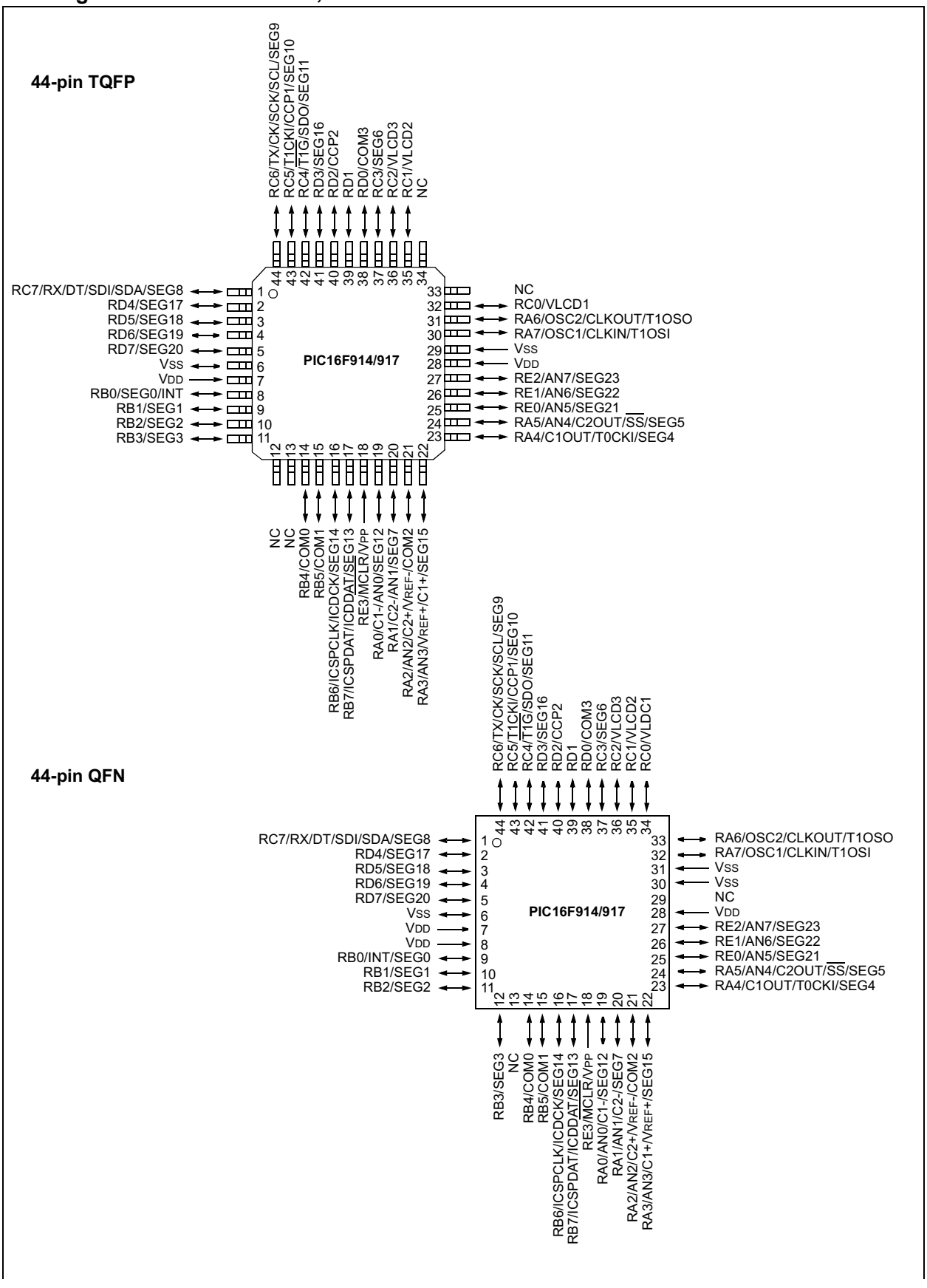


TABLE 4: PIC16F914/917 44-PIN (TQFP) SUMMARY

I/O	Pin	A/D	LCD	Comparators	Timers	CCP	AUSART	SSP	Interrupt	Pull-Up	Basic
RA0	19	AN0	SEG12	C1-	—	—	—	—	—	—	—
RA1	20	AN1	SEG7	C2-	—	—	—	—	—	—	—
RA2	21	AN2/VREF-	COM2	C2+	—	—	—	—	—	—	—
RA3	22	AN3/VREF+	SEG15	C1+	—	—	—	—	—	—	—
RA4	23	—	SEG4	C1OUT	T0CKI	—	—	—	—	—	—
RA5	24	AN4	SEG5	C2OUT	—	—	—	$\overline{SS}$	—	—	—
RA6	31	—	—	—	T1OSO	—	—	—	—	—	OSC2/CLKOUT
RA7	30	—	—	—	T1OSI	—	—	—	—	—	OSC1/CLKIN
RB0	8	—	SEG0	—	—	—	—	—	INT	Y	—
RB1	9	—	SEG1	—	—	—	—	—	—	Y	—
RB2	10	—	SEG2	—	—	—	—	—	—	Y	—
RB3	11	—	SEG3	—	—	—	—	—	—	Y	—
RB4	14	—	COM0	—	—	—	—	—	IOC	Y	—
RB5	15	—	COM1	—	—	—	—	—	IOC	Y	—
RB6	16	—	SEG14	—	—	—	—	—	IOC	Y	ICSPCLK/ICDCK
RB7	17	—	SEG13	—	—	—	—	—	IOC	Y	ICSPDAT/ICDDAT
RC0	32	—	VLCD1	—	—	—	—	—	—	—	—
RC1	35	—	VLCD2	—	—	—	—	—	—	—	—
RC2	36	—	VLCD3	—	—	—	—	—	—	—	—
RC3	37	—	SEG6	—	—	—	—	—	—	—	—
RC4	42	—	SEG11	—	$\overline{T1G}$	—	—	SDO	—	—	—
RC5	43	—	SEG10	—	T1CKI	CCP1	—	—	—	—	—
RC6	44	—	SEG9	—	—	—	TX/CK	SCK/SCL	—	—	—
RC7	1	—	SEG8	—	—	—	RX/DT	SDI/SDA	—	—	—
RD0	38	—	COM3	—	—	—	—	—	—	—	—
RD1	39	—	—	—	—	—	—	—	—	—	—
RD2	40	—	—	—	—	CCP2	—	—	—	—	—
RD3	41	—	SEG16	—	—	—	—	—	—	—	—
RD4	2	—	SEG17	—	—	—	—	—	—	—	—
RD5	3	—	SEG18	—	—	—	—	—	—	—	—
RD6	4	—	SEG19	—	—	—	—	—	—	—	—
RD7	5	—	SEG20	—	—	—	—	—	—	—	—
RE0	25	AN5	SEG21	—	—	—	—	—	—	—	—
RE1	26	AN6	SEG22	—	—	—	—	—	—	—	—
RE2	27	AN7	SEG23	—	—	—	—	—	—	—	—
RE3	18	—	—	—	—	—	—	—	—	Y(1)	$\overline{MCLR}$/VPP
—	7	—	—	—	—	—	—	—	—	—	VDD
—	28	—	—	—	—	—	—	—	—	—	VDD
—	6	—	—	—	—	—	—	—	—	—	VSS
—	29	—	—	—	—	—	—	—	—	—	VSS
—	12	—	—	—	—	—	—	—	—	—	NC
—	13	—	—	—	—	—	—	—	—	—	NC
—	33	—	—	—	—	—	—	—	—	—	NC
—	34	—	—	—	—	—	—	—	—	—	NC

Note 1: Pull-up enabled only with external $\overline{MCLR}$ configuration.

TABLE 5: PIC16F914/917 44-PIN (QFN) SUMMARY

I/O	Pin	A/D	LCD	Comparators	Timers	CCP	AUSART	SSP	Interrupt	Pull-Up	Basic
RA0	19	AN0	SEG12	C1-	—	—	—	—	—	—	—
RA1	20	AN1	SEG7	C2-	—	—	—	—	—	—	—
RA2	21	AN2/VREF-	COM2	C2+	—	—	—	—	—	—	—
RA3	22	AN3/VREF+	SEG15	C1+	—	—	—	—	—	—	—
RA4	23	—	SEG4	C1OUT	T0CKI	—	—	—	—	—	—
RA5	24	AN4	SEG5	C2OUT	—	—	—	$\overline{SS}$	—	—	—
RA6	33	—	—	—	T1OSO	—	—	—	—	—	OSC2/CLKOUT
RA7	32	—	—	—	T1OSI	—	—	—	—	—	OSC1/CLKIN
RB0	9	—	SEG0	—	—	—	—	—	INT	Y	—
RB1	10	—	SEG1	—	—	—	—	—	—	Y	—
RB2	11	—	SEG2	—	—	—	—	—	—	Y	—
RB3	12	—	SEG3	—	—	—	—	—	—	Y	—
RB4	14	—	COM0	—	—	—	—	—	IOC	Y	—
RB5	15	—	COM1	—	—	—	—	—	IOC	Y	—
RB6	16	—	SEG14	—	—	—	—	—	IOC	Y	ICSPCLK/ICDCK
RB7	17	—	SEG13	—	—	—	—	—	IOC	Y	ICSPDAT/ICDDAT
RC0	34	—	VLCD1	—	—	—	—	—	—	—	—
RC1	35	—	VLCD2	—	—	—	—	—	—	—	—
RC2	36	—	VLCD3	—	—	—	—	—	—	—	—
RC3	37	—	SEG6	—	—	—	—	—	—	—	—
RC4	42	—	SEG11	—	$\overline{T1G}$	—	—	SDO	—	—	—
RC5	43	—	SEG10	—	T1CKI	CCP1	—	—	—	—	—
RC6	44	—	SEG9	—	—	—	TX/CK	SCK/SCL	—	—	—
RC7	1	—	SEG8	—	—	—	RX/DT	SDI/SDA	—	—	—
RD0	38	—	COM3	—	—	—	—	—	—	—	—
RD1	39	—	—	—	—	—	—	—	—	—	—
RD2	40	—	—	—	—	CCP2	—	—	—	—	—
RD3	41	—	SEG16	—	—	—	—	—	—	—	—
RD4	2	—	SEG17	—	—	—	—	—	—	—	—
RD5	3	—	SEG18	—	—	—	—	—	—	—	—
RD6	4	—	SEG19	—	—	—	—	—	—	—	—
RD7	5	—	SEG20	—	—	—	—	—	—	—	—
RE0	25	AN5	SEG21	—	—	—	—	—	—	—	—
RE1	26	AN6	SEG22	—	—	—	—	—	—	—	—
RE2	27	AN7	SEG23	—	—	—	—	—	—	—	—
RE3	18	—	—	—	—	—	—	—	—	Y(1)	$\overline{MCLR}$/VPP
—	7	—	—	—	—	—	—	—	—	—	VDD
—	8	—	—	—	—	—	—	—	—	—	VDD
—	28	—	—	—	—	—	—	—	—	—	VDD
—	6	—	—	—	—	—	—	—	—	—	VSS
—	30	—	—	—	—	—	—	—	—	—	VSS
—	13	—	—	—	—	—	—	—	—	—	NC
—	29	—	—	—	—	—	—	—	—	—	NC

Note 1: Pull-up enabled only with external $\overline{MCLR}$ configuration.

Pin Diagram – PIC16F946

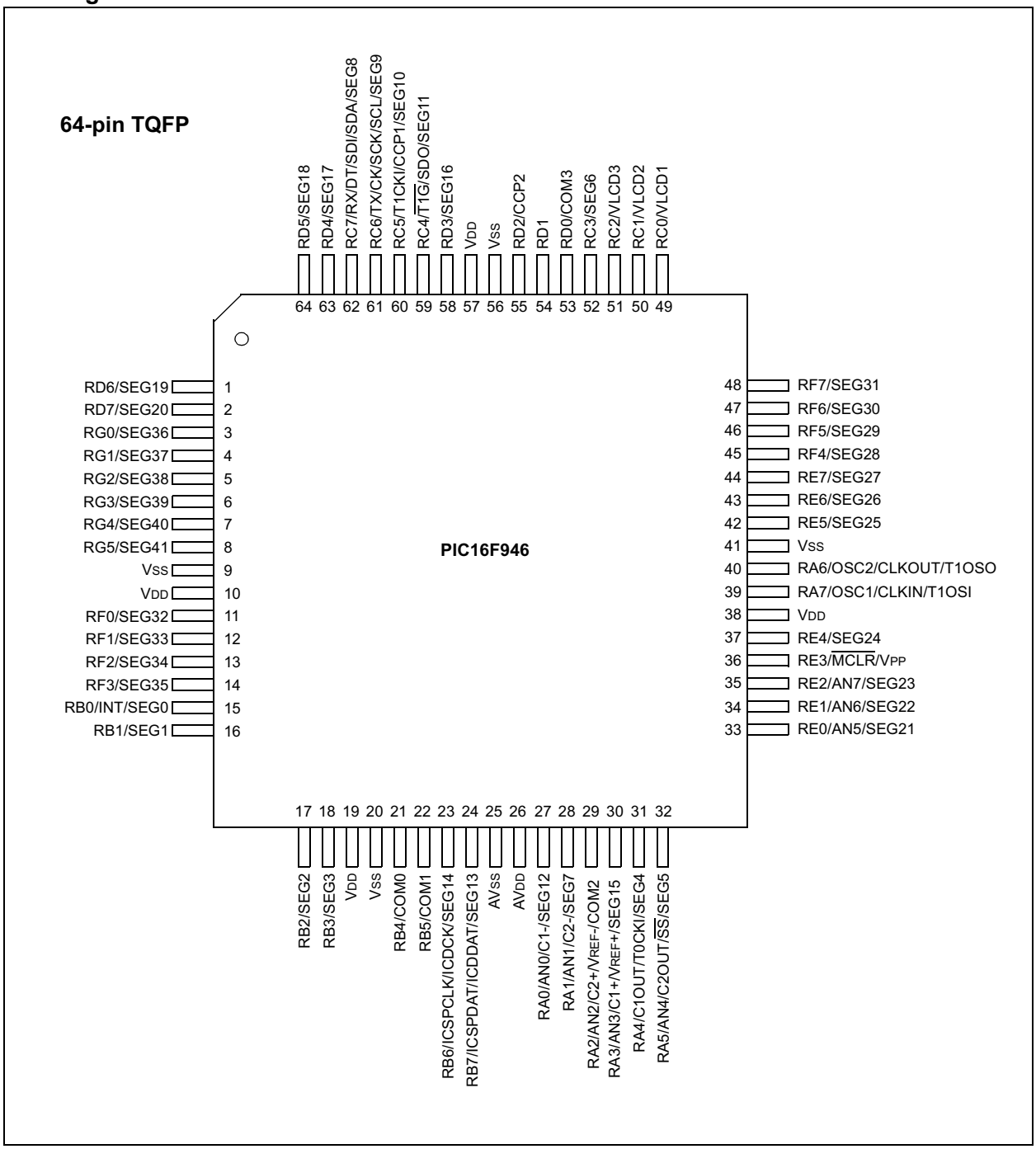


TABLE 6: PIC16F946 64-PIN (TQFP) SUMMARY

I/O	Pin	A/D	LCD	Comparators	Timers	CCP	AUSART	SSP	Interrupt	Pull-Up	Basic
RA0	27	AN0	SEG12	C1-	—	—	—	—	—	—	—
RA1	28	AN1	SEG7	C2-	—	—	—	—	—	—	—
RA2	29	AN2/VREF-	COM2	C2+	—	—	—	—	—	—	—
RA3	30	AN3/VREF+	SEG15	C1+	—	—	—	—	—	—	—
RA4	31	—	SEG4	C1OUT	T0CKI	—	—	—	—	—	—
RA5	32	AN4	—	C2OUT	—	—	—	$\overline{SS}$	—	—	—
RA6	40	SEG5	—	—	T1OSO	—	—	—	—	—	OSC2/CLKOUT
RA7	39	—	—	—	T1OSI	—	—	—	—	—	OSC1/CLKIN
RB0	15	—	SEG0	—	—	—	—	—	INT	Y	—
RB1	16	—	SEG1	—	—	—	—	—	—	Y	—
RB2	17	—	SEG2	—	—	—	—	—	—	Y	—
RB3	18	—	SEG3	—	—	—	—	—	—	Y	—
RB4	21	—	COM0	—	—	—	—	—	IOC	Y	—
RB5	22	—	COM1	—	—	—	—	—	IOC	Y	—
RB6	23	—	SEG14	—	—	—	—	—	IOC	Y	ICSPCLK/ICDCK
RB7	24	—	SEG13	—	—	—	—	—	IOC	Y	ICSPDAT/ICDDAT
RC0	49	—	VLCD1	—	—	—	—	—	—	—	—
RC1	50	—	VLCD2	—	—	—	—	—	—	—	—
RC2	51	—	VLCD3	—	—	—	—	—	—	—	—
RC3	52	—	SEG6	—	—	—	—	—	—	—	—
RC4	59	—	SEG11	—	$\overline{T1G}$	—	—	SDO	—	—	—
RC5	60	—	SEG10	—	T1CKI	CCP1	—	—	—	—	—
RC6	61	—	SEG9	—	—	—	TX/CK	SCK/SCL	—	—	—
RC7	62	—	SEG8	—	—	—	RX/DT	SDI/SDA	—	—	—
RD0	53	—	COM3	—	—	—	—	—	—	—	—
RD1	54	—	—	—	—	—	—	—	—	—	—
RD2	55	—	—	—	—	CCP2	—	—	—	—	—
RD3	58	—	SEG16	—	—	—	—	—	—	—	—
RD4	63	—	SEG17	—	—	—	—	—	—	—	—
RD5	64	—	SEG18	—	—	—	—	—	—	—	—
RD6	1	—	SEG19	—	—	—	—	—	—	—	—
RD7	2	—	SEG20	—	—	—	—	—	—	—	—
RE0	33	AN5	SEG21	—	—	—	—	—	—	—	—
RE1	34	AN6	SEG22	—	—	—	—	—	—	—	—
RE2	35	AN7	SEG23	—	—	—	—	—	—	—	—
RE3	36	—	—	—	—	—	—	—	—	Y(1)	$\overline{MCLR}$/VPP
RE4	37	—	SEG24	—	—	—	—	—	—	—	—
RE5	42	—	SEG25	—	—	—	—	—	—	—	—
RE6	43	—	SEG26	—	—	—	—	—	—	—	—
RE7	44	—	SEG27	—	—	—	—	—	—	—	—
RF0	11	—	SEG32	—	—	—	—	—	—	—	—
RF1	12	—	SEG33	—	—	—	—	—	—	—	—
RF2	13	—	SEG34	—	—	—	—	—	—	—	—

Note 1: Pull-up enabled only with external $\overline{MCLR}$ configuration.

TABLE 6: PIC16F946 64-PIN (TQFP) SUMMARY (CONTINUED)

I/O	Pin	A/D	LCD	Comparators	Timers	CCP	AUSART	SSP	Interrupt	Pull-Up	Basic
RF3	14	—	SEG35	—	—	—	—	—	—	—	—
RF4	45	—	SEG28	—	—	—	—	—	—	—	—
RF5	46	—	SEG29	—	—	—	—	—	—	—	—
RF6	47	—	SEG30	—	—	—	—	—	—	—	—
RF7	48	—	SEG31	—	—	—	—	—	—	—	—
RG0	3	—	SEG36	—	—	—	—	—	—	—	—
RG1	4	—	SEG37	—	—	—	—	—	—	—	—
RG2	5	—	SEG38	—	—	—	—	—	—	—	—
RG3	6	—	SEG39	—	—	—	—	—	—	—	—
RG4	7	—	SEG40	—	—	—	—	—	—	—	—
RG5	8	—	SEG41	—	—	—	—	—	—	—	—
—	26	—	—	—	—	—	—	—	—	—	AV_{DD}
—	25	—	—	—	—	—	—	—	—	—	AV_{SS}
—	10	—	—	—	—	—	—	—	—	—	V_{DD}
—	19	—	—	—	—	—	—	—	—	—	V_{DD}
—	38	—	—	—	—	—	—	—	—	—	V_{DD}
—	57	—	—	—	—	—	—	—	—	—	V_{DD}
—	9	—	—	—	—	—	—	—	—	—	V_{SS}
—	20	—	—	—	—	—	—	—	—	—	V_{SS}
—	41	—	—	—	—	—	—	—	—	—	V_{SS}
—	56	—	—	—	—	—	—	—	—	—	V_{SS}

Note 1: Pull-up enabled only with external $\overline{MCLR}$ configuration.

Table of Contents

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** or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. 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., DS30000A is version A of document DS30000).

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.

NOTES:

1.0 DEVICE OVERVIEW

The PIC16F91X/946 devices are covered by this data sheet. They are available in 28/40/44/64-pin packages. Figure 1-1 shows a block diagram of the PIC16F913/916 device, Figure 1-2 shows a block diagram of the PIC16F914/917 device, and Figure 1-3 shows a block diagram of the PIC16F946 device. Table 1-1 shows the pinout descriptions.

FIGURE 1-1: PIC16F913/916 BLOCK DIAGRAM

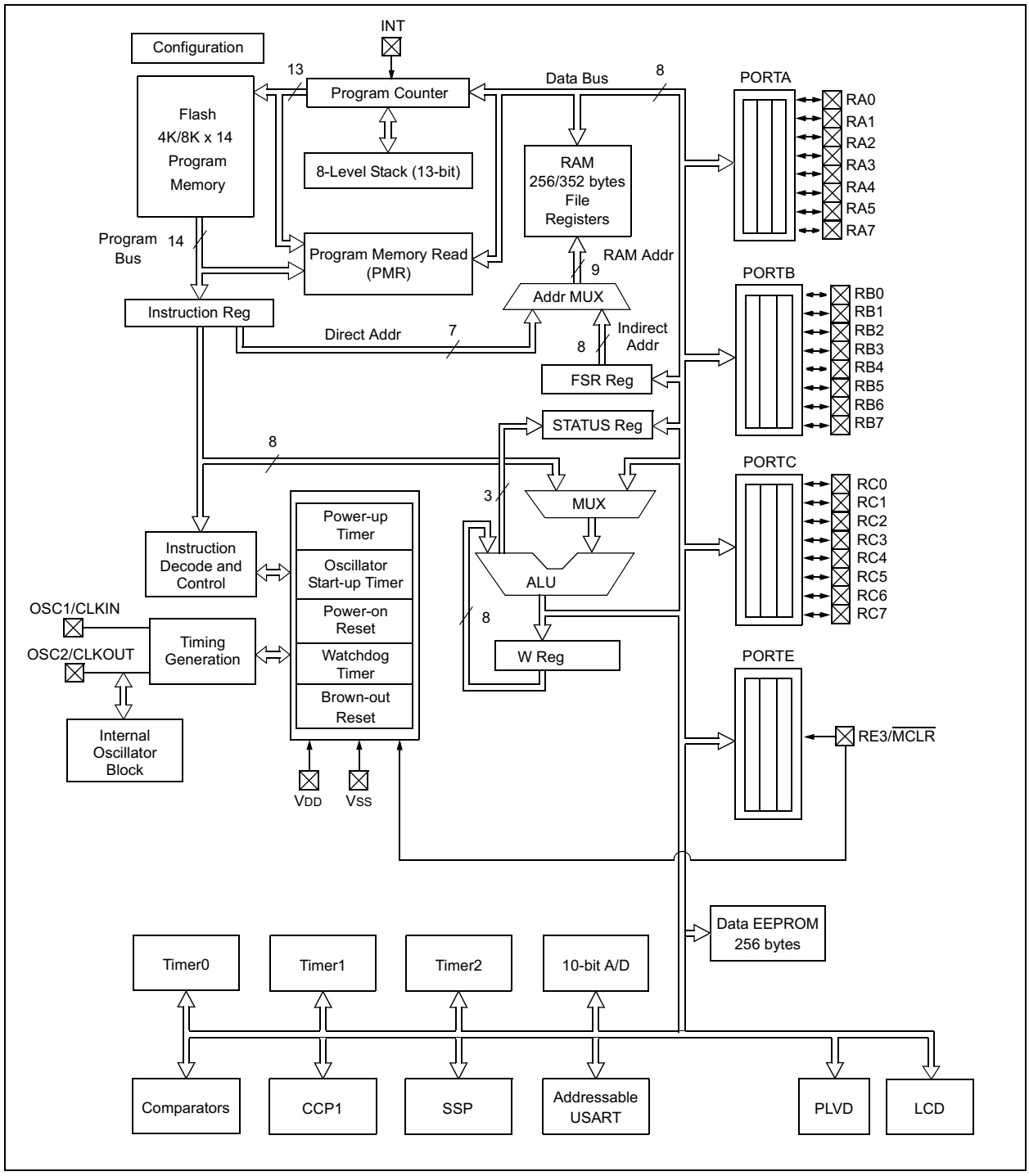


FIGURE 1-2: PIC16F914/917 BLOCK DIAGRAM

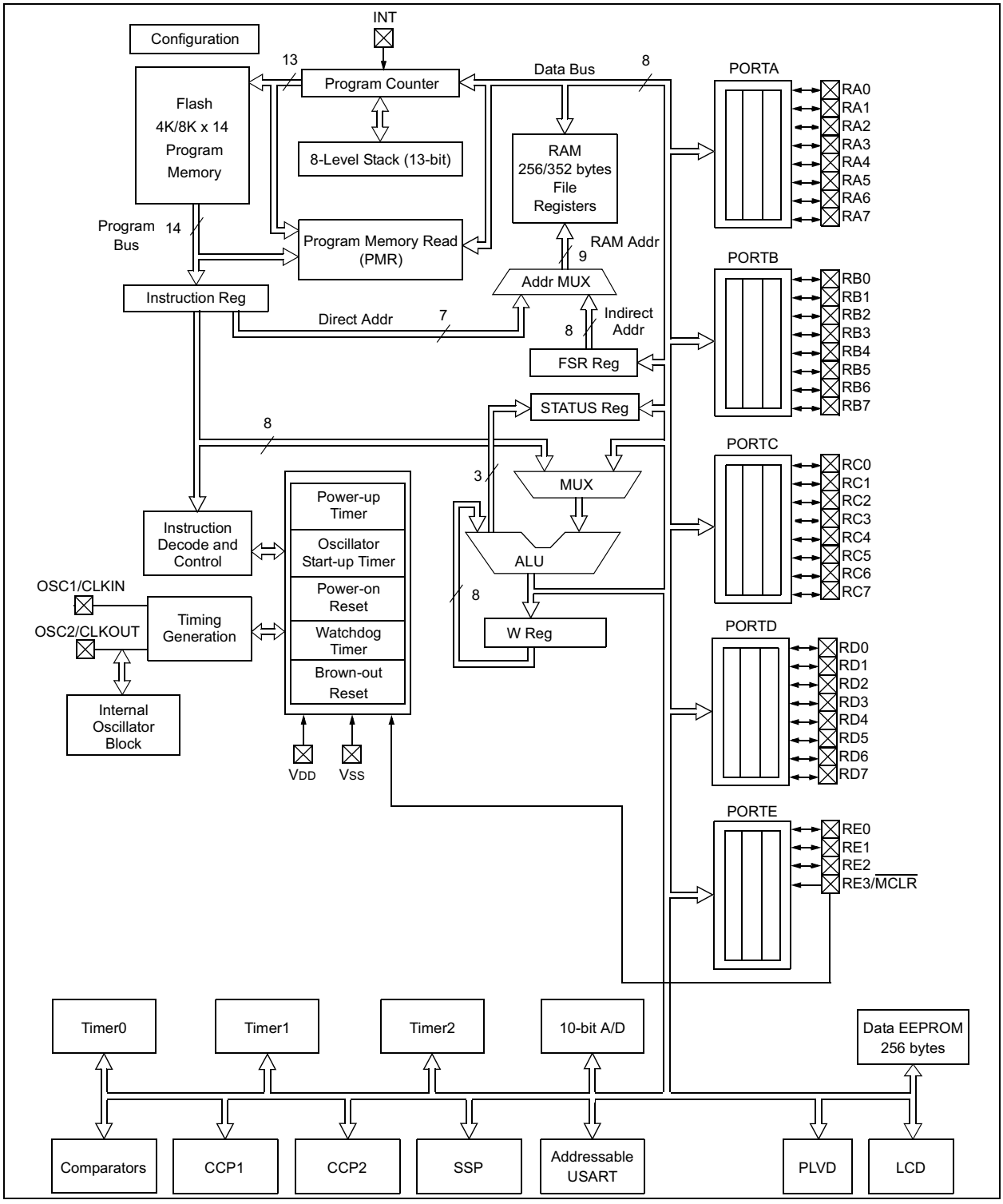


FIGURE 1-3: PIC16F946 BLOCK DIAGRAM

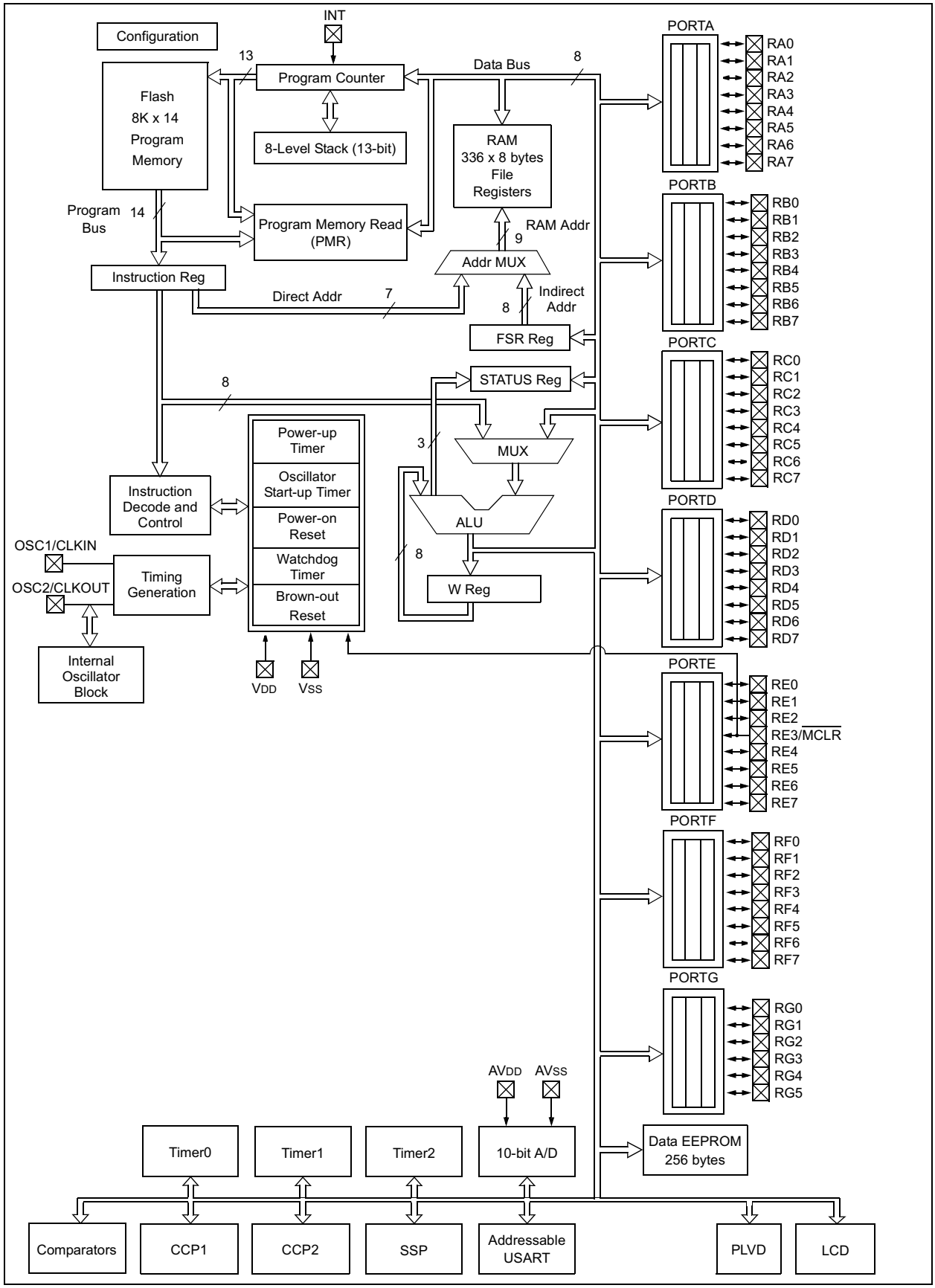


TABLE 1-1: PIC16F91X/946 PINOUT DESCRIPTIONS

Name	Function	Input Type	Output Type	Description
RA0/AN0/C1-/SEG12	RA0	TTL	CMOS	General purpose I/O.
	AN0	AN	—	Analog input Channel 0.
	C1-	AN	—	Comparator 1 negative input.
	SEG12	—	AN	LCD analog output.
RA1/AN1/C2-/SEG7	RA1	TTL	CMOS	General purpose I/O.
	AN1	AN	—	Analog input Channel 1.
	C2-	AN	—	Comparator 2 negative input.
	SEG7	—	AN	LCD analog output.
RA2/AN2/C2+/VREF-/COM2	RA2	TTL	CMOS	General purpose I/O.
	AN2	AN	—	Analog input Channel 2.
	C2+	AN	—	Comparator 2 positive input.
	VREF-	AN	—	External A/D Voltage Reference – negative.
	COM2	—	AN	LCD analog output.
RA3/AN3/C1+/VREF+/COM3(1)/ SEG15	RA3	TTL	CMOS	General purpose I/O.
	AN3	AN	—	Analog input Channel 3.
	C1+	AN	—	Comparator 1 positive input.
	VREF+	AN	—	External A/D Voltage Reference – positive.
	COM3(1)	—	AN	LCD analog output.
	SEG15	—	AN	LCD analog output.
RA4/C1OUT/T0CKI/SEG4	RA4	TTL	CMOS	General purpose I/O.
	C1OUT	—	CMOS	Comparator 1 output.
	T0CKI	ST	—	Timer0 clock input.
	SEG4	—	AN	LCD analog output.
RA5/AN4/C2OUT/$\overline{SS}$/SEG5	RA5	TTL	CMOS	General purpose I/O.
	AN4	AN	—	Analog input Channel 4.
	C2OUT	—	CMOS	Comparator 2 output.
	$\overline{SS}$	TTL	—	Slave select input.
	SEG5	—	AN	LCD analog output.
RA6/OSC2/CLKOUT/T1OSO	RA6	TTL	CMOS	General purpose I/O.
	OSC2	—	XTAL	Crystal/Resonator.
	CLKOUT	—	CMOS	TOSC/4 reference clock.
	T1OSO	—	XTAL	Timer1 oscillator output.
RA7/OSC1/CLKIN/T1OSI	RA7	TTL	CMOS	General purpose I/O.
	OSC1	XTAL	—	Crystal/Resonator.
	CLKIN	ST	—	Clock input.
	T1OSI	XTAL	—	Timer1 oscillator input.
RB0/INT/SEG0	RB0	TTL	CMOS	General purpose I/O. Individually enabled pull-up.
	INT	ST	—	External interrupt pin.
	SEG0	—	AN	LCD analog output.

Legend: AN = Analog input or output; CMOS = CMOS compatible input or output; OD = Open Drain
TTL = TTL compatible input; ST = Schmitt Trigger input with CMOS levels; P = Power
HV = High Voltage; XTAL = Crystal

Note 1: COM3 is available on RA3 for the PIC16F913/916 and on RD0 for the PIC16F914/917 and PIC16F946.
2: Pins available on PIC16F914/917 and PIC16F946 only.
3: Pins available on PIC16F946 only.
4: I^2C Schmitt trigger inputs have special input levels.

TABLE 1-1: PIC16F91X/946 PINOUT DESCRIPTIONS (CONTINUED)

Name	Function	Input Type	Output Type	Description
RB1/SEG1	RB1	TTL	CMOS	General purpose I/O. Individually enabled pull-up.
	SEG1	—	AN	LCD analog output.
RB2/SEG2	RB2	TTL	CMOS	General purpose I/O. Individually enabled pull-up.
	SEG2	—	AN	LCD analog output.
RB3/SEG3	RB3	TTL	CMOS	General purpose I/O. Individually enabled pull-up.
	SEG3	—	AN	LCD analog output.
RB4/COM0	RB4	TTL	CMOS	General purpose I/O. Individually controlled interrupt-on-change. Individually enabled pull-up.
	COM0	—	AN	LCD analog output.
RB5/COM1	RB5	TTL	CMOS	General purpose I/O. Individually controlled interrupt-on-change. Individually enabled pull-up.
	COM1	—	AN	LCD analog output.
RB6/ICSPCLK/ICDCK/SEG14	RB6	TTL	CMOS	General purpose I/O. Individually controlled interrupt-on-change. Individually enabled pull-up.
	ICSPCLK	ST	—	ICSP™ clock.
	ICDCK	ST	—	ICD clock.
	SEG14	—	AN	LCD analog output.
RB7/ICSPDAT/ICDDAT/SEG13	RB7	TTL	CMOS	General purpose I/O. Individually controlled interrupt-on-change. Individually enabled pull-up.
	ICSPDAT	ST	CMOS	ICSP Data I/O.
	ICDDAT	ST	CMOS	ICD Data I/O.
	SEG13	—	AN	LCD analog output.
RC0/VLCD1	RC0	ST	CMOS	General purpose I/O.
	VLCD1	AN	—	LCD analog input.
RC1/VLCD2	RC1	ST	CMOS	General purpose I/O.
	VLCD2	AN	—	LCD analog input.
RC2/VLCD3	RC2	ST	CMOS	General purpose I/O.
	VLCD3	AN	—	LCD analog input.
RC3/SEG6	RC3	ST	CMOS	General purpose I/O.
	SEG6	—	AN	LCD analog output.
RC4/$\overline{T1G}$/SDO/SEG11	RC4	ST	CMOS	General purpose I/O.
	$\overline{T1G}$	ST	—	Timer1 gate input.
	SDO	—	CMOS	Serial data output.
	SEG11	—	AN	LCD analog output.
RC5/T1CKI/CCP1/SEG10	RC5	ST	CMOS	General purpose I/O.
	T1CKI	ST	—	Timer1 clock input.
	CCP1	ST	CMOS	Capture 1 input/Compare 1 output/PWM 1 output.
	SEG10	—	AN	LCD analog output.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open Drain
TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels P = Power
HV = High Voltage XTAL = Crystal

Note 1: COM3 is available on RA3 for the PIC16F913/916 and on RD0 for the PIC16F914/917 and PIC16F946.
2: Pins available on PIC16F914/917 and PIC16F946 only.
3: Pins available on PIC16F946 only.
4: I^2C Schmitt trigger inputs have special input levels.

TABLE 1-1: PIC16F91X/946 PINOUT DESCRIPTIONS (CONTINUED)

Name	Function	Input Type	Output Type	Description
RC6/TX/CK/SCK/SCL/SEG9	RC6	ST	CMOS	General purpose I/O.
	TX	—	CMOS	USART asynchronous serial transmit.
	CK	ST	CMOS	USART synchronous serial clock.
	SCK	ST	CMOS	SPI clock.
	SCL	ST(4)	OD	I^2C™ clock.
	SEG9	—	AN	LCD analog output.
RC7/RX/DT/SDI/SDA/SEG8	RC7	ST	CMOS	General purpose I/O.
	RX	ST	—	USART asynchronous serial receive.
	DT	ST	CMOS	USART synchronous serial data.
	SDI	ST	CMOS	SPI data input.
	SDA	ST(4)	OD	I^2C™ data.
	SEG8	—	AN	LCD analog output.
RD0/COM3(1, 2)	RD0	ST	CMOS	General purpose I/O.
	COM3	—	AN	LCD analog output.
RD1(2)	RD1	ST	CMOS	General purpose I/O.
RD2/CCP2(2)	RD2	ST	CMOS	General purpose I/O.
	CCP2	ST	CMOS	Capture 2 input/Compare 2 output/PWM 2 output.
RD3/SEG16(2)	RD3	ST	CMOS	General purpose I/O.
	SEG16	—	AN	LCD analog output.
RD4/SEG17(2)	RD4	ST	CMOS	General purpose I/O.
	SEG17	—	AN	LCD analog output.
RD5/SEG18(2)	RD5	ST	CMOS	General purpose I/O.
	SEG18	—	AN	LCD analog output.
RD6/SEG19(2)	RD6	ST	CMOS	General purpose I/O.
	SEG19	—	AN	LCD analog output.
RD7/SEG20(2)	RD7	ST	CMOS	General purpose I/O.
	SEG20	—	AN	LCD analog output.
RE0/AN5/SEG21(2)	RE0	ST	CMOS	General purpose I/O.
	AN5	AN	—	Analog input Channel 5.
	SEG21	—	AN	LCD analog output.
RE1/AN6/SEG22(2)	RE1	ST	CMOS	General purpose I/O.
	AN6	AN	—	Analog input Channel 6.
	SEG22	—	AN	LCD analog output.
RE2/AN7/SEG23(2)	RE2	ST	CMOS	General purpose I/O.
	AN7	AN	—	Analog input Channel 7.
	SEG23	—	AN	LCD analog output.
RE3/$\overline{MCLR}$/V_{PP}	RE3	ST	—	Digital input only.
	$\overline{MCLR}$	ST	—	Master Clear with internal pull-up.
	V_{PP}	HV	—	Programming voltage.

Legend: AN = Analog input or output; CMOS = CMOS compatible input or output; OD = Open Drain
TTL = TTL compatible input; ST = Schmitt Trigger input with CMOS levels; P = Power
HV = High Voltage; XTAL = Crystal

Note 1: COM3 is available on RA3 for the PIC16F913/916 and on RD0 for the PIC16F914/917 and PIC16F946.
2: Pins available on PIC16F914/917 and PIC16F946 only.
3: Pins available on PIC16F946 only.
4: I^2C Schmitt trigger inputs have special input levels.

TABLE 1-1: PIC16F91X/946 PINOUT DESCRIPTIONS (CONTINUED)

Name	Function	Input Type	Output Type	Description
RE4/SEG24(3)	RE4	ST	CMOS	General purpose I/O.
	SEG24	—	AN	LCD analog output.
RE5/SEG25(3)	RE5	ST	CMOS	General purpose I/O.
	SEG25	—	AN	LCD analog output.
RE6/SEG26(3)	RE6	ST	CMOS	General purpose I/O.
	SEG26	—	AN	LCD analog output.
RE7/SEG27(3)	RE7	ST	CMOS	General purpose I/O.
	SEG27	—	AN	LCD analog output.
RF0/SEG32(3)	RF0	ST	CMOS	General purpose I/O.
	SEG32	—	AN	LCD analog output.
RF1/SEG33(3)	RF1	ST	CMOS	General purpose I/O.
	SEG33	—	AN	LCD analog output.
RF2/SEG34(3)	RF2	ST	CMOS	General purpose I/O.
	SEG34	—	AN	LCD analog output.
RF3/SEG35(3)	RF3	ST	CMOS	General purpose I/O.
	SEG35	—	AN	LCD analog output.
RF4/SEG28(3)	RF4	ST	CMOS	General purpose I/O.
	SEG28	—	AN	LCD analog output.
RF5/SEG29(3)	RF5	ST	CMOS	General purpose I/O.
	SEG29	—	AN	LCD analog output.
RF6/SEG30(3)	RF6	ST	CMOS	General purpose I/O.
	SEG30	—	AN	LCD analog output.
RF7/SEG31(3)	RF7	ST	CMOS	General purpose I/O.
	SEG31	—	AN	LCD analog output.
RG0/SEG36(3)	RG0	ST	CMOS	General purpose I/O.
	SEG36	—	AN	LCD analog output.
RG1/SEG37(3)	RG1	ST	CMOS	General purpose I/O.
	SEG37	—	AN	LCD analog output.
RG2/SEG38(3)	RG2	ST	CMOS	General purpose I/O.
	SEG38	—	AN	LCD analog output.
RG3/SEG39(3)	RG3	ST	CMOS	General purpose I/O.
	SEG39	—	AN	LCD analog output.
RG4/SEG40(3)	RG4	ST	CMOS	General purpose I/O.
	SEG10	—	AN	LCD analog output.
RG5/SEG41(3)	RG5	ST	CMOS	General purpose I/O.
	SEG41	—	AN	LCD analog output.
AVDD(3)	AVDD	P	—	Analog power supply for microcontroller.
AVSS(3)	AVSS	P	—	Analog ground reference for microcontroller.
VDD	VDD	P	—	Power supply for microcontroller.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open Drain
TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels P = Power
HV = High Voltage XTAL = Crystal

Note 1: COM3 is available on RA3 for the PIC16F913/916 and on RD0 for the PIC16F914/917 and PIC16F946.
2: Pins available on PIC16F914/917 and PIC16F946 only.
3: Pins available on PIC16F946 only.
4: I^2C Schmitt trigger inputs have special input levels.

TABLE 1-1: PIC16F91X/946 PINOUT DESCRIPTIONS (CONTINUED)

Name	Function	Input Type	Output Type	Description
VSS	VSS	P	—	Ground reference for microcontroller.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open Drain
TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels P = Power
HV = High Voltage XTAL = Crystal

Note 1: COM3 is available on RA3 for the PIC16F913/916 and on RD0 for the PIC16F914/917 and PIC16F946.
2: Pins available on PIC16F914/917 and PIC16F946 only.
3: Pins available on PIC16F946 only.
4: I^2C Schmitt trigger inputs have special input levels.

2.0 MEMORY ORGANIZATION

2.1 Program Memory Organization

The PIC16F91X/946 has a 13-bit program counter capable of addressing a 4K x 14 program memory space for the PIC16F913/914 (0000h-0FFFh) and an 8K x 14 program memory space for the PIC16F916/917 and PIC16F946 (0000h-1FFFh). Accessing a location above the memory boundaries for the PIC16F913 and PIC16F914 will cause a wrap around within the first 4K x 14 space. The Reset vector is at 0000h and the interrupt vector is at 0004h.

FIGURE 2-1: PROGRAM MEMORY MAP AND STACK FOR THE PIC16F913/914

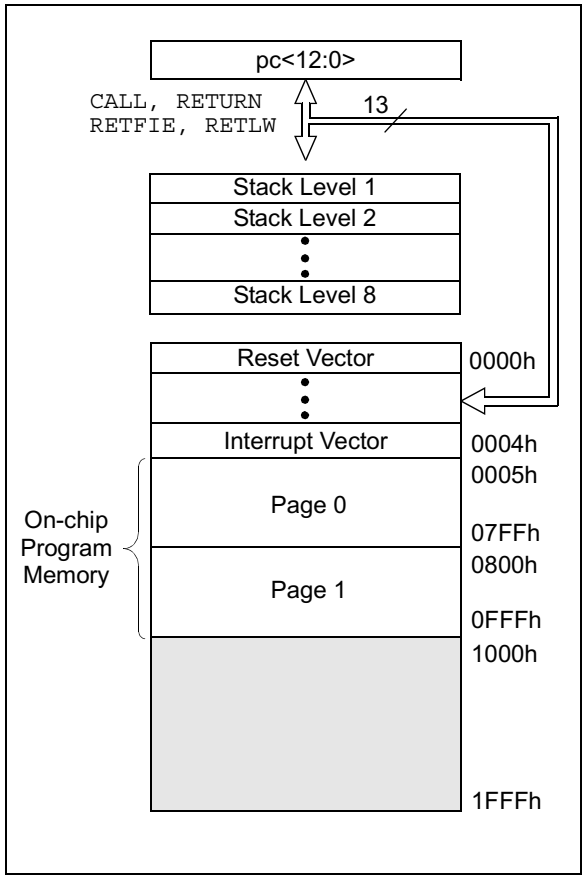
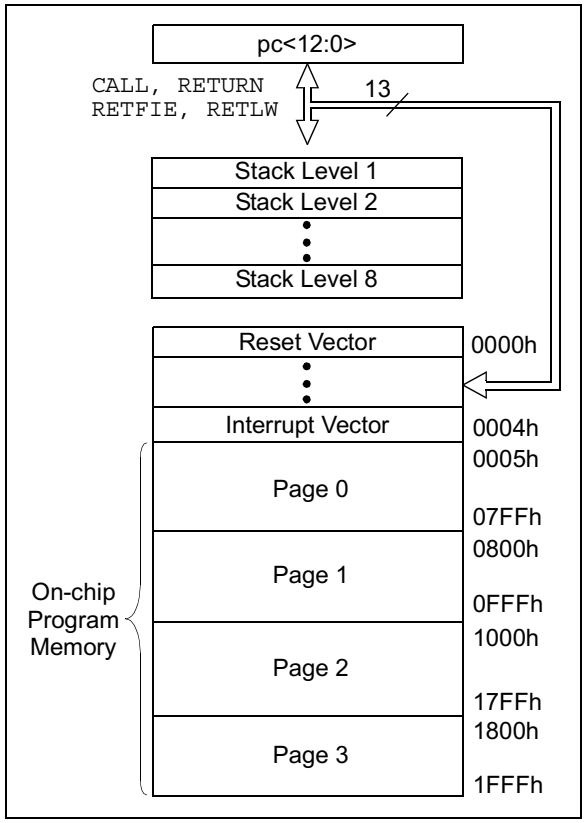


FIGURE 2-2: PROGRAM MEMORY MAP AND STACK FOR THE PIC16F916/917/PIC16F946



2.2 Data Memory Organization

The data memory is partitioned into multiple banks which contain the General Purpose Registers (GPRs) and the Special Function Registers (SFRs). Bits RP0 and RP1 are bank select bits.

RP1	RP0		
0	0	→	Bank 0 is selected
0	1	→	Bank 1 is selected
1	0	→	Bank 2 is selected
1	1	→	Bank 3 is selected

Each bank extends up to 7Fh (128 bytes). The lower locations of each bank are reserved for the Special Function Registers. Above the Special Function Registers are the General Purpose Registers, implemented as static RAM. All implemented banks contain Special Function Registers. Some frequently used Special Function Registers from one bank are mirrored in another bank for code reduction and quicker access.

2.2.1 GENERAL PURPOSE REGISTER FILE

The register file is organized as 256 x 8 bits in the PIC16F913/914, 352 x 8 bits in the PIC16F916/917 and 336 x 8 bits in the PIC16F946. Each register is accessed either directly or indirectly through the File Select Register (FSR) (see **Section 2.5 "Indirect Addressing, INDF and FSR Registers"**).

2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and peripheral functions for controlling the desired operation of the device (see Tables 2-1, 2-2, 2-3 and 2-4). These registers are static RAM.

The Special Function Registers can be classified into two sets: core and peripheral. The Special Function Registers associated with the "core" are described in this section. Those related to the operation of the peripheral features are described in the section of that peripheral feature.

FIGURE 2-3: PIC16F913/916 SPECIAL FUNCTION REGISTERS

	File Address		File Address		File Address		File Address
Indirect addr. (1)	00h	Indirect addr. (1)	80h	Indirect addr. (1)	100h	Indirect addr. (1)	180h
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h	WDTCON	105h		185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
PORTC	07h	TRISC	87h	LCDCON	107h		187h
	08h		88h	LCDPS	108h		188h
PORTE	09h	TRISE	89h	LVDCON	109h		189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	EEDATL	10Ch	EECON1	18Ch
PIR2	0Dh	PIE2	8Dh	EEADRL	10Dh	EECON2(1)	18Dh
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh	Reserved	18Eh
TMR1H	0Fh	OSCCON	8Fh	EEADRH	10Fh	Reserved	18Fh
T1CON	10h	OSCTUNE	90h	LCDDATA0	110h		190h
TMR2	11h	ANSEL	91h	LCDDATA1	111h		
T2CON	12h	PR2	92h		112h		
SSPBUF	13h	SSPADD	93h	LCDDATA3	113h		
SSPCON	14h	SSPSTAT	94h	LCDDATA4	114h		
CCPR1L	15h	WPUB	95h		115h		
CCPR1H	16h	IOCB	96h	LCDDATA6	116h		
CCP1CON	17h	CMCON1	97h	LCDDATA7	117h		
RCSTA	18h	TXSTA	98h		118h		
TXREG	19h	SPBRG	99h	LCDDATA9	119h	General Purpose Register(2)	
RCREG	1Ah		9Ah	LCDDATA10	11Ah		
	1Bh		9Bh		11Bh	96 Bytes	
	1Ch	CMCON0	9Ch	LCDSE0	11Ch		
	1Dh	VRCON	9Dh	LCDSE1	11Dh		
ADRESH	1Eh	ADRESL	9Eh		11Eh		
ADCON0	1Fh	ADCON1	9Fh		11Fh		
General Purpose Register 96 Bytes	20h	General Purpose Register 80 Bytes	A0h	General Purpose Register 80 Bytes	120h		
			EFh		16Fh		1EFh
	7Fh	accesses 70h-7Fh	F0h FFh	accesses 70h-7Fh	170h 17Fh	accesses 70h-7Fh	1F0h 1FFh
Bank 0		Bank 1		Bank 2		Bank 3	

Unimplemented data memory locations, read as '0'.

Note 1: Not a physical register.

2: On the PIC16F913, unimplemented data memory locations, read as '0'.

FIGURE 2-4: PIC16F914/917 SPECIAL FUNCTION REGISTERS

	File Address		File Address		File Address		File Address
Indirect addr. (1)	00h	Indirect addr. (1)	80h	Indirect addr. (1)	100h	Indirect addr. (1)	180h
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h	WDTCON	105h		185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
PORTC	07h	TRISC	87h	LCDCON	107h		187h
PORTD	08h	TRISD	88h	LCDPS	108h		188h
PORTE	09h	TRISE	89h	LVDCON	109h		189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	EEDATL	10Ch	EECON1	18Ch
PIR2	0Dh	PIE2	8Dh	EEADRL	10Dh	EECON2(1)	18Dh
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh	Reserved	18Eh
TMR1H	0Fh	OSCCON	8Fh	EEADRH	10Fh	Reserved	18Fh
T1CON	10h	OSCTUNE	90h	LCDDATA0	110h		190h
TMR2	11h	ANSEL	91h	LCDDATA1	111h		
T2CON	12h	PR2	92h	LCDDATA2	112h		
SSPBUF	13h	SSPADD	93h	LCDDATA3	113h		
SSPCON	14h	SSPSTAT	94h	LCDDATA4	114h		
CCPR1L	15h	WPUB	95h	LCDDATA5	115h		
CCPR1H	16h	IOCB	96h	LCDDATA6	116h		
CCP1CON	17h	CMCON1	97h	LCDDATA7	117h		
RCSTA	18h	TXSTA	98h	LCDDATA8	118h		
TXREG	19h	SPBRG	99h	LCDDATA9	119h	General Purpose Register(2)	
RCREG	1Ah		9Ah	LCDDATA10	11Ah		
CCPR2L	1Bh		9Bh	LCDDATA11	11Bh		
CCPR2H	1Ch	CMCON0	9Ch	LCDSE0	11Ch		
CCP2CON	1Dh	VRCON	9Dh	LCDSE1	11Dh	96 Bytes	
ADRESH	1Eh	ADRESL	9Eh	LCDSE2	11Eh		
ADCON0	1Fh	ADCON1	9Fh		11Fh		
	20h		A0h		120h		
General Purpose Register 96 Bytes		General Purpose Register 80 Bytes		General Purpose Register 80 Bytes			
			EFh		16Fh		1EFh
		accesses 70h-7Fh	F0h	accesses 70h-7Fh	170h	accesses 70h-7Fh	1F0h
	7Fh		FFh		17Fh		1FFh
Bank 0		Bank 1		Bank 2		Bank 3	

Unimplemented data memory locations, read as '0'.

Note 1: Not a physical register.

2: On the PIC16F914, unimplemented data memory locations, read as '0'.

FIGURE 2-5: PIC16F946 SPECIAL FUNCTION REGISTERS

	File Address		File Address		File Address		File Address
Indirect addr. (1)	00h	Indirect addr. (1)	80h	Indirect addr. (1)	100h	Indirect addr. (1)	180h
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h	WDTCON	105h	TRISF	185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
PORTC	07h	TRISC	87h	LCDCON	107h	TRISG	187h
PORTD	08h	TRISD	88h	LCDPS	108h	PORTF	188h
PORTE	09h	TRISE	89h	LVDCON	109h	PORTG	189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	EEDATL	10Ch	EECON1	18Ch
PIR2	0Dh	PIE2	8Dh	EEADRL	10Dh	EECON2(1)	18Dh
TMR1L	0Eh	PCON	8Eh	EEDATH	10Eh	Reserved	18Eh
TMR1H	0Fh	OSCCON	8Fh	EEADRH	10Fh	Reserved	18Fh
T1CON	10h	OSCTUNE	90h	LCDDATA0	110h	LCDDATA12	190h
TMR2	11h	ANSEL	91h	LCDDATA1	111h	LCDDATA13	191h
T2CON	12h	PR2	92h	LCDDATA2	112h	LCDDATA14	192h
SSPBUF	13h	SSPADD	93h	LCDDATA3	113h	LCDDATA15	193h
SSPCON	14h	SSPSTAT	94h	LCDDATA4	114h	LCDDATA16	194h
CCPR1L	15h	WPUB	95h	LCDDATA5	115h	LCDDATA17	195h
CCPR1H	16h	IOCB	96h	LCDDATA6	116h	LCDDATA18	196h
CCP1CON	17h	CMCON1	97h	LCDDATA7	117h	LCDDATA19	197h
RCSTA	18h	TXSTA	98h	LCDDATA8	118h	LCDDATA20	198h
TXREG	19h	SPBRG	99h	LCDDATA9	119h	LCDDATA21	199h
RCREG	1Ah		9Ah	LCDDATA10	11Ah	LCDDATA22	19Ah
CCPR2L	1Bh		9Bh	LCDDATA11	11Bh	LCDDATA23	19Bh
CCPR2H	1Ch	CMCON0	9Ch	LCDSE0	11Ch	LCDSE3	19Ch
CCP2CON	1Dh	VRCON	9Dh	LCDSE1	11Dh	LCDSE4	19Dh
ADRESH	1Eh	ADRESL	9Eh	LCDSE2	11Eh	LCDSE5	19Eh
ADCON0	1Fh	ADCON1	9Fh		11Fh		19Fh
General Purpose Register 96 Bytes	20h – 7Fh	General Purpose Register 80 Bytes	A0h – EFh	General Purpose Register 80 Bytes	120h – 16Fh	General Purpose Register 80 Bytes	1A0h – 1EFh
accesses 70h-7Fh		accesses 70h-7Fh	F0h – FFh	accesses 70h-7Fh	170h – 17Fh	accesses 70h-7Fh	1F0h – 1FFh
Bank 0		Bank 1		Bank 2		Bank 3	

Unimplemented data memory locations, read as '0'.

Note 1: Not a physical register.

TABLE 2-1: PIC16F91X/946 SPECIAL FUNCTION REGISTERS SUMMARY BANK 0

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Page
Bank 0											
00h	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	41,226
01h	TMR0	Timer0 Module Register								xxxx xxxx	99,226
02h	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	40,226
03h	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	32,226
04h	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	41,226
05h	PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx xxxx	44,226
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	54,226
07h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	62,226
08h	PORTD(2)	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx xxxx	71,226
09h	PORTE	RE7(3)	RE6(3)	RE5(3)	RE4(3)	RE3	RE2(2)	RE1(2)	RE0(2)	xxxx xxxx	76,226
0Ah	PCLATH	—	—	—	Write Buffer for upper 5 bits of Program Counter					---0 0000	40,226
0Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	34,226
0Ch	PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	37,226
0Dh	PIR2	OSFIF	C2IF	C1IF	LCDIF	—	LVDIF	—	CCP2IF(2)	0000 -0-0	38,226
0Eh	TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1								xxxx xxxx	102,226
0Fh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1								xxxx xxxx	102,226
10h	T1CON	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYNC}$	TMR1CS	TMR1ON	0000 0000	105,226
11h	TMR2	Timer2 Module Register								0000 0000	107,226
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	108,226
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	196,226
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	195,226
15h	CCPR1L	Capture/Compare/PWM Register 1 (LSB)								xxxx xxxx	213,226
16h	CCPR1H	Capture/Compare/PWM Register 1 (MSB)								xxxx xxxx	213,226
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	212,226
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	131,226
19h	TXREG	USART Transmit Data Register								0000 0000	130,226
1Ah	RCREG	USART Receive Data Register								0000 0000	128,227
1Bh(2)	CCPR2L	Capture/Compare/PWM Register 2 (LSB)								xxxx xxxx	213,227
1Ch(2)	CCPR2H	Capture/Compare/PWM Register 2 (MSB)								xxxx xxxx	213,227
1Dh(2)	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	212,227
1Eh	ADRESH	A/D Result Register High Byte								xxxx xxxx	182,227
1Fh	ADCON0	ADFM	VCFG1	VCFG0	CHS2	CHS1	CHS0	GO/$\overline{DONE}$	ADON	0000 0000	180,227

Legend: - = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition, shaded = unimplemented

Note 1: Other (non Power-up) Resets include $\overline{MCLR}$ Reset and Watchdog Timer Reset during normal operation.

2: PIC16F914/917 and PIC16F946 only, forced '0' on PIC16F913/916.

3: PIC16F946 only, forced to '0' on PIC16F91X.

TABLE 2-2: PIC16F91X/946 SPECIAL FUNCTION REGISTERS SUMMARY BANK 1

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Page
Bank 1											
80h	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	41,226
81h	OPTION_REG	$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	33,227
82h	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	40,226
83h	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	32,226
84h	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	41,226
85h	TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	44,227
86h	TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	54,227
87h	TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	62,227
88h	TRISD(3)	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	1111 1111	71,227
89h	TRISE	TRISE7(2)	TRISE6(2)	TRISE5(2)	TRISE4(2)	TRISE3(5)	TRISE2(3)	TRISE1(3)	TRISE0(3)	1111 1111	76,227
8Ah	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	40,226
8Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	34,226
8Ch	PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	35,227
8Dh	PIE2	OSFIE	C2IE	C1IE	LCDIE	—	LVDIE	—	CCP2IE(3)	0000 -0-0	36,227
8Eh	PCON	—	—	—	SBOREN	—	—	$\overline{POR}$	$\overline{BOR}$	---1 --qq	39,227
8Fh	OSCCON	—	IRCF2	IRCF1	IRCF0	OSTS(4)	HTS	LTS	SCS	-110 q000	88,227
90h	OSCTUNE	—	—	—	TUN4	TUN3	TUN2	TUN1	TUN0	---0 0000	92,227
91h	ANSEL	ANS7(3)	ANS6(3)	ANS5(3)	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	43,227
92h	PR2	Timer2 Period Register								1111 1111	107,227
93h	SSPADD	Synchronous Serial Port (I^2C mode) Address Register								0000 0000	202,227
94h	SSPSTAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	0000 0000	194,227
95h	WPUB	WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0	1111 1111	55,227
96h	IOCB	IOCB7	IOCB6	IOCB5	IOCB4	—	—	—	—	0000 ----	54,227
97h	CMCON1	—	—	—	—	—	—	T1GSS	C2SYNC	---- --10	117,227
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	130,227
99h	SPBRG	SPBRG7	SPBRG6	SPBRG5	SPBRG4	SPBRG3	SPBRG2	SPBRG1	SPBRG0	0000 0000	132,227
9Ah	—	Unimplemented								—	—
9Bh	—	Unimplemented								—	—
9Ch	CMCON0	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0000	116,227
9Dh	VRCON	VREN	—	VRR	—	VR3	VR2	VR1	VR0	0-0- 0000	118,227
9Eh	ADRESL	A/D Result Register Low Byte								xxxx xxxx	182,227
9Fh	ADCON1	—	ADCS2	ADCS1	ADCS0	—	—	—	—	-000 ----	181,227

Legend: - = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition, shaded = unimplemented

Note 1: Other (non Power-up) Resets include $\overline{MCLR}$ Reset and Watchdog Timer Reset during normal operation.

2: PIC16F946 only, forced '0' on PIC16F91X.

3: PIC16F914/917 and PIC16F946 only, forced '0' on PIC16F913/916.

4: The value of the OSTS bit is dependent on the value of the Configuration Word (CONFIG) of the device. See **Section 4.2 "Oscillator Control"**.

5: Bit is read-only; TRISE3 = 1 always.

TABLE 2-3: PIC16F91X/946 SPECIAL FUNCTION REGISTERS SUMMARY BANK 2

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Page
Bank 2											
100h	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	41,226
101h	TMR0	Timer0 Module Register								xxxx xxxx	99,226
102h	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	40,226
103h	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	32,226
104h	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	41,226
105h	WDTCON	—	—	—	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN	---0 1000	235,227
106h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	54,226
107h	LCDCON	LCDEN	$\overline{SLPEN}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	145,227
108h	LCDPS	WFT	BIASMD	LCDA	WA	LP3	LP2	LP1	LP0	0000 0000	146,227
109h	LVDCON	—	—	IRVST	LVDEN	—	LVDL2	LVDL1	LVDL0	--00 -100	145,228
10Ah	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	40,226
10Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	34,226
10Ch	EEDATL	EEDATL7	EEDATL6	EEDATL5	EEDATL4	EEDATL3	EEDATL2	EEDATL1	EEDATL0	0000 0000	188,228
10Dh	EEADRL	EEADRL7	EEADRL6	EEADRL5	EEADRL4	EEADRL3	EEADRL2	EEADRL1	EEADRL0	0000 0000	188,228
10Eh	EEDATH	—	—	EEDATH5	EEDATH4	EEDATH3	EEDATH2	EEDATH1	EEDATH0	--00 0000	188,228
10Fh	EEADRH	—	—	—	EEADRH4	EEADRH3	EEADRH2	EEADRH1	EEADRH0	---0 0000	188,228
110h	LCDDATA0	SEG7 COM0	SEG6 COM0	SEG5 COM0	SEG4 COM0	SEG3 COM0	SEG2 COM0	SEG1 COM0	SEG0 COM0	xxxx xxxx	147,228
111h	LCDDATA1	SEG15 COM0	SEG14 COM0	SEG13 COM0	SEG12 COM0	SEG11 COM0	SEG10 COM0	SEG9 COM0	SEG8 COM0	xxxx xxxx	147,228
112h	LCDDATA2(2)	SEG23 COM0	SEG22 COM0	SEG21 COM0	SEG20 COM0	SEG19 COM0	SEG18 COM0	SEG17 COM0	SEG16 COM0	xxxx xxxx	147,228
113h	LCDDATA3	SEG7 COM1	SEG6 COM1	SEG5 COM1	SEG4 COM1	SEG3 COM1	SEG2 COM1	SEG1 COM1	SEG0 COM1	xxxx xxxx	147,228
114h	LCDDATA4	SEG15 COM1	SEG14 COM1	SEG13 COM1	SEG12 COM1	SEG11 COM1	SEG10 COM1	SEG9 COM1	SEG8 COM1	xxxx xxxx	147,228
115h	LCDDATA5(2)	SEG23 COM1	SEG22 COM1	SEG21 COM1	SEG20 COM1	SEG19 COM1	SEG18 COM1	SEG17 COM1	SEG16 COM1	xxxx xxxx	147,228
116h	LCDDATA6	SEG7 COM2	SEG6 COM2	SEG5 COM2	SEG4 COM2	SEG3 COM2	SEG2 COM2	SEG1 COM2	SEG0 COM2	xxxx xxxx	147,228
117h	LCDDATA7	SEG15 COM2	SEG14 COM2	SEG13 COM2	SEG12 COM2	SEG11 COM2	SEG10 COM2	SEG9 COM2	SEG8 COM2	xxxx xxxx	147,228
118h	LCDDATA8(2)	SEG23 COM2	SEG22 COM2	SEG21 COM2	SEG20 COM2	SEG19 COM2	SEG18 COM2	SEG17 COM2	SEG16 COM2	xxxx xxxx	147,228
119h	LCDDATA9	SEG7 COM3	SEG6 COM3	SEG5 COM3	SEG4 COM3	SEG3 COM3	SEG2 COM3	SEG1 COM3	SEG0 COM3	xxxx xxxx	147,228
11Ah	LCDDATA10	SEG15 COM3	SEG14 COM3	SEG13 COM3	SEG12 COM3	SEG11 COM3	SEG10 COM3	SEG9 COM3	SEG8 COM3	xxxx xxxx	147,228
11Bh	LCDDATA11(2)	SEG23 COM3	SEG22 COM3	SEG21 COM3	SEG20 COM3	SEG19 COM3	SEG18 COM3	SEG17 COM3	SEG16 COM3	xxxx xxxx	147,228
11Ch	LCDSE0(3)	SE7	SE6	SE5	SE4	SE3	SE2	SE1	SE0	0000 0000	147,228
11Dh	LCDSE1(3)	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	147,228
11Eh	LCDSE2(2,3)	SE23	SE22	SE21	SE20	SE19	SE18	SE17	SE16	0000 0000	147,228
11Fh	—	Unimplemented								—	—

Legend: – = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition, shaded = unimplemented

Note 1: Other (non Power-up) Resets include $\overline{MCLR}$ Reset and Watchdog Timer Reset during normal operation.

2: PIC16F914/917 and PIC16F946 only.

3: This register is only initialized by a POR or BOR reset and is unchanged by other Resets.

TABLE 2-4: PIC16F91X/946 SPECIAL FUNCTION REGISTERS SUMMARY BANK 3

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Page
Bank 3											
180h	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	41,226
181h	OPTION_REG	$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	33,227
182h	PCL	Program Counter (PC) Least Significant Byte								0000 0000	40,226
183h	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	32,226
184h	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	41,226
185h	TRISF(3)	TRISF7	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2	TRISF1	TRISF0	1111 1111	81,228
186h	TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	54,227
187h	TRISG(3)	—	—	TRISG5	TRISG4	TRISG3	TRISG2	TRISG1	TRISG0	--11 1111	84,228
188h	PORTF(3)	RF7	RF6	RF5	RF4	RF3	RF2	RF1	RF0	xxxx xxxx	81,228
189h	PORTG(3)	—	—	RG5	RG4	RG3	RG2	RG1	RG0	--xx xxxx	84,228
18Ah	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	40,226
18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	34,226
18Ch	EECON1	EEPGD	—	—	—	WRERR	WREN	WR	RD	0--- x000	189,229
18Dh	EECON2	EEPROM Control Register 2 (not a physical register)								---- ----	187
18Eh	—	Reserved								—	—
18Fh	—	Reserved								—	—
190h	LCDDATA12(3)	SEG31 COM0	SEG30 COM0	SEG29 COM0	SEG28 COM0	SEG27 COM0	SEG26 COM0	SEG25 COM0	SEG24 COM0	xxxx xxxx	147,228
191h	LCDDATA13(3)	SEG39 COM0	SEG38 COM0	SEG37 COM0	SEG36 COM0	SEG35 COM0	SEG34 COM0	SE33 COM0	SEG32 COM0	xxxx xxxx	147,228
192h	LCDDATA14(3)	—	—	—	—	—	—	SEG41 COM0	SEG40 COM0	---- --xx	147,228
193h	LCDDATA15(3)	SEG31 COM1	SEG30 COM1	SEG29 COM1	SEG28 COM1	SEG27 COM1	SEG26 COM1	SEG25 COM1	SEG24 COM1	xxxx xxxx	147,228
194h	LCDDATA16(3)	SEG39 COM1	SEG38 COM1	SEG37 COM1	SEG36 COM1	SEG35 COM1	SEG34 COM1	SEG33 COM1	SEG32 COM1	xxxx xxxx	147,228
195h	LCDDATA17(3)	—	—	—	—	—	—	SEG41 COM1	SEG40 COM1	---- --xx	147,228
196h	LCDDATA18(3)	SEG31 COM2	SEG30 COM2	SEG29 COM2	SEG28 COM2	SEG27 COM2	SEG26 COM2	SEG25 COM2	SEG24 COM2	xxxx xxxx	147,228
197h	LCDDATA19(3)	SEG39 COM2	SEG38 COM2	SEG37 COM2	SEG36 COM2	SEG35 COM2	SEG34 COM2	SEG33 COM2	SEG32 COM2	xxxx xxxx	147,228
198h	LCDDATA20(3)	—	—	—	—	—	—	SEG41 COM2	SEG40 COM2	---- --xx	147,228
199h	LCDDATA21(3)	SEG31 COM3	SEG30 COM3	SEG29 COM3	SEG28 COM3	SEG27 COM3	SEG26 COM3	SEG25 COM3	SEG24 COM3	xxxx xxxx	147,228
19Ah	LCDDATA22(3)	SEG39 COM3	SEG38 COM3	SEG37 COM3	SEG36 COM3	SEG35 COM3	SEG34 COM3	SEG33 COM3	SEG32 COM3	xxxx xxxx	147,228
19Bh	LCDDATA23(3)	—	—	—	—	—	—	SEG41 COM3	SEG40 COM3	---- --xx	147,228
19Ch	LCDSE3(2, 3)	SE31	SE30	SE29	SE28	SE27	SE26	SE25	SE24	0000 0000	147,229
19Dh	LCDSE4(2, 3)	SE39	SE38	SE37	SE36	SE35	SE34	SE33	SE32	0000 0000	147,229
19Eh	LCDSE5(2, 3)	—	—	—	—	—	—	SE41	SE40	---- --00	147,229
19Fh	—	Unimplemented								—	—

Legend: - = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition, shaded = unimplemented

Note 1: Other (non Power-up) Resets include $\overline{MCLR}$ Reset and Watchdog Timer Reset during normal operation.

2: This register is only initialized by a POR or BOR reset and is unchanged by other Resets.

3: PIC16F946 only.

2.2.2.1 STATUS Register

The STATUS register, shown in Register 2-1, contains:

- the arithmetic status of the ALU
- the Reset status
- the bank select bits for data memory (SRAM)

The STATUS register can be the destination for any instruction, like any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the $\overline{TO}$ and $\overline{PD}$ bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, `CLRF STATUS` will clear the upper three bits and set the Z bit. This leaves the STATUS register as '`000u u1uu`' (where `u` = unchanged).

It is recommended, therefore, that only `BCF`, `BSF`, `SWAPF` and `MOVWF` instructions are used to alter the STATUS register, because these instructions do not affect any Status bits. For other instructions not affecting any Status bits (see **Section 17.0 "Instruction Set Summary"**).

Note 1: The C and DC bits operate as $\overline{Borrow}$ and $\overline{Digit\ Borrow}$ out bits, respectively, in subtraction.

REGISTER 2-1: STATUS: STATUS REGISTER

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC(1)	C(1)
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **IRP:** Register Bank Select bit (used for indirect addressing)
`1` = Bank 2, 3 (100h-1FFh)
`0` = Bank 0, 1 (00h-FFh)

bit 6-5 **RP<1:0>:** Register Bank Select bits (used for direct addressing)
`00` = Bank 0 (00h-7Fh)
`01` = Bank 1 (80h-FFh)
`10` = Bank 2 (100h-17Fh)
`11` = Bank 3 (180h-1FFh)

bit 4 **$\overline{TO}$:** Time-out bit
`1` = After power-up, `CLRWDT` instruction or `SLEEP` instruction
`0` = A WDT time-out occurred

bit 3 **$\overline{PD}$:** Power-down bit
`1` = After power-up or by the `CLRWDT` instruction
`0` = By execution of the `SLEEP` instruction

bit 2 **Z:** Zero bit
`1` = The result of an arithmetic or logic operation is zero
`0` = The result of an arithmetic or logic operation is not zero

bit 1 **DC:** Digit Carry/$\overline{Borrow}$ bit (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions)(1)
`1` = A carry-out from the 4th low-order bit of the result occurred
`0` = No carry-out from the 4th low-order bit of the result

bit 0 **C:** Carry/$\overline{Borrow}$ bit(1) (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions)(1)
`1` = A carry-out from the Most Significant bit of the result occurred
`0` = No carry-out from the Most Significant bit of the result occurred

Note 1: For $\overline{Borrow}$, the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (`RRF`, `RLF`) instructions, this bit is loaded with either the high-order or low-order bit of the source register.

2.2.2.2 OPTION register

The OPTION register, shown in Register 2-2, is a readable and writable register, which contains various control bits to configure:

- Timer0/WDT prescaler
- External RB0/INT interrupt
- Timer0
- Weak pull-ups on PORTB

Note: To achieve a 1:1 prescaler assignment for Timer0, assign the prescaler to the WDT by setting PSA bit of the OPTION register to '1'. See **Section 6.3 "Timer1 Prescaler"**.

REGISTER 2-2: OPTION_REG: OPTION REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **$\overline{\text{RBPU}}$:** PORTB Pull-up Enable bit
1 = PORTB pull-ups are disabled
0 = PORTB pull-ups are enabled by individual bits in the WPUB register

bit 6 **INTEDG:** Interrupt Edge Select bit
1 = Interrupt on rising edge of RB0/INT pin
0 = Interrupt on falling edge of RB0/INT pin

bit 5 **T0CS:** Timer0 Clock Source Select bit
1 = Transition on RA4/T0CKI pin
0 = Internal instruction cycle clock (FOSC/4)

bit 4 **T0SE:** Timer0 Source Edge Select bit
1 = Increment on high-to-low transition on RA4/T0CKI pin
0 = Increment on low-to-high transition on RA4/T0CKI pin

bit 3 **PSA:** Prescaler Assignment bit
1 = Prescaler is assigned to the WDT
0 = Prescaler is assigned to the Timer0 module

bit 2-0 **PS<2:0>:** Prescaler Rate Select bits

Bit Value	Timer0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

2.2.2.3 INTCON Register

The INTCON register is a readable and writable register, which contains the various enable and flag bits for TMR0 register overflow, PORTB change and external RB0/INT/SEG0 pin interrupts.

Note:	Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-3: INTCON: INTERRUPT CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	PEIE	T0IE	INTE	RBIE(1)	T0IF(2)	INTF	RBIF
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **GIE:** Global Interrupt Enable bit
1 = Enables all unmasked interrupts
0 = Disables all interrupts

bit 6 **PEIE:** Peripheral Interrupt Enable bit
1 = Enables all unmasked peripheral interrupts
0 = Disables all peripheral interrupts

bit 5 **T0IE:** Timer0 Overflow Interrupt Enable bit
1 = Enables the Timer0 interrupt
0 = Disables the Timer0 interrupt

bit 4 **INTE:** RB0/INT External Interrupt Enable bit
1 = Enables the RB0/INT external interrupt
0 = Disables the RB0/INT external interrupt

bit 3 **RBIE:** PORTB Change Interrupt Enable bit(1)
1 = Enables the PORTB change interrupt
0 = Disables the PORTB change interrupt

bit 2 **T0IF:** Timer0 Overflow Interrupt Flag bit(2)
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1 **INTF:** RB0/INT External Interrupt Flag bit
1 = The RB0/INT external interrupt occurred (must be cleared in software)
0 = The RB0/INT external interrupt did not occur

bit 0 **RBIF:** PORTB Change Interrupt Flag bit
1 = When at least one of the PORTB general purpose I/O pins changed state (must be cleared in software)
0 = None of the PORTB general purpose I/O pins have changed state

Note 1: The appropriate bits in the IOCB register must also be set.
2: T0IF bit is set when Timer0 rolls over. Timer0 is unchanged on Reset and should be initialized before clearing T0IF bit.

2.2.2.4 PIE1 Register

The PIE1 register contains the interrupt enable bits, as shown in Register 2-4.

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

REGISTER 2-4: PIE1: PERIPHERAL INTERRUPT ENABLE REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **EEIE:** EE Write Complete Interrupt Enable bit
1 = Enables the EE write complete interrupt
0 = Disables the EE write complete interrupt

bit 6 **ADIE:** A/D Converter (ADC) Interrupt Enable bit
1 = Enables the ADC interrupt
0 = Disables the ADC interrupt

bit 5 **RCIE:** USART Receive Interrupt Enable bit
1 = Enables the USART receive interrupt
0 = Disables the USART receive interrupt

bit 4 **TXIE:** USART Transmit Interrupt Enable bit
1 = Enables the USART transmit interrupt
0 = Disables the USART transmit interrupt

bit 3 **SSPIE:** Synchronous Serial Port (SSP) Interrupt Enable bit
1 = Enables the SSP interrupt
0 = Disables the SSP interrupt

bit 2 **CCP1IE:** CCP1 Interrupt Enable bit
1 = Enables the CCP1 interrupt
0 = Disables the CCP1 interrupt

bit 1 **TMR2IE:** TMR2 to PR2 Match Interrupt Enable bit
1 = Enables the Timer2 to PR2 match interrupt
0 = Disables the Timer2 to PR2 match interrupt

bit 0 **TMR1IE:** Timer1 Overflow Interrupt Enable bit
1 = Enables the Timer1 overflow interrupt
0 = Disables the Timer1 overflow interrupt

2.2.2.5 PIE2 Register

The PIE2 register contains the interrupt enable bits, as shown in Register 2-5.

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

REGISTER 2-5: PIE2: PERIPHERAL INTERRUPT ENABLE REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0
OSFIE	C2IE	C1IE	LCDIE	—	LVDIE	—	CCP2IE(1)
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **OSFIE:** Oscillator Fail Interrupt Enable bit
1 = Enables oscillator fail interrupt
0 = Disables oscillator fail interrupt

bit 6 **C2IE:** Comparator C2 Interrupt Enable bit
1 = Enables Comparator C2 interrupt
0 = Disables Comparator C2 interrupt

bit 5 **C1IE:** Comparator C1 Interrupt Enable bit
1 = Enables Comparator C1 interrupt
0 = Disables Comparator C1 interrupt

bit 4 **LCDIE:** LCD Module Interrupt Enable bit
1 = Enables LCD interrupt
0 = Disables LCD interrupt

bit 3 **Unimplemented:** Read as '0'

bit 2 **LVDIE:** Low Voltage Detect Interrupt Enable bit
1 = Enables LVD Interrupt
0 = Disables LVD Interrupt

bit 1 **Unimplemented:** Read as '0'

bit 0 **CCP2IE:** CCP2 Interrupt Enable bit(1)
1 = Enables the CCP2 interrupt
0 = Disables the CCP2 interrupt

Note 1: PIC16F914/PIC16F917/PIC16F946 only.

2.2.2.6 PIR1 Register

The PIR1 register contains the interrupt flag bits, as shown in Register 2-6.

Note:	Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-6: PIR1: PERIPHERAL INTERRUPT REQUEST REGISTER 1

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **EEIF:** EE Write Operation Interrupt Flag bit
- 1 = The write operation completed (must be cleared in software)
- 0 = The write operation has not completed or has not started

bit 6 **ADIF:** A/D Converter Interrupt Flag bit
- 1 = A/D conversion complete (must be cleared in software)
- 0 = A/D conversion has not completed or has not been started

bit 5 **RCIF:** USART Receive Interrupt Flag bit
- 1 = The USART receive buffer is full (cleared by reading RCREG)
- 0 = The USART receive buffer is not full

bit 4 **TXIF:** USART Transmit Interrupt Flag bit
- 1 = The USART transmit buffer is empty (cleared by writing to TXREG)
- 0 = The USART transmit buffer is full

bit 3 **SSPIF:** Synchronous Serial Port (SSP) Interrupt Flag bit
- 1 = The Transmission/Reception is complete (must be cleared in software)
- 0 = Waiting to Transmit/Receive

bit 2 **CCP1IF:** CCP1 Interrupt Flag bit

Capture mode:
- 1 = A TMR1 register capture occurred (must be cleared in software)
- 0 = No TMR1 register capture occurred

Compare mode:
- 1 = A TMR1 register compare match occurred (must be cleared in software)
- 0 = No TMR1 register compare match occurred

PWM mode
Unused in this mode

bit 1 **TMR2IF:** Timer2 to PR2 Interrupt Flag bit
- 1 = A Timer2 to PR2 match occurred (must be cleared in software)
- 0 = No Timer2 to PR2 match occurred

bit 0 **TMR1IF:** Timer1 Overflow Interrupt Flag bit
- 1 = The TMR1 register overflowed (must be cleared in software)
- 0 = The TMR1 register did not overflow

2.2.2.7 PIR2 Register

The PIR2 register contains the interrupt flag bits, as shown in Register 2-7.

Note:	Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-7: PIR2: PERIPHERAL INTERRUPT REQUEST REGISTER 2

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0
OSFIF	C2IF	C1IF	LCDIF	—	LVDIF	—	CCP2IF(1)
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **OSFIF:** Oscillator Fail Interrupt Flag bit
1 = System oscillator failed, clock input has changed to INTOSC (must be cleared in software)
0 = System clock operating

bit 6 **C2IF:** Comparator C2 Interrupt Flag bit
1 = Comparator output (C2OUT bit) has changed (must be cleared in software)
0 = Comparator output (C2OUT bit) has not changed

bit 5 **C1IF:** Comparator C1 Interrupt Flag bit
1 = Comparator output (C1OUT bit) has changed (must be cleared in software)
0 = Comparator output (C1OUT bit) has not changed

bit 4 **LCDIF:** LCD Module Interrupt bit
1 = LCD has generated an interrupt
0 = LCD has not generated an interrupt

bit 3 **Unimplemented:** Read as '0'

bit 2 **LVDIF:** Low Voltage Detect Interrupt Flag bit
1 = LVD has generated an interrupt
0 = LVD has not generated an interrupt

bit 1 **Unimplemented:** Read as '0'

bit 0 **CCP2IF:** CCP2 Interrupt Flag bit(1)
Capture Mode:
1 = A TMR1 register capture occurred (must be cleared in software)
0 = No TMR1 register capture occurred
Compare Mode:
1 = A TMR1 register compare match occurred (must be cleared in software)
0 = No TMR1 register compare match occurred
PWM mode:
Unused in this mode

Note 1: PIC16F914/PIC16F917/PIC16F946 only.

2.2.2.8 PCON Register

The Power Control (PCON) register contains flag bits (see Table 16-2) to differentiate between a:

- Power-on Reset ($\overline{POR}$)
- Brown-out Reset ($\overline{BOR}$)
- Watchdog Timer Reset (WDT)
- External $\overline{MCLR}$ Reset

The PCON register also controls the software enable of the BOR.

The PCON register bits are shown in Register 2-8.

REGISTER 2-8: PCON: POWER CONTROL REGISTER

U-0	U-0	U-0	R/W-1	U-0	U-0	R/W-0	R/W-x
—	—	—	SBOREN	—	—	$\overline{POR}$	$\overline{BOR}$
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-5 **Unimplemented:** Read as '0'

bit 4 **SBOREN:** Software BOR Enable bit(1)
- 1 = BOR enabled
- 0 = BOR disabled

bit 3-2 **Unimplemented:** Read as '0'

bit 1 **$\overline{POR}$:** Power-on Reset Status bit
- 1 = No Power-on Reset occurred
- 0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

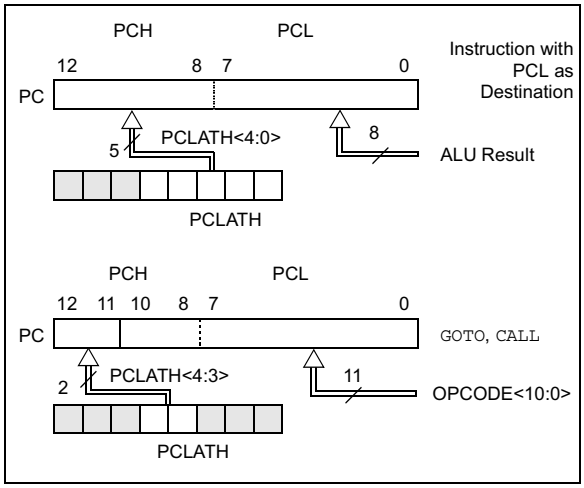
bit 0 **$\overline{BOR}$:** Brown-out Reset Status bit
- 1 = No Brown-out Reset occurred
- 0 = A Brown-out Reset occurred (must be set in software after a Power-on Reset or Brown-out Reset occurs)

Note 1: Set BOREN<1:0> = 01 in the Configuration Word register for this bit to control the $\overline{BOR}$.

2.3 PCL and PCLATH

The Program Counter (PC) is 13 bits wide. The low byte comes from the PCL register, which is a readable and writable register. The high byte (PC<12:8>) is not directly readable or writable and comes from PCLATH. On any Reset, the PC is cleared. Figure 2-6 shows the two situations for the loading of the PC. The upper example in Figure 2-6 shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in Figure 2-6 shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

FIGURE 2-6: LOADING OF PC IN DIFFERENT SITUATIONS



2.3.1 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL). When performing a table read using a computed GOTO method, care should be exercised if the table location crosses a PCL memory boundary (each 256-byte block). Refer to the Application Note AN556, *"Implementing a Table Read"* (DS00556).

2.3.2 STACK

The PIC16F91X/946 family has an 8-level x 13-bit wide hardware stack (see Figures 2-1 and 2-2). The stack space is not part of either program or data space and the Stack Pointer is not readable or writable. The PC is PUSHed onto the stack when a CALL instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not affected by a PUSH or POP operation.

The stack operates as a circular buffer. This means that after the stack has been PUSHed eight times, the ninth PUSH overwrites the value that was stored from the first PUSH. The tenth PUSH overwrites the second PUSH (and so on).

Note 1:	There are no Status bits to indicate stack overflow or stack underflow conditions.
2:	There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, RETURN, RETLW and RETFIE instructions or the vectoring to an interrupt address.

2.4 Program Memory Paging

All PIC16F91X/946 devices are capable of addressing a continuous 8K word block of program memory. The CALL and GOTO instructions provide only 11 bits of address to allow branching within any 2K program memory page. When doing a CALL or GOTO instruction, the upper 2 bits of the address are provided by PCLATH<4:3>. When doing a CALL or GOTO instruction, the user must ensure that the page select bits are programmed so that the desired program memory page is addressed. If a return from a CALL instruction (or interrupt) is executed, the entire 13-bit PC is POPed off the stack. Therefore, manipulation of the PCLATH<4:3> bits is not required for the RETURN instructions (which POPs the address from the stack).

Note:	The contents of the PCLATH register are unchanged after a RETURN or RETFIE instruction is executed. The user must rewrite the contents of the PCLATH register for any subsequent subroutine calls or GOTO instructions.

Example 2-1 shows the calling of a subroutine in page 1 of the program memory. This example assumes that PCLATH is saved and restored by the Interrupt Service Routine (if interrupts are used).

EXAMPLE 2-1: CALL OF A SUBROUTINE IN PAGE 1 FROM PAGE 0

```
        ORG 500h
        BCF PCLATH,4
        BSF PCLATH,3    ;Select page 1
                        ;(800h-FFFh)
        CALL SUB1_P1    ;Call subroutine in
        :               ;page 1 (800h-FFFh)
        :
        ORG 900h        ;page 1 (800h-FFFh)
SUB1_P1
        :               ;called subroutine
                        ;page 1 (800h-FFFh)
        :
        RETURN          ;return to
                        ;Call subroutine
                        ;in page 0
                        ;(000h-7FFh)
```

2.5 Indirect Addressing, INDF and FSR Registers

The INDF register is not a physical register. Addressing the INDF register will cause indirect addressing.

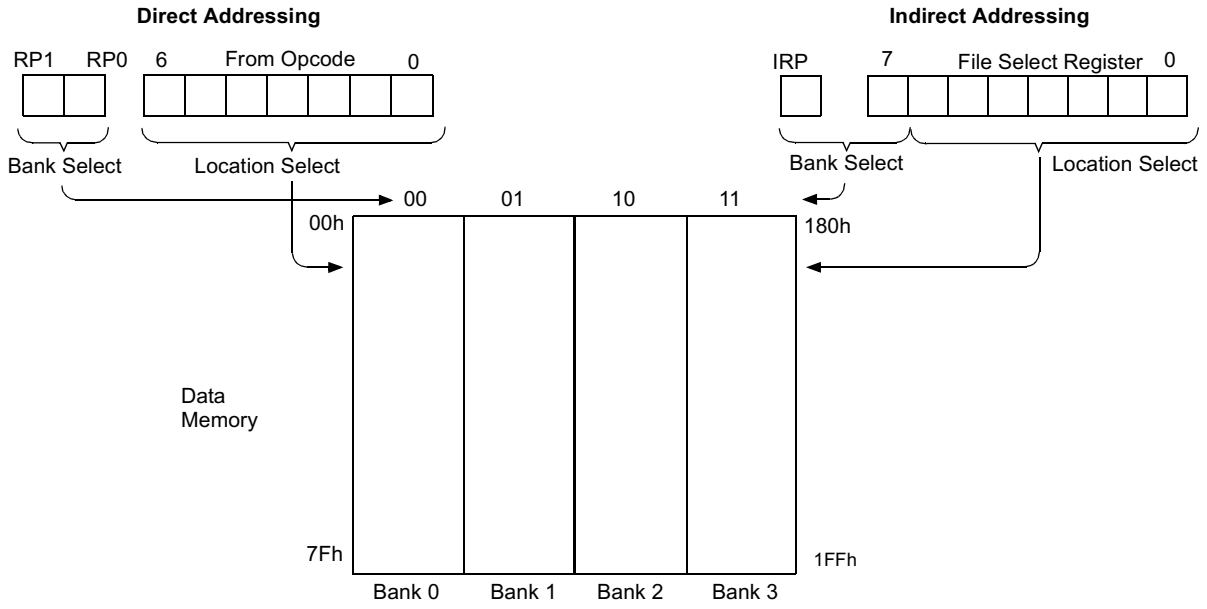
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select Register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no operation (although Status bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit of the STATUS register, as shown in Figure 2-7.

A simple program to clear RAM location 020h-02Fh using indirect addressing is shown in Example 2-2.

EXAMPLE 2-2: INDIRECT ADDRESSING

```
     MOVLW    020h   ;initialize pointer
     MOVWF    FSR    ;to RAM
     BANKISEL 020h
NEXT CLRF     INDF   ;clear INDF register
     INCF     FSR    ;inc pointer
     BTFSS    FSR,4  ;all done?
     GOTO     NEXT   ;no clear next
CONTINUE             ;yes continue
```

FIGURE 2-7: DIRECT/INDIRECT ADDRESSING PIC16F91X/946



Note: For memory map detail, see Figures 2-3 and 2-4.

NOTES:

3.0 I/O PORTS

The PIC16F913/914/916/917/946 family of devices includes several 8-bit PORT registers along with their corresponding TRIS registers and one four bit port:

- PORTA and TRISA
- PORTB and TRISB
- PORTC and TRISC
- PORTD and TRISD(1)
- PORTE and TRISE
- PORTF and TRISF(2)
- PORTG and TRISG(2)

Note 1:	PIC16F914/917 and PIC16F946 only.
2:	PIC16F946 only

PORTA, PORTB, PORTC and RE3/$\overline{MCLR}$/VPP are implemented on all devices. PORTD and RE<2:0> (PORTE) are implemented only on the PIC16F914/917 and PIC16F946. RE<7:4> (PORTE), PORTF and PORTG are implemented only on the PIC16F946.

3.1 ANSEL Register

The ANSEL register (Register 3-1) is used to configure the Input mode of an I/O pin to analog. Setting the appropriate ANSEL bit high will cause all digital reads on the pin to be read as '0' and allow analog functions on the pin to operate correctly.

The state of the ANSEL bits has no affect on digital output functions. A pin with TRIS clear and ANSEL set will still operate as a digital output, but the Input mode will be analog. This can cause unexpected behavior when executing read-modify-write instructions on the affected port.

REGISTER 3-1: ANSEL: ANALOG SELECT REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
ANS7(2)	ANS6(2)	ANS5(2)	ANS4	ANS3	ANS2	ANS1	ANS0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **ANS<7:0>**: Analog Select bits
Analog select between analog or digital function on pins AN<7:0>, respectively.
1 = Analog input. Pin is assigned as analog input(1).
0 = Digital I/O. Pin is assigned to port or special function.

Note 1: Setting a pin to an analog input automatically disables the digital input circuitry, weak pull-ups, and interrupt-on-change if available. The corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

2: PIC16F914/PIC16F917/PIC16F946 only.

3.2 PORTA and TRISA Registers

PORTA is a 8-bit wide, bidirectional port. The corresponding data direction register is TRISA (Register 3-3). Setting a TRISA bit (= 1) will make the corresponding PORTA pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISA bit (= 0) will make the corresponding PORTA pin an output (i.e., put the contents of the output latch on the selected pin). Example 3-1 shows how to initialize PORTA.

Five of the pins of PORTA can be configured as analog inputs. These pins, RA5 and RA<3:0>, are configured as analog inputs on device power-up and must be reconfigured by the user to be used as I/O's. This is done by writing the appropriate values to the CMCON0 and ANSEL registers (see Example 3-1).

Reading the PORTA register (Register 3-2) reads the status of the pins, whereas writing to it will write to the PORT latch. All write operations are read-modify-write operations. Therefore, a write to a port means that the port pins are read, this value is modified and then written to the PORT data latch.

The TRISA register controls the direction of the PORTA pins, even when they are being used as analog inputs. The user must ensure the bits in the TRISA register are maintained set when using them as analog inputs. I/O pins configured as analog inputs always read '0'.

Note 1:	The CMCON0 and ANSEL registers must be initialized to configure an analog channel as a digital input. Pins configured as analog inputs will read '0'.

EXAMPLE 3-1: INITIALIZING PORTA

```
BANKSEL PORTA   ;
CLRF    PORTA   ;Init PORTA
BANKSEL TRISA   ;
MOVLW   07h     ;Set RA<2:0> to
MOVWF   CMCON0  ;digital I/O
CLRF    ANSEL   ;Make all PORTA digital I/O
MOVLW   0F0h    ;Set RA<7:4> as inputs
MOVWF   TRISA   ;and set RA<3:0> as outputs
```

REGISTER 3-2: PORTA: PORTA REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **RA<7:0>**: PORTA I/O Pin bits
1 = Port pin is >V_{IH} min.
0 = Port pin is <V_{IL} max.

REGISTER 3-3: TRISA: PORTA TRI-STATE REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **TRISA<7:0>:** PORTA Tri-State Control bits
1 = PORTA pin configured as an input (tri-stated)
0 = PORTA pin configured as an output

Note 1: TRISA<7:6> always reads '1' in XT, HS and LP Oscillator modes.

3.2.1 PIN DESCRIPTIONS AND DIAGRAMS

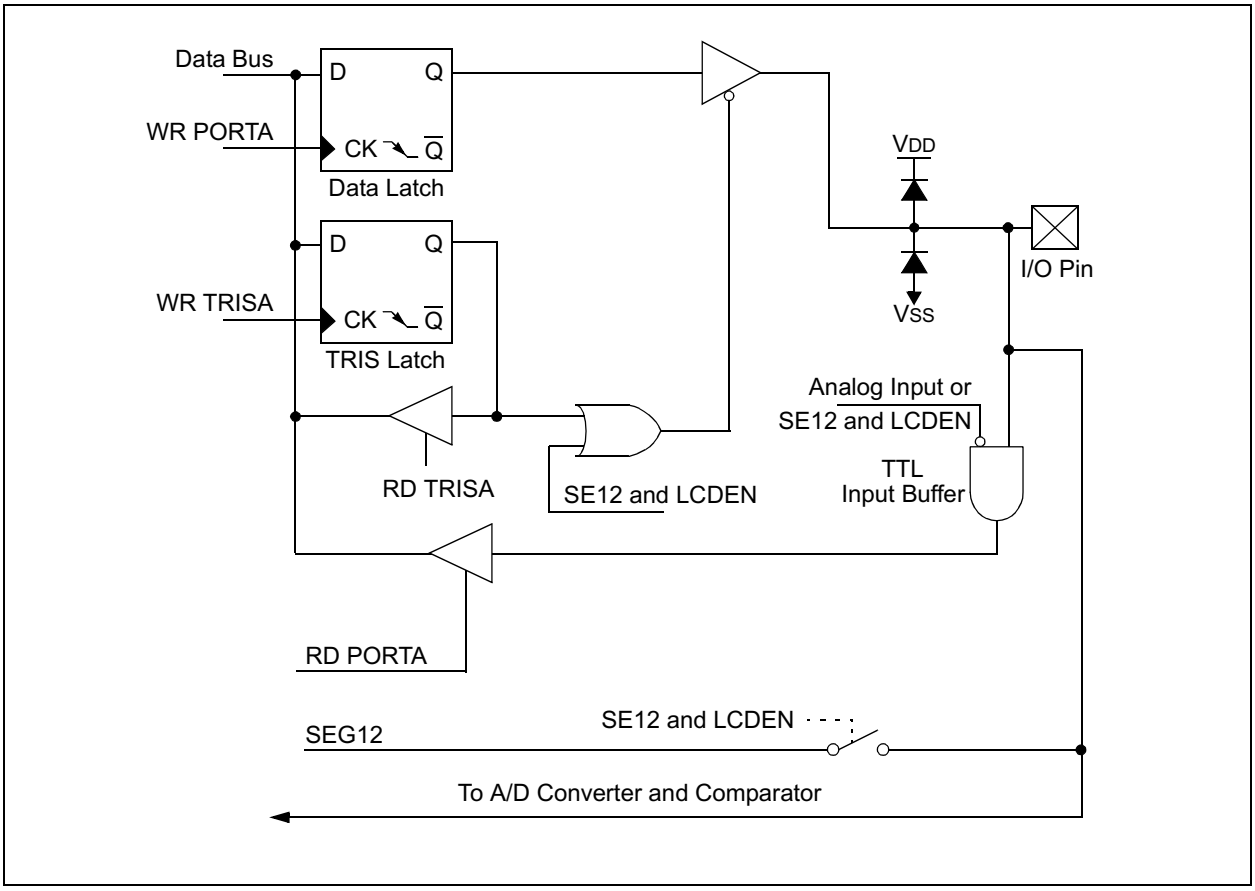
Each PORTA pin is multiplexed with other functions. The pins and their combined functions are briefly described here. For specific information about individual functions, refer to the appropriate section in this data sheet.

3.2.1.1 RA0/AN0/C1-/SEG12

Figure 3-1 shows the diagram for this pin. The RA0 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the ADC
- an analog input for Comparator C1
- an analog output for the LCD

FIGURE 3-1: BLOCK DIAGRAM OF RA0

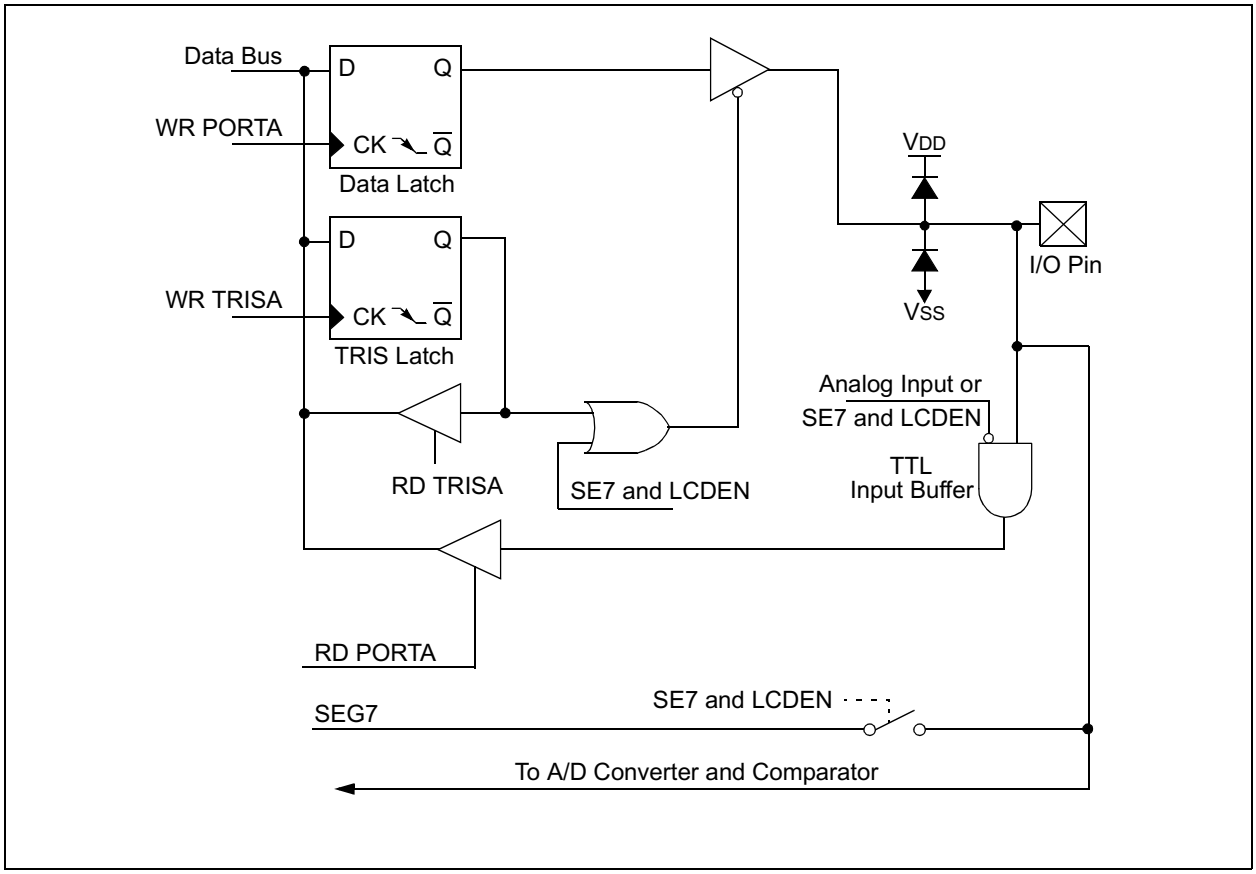


3.2.1.2 RA1/AN1/C2-/SEG7

Figure 3-2 shows the diagram for this pin. The RA1 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the ADC
- an analog input for Comparator C2
- an analog output for the LCD

FIGURE 3-2: BLOCK DIAGRAM OF RA1

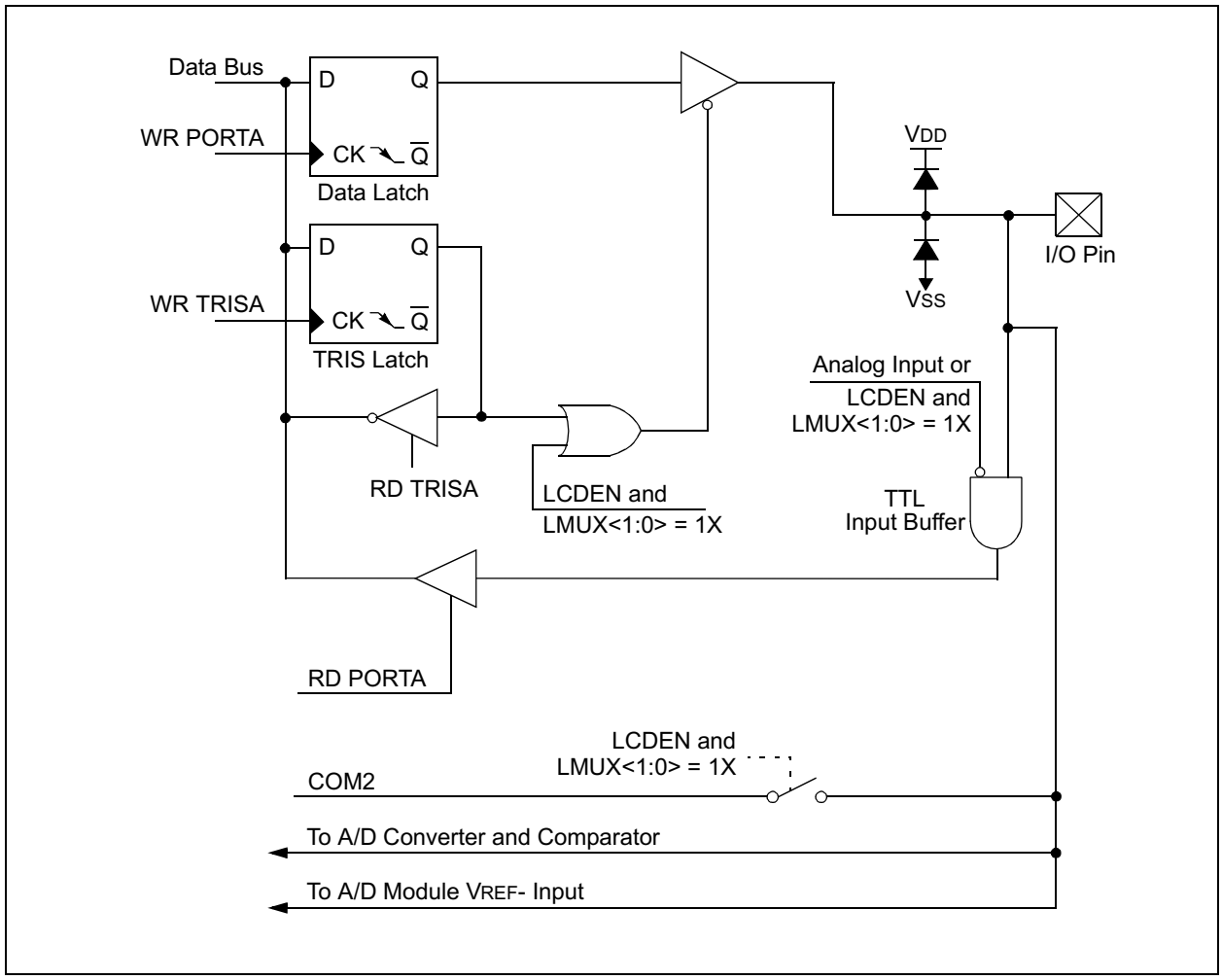


3.2.1.3 RA2/AN2/C2+/VREF-/COM2

Figure 3-3 shows the diagram for this pin. The RA2 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the ADC
- an analog input for Comparator C2
- a voltage reference input for the ADC
- an analog output for the LCD

FIGURE 3-3: BLOCK DIAGRAM OF RA2

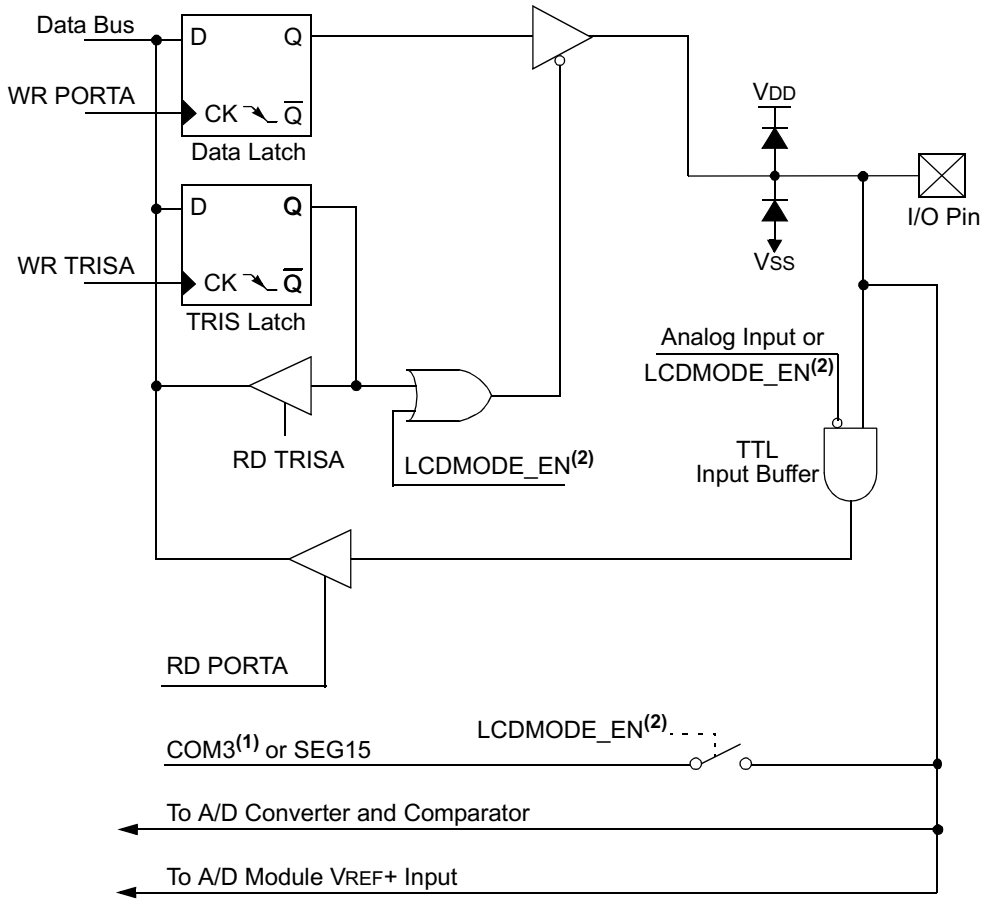


3.2.1.4 RA3/AN3/C1+/VREF+/COM3/SEG15

Figure 3-4 shows the diagram for this pin. The RA3 pin is configurable to function as one of the following:

- a general purpose input
- an analog input for the ADC
- an analog input from Comparator C1
- a voltage reference input for the ADC
- analog outputs for the LCD

FIGURE 3-4: BLOCK DIAGRAM OF RA3



Note 1: PIC16F913/916 only.

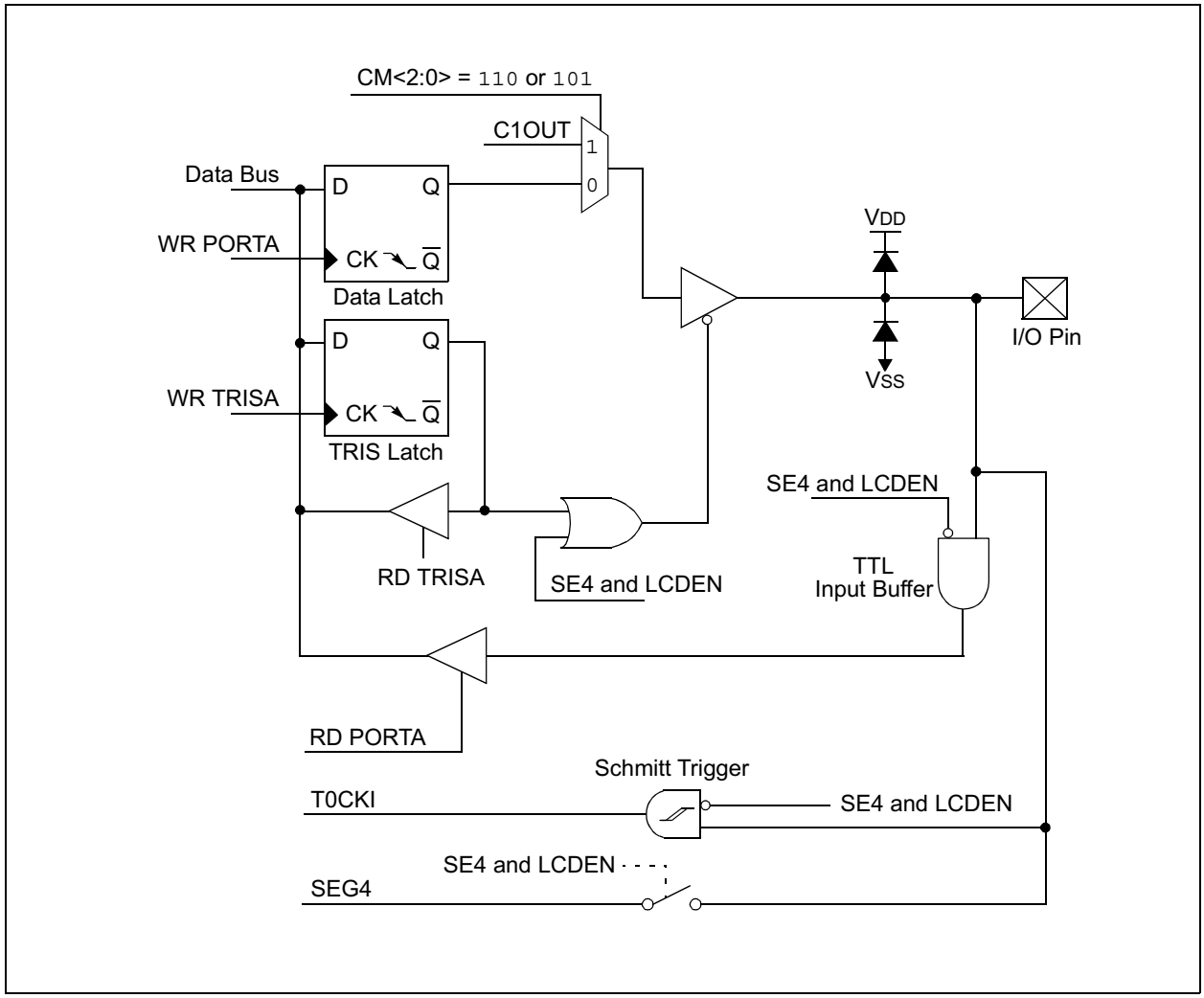
2: For the PIC16F913/916, the LCDMODE_EN = LCDEN and (SE15 or LMUX<1:0> = 11). For the PIC16F914/917 and PIC16F946, the LCDMODE_EN = LCDEN and SE15.

3.2.1.5 RA4/C1OUT/T0CKI/SEG4

Figure 3-5 shows the diagram for this pin. The RA4 pin is configurable to function as one of the following:

- a general purpose I/O
- a digital output from Comparator C1
- a clock input for Timer0
- an analog output for the LCD

FIGURE 3-5: BLOCK DIAGRAM OF RA4

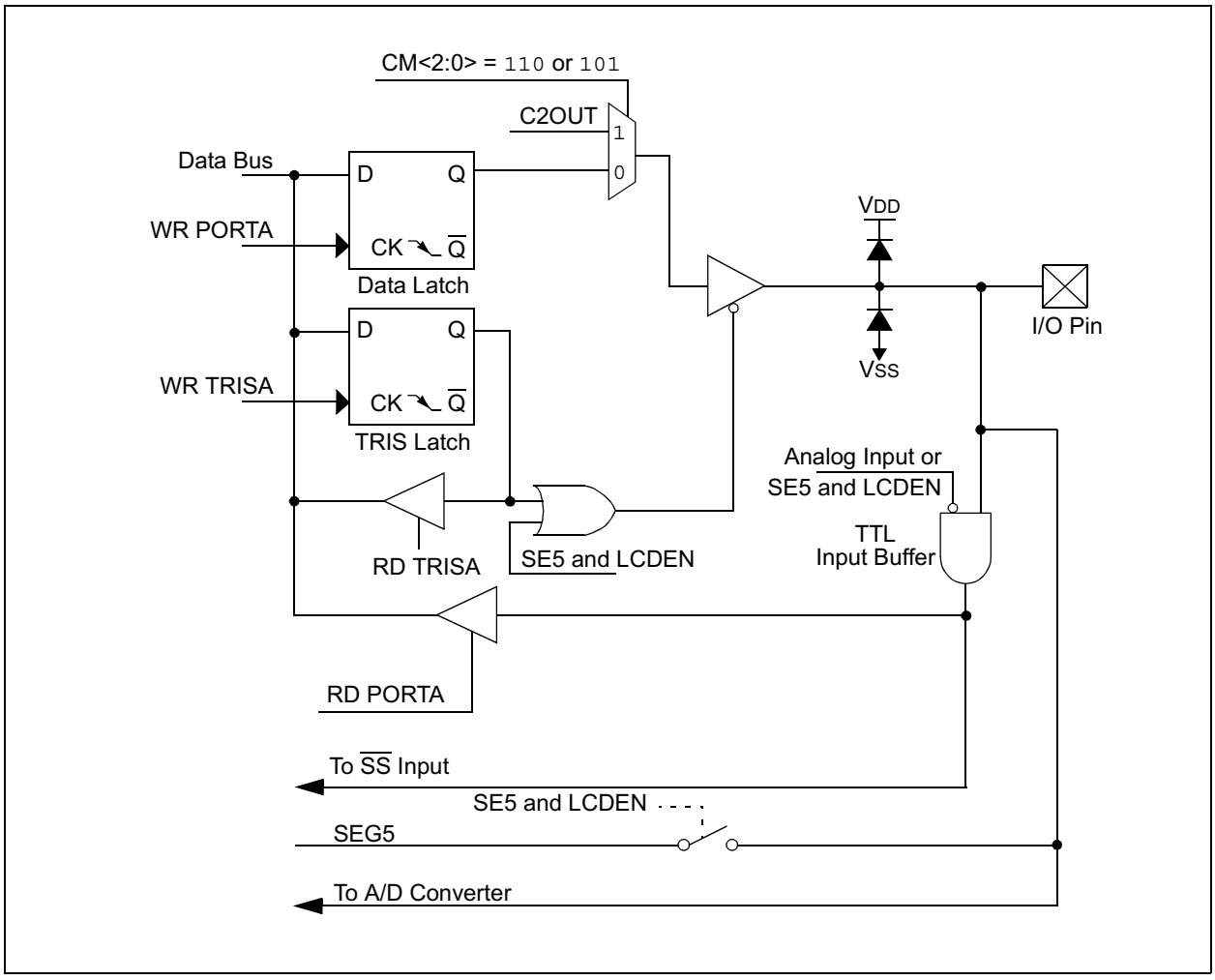


3.2.1.6 RA5/AN4/C2OUT/$\overline{SS}$/SEG5

Figure 3-6 shows the diagram for this pin. The RA5 pin is configurable to function as one of the following:

- a general purpose I/O
- a digital output from Comparator C2
- a slave select input
- an analog output for the LCD
- an analog input for the ADC

FIGURE 3-6: BLOCK DIAGRAM OF RA5

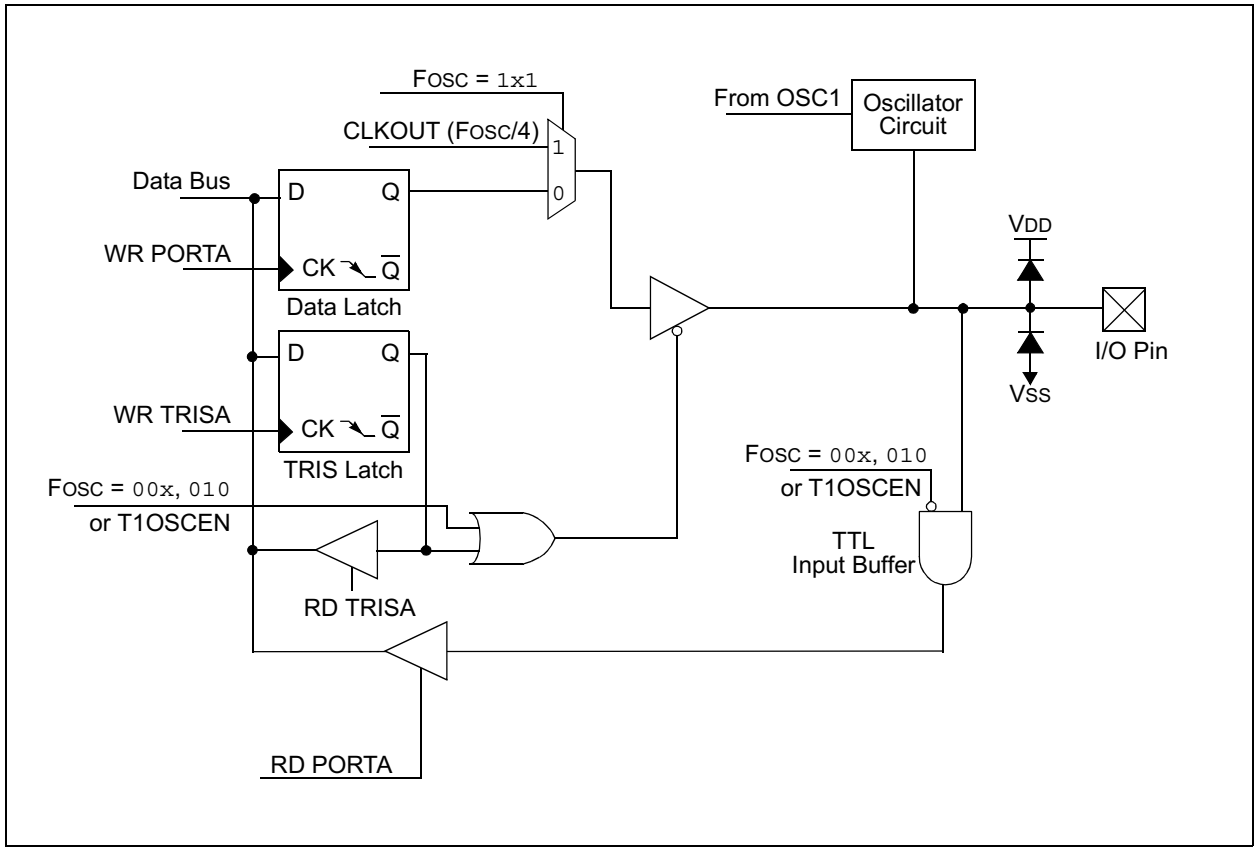


3.2.1.7 RA6/OSC2/CLKOUT/T1OSO

Figure 3-7 shows the diagram for this pin. The RA6 pin is configurable to function as one of the following:

- a general purpose I/O
- a crystal/resonator connection
- a clock output
- a Timer1 oscillator connection

FIGURE 3-7: BLOCK DIAGRAM OF RA6



3.2.1.8 RA7/OSC1/CLKIN/T1OSI

Figure 3-8 shows the diagram for this pin. The RA7 pin is configurable to function as one of the following:

- a general purpose I/O
- a crystal/resonator connection
- a clock input
- a Timer1 oscillator connection

FIGURE 3-8: BLOCK DIAGRAM OF RA7

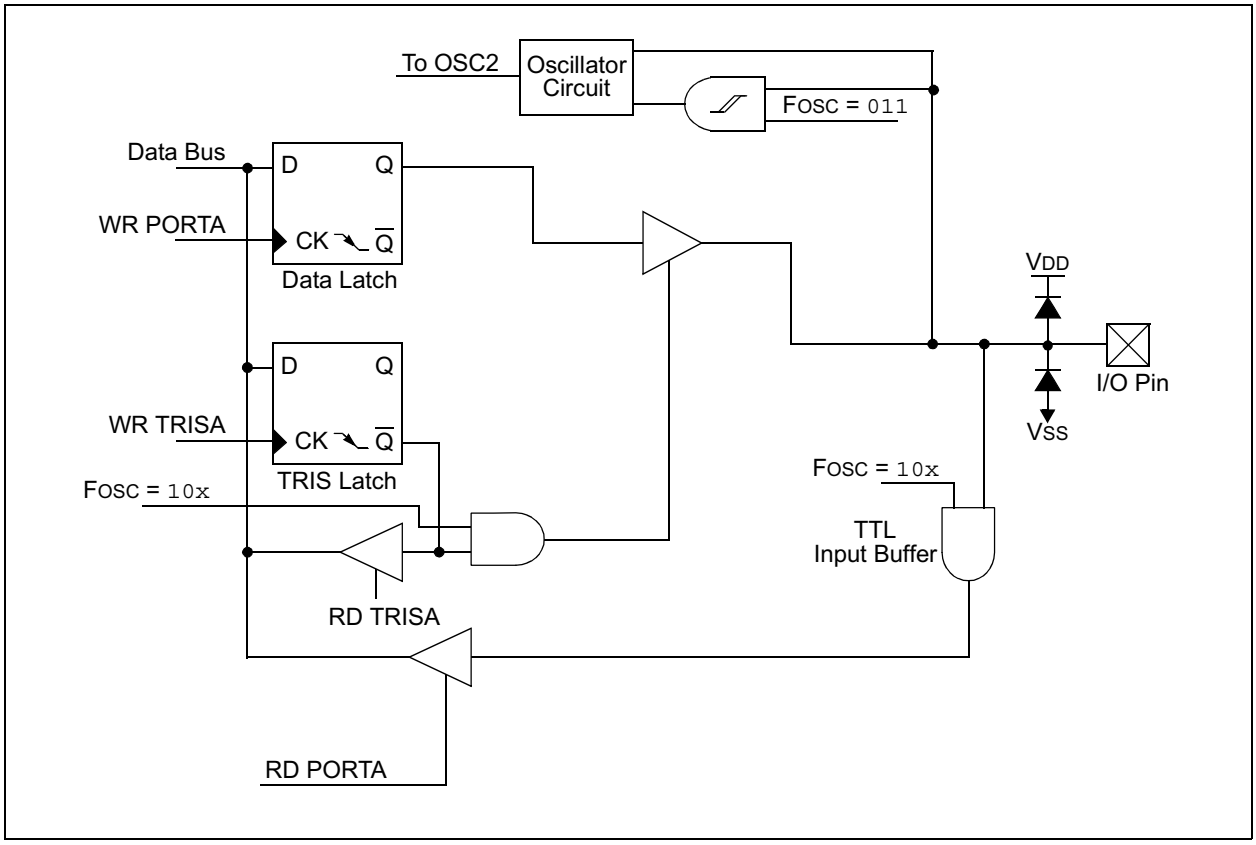


TABLE 3-1: SUMMARY OF REGISTERS ASSOCIATED WITH PORTA

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
ADCON0	ADFM	VCFG1	VCFG0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	ADON	0000 0000	0000 0000
ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
CMCON0	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0000	0000 0000
CONFIG(1)	$\overline{\text{CPD}}$	$\overline{\text{CP}}$	MCLRE	$\overline{\text{PWRTE}}$	WDTE	FOSC2	FOSC1	FOSC0	—	—
OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE0	SE7	SE6	SE5	SE4	SE3	SE2	SE1	SE0	0000 0000	uuuu uuuu
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	uuuu uuuu
PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx xxxx	uuuu uuuu
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
T1CON	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	0000 0000	uuuu uuuu
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTA.
Note 1: See Configuration Word register (CONFIG) for operation of all register bits.

3.3 PORTB and TRISB Registers

PORTB is an 8-bit bidirectional I/O port. All PORTB pins can have a weak pull-up feature, and PORTB<7:4> implements an interrupt-on-input change function.

PORTB is also used for the Serial Flash programming interface and ICD interface.

EXAMPLE 3-2: INITIALIZING PORTB

```
BANKSEL PORTB       ;
CLRF    PORTB       ;Init PORTB
BANKSEL TRISB       ;
MOVLW   0FFh        ;Set RB<7:0> as inputs
MOVWF   TRISB       ;
```

3.4 Additional PORTB Pin Functions

RB<7:6> are used as data and clock signals, respectively, for both serial programming and the in-circuit debugger features on the device. Also, RB0 can be configured as an external interrupt input.

3.4.1 WEAK PULL-UPS

Each of the PORTB pins has an individually configurable internal weak pull-up. Control bits WPUB<7:0> enable or disable each pull-up. Refer to Register 3-7. Each weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset by the $\overline{\text{RBPU}}$ bit of the OPTION register.

3.4.2 INTERRUPT-ON-CHANGE

Four of the PORTB pins are individually configurable as an interrupt-on-change pin. Control bits IOCB<7:4> enable or disable the interrupt function for each pin. Refer to Register 3-6. The interrupt-on-change feature is disabled on a Power-on Reset.

For enabled interrupt-on-change pins, the values are compared with the old value latched on the last read of PORTB. The 'mismatch' outputs of the last read are OR'd together to set the PORTB Change Interrupt flag bit (RBIF) in the INTCON register (Register 2-3).

This interrupt can wake the device from Sleep. The user, in the Interrupt Service Routine, clears the interrupt by:

a) Any read or write of PORTB. This will end the mismatch condition.
b) Clear the flag bit RBIF.

A mismatch condition will continue to set flag bit RBIF. Reading or writing PORTB will end the mismatch condition and allow flag bit RBIF to be cleared. The latch holding the last read value is not affected by a $\overline{\text{MCLR}}$ nor Brown-out Reset. After these Resets, the RBIF flag will continue to be set if a mismatch is present.

Note:	If a change on the I/O pin should occur when the read operation is being executed (start of the Q2 cycle), then the RBIF interrupt flag may not get set. Furthermore, since a read or write on a port affects all bits of that port, care must be taken when using multiple pins in Interrupt-on-change mode. Changes on one pin may not be seen while servicing changes on another pin.

REGISTER 3-4: PORTB: PORTB REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **RB<7:0>**: PORTB I/O Pin bits
1 = Port pin is >VIH min.
0 = Port pin is <VIL max.

REGISTER 3-5: TRISB: PORTB TRI-STATE REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **TRISB<7:0>:** PORTB Tri-State Control bits
1 = PORTB pin configured as an input (tri-stated)
0 = PORTB pin configured as an output

REGISTER 3-6: IOCB: PORTB INTERRUPT-ON-CHANGE REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
IOCB7	IOCB6	IOCB5	IOCB4	—	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-4 **IOCB<7:4>:** Interrupt-on-Change bits
1 = Interrupt-on-change enabled
0 = Interrupt-on-change disabled

bit 3-0 **Unimplemented:** Read as '0'

REGISTER 3-7: WPUB: WEAK PULL-UP REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **WPUB<7:0>:** Weak Pull-up Register bits
1 = Pull-up enabled
0 = Pull-up disabled

Note 1: Global $\overline{\text{RBPU}}$ must be enabled for individual pull-ups to be enabled.
2: The weak pull-up device is automatically disabled if the pin is in Output mode (TRISx<7:0> = 0).

3.4.3 PIN DESCRIPTIONS AND DIAGRAMS

Each PORTB pin is multiplexed with other functions. The pins and their combined functions are briefly described here. For specific information about individual functions such as the LCD or interrupts, refer to the appropriate section in this data sheet.

3.4.3.1 RB0/INT/SEG0

Figure 3-9 shows the diagram for this pin. The RB0 pin is configurable to function as one of the following:

- a general purpose I/O
- an external edge triggered interrupt
- an analog output for the LCD

3.4.3.2 RB1/SEG1

Figure 3-9 shows the diagram for this pin. The RB1 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.4.3.3 RB2/SEG2

Figure 3-9 shows the diagram for this pin. The RB2 pin is configurable to function as one of the following:

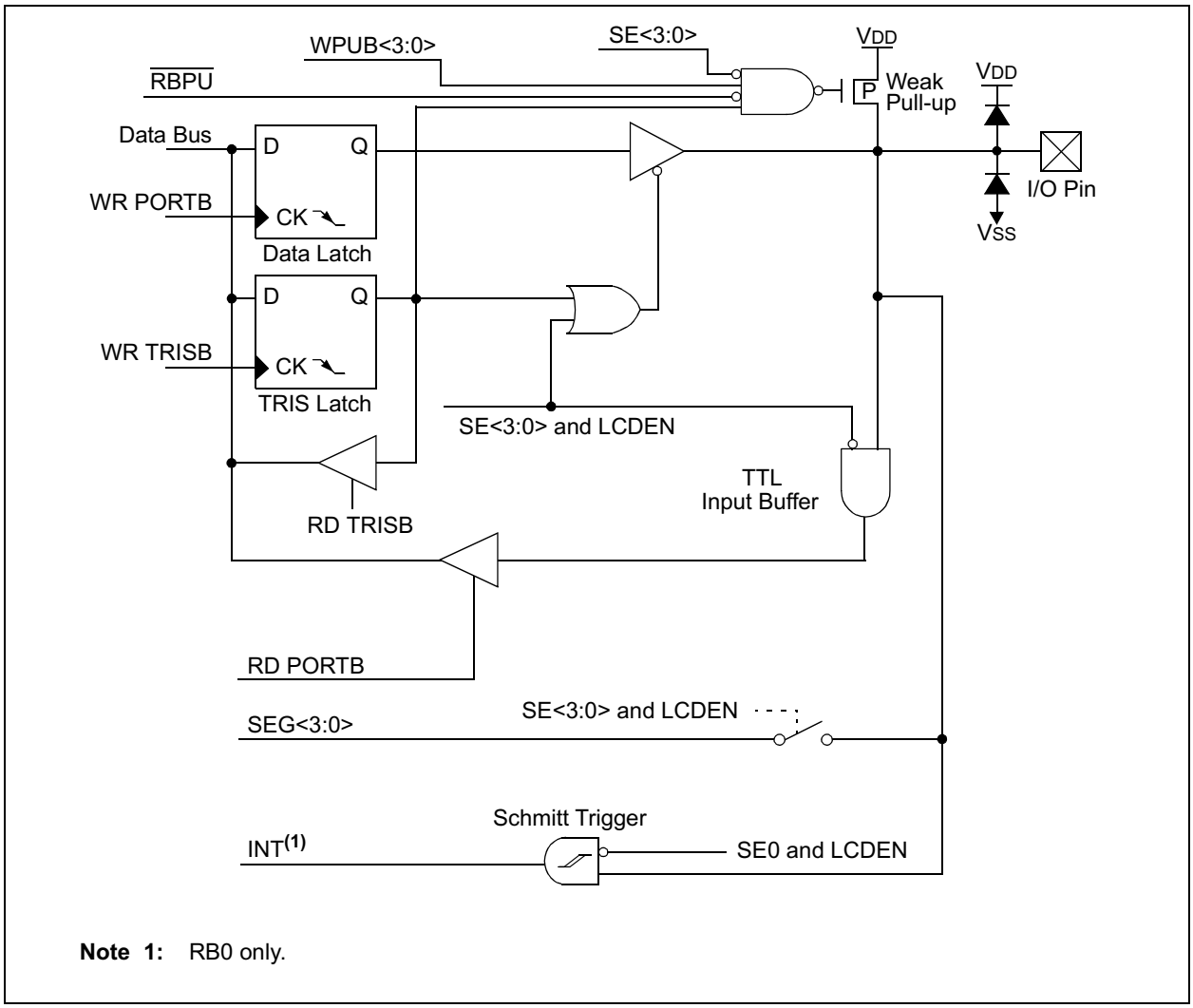
- a general purpose I/O
- an analog output for the LCD

3.4.3.4 RB3/SEG3

Figure 3-9 shows the diagram for this pin. The RB3 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

FIGURE 3-9: BLOCK DIAGRAM OF RB<3:0>



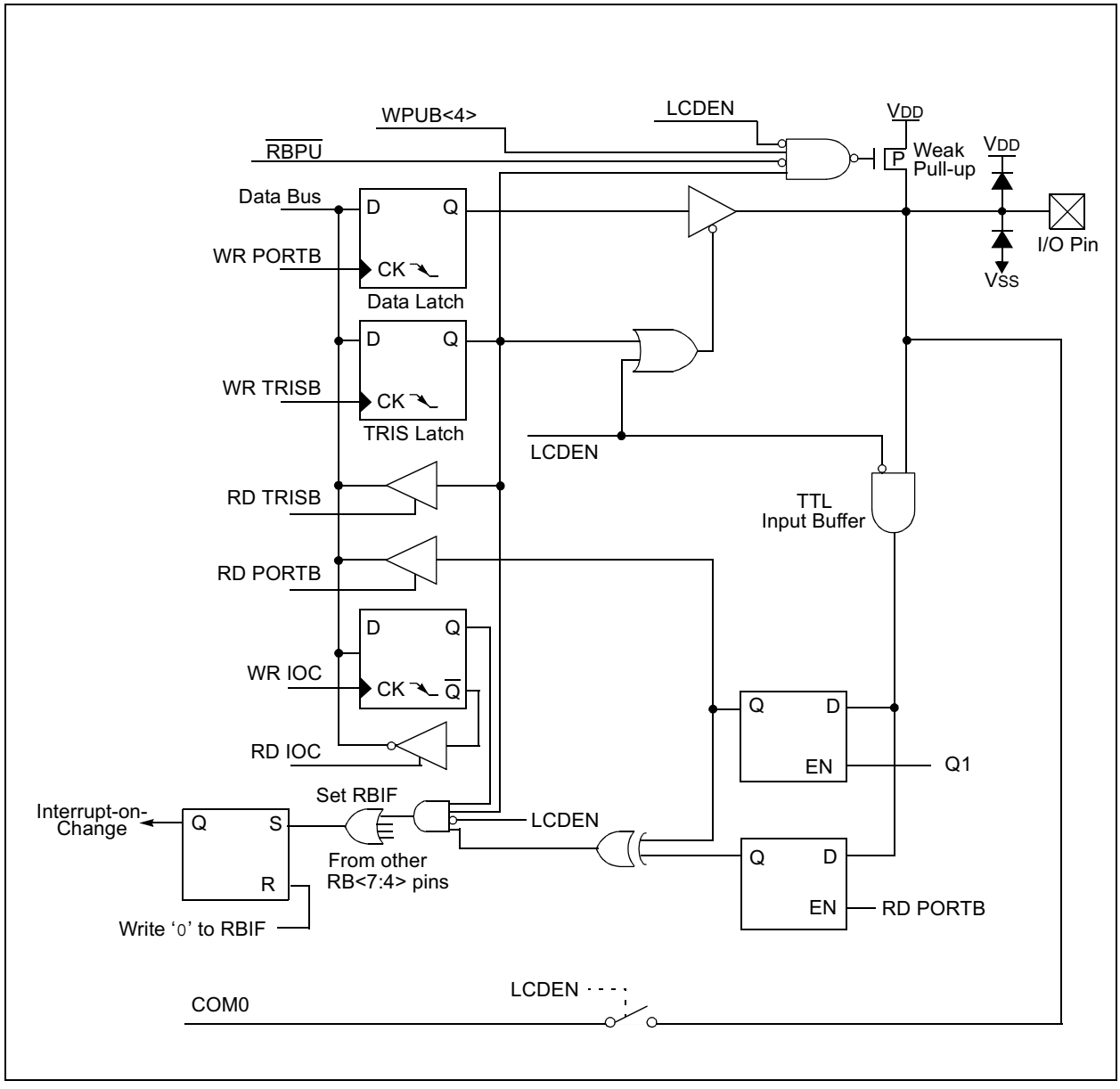
Note 1: RB0 only.

3.4.3.5 RB4/COM0

Figure 3-10 shows the diagram for this pin. The RB4 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

FIGURE 3-10: BLOCK DIAGRAM OF RB4

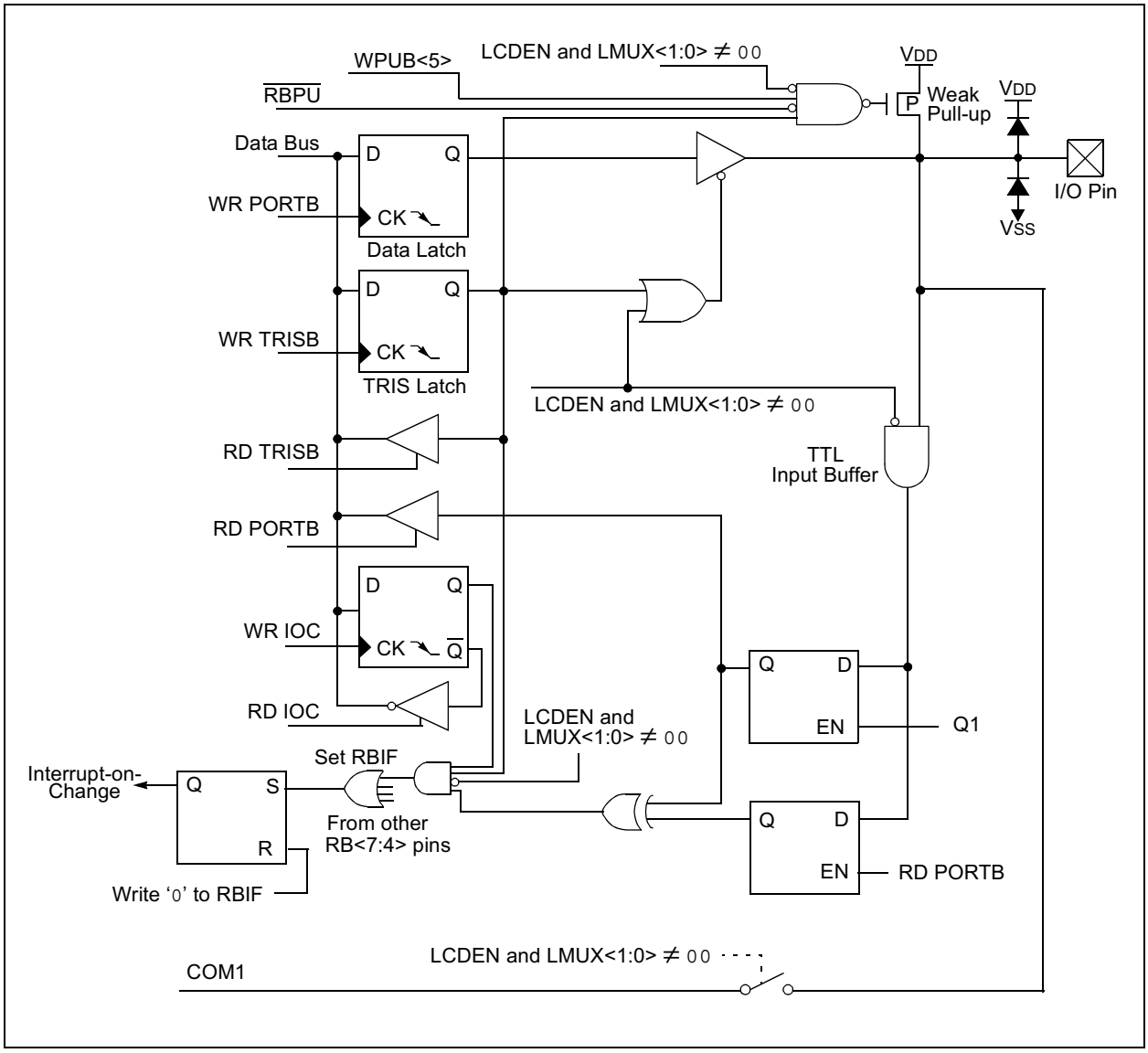


3.4.3.6 RB5/COM1

Figure 3-11 shows the diagram for this pin. The RB5 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

FIGURE 3-11: BLOCK DIAGRAM OF RB5

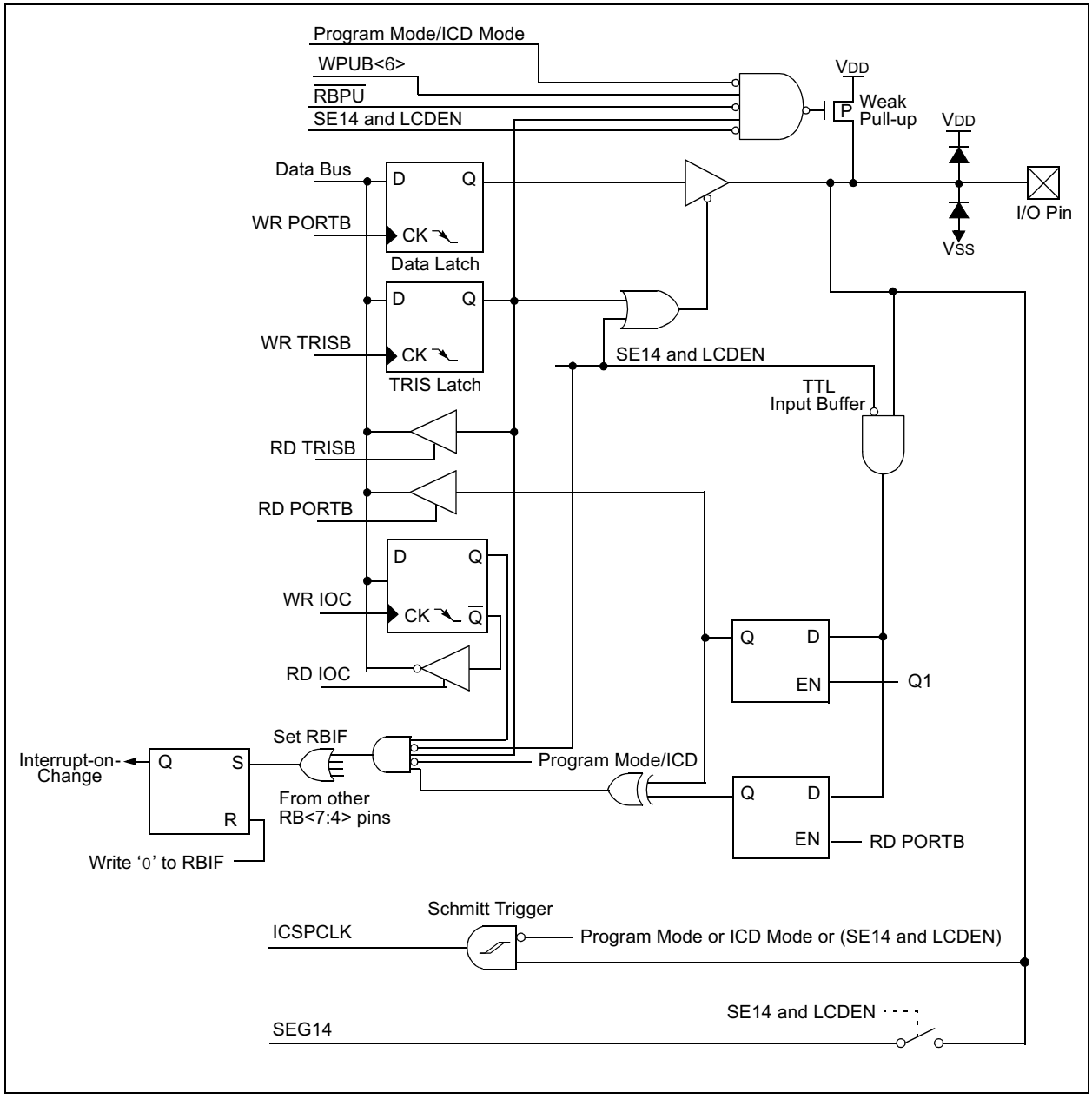


3.4.3.7 RB6/ICSPCLK/ICDCK/SEG14

Figure 3-12 shows the diagram for this pin. The RB6 pin is configurable to function as one of the following:

- a general purpose I/O
- an In-Circuit Serial Programming™ clock
- an ICD clock input
- an analog output for the LCD

FIGURE 3-12: BLOCK DIAGRAM OF RB6



3.4.3.8 RB7/ICSPDAT/ICDDAT/SEG13

Figure 3-13 shows the diagram for this pin. The RB7 pin is configurable to function as one of the following:

- a general purpose I/O
- an In-Circuit Serial Programming™ I/O
- an ICD data I/O
- an analog output for the LCD

FIGURE 3-13: BLOCK DIAGRAM OF RB7

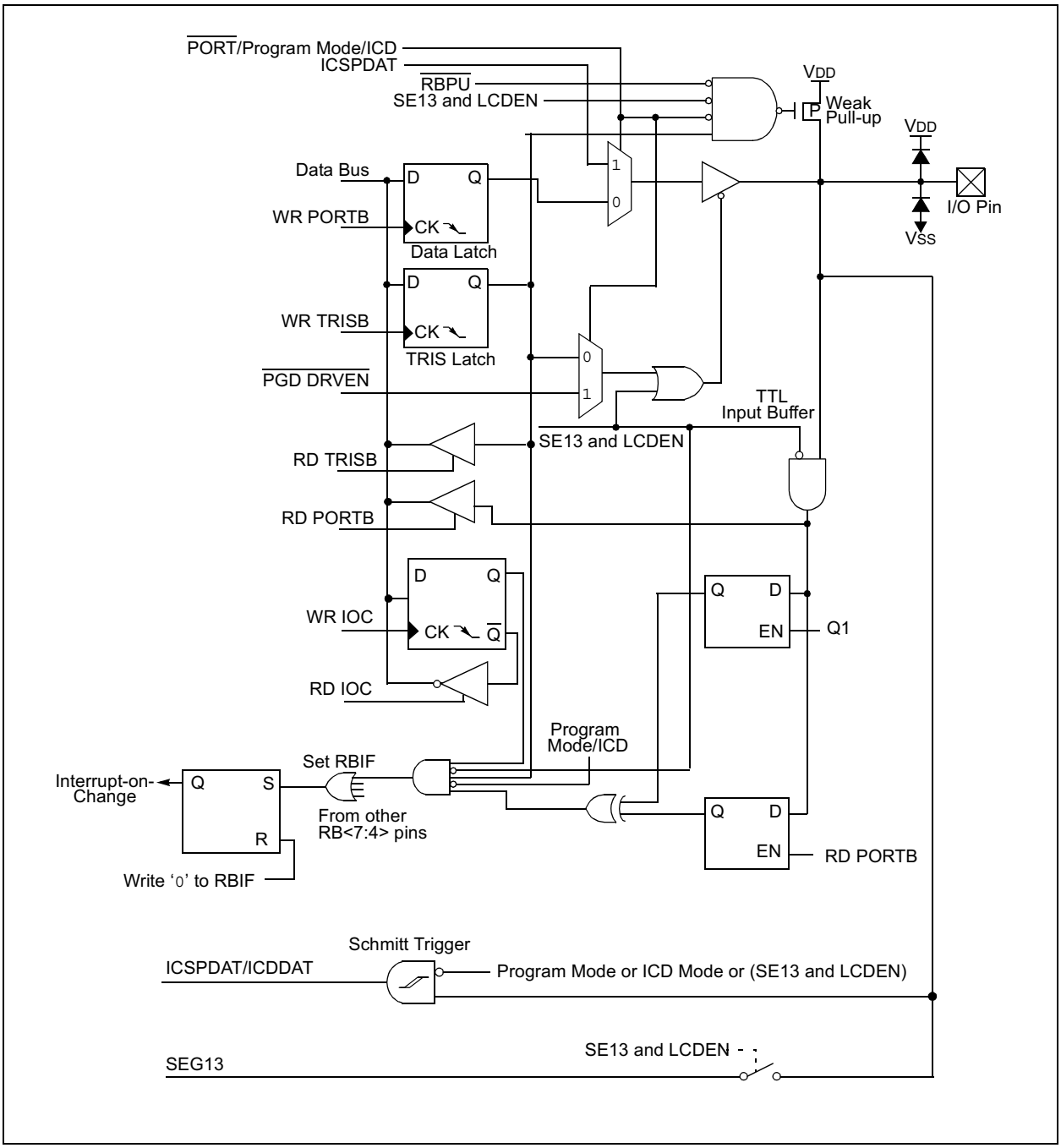


TABLE 3-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
IOCB	IOCB7	IOCB6	IOCB5	IOCB4	—	—	—	—	0000 ----	0000 ----
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE0	SE7	SE6	SE5	SE4	SE3	SE2	SE1	SE0	0000 0000	uuuu uuuu
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	uuuu uuuu
OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
WPUB	WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTB.
Note 1: This register is only initialized by a POR or BOR reset and is unchanged by other Resets.
2: Configuration Word register bit $\overline{\text{DEBUG}}$ <12> is also associated with PORTB. See Register 16-1 for more details.

3.5 PORTC and TRISC Registers

PORTC is an 8-bit bidirectional port. PORTC is multiplexed with several peripheral functions. PORTC pins have Schmitt Trigger input buffers.

All PORTC pins have latch bits (PORTC register). They will modify the contents of the PORTC latch (when written); thus, modifying the value driven out on a pin if the corresponding TRISC bit is configured for output.

EXAMPLE 3-3: INITIALIZING PORTC

```
BANKSEL PORTC       ;
CLRF    PORTC       ;Init PORTC
BANKSEL TRISC       ;
MOVLW   0FFh        ;Set RC<7:0> as inputs
MOVWF   TRISC       ;
BANKSEL LCDCON      ;
CLRF    LCDCON      ;Disable VLCD<3:1>
                    ;inputs on RC<2:0>
```

REGISTER 3-8: PORTC: PORTC REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 7-0 **RC<7:0>**: PORTC I/O Pin bits
1 = Port pin is >VIH min.
0 = Port pin is <VIL max.

REGISTER 3-9: TRISC: PORTC TRI-STATE REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 7-0 **TRISC<7:0>:** PORTC Tri-State Control bits
1 = PORTC pin configured as an input (tri-stated)
0 = PORTC pin configured as an output

3.5.1 PIN DESCRIPTIONS AND DIAGRAMS

Each PORTC pin is multiplexed with other functions. The pins and their combined functions are briefly described here. For specific information about individual functions such as the LCD or SSP, refer to the appropriate section in this data sheet.

3.5.1.1 RC0/VLCD1

Figure 3-14 shows the diagram for this pin. The RC0 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the LCD bias voltage

3.5.1.2 RC1/VLCD2

Figure 3-15 shows the diagram for this pin. The RC1 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the LCD bias voltage

3.5.1.3 RC2/VLCD3

Figure 3-16 shows the diagram for this pin. The RC2 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the LCD bias voltage

FIGURE 3-14: BLOCK DIAGRAM OF RC0

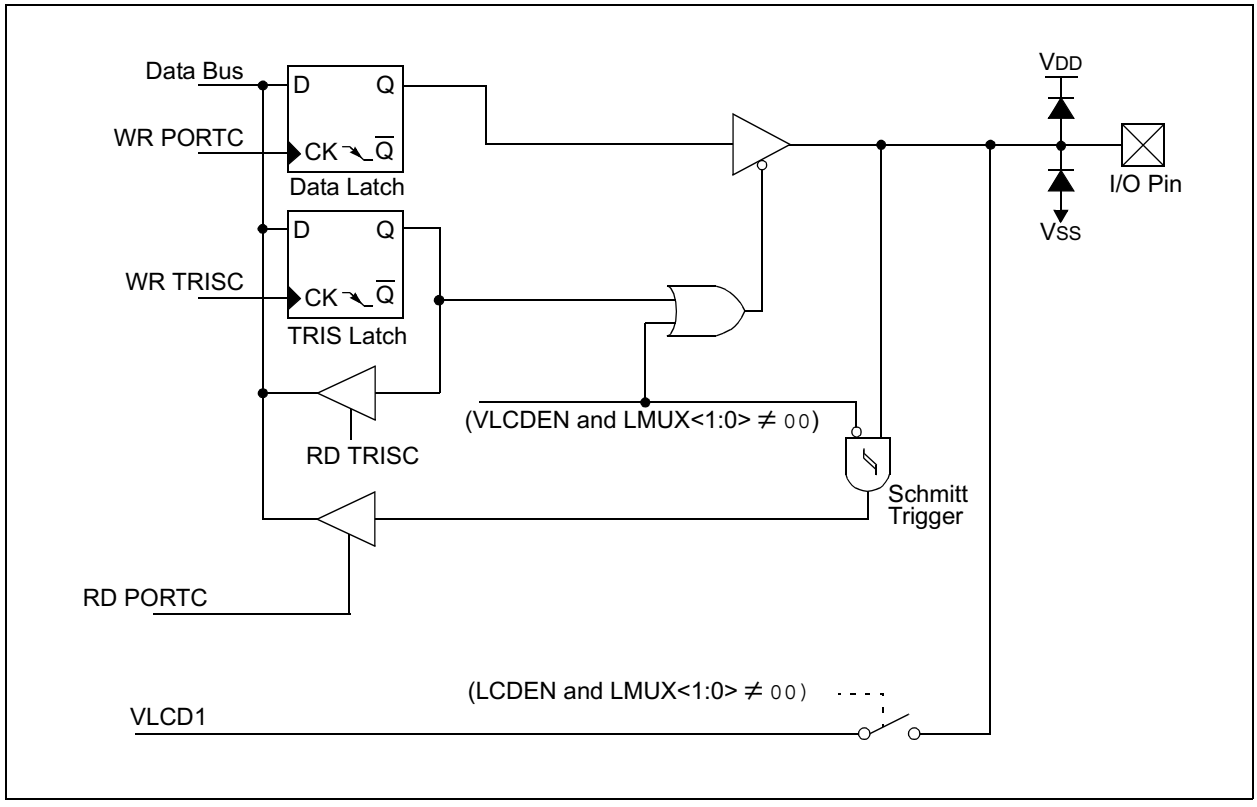


FIGURE 3-15: BLOCK DIAGRAM OF RC1

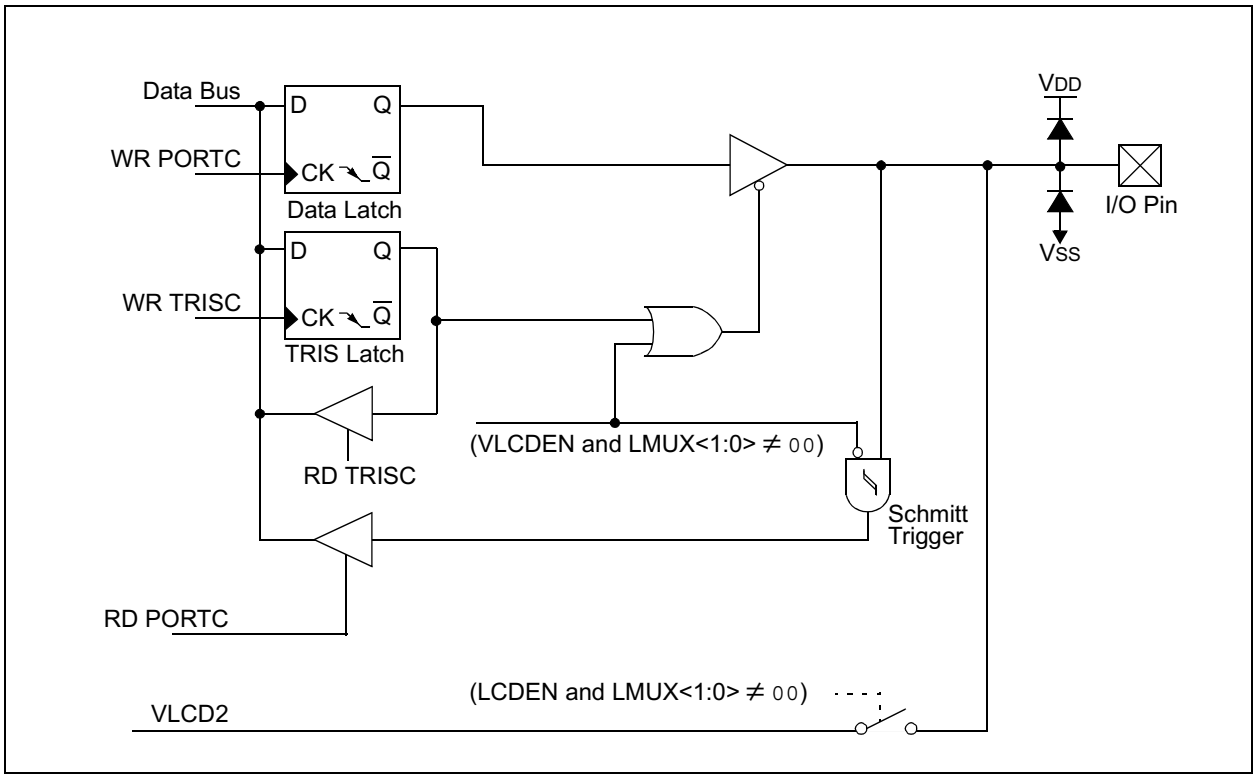
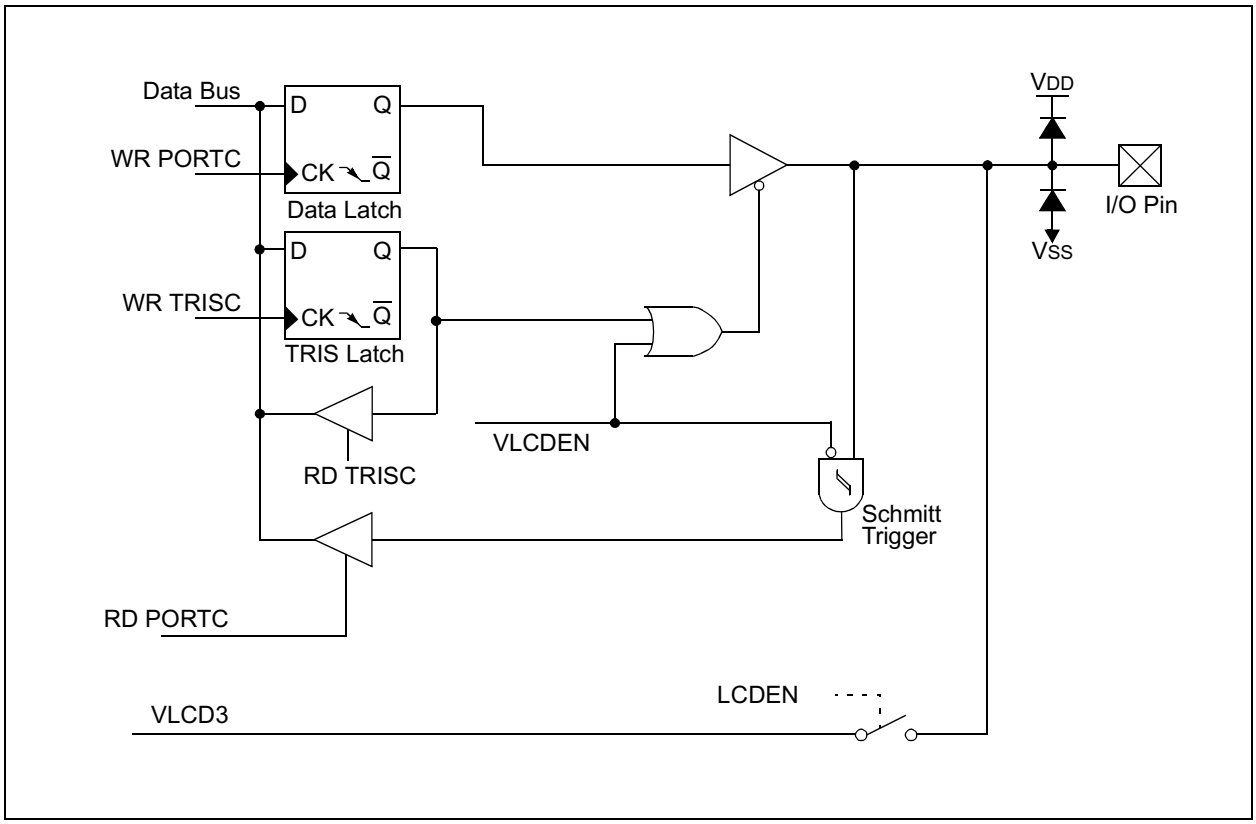


FIGURE 3-16: BLOCK DIAGRAM OF RC2

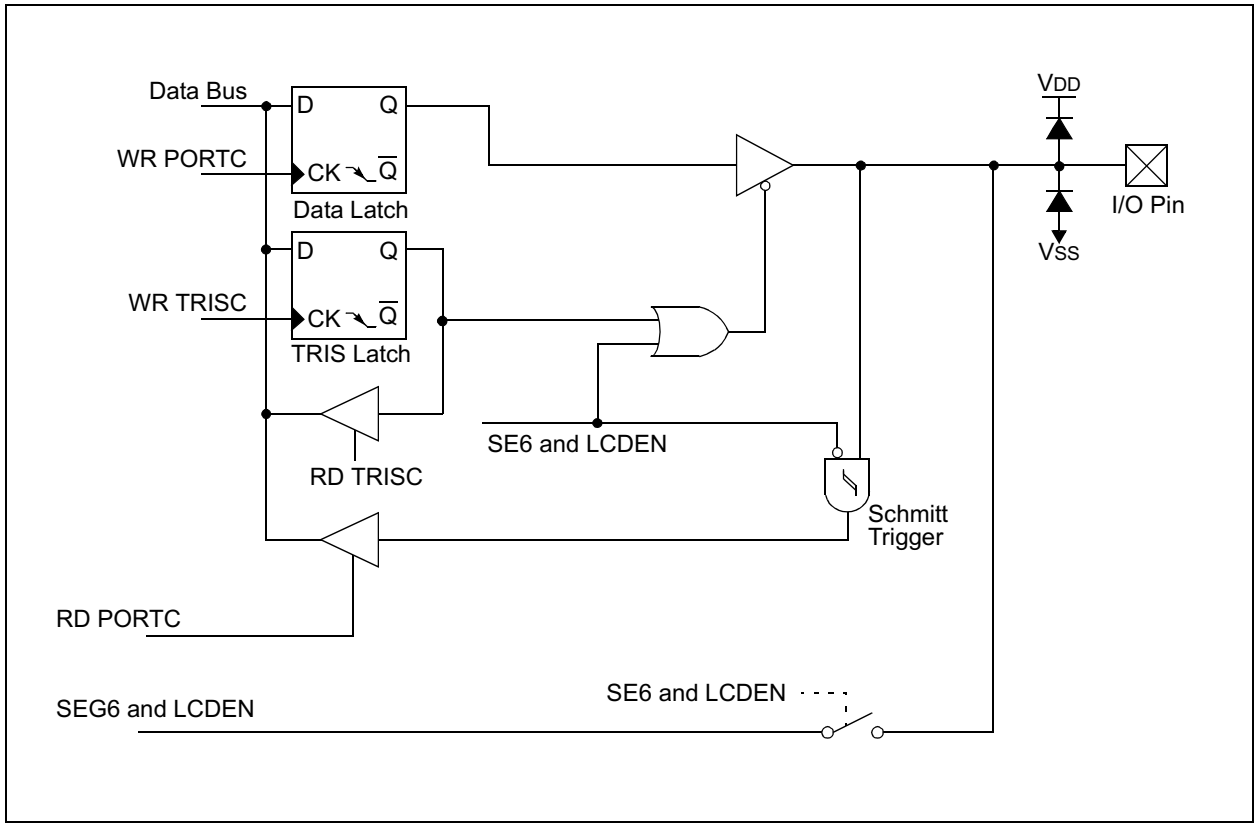


3.5.1.4 RC3/SEG6

Figure 3-17 shows the diagram for this pin. The RC3 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

FIGURE 3-17: BLOCK DIAGRAM OF RC3

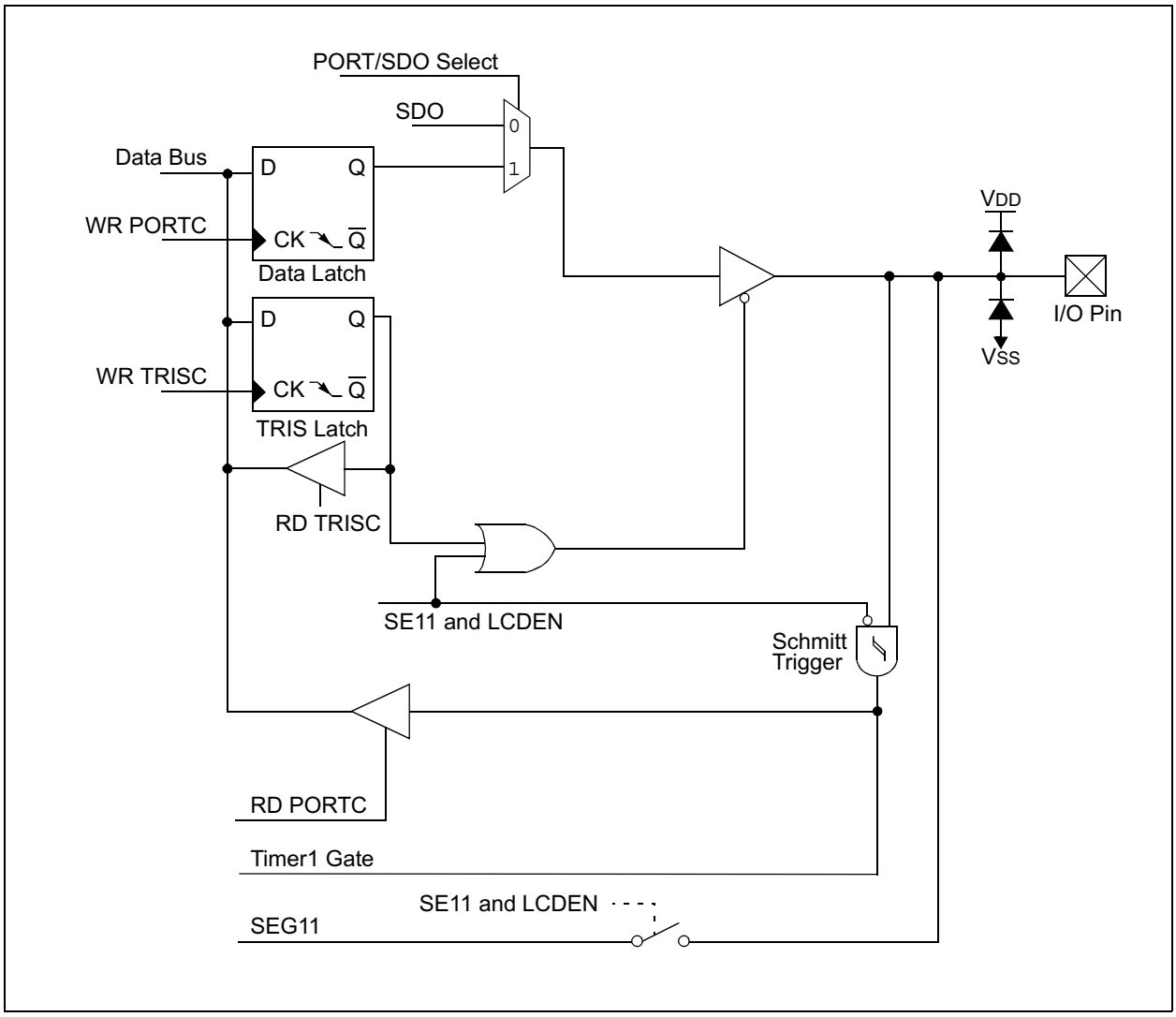


3.5.1.5 RC4/$\overline{T1G}$/SDO/SEG11

Figure 3-18 shows the diagram for this pin. The RC4pin is configurable to function as one of the following:

- a general purpose I/O
- a Timer1 gate input
- a serial data output
- an analog output for the LCD

FIGURE 3-18: BLOCK DIAGRAM OF RC4

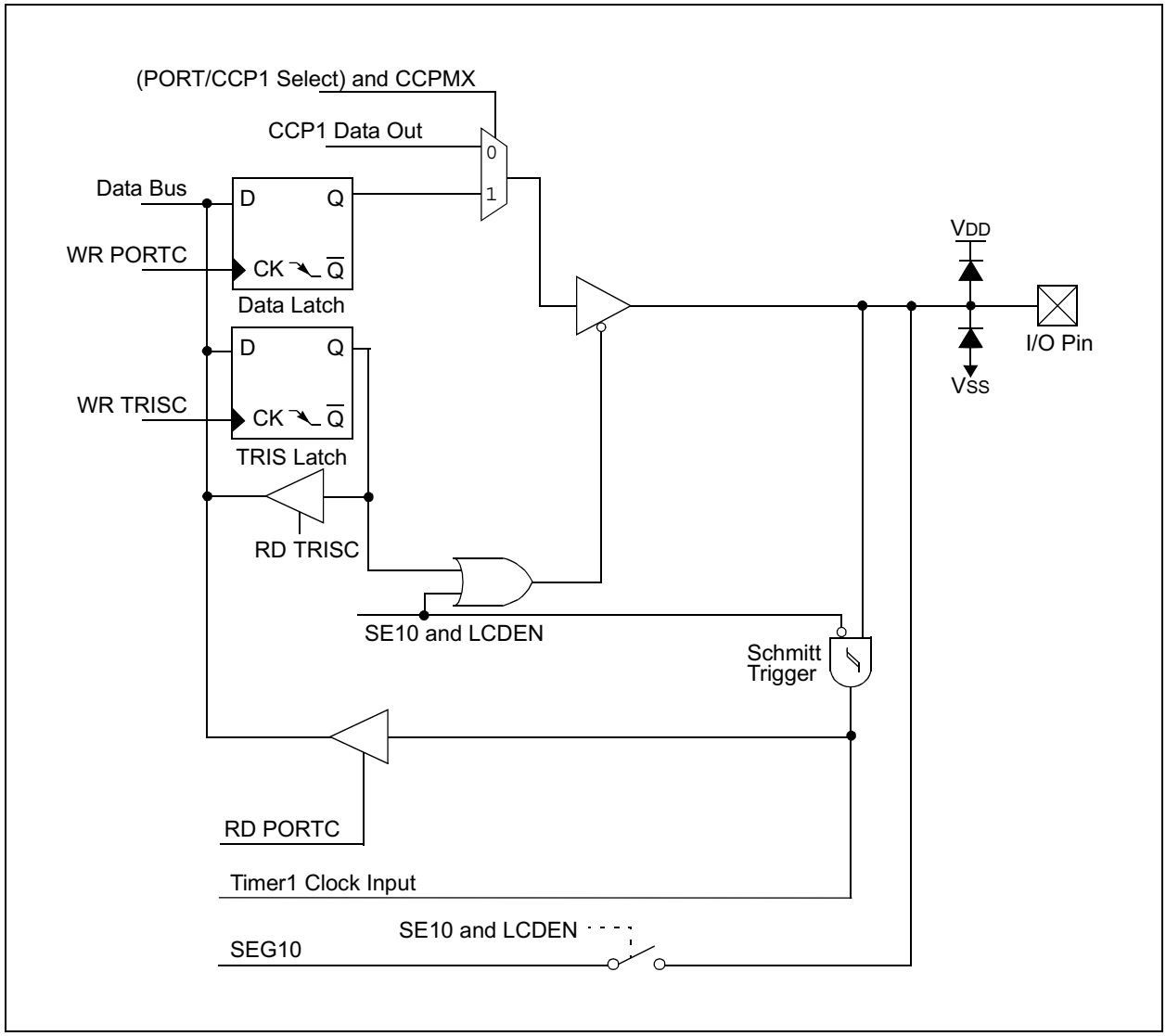


3.5.1.6 RC5/T1CKI/CCP1/SEG10

Figure 3-19 shows the diagram for this pin. The RC5 pin is configurable to function as one of the following:

- a general purpose I/O
- a Timer1 clock input
- a Capture input, Compare output or PWM output
- an analog output for the LCD

FIGURE 3-19: BLOCK DIAGRAM OF RC5

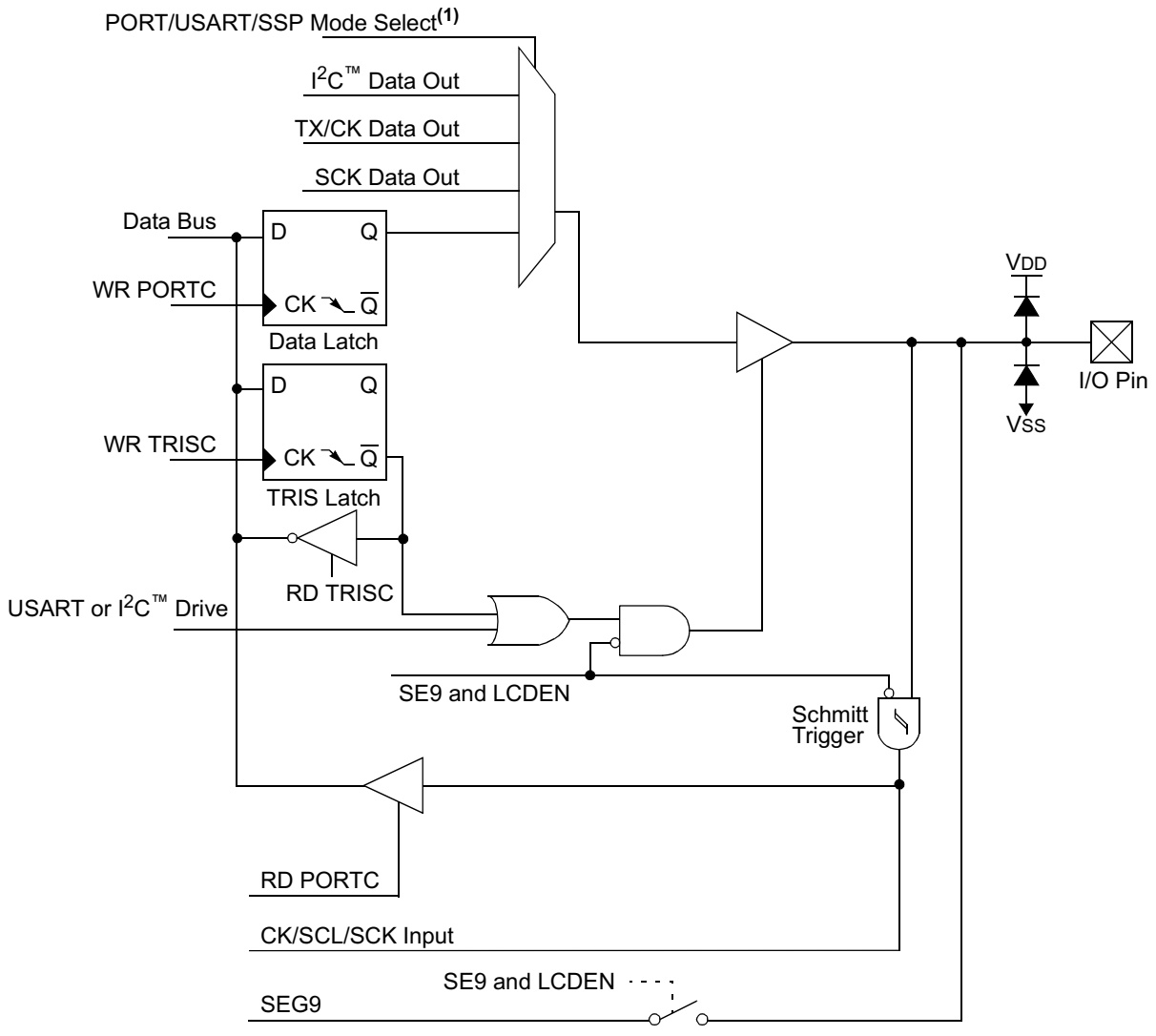


3.5.1.7 RC6/TX/CK/SCK/SCL/SEG9

Figure 3-20 shows the diagram for this pin. The RC6 pin is configurable to function as one of the following:

- a general purpose I/O
- an asynchronous serial output
- a synchronous clock I/O
- a SPI clock I/O
- an I^2C data I/O
- an analog output for the LCD

FIGURE 3-20: BLOCK DIAGRAM OF RC6



Note 1: If all three data output sources are enabled, the following priority order will be used:

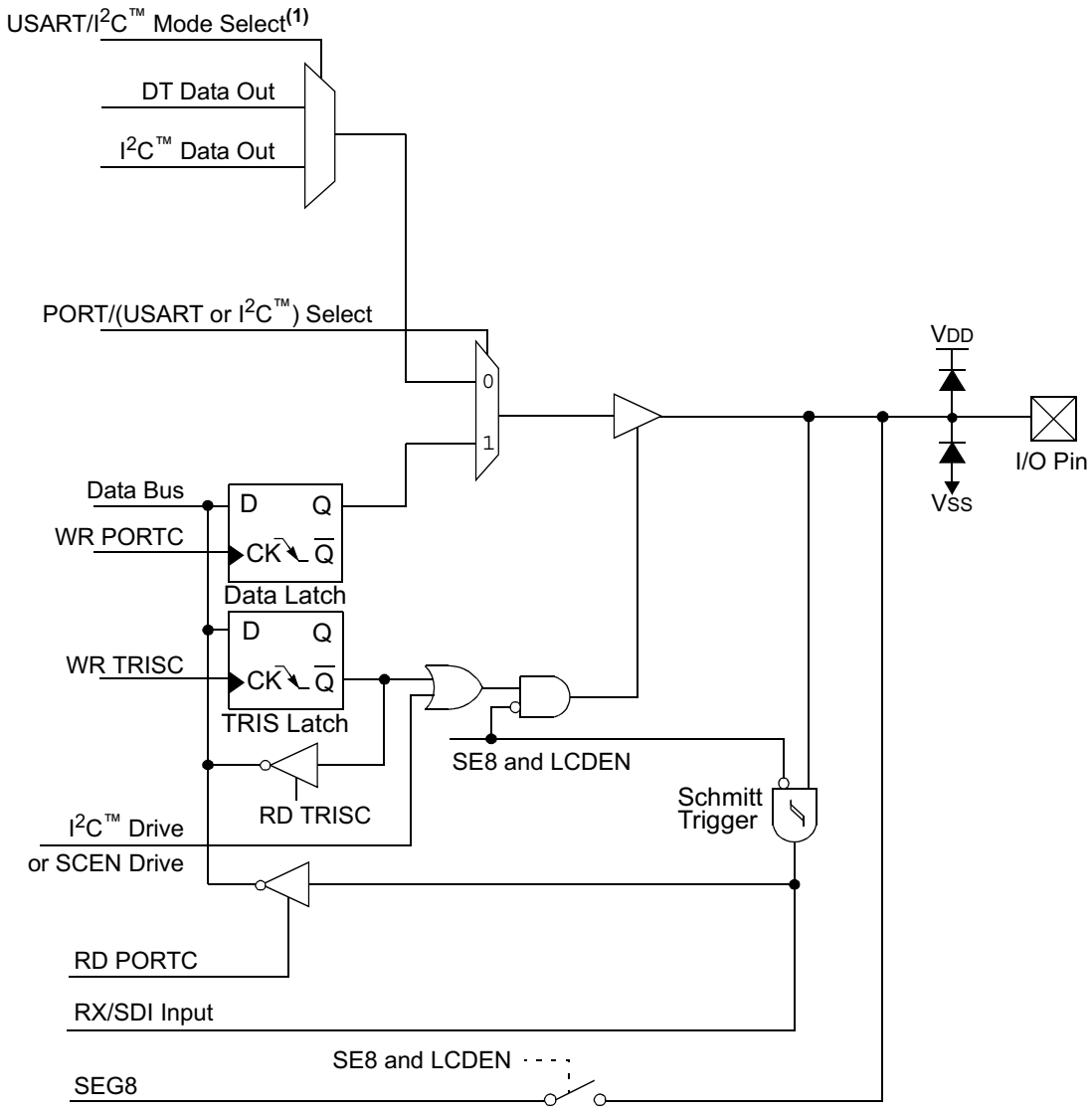
- USART data (highest)
- SSP data
- PORT data (lowest)

3.5.1.8 RC7/RX/DT/SDI/SDA/SEG8

Figure 3-21 shows the diagram for this pin. The RC7 pin is configurable to function as one of the following:

- a general purpose I/O
- an asynchronous serial input
- a synchronous serial data I/O
- a SPI data input
- an I^2C data I/O
- an analog output for the LCD

FIGURE 3-21: BLOCK DIAGRAM OF RC7



Note 1: If all three data output sources are enabled, the following priority order will be used:

- USART data (highest)
- SSP data
- PORT data (lowest)

TABLE 3-3: SUMMARY OF REGISTERS ASSOCIATED WITH PORTC

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE0	SE7	SE6	SE5	SE4	SE3	SE2	SE1	SE0	0000 0000	uuuu uuuu
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	uuuu uuuu
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
T1CON	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	0000 0000	uuuu uuuu
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTC.

3.6 PORTD and TRISD Registers

PORTD is an 8-bit port with Schmitt Trigger input buffers. Each pin is individually configured as an input or output. PORTD is only available on the PIC16F914/917 and PIC16F946.

EXAMPLE 3-4: INITIALIZING PORTD

```
BANKSEL PORTD       ;
CLRF    PORTD       ;Init PORTD
BANKSEL TRISD       ;
MOVLW   0FF         ;Set RD<7:0> as inputs
MOVWF   TRISD       ;
```

REGISTER 3-10: PORTD: PORTD REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 7-0 **RD<7:0>**: PORTD I/O Pin bits
1 = Port pin is >VIH min.
0 = Port pin is <VIL max.

REGISTER 3-11: TRISD: PORTD TRI-STATE REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 7-0 **TRISD<7:0>:** PORTD Tri-State Control bits
1 = PORTD pin configured as an input (tri-stated)
0 = PORTD pin configured as an output

3.6.1 PIN DESCRIPTIONS AND DIAGRAMS

Each PORTD pin is multiplexed with other functions. The pins and their combined functions are briefly described here. For specific information about individual functions such as the Comparator or the ADC, refer to the appropriate section in this data sheet.

3.6.1.1 RD0/COM3

Figure 3-22 shows the diagram for this pin. The RD0 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.6.1.2 RD1

Figure 3-23 shows the diagram for this pin. The RD1 pin is configurable to function as one of the following:

- a general purpose I/O

3.6.1.3 RD2/CCP2

Figure 3-24 shows the diagram for this pin. The RD2 pin is configurable to function as one of the following:

- a general purpose I/O
- a Capture input, Compare output or PWM output

3.6.1.4 RD3/SEG16

Figure 3-25 shows the diagram for this pin. The RD3 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.6.1.5 RD4/SEG17

Figure 3-25 shows the diagram for this pin. The RD4 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.6.1.6 RD5/SEG18

Figure 3-25 shows the diagram for this pin. The RD5 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.6.1.7 RD6/SEG19

Figure 3-25 shows the diagram for this pin. The RD6 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.6.1.8 RD7/SEG20

Figure 3-25 shows the diagram for this pin. The RD7 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

FIGURE 3-22: BLOCK DIAGRAM OF RD0

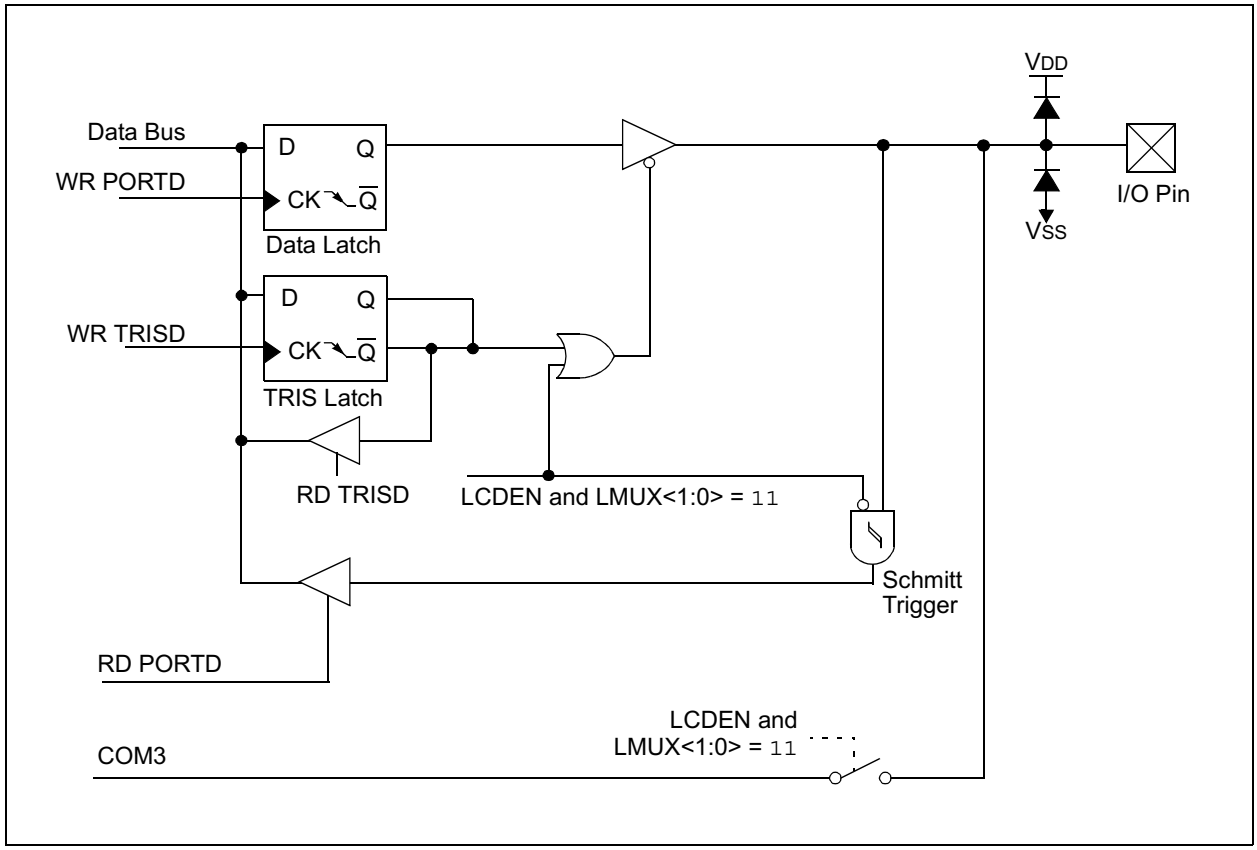


FIGURE 3-23: BLOCK DIAGRAM OF RD1

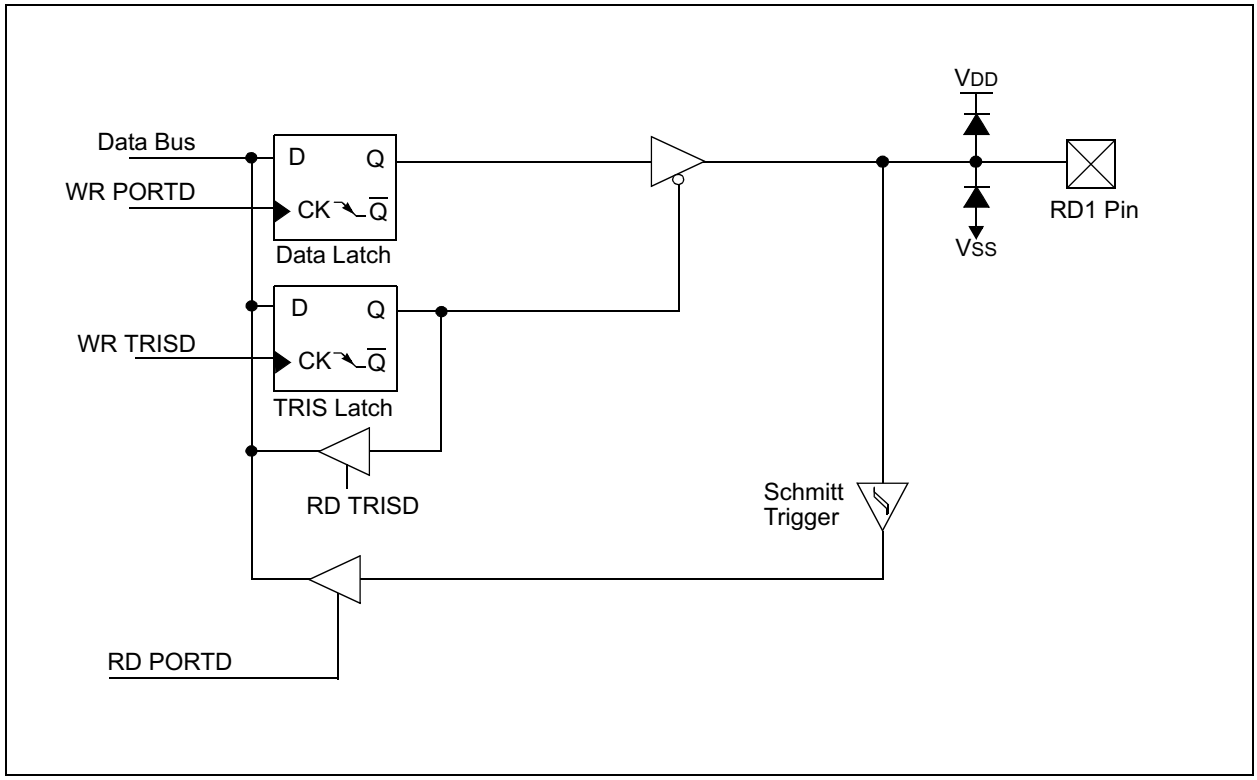


FIGURE 3-24: BLOCK DIAGRAM OF RD2

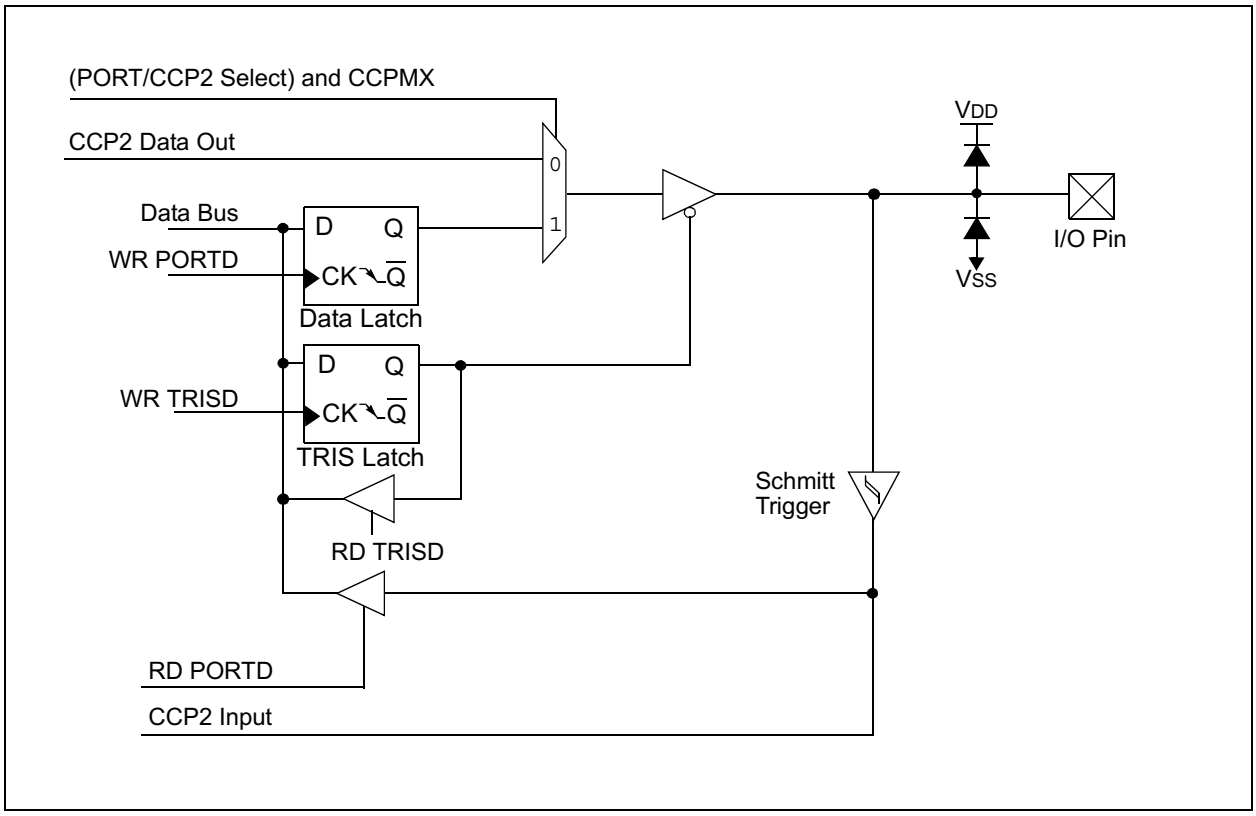


FIGURE 3-25: BLOCK DIAGRAM OF RD<7:3>

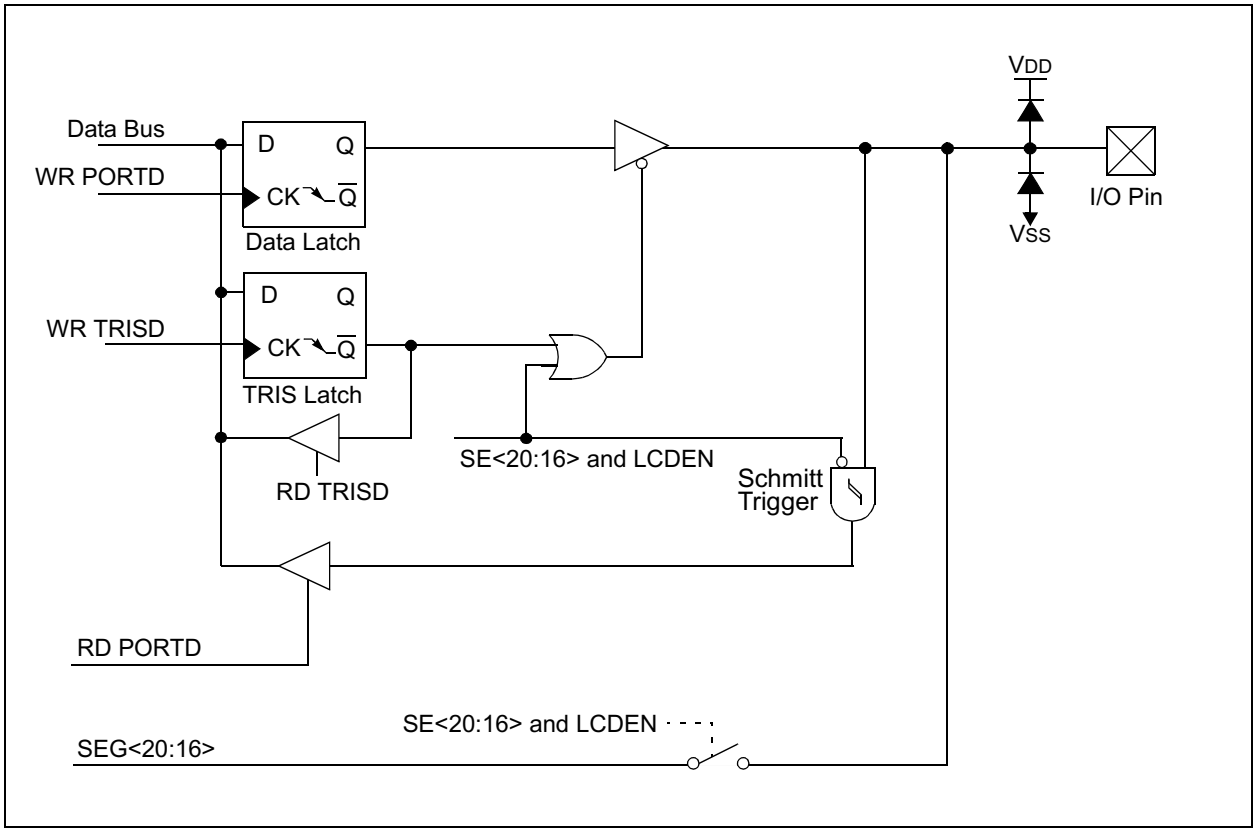


TABLE 3-4: SUMMARY OF REGISTERS ASSOCIATED WITH PORTD(1)

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
CCP2CON(1)	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE2(1)	SE23	SE22	SE21	SE20	SE19	SE18	SE17	SE16	0000 0000	uuuu uuuu
PORTD(1)	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx xxxx	uuuu uuuu
TRISD(1)	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTD.
Note 1: PIC16F914/917 and PIC16F946 only.

3.7 PORTE and TRISE Registers

PORTE is a 1-bit, 4-bit or 8-bit port with Schmitt Trigger input buffers. RE<7:4, 2:0> are individually configured as inputs or outputs and RE3 is only available as an input if MCLRE is '0' in Configuration Word (Register 16-1).

RE<2:0> are only available on the PIC16F914/917 and PIC16F946. RE<7:4> are only available on the PIC16F946.

EXAMPLE 3-5: INITIALIZING PORTE

```
BANKSEL PORTE          ;
CLRF    PORTE          ;Init PORTE
BANKSEL TRISE          ;
MOVLW   0Fh            ;Set RE<3:0> as inputs
MOVWF   TRISE          ;
CLRF    ANSEL          ;Make RE<2:0> as I/O's
```

REGISTER 3-12: PORTE: PORTE REGISTER

R/W-x	R/W-x	R/W-x	R/W-x	R-x	R/W-x	R/W-x	R/W-x
RE7(1,3)	RE6(1,3)	RE5(1,3)	RE4(1,3)	RE3	RE2(2,4)	RE1(2,4)	RE0(2,4)
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **RE<7:0>**: PORTE I/O Pin bits
1 = Port pin is >VIH min.
0 = Port pin is <VIL max.

Note 1: PIC16F946 only.
2: PIC16F914/917 and PIC16F946 only.
3: PIC16F91X, Read as '0'.
4: PIC16F913/916, Read as '0'.

REGISTER 3-13: TRISE: PORTE TRI-STATE REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R-1	R/W-1	R/W-1	R/W-1
TRISE7(1,3)	TRISE6(1,3)	TRISE5(1,3)	TRISE4(1,3)	TRISE3	TRISE2(2,4)	TRISE1(2,4)	TRISE0(2,4)
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **TRISE<7:0>:** PORTE Tri-State Control bits
1 = PORTE pin configured as an input (tri-stated)
0 = PORTE pin configured as an output

Note 1: PIC16F946 only.
2: PIC16F914/917 and PIC16F946 only.
3: PIC16F91X, Read as '0'.
4: PIC16F913/916, Read as '0'.

3.7.1 PIN DESCRIPTIONS AND DIAGRAMS

Each PORTE pin is multiplexed with other functions. The pins and their combined functions are briefly described here. For specific information about individual functions such as the Comparator or the ADC, refer to the appropriate section in this data sheet.

3.7.1.1 RE0/AN5/SEG21(1)

Figure 3-26 shows the diagram for this pin. The RE0 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the ADC
- an analog output for the LCD

3.7.1.2 RE1/AN6/SEG22(1)

Figure 3-26 shows the diagram for this pin. The RE1 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the ADC
- an analog output for the LCD

3.7.1.3 RE2/AN7/SEG23(1)

Figure 3-26 shows the diagram for this pin. The RE2 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the ADC
- an analog output for the LCD

3.7.1.4 RE3/$\overline{\text{MCLR}}$/VPP

Figure 3-27 shows the diagram for this pin. The RE3 pin is configurable to function as one of the following:

- a digital input only
- as Master Clear Reset with weak pull-up
- a programming voltage reference input

3.7.1.5 RE4/SEG24(2)

Figure 3-28 shows the diagram for this pin. The RE4/SEG24 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.7.1.6 RE5/SEG25(2)

Figure 3-28 shows the diagram for this pin. The RE5/SEG25 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.7.1.7 RE6/SEG26(2)

Figure 3-28 shows the diagram for this pin. The RE6/SEG26 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.7.1.8 RE7/SEG27(2)

Figure 3-28 shows the diagram for this pin. The RE7/SEG27 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

Note 1: Pin is available on the PIC16F914/917 and PIC16F946 only.

2: Pin is available on the PIC16F946 only.

FIGURE 3-26: BLOCK DIAGRAM OF RE<2:0> (PIC16F914/917 AND PIC16F946 ONLY)

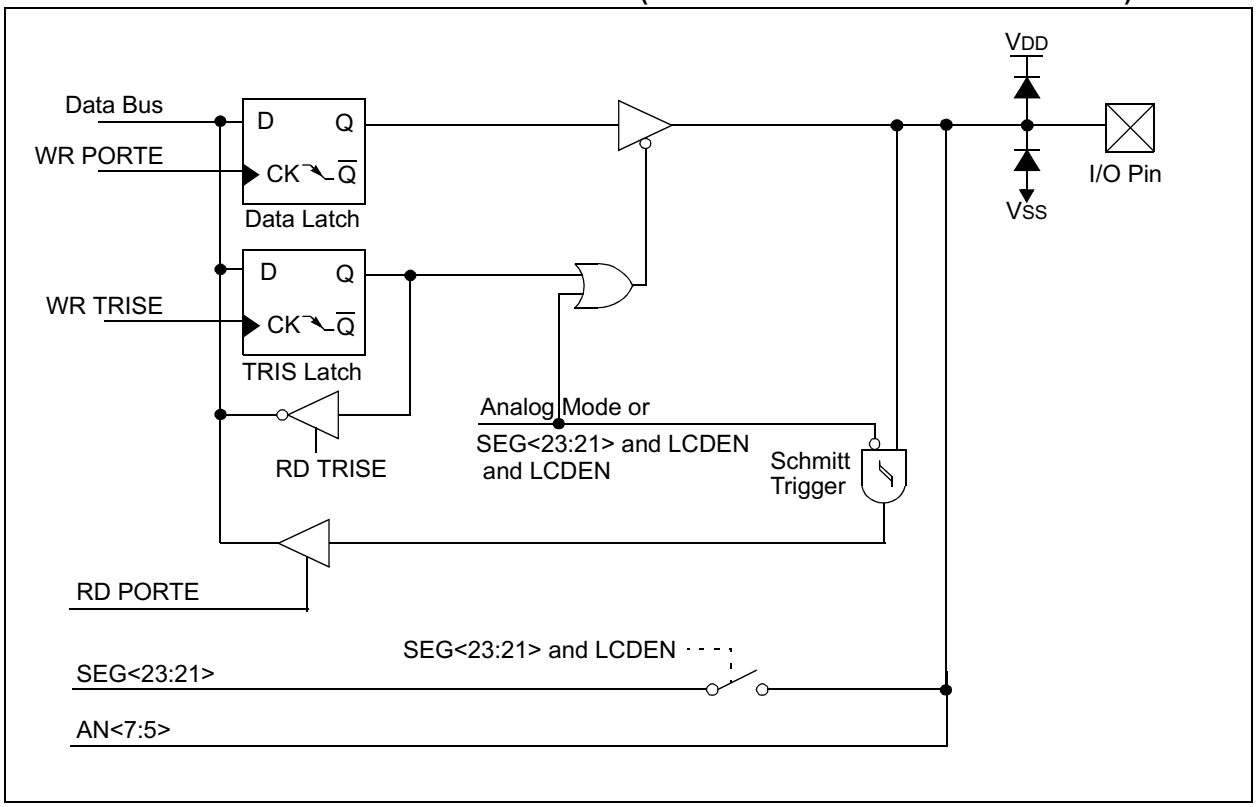


FIGURE 3-27: BLOCK DIAGRAM OF RE3

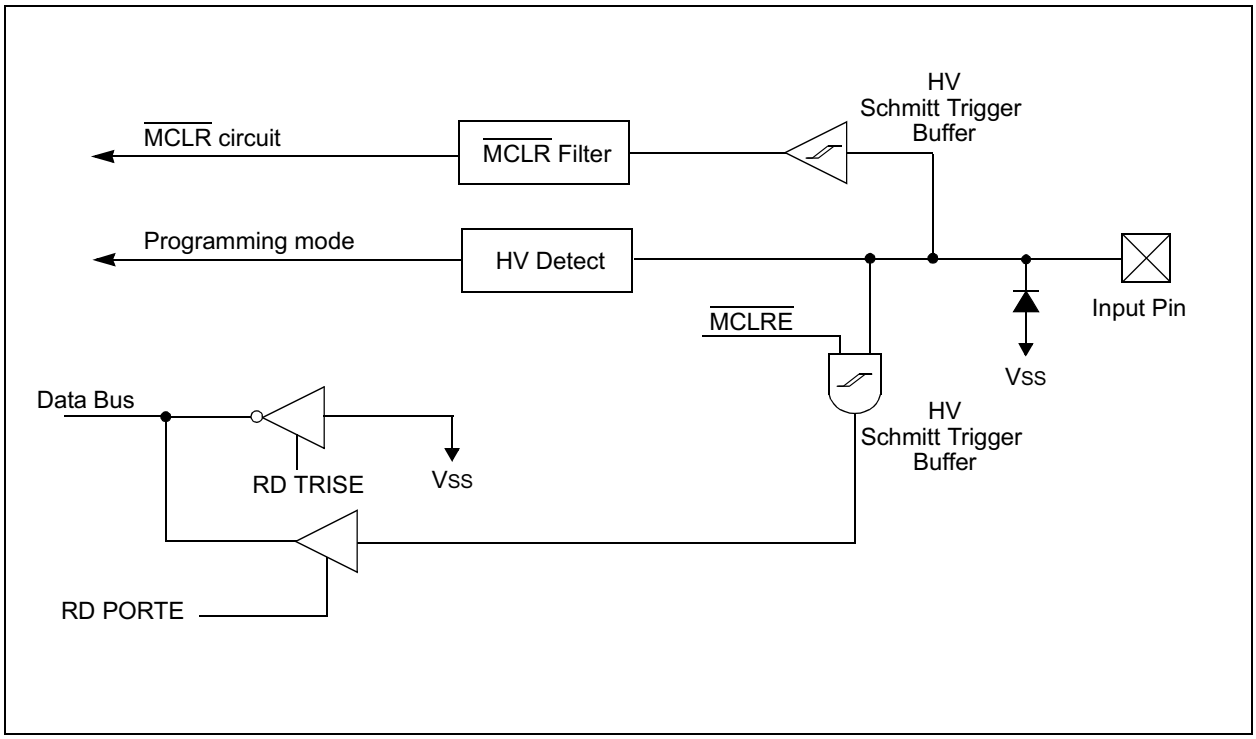


FIGURE 3-28: BLOCK DIAGRAM OF RE<7:4> (PIC16F946 ONLY)

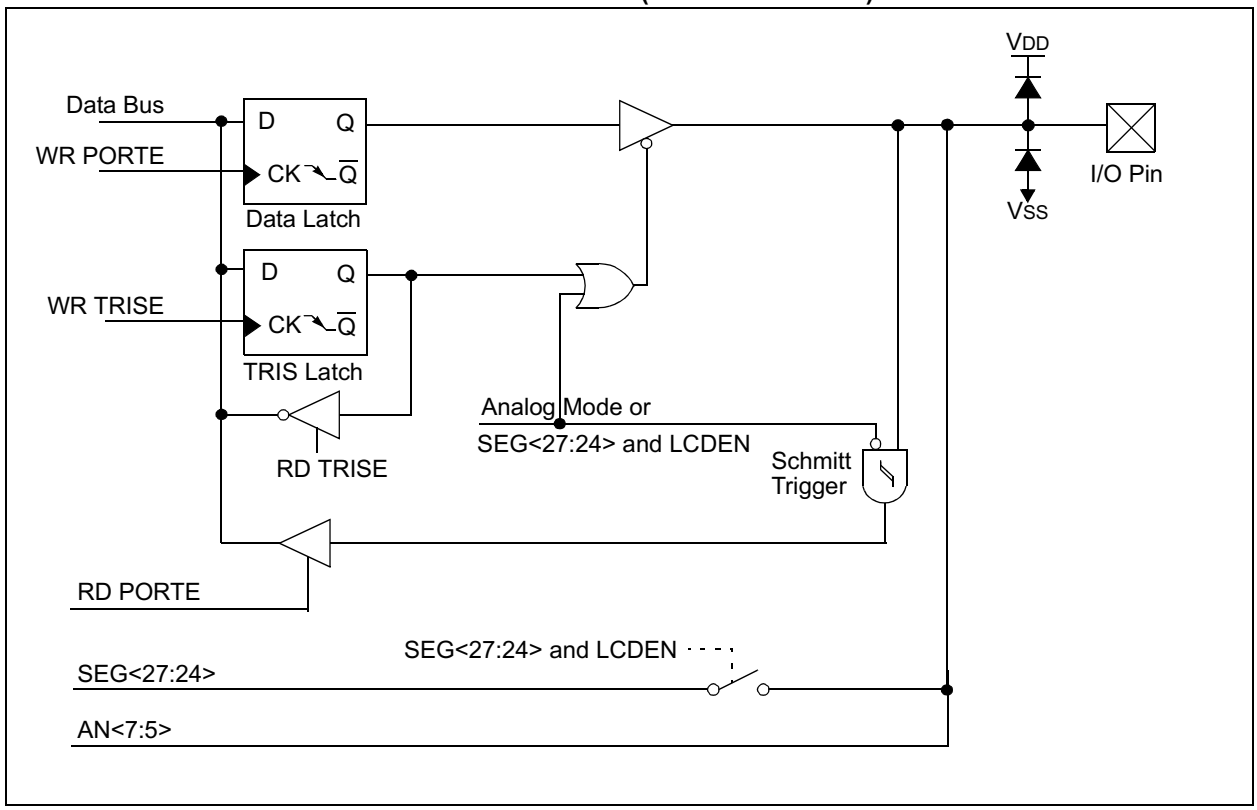


TABLE 3-5: SUMMARY OF REGISTERS ASSOCIATED WITH PORTE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
ADCON0	ADFM	VCFG1	VCFG0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	ADON	0000 0000	0000 0000
ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE2(1,2)	SE23	SE22	SE21	SE20	SE19	SE18	SE17	SE16	0000 0000	uuuu uuuu
LCDSE3(1, 3)	SE31	SE30	SE29	SE28	SE27	SE26	SE25	SE24	0000 0000	uuuu uuuu
PORTE	RE7(3)	RE6(3)	RE5(3)	RE4(3)	RE3	RE2(2)	RE1(2)	RE0(2)	xxxx xxxx	uuuu uuuu
TRISE	TRISE7(3)	TRISE6(3)	TRISE5(3)	TRISE4(3)	TRISE3(4)	TRISE2(2)	TRISE1(2)	TRISE0(2)	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTC.

Note 1: This register is only initialized by a POR or BOR reset and is unchanged by other Resets.

2: PIC16F914/917 and PIC16F946 only.

3: PIC16F946 only.

4: Bit is read-only; TRISE = 1 always.

3.8 PORTF and TRISF Registers

PORTF is an 8-bit port with Schmitt Trigger input buffers. RF<7:0> are individually configured as inputs or outputs, depending on the state of the port direction. The port bits are also multiplexed with LCD segment functions. PORTF is available on the PIC16F946 only.

EXAMPLE 3-6: INITIALIZING PORTF

```
BANKSEL PORTF       ;
CLRF    PORTF       ;Init PORTF
BANKSEL TRISF       ;
MOVLW   0FFh        ;Set RF<7:0> as inputs
MOVWF   TRISF       ;
```

REGISTER 3-14: PORTF: PORTF REGISTER(1)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
RF7	RF6	RF5	RF4	RF3	RF2	RF1	RF0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **RF<7:0>**: PORTF I/O Pin bits
1 = Port pin is >VIH min.
0 = Port pin is <VIL max.

Note 1: PIC16F946 only.

REGISTER 3-15: TRISF: PORTF TRI-STATE REGISTER(1)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISF7	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2	TRISF1	TRISF0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **TRISF<7:0>:** PORTF Tri-State Control bits
1 = PORTF pin configured as an input (tri-stated)
0 = PORTF pin configured as an output

Note 1: PIC16F946 only.

3.8.1 PIN DESCRIPTIONS AND DIAGRAMS

Each PORTF pin is multiplexed with other functions. The pins and their combined functions are briefly described here. For specific information about individual functions, refer to the appropriate section in this data sheet.

3.8.1.1 RF0/SEG32

Figure 3-29 shows the diagram for this pin. The RF0 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.8.1.2 RF1/SEG33

Figure 3-29 shows the diagram for this pin. The RF1 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.8.1.3 RF2/SEG34

Figure 3-29 shows the diagram for this pin. The RF2 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.8.1.4 RF3/SEG35

Figure 3-29 shows the diagram for this pin. The RF3 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.8.1.5 RF4/SEG28

Figure 3-29 shows the diagram for this pin. The RF4 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.8.1.6 RF5/SEG29

Figure 3-29 shows the diagram for this pin. The RF5 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.8.1.7 RF6/SEG30

Figure 3-29 shows the diagram for this pin. The RF6 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.8.1.8 RF7/SEG31

Figure 3-29 shows the diagram for this pin. The RF7 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

FIGURE 3-29: BLOCK DIAGRAM OF RF<7:0>

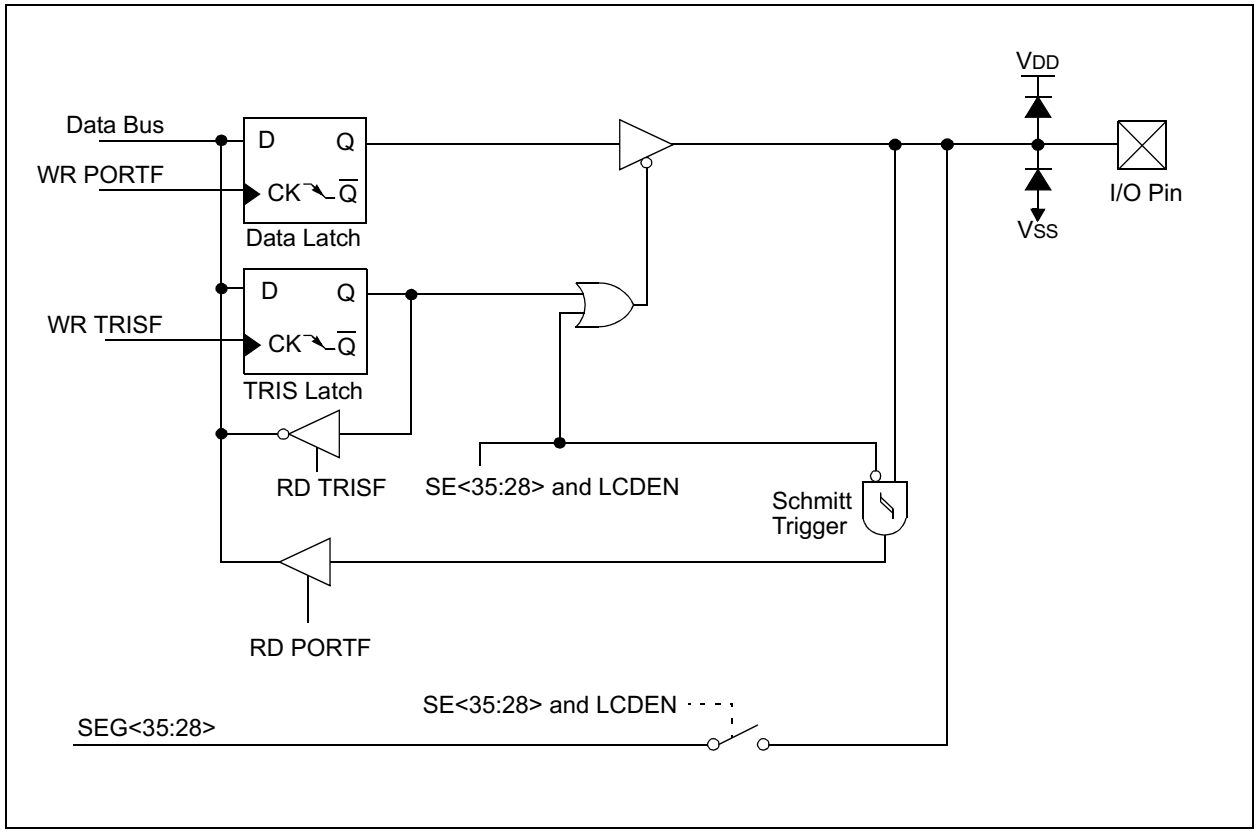


TABLE 3-6: SUMMARY OF REGISTERS ASSOCIATED WITH PORTF(1)

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE3(1)	SE31	SE30	SE29	SE28	SE27	SE26	SE25	SE24	0000 0000	uuuu uuuu
LCDSE4(1)	SE39	SE38	SE37	SE36	SE35	SE34	SE33	SE32	0000 0000	uuuu uuuu
PORTF(1)	RF7	RF6	RF5	RF4	RF3	RF2	RF1	RF0	xxxx xxxx	uuuu uuuu
TRISF(1)	TRISF7	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2	TRISF1	TRISF0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, – = unimplemented locations read as '0'. Shaded cells are not used by PORTF.

Note 1: PIC16F946 only.

3.9 PORTG and TRISG Registers

PORTG is an 8-bit port with Schmitt Trigger input buffers. RG<5:0> are individually configured as inputs or outputs, depending on the state of the port direction. The port bits are also multiplexed with LCD segment functions. PORTG is available on the PIC16F946 only.

EXAMPLE 3-7: INITIALIZING PORTG

```
BANKSEL PORTG       ;
CLRF    PORTG       ;Init PORTG
BANKSEL TRISG       ;
MOVLW   3Fh         ;Set RG<5:0> as inputs
MOVWF   TRISG       ;
```

REGISTER 3-16: PORTG: PORTG REGISTER(1)

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	RG5	RG4	RG3	RG2	RG1	RG0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **RG<5:0>**: PORTG I/O Pin bits
1 = Port pin is >VIH min.
0 = Port pin is <VIL max.

Note 1: PIC16F946 only.

REGISTER 3-17: TRISG: PORTG TRI-STATE REGISTER(1)

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	TRISG5	TRISG4	TRISG3	TRISG2	TRISG1	TRISG0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **TRISF<5:0>:** PORTG Tri-State Control bits
1 = PORTG pin configured as an input (tri-stated)
0 = PORTG pin configured as an output

Note 1: PIC16F946 only.

3.9.1 PIN DESCRIPTIONS AND DIAGRAMS

Each PORTG pin is multiplexed with other functions. The pins and their combined functions are briefly described here. For specific information about individual functions, refer to the appropriate section in this data sheet.

3.9.1.1 RG0/SEG36

Figure 3-30 shows the diagram for this pin. The RG0 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.9.1.2 RG1/SEG37

Figure 3-30 shows the diagram for this pin. The RG1 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.9.1.3 RG2/SEG38

Figure 3-30 shows the diagram for this pin. The RG2 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.9.1.4 RG3/SEG39

Figure 3-30 shows the diagram for this pin. The RG3 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.9.1.5 RG4/SEG40

Figure 3-30 shows the diagram for this pin. The RG4 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

3.9.1.6 RG5/SEG41

Figure 3-30 shows the diagram for this pin. The RG5 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog output for the LCD

FIGURE 3-30: BLOCK DIAGRAM OF RG<5:0>

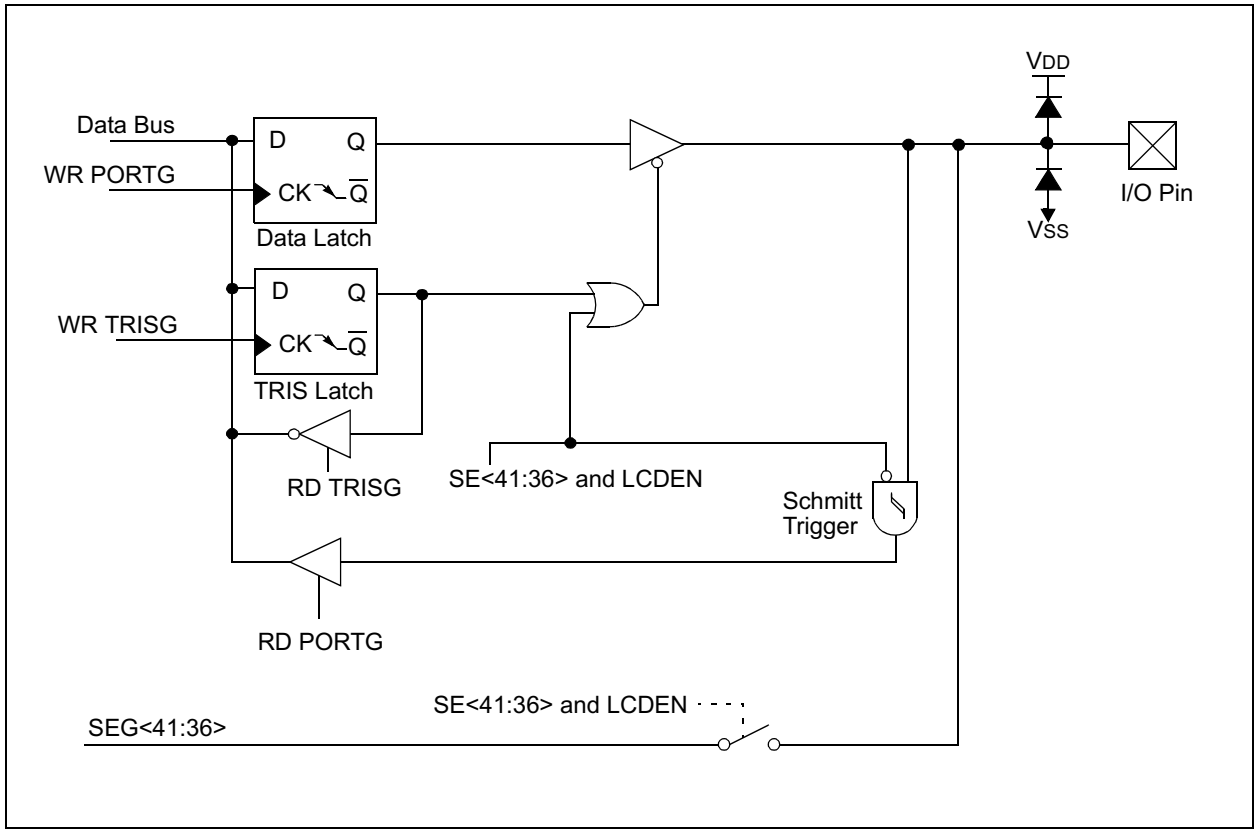


TABLE 3-7: SUMMARY OF REGISTERS ASSOCIATED WITH PORTG(1)

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE4(1)	SE39	SE38	SE37	SE36	SE35	SE34	SE33	SE32	0000 0000	uuuu uuuu
LCDSE5(1)	—	—	—	—	—	—	SE41	SE40	---- --00	---- --uu
PORTG(1)	—	—	RG5	RG4	RG3	RG2	RG1	RG0	--xx xxxx	--uu uuuu
TRISG(1)	—	—	TRISG5	TRISG4	TRISG3	TRISG2	TRISG1	TRISG0	--11 1111	--11 1111

Legend: x = unknown, u = unchanged, – = unimplemented locations read as '0'. Shaded cells are not used by PORTG.

Note 1: PIC16F946 only.

4.0 OSCILLATOR MODULE (WITH FAIL-SAFE CLOCK MONITOR)

4.1 Overview

The Oscillator module has a wide variety of clock sources and selection features that allow it to be used in a wide range of applications while maximizing performance and minimizing power consumption. Figure 4-1 illustrates a block diagram of the Oscillator module.

Clock sources can be configured from external oscillators, quartz crystal resonators, ceramic resonators and Resistor-Capacitor (RC) circuits. In addition, the system clock source can be configured from one of two internal oscillators, with a choice of speeds selectable via software. Additional clock features include:

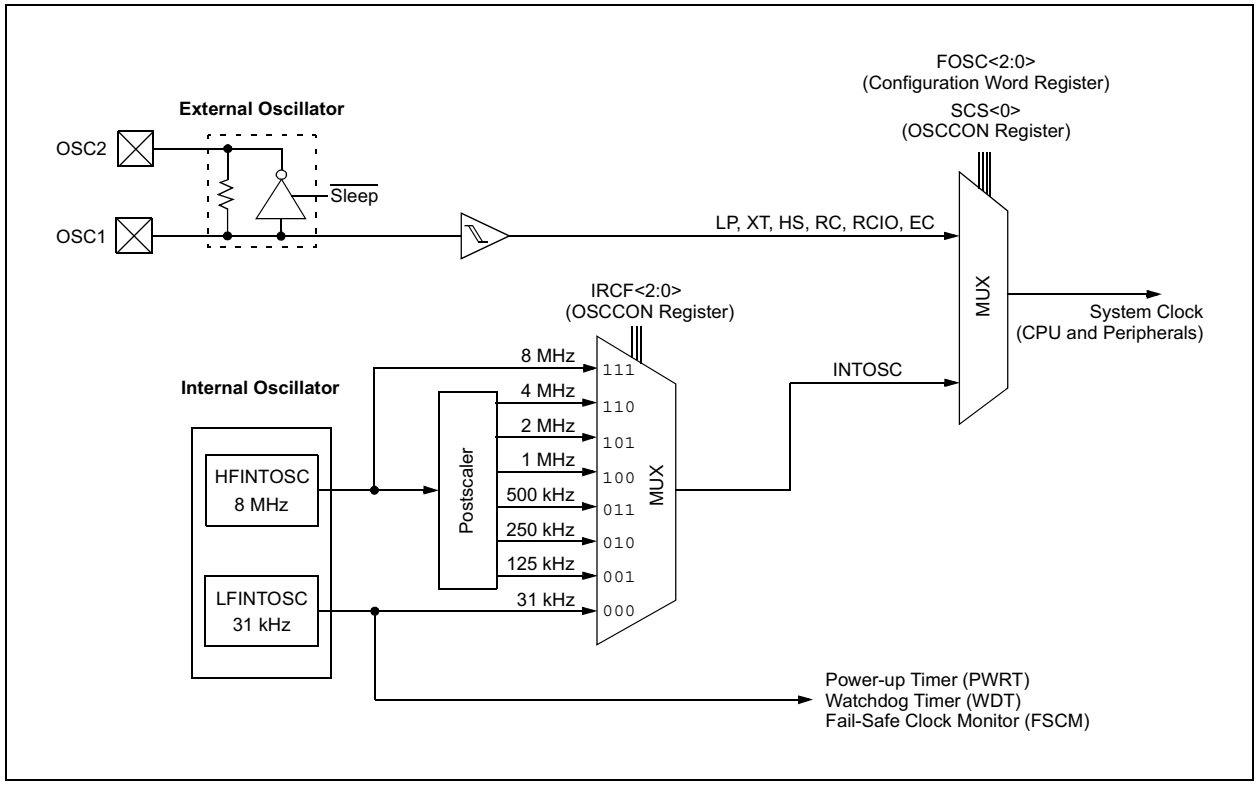
- Selectable system clock source between external or internal via software.
- Two-Speed Start-up mode, which minimizes latency between external oscillator start-up and code execution.
- Fail-Safe Clock Monitor (FSCM) designed to detect a failure of the external clock source (LP, XT, HS, EC or RC modes) and switch automatically to the internal oscillator.

The Oscillator module can be configured in one of eight clock modes.

1. EC – External clock with I/O on OSC2/CLKOUT.
2. LP – 32 kHz Low-Power Crystal mode.
3. XT – Medium Gain Crystal or Ceramic Resonator Oscillator mode.
4. HS – High Gain Crystal or Ceramic Resonator mode.
5. RC – External Resistor-Capacitor (RC) with FOSC/4 output on OSC2/CLKOUT.
6. RCIO – External Resistor-Capacitor (RC) with I/O on OSC2/CLKOUT.
7. INTOSC – Internal oscillator with FOSC/4 output on OSC2 and I/O on OSC1/CLKIN.
8. INTOSCIO – Internal oscillator with I/O on OSC1/CLKIN and OSC2/CLKOUT.

Clock Source modes are configured by the FOSC<2:0> bits in the Configuration Word register (CONFIG). The internal clock can be generated from two internal oscillators. The HFINTOSC is a calibrated high-frequency oscillator. The LFINTOSC is an uncalibrated low-frequency oscillator.

FIGURE 4-1: SIMPLIFIED PIC® MCU CLOCK SOURCE BLOCK DIAGRAM



4.2 Oscillator Control

The Oscillator Control (OSCCON) register (Figure 4-1) controls the system clock and frequency selection options. The OSCCON register contains the following bits:

- Frequency selection bits (IRCF)
- Frequency Status bits (HTS, LTS)
- System clock control bits (OSTS, SCS)

REGISTER 4-1: OSCCON: OSCILLATOR CONTROL REGISTER

U-0	R/W-1	R/W-1	R/W-0	R-1	R-0	R-0	R/W-0
—	IRCF2	IRCF1	IRCF0	OSTS(1)	HTS	LTS	SCS
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **Unimplemented:** Read as '0'

bit 6-4 **IRCF<2:0>:** Internal Oscillator Frequency Select bits
- 111 = 8 MHz
- 110 = 4 MHz (default)
- 101 = 2 MHz
- 100 = 1 MHz
- 011 = 500 kHz
- 010 = 250 kHz
- 001 = 125 kHz
- 000 = 31 kHz (LFINTOSC)

bit 3 **OSTS:** Oscillator Start-up Time-out Status bit(1)
- 1 = Device is running from the clock defined by FOSC<2:0> of the Configuration Word
- 0 = Device is running from the internal oscillator (HFINTOSC or LFINTOSC)

bit 2 **HTS:** HFINTOSC Status bit (High Frequency – 8 MHz to 125 kHz)
- 1 = HFINTOSC is stable
- 0 = HFINTOSC is not stable

bit 1 **LTS:** LFINTOSC Stable bit (Low Frequency – 31 kHz)
- 1 = LFINTOSC is stable
- 0 = LFINTOSC is not stable

bit 0 **SCS:** System Clock Select bit
- 1 = Internal oscillator is used for system clock
- 0 = Clock source defined by FOSC<2:0> of the Configuration Word

Note 1: Bit resets to '0' with Two-Speed Start-up and LP, XT or HS selected as the Oscillator mode or Fail-Safe mode is enabled.

4.3 Clock Source Modes

Clock Source modes can be classified as external or internal.

- External Clock modes rely on external circuitry for the clock source. Examples are: Oscillator modules (EC mode), quartz crystal resonators or ceramic resonators (LP, XT and HS modes) and Resistor-Capacitor (RC) mode circuits.
- Internal clock sources are contained internally within the Oscillator module. The Oscillator module has two internal oscillators: the 8 MHz High-Frequency Internal Oscillator (HFINTOSC) and the 31 kHz Low-Frequency Internal Oscillator (LFINTOSC).

The system clock can be selected between external or internal clock sources via the System Clock Select (SCS) bit of the OSCCON register. See **Section 4.6 "Clock Switching"** for additional information.

4.4 External Clock Modes

4.4.1 OSCILLATOR START-UP TIMER (OST)

If the Oscillator module is configured for LP, XT or HS modes, the Oscillator Start-up Timer (OST) counts 1024 oscillations from OSC1. This occurs following a Power-on Reset (POR) and when the Power-up Timer (PWRT) has expired (if configured), or a wake-up from Sleep. During this time, the program counter does not increment and program execution is suspended. The OST ensures that the oscillator circuit, using a quartz crystal resonator or ceramic resonator, has started and is providing a stable system clock to the Oscillator module. When switching between clock sources, a delay is required to allow the new clock to stabilize. These oscillator delays are shown in Table 4-1.

In order to minimize latency between external oscillator start-up and code execution, the Two-Speed Clock Start-up mode can be selected (see **Section 4.7 "Two-Speed Clock Start-up Mode"**).

TABLE 4-1: OSCILLATOR DELAY EXAMPLES

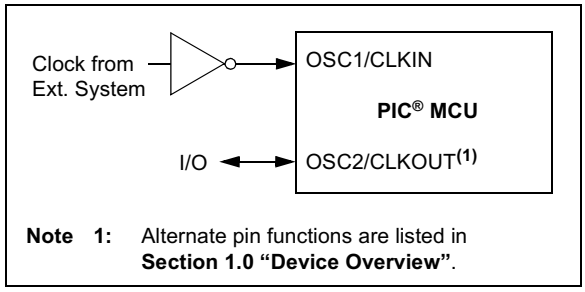
Switch From	Switch To	Frequency	Oscillator Delay
Sleep/POR	LFINTOSC HFINTOSC	31 kHz 125 kHz to 8 MHz	Oscillator Warm-Up Delay (TWARM)
Sleep/POR	EC, RC	DC – 20 MHz	2 instruction cycles
LFINTOSC (31 kHz)	EC, RC	DC – 20 MHz	1 cycle of each
Sleep/POR	LP, XT, HS	32 kHz to 20 MHz	1024 Clock Cycles (OST)
LFINTOSC (31 kHz)	HFINTOSC	125 kHz to 8 MHz	1 μs (approx.)

4.4.2 EC MODE

The External Clock (EC) mode allows an externally generated logic level as the system clock source. When operating in this mode, an external clock source is connected to the OSC1 input and the OSC2 is available for general purpose I/O. Figure 4-2 shows the pin connections for EC mode.

The Oscillator Start-up Timer (OST) is disabled when EC mode is selected. Therefore, there is no delay in operation after a Power-on Reset (POR) or wake-up from Sleep. Because the PIC® MCU design is fully static, stopping the external clock input will have the effect of halting the device while leaving all data intact. Upon restarting the external clock, the device will resume operation as if no time had elapsed.

FIGURE 4-2: EXTERNAL CLOCK (EC) MODE OPERATION



Note 1: Alternate pin functions are listed in **Section 1.0 "Device Overview"**.

4.4.3 LP, XT, HS MODES

The LP, XT and HS modes support the use of quartz crystal resonators or ceramic resonators connected to OSC1 and OSC2 (Figure 4-3). The mode selects a low, medium or high gain setting of the internal inverter-amplifier to support various resonator types and speed.

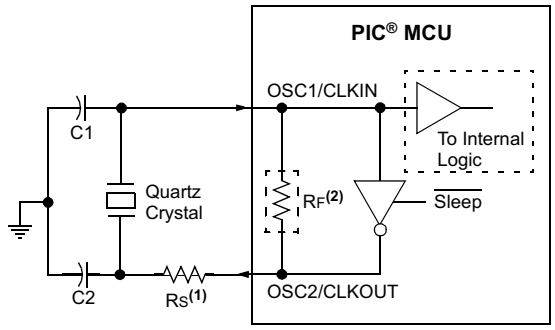
LP Oscillator mode selects the lowest gain setting of the internal inverter-amplifier. LP mode current consumption is the least of the three modes. This mode is designed to drive only 32.768 kHz tuning-fork type crystals (watch crystals).

XT Oscillator mode selects the intermediate gain setting of the internal inverter-amplifier. XT mode current consumption is the medium of the three modes. This mode is best suited to drive resonators with a medium drive level specification.

HS Oscillator mode selects the highest gain setting of the internal inverter-amplifier. HS mode current consumption is the highest of the three modes. This mode is best suited for resonators that require a high drive setting.

Figure 4-3 and Figure 4-4 show typical circuits for quartz crystal and ceramic resonators, respectively.

FIGURE 4-3: QUARTZ CRYSTAL OPERATION (LP, XT OR HS MODE)



Note 1: A series resistor (RS) may be required for quartz crystals with low drive level.

2: The value of RF varies with the Oscillator mode selected (typically between 2 MΩ to 10 MΩ).

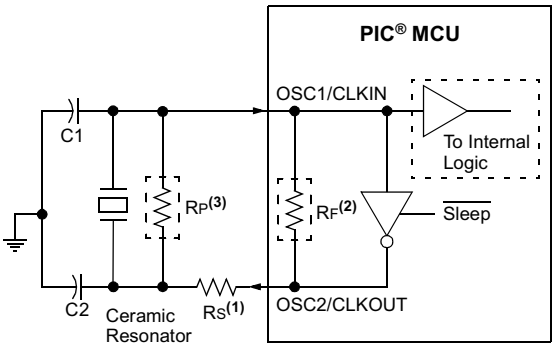
Note 1: Quartz crystal characteristics vary according to type, package and manufacturer. The user should consult the manufacturer data sheets for specifications and recommended application.

2: Always verify oscillator performance over the VDD and temperature range that is expected for the application.

3: For oscillator design assistance, reference the following Microchip Applications Notes:

- AN826, "*Crystal Oscillator Basics and Crystal Selection for rfPIC® and PIC® Devices*" (DS00826)
- AN849, "*Basic PIC® Oscillator Design*" (DS00849)
- AN943, "*Practical PIC® Oscillator Analysis and Design*" (DS00943)
- AN949, "*Making Your Oscillator Work*" (DS00949)

FIGURE 4-4: CERAMIC RESONATOR OPERATION (XT OR HS MODE)



Note 1: A series resistor (RS) may be required for ceramic resonators with low drive level.

2: The value of RF varies with the Oscillator mode selected (typically between 2 MΩ to 10 MΩ).

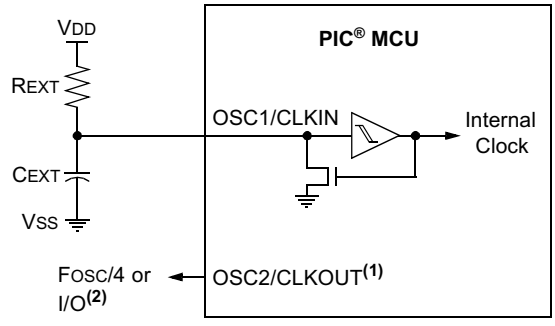
3: An additional parallel feedback resistor (RP) may be required for proper ceramic resonator operation.

4.4.4 EXTERNAL RC MODES

The external Resistor-Capacitor (RC) modes support the use of an external RC circuit. This allows the designer maximum flexibility in frequency choice while keeping costs to a minimum when clock accuracy is not required. There are two modes: RC and RCIO.

In RC mode, the RC circuit connects to OSC1. OSC2/CLKOUT outputs the RC oscillator frequency divided by 4. This signal may be used to provide a clock for external circuitry, synchronization, calibration, test or other application requirements. Figure 4-5 shows the external RC mode connections.

FIGURE 4-5: EXTERNAL RC MODES



Recommended values: 10 kΩ ≤ REXT ≤ 100 kΩ, <3V
3 kΩ ≤ REXT ≤ 100 kΩ, 3-5V
CEXT > 20 pF, 2-5V

Note 1: Alternate pin functions are listed in **Section 1.0 "Device Overview"**.
2: Output depends upon RC or RCIO clock mode.

In RCIO mode, the RC circuit is connected to OSC1. OSC2 becomes an additional general purpose I/O pin.

The RC oscillator frequency is a function of the supply voltage, the resistor (REXT) and capacitor (CEXT) values and the operating temperature. Other factors affecting the oscillator frequency are:

- threshold voltage variation
- component tolerances
- packaging variations in capacitance

The user also needs to take into account variation due to tolerance of external RC components used.

4.5 Internal Clock Modes

The Oscillator module has two independent, internal oscillators that can be configured or selected as the system clock source.

1. The **HFINTOSC** (High-Frequency Internal Oscillator) is factory calibrated and operates at 8 MHz. The frequency of the HFINTOSC can be user-adjusted via software using the OSCTUNE register (Register 4-2).
2. The **LFINTOSC** (Low-Frequency Internal Oscillator) is uncalibrated and operates at 31 kHz.

The system clock speed can be selected via software using the Internal Oscillator Frequency Select bits IRCF<2:0> of the OSCCON register.

The system clock can be selected between external or internal clock sources via the System Clock Selection (SCS) bit of the OSCCON register. See **Section 4.6 "Clock Switching"** for more information.

4.5.1 INTOSC AND INTOSCIO MODES

The INTOSC and INTOSCIO modes configure the internal oscillators as the system clock source when the device is programmed using the oscillator selection or the FOSC<2:0> bits in the Configuration Word register (CONFIG). See **Section 16.0 "Special Features of the CPU"** for more information.

In **INTOSC** mode, OSC1/CLKIN is available for general purpose I/O. OSC2/CLKOUT outputs the selected internal oscillator frequency divided by 4. The CLKOUT signal may be used to provide a clock for external circuitry, synchronization, calibration, test or other application requirements.

In **INTOSCIO** mode, OSC1/CLKIN and OSC2/CLKOUT are available for general purpose I/O.

4.5.2 HFINTOSC

The High-Frequency Internal Oscillator (HFINTOSC) is a factory calibrated 8 MHz internal clock source. The frequency of the HFINTOSC can be altered via software using the OSCTUNE register (Register 4-2).

The output of the HFINTOSC connects to a postscaler and multiplexer (see Figure 4-1). One of seven frequencies can be selected via software using the IRCF<2:0> bits of the OSCCON register. See **Section 4.5.4 "Frequency Select Bits (IRCF)"** for more information.

The HFINTOSC is enabled by selecting any frequency between 8 MHz and 125 kHz by setting the IRCF<2:0> bits of the OSCCON register ≠ 000. Then, set the System Clock Source (SCS) bit of the OSCCON register to '1' or enable Two-Speed Start-up by setting the IESO bit in the Configuration Word register (CONFIG) to '1'.

The HF Internal Oscillator (HTS) bit of the OSCCON register indicates whether the HFINTOSC is stable or not.

4.5.2.1 OSCTUNE Register

The HFINTOSC is factory calibrated but can be adjusted in software by writing to the OSCTUNE register (Register 4-2).

The default value of the OSCTUNE register is '0'. The value is a 5-bit two's complement number.

When the OSCTUNE register is modified, the HFINTOSC frequency will begin shifting to the new frequency. Code execution continues during this shift. There is no indication that the shift has occurred.

OSCTUNE does not affect the LFINTOSC frequency. Operation of features that depend on the LFINTOSC clock source frequency, such as the Power-up Timer (PWRT), Watchdog Timer (WDT), Fail-Safe Clock Monitor (FSCM) and peripherals, are *not* affected by the change in frequency.

REGISTER 4-2: OSCTUNE: OSCILLATOR TUNING REGISTER

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	TUN4	TUN3	TUN2	TUN1	TUN0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-5 **Unimplemented:** Read as '0'

bit 4-0 **TUN<4:0>:** Frequency Tuning bits

01111 = Maximum frequency
01110 =
•
•
•
00001 =
00000 = Oscillator module is running at the factory-calibrated frequency.
11111 =
•
•
•
10000 = Minimum frequency

4.5.3 LFINTOSC

The Low-Frequency Internal Oscillator (LFINTOSC) is an uncalibrated 31 kHz internal clock source.

The output of the LFINTOSC connects to a postscaler and multiplexer (see Figure 4-1). Select 31 kHz, via software, using the IRCF<2:0> bits of the OSCCON register. See **Section 4.5.4 "Frequency Select Bits (IRCF)"** for more information. The LFINTOSC is also the frequency for the Power-up Timer (PWRT), Watchdog Timer (WDT) and Fail-Safe Clock Monitor (FSCM).

The LFINTOSC is enabled by selecting 31 kHz (IRCF<2:0> bits of the OSCCON register = `000`) as the system clock source (SCS bit of the OSCCON register = `1`), or when any of the following are enabled:

- Two-Speed Start-up IESO bit of the Configuration Word register = `1` and IRCF<2:0> bits of the OSCCON register = `000`
- Power-up Timer (PWRT)
- Watchdog Timer (WDT)
- Fail-Safe Clock Monitor (FSCM)

The LF Internal Oscillator (LTS) bit of the OSCCON register indicates whether the LFINTOSC is stable or not.

4.5.4 FREQUENCY SELECT BITS (IRCF)

The output of the 8 MHz HFINTOSC and 31 kHz LFINTOSC connects to a postscaler and multiplexer (see Figure 4-1). The Internal Oscillator Frequency Select bits IRCF<2:0> of the OSCCON register select the frequency output of the internal oscillators. One of eight frequencies can be selected via software:

- 8 MHz
- 4 MHz (Default after Reset)
- 2 MHz
- 1 MHz
- 500 kHz
- 250 kHz
- 125 kHz
- 31 kHz (LFINTOSC)

Note:	Following any Reset, the IRCF<2:0> bits of the OSCCON register are set to '`110`' and the frequency selection is set to 4 MHz. The user can modify the IRCF bits to select a different frequency.

4.5.5 HF AND LF INTOSC CLOCK SWITCH TIMING

When switching between the LFINTOSC and the HFINTOSC, the new oscillator may already be shut down to save power (see Figure 4-6). If this is the case, there is a delay after the IRCF<2:0> bits of the OSCCON register are modified before the frequency selection takes place. The LTS and HTS bits of the OSCCON register will reflect the current active status of the LFINTOSC and HFINTOSC oscillators. The timing of a frequency selection is as follows:

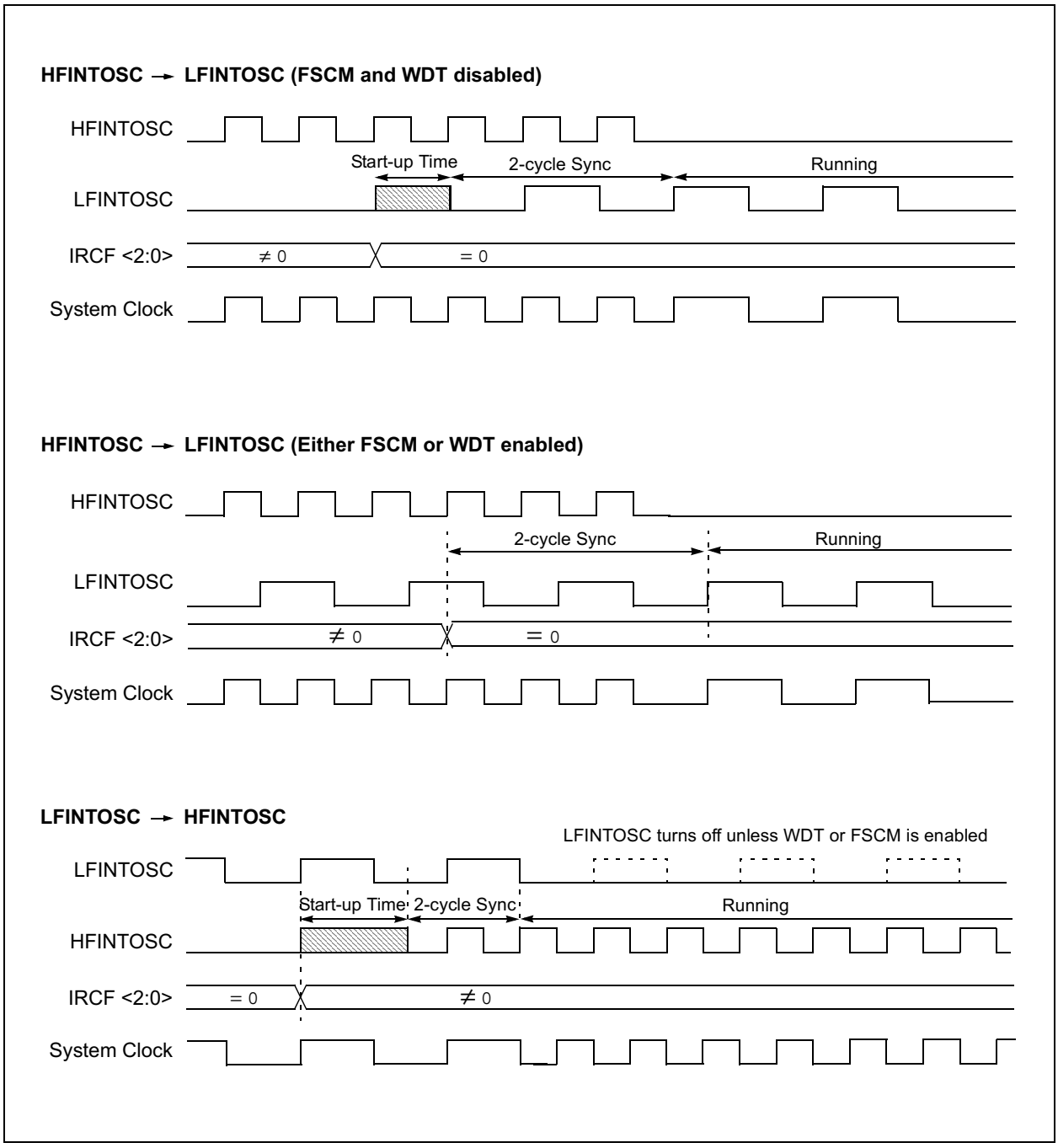
1. IRCF<2:0> bits of the OSCCON register are modified.
2. If the new clock is shut down, a clock start-up delay is started.
3. Clock switch circuitry waits for a falling edge of the current clock.
4. CLKOUT is held low and the clock switch circuitry waits for a rising edge in the new clock.
5. CLKOUT is now connected with the new clock. LTS and HTS bits of the OSCCON register are updated as required.
6. Clock switch is complete.

See Figure 4-1 for more details.

If the internal oscillator speed selected is between 8 MHz and 125 kHz, there is no start-up delay before the new frequency is selected. This is because the old and new frequencies are derived from the HFINTOSC via the postscaler and multiplexer.

Start-up delay specifications are located under the oscillator parameters of **Section 19.0 "Electrical Specifications"**.

FIGURE 4-6: INTERNAL OSCILLATOR SWITCH TIMING



4.6 Clock Switching

The system clock source can be switched between external and internal clock sources via software using the System Clock Select (SCS) bit of the OSCCON register.

4.6.1 SYSTEM CLOCK SELECT (SCS) BIT

The System Clock Select (SCS) bit of the OSCCON register selects the system clock source that is used for the CPU and peripherals.

- When the SCS bit of the OSCCON register = 0, the system clock source is determined by configuration of the FOSC<2:0> bits in the Configuration Word register (CONFIG).
- When the SCS bit of the OSCCON register = 1, the system clock source is chosen by the internal oscillator frequency selected by the IRCF<2:0> bits of the OSCCON register. After a Reset, the SCS bit of the OSCCON register is always cleared.

Note:	Any automatic clock switch, which may occur from Two-Speed Start-up or Fail-Safe Clock Monitor, does not update the SCS bit of the OSCCON register. The user can monitor the OSTS bit of the OSCCON register to determine the current system clock source.

4.6.2 OSCILLATOR START-UP TIME-OUT STATUS (OSTS) BIT

The Oscillator Start-up Time-out Status (OSTS) bit of the OSCCON register indicates whether the system clock is running from the external clock source, as defined by the FOSC<2:0> bits in the Configuration Word register (CONFIG), or from the internal clock source. In particular, OSTS indicates that the Oscillator Start-up Timer (OST) has timed out for LP, XT or HS modes.

4.7 Two-Speed Clock Start-up Mode

Two-Speed Start-up mode provides additional power savings by minimizing the latency between external oscillator start-up and code execution. In applications that make heavy use of the Sleep mode, Two-Speed Start-up will remove the external oscillator start-up time from the time spent awake and can reduce the overall power consumption of the device.

This mode allows the application to wake-up from Sleep, perform a few instructions using the INTOSC as the clock source and go back to Sleep without waiting for the primary oscillator to become stable.

Note:	Executing a SLEEP instruction will abort the oscillator start-up time and will cause the OSTS bit of the OSCCON register to remain clear.

When the Oscillator module is configured for LP, XT or HS modes, the Oscillator Start-up Timer (OST) is enabled (see **Section 4.4.1 "Oscillator Start-up Timer (OST)"**). The OST will suspend program execution until 1024 oscillations are counted. Two-Speed Start-up mode minimizes the delay in code execution by operating from the internal oscillator as the OST is counting. When the OST count reaches 1024 and the OSTS bit of the OSCCON register is set, program execution switches to the external oscillator.

4.7.1 TWO-SPEED START-UP MODE CONFIGURATION

Two-Speed Start-up mode is configured by the following settings:

- IESO (of the Configuration Word register) = 1; Internal/External Switchover bit (Two-Speed Start-up mode enabled).
- SCS (of the OSCCON register) = 0.
- FOSC<2:0> bits in the Configuration Word register (CONFIG) configured for LP, XT or HS mode.

Two-Speed Start-up mode is entered after:

- Power-on Reset (POR) and, if enabled, after Power-up Timer (PWRT) has expired, or
- Wake-up from Sleep.

If the external clock oscillator is configured to be anything other than LP, XT or HS mode, then Two-Speed Start-up is disabled. This is because the external clock oscillator does not require any stabilization time after POR or an exit from Sleep.

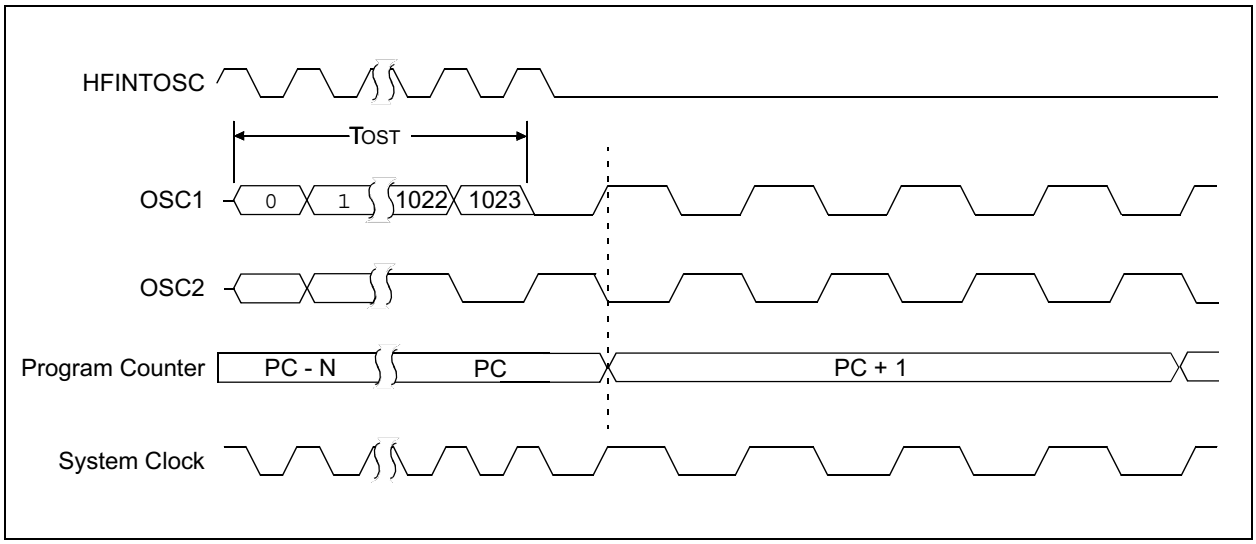
4.7.2 TWO-SPEED START-UP SEQUENCE

1. Wake-up from Power-on Reset or Sleep.
2. Instructions begin execution by the internal oscillator at the frequency set in the IRCF<2:0> bits of the OSCCON register.
3. OST enabled to count 1024 clock cycles.
4. OST timed out, wait for falling edge of the internal oscillator.
5. OSTS is set.
6. System clock held low until the next falling edge of new clock (LP, XT or HS mode).
7. System clock is switched to external clock source.

4.7.3 CHECKING TWO-SPEED CLOCK STATUS

Checking the state of the OSTS bit of the OSCCON register will confirm if the microcontroller is running from the external clock source, as defined by the FOSC<2:0> bits in the Configuration Word register (CONFIG), or the internal oscillator.

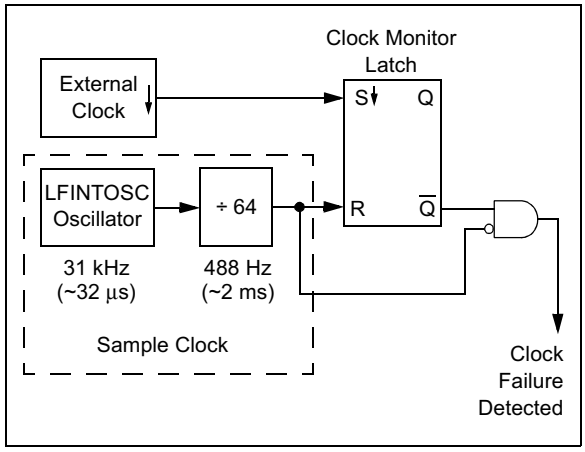
FIGURE 4-7: TWO-SPEED START-UP



4.8 Fail-Safe Clock Monitor

The Fail-Safe Clock Monitor (FSCM) allows the device to continue operating should the external oscillator fail. The FSCM can detect oscillator failure any time after the Oscillator Start-up Timer (OST) has expired. The FSCM is enabled by setting the FCMEN bit in the Configuration Word register (CONFIG). The FSCM is applicable to all external oscillator modes (LP, XT, HS, EC, RC and RCIO).

FIGURE 4-8: FSCM BLOCK DIAGRAM



4.8.1 FAIL-SAFE DETECTION

The FSCM module detects a failed oscillator by comparing the external oscillator to the FSCM sample clock. The sample clock is generated by dividing the LFINTOSC by 64. See Figure 4-8. Inside the fail detector block is a latch. The external clock sets the latch on each falling edge of the external clock. The sample clock clears the latch on each rising edge of the sample clock. A failure is detected when an entire half-cycle of the sample clock elapses before the primary clock goes low.

4.8.2 FAIL-SAFE OPERATION

When the external clock fails, the FSCM switches the device clock to an internal clock source and sets the bit flag OSFIF of the PIR2 register. Setting this flag will generate an interrupt if the OSFIE bit of the PIE2 register is also set. The device firmware can then take steps to mitigate the problems that may arise from a failed clock. The system clock will continue to be sourced from the internal clock source until the device firmware successfully restarts the external oscillator and switches back to external operation.

The internal clock source chosen by the FSCM is determined by the IRCF<2:0> bits of the OSCCON register. This allows the internal oscillator to be configured before a failure occurs.

4.8.3 FAIL-SAFE CONDITION CLEARING

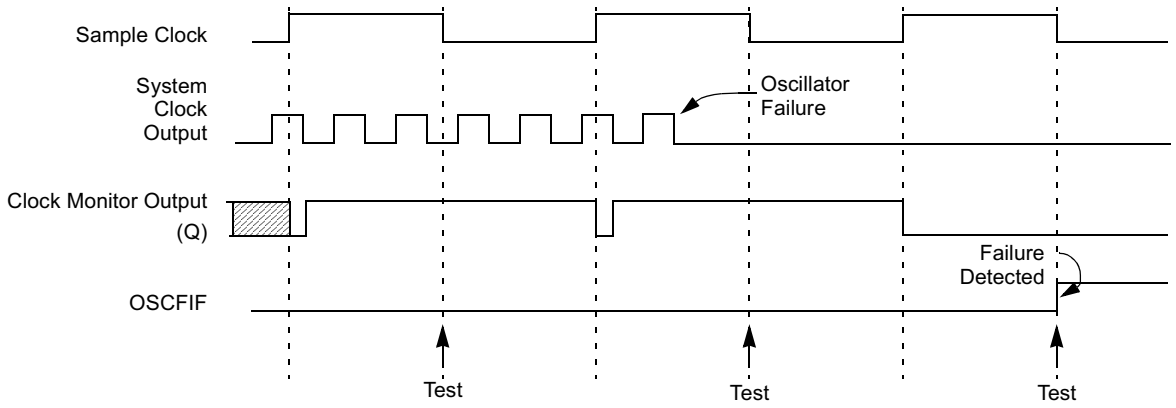
The Fail-Safe condition is cleared after a Reset, executing a SLEEP instruction or toggling the SCS bit of the OSCCON register. When the SCS bit is toggled, the OST is restarted. While the OST is running, the device continues to operate from the INTOSC selected in OSCCON. When the OST times out, the Fail-Safe condition is cleared and the device will be operating from the external clock source. The Fail-Safe condition must be cleared before the OSFIF flag can be cleared.

4.8.4 RESET OR WAKE-UP FROM SLEEP

The FSCM is designed to detect an oscillator failure after the Oscillator Start-up Timer (OST) has expired. The OST is used after waking up from Sleep and after any type of Reset. The OST is not used with the EC or RC Clock modes so that the FSCM will be active as soon as the Reset or wake-up has completed. When the FSCM is enabled, the Two-Speed Start-up is also enabled. Therefore, the device will always be executing code while the OST is operating.

Note:	Due to the wide range of oscillator start-up times, the Fail-Safe circuit is not active during oscillator start-up (i.e., after exiting Reset or Sleep). After an appropriate amount of time, the user should check the OSTS bit of the OSCCON register to verify the oscillator start-up and that the system clock switchover has successfully completed.

FIGURE 4-9: FSCM TIMING DIAGRAM



Note: The system clock is normally at a much higher frequency than the sample clock. The relative frequencies in this example have been chosen for clarity.

TABLE 4-2: SUMMARY OF REGISTERS ASSOCIATED WITH CLOCK SOURCES

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets(1)
CONFIG(2)	$\overline{CPD}$	$\overline{CP}$	MCLRE	$\overline{PWRTE}$	WDTE	FOSC2	FOSC1	FOSC0	—	—
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
OSCCON	—	IRCF2	IRCF1	IRCF0	OSTS	HTS	LTS	SCS	-110 x000	-110 x000
OSCTUNE	—	—	—	TUN4	TUN3	TUN2	TUN1	TUN0	---0 0000	---u uuuu
PIE2	OSFIE	C2IE	C1IE	LCDIE	—	LVDIE	—	CCP2IE	0000 -0-0	0000 -0-0
PIR2	OSFIF	C2IF	C1IF	LCDIF	—	LVDIF	—	CCP2IF	0000 -0-0	0000 -0-0
T1CON	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYNC}$	TMR1CS	TMR1ON	0000 0000	0000 0000

Legend: x = unknown, u = unchanged, – = unimplemented locations read as '0'. Shaded cells are not used by oscillators.
Note 1: Other (non Power-up) Resets include $\overline{MCLR}$ Reset and Watchdog Timer Reset during normal operation.
2: See Configuration Word register (CONFIG) for operation of all register bits.

5.0 TIMER0 MODULE

The Timer0 module is an 8-bit timer/counter with the following features:

- 8-bit timer/counter register (TMR0)
- 8-bit prescaler (shared with Watchdog Timer)
- Programmable internal or external clock source
- Programmable external clock edge selection
- Interrupt on overflow

Figure 5-1 is a block diagram of the Timer0 module.

5.1 Timer0 Operation

When used as a timer, the Timer0 module can be used as either an 8-bit timer or an 8-bit counter.

5.1.1 8-BIT TIMER MODE

When used as a timer, the Timer0 module will increment every instruction cycle (without prescaler). Timer mode is selected by clearing the T0CS bit of the OPTION register to '0'.

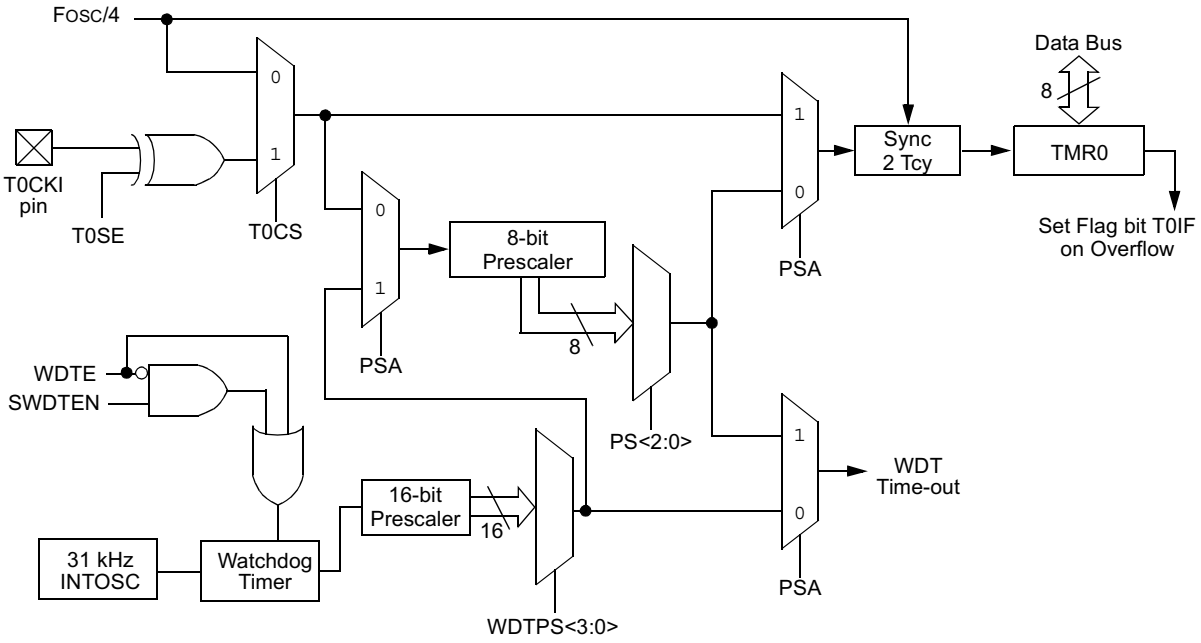
When TMR0 is written, the increment is inhibited for two instruction cycles immediately following the write.

Note:	The value written to the TMR0 register can be adjusted, in order to account for the two instruction cycle delay when TMR0 is written.

5.1.2 8-BIT COUNTER MODE

When used as a counter, the Timer0 module will increment on every rising or falling edge of the T0CKI pin. The incrementing edge is determined by the T0SE bit of the Option register. Counter mode is selected by setting the T0CS bit of the Option register to '1'.

FIGURE 5-1: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER



Note 1: T0SE, T0CS, PSA, PS<2:0> are bits in the Option register.
2: SWDTEN and WDTPS<3:0> are bits in the WDTCON register.
3: WDTE bit is in the Configuration Word register.

5.1.3 SOFTWARE PROGRAMMABLE PRESCALER

A single software programmable prescaler is available for use with either Timer0 or the Watchdog Timer (WDT), but not both simultaneously. The prescaler assignment is controlled by the PSA bit of the Option register. To assign the prescaler to Timer0, the PSA bit must be cleared to a '0'.

There are 8 prescaler options for the Timer0 module ranging from 1:2 to 1:256. The prescale values are selectable via the PS<2:0> bits of the OPTION register. In order to have a 1:1 prescaler value for the Timer0 module, the prescaler must be assigned to the WDT module.

The prescaler is not readable or writable. When assigned to the Timer0 module, all instructions writing to the TMR0 register will clear the prescaler.

When the prescaler is assigned to WDT, a CLRWDT instruction will clear the prescaler along with the WDT.

5.1.3.1 Switching Prescaler Between Timer0 and WDT Modules

As a result of having the prescaler assigned to either Timer0 or the WDT, it is possible to generate an unintended device Reset when switching prescaler values. When changing the prescaler assignment from Timer0 to the WDT module, the instruction sequence shown in Example 5-1, must be executed.

EXAMPLE 5-1: CHANGING PRESCALER (TIMER0 → WDT)

```
BANKSEL  TMR0            ;
CLRWDT                   ;Clear WDT
CLRF     TMR0            ;Clear TMR0 and
                         ;prescaler
BANKSEL  OPTION_REG      ;
BSF      OPTION_REG,PSA  ;Select WDT
CLRWDT                   ;
                         ;
MOVLW    b'11111000'     ;Mask prescaler
ANDWF    OPTION_REG,W    ;bits
IORLW    b'00000101'     ;Set WDT prescaler
MOVWF    OPTION_REG      ;to 1:32
```

When changing the prescaler assignment from the WDT to the Timer0 module, the following instruction sequence must be executed (see Example 5-2).

EXAMPLE 5-2: CHANGING PRESCALER (WDT → TIMER0)

```
CLRWDT                ;Clear WDT and
                      ;prescaler
BANKSEL OPTION_REG    ;
MOVLW   b'11110000'   ;Mask TMR0 select and
ANDWF   OPTION_REG,W  ;prescaler bits
IORLW   b'00000011'   ;Set prescale to 1:16
MOVWF   OPTION_REG    ;
```

5.1.4 TIMER0 INTERRUPT

Timer0 will generate an interrupt when the TMR0 register overflows from FFh to 00h. The T0IF interrupt flag bit of the INTCON register is set every time the TMR0 register overflows, regardless of whether or not the Timer0 interrupt is enabled. The T0IF bit must be cleared in software. The Timer0 interrupt enable is the T0IE bit of the INTCON register.

Note:	The Timer0 interrupt cannot wake the processor from Sleep since the timer is frozen during Sleep.

5.1.5 USING TIMER0 WITH AN EXTERNAL CLOCK

When Timer0 is in Counter mode, the synchronization of the T0CKI input and the Timer0 register is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks. Therefore, the high and low periods of the external clock source must meet the timing requirements as shown in **Section 19.0 "Electrical Specifications"**

REGISTER 5-1: OPTION_REG: OPTION REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **$\overline{\text{RBPU}}$:** PORTB Pull-up Enable bit
1 = PORTB pull-ups are disabled
0 = PORTB pull-ups are enabled by individual PORT latch values

bit 6 **INTEDG:** Interrupt Edge Select bit
1 = Interrupt on rising edge of INT pin
0 = Interrupt on falling edge of INT pin

bit 5 **T0CS:** TMR0 Clock Source Select bit
1 = Transition on T0CKI pin
0 = Internal instruction cycle clock (FOSC/4)

bit 4 **T0SE:** TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on T0CKI pin
0 = Increment on low-to-high transition on T0CKI pin

bit 3 **PSA:** Prescaler Assignment bit
1 = Prescaler is assigned to the WDT
0 = Prescaler is assigned to the Timer0 module

bit 2-0 **PS<2:0>:** Prescaler Rate Select bits

BIT VALUE	TMR0 RATE	WDT RATE
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

Note 1: A dedicated 16-bit WDT postscaler is available. See **Section 16.4 "Watchdog Timer (WDT)"** for more information.

TABLE 5-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER0

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
TMR0	Timer0 Module Register								xxxx xxxx	uuuu uuuu
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111

Legend: - = Unimplemented locations, read as '0', u = unchanged, x = unknown. Shaded cells are not used by the Timer0 module.

6.0 TIMER1 MODULE WITH GATE CONTROL

The Timer1 module is a 16-bit timer/counter with the following features:

- 16-bit timer/counter register pair (TMR1H:TMR1L)
- Programmable internal or external clock source
- 3-bit prescaler
- Optional LP oscillator
- Synchronous or asynchronous operation
- Timer1 gate (count enable) via comparator or $\overline{T1G}$ pin
- Interrupt on overflow
- Wake-up on overflow (external clock, Asynchronous mode only)
- Clock source for LCD module

Figure 6-1 is a block diagram of the Timer1 module.

6.1 Timer1 Operation

The Timer1 module is a 16-bit incrementing counter which is accessed through the TMR1H:TMR1L register pair. Writes to TMR1H or TMR1L directly update the counter.

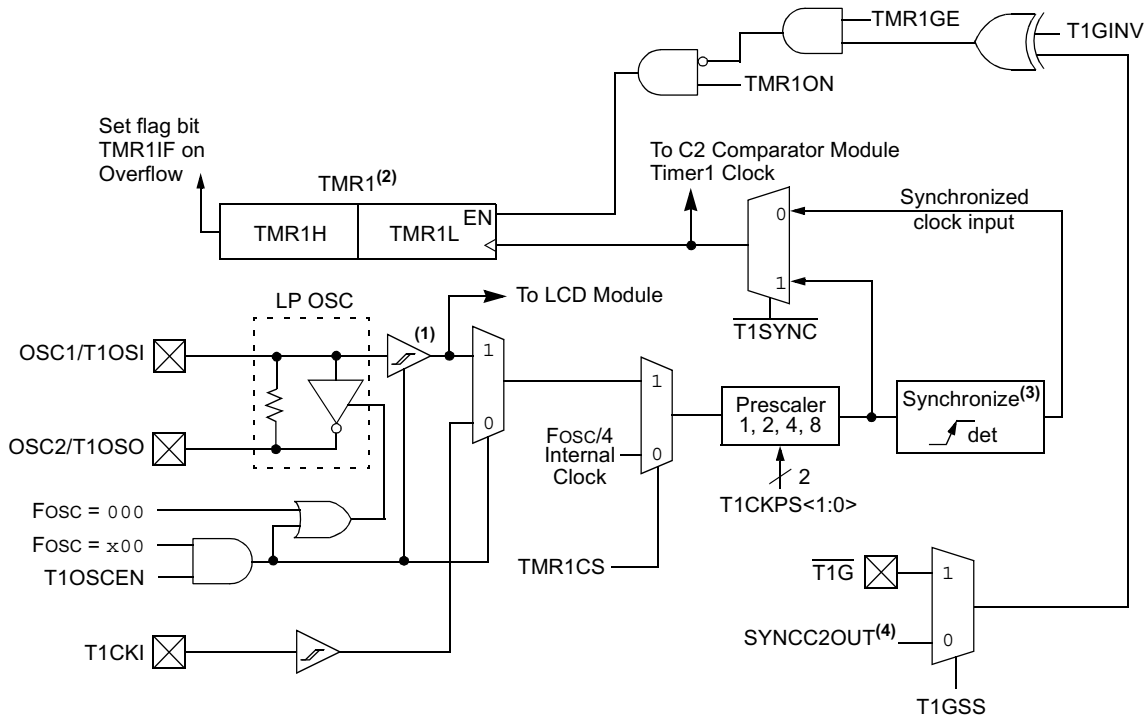
When used with an internal clock source, the module is a timer. When used with an external clock source, the module can be used as either a timer or counter.

6.2 Clock Source Selection

The TMR1CS bit of the T1CON register is used to select the clock source. When TMR1CS = 0, the clock source is FOSC/4. When TMR1CS = 1, the clock source is supplied externally.

Clock Source	TMR1CS
FOSC/4	0
T1CKI pin	1

FIGURE 6-1: TIMER1 BLOCK DIAGRAM



Note 1: ST Buffer is low power type when using LP oscillator, or high speed type when using T1CKI.
2: Timer1 register increments on rising edge.
3: Synchronize does not operate while in Sleep.
4: SYNCC2OUT is synchronized when the C2SYNC bit of the CMCON1 register is set.

6.2.1 INTERNAL CLOCK SOURCE

When the internal clock source is selected, the TMR1H:TMR1L register pair will increment on multiples of TCY as determined by the Timer1 prescaler.

6.2.2 EXTERNAL CLOCK SOURCE

When the external clock source is selected, the Timer1 module may work as a timer or a counter.

When counting, Timer1 is incremented on the rising edge of the external clock input T1CKI. In addition, the Counter mode clock can be synchronized to the microcontroller system clock or run asynchronously.

In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge after one or more of the following conditions:

- Timer1 is enabled after POR or BOR Reset
- A write to TMR1H or TMR1L
- T1CKI is high when Timer1 is disabled and when Timer1 is reenabled T1CKI is low. See Figure 6-2.

6.3 Timer1 Prescaler

Timer1 has four prescaler options allowing 1, 2, 4 or 8 divisions of the clock input. The T1CKPS bits of the T1CON register control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMR1H or TMR1L.

6.4 Timer1 Oscillator

A low-power 32.768 kHz crystal oscillator is built-in between pins OSC1 (input) and OSC2 (amplifier output). The oscillator is enabled by setting the T1OSCEN control bit of the T1CON register. The oscillator will continue to run during Sleep.

The Timer1 oscillator is shared with the system LP oscillator. Thus, Timer1 can use this mode only when the primary system clock is derived from the internal oscillator or when in LP oscillator mode. The user must provide a software time delay to ensure proper oscillator start-up.

TRISA7 and TRISA6 bits are set when the Timer1 oscillator is enabled. RA7 and RA6 bits read as '0' and TRISA7 and TRISA6 bits read as '1'.

Note:	The oscillator requires a start-up and stabilization time before use. Thus, T1OSCEN should be set and a suitable delay observed prior to enabling Timer1.

6.5 Timer1 Operation in Asynchronous Counter Mode

If control bit $\overline{\text{T1SYNC}}$ of the T1CON register is set, the external clock input is not synchronized. The timer continues to increment asynchronous to the internal phase clocks. The timer will continue to run during Sleep and can generate an interrupt on overflow, which will wake-up the processor. However, special precautions in software are needed to read/write the timer (see **Section 6.5.1 "Reading and Writing Timer1 in Asynchronous Counter Mode"**).

Note 1:	When switching from synchronous to asynchronous operation, it is possible to skip an increment. When switching from asynchronous to synchronous operation, it is possible to produce an additional increment.

6.5.1 READING AND WRITING TIMER1 IN ASYNCHRONOUS COUNTER MODE

Reading TMR1H or TMR1L while the timer is running from an external asynchronous clock will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself poses certain problems, since the timer may overflow between the reads.

For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers, while the register is incrementing. This may produce an unpredictable value in the TMR1H:TMR1L register pair.

6.6 Timer1 Gate

Timer1 gate source is software configurable to be the $\overline{\text{T1G}}$ pin or the output of Comparator C2. This allows the device to directly time external events using $\overline{\text{T1G}}$ or analog events using Comparator C2. See the CMCON1 register (Register 8-2) for selecting the Timer1 gate source. This feature can simplify the software for a Delta-Sigma A/D converter and many other applications. For more information on Delta-Sigma A/D converters, see the Microchip web site (www.microchip.com).

Note:	TMR1GE bit of the T1CON register must be set to use the Timer1 gate.

Timer1 gate can be inverted using the T1GINV bit of the T1CON register, whether it originates from the $\overline{\text{T1G}}$ pin or Comparator C2 output. This configures Timer1 to measure either the active-high or active-low time between events.

6.7 Timer1 Interrupt

The Timer1 register pair (TMR1H:TMR1L) increments to FFFFh and rolls over to 0000h. When Timer1 rolls over, the Timer1 interrupt flag bit of the PIR1 register is set. To enable the interrupt on rollover, you must set these bits:

- Timer1 interrupt enable bit of the PIE1 register
- PEIE bit of the INTCON register
- GIE bit of the INTCON register

The interrupt is cleared by clearing the TMR1IF bit in the Interrupt Service Routine.

Note:	The TMR1H:TMR1L register pair and the TMR1IF bit should be cleared before enabling interrupts.

6.8 Timer1 Operation During Sleep

Timer1 can only operate during Sleep when setup in Asynchronous Counter mode. In this mode, an external crystal or clock source can be used to increment the counter. To set up the timer to wake the device:

- TMR1ON bit of the T1CON register must be set
- TMR1IE bit of the PIE1 register must be set
- PEIE bit of the INTCON register must be set

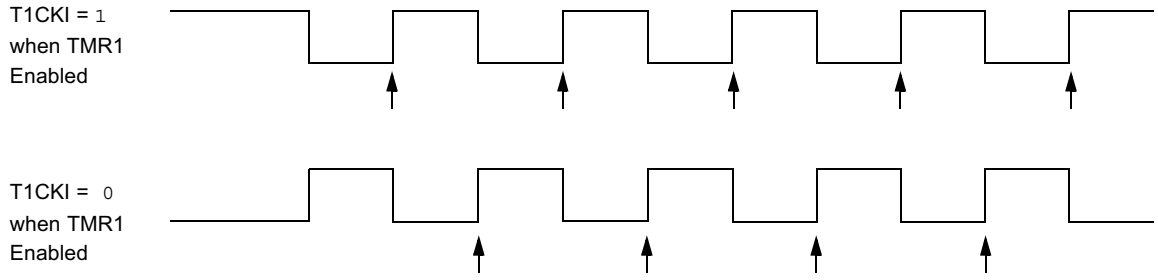
The device will wake-up on an overflow and execute the next instruction. If the GIE bit of the INTCON register is set, the device will call the Interrupt Service Routine (0004h).

6.9 Clock Source for LCD Module

The Timer1 oscillator can be used to provide a clock for the LCD module. This clock may be configured to remain running during Sleep.

For more information, see **Section 10.0 "Liquid Crystal Display (LCD) Driver Module"**.

FIGURE 6-2: TIMER1 INCREMENTING EDGE



Note 1: Arrows indicate counter increments.

2: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge of the clock.

6.10 Timer1 Control Register

The Timer1 Control register (T1CON), shown in Register 6-1, is used to control Timer1 and select the various features of the Timer1 module.

REGISTER 6-1: T1CON: TIMER 1 CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T1GINV(1)	TMR1GE(2)	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **T1GINV:** Timer1 Gate Invert bit(1)
1 = Timer1 gate is active-high (Timer1 counts when gate is high)
0 = Timer1 gate is active-low (Timer1 counts when gate is low)

bit 6 **TMR1GE:** Timer1 Gate Enable bit(2)
If TMR1ON = 0:
This bit is ignored
If TMR1ON = 1:
1 = Timer1 counting is controlled by the Timer1 Gate function
0 = Timer1 is always counting

bit 5-4 **T1CKPS<1:0>:** Timer1 Input Clock Prescale Select bits
11 = 1:8 Prescale Value
10 = 1:4 Prescale Value
01 = 1:2 Prescale Value
00 = 1:1 Prescale Value

bit 3 **T1OSCEN:** LP Oscillator Enable Control bit
If INTOSC without CLKOUT oscillator is active:
1 = LP oscillator is enabled for Timer1 clock
0 = LP oscillator is off
Else:
This bit is ignored. LP oscillator is disabled.

bit 2 **$\overline{\text{T1SYNC}}$:** Timer1 External Clock Input Synchronization Control bit
TMR1CS = 1:
1 = Do not synchronize external clock input
0 = Synchronize external clock input
TMR1CS = 0:
This bit is ignored. Timer1 uses the internal clock.

bit 1 **TMR1CS:** Timer1 Clock Source Select bit
1 = External clock from T1CKI pin (on the rising edge)
0 = Internal clock (FOSC/4)

bit 0 **TMR1ON:** Timer1 On bit
1 = Enables Timer1
0 = Stops Timer1

Note 1: T1GINV bit inverts the Timer1 gate logic, regardless of source.
2: TMR1GE bit must be set to use either $\overline{\text{T1G}}$ pin or C2OUT, as selected by the T1GSS bit of the CMCON1 register, as a Timer1 gate source.

TABLE 6-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER1

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
CMCON1	—	—	—	—	—	—	T1GSS	C2SYNC	---- --10	---- --10
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
T1CON	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	0000 0000	uuuu uuuu

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by the Timer1 module.

7.0 TIMER2 MODULE

The Timer2 module is an 8-bit timer with the following features:

- 8-bit timer register (TMR2)
- 8-bit period register (PR2)
- Interrupt on TMR2 match with PR2
- Software programmable prescaler (1:1, 1:4, 1:16)
- Software programmable postscaler (1:1 to 1:16)

See Figure 7-1 for a block diagram of Timer2.

7.1 Timer2 Operation

The clock input to the Timer2 module is the system instruction clock (FOSC/4). The clock is fed into the Timer2 prescaler, which has prescale options of 1:1, 1:4 or 1:16. The output of the prescaler is then used to increment the TMR2 register.

The values of TMR2 and PR2 are constantly compared to determine when they match. TMR2 will increment from 00h until it matches the value in PR2. When a match occurs, two things happen:

- TMR2 is reset to 00h on the next increment cycle.
- The Timer2 postscaler is incremented.

The match output of the Timer2/PR2 comparator is then fed into the Timer2 postscaler. The postscaler has postscale options of 1:1 to 1:16 inclusive. The output of the Timer2 postscaler is used to set the TMR2IF interrupt flag bit in the PIR1 register.

The TMR2 and PR2 registers are both fully readable and writable. On any Reset, the TMR2 register is set to 00h and the PR2 register is set to FFh.

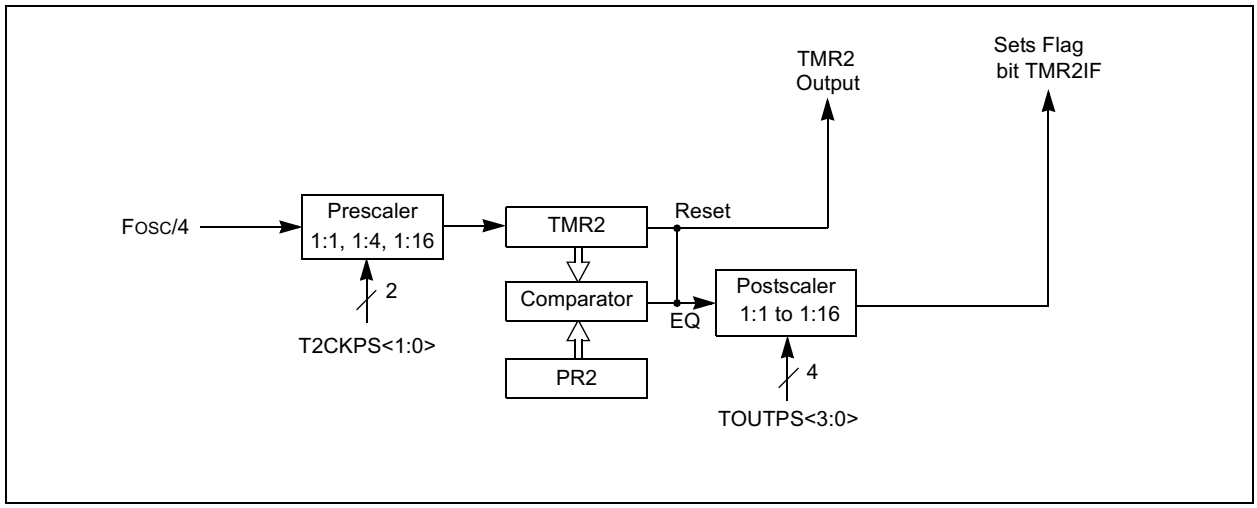
Timer2 is turned on by setting the TMR2ON bit in the T2CON register to a '1'. Timer2 is turned off by clearing the TMR2ON bit to a '0'.

The Timer2 prescaler is controlled by the T2CKPS bits in the T2CON register. The Timer2 postscaler is controlled by the TOUTPS bits in the T2CON register. The prescaler and postscaler counters are cleared when:

- A write to TMR2 occurs.
- A write to T2CON occurs.
- Any device Reset occurs (Power-on Reset, $\overline{\text{MCLR}}$ Reset, Watchdog Timer Reset, or Brown-out Reset).

Note: TMR2 is not cleared when T2CON is written.

FIGURE 7-1: TIMER2 BLOCK DIAGRAM



REGISTER 7-1: T2CON: TIMER 2 CONTROL REGISTER

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **Unimplemented:** Read as '0'

bit 6-3 **TOUTPS<3:0>:** Timer2 Output Postscaler Select bits

0000 = 1:1 Postscaler
0001 = 1:2 Postscaler
0010 = 1:3 Postscaler
0011 = 1:4 Postscaler
0100 = 1:5 Postscaler
0101 = 1:6 Postscaler
0110 = 1:7 Postscaler
0111 = 1:8 Postscaler
1000 = 1:9 Postscaler
1001 = 1:10 Postscaler
1010 = 1:11 Postscaler
1011 = 1:12 Postscaler
1100 = 1:13 Postscaler
1101 = 1:14 Postscaler
1110 = 1:15 Postscaler
1111 = 1:16 Postscaler

bit 2 **TMR2ON:** Timer2 On bit

1 = Timer2 is on
0 = Timer2 is off

bit 1-0 **T2CKPS<1:0>:** Timer2 Clock Prescale Select bits

00 = Prescaler is 1
01 = Prescaler is 4
1x = Prescaler is 16

TABLE 7-1: SUMMARY OF REGISTERS ASSOCIATED WITH TIMER2

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PR2	Timer2 Module Period Register								1111 1111	1111 1111
TMR2	Holding Register for the 8-bit TMR2 Register								0000 0000	0000 0000
T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used for Timer2 module.

8.0 COMPARATOR MODULE

Comparators are used to interface analog circuits to a digital circuit by comparing two analog voltages and providing a digital indication of their relative magnitudes. The comparators are very useful mixed signal building blocks because they provide analog functionality independent of the program execution. The Analog Comparator module includes the following features:

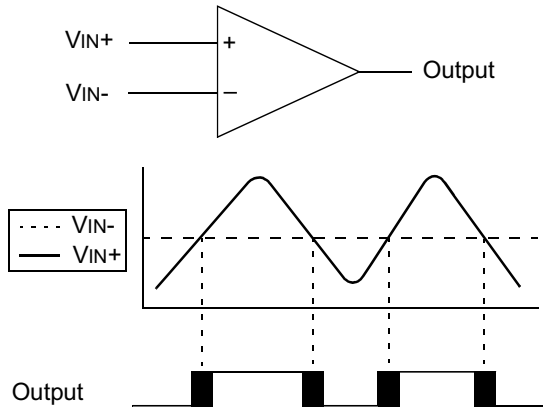
- Dual comparators
- Multiple comparator configurations
- Comparator outputs are available internally/externally
- Programmable output polarity
- Interrupt-on-change
- Wake-up from Sleep
- Timer1 gate (count enable)
- Output synchronization to Timer1 clock input
- Programmable voltage reference

Note:	Only Comparator C2 can be linked to Timer1.

8.1 Comparator Overview

A comparator is shown in Figure 8-1 along with the relationship between the analog input levels and the digital output. When the analog voltage at VIN+ is less than the analog voltage at VIN-, the output of the comparator is a digital low level. When the analog voltage at VIN+ is greater than the analog voltage at VIN-, the output of the comparator is a digital high level.

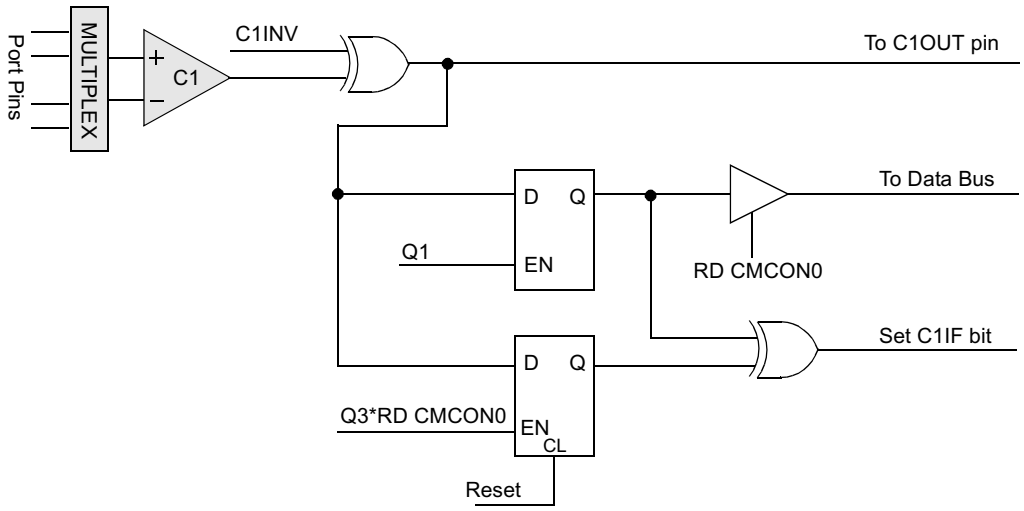
FIGURE 8-1: SINGLE COMPARATOR



Note: The black areas of the output of the comparator represents the uncertainty due to input offsets and response time.

This device contains two comparators as shown in Figure 8-2 and Figure 8-3. The comparators are not independently configurable.

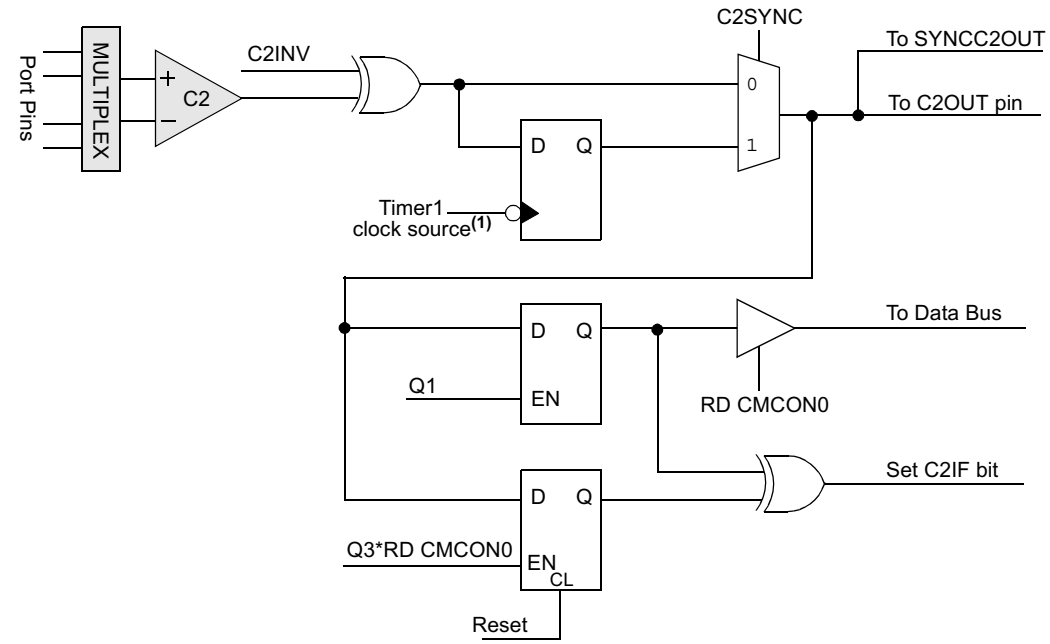
FIGURE 8-2: COMPARATOR C1 OUTPUT BLOCK DIAGRAM



Note 1: Q1 and Q3 are phases of the four-phase system clock (FOSC).

2: Q1 is held high during Sleep mode.

FIGURE 8-3: COMPARATOR C2 OUTPUT BLOCK DIAGRAM



Note 1: Comparator output is latched on falling edge of Timer1 clock source.

2: Q1 and Q3 are phases of the four-phase system clock (FOSC).

3: Q1 is held high during Sleep mode.

8.1.1 ANALOG INPUT CONNECTION CONSIDERATIONS

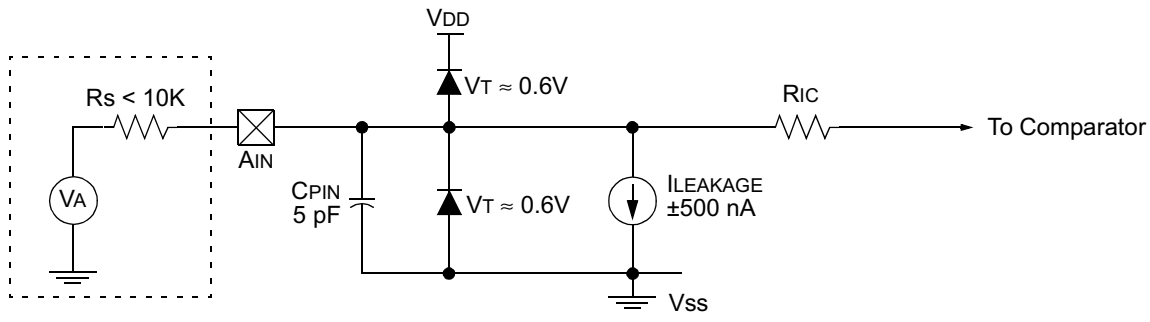
A simplified circuit for an analog input is shown in Figure 8-4. Since the analog input pins share their connection with a digital input, they have reverse biased ESD protection diodes to VDD and VSS. The analog input, therefore, must be between VSS and VDD. If the input voltage deviates from this range by more than 0.6V in either direction, one of the diodes is forward biased and a latch-up may occur.

A maximum source impedance of 10 kΩ is recommended for the analog sources. Also, any external component connected to an analog input pin, such as a capacitor or a Zener diode, should have very little leakage current to minimize inaccuracies introduced.

Note 1: When reading a PORT register, all pins configured as analog inputs will read as a '0'. Pins configured as digital inputs will convert as an analog input, according to the input specification.

2: Analog levels on any pin defined as a digital input, may cause the input buffer to consume more current than is specified.

FIGURE 8-4: ANALOG INPUT MODEL



Legend: CPIN = Input Capacitance
ILEAKAGE = Leakage Current at the pin due to various junctions
RIC = Interconnect Resistance
RS = Source Impedance
VA = Analog Voltage
VT = Threshold Voltage

8.2 Comparator Configuration

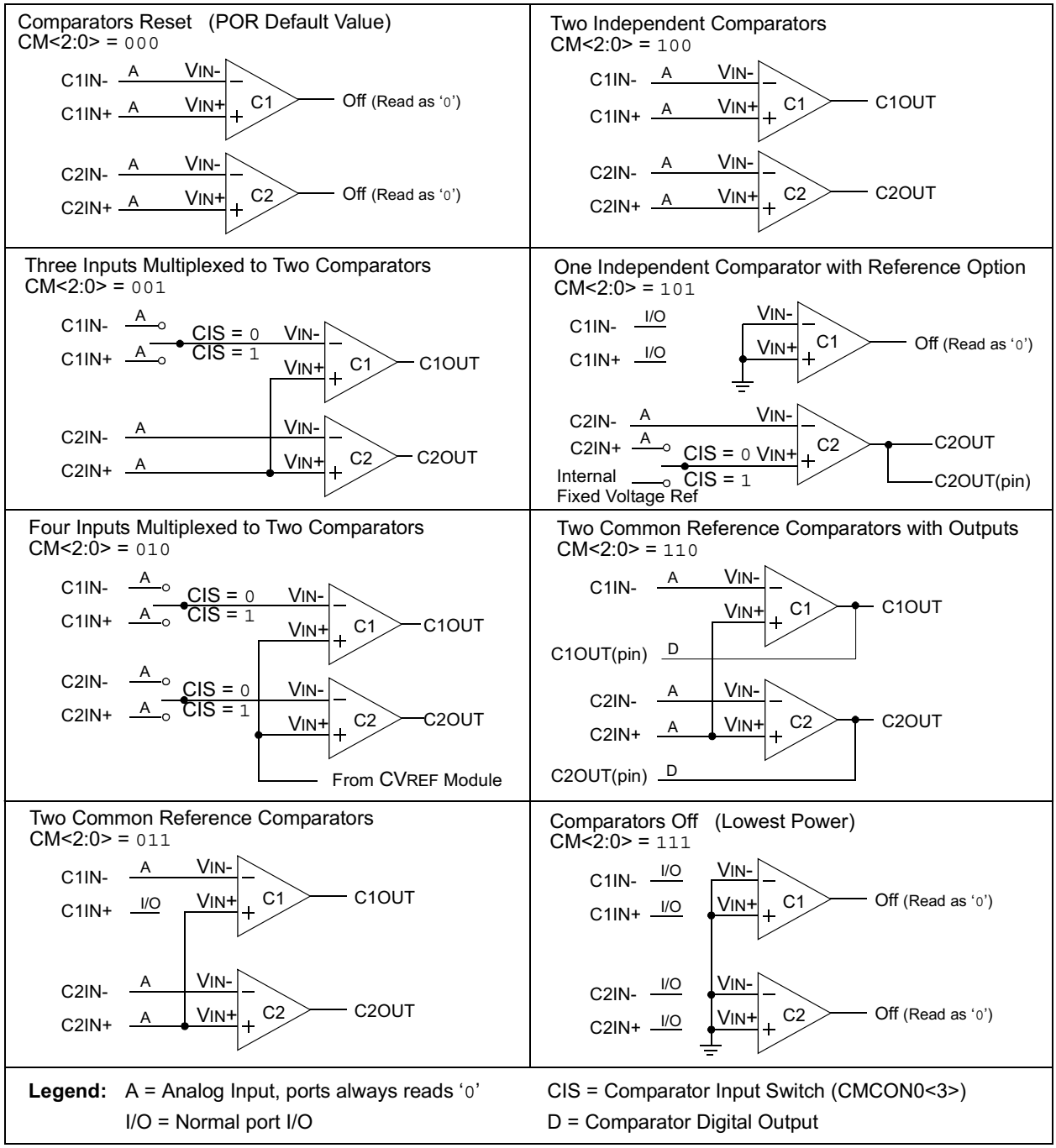
There are eight modes of operation for the comparator. The CM<2:0> bits of the CMCON0 register are used to select these modes as shown in Figure 8-5. I/O lines change as a function of the mode and are designated as follows:

- Analog function (A): digital input buffer is disabled
- Digital function (D): comparator digital output, overrides port function
- Normal port function (I/O): independent of comparator

The port pins denoted as “A” will read as a ‘0’ regardless of the state of the I/O pin or the I/O control TRIS bit. Pins used as analog inputs should also have the corresponding TRIS bit set to ‘1’ to disable the digital output driver. Pins denoted as “D” should have the corresponding TRIS bit set to ‘0’ to enable the digital output driver.

Note:	Comparator interrupts should be disabled during a Comparator mode change to prevent unintended interrupts.

FIGURE 8-5: COMPARATOR I/O OPERATING MODES



Legend: A = Analog Input, ports always reads '0'
I/O = Normal port I/O
CIS = Comparator Input Switch (CMCON0<3>)
D = Comparator Digital Output

8.3 Comparator Control

The CMCON0 register (Register 8-1) provides access to the following comparator features:

- Mode selection
- Output state
- Output polarity
- Input switch

8.3.1 COMPARATOR OUTPUT STATE

Each comparator state can always be read internally via the associated CxOUT bit of the CMCON0 register. The comparator outputs are directed to the CxOUT pins when CM<2:0> = 110. When this mode is selected, the TRIS bits for the associated CxOUT pins must be cleared to enable the output drivers.

8.3.2 COMPARATOR OUTPUT POLARITY

Inverting the output of a comparator is functionally equivalent to swapping the comparator inputs. The polarity of a comparator output can be inverted by setting the CxINV bits of the CMCON0 register. Clearing CxINV results in a non-inverted output. A complete table showing the output state versus input conditions and the polarity bit is shown in Table 8-1.

TABLE 8-1: OUTPUT STATE VS. INPUT CONDITIONS

Input Conditions	CxINV	CxOUT
$V_{IN-} > V_{IN+}$	0	0
$V_{IN-} < V_{IN+}$	0	1
$V_{IN-} > V_{IN+}$	1	1
$V_{IN-} < V_{IN+}$	1	0

Note: CxOUT refers to both the register bit and output pin.

8.3.3 COMPARATOR INPUT SWITCH

The inverting input of the comparators may be switched between two analog pins or an analog input pin and and the fixed voltage reference in the following modes:

- CM<2:0> = 001 (Comparator C1 only)
- CM<2:0> = 010 (Comparators C1 and C2)
- CM<2:0> = 101 (Comparator C2 only)

In the above modes, both pins remain in Analog mode regardless of which pin is selected as the input. The CIS bit of the CMCON0 register controls the comparator input switch.

8.4 Comparator Response Time

The comparator output is indeterminate for a period of time after the change of an input source or the selection of a new reference voltage. This period is referred to as the response time. The response time of the comparator differs from the settling time of the voltage reference. Therefore, both of these times must be considered when determining the total response time to a comparator input change. See the Comparator and Voltage Reference Specifications in **Section 19.0 "Electrical Specifications"** for more details.

8.5 Comparator Interrupt Operation

The comparator interrupt flag is set whenever there is a change in the output value of the comparator. Changes are recognized by means of a mismatch circuit which consists of two latches and an exclusive-or gate (see Figure 8-2 and Figure 8-3). One latch is updated with the comparator output level when the CMCON0 register is read. This latch retains the value until the next read of the CMCON0 register or the occurrence of a Reset. The other latch of the mismatch circuit is updated on every Q1 system clock. A mismatch condition will occur when a comparator output change is clocked through the second latch on the Q1 clock cycle. The mismatch condition will persist, holding the CxIF bit of the PIR2 register true, until either the CMCON0 register is read or the comparator output returns to the previous state.

Note: A write operation to the CMCON0 register will also clear the mismatch condition because all writes include a read operation at the beginning of the write cycle.

Software will need to maintain information about the status of the comparator output to determine the actual change that has occurred.

The CxIF bit of the PIR2 register is the comparator interrupt flag. This bit must be reset in software by clearing it to '0'. Since it is also possible to write a '1' to this register, a simulated interrupt may be initiated.

The CxIE bit of the PIE2 register and the PEIE and GIE bits of the INTCON register must all be set to enable comparator interrupts. If any of these bits are cleared, the interrupt is not enabled, although the CxIF bit of the PIR2 register will still be set if an interrupt condition occurs.

The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

a) Any read or write of CMCON0. This will end the mismatch condition. See Figures 8-6 and 8-7

b) Clear the CxIF interrupt flag.

A persistent mismatch condition will preclude clearing the CxIF interrupt flag. Reading CMCON0 will end the mismatch condition and allow the CxIF bit to be cleared.

FIGURE 8-6: COMPARATOR INTERRUPT TIMING W/O CMCON0 READ

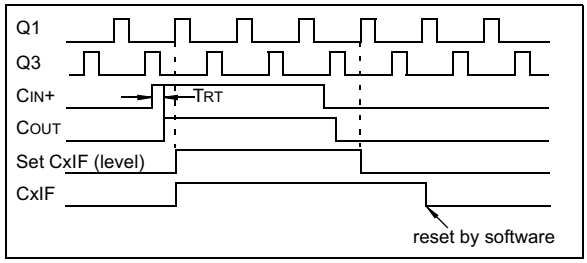
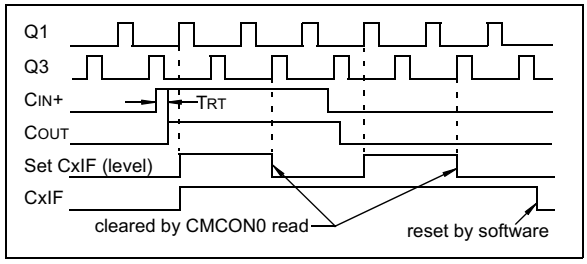


FIGURE 8-7: COMPARATOR INTERRUPT TIMING WITH CMCON0 READ



Note 1: If a change in the CMCON0 register (CxOUT) occurs when a read operation is being executed (start of the Q2 cycle), then the CxIF Interrupt Flag bit of the PIR2 register may not get set.

2: When either comparator is first enabled, bias circuitry in the Comparator module may cause an invalid output from the comparator until the bias circuitry is stable. Allow about 1 μs for bias settling then clear the mismatch condition and interrupt flags before enabling comparator interrupts.

8.6 Operation During Sleep

The comparator, if enabled before entering Sleep mode, remains active during Sleep. The additional current consumed by the comparator is shown separately in the **Section 19.0 "Electrical Specifications"**. If the comparator is not used to wake the device, power consumption can be minimized while in Sleep mode by turning off the comparator. The comparator is turned off by selecting mode CM<2:0> = 000 or CM<2:0> = 111 of the CMCON0 register.

A change to the comparator output can wake-up the device from Sleep. To enable the comparator to wake the device from Sleep, the CxIE bit of the PIE2 register and the PEIE bit of the INTCON register must be set. The instruction following the Sleep instruction always executes following a wake from Sleep. If the GIE bit of the INTCON register is also set, the device will then execute the Interrupt Service Routine.

8.7 Effects of a Reset

A device Reset forces the CMCON0 and CMCON1 registers to their Reset states. This forces the Comparator module to be in the Comparator Reset mode (CM<2:0> = 000). Thus, all comparator inputs are analog inputs with the comparator disabled to consume the smallest current possible.

REGISTER 8-1: CMCON0: COMPARATOR CONFIGURATION REGISTER

R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **C2OUT:** Comparator 2 Output bit
When C2INV = 0:
1 = C2 VIN+ > C2 VIN-
0 = C2 VIN+ < C2 VIN-
When C2INV = 1:
1 = C2 VIN+ < C2 VIN-
0 = C2 VIN+ > C2 VIN-

bit 6 **C1OUT:** Comparator 1 Output bit
When C1INV = 0:
1 = C1 VIN+ > C1 VIN-
0 = C1 VIN+ < C1 VIN-
When C1INV = 1:
1 = C1 VIN+ < C1 VIN-
0 = C1 VIN+ > C1 VIN-

bit 5 **C2INV:** Comparator 2 Output Inversion bit
1 = C2 output inverted
0 = C2 output not inverted

bit 4 **C1INV:** Comparator 1 Output Inversion bit
1 = C1 Output inverted
0 = C1 Output not inverted

bit 3 **CIS:** Comparator Input Switch bit
When CM<2:0> = 010:
1 = C1IN+ connects to C1 VIN-
C2IN+ connects to C2 VIN-
0 = C1IN- connects to C1 VIN-
C2IN- connects to C2 VIN-
When CM<2:0> = 001:
1 = C1IN+ connects to C1 VIN-
0 = C1IN- connects to C1 VIN-
When CM<2:0> = 101: (16F91x/946)
1 = C2 VIN+ connects to fixed voltage reference
0 = C2 VIN+ connects to C2IN+

bit 2-0 **CM<2:0>:** Comparator Mode bits (See Figure 8-5)
000 = Comparators off. CxIN pins are configured as analog
001 = Three inputs multiplexed to two comparators
010 = Four inputs multiplexed to two comparators
011 = Two common reference comparators
100 = Two independent comparators
101 = One independent comparator
110 = Two comparators with outputs and common reference
111 = Comparators off. CxIN pins are configured as digital I/O

8.8 Comparator C2 Gating Timer1

This feature can be used to time the duration or interval of analog events. Clearing the T1GSS bit of the CMCON1 register will enable Timer1 to increment based on the output of Comparator C2. This requires that Timer1 is on and gating is enabled. See **Section 6.0 "Timer1 Module with Gate Control"** for details.

It is recommended to synchronize Comparator C2 with Timer1 by setting the C2SYNC bit when the comparator is used as the Timer1 gate source. This ensures Timer1 does not miss an increment if the comparator changes during an increment.

8.9 Synchronizing Comparator C2 Output to Timer1

The output of Comparator C2 can be synchronized with Timer1 by setting the C2SYNC bit of the CMCON1 register. When enabled, the comparator output is latched on the falling edge of the Timer1 clock source. If a prescaler is used with Timer1, the comparator output is latched after the prescaling function. To prevent a race condition, the comparator output is latched on the falling edge of the Timer1 clock source and Timer1 increments on the rising edge of its clock source. Reference the comparator block diagrams (Figure 8-2 and Figure 8-3) and the Timer1 Block Diagram (Figure 6-1) for more information.

REGISTER 8-2: CMCON1: COMPARATOR CONFIGURATION REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0
—	—	—	—	—	—	T1GSS	C2SYNC
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-2 **Unimplemented:** Read as '0'

bit 1 **T1GSS:** Timer1 Gate Source Select bit(1)
- 1 = Timer1 gate source is $\overline{T1G}$ pin (pin should be configured as digital input)
- 0 = Timer1 gate source is Comparator C2 output

bit 0 **C2SYNC:** Comparator C2 Output Synchronization bit(2)
- 1 = Output is synchronized with falling edge of Timer1 clock
- 0 = Output is asynchronous

Note 1: Refer to **Section 6.6 "Timer1 Gate"**.
2: Refer to Figure 8-3.

8.10 Comparator Voltage Reference

The Comparator Voltage Reference module provides an internally generated voltage reference for the comparators. The following features are available:

- Independent from Comparator operation
- Two 16-level voltage ranges
- Output clamped to VSS
- Ratiometric with VDD

The VRCON register (Register 8-3) controls the Voltage Reference module shown in Figure 8-8.

8.10.1 INDEPENDENT OPERATION

The comparator voltage reference is independent of the comparator configuration. Setting the VREN bit of the VRCON register will enable the voltage reference.

8.10.2 OUTPUT VOLTAGE SELECTION

The CVREF voltage reference has 2 ranges with 16 voltage levels in each range. Range selection is controlled by the VRR bit of the VRCON register. The 16 levels are set with the VR<3:0> bits of the VRCON register.

The CVREF output voltage is determined by the following equations:

EQUATION 8-1: CVREF OUTPUT VOLTAGE

$$\begin{aligned} &VRR = 1 \text{ (low range):} \\ &\quad CVREF = (VR<3:0>/24) \times VDD \\ &VRR = 0 \text{ (high range):} \\ &\quad CVREF = (VDD/4) + (VR<3:0> \times VDD/32) \end{aligned}$$

The full range of VSS to VDD cannot be realized due to the construction of the module. See Figure 8-8.

8.10.3 OUTPUT CLAMPED TO VSS

The CVREF output voltage can be set to Vss with no power consumption by configuring VRCON as follows:

- VREN = 0
- VRR = 1
- VR<3:0> = 0000

This allows the comparator to detect a zero-crossing while not consuming additional CVREF module current.

8.10.4 OUTPUT RATIOMETRIC TO VDD

The comparator voltage reference is VDD derived and therefore, the CVREF output changes with fluctuations in VDD. The tested absolute accuracy of the Comparator Voltage Reference can be found in **Section 19.0 "Electrical Specifications"**.

REGISTER 8-3: VRCON: VOLTAGE REFERENCE CONTROL REGISTER

R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
VREN	—	VRR	—	VR3	VR2	VR1	VR0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **VREN:** CVREF Enable bit
1 = CVREF circuit powered on
0 = CVREF circuit powered down, no IDD drain and CVREF = VSS.

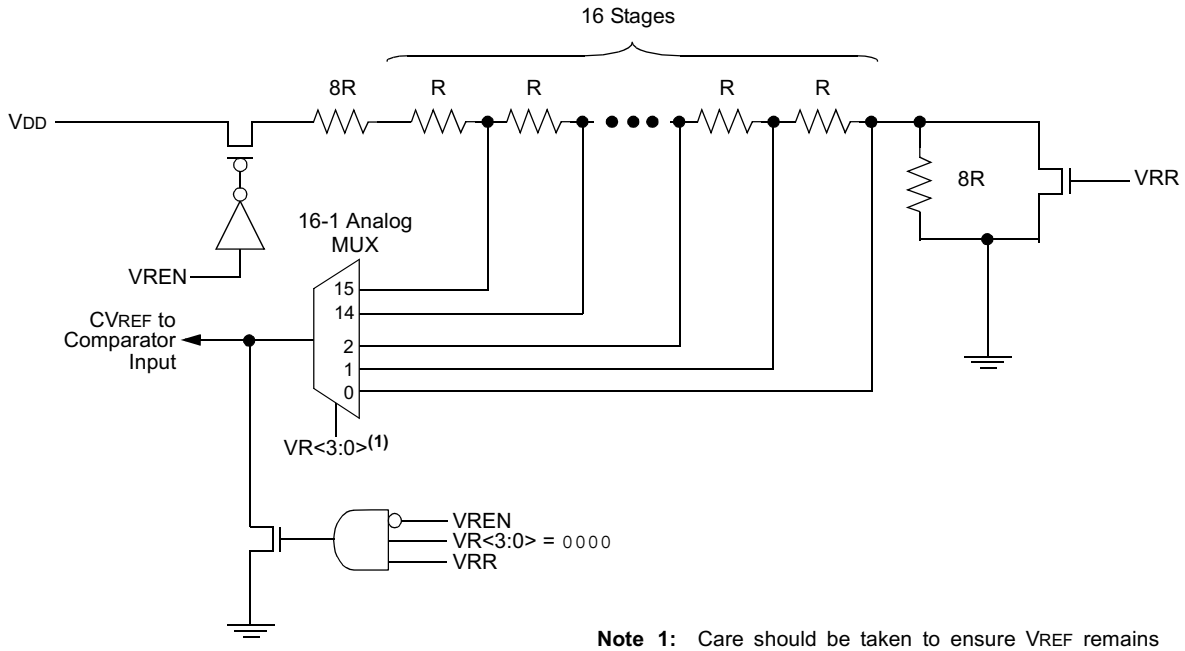
bit 6 **Unimplemented:** Read as '0'

bit 5 **VRR:** CVREF Range Selection bit
1 = Low range
0 = High range

bit 4 **Unimplemented:** Read as '0'

bit 3-0 **VR<3:0>:** CVREF Value Selection bits ($0 \leq$ VR<3:0> ≤ 15)
When VRR = 1: CVREF = (VR<3:0>/24) * VDD
When VRR = 0: CVREF = VDD/4 + (VR<3:0>/32) * VDD

FIGURE 8-8: COMPARATOR VOLTAGE REFERENCE BLOCK DIAGRAM



Note 1: Care should be taken to ensure VREF remains within the comparator common mode input range. See **Section 19.0 "Electrical Specifications"** for more detail.

TABLE 8-2: SUMMARY OF REGISTERS ASSOCIATED WITH THE COMPARATOR AND VOLTAGE REFERENCE MODULES

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
CMCON0	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0000	0000 0000
CMCON1	—	—	—	—	—	—	T1GSS	C2SYNC	---- --10	---- --10
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE2	OSFIE	C2IE	C1IE	LCDIE	—	LVDIE	—	CCP2IE	0000 -0-0	0000 -0-0
PIR2	OSFIF	C2IF	C1IF	LCDIF	—	LVDIF	—	CCP2IF	0000 -0-0	0000 -0-0
PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx xxxx	uuuu uuuu
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111
VRCON	VREN	—	VRR	—	VR3	VR2	VR1	VR0	0-0- 0000	0000 0000

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used for comparator.

NOTES:

9.0 ADDRESSABLE UNIVERSAL SYNCHRONOUS ASYNCHRONOUS RECEIVER TRANSMITTER (AUSART)

The Addressable Universal Synchronous Asynchronous Receiver Transmitter (AUSART) module is a serial I/O communications peripheral. It contains all the clock generators, shift registers and data buffers necessary to perform an input or output serial data transfer independent of device program execution. The AUSART, also known as a Serial Communications Interface (SCI), can be configured as a full-duplex asynchronous system or half-duplex synchronous system. Full-Duplex mode is useful for communications with peripheral systems, such as CRT terminals and personal computers. Half-Duplex Synchronous mode is intended for communications with peripheral devices, such as A/D or D/A integrated circuits, serial EEPROMs or other microcontrollers. These devices typically do not have internal clocks for baud rate generation and require the external clock signal provided by a master synchronous device.

The AUSART module includes the following capabilities:

- Full-duplex asynchronous transmit and receive
- Two-character input buffer
- One-character output buffer
- Programmable 8-bit or 9-bit character length
- Address detection in 9-bit mode
- Input buffer overrun error detection
- Received character framing error detection
- Half-duplex synchronous master
- Half-duplex synchronous slave
- Sleep operation

Block diagrams of the AUSART transmitter and receiver are shown in Figure 9-1 and Figure 9-2.

FIGURE 9-1: AUSART TRANSMIT BLOCK DIAGRAM

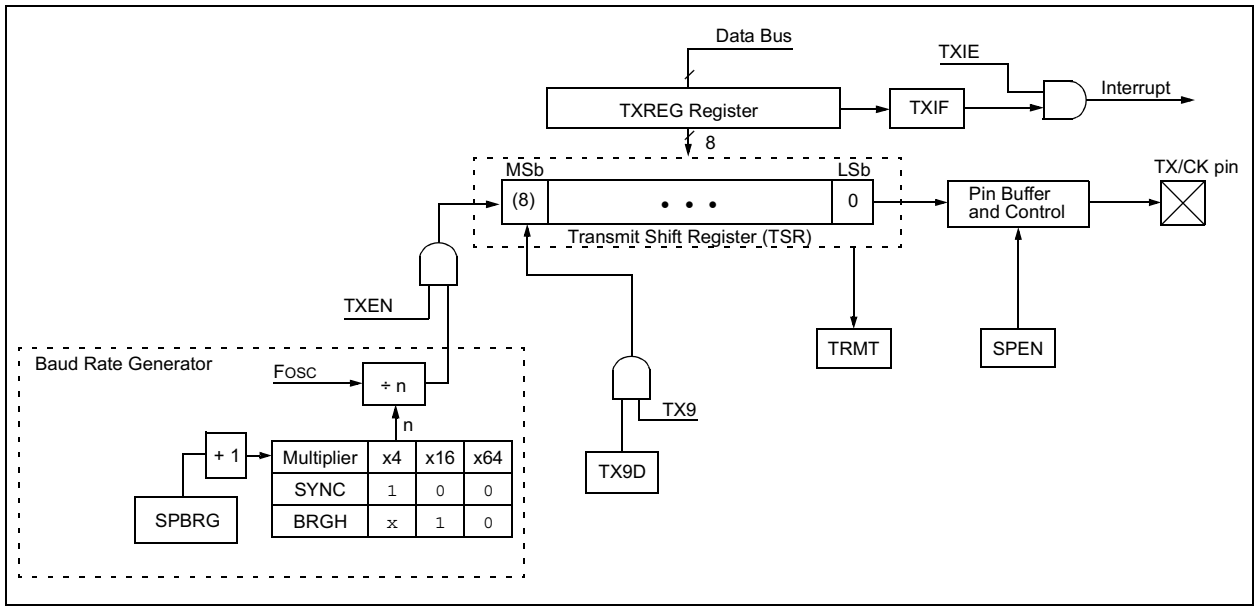
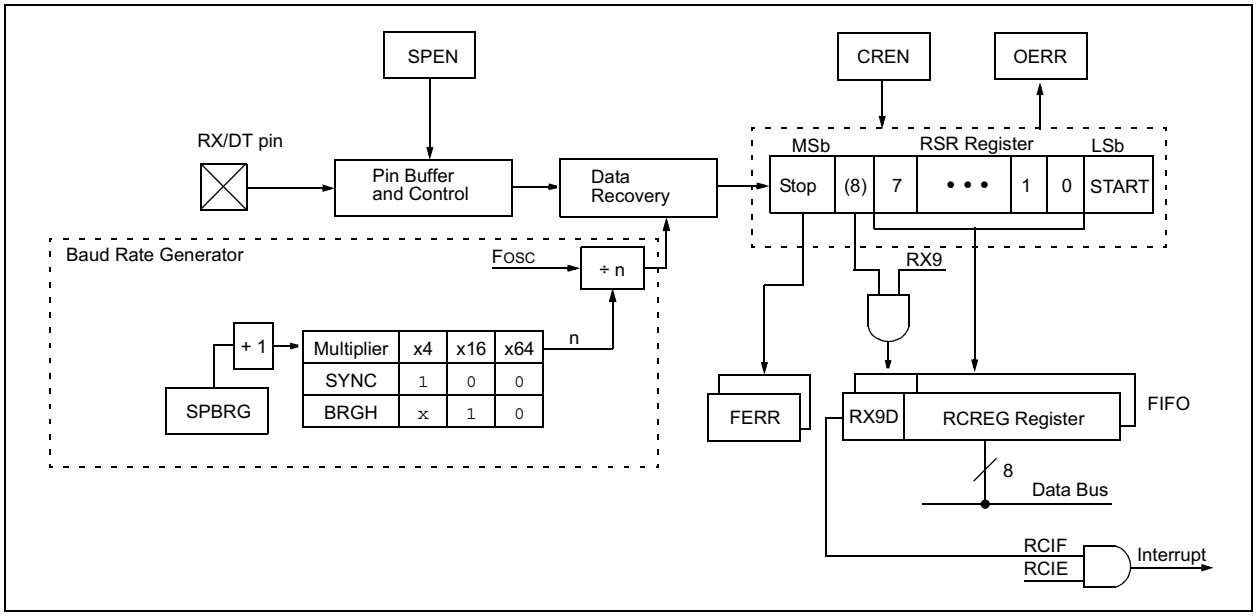


FIGURE 9-2: AUSART RECEIVE BLOCK DIAGRAM



The operation of the AUSART module is controlled through two registers:

- Transmit Status and Control (TXSTA)
- Receive Status and Control (RCSTA)

These registers are detailed in Register 9-1 and Register 9-2 respectively.

9.1 AUSART Asynchronous Mode

The AUSART transmits and receives data using the standard non-return-to-zero (NRZ) format. NRZ is implemented with two levels: a VOH mark state which represents a '1' data bit, and a VOL space state which represents a '0' data bit. NRZ refers to the fact that consecutively transmitted data bits of the same value stay at the output level of that bit without returning to a neutral level between each bit transmission. An NRZ transmission port idles in the mark state. Each character transmission consists of one Start bit followed by eight or nine data bits and is always terminated by one or more Stop bits. The Start bit is always a space and the Stop bits are always marks. The most common data format is 8 bits. Each transmitted bit persists for a period of 1/(Baud Rate). An on-chip dedicated 8-bit Baud Rate Generator is used to derive standard baud rate frequencies from the system oscillator. See Table 9-5 for examples of baud rate configurations.

The AUSART transmits and receives the LSb first. The AUSART's transmitter and receiver are functionally independent, but share the same data format and baud rate. Parity is not supported by the hardware, but can be implemented in software and stored as the ninth data bit.

9.1.1 AUSART ASYNCHRONOUS TRANSMITTER

The AUSART transmitter block diagram is shown in Figure 9-1. The heart of the transmitter is the serial Transmit Shift Register (TSR), which is not directly accessible by software. The TSR obtains its data from the transmit buffer, which is the TXREG register.

9.1.1.1 Enabling the Transmitter

The AUSART transmitter is enabled for asynchronous operations by configuring the following three control bits:

- TXEN = 1
- SYNC = 0
- SPEN = 1

All other AUSART control bits are assumed to be in their default state.

Setting the TXEN bit of the TXSTA register enables the transmitter circuitry of the AUSART. Clearing the SYNC bit of the TXSTA register configures the AUSART for asynchronous operation. Setting the SPEN bit of the RCSTA register enables the AUSART and automatically configures the TX/CK I/O pin as an output.

The LCD SEG9 function must be disabled by clearing the SE9 bit of the LCDSE1 register, if the TX/CK pin is shared with the LCD peripheral.

Note 1: When the SPEN bit is set the RX/DT I/O pin is automatically configured as an input, regardless of the state of the corresponding TRIS bit and whether or not the AUSART receiver is enabled. The RX/DT pin data can be read via a normal PORT read but PORT latch data output is precluded.

2: The TXIF transmitter interrupt flag is set when the TXEN enable bit is set.

9.1.1.2 Transmitting Data

A transmission is initiated by writing a character to the TXREG register. If this is the first character, or the previous character has been completely flushed from the TSR, the data in the TXREG is immediately transferred to the TSR register. If the TSR still contains all or part of a previous character, the new character data is held in the TXREG until the Stop bit of the previous character has been transmitted. The pending character in the TXREG is then transferred to the TSR in one TCY immediately following the Stop bit transmission. The transmission of the Start bit, data bits and Stop bit sequence commences immediately following the transfer of the data to the TSR from the TXREG.

9.1.1.3 Transmit Interrupt Flag

The TXIF interrupt flag bit of the PIR1 register is set whenever the AUSART transmitter is enabled and no character is being held for transmission in the TXREG. In other words, the TXIF bit is only clear when the TSR is busy with a character and a new character has been queued for transmission in the TXREG. The TXIF flag bit is not cleared immediately upon writing TXREG. TXIF becomes valid in the second instruction cycle following the write execution. Polling TXIF immediately following the TXREG write will return invalid results. The TXIF bit is read-only, it cannot be set or cleared by software.

The TXIF interrupt can be enabled by setting the TXIE interrupt enable bit of the PIE1 register. However, the TXIF flag bit will be set whenever the TXREG is empty, regardless of the state of TXIE enable bit.

To use interrupts when transmitting data, set the TXIE bit only when there is more data to send. Clear the TXIE interrupt enable bit upon writing the last character of the transmission to the TXREG.

9.1.1.4 TSR Status

The TRMT bit of the TXSTA register indicates the status of the TSR register. This is a read-only bit. The TRMT bit is set when the TSR register is empty and is cleared when a character is transferred to the TSR register from the TXREG. The TRMT bit remains clear until all bits have been shifted out of the TSR register. No interrupt logic is tied to this bit, so the user has to poll this bit to determine the TSR status.

Note:	The TSR register is not mapped in data memory, so it is not available to the user.

9.1.1.5 Transmitting 9-Bit Characters

The AUSART supports 9-bit character transmissions. When the TX9 bit of the TXSTA register is set the AUSART will shift 9 bits out for each character transmitted. The TX9D bit of the TXSTA register is the ninth, and Most Significant, data bit. When transmitting 9-bit data, the TX9D data bit must be written before writing the 8 Least Significant bits into the TXREG. All nine bits of data will be transferred to the TSR shift register immediately after the TXREG is written.

A special 9-bit Address mode is available for use with multiple receivers. See **Section 9.1.2.7 "Address Detection"** for more information on the Address mode.

9.1.1.6 Asynchronous Transmission Set-up:

1. Initialize the SPBRG register and the BRGH bit to achieve the desired baud rate (see **Section 9.2 "AUSART Baud Rate Generator (BRG)"**).
2. Enable the asynchronous serial port by clearing the SYNC bit and setting the SPEN bit.
3. If 9-bit transmission is desired, set the TX9 control bit. A set ninth data bit will indicate that the 8 Least Significant data bits are an address when the receiver is set for address detection.
4. Enable the transmission by setting the TXEN control bit. This will cause the TXIF interrupt bit to be set.
5. If interrupts are desired, set the TXIE interrupt enable bit of the PIE1 register. An interrupt will occur immediately provided that the GIE and PEIE bits of the INTCON register are also set.
6. If 9-bit transmission is selected, the ninth bit should be loaded into the TX9D data bit.
7. Load 8-bit data into the TXREG register. This will start the transmission.

FIGURE 9-3: ASYNCHRONOUS TRANSMISSION

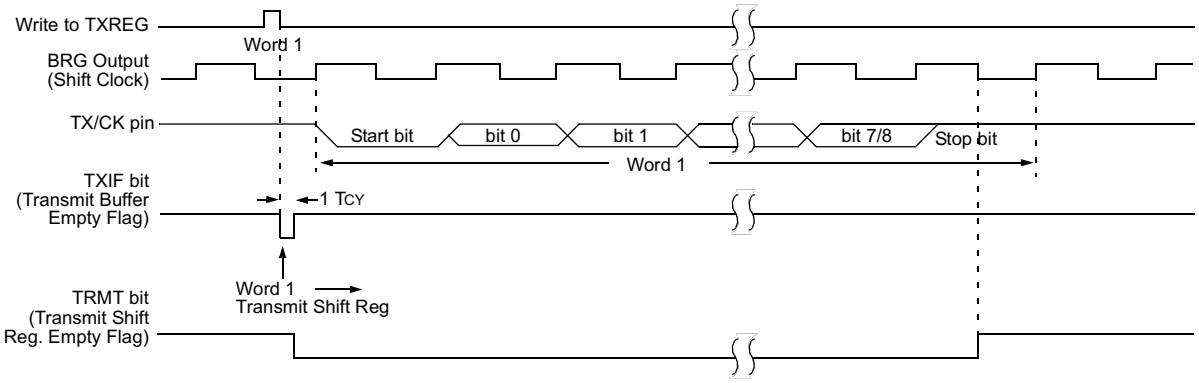
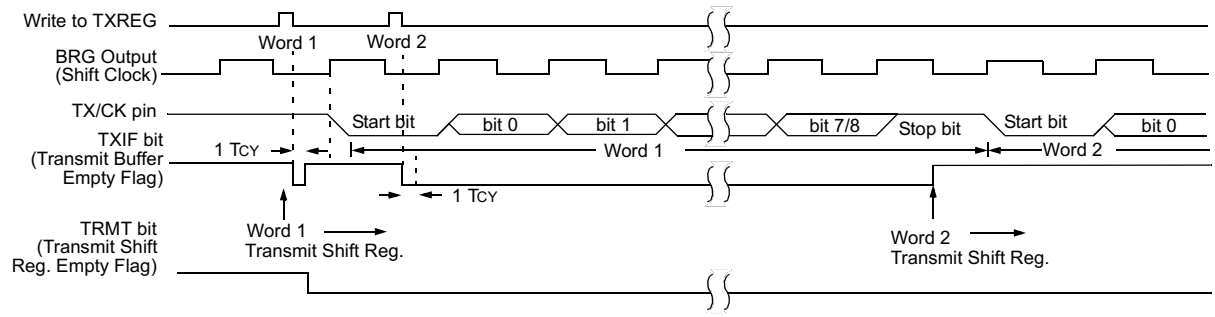


FIGURE 9-4: ASYNCHRONOUS TRANSMISSION (BACK-TO-BACK)



Note: This timing diagram shows two consecutive transmissions.

TABLE 9-1: REGISTERS ASSOCIATED WITH ASYNCHRONOUS TRANSMISSION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	0000 0000
PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
SPBRG	BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0	0000 0000	0000 0000
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
TXREG	AUSART Transmit Data Register								0000 0000	0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for Asynchronous Transmission.

9.1.2 AUSART ASYNCHRONOUS RECEIVER

The Asynchronous mode is typically used in RS-232 systems. The receiver block diagram is shown in Figure 9-2. The data is received on the RX/DT pin and drives the data recovery block. The data recovery block is actually a high-speed shifter operating at 16 times the baud rate, whereas the serial Receive Shift Register (RSR) operates at the bit rate. When all 8 or 9 bits of the character have been shifted in, they are immediately transferred to a two character First-In First-Out (FIFO) memory. The FIFO buffering allows reception of two complete characters and the start of a third character before software must start servicing the AUSART receiver. The FIFO and RSR registers are not directly accessible by software. Access to the received data is via the RCREG register.

9.1.2.1 Enabling the Receiver

The AUSART receiver is enabled for asynchronous operation by configuring the following three control bits:

- CREN = 1
- SYNC = 0
- SPEN = 1

All other AUSART control bits are assumed to be in their default state.

Setting the CREN bit of the RCSTA register enables the receiver circuitry of the AUSART. Clearing the SYNC bit of the TXSTA register configures the AUSART for asynchronous operation. Setting the SPEN bit of the RCSTA register enables the AUSART and automatically configures the RX/DT I/O pin as an input.

The LCD SEG8 function must be disabled by clearing the SE8 bit of the LCDSE1 register, if the RX/DT pin is shared with the LCD peripheral.

Note:	When the SPEN bit is set the TX/CK I/O pin is automatically configured as an output, regardless of the state of the corresponding TRIS bit and whether or not the AUSART transmitter is enabled. The PORT latch is disconnected from the output driver so it is not possible to use the TX/CK pin as a general purpose output.

9.1.2.2 Receiving Data

The receiver data recovery circuit initiates character reception on the falling edge of the first bit. The first bit, also known as the Start bit, is always a zero. The data recovery circuit counts one-half bit time to the center of the Start bit and verifies that the bit is still a zero. If it is not a zero then the data recovery circuit aborts character reception, without generating an error, and resumes looking for the falling edge of the Start bit. If the Start bit zero verification succeeds then the data recovery circuit counts a full bit time to the center of the next bit. The bit is then sampled by a majority detect circuit and the resulting '0' or '1' is shifted into the RSR. This repeats until all data bits have been sampled and shifted into the RSR. One final bit time is measured and the level sampled. This is the Stop bit, which is always a '1'. If the data recovery circuit samples a '0' in the Stop bit position then a framing error is set for this character, otherwise the framing error is cleared for this character. See **Section 9.1.2.4 "Receive Framing Error"** for more information on framing errors.

Immediately after all data bits and the Stop bit have been received, the character in the RSR is transferred to the AUSART receive FIFO and the RCIF interrupt flag bit of the PIR1 register is set. The top character in the FIFO is transferred out of the FIFO by reading the RCREG register.

Note:	If the receive FIFO is overrun, no additional characters will be received until the overrun condition is cleared. See ***Section 9.1.2.5 "Receive Overrun Error"*** for more information on overrun errors.

9.1.2.3 Receive Interrupts

The RCIF interrupt flag bit of the PIR1 register is set whenever the AUSART receiver is enabled and there is an unread character in the receive FIFO. The RCIF interrupt flag bit is read-only, it cannot be set or cleared by software.

RCIF interrupts are enabled by setting all of the following bits:

- RCIE interrupt enable bit of the PIE1 register
- PEIE peripheral interrupt enable bit of the INTCON register
- GIE global interrupt enable bit of the INTCON register

The RCIF interrupt flag bit of the PIR1 register will be set when there is an unread character in the FIFO, regardless of the state of interrupt enable bits.

9.1.2.4 Receive Framing Error

Each character in the receive FIFO buffer has a corresponding framing error Status bit. A framing error indicates that a Stop bit was not seen at the expected time. The framing error status is accessed via the FERR bit of the RCSTA register. The FERR bit represents the status of the top unread character in the receive FIFO. Therefore, the FERR bit must be read before reading the RCREG.

The FERR bit is read-only and only applies to the top unread character in the receive FIFO. A framing error (FERR = 1) does not preclude reception of additional characters. It is not necessary to clear the FERR bit. Reading the next character from the FIFO buffer will advance the FIFO to the next character and the next corresponding framing error.

The FERR bit can be forced clear by clearing the SPEN bit of the RCSTA register which resets the AUSART. Clearing the CREN bit of the RCSTA register does not affect the FERR bit. A framing error by itself does not generate an interrupt.

Note:	If all receive characters in the receive FIFO have framing errors, repeated reads of the RCREG will not clear the FERR bit.

9.1.2.5 Receive Overrun Error

The receive FIFO buffer can hold two characters. An overrun error will be generated If a third character, in its entirety, is received before the FIFO is accessed. When this happens the OERR bit of the RCSTA register is set. The characters already in the FIFO buffer can be read but no additional characters will be received until the error is cleared. The error must be cleared by either clearing the CREN bit of the RCSTA register.

9.1.2.6 Receiving 9-bit Characters

The AUSART supports 9-bit character reception. When the RX9 bit of the RCSTA register is set the AUSART will shift 9 bits into the RSR for each character received. The RX9D bit of the RCSTA register is the ninth and Most Significant data bit of the top unread character in the receive FIFO. When reading 9-bit data from the receive FIFO buffer, the RX9D data bit must be read before reading the 8 Least Significant bits from the RCREG.

9.1.2.7 Address Detection

A special Address Detection mode is available for use when multiple receivers share the same transmission line, such as in RS-485 systems. Address detection is enabled by setting the ADDEN bit of the RCSTA register.

Address detection requires 9-bit character reception. When address detection is enabled, only characters with the ninth data bit set will be transferred to the receive FIFO buffer, thereby setting the RCIF interrupt bit of the PIR1 register. All other characters will be ignored.

Upon receiving an address character, user software determines if the address matches its own. Upon address match, user software must disable address detection by clearing the ADDEN bit before the next Stop bit occurs. When user software detects the end of the message, determined by the message protocol used, software places the receiver back into the Address Detection mode by setting the ADDEN bit.

9.1.2.8 Asynchronous Reception Set-up:

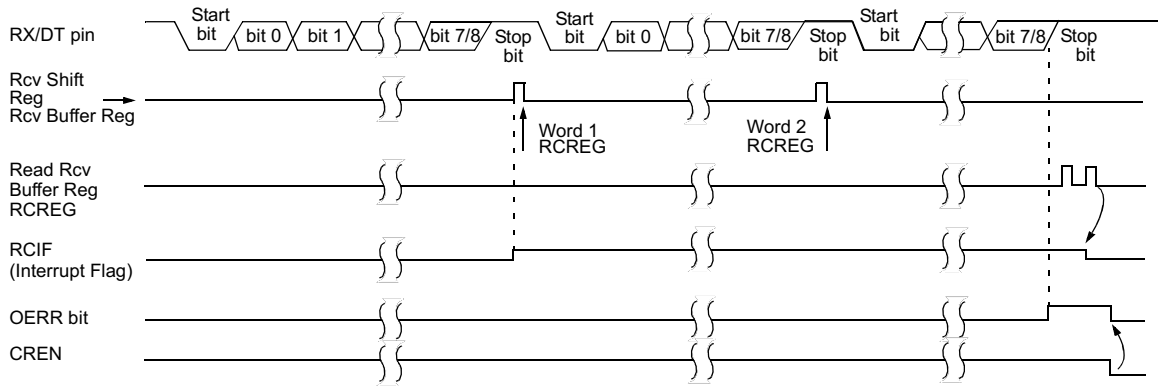
1. Initialize the SPBRG register and the BRGH bit to achieve the desired baud rate (see **Section 9.2 "AUSART Baud Rate Generator (BRG)"**).
2. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
3. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
4. If 9-bit reception is desired, set the RX9 bit.
5. Enable reception by setting the CREN bit.
6. The RCIF interrupt flag bit of the PIR1 register will be set when a character is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE bit of the PIE1 register was also set.
7. Read the RCSTA register to get the error flags and, if 9-bit data reception is enabled, the ninth data bit.
8. Get the received 8 Least Significant data bits from the receive buffer by reading the RCREG register.
9. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.

9.1.2.9 9-bit Address Detection Mode Set-up

This mode would typically be used in RS-485 systems. To set up an Asynchronous Reception with Address Detect Enable:

1. Initialize the SPBRG register and the BRGH bit to achieve the desired baud rate (see **Section 9.2 "AUSART Baud Rate Generator (BRG)"**).
2. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
3. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
4. Enable 9-bit reception by setting the RX9 bit.
5. Enable address detection by setting the ADDEN bit.
6. Enable reception by setting the CREN bit.
7. The RCIF interrupt flag bit of the PIR1 register will be set when a character with the ninth bit set is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit of the PIE1 register was also set.
8. Read the RCSTA register to get the error flags. The ninth data bit will always be set.
9. Get the received 8 Least Significant data bits from the receive buffer by reading the RCREG register. Software determines if this is the device's address.
10. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.
11. If the device has been addressed, clear the ADDEN bit to allow all received data into the receive buffer and generate interrupts.

FIGURE 9-5: ASYNCHRONOUS RECEPTION



Note: This timing diagram shows three words appearing on the RX input. The RCREG (receive buffer) is read after the third word, causing the OERR (overrun) bit to be set.

TABLE 9-2: REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	0000 0000
PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCREG	AUSART Receive Data Register								0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
SPBRG	BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0	0000 0000	0000 0000
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for Asynchronous Reception.

REGISTER 9-1: TXSTA: TRANSMIT STATUS AND CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R-1	R/W-0
CSRC	TX9	TXEN(1)	SYNC	—	BRGH	TRMT	TX9D
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **CSRC:** Clock Source Select bit
Asynchronous mode:
Don't care
Synchronous mode:
1 = Master mode (clock generated internally from BRG)
0 = Slave mode (clock from external source)

bit 6 **TX9:** 9-bit Transmit Enable bit
1 = Selects 9-bit transmission
0 = Selects 8-bit transmission

bit 5 **TXEN:** Transmit Enable bit(1)
1 = Transmit enabled
0 = Transmit disabled

bit 4 **SYNC:** AUSART Mode Select bit
1 = Synchronous mode
0 = Asynchronous mode

bit 3 **Unimplemented:** Read as '0'

bit 2 **BRGH:** High Baud Rate Select bit
Asynchronous mode:
1 = High speed
0 = Low speed
Synchronous mode:
Unused in this mode

bit 1 **TRMT:** Transmit Shift Register Status bit
1 = TSR empty
0 = TSR full

bit 0 **TX9D:** Ninth bit of Transmit Data
Can be address/data bit or a parity bit.

Note 1: SREN/CREN overrides TXEN in Sync mode.

REGISTER 9-2: RCSTA: RECEIVE STATUS AND CONTROL REGISTER(1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0	R-0	R-x
SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **SPEN:** Serial Port Enable bit
1 = Serial port enabled (configures RX/DT and TX/CK pins as serial port pins)
0 = Serial port disabled (held in Reset)

bit 6 **RX9:** 9-bit Receive Enable bit
1 = Selects 9-bit reception
0 = Selects 8-bit reception

bit 5 **SREN:** Single Receive Enable bit
Asynchronous mode:
Don't care
Synchronous mode – Master:
1 = Enables single receive
0 = Disables single receive
This bit is cleared after reception is complete.
Synchronous mode – Slave
Don't care

bit 4 **CREN:** Continuous Receive Enable bit
Asynchronous mode:
1 = Enables receiver
0 = Disables receiver
Synchronous mode:
1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN)
0 = Disables continuous receive

bit 3 **ADDEN:** Address Detect Enable bit
Asynchronous mode 9-bit (RX9 = 1):
1 = Enables address detection, enable interrupt and load the receive buffer when RSR<8> is set
0 = Disables address detection, all bytes are received and ninth bit can be used as parity bit
Asynchronous mode 8-bit (RX9 = 0):
Don't care
Synchronous mode:
Must be set to '0'

bit 2 **FERR:** Framing Error bit
1 = Framing error (can be updated by reading RCREG register and receive next valid byte)
0 = No framing error

bit 1 **OERR:** Overrun Error bit
1 = Overrun error (can be cleared by clearing bit CREN)
0 = No overrun error

bit 0 **RX9D:** Ninth bit of Received Data
This can be address/data bit or a parity bit and must be calculated by user firmware.

9.2 AUSART Baud Rate Generator (BRG)

The Baud Rate Generator (BRG) is an 8-bit timer that is dedicated to the support of both the asynchronous and synchronous AUSART operation.

The SPBRG register determines the period of the free running baud rate timer. In Asynchronous mode the multiplier of the baud rate period is determined by the BRGH bit of the TXSTA register. In Synchronous mode, the BRGH bit is ignored.

Table 9-3 contains the formulas for determining the baud rate. Example 9-1 provides a sample calculation for determining the baud rate and baud rate error.

Typical baud rates and error values for various asynchronous modes have been computed for your convenience and are shown in Table 9-3. It may be advantageous to use the high baud rate (BRGH = 1), to reduce the baud rate error.

Writing a new value to the SPBRG register causes the BRG timer to be reset (or cleared). This ensures that the BRG does not wait for a timer overflow before outputting the new baud rate.

EXAMPLE 9-1: CALCULATING BAUD RATE ERROR

For a device with FOSC of 16 MHz, desired baud rate of 9600, Asynchronous mode:

$$Desired\ Baud\ Rate = \frac{FOSC}{64(SPBRG + 1)}$$

Solving for SPBRG:

$$X = \frac{\frac{FOSC}{Desired\ Baud\ Rate}}{64} - 1$$

$$= \frac{\frac{16000000}{9600}}{64} - 1$$

$$= [25.042] = 25$$

$$Calculated\ Baud\ Rate = \frac{16000000}{64(25 + 1)}$$

$$= 9615$$

$$Error = \frac{Calc.\ Baud\ Rate - Desired\ Baud\ Rate}{Desired\ Baud\ Rate}$$

$$= \frac{(9615 - 9600)}{9600} = 0.16\%$$

TABLE 9-3: BAUD RATE FORMULAS

Configuration Bits		AUSART Mode	Baud Rate Formula
SYNC	BRGH		
0	0	Asynchronous	FOSC/[64 (n+1)]
0	1	Asynchronous	FOSC/[16 (n+1)]
1	x	Synchronous	FOSC/[4 (n+1)]

Legend: x = Don't care, n = value of SPBRG register

TABLE 9-4: REGISTERS ASSOCIATED WITH THE BAUD RATE GENERATOR

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
SPBRG	BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0	0000 0000	0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for the Baud Rate Generator.

TABLE 9-5: BAUD RATES FOR ASYNCHRONOUS MODES

BAUD RATE	SYNC = 0, BRGH = 0											
	Fosc = 20.000 MHz			Fosc = 18.432 MHz			Fosc = 11.0592 MHz			Fosc = 8.000 MHz		
	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	—	—	—	—	—	—	—	—	—	—	—	—
1200	1221	1.73	255	1200	0.00	239	1200	0.00	143	1202	0.16	103
2400	2404	0.16	129	2400	0.00	119	2400	0.00	71	2404	0.16	51
9600	9470	-1.36	32	9600	0.00	29	9600	0.00	17	9615	0.16	12
10417	10417	0.00	29	10286	-1.26	27	10165	-2.42	16	10417	0.00	11
19.2k	19.53k	1.73	15	19.20k	0.00	14	19.20k	0.00	8	—	—	—
57.6k	—	—	—	57.60k	0.00	7	57.60k	0.00	2	—	—	—
115.2k	—	—	—	—	—	—	—	—	—	—	—	—

BAUD RATE	SYNC = 0, BRGH = 0											
	Fosc = 4.000 MHz			Fosc = 3.6864 MHz			Fosc = 2.000 MHz			Fosc = 1.000 MHz		
	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	300	0.16	207	300	0.00	191	300	0.16	103	300	0.16	51
1200	1202	0.16	51	1200	0.00	47	1202	0.16	25	1202	0.16	12
2400	2404	0.16	25	2400	0.00	23	2404	0.16	12	—	—	—
9600	—	—	—	9600	0.00	5	—	—	—	—	—	—
10417	10417	0.00	5	—	—	—	10417	0.00	2	—	—	—
19.2k	—	—	—	19.20k	0.00	2	—	—	—	—	—	—
57.6k	—	—	—	57.60k	0.00	0	—	—	—	—	—	—
115.2k	—	—	—	—	—	—	—	—	—	—	—	—

BAUD RATE	SYNC = 0, BRGH = 1											
	Fosc = 20.000 MHz			Fosc = 18.432 MHz			Fosc = 11.0592 MHz			Fosc = 8.000 MHz		
	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	—	—	—	—	—	—	—	—	—	—	—	—
1200	—	—	—	—	—	—	—	—	—	—	—	—
2400	—	—	—	—	—	—	—	—	—	2404	0.16	207
9600	9615	0.16	129	9600	0.00	119	9600	0.00	71	9615	0.16	51
10417	10417	0.00	119	10378	-0.37	110	10473	0.53	65	10417	0.00	47
19.2k	19.23k	0.16	64	19.20k	0.00	59	19.20k	0.00	35	19231	0.16	25
57.6k	56.82k	-1.36	21	57.60k	0.00	19	57.60k	0.00	11	55556	-3.55	8
115.2k	113.64k	-1.36	10	115.2k	0.00	9	115.2k	0.00	5	—	—	—

TABLE 9-5: BAUD RATES FOR ASYNCHRONOUS MODES

BAUD RATE	SYNC = 0, BRGH = 1											
	FOSC = 4.000 MHz			FOSC = 3.6864 MHz			FOSC = 2.000 MHz			FOSC = 1.000 MHz		
	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	—	—	—	—	—	—	—	—	—	300	0.16	207
1200	1202	0.16	207	1200	0.00	191	1202	0.16	103	1202	0.16	51
2400	2404	0.16	103	2400	0.00	95	2404	0.16	51	2404	0.16	25
9600	9615	0.16	25	9600	0.00	23	9615	0.16	12	—	—	—
10417	10417	0.00	23	10473	0.53	21	10417	0.00	11	10417	0.00	5
19.2k	19.23k	0.16	12	19.2k	0.00	11	—	—	—	—	—	—
57.6k	—	—	—	57.60k	0.00	3	—	—	—	—	—	—
115.2k	—	—	—	115.2k	0.00	1	—	—	—	—	—	—

9.3 AUSART Synchronous Mode

Synchronous serial communications are typically used in systems with a single master and one or more slaves. The master device contains the necessary circuitry for baud rate generation and supplies the clock for all devices in the system. Slave devices can take advantage of the master clock by eliminating the internal clock generation circuitry.

There are two signal lines in Synchronous mode: a bidirectional data line and a clock line. Slaves use the external clock supplied by the master to shift the serial data into and out of their respective receive and transmit shift registers. Since the data line is bidirectional, synchronous operation is half-duplex only. Half-duplex refers to the fact that master and slave devices can receive and transmit data but not both simultaneously. The AUSART can operate as either a master or slave device.

Start and Stop bits are not used in synchronous transmissions.

9.3.1 SYNCHRONOUS MASTER MODE

The following bits are used to configure the AUSART for Synchronous Master operation:

- SYNC = 1
- CSRC = 1
- SREN = 0 (for transmit); SREN = 1 (for receive)
- CREN = 0 (for transmit); CREN = 1 (for receive)
- SPEN = 1

Setting the SYNC bit of the TXSTA register configures the device for synchronous operation. Setting the CSRC bit of the TXSTA register configures the device as a master. Clearing the SREN and CREN bits of the RCSTA register ensures that the device is in the Transmit mode, otherwise the device will be configured to receive. Setting the SPEN bit of the RCSTA register enables the AUSART.

The LCD SEG8 and SEG9 functions must be disabled by clearing the SE8 and SE9 bits of the LCDSE1 register, if the RX/DT and TX/CK pins are shared with the LCD peripheral.

9.3.1.1 Master Clock

Synchronous data transfers use a separate clock line, which is synchronous with the data. A device configured as a master transmits the clock on the TX/CK line. The TX/CK pin output driver is automatically enabled when the AUSART is configured for synchronous transmit or receive operation. Serial data bits change on the leading edge to ensure they are valid at the trailing edge of each clock. One clock cycle is generated for each data bit. Only as many clock cycles are generated as there are data bits.

9.3.1.2 Synchronous Master Transmission

Data is transferred out of the device on the RX/DT pin. The RX/DT and TX/CK pin output drivers are automatically enabled when the AUSART is configured for synchronous master transmit operation.

A transmission is initiated by writing a character to the TXREG register. If the TSR still contains all or part of a previous character the new character data is held in the TXREG until the last bit of the previous character has been transmitted. If this is the first character, or the previous character has been completely flushed from the TSR, the data in the TXREG is immediately transferred to the TSR. The transmission of the character commences immediately following the transfer of the data to the TSR from the TXREG.

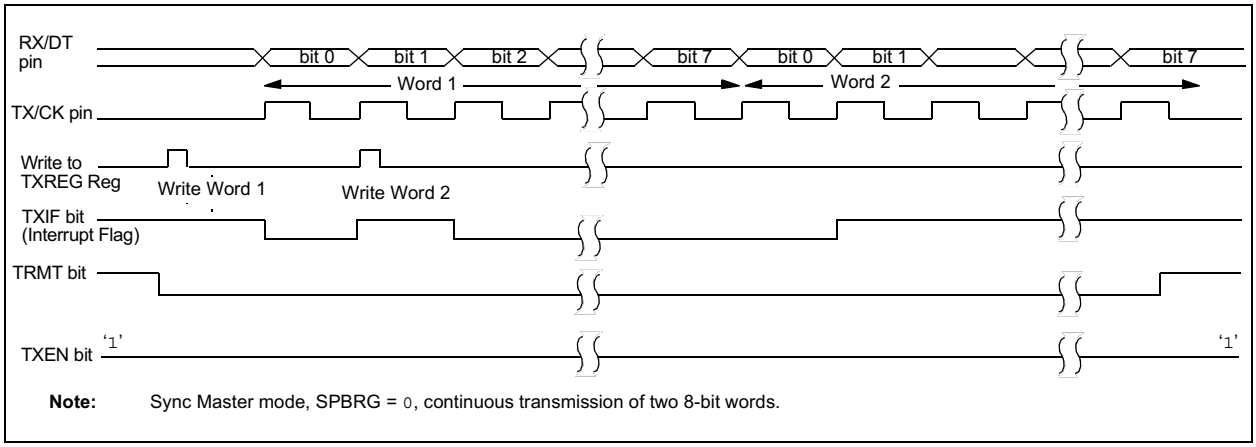
Each data bit changes on the leading edge of the master clock and remains valid until the subsequent leading clock edge.

Note:	The TSR register is not mapped in data memory, so it is not available to the user.

9.3.1.3 Synchronous Master Transmission Set-up:

1. Initialize the SPBRG register and the BRGH bit to achieve the desired baud rate (see **Section 9.2 "AUSART Baud Rate Generator (BRG)"**).
2. Enable the synchronous master serial port by setting bits SYNC, SPEN, and CSRC.
3. Disable Receive mode by clearing bits SREN and CREN.
4. Enable Transmit mode by setting the TXEN bit.
5. If 9-bit transmission is desired, set the TX9 bit.
6. If interrupts are desired, set the TXIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
7. If 9-bit transmission is selected, the ninth bit should be loaded in the TX9D bit.
8. Start transmission by loading data to the TXREG register.

FIGURE 9-6: SYNCHRONOUS TRANSMISSION



Note: Sync Master mode, SPBRG = 0, continuous transmission of two 8-bit words.

FIGURE 9-7: SYNCHRONOUS TRANSMISSION (THROUGH TXEN)

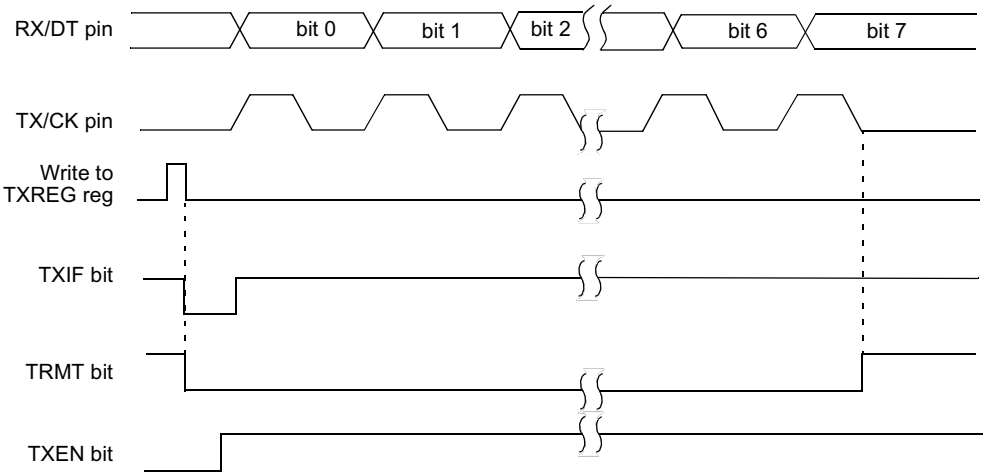


TABLE 9-6: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER TRANSMISSION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	0000 0000
PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
SPBRG	BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0	0000 0000	0000 0000
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
TXREG	AUSART Transmit Data Register								0000 0000	0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for Synchronous Master Transmission.

9.3.1.4 Synchronous Master Reception

Data is received at the RX/DT pin. The RX/DT pin output driver is automatically disabled when the AUSART is configured for synchronous master receive operation.

In Synchronous mode, reception is enabled by setting either the Single Receive Enable bit (SREN of the RCSTA register) or the Continuous Receive Enable bit (CREN of the RCSTA register).

When SREN is set and CREN is clear, only as many clock cycles are generated as there are data bits in a single character. The SREN bit is automatically cleared at the completion of one character. When CREN is set, clocks are continuously generated until CREN is cleared. If CREN is cleared in the middle of a character the CK clock stops immediately and the partial character is discarded. If SREN and CREN are both set, then SREN is cleared at the completion of the first character and CREN takes precedence.

To initiate reception, set either SREN or CREN. Data is sampled at the RX/DT pin on the trailing edge of the TX/CK clock pin and is shifted into the Receive Shift Register (RSR). When a complete character is received into the RSR, the RCIF bit of the PIR1 register is set and the character is automatically transferred to the two character receive FIFO. The Least Significant eight bits of the top character in the receive FIFO are available in RCREG. The RCIF bit remains set as long as there are un-read characters in the receive FIFO.

9.3.1.5 Slave Clock

Synchronous data transfers use a separate clock line, which is synchronous with the data. A device configured as a slave receives the clock on the TX/CK line. The TX/CK pin output driver is automatically disabled when the device is configured for synchronous slave transmit or receive operation. Serial data bits change on the leading edge to ensure they are valid at the trailing edge of each clock. One data bit is transferred for each clock cycle. Only as many clock cycles should be received as there are data bits.

9.3.1.6 Receive Overrun Error

The receive FIFO buffer can hold two characters. An overrun error will be generated if a third character, in its entirety, is received before RCREG is read to access the FIFO. When this happens the OERR bit of the RCSTA register is set. Previous data in the FIFO will not be overwritten. The two characters in the FIFO buffer can be read, however, no additional characters will be received until the error is cleared. The OERR bit can only be cleared by clearing the overrun condition. If the overrun error occurred when the SREN bit is set and CREN is clear then the error is cleared by reading RCREG. If the overrun occurred when the CREN bit is set then the error condition is cleared by either clearing the CREN bit of the RCSTA register.

9.3.1.7 Receiving 9-bit Characters

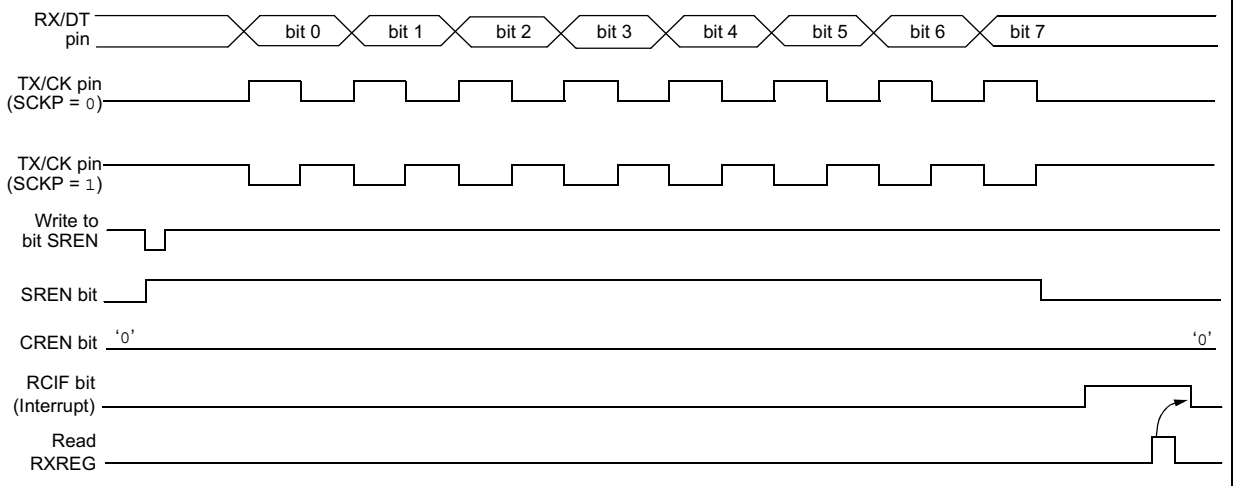
The AUSART supports 9-bit character reception. When the RX9 bit of the RCSTA register is set the AUSART will shift 9-bits into the RSR for each character received. The RX9D bit of the RCSTA register is the ninth, and Most Significant, data bit of the top unread character in the receive FIFO. When reading 9-bit data from the receive FIFO buffer, the RX9D data bit must be read before reading the 8 Least Significant bits from the RCREG.

Address detection in Synchronous modes is not supported, therefore the ADDEN bit of the RCSTA register must be cleared.

9.3.1.8 Synchronous Master Reception Set-up:

1. Initialize the SPBRG register for the appropriate baud rate. Set or clear the BRGH bit, as required, to achieve the desired baud rate.
2. Enable the synchronous master serial port by setting bits SYNC, SPEN and CSRC.
3. Ensure bits CREN and SREN are clear.
4. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
5. If 9-bit reception is desired, set bit RX9.
6. Verify address detection is disabled by clearing the ADDEN bit of the RCSTA register.
7. Start reception by setting the SREN bit or for continuous reception, set the CREN bit.
8. Interrupt flag bit RCIF of the PIR1 register will be set when reception of a character is complete. An interrupt will be generated if the RCIE interrupt enable bit of the PIE1 register was set.
9. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
10. Read the 8-bit received data by reading the RCREG register.
11. If an overrun error occurs, clear the error by either clearing the CREN bit of the RCSTA register or by clearing the SPEN bit which resets the AUSART.

FIGURE 9-8: SYNCHRONOUS RECEPTION (MASTER MODE, SREN)



Note: Timing diagram demonstrates Sync Master mode with bit SREN = 1 and bit BRGH = 0.

TABLE 9-7: REGISTERS ASSOCIATED WITH SYNCHRONOUS MASTER RECEPTION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	0000 0000
PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCREG	AUSART Receive Data Register								0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000X	0000 000X
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for Synchronous Master Reception.

9.3.2 SYNCHRONOUS SLAVE MODE

The following bits are used to configure the AUSART for Synchronous slave operation:

- SYNC = 1
- CSRC = 0
- SREN = 0 (for transmit); SREN = 1 (for receive)
- CREN = 0 (for transmit); CREN = 1 (for receive)
- SPEN = 1

Setting the SYNC bit of the TXSTA register configures the device for synchronous operation. Clearing the CSRC bit of the TXSTA register configures the device as a slave. Clearing the SREN and CREN bits of the RCSTA register ensures that the device is in the Transmit mode, otherwise the device will be configured to receive. Setting the SPEN bit of the RCSTA register enables the AUSART.

The LCD SEG8 and SEG9 functions must be disabled by clearing the SE8 and SE9 bits of the LCDSE1 register, if the RX/DT and TX/CK pins are shared with the LCD peripheral.

9.3.2.1 AUSART Synchronous Slave Transmit

The operation of the Synchronous Master and Slave modes are identical (see **Section 9.3.1.2 "Synchronous Master Transmission")**, except in the case of the Sleep mode.

If two words are written to the TXREG and then the SLEEP instruction is executed, the following will occur:

1. The first character will immediately transfer to the TSR register and transmit.
2. The second word will remain in TXREG register.
3. The TXIF bit will not be set.
4. After the first character has been shifted out of TSR, the TXREG register will transfer the second character to the TSR and the TXIF bit will now be set.
5. If the PEIE and TXIE bits are set, the interrupt will wake the device from Sleep and execute the next instruction. If the GIE bit is also set, the program will call the Interrupt Service Routine.

9.3.2.2 Synchronous Slave Transmission Set-up:

1. Set the SYNC and SPEN bits and clear the CSRC bit.
2. Clear the CREN and SREN bits.
3. If using interrupts, ensure that the GIE and PEIE bits of the INTCON register are set and set the TXIE bit.
4. If 9-bit transmission is desired, set the TX9 bit.
5. Enable transmission by setting the TXEN bit.
6. Verify address detection is disabled by clearing the ADDEN bit of the RCSTA register.
7. If 9-bit transmission is selected, insert the Most Significant bit into the TX9D bit.
8. Start transmission by writing the Least Significant 8 bits to the TXREG register.

TABLE 9-8: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE TRANSMISSION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	0000 0000
PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000X	0000 000X
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
TXREG	AUSART Transmit Data Register								0000 0000	0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for Synchronous Slave Transmission.

9.3.2.3 AUSART Synchronous Slave Reception

The operation of the Synchronous Master and Slave modes is identical (**Section 9.3.1.4 "Synchronous Master Reception"**), with the following exceptions:

- Sleep
- CREN bit is always set, therefore the receiver is never Idle
- SREN bit, which is a "don't care" in Slave mode

A character may be received while in Sleep mode by setting the CREN bit prior to entering Sleep. Once the word is received, the RSR register will transfer the data to the RCREG register. If the RCIE interrupt enable bit of the PIE1 register is set, the interrupt generated will wake the device from Sleep and execute the next instruction. If the GIE bit is also set, the program will branch to the interrupt vector.

9.3.2.4 Synchronous Slave Reception Set-up:

1. Set the SYNC and SPEN bits and clear the CSRC bit.
2. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
3. If 9-bit reception is desired, set the RX9 bit.
4. Verify address detection is disabled by clearing the ADDEN bit of the RCSTA register.
5. Set the CREN bit to enable reception.
6. The RCIF bit of the PIR1 register will be set when reception is complete. An interrupt will be generated if the RCIE bit of the PIE1 register was set.
7. If 9-bit mode is enabled, retrieve the Most Significant bit from the RX9D bit of the RCSTA register.
8. Retrieve the 8 Least Significant bits from the receive FIFO by reading the RCREG register.
9. If an overrun error occurs, clear the error by either clearing the CREN bit of the RCSTA register.

TABLE 9-9: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE RECEPTION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	0000 0000
PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCREG	AUSART Receive Data Register								0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000X	0000 000X
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used for Synchronous Slave Reception.

9.4 AUSART Operation During Sleep

The AUSART will remain active during Sleep only in the Synchronous Slave mode. All other modes require the system clock and therefore cannot generate the necessary signals to run the Transmit or Receive Shift registers during Sleep.

Synchronous Slave mode uses an externally generated clock to run the Transmit and Receive Shift registers.

9.4.1 SYNCHRONOUS RECEIVE DURING SLEEP

To receive during Sleep, all the following conditions must be met before entering Sleep mode:

- RCSTA and TXSTA Control registers must be configured for Synchronous Slave Reception (see **Section 9.3.2.4 "Synchronous Slave Reception Set-up:"**).
- If interrupts are desired, set the RCIE bit of the PIE1 register and the PEIE bit of the INTCON register.
- The RCIF interrupt flag must be cleared by reading RCREG to unload any pending characters in the receive buffer.

Upon entering Sleep mode, the device will be ready to accept data and clocks on the RX/DT and TX/CK pins, respectively. When the data word has been completely clocked in by the external device, the RCIF interrupt flag bit of the PIR1 register will be set. Thereby, waking the processor from Sleep.

Upon waking from Sleep, the instruction following the SLEEP instruction will be executed. If the GIE global interrupt enable bit of the INTCON register is also set, then the Interrupt Service Routine at address 004h will be called.

9.4.2 SYNCHRONOUS TRANSMIT DURING SLEEP

To transmit during Sleep, all the following conditions must be met before entering Sleep mode:

- RCSTA and TXSTA Control registers must be configured for Synchronous Slave Transmission (see **Section 9.3.2.2 "Synchronous Slave Transmission Set-up:"**).
- The TXIF interrupt flag must be cleared by writing the output data to the TXREG, thereby filling the TSR and transmit buffer.
- If interrupts are desired, set the TXIE bit of the PIE1 register and the PEIE bit of the INTCON register.

Upon entering Sleep mode, the device will be ready to accept clocks on TX/CK pin and transmit data on the RX/DT pin. When the data word in the TSR has been completely clocked out by the external device, the pending byte in the TXREG will transfer to the TSR and the TXIF flag will be set. Thereby, waking the processor from Sleep. At this point, the TXREG is available to accept another character for transmission, which will clear the TXIF flag.

Upon waking from Sleep, the instruction following the SLEEP instruction will be executed. If the GIE global interrupt enable bit is also set then the Interrupt Service Routine at address 0004h will be called.

NOTES:

10.0 LIQUID CRYSTAL DISPLAY (LCD) DRIVER MODULE

The Liquid Crystal Display (LCD) driver module generates the timing control to drive a static or multiplexed LCD panel. In the PIC16F913/916 devices, the module drives the panels of up to four commons and up to 16 segments. In the PIC16F914/917 devices, the module drives the panels of up to four commons and up to 24 segments. In the PIC16F946 device, the module drives the panels of up to four commons and up to 42 segments. The LCD module also provides control of the LCD pixel data.

The LCD driver module supports:

- Direct driving of LCD panel
- Three LCD clock sources with selectable prescaler
- Up to four commons:
  - Static (1 common)
  - 1/2 multiplex (2 commons)
  - 1/3 multiplex (3 commons)
  - 1/4 multiplex (4 commons)
- Segments up to:
  - 16 (PIC16F913/916)
  - 24 (PIC16F914/917)
  - 42 (PIC16F946)
- Static, 1/2 or 1/3 LCD Bias

Note:	COM3 and SEG15 share the same physical pin on the PIC16F913/916, therefore SEG15 is not available when using 1/4 multiplex displays.

10.1 LCD Registers

The module contains the following registers:

- LCD Control Register (LCDCON)
- LCD Phase Register (LCDPS)
- Up to 6 LCD Segment Enable Registers (LCDSEn)
- Up to 24 LCD Data Registers (LCDDATA)

TABLE 10-1: LCD SEGMENT AND DATA REGISTERS

Device	# of LCD Registers	
	Segment Enable	Data
PIC16F913/916	2	8
PIC16F914/917	3	12
PIC16F946	6	24

The LCDCON register (Register 10-1) controls the operation of the LCD driver module. The LCDPS register (Register 10-2) configures the LCD clock source prescaler and the type of waveform; Type-A or Type-B. The LCDSE registers (Register 10-3) configure the functions of the port pins.

The following LCDSE registers are available:

- LCDSE0 SE<7:0>
- LCDSE1 SE<15:8>
- LCDSE2 SE<23:16>**(1)**
- LCDSE3 SE<31:24>**(2)**
- LCDSE4 SE<39:32>**(2)**
- LCDSE5 SE<41:40>**(2)**

Note 1:	PIC16F914/917 and PIC16F946 only.
2:	PIC16F946 only.

Once the module is initialized for the LCD panel, the individual bits of the LCDDATA<11:0> registers are cleared/set to represent a clear/dark pixel, respectively:

- LCDDATA0 SEG<7:0>COM0
- LCDDATA1 SEG<15:8>COM0
- LCDDATA2 SEG<23:16>COM0
- LCDDATA3 SEG<7:0>COM1
- LCDDATA4 SEG<15:8>COM1
- LCDDATA5 SEG<23:16>COM1
- LCDDATA6 SEG<7:0>COM2
- LCDDATA7 SEG<15:8>COM2
- LCDDATA8 SEG<23:16>COM2
- LCDDATA9 SEG<7:0>COM3
- LCDDATA10 SEG<15:8>COM3
- LCDDATA11 SEG<23:16>COM3

The following additional registers are available on the PIC16F946 only:

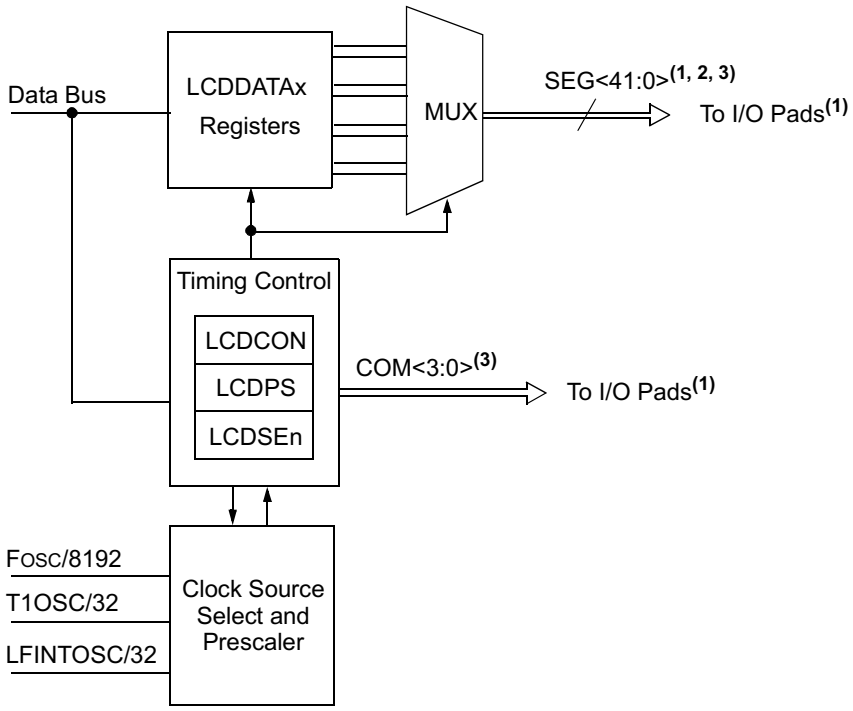
- LCDDATA12 SEG<31:24>COM0
- LCDDATA13 SEG<39:32>COM0
- LCDDATA14 SEG<41:40>COM0
- LCDDATA15 SEG<31:24>COM1
- LCDDATA16 SEG<39:32>COM1
- LCDDATA17 SEG<41:40>COM1
- LCDDATA18 SEG<31:24>COM2
- LCDDATA19 SEG<39:32>COM2
- LCDDATA20 SEG<41:40>COM2
- LCDDATA21 SEG<31:24>COM3
- LCDDATA22 SEG<39:32>COM3
- LCDDATA23 SEG<41:40>COM3

As an example, LCDDATAx is detailed in Register 10-4.

Once the module is configured, the LCDEN bit of the LCDCON register is used to enable or disable the LCD module. The LCD panel can also operate during Sleep by clearing the $\overline{\text{SLPEN}}$ bit of the LCDCON register.

Note:	The LCDDATA2, LCDDATA5, LCDDATA8 and LCDDATA11 registers are not implemented in the PIC16F913/916 devices.

FIGURE 10-1: LCD DRIVER MODULE BLOCK DIAGRAM



Note 1: These are not directly connected to the I/O pads, but may be tri-stated, depending on the configuration of the LCD module.

2: SEG<23:0> on PIC16F914/917, SEG<15:0> on PIC16F913/916.

3: COM3 and SEG15 share the same physical pin on the PIC16F913/916, therefore SEG15 is not available when using 1/4 multiplex displays.

REGISTER 10-1: LCDCON: LIQUID CRYSTAL DISPLAY CONTROL REGISTER

R/W-0	R/W-0	R/C-0	R/W-1	R/W-0	R/W-0	R/W-1	R/W-1
LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
C = Only clearable bit	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown
-n = Value at POR			

bit 7 **LCDEN:** LCD Driver Enable bit
1 = LCD driver module is enabled
0 = LCD driver module is disabled

bit 6 **$\overline{\text{SLPEN}}$:** LCD Driver Enable in Sleep mode bit
1 = LCD driver module is disabled in Sleep mode
0 = LCD driver module is enabled in Sleep mode

bit 5 **WERR:** LCD Write Failed Error bit
1 = LCDDATAx register written while the WA bit of the LCDPS register = 0 (must be cleared in software)
0 = No LCD write error

bit 4 **VLCDEN:** LCD Bias Voltage Pins Enable bit
1 = VLCD pins are enabled
0 = VLCD pins are disabled

bit 3-2 **CS<1:0>:** Clock Source Select bits
00 = FOSC/8192
01 = T1OSC (Timer1)/32
1x = LFINTOSC (31 kHz)/32

bit 1-0 **LMUX<1:0>:** Commons Select bits

LMUX<1:0>	Multiplex	Maximum Number of Pixels			Bias
		PIC16F913/916	PIC16F914/917	PIC16F946	
00	Static (COM0)	16	24	42	Static
01	1/2 (COM<1:0>)	32	48	84	1/2 or 1/3
10	1/3 (COM<2:0>)	48	72	126	1/2 or 1/3
11	1/4 (COM<3:0>)	60(1)	96	168	1/3

Note 1: On PIC16F913/916 devices, COM3 and SEG15 are shared on one pin, limiting the device from driving 64 pixels.

REGISTER 10-2: LCDPS: LCD PRESCALER SELECT REGISTER

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
WFT	BIASMD	LCDA	WA	LP3	LP2	LP1	LP0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **WFT:** Waveform Type Select bit
1 = Type-B waveform (phase changes on each frame boundary)
0 = Type-A waveform (phase changes within each common interval)

bit 6 **BIASMD:** Bias Mode Select bit
When LMUX<1:0> = 00:
0 = Static Bias mode (do not set this bit to '1')
When LMUX<1:0> = 01:
1 = 1/2 Bias mode
0 = 1/3 Bias mode
When LMUX<1:0> = 10:
1 = 1/2 Bias mode
0 = 1/3 Bias mode
When LMUX<1:0> = 11:
0 = 1/3 Bias mode (do not set this bit to '1')

bit 5 **LCDA:** LCD Active Status bit
1 = LCD driver module is active
0 = LCD driver module is inactive

bit 4 **WA:** LCD Write Allow Status bit
1 = Write into the LCDDATAx registers is allowed
0 = Write into the LCDDATAx registers is not allowed

bit 3-0 **LP<3:0>:** LCD Prescaler Select bits
1111 = 1:16
1110 = 1:15
1101 = 1:14
1100 = 1:13
1011 = 1:12
1010 = 1:11
1001 = 1:10
1000 = 1:9
0111 = 1:8
0110 = 1:7
0101 = 1:6
0100 = 1:5
0011 = 1:4
0010 = 1:3
0001 = 1:2
0000 = 1:1

REGISTER 10-3: LCDSEn: LCD SEGMENT ENABLE REGISTERS

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SEn	SEn	SEn	SEn	SEn	SEn	SEn	SEn
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 7-0 **SEn:** Segment Enable bits
1 = Segment function of the pin is enabled
0 = I/O function of the pin is enabled

REGISTER 10-4: LCDDATAx: LCD DATA REGISTERS

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SEGx-COMy	SEGx-COMy	SEGx-COMy	SEGx-COMy	SEGx-COMy	SEGx-COMy	SEGx-COMy	SEGx-COMy
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’	
-n = Value at POR	‘1’ = Bit is set	‘0’ = Bit is cleared	x = Bit is unknown

bit 7-0 **SEGx-COMy:** Pixel On bits
1 = Pixel on (dark)
0 = Pixel off (clear)

10.2 LCD Clock Source Selection

The LCD driver module has 3 possible clock sources:

- FOSC/8192
- T1OSC/32
- LFINTOSC/32

The first clock source is the system clock divided by 8192 (FOSC/8192). This divider ratio is chosen to provide about 1 kHz output when the system clock is 8 MHz. The divider is not programmable. Instead, the LCD prescaler bits LP<3:0> of the LCDPS register are used to set the LCD frame clock rate.

The second clock source is the T1OSC/32. This also gives about 1 kHz when a 32.768 kHz crystal is used with the Timer1 oscillator. To use the Timer1 oscillator as a clock source, the T1OSCEN bit of the T1CON register should be set.

The third clock source is the 31 kHz LFINTOSC/32, which provides approximately 1 kHz output.

The second and third clock sources may be used to continue running the LCD while the processor is in Sleep.

Using bits CS<1:0> of the LCDCON register can select any of these clock sources.

10.2.1 LCD PRESCALER

A 4-bit counter is available as a prescaler for the LCD clock. The prescaler is not directly readable or writable; its value is set by the LP<3:0> bits of the LCDPS register, which determine the prescaler assignment and prescale ratio.

The prescale values are selectable from 1:1 through 1:16.

10.3 LCD Bias Types

The LCD driver module can be configured into one of three bias types:

- Static Bias (2 voltage levels: VSS and VDD)
- 1/2 Bias (3 voltage levels: VSS, 1/2 VDD and VDD)
- 1/3 Bias (4 voltage levels: VSS, 1/3 VDD, 2/3 VDD and VDD)

This module uses an external resistor ladder to generate the LCD bias voltages.

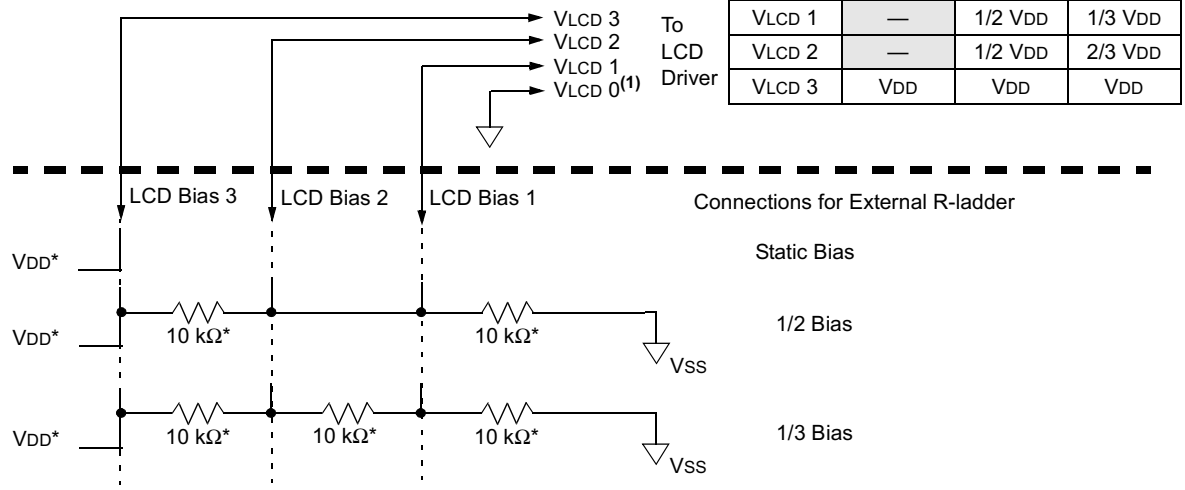
The external resistor ladder should be connected to the VLCD1 pin (Bias 1), VLCD2 pin (Bias 2), VLCD3 pin (Bias 3) and VSS. The VLCD3 pin should also be connected to VDD.

Figure 10-2 shows the proper way to connect the resistor ladder to the Bias pins..

Note:	VLCD pins used to supply LCD bias voltage are enabled on power-up (POR) and must be disabled by the user by clearing the VLCDEN bit of the LCDCON register.

FIGURE 10-2: LCD BIAS RESISTOR LADDER CONNECTION DIAGRAM

	Static Bias	1/2 Bias	1/3 Bias
VLCD 0	VSS	VSS	VSS
VLCD 1	—	1/2 VDD	1/3 VDD
VLCD 2	—	1/2 VDD	2/3 VDD
VLCD 3	VDD	VDD	VDD



* These values are provided for design guidance only and should be optimized for the application by the designer.

Note 1: Internal connection.

10.4 LCD Multiplex Types

The LCD driver module can be configured into one of four multiplex types:

- Static (only COM0 is used)
- 1/2 multiplex (COM<1:0> are used)
- 1/3 multiplex (COM<2:0> are used)
- 1/4 multiplex (COM<3:0> are used)

The LMUX<1:0> bit setting of the LCDCON register decides the function of RB5, RA2 or either RA3 or RD0 pins (see Table 10-2 for details).

If the pin is a digital I/O, the corresponding TRIS bit controls the data direction. If the pin is a COM drive, then the TRIS setting of that pin is overridden.

Note:	On a Power-on Reset, the LMUX<1:0> bits of the LCDCON register are '11'.

TABLE 10-2: RA3/RD0, RA2, RB5 FUNCTION

Multiplex	LMUX <1:0>	RA3/RD0(1)	RA2	RB5
Static	00	Digital I/O	Digital I/O	Digital I/O
1/2	01	Digital I/O	Digital I/O	COM1 Driver
1/3	10	Digital I/O	COM2 Driver	COM1 Driver
1/4	11	COM3 Driver	COM2 Driver	COM1 Driver

Note 1: RA3 for PIC16F913/916, RD0 for PIC16F914/917 and PIC16F946

10.5 Segment Enables

The LCDSEn registers are used to select the pin function for each segment pin. The selection allows each pin to operate as either an LCD segment driver or as one of the pin's alternate functions. To configure the pin as a segment pin, the corresponding bits in the LCDSEn registers must be set to '1'.

If the pin is a digital I/O, the corresponding TRIS bit controls the data direction. Any bit set in the LCDSEn registers overrides any bit settings in the corresponding TRIS register.

Note:	On a Power-on Reset, these pins are configured as digital I/O.

10.6 Pixel Control

The LCDDATAx registers contain bits which define the state of each pixel. Each bit defines one unique pixel.

Register 10-4 shows the correlation of each bit in the LCDDATAx registers to the respective common and segment signals.

Any LCD pixel location not being used for display can be used as general purpose RAM.

10.7 LCD Frame Frequency

The rate at which the COM and SEG outputs change is called the LCD frame frequency.

TABLE 10-3: FRAME FREQUENCY FORMULAS

Multiplex	Frame Frequency =
Static	Clock source/(4 x 1 x (LP<3:0> + 1))
1/2	Clock source/(2 x 2 x (LP<3:0> + 1))
1/3	Clock source/(1 x 3 x (LP<3:0> + 1))
1/4	Clock source/(1 x 4 x (LP<3:0> + 1))

Note: Clock source is FOSC/8192, T1OSC/32 or LFINTOSC/32.

TABLE 10-4: APPROXIMATE FRAME FREQUENCY (IN Hz) USING Fosc @ 8 MHz, TIMER1 @ 32.768 kHz OR LFINTOSC

LP<3:0>	Static	1/2	1/3	1/4
2	85	85	114	85
3	64	64	85	64
4	51	51	68	51
5	43	43	57	43
6	37	37	49	37
7	32	32	43	32

FIGURE 10-3: LCD CLOCK GENERATION

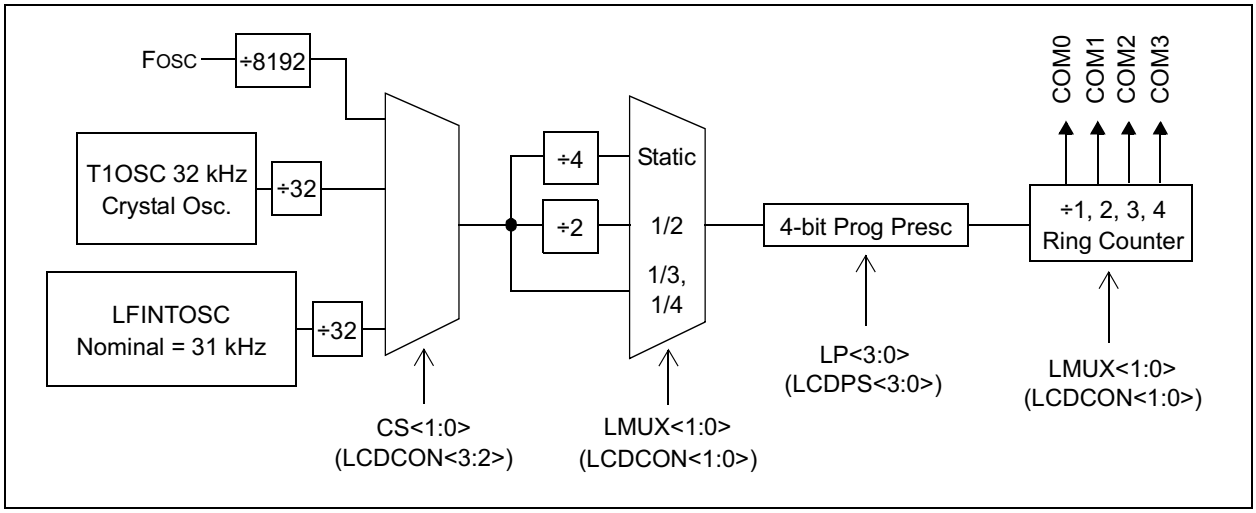


FIGURE 10-4: LCD SEGMENT MAPPING WORKSHEET (SHEET 1 OF 2)

LCD Function	COM0		COM1		COM2		COM3		Pin No.			PORT	Alternate Functions
	LCDDATAx Address	LCD Segment	LCDDATAx Address	LCD Segment	LCDDATAx Address	LCD Segment	LCDDATAx Address	LCD Segment	28-pin	40-pin	64-pin		
SEG0	LCDDATA0, 0		LCDDATA3, 0		LCDDATA6, 0		LCDDATA9, 0		21	33	15	RB0	INT
SEG1	LCDDATA0, 1		LCDDATA3, 1		LCDDATA6, 1		LCDDATA9, 1		22	34	16	RB1	
SEG2	LCDDATA0, 2		LCDDATA3, 2		LCDDATA6, 2		LCDDATA9, 2		23	35	17	RB2	
SEG3	LCDDATA0, 3		LCDDATA3, 3		LCDDATA6, 3		LCDDATA9, 3		24	36	18	RB3	
SEG4	LCDDATA0, 4		LCDDATA3, 4		LCDDATA6, 4		LCDDATA9, 4		6	6	31	RA4	C1OUT/T0CKI
SEG5	LCDDATA0, 5		LCDDATA3, 5		LCDDATA6, 5		LCDDATA9, 5		7	7	32	RA5	C2OUT/AN4/$\overline{SS}$
SEG6	LCDDATA0, 6		LCDDATA3, 6		LCDDATA6, 6		LCDDATA9, 6		14	18	52	RC3	
SEG7	LCDDATA0, 7		LCDDATA3, 7		LCDDATA6, 7		LCDDATA9, 7		3	3	28	RA1	AN1/C2-
SEG8	LCDDATA1, 0		LCDDATA4, 0		LCDDATA7, 0		LCDDATA10, 0		18	26	62	RC7	RX/DT/SDI/SDA
SEG9	LCDDATA1, 1		LCDDATA4, 1		LCDDATA7, 1		LCDDATA10, 1		17	25	61	RC6	TX/CK/SCK/SCL
SEG10	LCDDATA1, 2		LCDDATA4, 2		LCDDATA7, 2		LCDDATA10, 2		16	24	60	RC5	T1CKI/CCP1
SEG11	LCDDATA1, 3		LCDDATA4, 3		LCDDATA7, 3		LCDDATA10, 3		15	23	59	RC4	$\overline{T1G}$/SDO
SEG12	LCDDATA1, 4		LCDDATA4, 4		LCDDATA7, 4		LCDDATA10, 4		2	2	27	RA0	AN0/C1-
SEG13	LCDDATA1, 5		LCDDATA4, 5		LCDDATA7, 5		LCDDATA10, 5		28	40	24	RB7	ICSPDAT/ICDDAT
SEG14	LCDDATA1, 6		LCDDATA4, 6		LCDDATA7, 6		LCDDATA10, 6		27	39	23	RB6	ICSPCLK/ICDCK
SEG15	LCDDATA1, 7		LCDDATA4, 7		LCDDATA7, 7		LCDDATA10, 7		5	5	30	RA3	AN3/V_{REF}+/COM3*
SEG16	LCDDATA2, 0		LCDDATA5, 0		LCDDATA8, 0		LCDDATA11, 0		—	26	58	RD3	
SEG17	LCDDATA2, 1		LCDDATA5, 1		LCDDATA8, 1		LCDDATA11, 1		—	27	63	RD4	
SEG18	LCDDATA2, 2		LCDDATA5, 2		LCDDATA8, 2		LCDDATA11, 2		—	28	64	RD5	
SEG19	LCDDATA2, 3		LCDDATA5, 3		LCDDATA8, 3		LCDDATA11, 3		—	29	1	RD6	
SEG20	LCDDATA2, 4		LCDDATA5, 4		LCDDATA8, 4		LCDDATA11, 4		—	30	2	RD7	
SEG21	LCDDATA2, 5		LCDDATA5, 5		LCDDATA8, 5		LCDDATA11, 5		—	8	33	RE0	AN5
SEG22	LCDDATA2, 6		LCDDATA5, 6		LCDDATA8, 6		LCDDATA11, 6		—	9	34	RE1	AN6
SEG23	LCDDATA2, 7		LCDDATA5, 7		LCDDATA8, 7		LCDDATA11, 7		—	10	35	RE2	AN7

☐ PIC16F914/917 and PIC16F946 only.
* = PIC16F913/916 only.

FIGURE 10-5: LCD SEGMENT MAPPING WORKSHEET (SHEET 2 OF 2)

LCD Function	COM0		COM1		COM2		COM3		Pin No.	PORT	Alternate Functions
	LCDDATAx Address	LCD Segment	LCDDATAx Address	LCD Segment	LCDDATAx Address	LCD Segment	LCDDATAx Address	LCD Segment	64-pin		
SEG24	LCDDATA12, 0		LCDDATA15, 0		LCDDATA18, 0		LCDDATA21, 0		37	RE4	
SEG25	LCDDATA12, 1		LCDDATA15, 1		LCDDATA18, 1		LCDDATA21, 1		42	RE5	
SEG26	LCDDATA12, 2		LCDDATA15, 2		LCDDATA18, 2		LCDDATA21, 2		43	RE6	
SEG27	LCDDATA12, 3		LCDDATA15, 3		LCDDATA18, 3		LCDDATA21, 3		44	RE7	
SEG28	LCDDATA12, 4		LCDDATA15, 4		LCDDATA18, 4		LCDDATA21, 4		45	RF4	
SEG29	LCDDATA12, 5		LCDDATA15, 5		LCDDATA18, 5		LCDDATA21, 5		46	RF5	
SEG30	LCDDATA12, 6		LCDDATA15, 6		LCDDATA18, 6		LCDDATA21, 6		47	RF6	
SEG31	LCDDATA12, 7		LCDDATA15, 7		LCDDATA18, 7		LCDDATA21, 7		48	RF7	
SEG32	LCDDATA13, 0		LCDDATA16, 0		LCDDATA19, 0		LCDDATA22, 0		11	RF0	
SEG33	LCDDATA13, 1		LCDDATA16, 1		LCDDATA19, 1		LCDDATA22, 1		12	RF1	
SEG34	LCDDATA13, 2		LCDDATA16, 2		LCDDATA19, 2		LCDDATA22, 2		13	RF2	
SEG35	LCDDATA13, 3		LCDDATA16, 3		LCDDATA19, 3		LCDDATA22, 3		14	RF3	
SEG36	LCDDATA13, 4		LCDDATA16, 4		LCDDATA19, 4		LCDDATA22, 4		3	RG0	
SEG37	LCDDATA13, 5		LCDDATA16, 5		LCDDATA19, 5		LCDDATA22, 5		4	RG1	
SEG38	LCDDATA13, 6		LCDDATA16, 6		LCDDATA19, 6		LCDDATA22, 6		5	RG2	
SEG39	LCDDATA13, 7		LCDDATA16, 7		LCDDATA19, 7		LCDDATA22, 7		6	RG3	
SEG40	LCDDATA14, 0		LCDDATA17, 0		LCDDATA20, 0		LCDDATA23, 0		7	RG4	
SEG41	LCDDATA14, 1		LCDDATA17, 1		LCDDATA20, 1		LCDDATA23, 1		8	RG5	

PIC16F946 only.

10.8 LCD Waveform Generation

LCD waveforms are generated so that the net AC voltage across the dark pixel should be maximized and the net AC voltage across the clear pixel should be minimized. The net DC voltage across any pixel should be zero.

The COM signal represents the time slice for each common, while the SEG contains the pixel data.

The pixel signal (COM-SEG) will have no DC component and it can take only one of the two rms values. The higher rms value will create a dark pixel and a lower rms value will create a clear pixel.

As the number of commons increases, the delta between the two rms values decreases. The delta represents the maximum contrast that the display can have.

The LCDs can be driven by two types of waveform: Type-A and Type-B. In Type-A waveform, the phase changes within each common type, whereas in Type-B waveform, the phase changes on each frame boundary. Thus, Type-A waveform maintains 0 VDC over a single frame, whereas Type-B waveform takes two frames.

Note 1:	If Sleep has to be executed with LCD Sleep disabled (LCDCON<SLPEN> is '1'), then care must be taken to execute Sleep only when VDC on all the pixels is '0'.
2:	When the LCD clock source is FOSC/8192, if Sleep is executed, irrespective of the LCDCON<SLPEN> setting, the LCD goes into Sleep. Thus, take care to see that VDC on all pixels is '0' when Sleep is executed.

Figure 10-6 through Figure 10-16 provide waveforms for static, half-multiplex, one-third-multiplex and quarter-multiplex drives for Type-A and Type-B waveforms.

FIGURE 10-6: TYPE-A/TYPE-B WAVEFORMS IN STATIC DRIVE

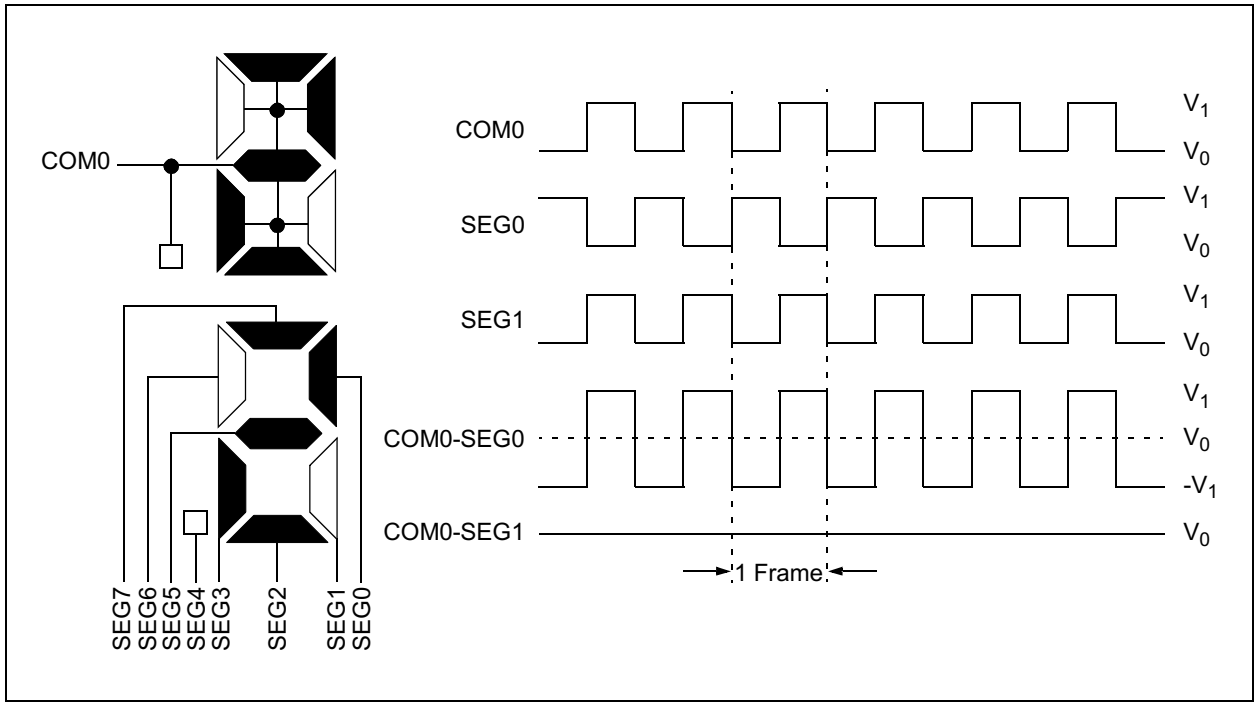


FIGURE 10-7: TYPE-A WAVEFORMS IN 1/2 MUX, 1/2 BIAS DRIVE

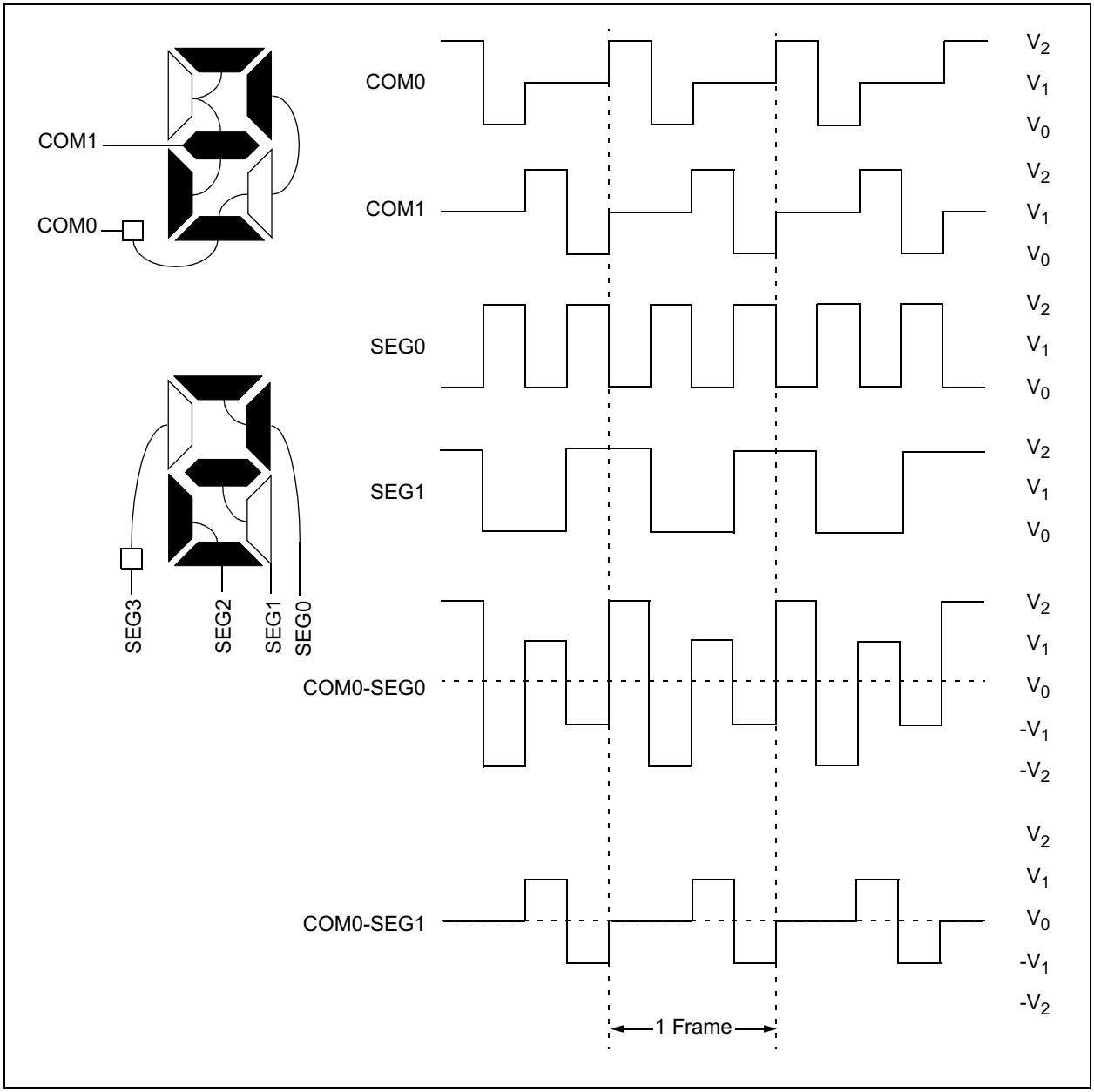


FIGURE 10-8: TYPE-B WAVEFORMS IN 1/2 MUX, 1/2 BIAS DRIVE

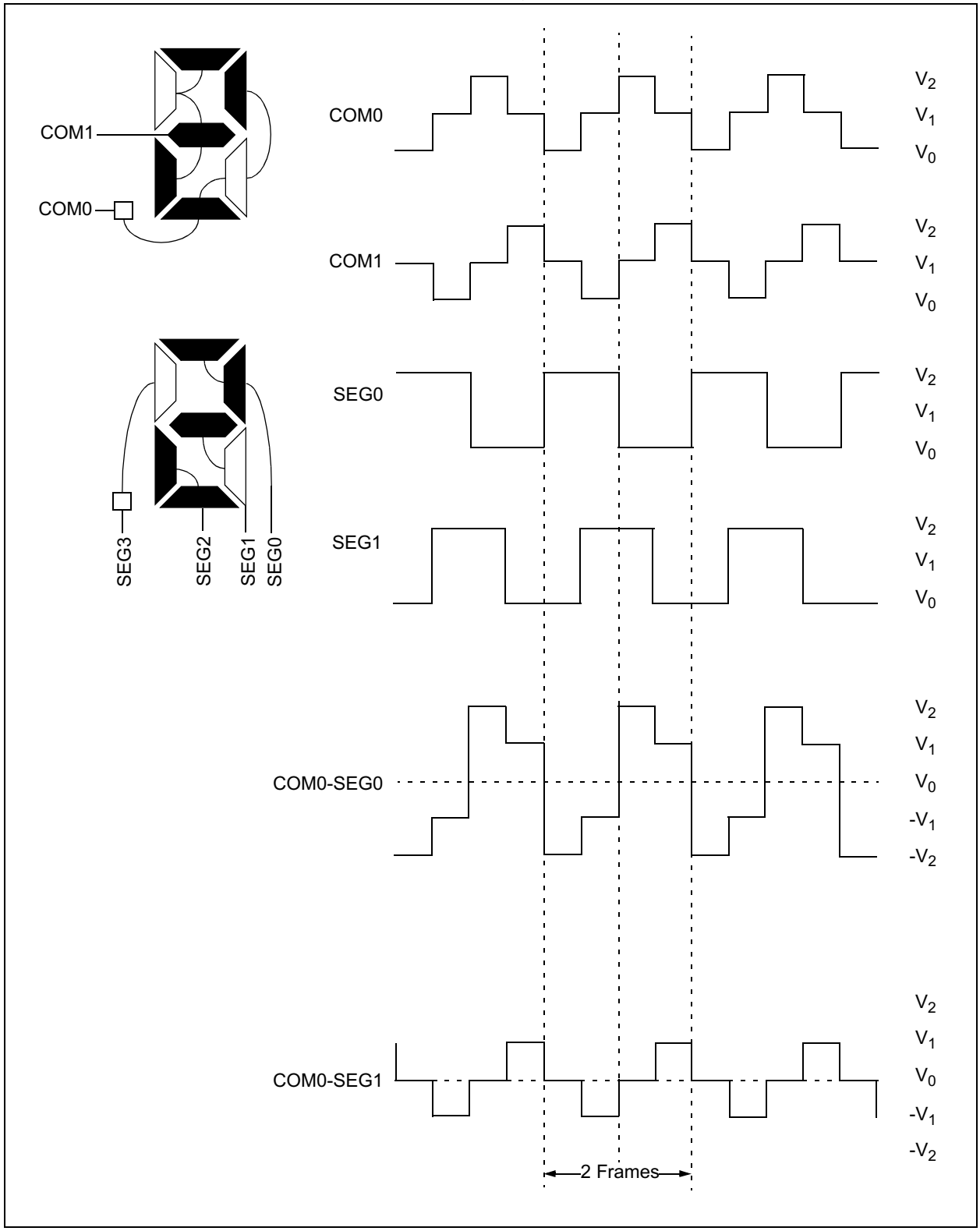


FIGURE 10-9: TYPE-A WAVEFORMS IN 1/2 MUX, 1/3 BIAS DRIVE

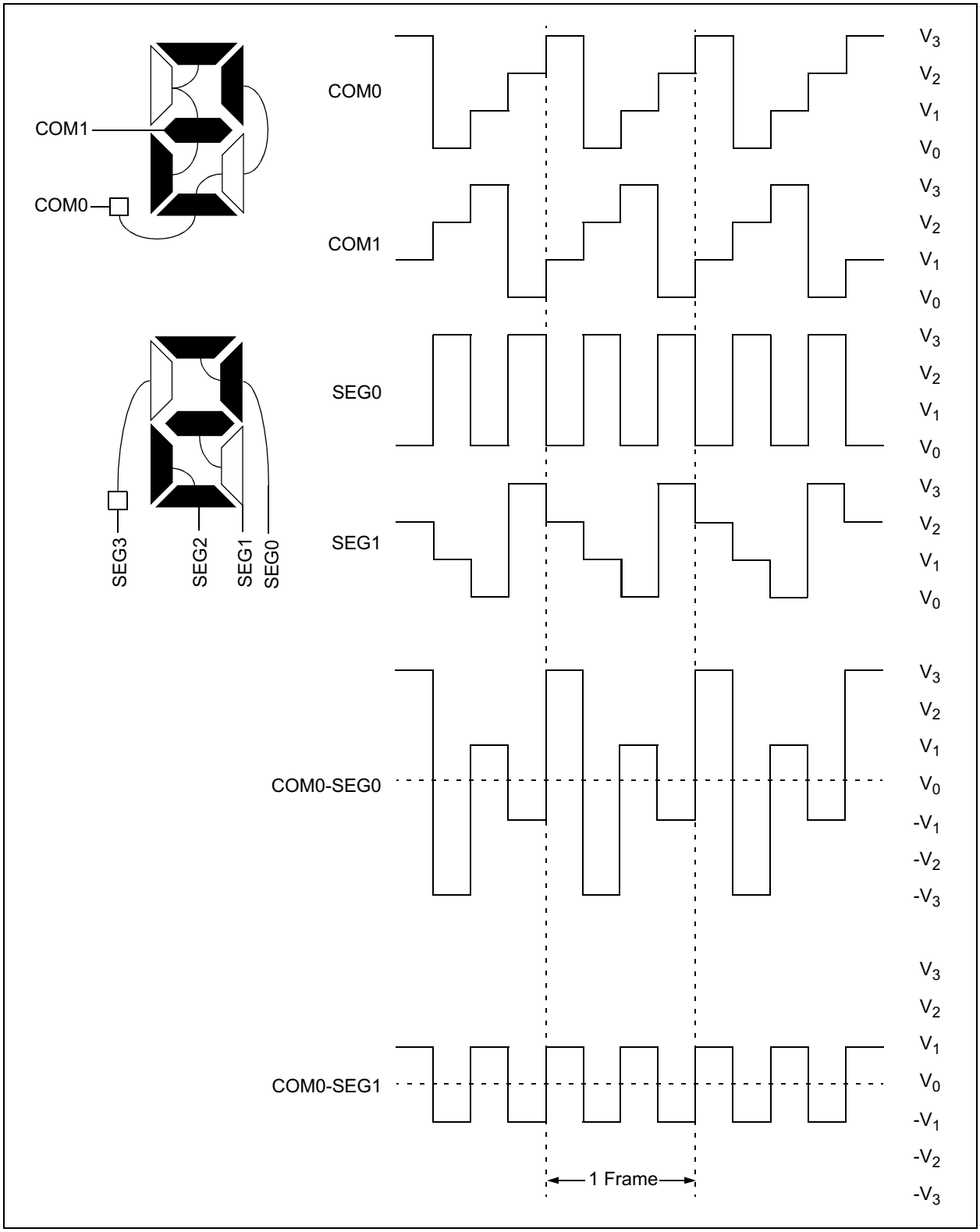


FIGURE 10-10: TYPE-B WAVEFORMS IN 1/2 MUX, 1/3 BIAS DRIVE

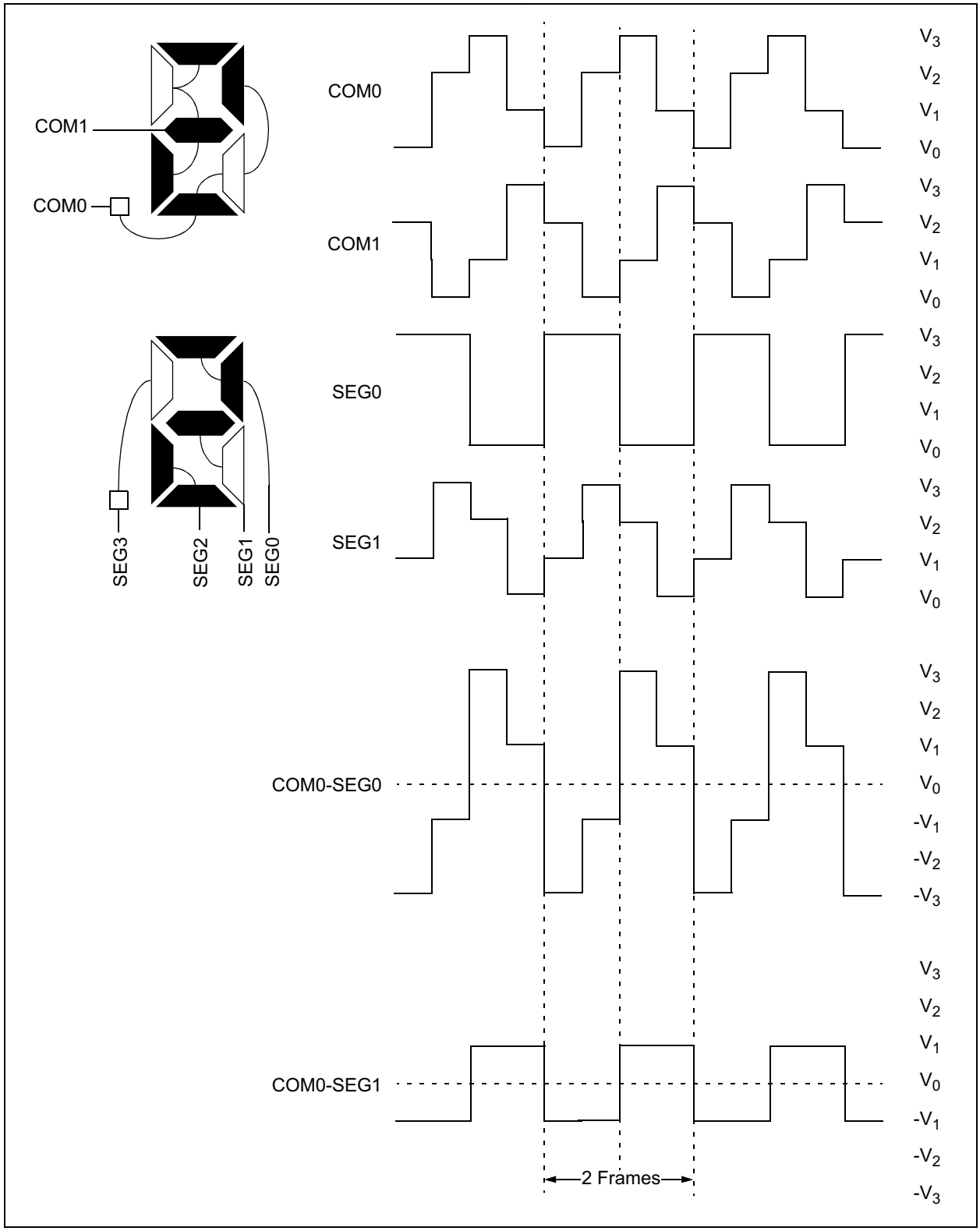


FIGURE 10-11: TYPE-A WAVEFORMS IN 1/3 MUX, 1/2 BIAS DRIVE

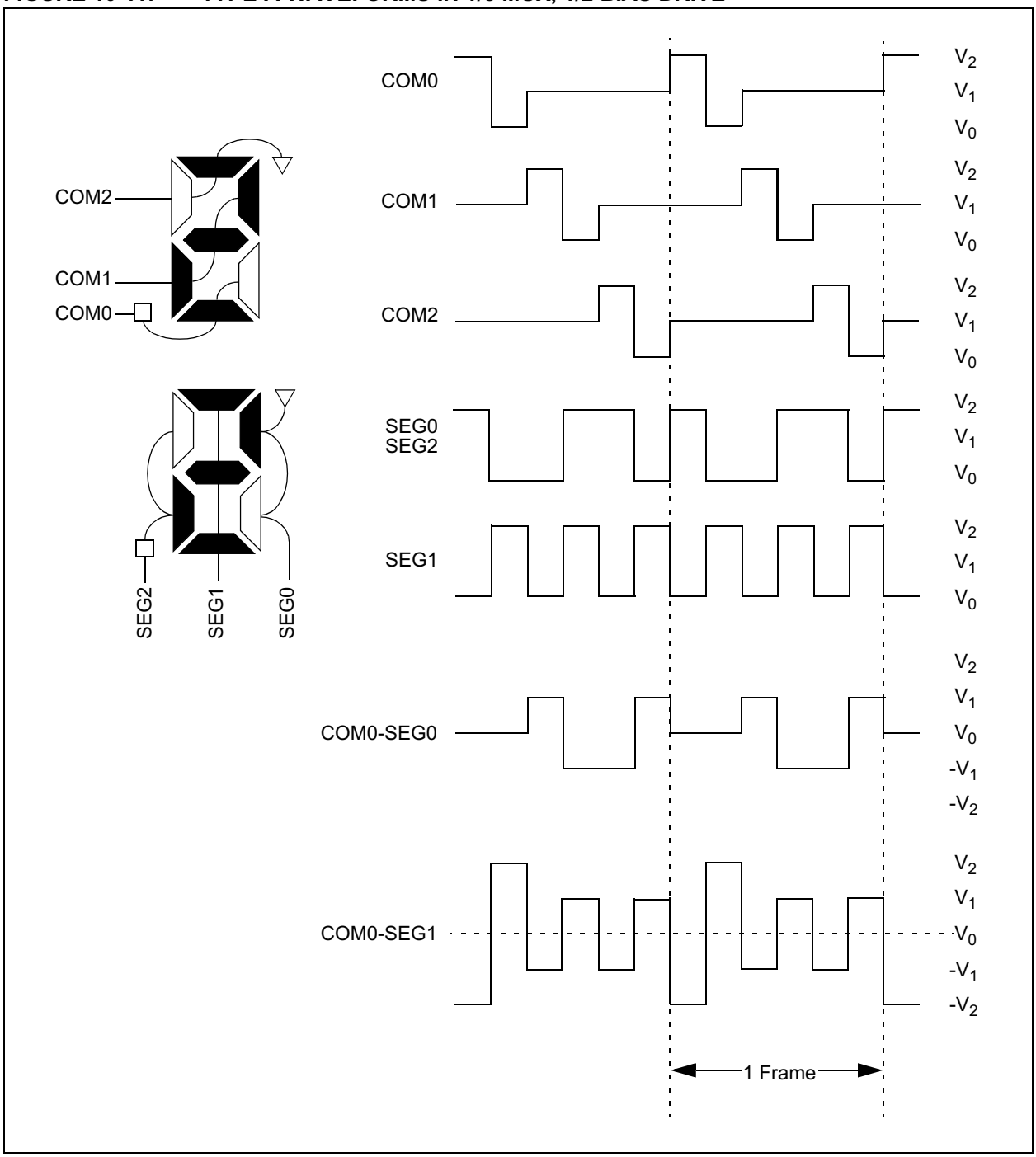


FIGURE 10-12: TYPE-B WAVEFORMS IN 1/3 MUX, 1/2 BIAS DRIVE

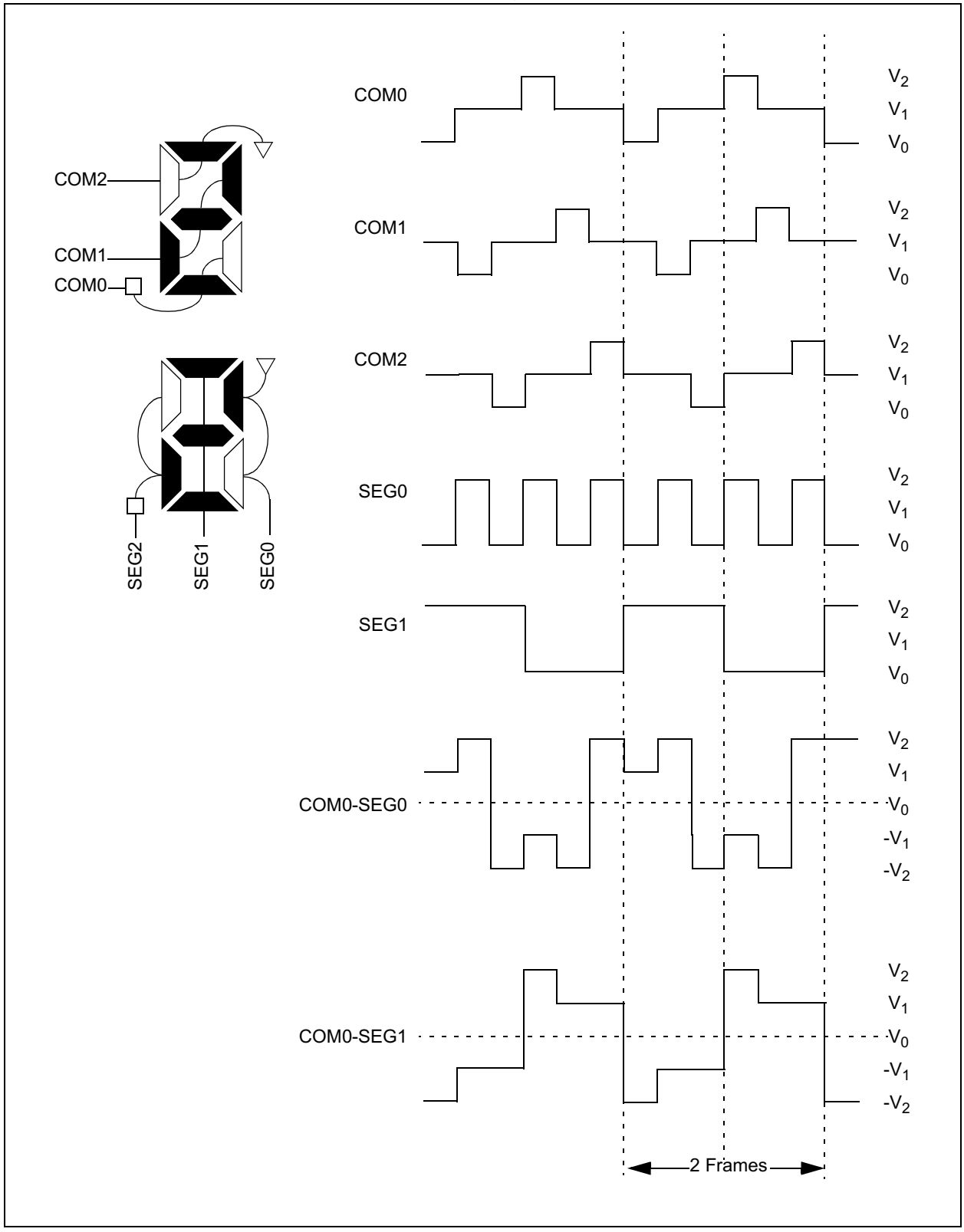


FIGURE 10-13: TYPE-A WAVEFORMS IN 1/3 MUX, 1/3 BIAS DRIVE

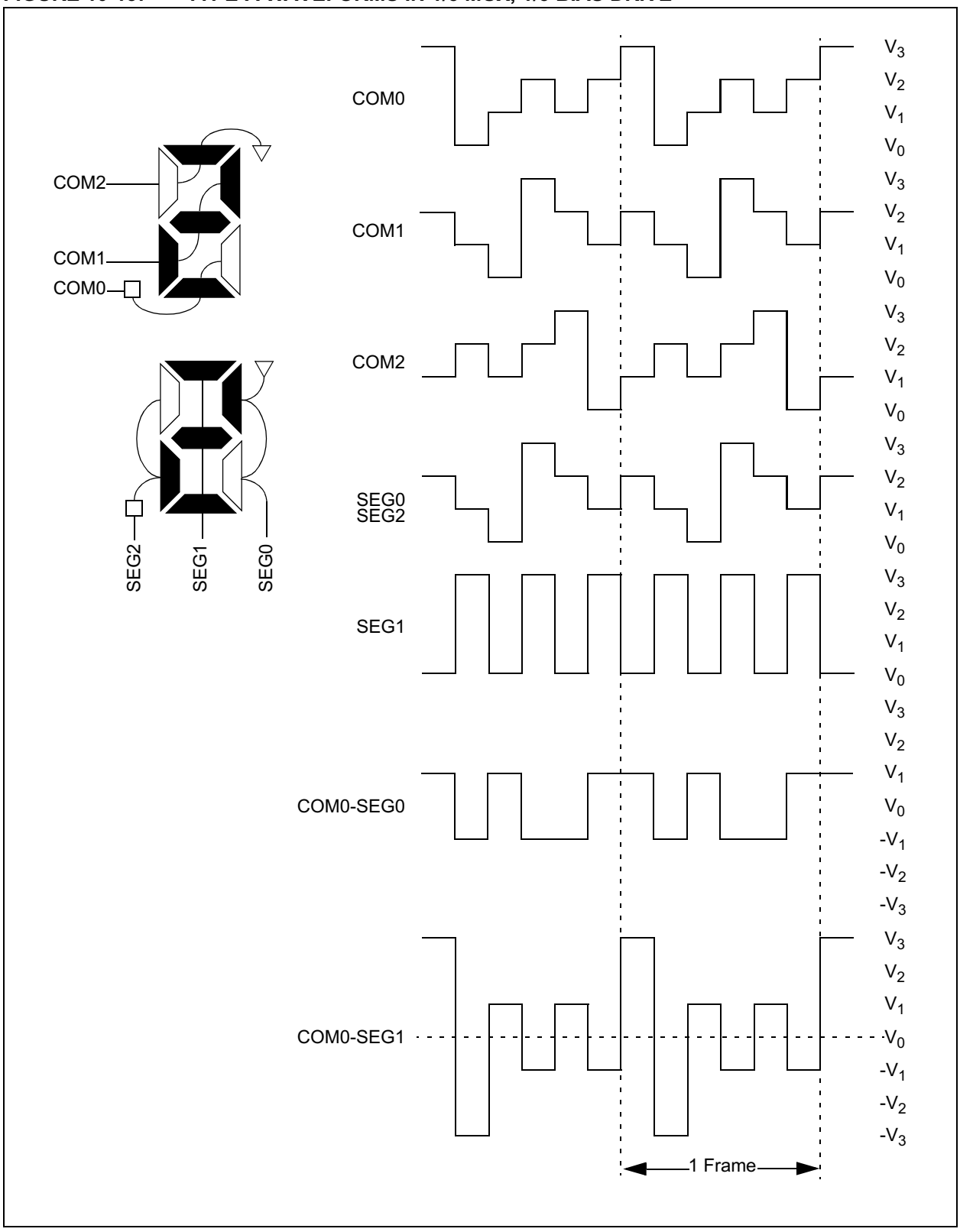


FIGURE 10-14: TYPE-B WAVEFORMS IN 1/3 MUX, 1/3 BIAS DRIVE

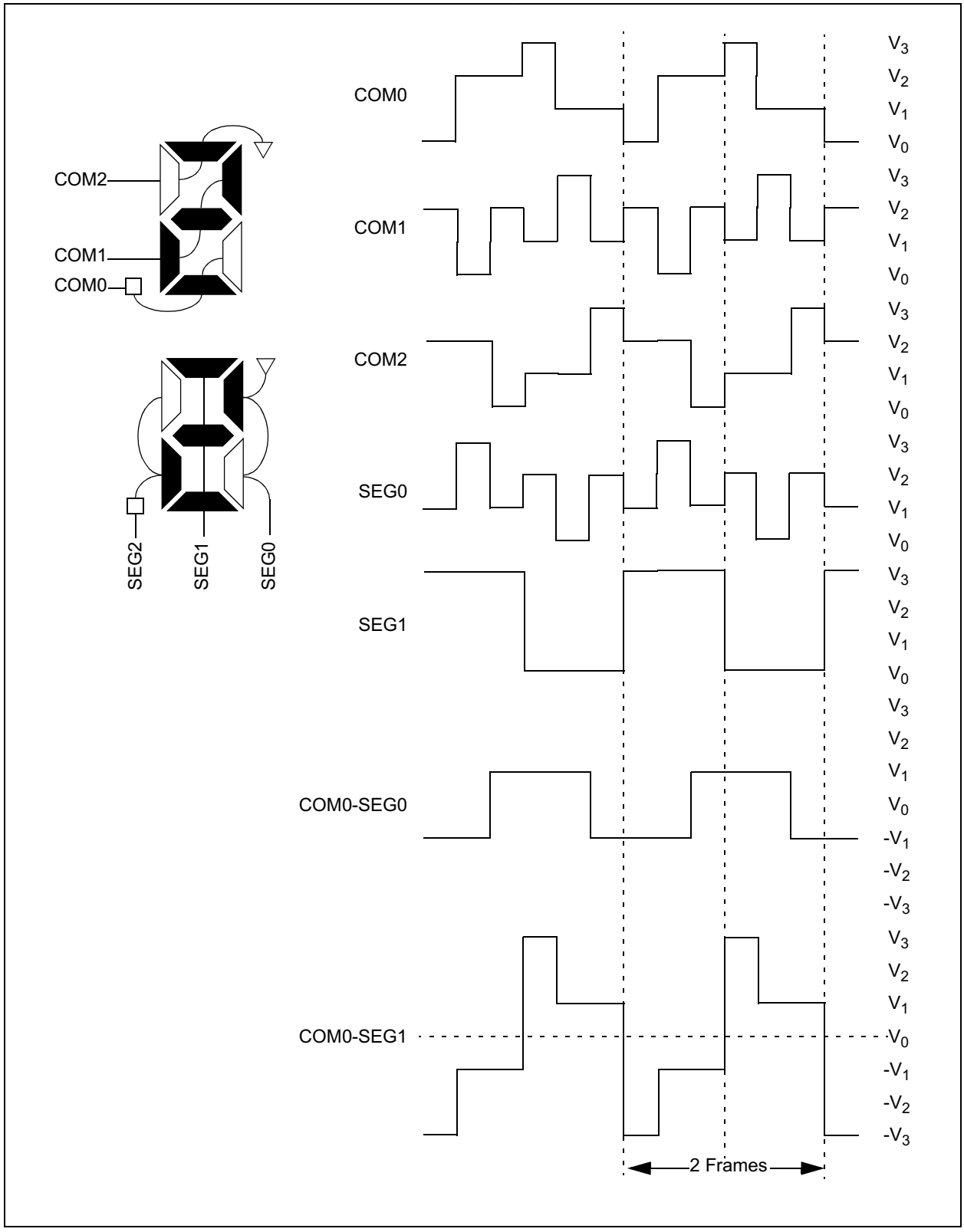


FIGURE 10-15: TYPE-A WAVEFORMS IN 1/4 MUX, 1/3 BIAS DRIVE

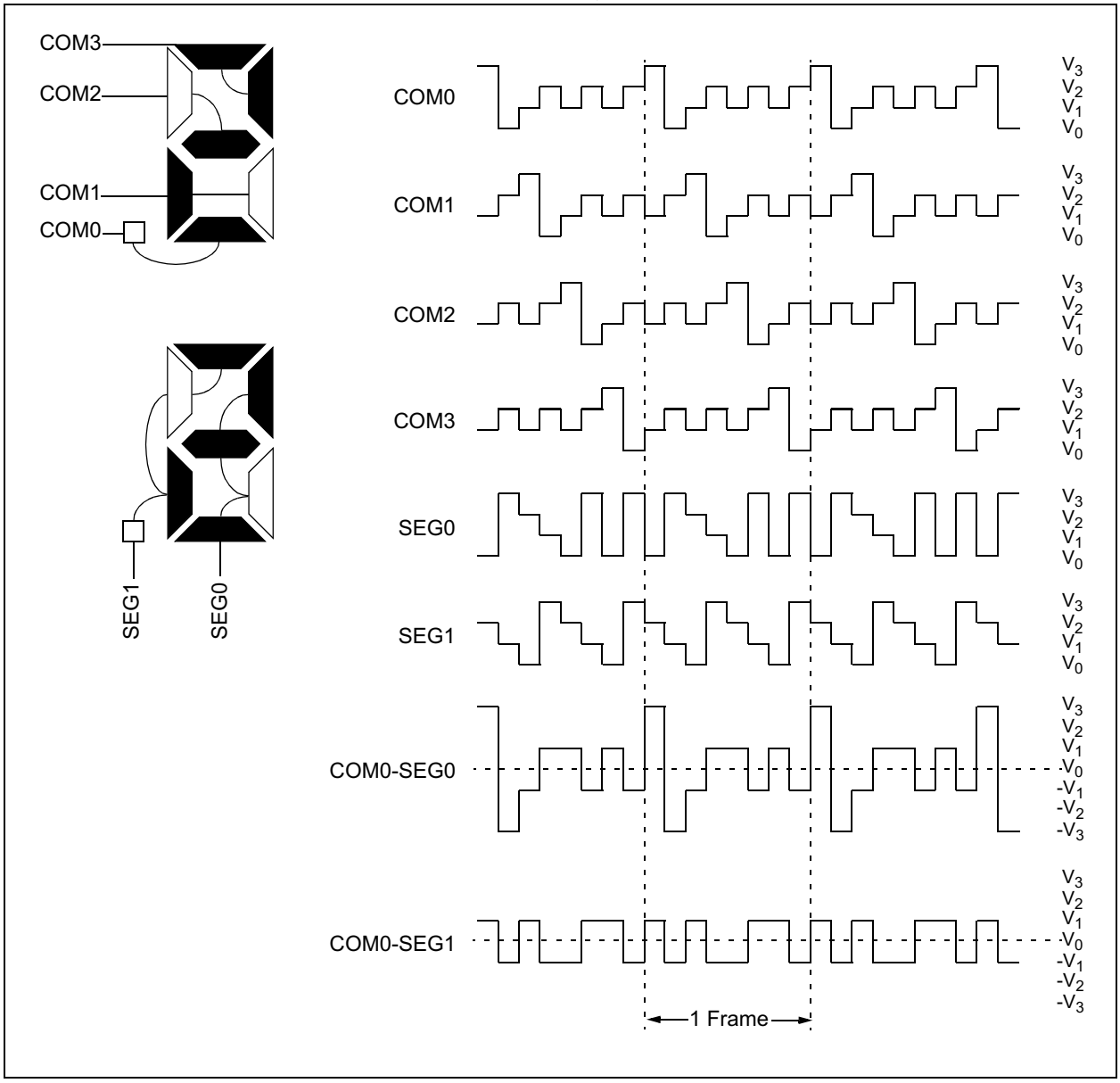
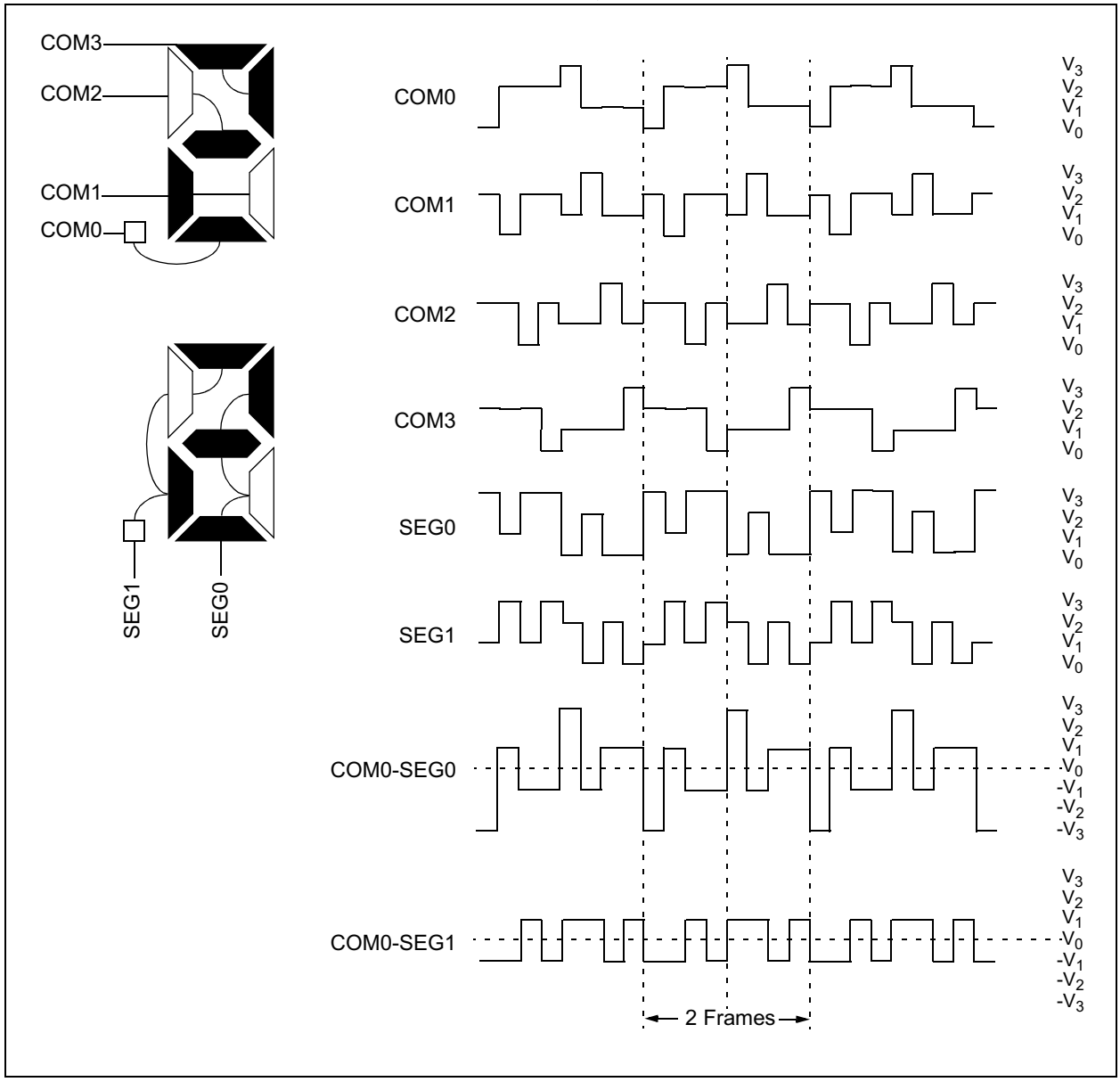


FIGURE 10-16: TYPE-B WAVEFORMS IN 1/4 MUX, 1/3 BIAS DRIVE



10.9 LCD Interrupts

The LCD timing generation provides an interrupt that defines the LCD frame timing.

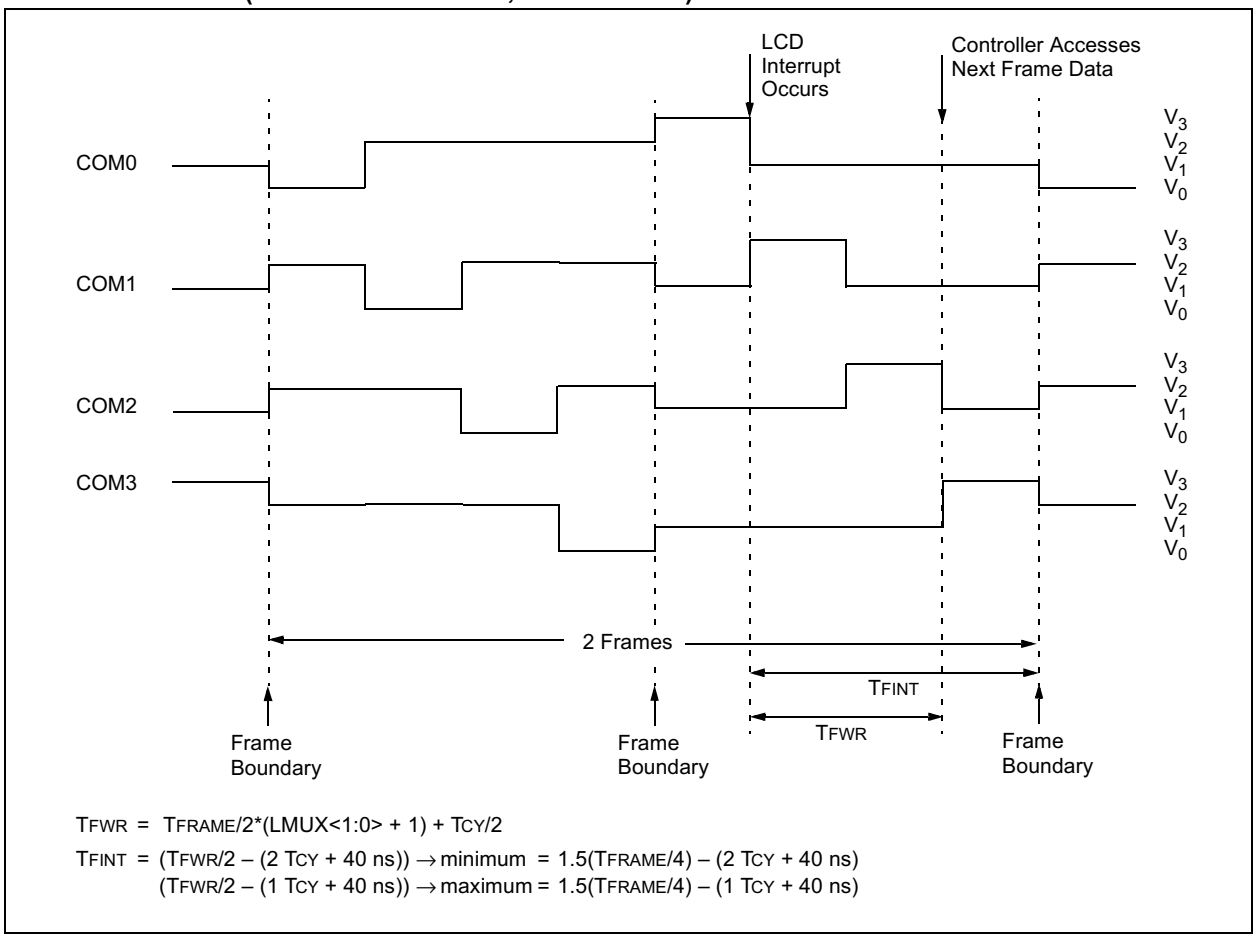
A new frame is defined to begin at the leading edge of the COM0 common signal. The interrupt will be set immediately after the LCD controller completes accessing all pixel data required for a frame. This will occur at a fixed interval before the frame boundary (TFINT), as shown in Figure 10-17. The LCD controller will begin to access data for the next frame within the interval from the interrupt to when the controller begins to access data after the interrupt (TFWR). New data must be written within TFWR, as this is when the LCD controller will begin to access the data for the next frame.

When the LCD driver is running with Type-B waveforms and the LMUX<1:0> bits are not equal to '00' (static drive), there are some additional issues that must be addressed. Since the DC voltage on the pixel takes two frames to maintain zero volts, the pixel data must not change between subsequent frames. If the pixel data were allowed to change, the waveform for the odd frames would not necessarily be the complement of the waveform generated in the even frames and a DC component would be introduced into the panel. Therefore, when using Type-B waveforms, the user must synchronize the LCD pixel updates to occur within a subframe after the frame interrupt.

To correctly sequence writing while in Type-B, the interrupt will only occur on complete phase intervals. If the user attempts to write when the write is disabled, the WERR bit of the LCDCON register is set and the write does not occur.

Note:	The interrupt is not generated when the Type-A waveform is selected and when the Type-B with no multiplex (static) is selected.

FIGURE 10-17: WAVEFORMS AND INTERRUPT TIMING IN QUARTER-DUTY CYCLE DRIVE (EXAMPLE – TYPE-B, NON-STATIC)



$T_{FWR} = T_{FRAME}/2*(LMUX<1:0> + 1) + T_{CY}/2$

$T_{FINT} = (T_{FWR}/2 – (2\ T_{CY} + 40\ ns)) \rightarrow minimum = 1.5(T_{FRAME}/4) – (2\ T_{CY} + 40\ ns)$
$(T_{FWR}/2 – (1\ T_{CY} + 40\ ns)) \rightarrow maximum = 1.5(T_{FRAME}/4) – (1\ T_{CY} + 40\ ns)$

10.10 Operation During Sleep

The LCD module can operate during Sleep. The selection is controlled by bit $\overline{\text{SLPEN}}$ of the LCDCON register. Setting the $\overline{\text{SLPEN}}$ bit allows the LCD module to go to Sleep. Clearing the $\overline{\text{SLPEN}}$ bit allows the module to continue to operate during Sleep.

If a SLEEP instruction is executed and $\overline{\text{SLPEN}}$ = 1, the LCD module will cease all functions and go into a very low-current Consumption mode. The module will stop operation immediately and drive the minimum LCD voltage on both segment and common lines. Figure 10-18 shows this operation.

To ensure that no DC component is introduced on the panel, the SLEEP instruction should be executed immediately after a LCD frame boundary. For Type-B multiplex (non-static), the LCD interrupt can be used to determine the frame boundary. See **Section 10.9 "LCD Interrupts"** for the formulas to calculate the delay. In all other modes, the LCDA bit can be used to determine when the display is active. To use this method, the following sequence should be used when wanting to enter into Sleep mode:

- Clear LCDEN
- Wait for LCDA to clear
- Drive all LCD pins to inactive state using PORT and TRIS registers
- Execute SLEEP instruction

Note:	When the LCDEN bit is cleared, the LCD module will be disabled at the completion of frame. At this time, the PORT pins will revert to digital functionality. To minimize power consumption due to floating digital inputs, the LCD pins should be driven low using the PORT and TRIS registers.

If a SLEEP instruction is executed and $\overline{\text{SLPEN}}$ = 0, the module will continue to display the current contents of the LCDDATA registers. To allow the module to continue operation while in Sleep, the clock source must be either the LFINTOSC or T1OSC external oscillator. While in Sleep, the LCD data cannot be changed. The LCD module current consumption will not decrease in this mode; however, the overall consumption of the device will be lower due to shut down of the core and other peripheral functions.

Table 10-5 shows the status of the LCD module during a Sleep while using each of the three available clock sources:

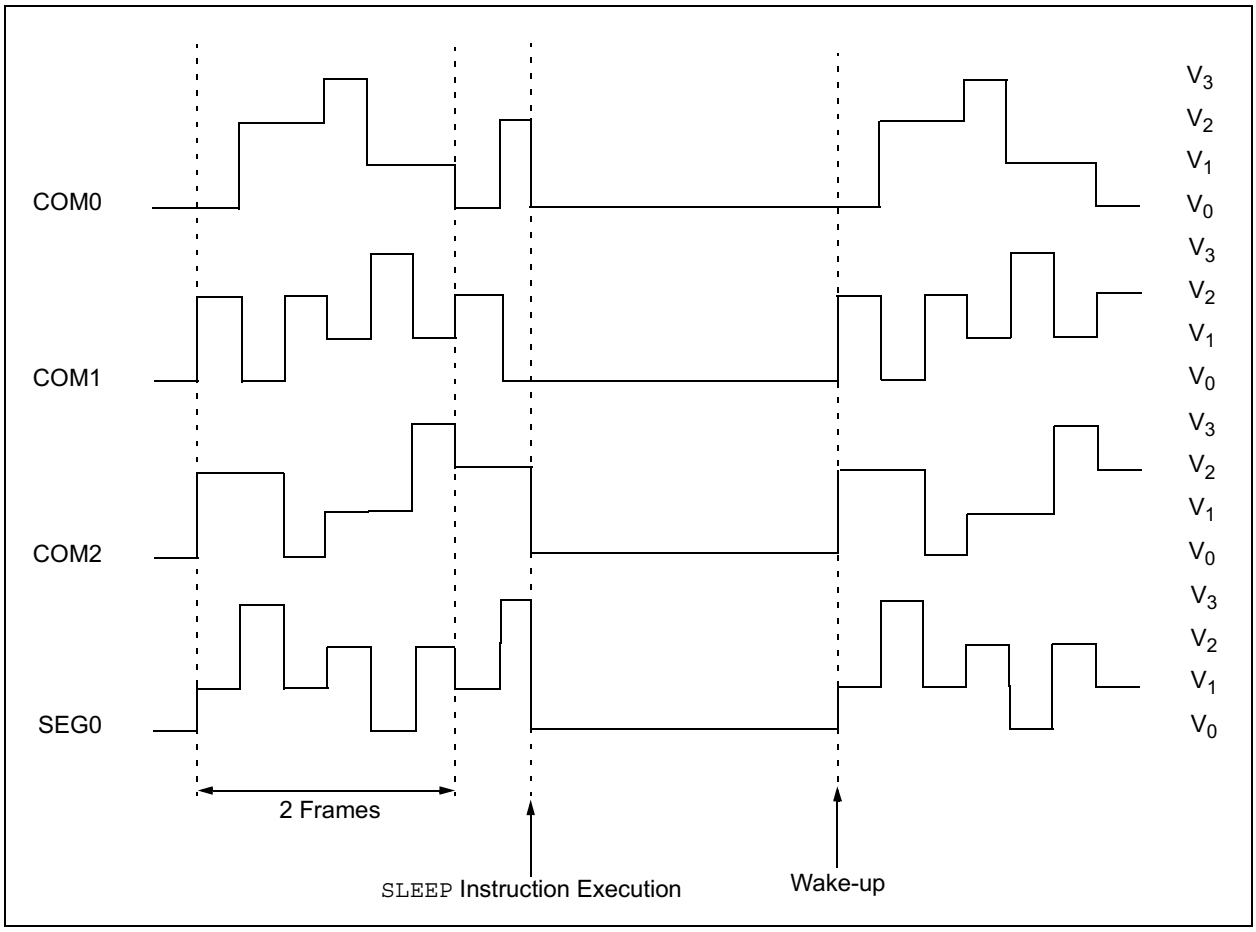
TABLE 10-5: LCD MODULE STATUS DURING SLEEP

Clock Source	$\overline{\text{SLPEN}}$	Operation During Sleep?
T1OSC	0	Yes
	1	No
LFINTOSC	0	Yes
	1	No
FOSC/4	0	No
	1	No

Note:	The LFINTOSC or external T1OSC oscillator must be used to operate the LCD module during Sleep.

If LCD interrupts are being generated (Type-B waveform with a multiplex mode not static) and LCDIE = 1, the device will awaken from Sleep on the next frame boundary.

FIGURE 10-18: SLEEP ENTRY/EXIT WHEN $\overline{\text{SLPEN}}$ = 1



10.11 Configuring the LCD Module

The following is the sequence of steps to configure the LCD module.

1. Select the frame clock prescale using bits LP<3:0> of the LCDPS register.
2. Configure the appropriate pins to function as segment drivers using the LCDSEn registers.
3. Configure the LCD module for the following using the LCDCON register:
   - Multiplex and Bias mode, bits LMUX<1:0>
   - Timing source, bits CS<1:0>
   - Sleep mode, bit $\overline{\text{SLPEN}}$
4. Write initial values to pixel data registers, LCDDATA0 through LCDDATA11 (LCDDATA23 on PIC16F946).
5. Clear LCD Interrupt Flag, LCDIF bit of the PIR2 register and if desired, enable the interrupt by setting bit LCDIE of the PIE2 register.
6. Enable bias voltage pins (VLCD<3:1>) by setting bit VLCDEN of the LCDCON register.
7. Enable the LCD module by setting bit LCDEN of the LCDCON register.

10.12 Disabling the LCD Module

To disable the LCD module, write all '0's to the LCDCON register.

10.13 LCD Current Consumption

When using the LCD module the current consumption consists of the following three factors:

1. The oscillator selected
2. The LCD bias source
3. The current required to charge the LCD segments

The current consumption of just the LCD module can be considered negligible compared to these other factors.

The oscillator selected:

For LCD operation during Sleep either the T1oc or the LFINTOSC sources need to be used as the main system oscillator may be disabled during Sleep. During Sleep the LFINTOSC current consumption is given by electrical parameter D021, where the LFINTOSC use the same internal oscillator circuitry as the Watchdog Timer.

The LCD bias source:

The LCD bias source, typically an external resistor ladder which will have its own current draw.

The current required to charge the LCD segments:

The LCD segments which can be modeled as capacitors which must be both charged and discharged every frame. The size of the LCD segment and its technology determines the segment's capacitance.

TABLE 10-6: REGISTERS ASSOCIATED WITH LCD OPERATION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
CMCON0	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0000	0000 0000
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDDATA0	SEG7 COM0	SEG6 COM0	SEG5 COM0	SEG4 COM0	SEG3 COM0	SEG2 COM0	SEG1 COM0	SEG0 COM0	xxxx xxxx	uuuu uuuu
LCDDATA1	SEG15 COM0	SEG14 COM0	SEG13 COM0	SEG12 COM0	SEG11 COM0	SEG10 COM0	SEG9 COM0	SEG8 COM0	xxxx xxxx	uuuu uuuu
LCDDATA2(2)	SEG23 COM0	SEG22 COM0	SEG21 COM0	SEG20 COM0	SEG19 COM0	SEG18 COM0	SEG17 COM0	SEG16 COM0	xxxx xxxx	uuuu uuuu
LCDDATA3	SEG7 COM1	SEG6 COM1	SEG5 COM1	SEG4 COM1	SEG3 COM1	SEG2 COM1	SEG1 COM1	SEG0 COM1	xxxx xxxx	uuuu uuuu
LCDDATA4	SEG15 COM1	SEG14 COM1	SEG13 COM1	SEG12 COM1	SEG11 COM1	SEG10 COM1	SEG9 COM1	SEG8 COM1	xxxx xxxx	uuuu uuuu
LCDDATA5(2)	SEG23 COM1	SEG22 COM1	SEG21 COM1	SEG20 COM1	SEG19 COM1	SEG18 COM1	SEG17 COM1	SEG16 COM1	xxxx xxxx	uuuu uuuu
LCDDATA6	SEG7 COM2	SEG6 COM2	SEG5 COM2	SEG4 COM2	SEG3 COM2	SEG2 COM2	SEG1 COM2	SEG0 COM2	xxxx xxxx	uuuu uuuu
LCDDATA7	SEG15 COM2	SEG14 COM2	SEG13 COM2	SEG12 COM2	SEG11 COM2	SEG10 COM2	SEG9 COM2	SEG8 COM2	xxxx xxxx	uuuu uuuu
LCDDATA8(2)	SEG23 COM2	SEG22 COM2	SEG21 COM2	SEG20 COM2	SEG19 COM2	SEG18 COM2	SEG17 COM2	SEG16 COM2	xxxx xxxx	uuuu uuuu
LCDDATA9	SEG7 COM3	SEG6 COM3	SEG5 COM3	SEG4 COM3	SEG3 COM3	SEG2 COM3	SEG1 COM3	SEG0 COM3	xxxx xxxx	uuuu uuuu
LCDDATA10	SEG15 COM3	SEG14 COM3	SEG13 COM3	SEG12 COM3	SEG11 COM3	SEG10 COM3	SEG9 COM3	SEG8 COM3	xxxx xxxx	uuuu uuuu
LCDDATA11(2)	SEG23 COM3	SEG22 COM3	SEG21 COM3	SEG20 COM3	SEG19 COM3	SEG18 COM3	SEG17 COM3	SEG16 COM3	xxxx xxxx	uuuu uuuu
LCDDATA12(3)	SEG31 COM0	SEG30 COM0	SEG29 COM0	SEG28 COM0	SEG27 COM0	SEG26 COM0	SEG25 COM0	SEG24 COM0	xxxx xxxx	uuuu uuuu
LCDDATA13(3)	SEG39 COM0	SEG38 COM0	SEG37 COM0	SEG36 COM0	SEG35 COM0	SEG34 COM0	SE33 COM0	SEG32 COM0	xxxx xxxx	uuuu uuuu
LCDDATA14(3)	—	—	—	—	—	—	SEG41 COM0	SEG40 COM0	---- --xx	---- --uu
LCDDATA15(3)	SEG31 COM1	SEG30 COM1	SEG29 COM1	SEG28 COM1	SEG27 COM1	SEG26 COM1	SEG25 COM1	SEG24 COM1	xxxx xxxx	uuuu uuuu
LCDDATA16(3)	SEG39 COM1	SEG38 COM1	SEG37 COM1	SEG36 COM1	SEG35 COM1	SEG34 COM1	SEG33 COM1	SEG32 COM1	xxxx xxxx	uuuu uuuu
LCDDATA17(3)	—	—	—	—	—	—	SEG41 COM1	SEG40 COM1	---- --xx	---- --uu
LCDDATA18(3)	SEG31 COM2	SEG30 COM2	SEG29 COM2	SEG28 COM2	SEG27 COM2	SEG26 COM2	SEG25 COM2	SEG24 COM2	xxxx xxxx	uuuu uuuu
LCDDATA19(3)	SEG39 COM2	SEG38 COM2	SEG37 COM2	SEG36 COM2	SEG35 COM2	SEG34 COM2	SEG33 COM2	SEG32 COM2	xxxx xxxx	uuuu uuuu
LCDDATA20(3)	—	—	—	—	—	—	SEG41 COM2	SEG40 COM2	---- --xx	---- --uu
LCDDATA21(3)	SEG31 COM3	SEG30 COM3	SEG29 COM3	SEG28 COM3	SEG27 COM3	SEG26 COM3	SEG25 COM3	SEG24 COM3	xxxx xxxx	uuuu uuuu
LCDDATA22(3)	SEG39 COM3	SEG38 COM3	SEG37 COM3	SEG36 COM3	SEG35 COM3	SEG34 COM3	SEG33 COM3	SEG32 COM3	xxxx xxxx	uuuu uuuu
LCDDATA23(3)	—	—	—	—	—	—	SEG41 COM3	SEG40 COM3	---- --xx	---- --uu
LCDPS	WFT	BIASMD	LCDA	WA	LP3	LP2	LP1	LP0	0000 0000	0000 0000
LCDSE0	SE7	SE6	SE5	SE4	SE3	SE2	SE1	SE0	0000 0000	uuuu uuuu
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	uuuu uuuu

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by the LCD module.
Note 1: These pins may be configured as port pins, depending on the oscillator mode selected.
2: PIC16F914/917 and PIC16F946 only.
3: PIC16F946 only.

TABLE 10-6: REGISTERS ASSOCIATED WITH LCD OPERATION (CONTINUED)

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
LCDSE2(2)	SE23	SE22	SE21	SE20	SE19	SE18	SE17	SE16	0000 0000	uuuu uuuu
LCDSE3(3)	SE31	SE30	SE29	SE28	SE27	SE26	SE25	SE24	0000 0000	0000 0000
LCDSE4(3)	SE39	SE38	SE37	SE36	SE35	SE34	SE33	SE32	0000 0000	0000 0000
LCDSE5(3)	—	—	—	—	—	—	SE41	SE40	---- --00	---- --00
PIE2	OSFIE	C2IE	C1IE	LCDIE	—	LVDIE	—	CCP2IE	0000 -0-0	0000 -0-0
PIR2	OSFIF	C2IF	C1IF	LCDIF	—	LVDIF	—	CCP2IF	0000 -0-0	0000 -0-0
T1CON	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	0000 0000	uuuu uuuu

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by the LCD module.
Note 1: These pins may be configured as port pins, depending on the oscillator mode selected.
2: PIC16F914/917 and PIC16F946 only.
3: PIC16F946 only.

NOTES:

11.0 PROGRAMMABLE LOW-VOLTAGE DETECT (PLVD) MODULE

The Programmable Low-Voltage Detect (PLVD) module is a power supply detector which monitors the internal power supply. This module is typically used in key fobs and other devices, where certain actions need to be taken as a result of a falling battery voltage.

The PLVD module includes the following capabilities:

- Eight programmable trip points
- Interrupt on falling VDD
- Stable reference indication
- Operation during Sleep

A Block diagram of the PLVD module is shown in Figure 11-1.

FIGURE 11-1: PLVD BLOCK DIAGRAM

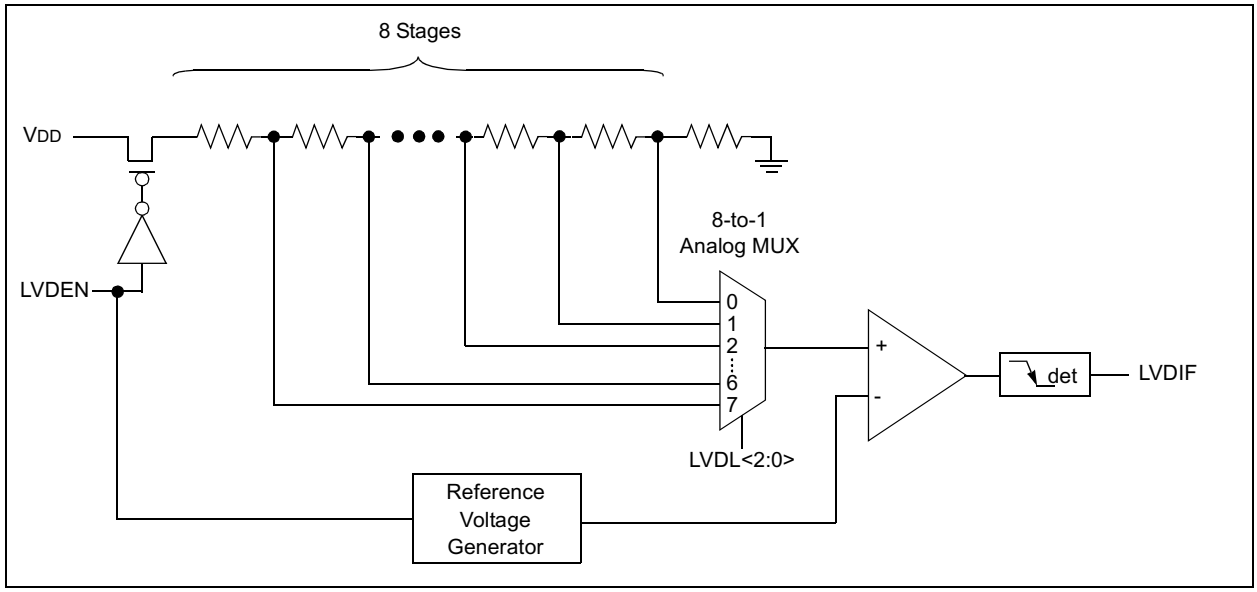
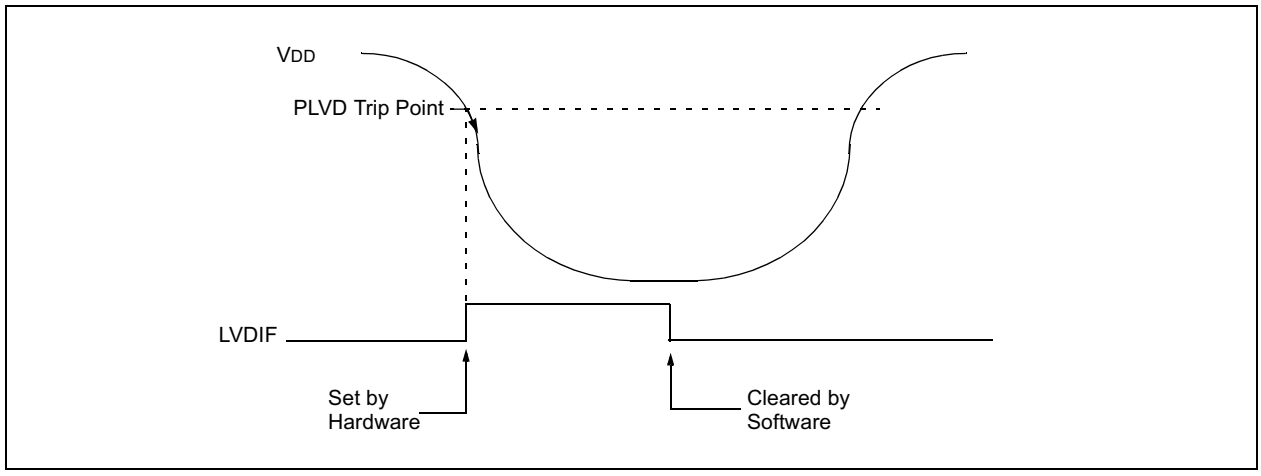


FIGURE 11-2: PLVD OPERATION



11.1 PLVD Operation

To setup the PLVD for operation, the following steps must be taken:

- Enable the module by setting the LVDEN bit of the LVDCON register.
- Configure the trip point by setting the LVDL<2:0> bits of the LVDCON register.
- Wait for the reference voltage to become stable. Refer to **Section 11.4 "Stable Reference Indication"**.
- Clear the LVDIF bit of the PIR2 register.

The LVDIF bit will be set when VDD falls below the PLVD trip point. The LVDIF bit remains set until cleared by software. Refer to Figure 11-2.

11.2 Programmable Trip Point

The PLVD trip point is selectable from one of eight voltage levels. The LVDL bits of the LVDCON register select the trip point. Refer to Register 11-1 for the available PLVD trip points.

11.3 Interrupt on Falling VDD

When VDD falls below the PLVD trip point, the falling edge detector will set the LVDIF bit. See Figure 11-2. An interrupt will be generated if the following bits are also set:

- GIE and PEIE bits of the INTCON register
- LVDIE bit of the PIE2 register

The LVDIF bit must be cleared by software. An interrupt can be generated from a simulated PLVD event when the LVDIF bit is set by software.

11.4 Stable Reference Indication

When the PLVD module is enabled, the reference voltage must be allowed to stabilize before the PLVD will provide a valid result. Refer to **Section 19.0 "Electrical Specifications"**, Table 19-13, for the stabilization time.

When the HFINTOSC is running, the IRVST bit of the LVDCON register indicates the stability of the voltage reference. The voltage reference is stable when the IRVST bit is set.

11.5 Operation During Sleep

To wake from Sleep, set the LVDIE bit of the PIE2 register and the PEIE bit of the INTCON register. When the LVDIE and PEIE bits are set, the device will wake from Sleep and execute the next instruction. If the GIE bit is also set, the program will call the Interrupt Service Routine upon completion of the first instruction after waking from Sleep.

REGISTER 11-1: LVDCON: LOW-VOLTAGE DETECT CONTROL REGISTER

U-0	U-0	R-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	—	IRVST(1)	LVDEN	—	LVDL2	LVDL1	LVDL0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-6 **Unimplemented:** Read as '0'

bit 5 **IRVST:** Internal Reference Voltage Stable Status Flag bit(1)
1 = Indicates that the PLVD is stable and PLVD interrupt is reliable
0 = Indicates that the PLVD is not stable and PLVD interrupt must not be enabled

bit 4 **LVDEN:** Low-Voltage Detect Module Enable bit
1 = Enables PLVD Module, powers up PLVD circuit and supporting reference circuitry
0 = Disables PLVD Module, powers down PLVD circuit and supporting reference circuitry

bit 3 **Unimplemented:** Read as '0'

bit 2-0 **LVDL<2:0>:** Low-Voltage Detection Level bits (nominal values)(3)
111 = 4.5V
110 = 4.2V
101 = 4.0V
100 = 2.3V (default)
011 = 2.2V
010 = 2.1V
001 = 2.0V(2)
000 = Reserved

Note 1: The IRVST bit is usable only when the HFINTOSC is running.
2: Not tested and below minimum operating conditions.
3: See **Section 19.0 "Electrical Specifications"**.

TABLE 11-1: REGISTERS ASSOCIATED WITH PROGRAMMABLE LOW-VOLTAGE DETECT

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
LVDCON	—	—	IRVST	LVDEN	—	LVDL2	LVDL1	LVDL0	--00 -100	--00 -100
PIE2	OSFIE	C2IE	C1IE	LCDIE	—	LVDIE	—	CCP2IE	0000 -0-0	0000 -0-0
PIR2	OSFIF	C2IF	C1IF	LCDIF	—	LVDIF	—	CCP2IF	0000 -0-0	0000 -0-0

Legend: x = unknown, - = unimplemented read as '0'. Shaded cells are not used by the PLVD module.

NOTES:

12.0 ANALOG-TO-DIGITAL CONVERTER (ADC) MODULE

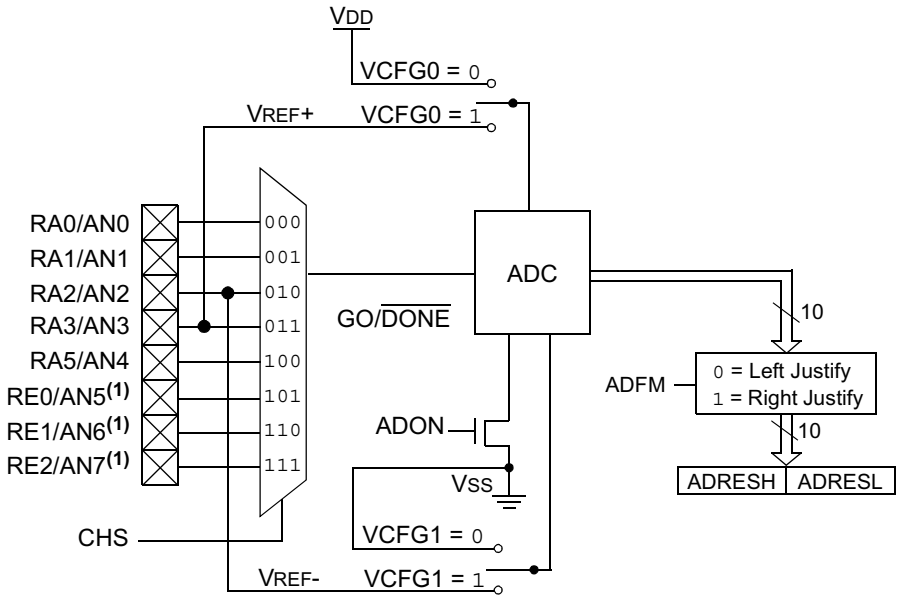
The Analog-to-Digital Converter (ADC) allows conversion of an analog input signal to a 10-bit binary representation of that signal. This device uses analog inputs, which are multiplexed into a single sample and hold circuit. The output of the sample and hold is connected to the input of the converter. The converter generates a 10-bit binary result via successive approximation and stores the conversion result into the ADC result registers (ADRESL and ADRESH).

The ADC voltage reference is software selectable to be either internally generated or externally supplied.

The ADC can generate an interrupt upon completion of a conversion. This interrupt can be used to wake-up the device from Sleep.

Figure 12-1 shows the block diagram of the ADC.

FIGURE 12-1: ADC BLOCK DIAGRAM



Note 1: These channels are only available on PIC16F914/917 and PIC16F946 devices.

12.1 ADC Configuration

When configuring and using the ADC the following functions must be considered:

- Port configuration
- Channel selection
- ADC voltage reference selection
- ADC conversion clock source
- Interrupt control
- Results formatting

12.1.1 PORT CONFIGURATION

The ADC can be used to convert both analog and digital signals. When converting analog signals, the I/O pin should be configured for analog by setting the associated TRIS and ANSEL bits. See the corresponding Port section for more information.

Note:	Analog voltages on any pin that is defined as a digital input may cause the input buffer to conduct excess current.

12.1.2 CHANNEL SELECTION

The CHS bits of the ADCON0 register determine which channel is connected to the sample and hold circuit.

When changing channels, a delay is required before starting the next conversion. Refer to **Section 12.2 "ADC Operation"** for more information.

12.1.3 ADC VOLTAGE REFERENCE

The VCFG bits of the ADCON0 register provide independent control of the positive and negative voltage references. The positive voltage reference can be either VDD or an external voltage source. Likewise, the negative voltage reference can be either VSS or an external voltage source.

12.1.4 CONVERSION CLOCK

The source of the conversion clock is software selectable via the ADCS bits of the ADCON1 register. There are seven possible clock options:

- FOSC/2
- FOSC/4
- FOSC/8
- FOSC/16
- FOSC/32
- FOSC/64
- FRC (dedicated internal oscillator)

The time to complete one bit conversion is defined as TAD. One full 10-bit conversion requires 11 TAD periods as shown in Figure 12-3.

For correct conversion, the appropriate TAD specification must be met. See A/D conversion requirements in **Section 19.0 "Electrical Specifications"** for more information. Table 12-1 gives examples of appropriate ADC clock selections.

Note:	Unless using the FRC, any changes in the system clock frequency will change the ADC clock frequency, which may adversely affect the ADC result.

TABLE 12-1: ADC CLOCK PERIOD (TAD) VS. DEVICE OPERATING FREQUENCIES (VDD ≥ 3.0V)

ADC Clock Period (TAD)		Device Frequency (FOSC)			
ADC Clock Source	**ADCS<2:0>**	**20 MHz**	**8 MHz**	**4 MHz**	**1 MHz**
FOSC/2	000	100 ns(2)	250 ns(2)	500 ns(2)	2.0 µs
FOSC/4	100	200 ns(2)	500 ns(2)	1.0 µs(2)	4.0 µs
FOSC/8	001	400 ns(2)	1.0 µs(2)	2.0 µs	8.0 µs(3)
FOSC/16	101	800 ns(2)	2.0 µs	4.0 µs	16.0 µs(3)
FOSC/32	010	1.6 µs	4.0 µs	8.0 µs(3)	32.0 µs(3)
FOSC/64	110	3.2 µs	8.0 µs(3)	16.0 µs(3)	64.0 µs(3)
FRC	x11	2-6 µs(1,4)	2-6 µs(1,4)	2-6 µs(1,4)	2-6 µs(1,4)

Legend: Shaded cells are outside of recommended range.

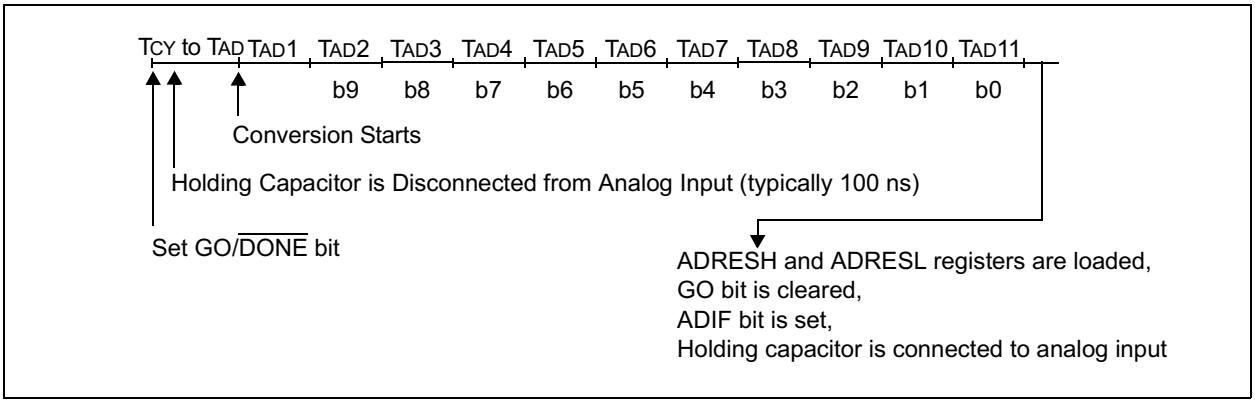
Note 1: The FRC source has a typical TAD time of 4 µs for VDD > 3.0V.

2: These values violate the minimum required TAD time.

3: For faster conversion times, the selection of another clock source is recommended.

4: When the device frequency is greater than 1 MHz, the FRC clock source is only recommended if the conversion will be performed during Sleep.

FIGURE 12-2: ANALOG-TO-DIGITAL CONVERSION TAD CYCLES



12.1.5 INTERRUPTS

The ADC module allows for the ability to generate an interrupt upon completion of an Analog-to-Digital conversion. The ADC interrupt flag is the ADIF bit in the PIR1 register. The ADC interrupt enable is the ADIE bit in the PIE1 register. The ADIF bit must be cleared in software.

Note:	The ADIF bit is set at the completion of every conversion, regardless of whether or not the ADC interrupt is enabled.

This interrupt can be generated while the device is operating or while in Sleep. If the device is in Sleep, the interrupt will wake-up the device. Upon waking from Sleep, the next instruction following the SLEEP instruction is always executed. If the user is attempting to wake-up from Sleep and resume in-line code execution, the global interrupt must be disabled. If the global interrupt is enabled, execution will switch to the Interrupt Service Routine.

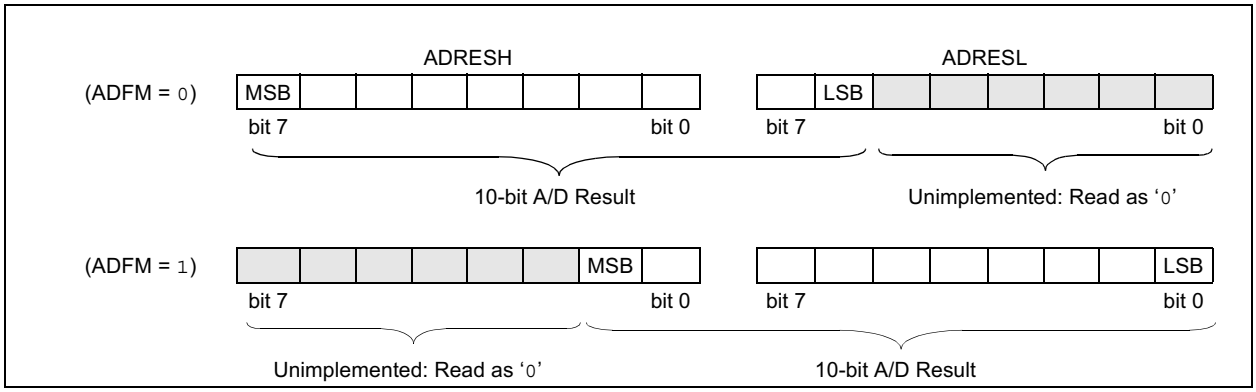
Please see **Section 12.1.5 "Interrupts"** for more information.

12.1.6 RESULT FORMATTING

The 10-bit A/D conversion result can be supplied in two formats, left justified or right justified. The ADFM bit of the ADCON0 register controls the output format.

Figure 12-4 shows the two output formats.

FIGURE 12-3: 10-BIT A/D CONVERSION RESULT FORMAT



12.2 ADC Operation

12.2.1 STARTING A CONVERSION

To enable the ADC module, the ADON bit of the ADCON0 register must be set to a '1'. Setting the GO/$\overline{\text{DONE}}$ bit of the ADCON0 register to a '1' will start the Analog-to-Digital conversion.

Note:	The GO/$\overline{\text{DONE}}$ bit should not be set in the same instruction that turns on the ADC. Refer to **Section 12.2.6 "A/D Conversion Procedure"**.

12.2.2 COMPLETION OF A CONVERSION

When the conversion is complete, the ADC module will:

- Clear the GO/$\overline{\text{DONE}}$ bit
- Set the ADIF flag bit
- Update the ADRESH:ADRESL registers with new conversion result

12.2.3 TERMINATING A CONVERSION

If a conversion must be terminated before completion, the GO/$\overline{\text{DONE}}$ bit can be cleared in software. The ADRESH:ADRESL registers will not be updated with the partially complete Analog-to-Digital conversion sample. Instead, the ADRESH:ADRESL register pair will retain the value of the previous conversion. Additionally, a 2 TAD delay is required before another acquisition can be initiated. Following this delay, an input acquisition is automatically started on the selected channel.

Note:	A device Reset forces all registers to their Reset state. Thus, the ADC module is turned off and any pending conversion is terminated.

12.2.4 ADC OPERATION DURING SLEEP

The ADC module can operate during Sleep. This requires the ADC clock source to be set to the FRC option. When the FRC clock source is selected, the ADC waits one additional instruction before starting the conversion. This allows the SLEEP instruction to be executed, which can reduce system noise during the conversion. If the ADC interrupt is enabled, the device will wake-up from Sleep when the conversion completes. If the ADC interrupt is disabled, the ADC module is turned off after the conversion completes, although the ADON bit remains set.

When the ADC clock source is something other than FRC, a SLEEP instruction causes the present conversion to be aborted and the ADC module is turned off, although the ADON bit remains set.

12.2.5 SPECIAL EVENT TRIGGER

The CCP Special Event Trigger allows periodic ADC measurements without software intervention. When this trigger occurs, the GO/$\overline{\text{DONE}}$ bit is set by hardware and the Timer1 counter resets to zero.

Using the Special Event Trigger does not assure proper ADC timing. It is the user's responsibility to ensure that the ADC timing requirements are met.

See **Section 15.0 "Capture/Compare/PWM (CCP) Module"** for more information.

12.2.6 A/D CONVERSION PROCEDURE

This is an example procedure for using the ADC to perform an Analog-to-Digital conversion:

1. Configure Port:
   - Disable pin output driver (See TRIS register)
   - Configure pin as analog
2. Configure the ADC module:
   - Select ADC conversion clock
   - Configure voltage reference
   - Select ADC input channel
   - Select result format
   - Turn on ADC module
3. Configure ADC interrupt (optional):
   - Clear ADC interrupt flag
   - Enable ADC interrupt
   - Enable peripheral interrupt
   - Enable global interrupt(1)
4. Wait the required acquisition time(2).
5. Start conversion by setting the GO/$\overline{\text{DONE}}$ bit.
6. Wait for ADC conversion to complete by one of the following:
   - Polling the GO/$\overline{\text{DONE}}$ bit
   - Waiting for the ADC interrupt (interrupts enabled)
7. Read ADC Result
8. Clear the ADC interrupt flag (required if interrupt is enabled).

Note 1: The global interrupt can be disabled if the user is attempting to wake-up from Sleep and resume in-line code execution.

2: See **Section 12.3 "A/D Acquisition Requirements"**.

EXAMPLE 12-1: A/D CONVERSION

```
;This code block configures the ADC
;for polling, Vdd reference, Frc clock
;and AN0 input.
;
;Conversion start & polling for completion
; are included.
;
BANKSEL   ADCON1       ;
MOVLW     B'01110000'  ;ADC Frc clock
MOVWF     ADCON1       ;
BANKSEL   TRISA        ;
BSF       TRISA,0      ;Set RA0 to input
BANKSEL   ANSEL        ;
BSF       ANSEL,0      ;Set RA0 to analog
BANKSEL   ADCON0       ;
MOVLW     B'10000001'  ;Right justify,
MOVWF     ADCON0       ;Vdd Vref, AN0, On
CALL      SampleTime   ;Acquisiton delay
BSF       ADCON0,GO    ;Start conversion
BTFSC     ADCON0,GO    ;Is conversion done?
GOTO      $-1          ;No, test again
BANKSEL   ADRESH       ;
MOVF      ADRESH,W     ;Read upper 2 bits
MOVWF     RESULTHI     ;store in GPR space
BANKSEL   ADRESL       ;
MOVF      ADRESL,W     ;Read lower 8 bits
MOVWF     RESULTLO     ;Store in GPR space
```

12.2.7 ADC REGISTER DEFINITIONS

The following registers are used to control the operation of the ADC.

REGISTER 12-1: ADCON0: A/D CONTROL REGISTER 0

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	VCFG1	VCFG0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	ADON
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **ADFM:** A/D Conversion Result Format Select bit
1 = Right justified
0 = Left justified

bit 6 **VCFG1:** Voltage Reference bit
1 = V_{REF}- pin
0 = V_{SS}

bit 5 **VCFG0:** Voltage Reference bit
1 = V_{REF}+ pin
0 = V_{SS}

bit 4-2 **CHS<2:0>:** Analog Channel Select bits
000 = AN0
001 = AN1
010 = AN2
011 = AN3
100 = AN4
101 = AN5**(1)**
110 = AN6**(1)**
111 = AN7**(1)**

bit 1 **GO/$\overline{\text{DONE}}$:** A/D Conversion Status bit
1 = A/D conversion cycle in progress. Setting this bit starts an A/D conversion cycle.
This bit is automatically cleared by hardware when the A/D conversion has completed.
0 = A/D conversion completed/not in progress

bit 0 **ADON:** ADC Enable bit
1 = ADC is enabled
0 = ADC is disabled and consumes no operating current

Note 1: Not available on 28-pin devices.

REGISTER 12-2: ADCON1: A/D CONTROL REGISTER 1

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	ADCS2	ADCS1	ADCS0	—	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **Unimplemented:** Read as '0'

bit 6-4 **ADCS<2:0>:** A/D Conversion Clock Select bits

000 = FOSC/2
001 = FOSC/8
010 = FOSC/32
x11 = FRC (clock derived from a dedicated internal oscillator = 500 kHz max.)
100 = FOSC/4
101 = FOSC/16
110 = FOSC/64

bit 3-0 **Unimplemented:** Read as '0'

REGISTER 12-3: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 0

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
ADRES9	ADRES8	ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **ADRES<9:2>**: ADC Result Register bits
Upper 8 bits of 10-bit conversion result

REGISTER 12-4: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 0

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
ADRES1	ADRES0	—	—	—	—	—	—
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-6 **ADRES<1:0>**: ADC Result Register bits
Lower 2 bits of 10-bit conversion result

bit 5-0 **Reserved**: Do not use.

REGISTER 12-5: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 1

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	—	—	—	ADRES9	ADRES8
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-2 **Reserved**: Do not use.

bit 1-0 **ADRES<9:8>**: ADC Result Register bits
Upper 2 bits of 10-bit conversion result

REGISTER 12-6: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 1

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2	ADRES1	ADRES0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **ADRES<7:0>**: ADC Result Register bits
Lower 8 bits of 10-bit conversion result

12.3 A/D Acquisition Requirements

For the ADC to meet its specified accuracy, the charge holding capacitor (CHOLD) must be allowed to fully charge to the input channel voltage level. The Analog Input model is shown in Figure 12-4. The source impedance (RS) and the internal sampling switch (RSS) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch (RSS) impedance varies over the device voltage (VDD), see Figure 12-4. **The maximum recommended impedance for analog sources is 10 kΩ.** As the source impedance is decreased, the acquisition time may be decreased. After the analog input channel is selected (or changed), an A/D acquisition must be done before the conversion can be started. To calculate the minimum acquisition time, Equation 12-1 may be used. This equation assumes that 1/2 LSb error is used (1024 steps for the ADC). The 1/2 LSb error is the maximum error allowed for the ADC to meet its specified resolution.

EQUATION 12-1: ACQUISITION TIME EXAMPLE

Assumptions: $Temperature = 50°C$ *and external impedance of* $10k\Omega$ $5.0V$ V_{DD}

$$\begin{aligned} T_{ACQ} &= \text{Amplifier Settling Time} + \text{Hold Capacitor Charging Time} + \text{Temperature Coefficient} \\ &= T_{AMP} + T_C + T_{COFF} \\ &= 2\mu s + T_C + [(\text{Temperature} - 25°C)(0.05\mu s/°C)] \end{aligned}$$

The value for T_C *can be approximated with the following equations:*

$$V_{APPLIED}\left(1 - \frac{1}{(2^{n+1}) - 1}\right) = V_{CHOLD} \quad ;[1]\ V_{CHOLD}\ \text{charged to within 1/2 lsb}$$

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{RC}}\right) = V_{CHOLD} \quad ;[2]\ V_{CHOLD}\ \text{charge response to}\ V_{APPLIED}$$

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{RC}}\right) = V_{APPLIED}\left(1 - \frac{1}{(2^{n+1}) - 1}\right) \quad ;\text{combining [1] and [2]}$$

Note: Where n = number of bits of the ADC.

Solving for T_C*:*

$$\begin{aligned} T_C &= -C_{HOLD}(R_{IC} + R_{SS} + R_S)\ ln(1/2047) \\ &= -10pF(1k\Omega + 7k\Omega + 10k\Omega)\ ln(0.0004885) \\ &= 1.37\mu s \end{aligned}$$

Therefore:

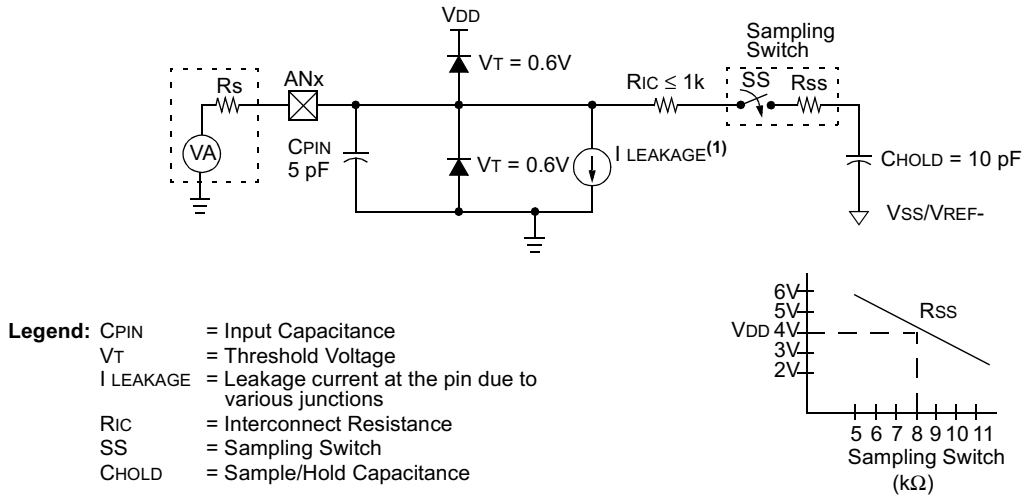
$$\begin{aligned} T_{ACQ} &= 2\mu S + 1.37\mu S + [(50°C - 25°C)(0.05\mu S/°C)] \\ &= 4.67\mu S \end{aligned}$$

Note 1: The reference voltage (VREF) has no effect on the equation, since it cancels itself out.

2: The charge holding capacitor (CHOLD) is not discharged after each conversion.

3: The maximum recommended impedance for analog sources is 10 kΩ. This is required to meet the pin leakage specification.

FIGURE 12-4: ANALOG INPUT MODEL



Legend: CPIN = Input Capacitance
VT = Threshold Voltage
I LEAKAGE = Leakage current at the pin due to various junctions
RIC = Interconnect Resistance
SS = Sampling Switch
CHOLD = Sample/Hold Capacitance

Note 1: See **Section 19.0 "Electrical Specifications"**.

FIGURE 12-5: ADC TRANSFER FUNCTION

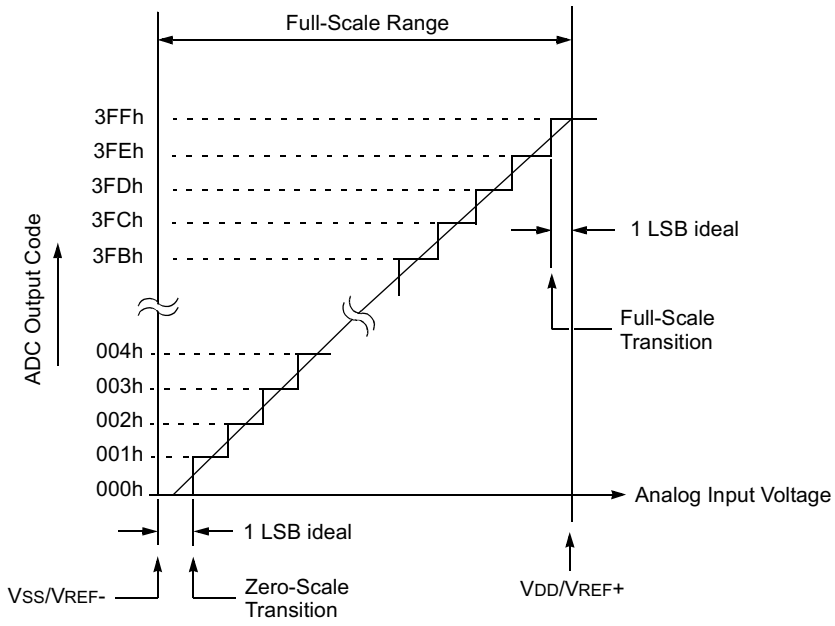


TABLE 12-2: SUMMARY OF ASSOCIATED ADC REGISTERS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
ADCON0	ADFM	VCFG1	VCFG0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	ADON	0000 0000	0000 0000
ADCON1	—	ADCS2	ADCS1	ADCS0	—	—	—	—	-000 ----	-000 ----
ANSEL	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
ADRESH	A/D Result Register High Byte								xxxx xxxx	uuuu uuuu
ADRESL	A/D Result Register Low Byte								xxxx xxxx	uuuu uuuu
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE0	SE7	SE6	SE5	SE4	SE3	SE2	SE1	SE0	0000 0000	0000 0000
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	0000 0000
LCDSE2(1)	SE23	SE22	SE21	SE20	SE19	SE18	SE17	SE16	0000 0000	0000 0000
PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx xxxx	uuuu uuuu
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
PORTE	RE7	RE6	RE5	RE4	RE3	RE2	RE1	RE0	xxxx xxxx	uuuu uuuu
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 ----	1111 ----
TRISE	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2	TRISE1	TRISE0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, – = unimplemented read as '0'. Shaded cells are not used for ADC module.

NOTES:

13.0 DATA EEPROM AND FLASH PROGRAM MEMORY CONTROL

Data EEPROM memory is readable and writable and the Flash program memory is readable during normal operation (full VDD range). These memories are not directly mapped in the register file space. Instead, they are indirectly addressed through the Special Function Registers. There are six SFRs used to access these memories:

- EECON1
- EECON2
- EEDATL
- EEDATH
- EEADRL
- EEADRH

When interfacing the data memory block, EEDATL holds the 8-bit data for read/write, and EEADRL holds the address of the EE data location being accessed. This device has 256 bytes of data EEPROM with an address range from 00h to FFh.

When interfacing the program memory block, the EEDATL and EEDATH registers form a 2-byte word that holds the 14-bit data for read, and the EEADRL and EEADRH registers form a 2-byte word that holds the 13-bit address of the EEPROM location being accessed. This family of devices has 4K and 8K words of program Flash with an address range from 0h-0FFFh and 0h-1FFFh. The program memory allows one word reads.

The EEPROM data memory allows byte read and write. A byte write automatically erases the location and writes the new data (erase before write).

The write time is controlled by an on-chip timer. The write/erase voltages are generated by an on-chip charge pump rated to operate over the voltage range of the device for byte or word operations.

When the device is code-protected, the CPU may continue to read and write the data EEPROM memory and read the program memory. When code-protected, the device programmer can no longer access data or program memory.

13.1 EEADRL and EEADRH Registers

The EEADRL and EEADRH registers can address up to a maximum of 256 bytes of data EEPROM or up to a maximum of 8K words of program Flash.

When selecting a program address value, the MSB of the address is written to the EEADRH register and the LSB is written to the EEADRL register. When selecting a data address value, only the LSB of the address is written to the EEADRL register.

13.1.1 EECON1 AND EECON2 REGISTERS

EECON1 is the control register for EE memory accesses.

Control bit EEPGD determines if the access will be a program or data memory access. When clear, as it is when reset, any subsequent operations will operate on the data memory. When set, any subsequent operations will operate on the program memory. Program memory can only be read.

Control bits RD and WR initiate read and write, respectively. These bits cannot be cleared, only set, in software. They are cleared in hardware at completion of the read or write operation. The inability to clear the WR bit in software prevents the accidental, premature termination of a write operation.

The WREN bit, when set, will allow a write operation to data EEPROM. On power-up, the WREN bit is clear. The WRERR bit is set when a write operation is interrupted by a $\overline{\text{MCLR}}$ or a WDT Time-out Reset during normal operation. In these situations, following Reset, the user can check the WRERR bit. The Data and Address registers will be cleared on the Reset. User code can then run an appropriate recovery routine.

Interrupt flag bit EEIF of the PIR1 register is set when write is complete. It must be cleared in the software.

EECON2 is not a physical register. Reading EECON2 will read all '0's. The EECON2 register is used exclusively in the data EEPROM write sequence.

REGISTER 13-1: EEDATL: EEPROM/PROGRAM MEMORY DATA LOW BYTE REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EEDATL7	EEDATL6	EEDATL5	EEDATL4	EEDATL3	EEDATL2	EEDATL1	EEDATL0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **EEDATL<7:0>**: Byte value to Write to or Read from data EEPROM bits or to Read from program memory

REGISTER 13-2: EEADRL: EEPROM/PROGRAM MEMORY ADDRESS LOW BYTE REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EEADRL7	EEADRL6	EEADRL5	EEADRL4	EEADRL3	EEADRL2	EEADRL1	EEADRL0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-0 **EEADRL<7:0>**: Specifies one of 256 locations for EEPROM Read/Write operation bits or low address byte for program memory reads

REGISTER 13-3: EEDATH: PROGRAM MEMORY DATA HIGH BYTE REGISTER

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	EEDATH5	EEDATH4	EEDATH3	EEDATH2	EEDATH1	EEDATH0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-6 **Unimplemented:** Read as '0'

bit 5-0 **EEDATH<5:0>**: Byte value to Read from program memory

REGISTER 13-4: EEADRH: PROGRAM MEMORY ADDRESS HIGH BYTE REGISTER

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	EEDATH4	EEDATH3	EEDATH2	EEDATH1	EEDATH0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-5 **Unimplemented:** Read as '0'

bit 4-0 **EEADRH<4:0>**: Specifies the high address byte for program memory reads

REGISTER 13-5: EECON1: EEPROM CONTROL REGISTER

R/W-x	U-0	U-0	U-0	R/W-x	R/W-0	R/S-0	R/S-0
EEPGD	—	—	—	WRERR	WREN	WR	RD
bit 7							bit 0

Legend:			
S = Bit can only be set			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **EEPGD:** Program/Data EEPROM Select bit
- 1 = Accesses program memory
- 0 = Accesses data memory

bit 6-4 **Unimplemented:** Read as '0'

bit 3 **WRERR:** EEPROM Error Flag bit
- 1 = A write operation is prematurely terminated (any $\overline{\text{MCLR}}$ Reset, any WDT Reset during normal operation or BOR Reset)
- 0 = The write operation completed

bit 2 **WREN:** EEPROM Write Enable bit
- 1 = Allows write cycles
- 0 = Inhibits write to the data EEPROM

bit 1 **WR**: Write Control bit

EEPGD = 1:
This bit is ignored

EEPGD = 0:
- 1 = Initiates a write cycle (The bit is cleared by hardware once write is complete. The WR bit can only be set, not cleared, in software.)
- 0 = Write cycle to the data EEPROM is complete

bit 0 **RD:** Read Control bit
- 1 = Initiates a memory read (the RD is cleared in hardware and can only be set, not cleared, in software.)
- 0 = Does not initiate a memory read

13.1.2 READING THE DATA EEPROM MEMORY

To read a data memory location, the user must write the address to the EEADRL register, clear the EEPGD control bit, and then set control bit RD of the EECON1 register. The data is available in the very next cycle, in the EEDATL register; therefore, it can be read in the next instruction. EEDATL will hold this value until another read or until it is written to by the user (during a write operation).

EXAMPLE 13-1: DATA EEPROM READ

```
BANKSEL EEADRL          ;
MOVF    DATA_EE_ADDR,W  ;Data Memory
MOVWF   EEADRL          ;Address to read
BANKSEL EECON1          ;
BCF     EECON1,EEPGD    ;Point to Data
                        ;memory
BSF     EECON1,RD       ;EE Read
BANKSEL EEDATL          ;
MOVF    EEDATL,W        ;W = EEPROM Data
```

13.1.3 WRITING TO THE DATA EEPROM MEMORY

To write an EEPROM data location, the user must first write the address to the EEADRL register and the data to the EEDATL register. Then the user must follow a specific sequence to initiate the write for each byte.

The write will not initiate if the sequence described below is not followed exactly (write 55h to EECON2, write AAh to EECON2, then set WR bit) for each byte. Interrupts should be disabled during this code segment.

Additionally, the WREN bit in EECON1 must be set to enable write. This mechanism prevents accidental writes to data EEPROM due to errant (unexpected) code execution (i.e., lost programs). The user should keep the WREN bit clear at all times, except when updating EEPROM. The WREN bit is not cleared by hardware.

After a write sequence has been initiated, clearing the WREN bit will not affect this write cycle. The WR bit will be inhibited from being set unless the WREN bit is set.

At the completion of the write cycle, the WR bit is cleared in hardware and the EE Write Complete Interrupt Flag bit (EEIF) is set. The user can either enable this interrupt or poll this bit. EEIF must be cleared by software.

The steps to write to EEPROM data memory are:

1. If step 10 is not implemented, check the WR bit to see if a write is in progress.
2. Write the address to EEADRL. Make sure that the address is not larger than the memory size of the device.
3. Write the 8-bit data value to be programmed in the EEDATL register.
4. Clear the EEPGD bit to point to EEPROM data memory.
5. Set the WREN bit to enable program operations.
6. Disable interrupts (if enabled).
7. Execute the special five instruction sequence:
   - Write 55h to EECON2 in two steps (first to W, then to EECON2)
   - Write AAh to EECON2 in two steps (first to W, then to EECON2)
   - Set the WR bit
8. Enable interrupts (if using interrupts).
9. Clear the WREN bit to disable program operations.
10. At the completion of the write cycle, the WR bit is cleared and the EEIF interrupt flag bit is set. (EEIF must be cleared by firmware.) If step 1 is not implemented, then firmware should check for EEIF to be set, or WR to clear, to indicate the end of the program cycle.

EXAMPLE 13-2: DATA EEPROM WRITE

```
        BANKSEL EECON1          ;
        BTFSC   EECON1,WR       ;Wait for write
        GOTO    $-1             ;to complete
        BANKSEL EEADRL          ;
        MOVF    DATA_EE_ADDR,W  ;Data Memory
        MOVWF   EEADRL          ;Address to write
        MOVF    DATA_EE_DATA,W  ;Data Memory Value
        MOVWF   EEDATL          ;to write
        BANKSEL EECON1          ;
        BCF     EECON1,EEPGD    ;Point to DATA
                                ;memory
        BSF     EECON1,WREN     ;Enable writes

        BCF     INTCON,GIE      ;Disable INTs.
Required
Sequence
        MOVLW   55h             ;
        MOVWF   EECON2          ;Write 55h
        MOVLW   AAh             ;
        MOVWF   EECON2          ;Write AAh
        BSF     EECON1,WR       ;Set WR bit to
                                ;begin write
        BSF     INTCON,GIE      ;Enable INTs.
        BCF     EECON1,WREN     ;Disable writes
```

13.1.4 READING THE FLASH PROGRAM MEMORY

To read a program memory location, the user must write two bytes of the address to the EEADRL and EEADRH registers, set the EEPGD control bit, and then set control bit RD of the EECON1 register. Once the read control bit is set, the program memory Flash controller will use the second instruction cycle to read the data. This causes the second instruction immediately following the "BSF EECON1,RD" instruction to be ignored. The data is available in the very next cycle, in the EEDATL and EEDATH registers; therefore, it can be read as two bytes in the following instructions. EEDATL and EEDATH registers will hold this value until another read or until it is written to by the user (during a write operation).

Note 1: The two instructions following a program memory read are required to be NOP's. This prevents the user from executing a two-cycle instruction on the next instruction after the RD bit is set.

2: If the WR bit is set when EEPGD = 1, the WR bit will be immediately reset to '0' and no operation will take place.

EXAMPLE 13-3: FLASH PROGRAM READ

```
        BANKSEL  EEADRL          ;
        MOVLW    MS_PROG_EE_ADDR;
        MOVWF    EEADRH          ;MS Byte of Program Address to read
        MOVLW    LS_PROG_EE_ADDR;
        MOVWF    EEADRL          ;LS Byte of Program Address to read
        BANKSEL  EECON1          ;
        BSF      EECON1, EEPGD   ;Point to PROGRAM memory
        BSF      EECON1, RD      ;EE Read
;
        NOP
        NOP                      ;Any instructions here are ignored as program
                                 ;memory is read in second cycle after BSF
;
        BANKSEL  EEDATL          ;
        MOVF     EEDATL, W       ;W = LS Byte of EEPROM Data program
        MOVWF    DATAL           ;
        MOVF     EEDATH, W       ;W = MS Byte of EEPROM Data program
        MOVWF    DATAH           ;
```

Required Sequence

FIGURE 13-1: FLASH PROGRAM MEMORY READ CYCLE EXECUTION

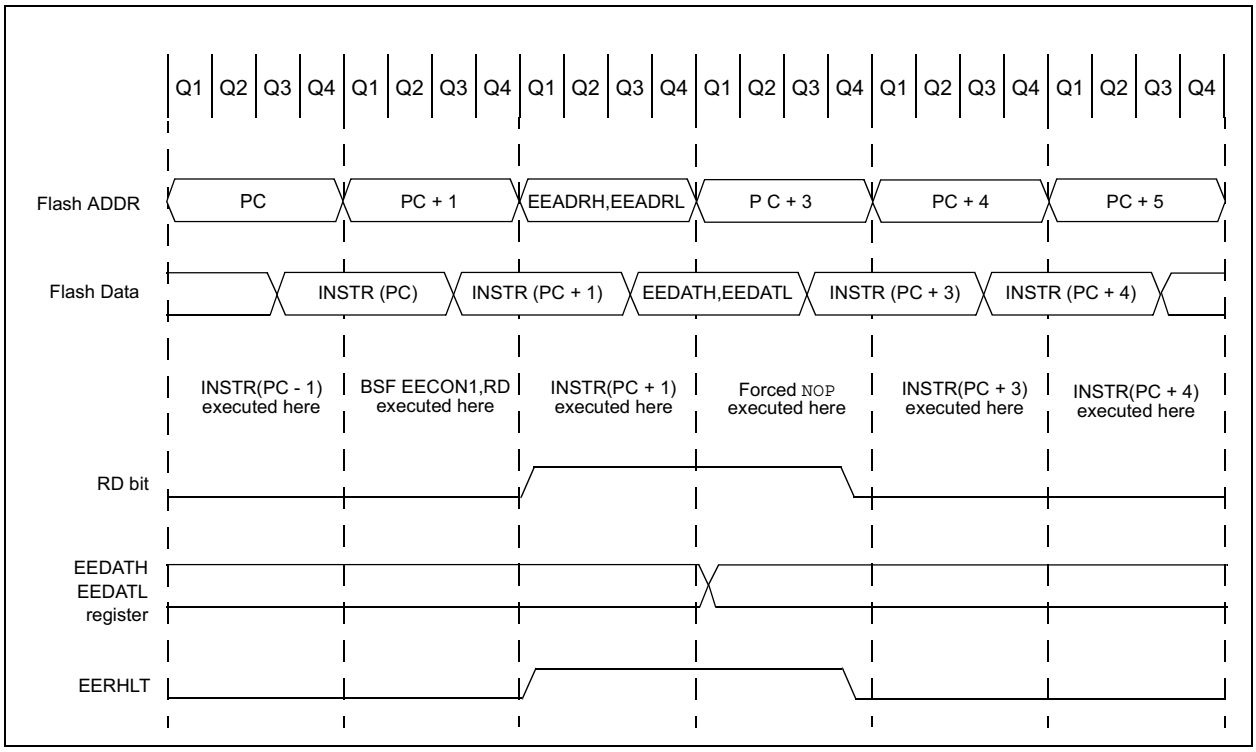


TABLE 13-1: SUMMARY OF ASSOCIATED REGISTERS WITH DATA EEPROM

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
EEADRH	—	—	—	EEADRH4	EEADRH3	EEADRH2	EEADRH1	EEADRH0	---0 0000	---0 0000
EEADRL	EEADRL7	EEADRL6	EEADRL5	EEADRL4	EEADRL3	EEADRL2	EEADRL1	EEADRL0	0000 0000	0000 0000
EECON1	EEPGD	—	—	—	WRERR	WREN	WR	RD	0--- x000	---- q000
EECON2	EEPROM Control Register 2 (not a physical register)								---- ----	---- ----
EEDATH	—	—	EEDATH5	EEDATH4	EEDATH3	EEDATH2	EEDATH1	EEDATH0	--00 0000	--00 0000
EEDATL	EEDATL7	EEDATL6	EEDATL5	EEDATL4	EEDATL3	EEDATL2	EEDATL1	EEDATL0	0000 0000	0000 0000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0', q = value depends upon condition.
Shaded cells are not used by data EEPROM module.

14.0 SSP MODULE OVERVIEW

The Synchronous Serial Port (SSP) module is a serial interface used to communicate with other peripheral or microcontroller devices. These peripheral devices may be serial EEPROMs, shift registers, display drivers, A/D converters, etc. The SSP module can operate in one of two modes:

- Serial Peripheral Interface (SPI)
- Inter-Integrated Circuit (I^2C™)

Refer to Application Note AN578, "*Use of the SSP Module in the Multi-Master Environment*" (DS00578).

14.1 SPI Mode

This section contains register definitions and operational characteristics of the SPI module.

The SPI mode allows 8 bits of data to be synchronously transmitted and received simultaneously. To accomplish communication, typically three pins are used:

- Serial Data Out (SDO)
- Serial Data In (SDI)
- Serial Clock (SCK)

Additionally, a fourth pin may be used when in a Slave mode of operation:

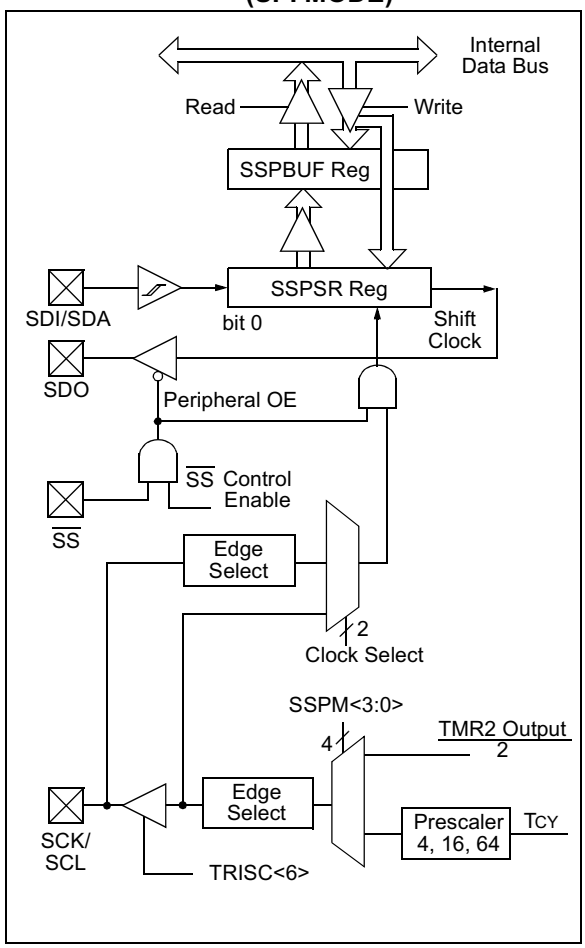
- Slave Select ($\overline{SS}$)

Note 1: When the SPI is in Slave mode with $\overline{SS}$ pin control enabled (SSPM<3:0> bits of the SSPCON register = 0100), the SPI module will reset if the $\overline{SS}$ pin is set to V_{DD}.

2: If the SPI is used in Slave mode with CKE = 1, then the $\overline{SS}$ pin control must be enabled.

3: When the SPI is in Slave mode with $\overline{SS}$ pin control enabled (SSPM<3:0> bits of the SSPCON register = 0100), the state of the $\overline{SS}$ pin can affect the state read back from the TRISC<4> bit. The peripheral OE signal from the SSP module into PORTC controls the state that is read back from the TRISC<4> bit (see **Section 19.0 "Electrical Specifications"** for information on PORTC). If read-write-modify instructions, such as BSF, are performed on the TRISC register while the $\overline{SS}$ pin is high, this will cause the TRISC<4> bit to be set, thus disabling the SDO output.

FIGURE 14-1: SSP BLOCK DIAGRAM (SPI MODE)



REGISTER 14-1: SSPSTAT: SYNC SERIAL PORT STATUS REGISTER

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **SMP**: SPI Data Input Sample Phase bit
SPI Master mode:
1 = Input data sampled at end of data output time
0 = Input data sampled at middle of data output time (Microwire)
SPI Slave mode:
SMP must be cleared when SPI is used in Slave mode
I^2C™ mode:
This bit must be maintained clear

bit 6 **CKE**: SPI Clock Edge Select bit
SPI mode, CKP = 0:
1 = Data stable on rising edge of SCK (Microwire alternate)
0 = Data stable on falling edge of SCK
SPI mode, CKP = 1:
1 = Data stable on falling edge of SCK (Microwire default)
0 = Data stable on rising edge of SCK
I^2C mode:
This bit must be maintained clear

bit 5 **D/$\overline{A}$**: DATA/$\overline{ADDRESS}$ bit (I^2C mode only)
1 = Indicates that the last byte received or transmitted was data
0 = Indicates that the last byte received or transmitted was address

bit 4 **P**: Stop bit (I^2C mode only)
This bit is cleared when the SSP module is disabled, or when the Start bit is detected last.
SSPEN is cleared.
1 = Indicates that a Stop bit has been detected last (this bit is '0' on Reset)
0 = Stop bit was not detected last

bit 3 **S**: Start bit (I^2C mode only)
This bit is cleared when the SSP module is disabled, or when the Stop bit is detected last.
SSPEN is cleared.
1 = Indicates that a Start bit has been detected last (this bit is '0' on Reset)
0 = Start bit was not detected last

bit 2 **R/$\overline{W}$**: READ/$\overline{WRITE}$ bit Information (I^2C mode only)
This bit holds the R/$\overline{W}$ bit information following the last address match. This bit is only valid from the address match to the next Start bit, Stop bit or $\overline{ACK}$ bit.
1 = Read
0 = Write

bit 1 **UA**: Update Address bit (10-bit I^2C mode only)
1 = Indicates that the user needs to update the address in the SSPADD register
0 = Address does not need to be updated

bit 0 **BF**: Buffer Full Status bit
Receive (SPI and I^2C modes):
1 = Receive complete, SSPBUF is full
0 = Receive not complete, SSPBUF is empty
Transmit (I^2C mode only):
1 = Transmit in progress, SSPBUF is full
0 = Transmit complete, SSPBUF is empty

REGISTER 14-2: SSPCON: SYNC SERIAL PORT CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV	SSPEN	CKP	SSPM3(2)	SSPM2(2)	SSPM1(2)	SSPM0(2)
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7 **WCOL**: Write Collision Detect bit
1 = The SSPBUF register is written while it is still transmitting the previous word (must be cleared in software)
0 = No collision

bit 6 **SSPOV**: Receive Overflow Indicator bit
In SPI mode:
1 = A new byte is received while the SSPBUF register is still holding the previous data. In case of overflow, the data in SSPSR is lost. Overflow can only occur in Slave mode. The user must read the SSPBUF, even if only transmitting data, to avoid setting overflow. In Master mode, the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPBUF register.
0 = No overflow
In I^2C™ mode:
1 = A byte is received while the SSPBUF register is still holding the previous byte. SSPOV is a "don't care" in Transmit mode. SSPOV must be cleared in software in either mode.
0 = No overflow

bit 5 **SSPEN**: Synchronous Serial Port Enable bit
In SPI mode:
1 = Enables serial port and configures SCK, SDO and SDI as serial port pins
0 = Disables serial port and configures these pins as I/O port pins
In I^2C mode:
1 = Enables the serial port and configures the SDA and SCL pins as serial port pins
0 = Disables serial port and configures these pins as I/O port pins
In both modes, when enabled, these pins must be properly configured as input or output.

bit 4 **CKP**: Clock Polarity Select bit
In SPI mode:
1 = Idle state for clock is a high level (Microwire default)
0 = Idle state for clock is a low level (Microwire alternate)
In I^2C mode:
SCK release control
1 = Enable clock
0 = Holds clock low (clock stretch). (Used to ensure data setup time.)

bit 3-0 **SSPM<3:0>**: Synchronous Serial Port Mode Select bits
0000 = SPI Master mode, clock = FOSC/4
0001 = SPI Master mode, clock = FOSC/16
0010 = SPI Master mode, clock = FOSC/64
0011 = SPI Master mode, clock = TMR2 output/2
0100 = SPI Slave mode, clock = SCK pin. $\overline{SS}$ pin control enabled.
0101 = SPI Slave mode, clock = SCK pin. $\overline{SS}$ pin control disabled. $\overline{SS}$ can be used as I/O pin.
0110 = I^2C Slave mode, 7-bit address
0111 = I^2C Slave mode, 10-bit address
1000 = Reserved
1001 = Reserved
1010 = Reserved
1011 = I^2C Firmware Controlled Master mode (slave IDLE)
1100 = Reserved
1101 = Reserved
1110 = I^2C Slave mode, 7-bit address with Start and Stop bit interrupts enabled
1111 = I^2C Slave mode, 10-bit address with Start and Stop bit interrupts enabled

14.2 Operation

When initializing the SPI, several options need to be specified. This is done by programming the appropriate control bits (SSPCON<5:0> and SSPSTAT<7:6>). These control bits allow the following to be specified:

- Master mode (SCK is the clock output)
- Slave mode (SCK is the clock input)
- Clock Polarity (Idle state of SCK)
- Data Input Sample Phase (middle or end of data output time)
- Clock Edge (output data on rising/falling edge of SCK)
- Clock Rate (Master mode only)
- Slave Select mode (Slave mode only)

The SSP consists of a transmit/receive shift register (SSPSR) and a buffer register (SSPBUF). The SSPSR shifts the data in and out of the device, MSb first. The SSPBUF holds the data that was written to the SSPSR until the received data is ready. Once the eight bits of data have been received, that byte is moved to the SSPBUF register. Then, the Buffer Full Status bit BF of the SSPSTAT register, and the interrupt flag bit SSPIF, are set. Any write to the SSPBUF register during transmission/reception of data will be ignored and the Write Collision Detect bit, WCOL of the SSPCON register, will be set. User software must clear the WCOL bit so that it can be determined if the following write(s) to the SSPBUF register completed successfully.

When the application software is expecting to receive valid data, the SSPBUF should be read before the next byte of data to transfer is written to the SSPBUF. Buffer Full bit BF of the SSPSTAT register indicates when SSPBUF has been loaded with the received data (transmission is complete). When the SSPBUF is read, the BF bit is cleared. This data may be irrelevant if the SPI is only a transmitter. Generally, the SSP interrupt is used to determine when the transmission/reception has completed. The SSPBUF must be read and/or written. If the interrupt method is not going to be used, then software polling can be done to ensure that a write collision does not occur. Example 14-1 shows the loading of the SSPBUF (SSPSR) for data transmission.

The SSPSR is not directly readable or writable and can only be accessed by addressing the SSPBUF register. Additionally, the SSP STATUS register (SSPSTAT) indicates the various status conditions.

EXAMPLE 14-1: LOADING THE SSPBUF (SSPSR) REGISTER

```
        BANKSEL   SSPSTAT          ;
LOOP    BTFSS     SSPSTAT, BF      ;Has data been received(transmit complete)?
        GOTO      LOOP             ;No
        BANKSEL   SSPBUF           ;
        MOVF      SSPBUF, W        ;WREG reg = contents of SSPBUF
        MOVWF     RXDATA           ;Save in user RAM, if data is meaningful
        MOVF      TXDATA, W        ;W reg = contents of TXDATA
        MOVWF     SSPBUF           ;New data to xmit
```

14.3 Enabling SPI I/O

To enable the serial port, SSP Enable bit SSPEN of the SSPCON register must be set. To reset or reconfigure SPI mode, clear the SSPEN bit, re-initialize the SSPCON registers and then set the SSPEN bit. This configures the SDI, SDO, SCK and $\overline{SS}$ pins as serial port pins. For the pins to behave as the serial port function, their data direction bits (in the TRISA and TRISC registers) should be set as follows:

- TRISC<7> bit must be set
- SDI is automatically controlled by the SPI module
- SDO must have TRISC<4> bit cleared
- SCK (Master mode) must have TRISC<6> bit cleared
- SCK (Slave mode) must have TRISC<6> bit set
- If enabled, $\overline{SS}$ must have TRISA<5> bit set

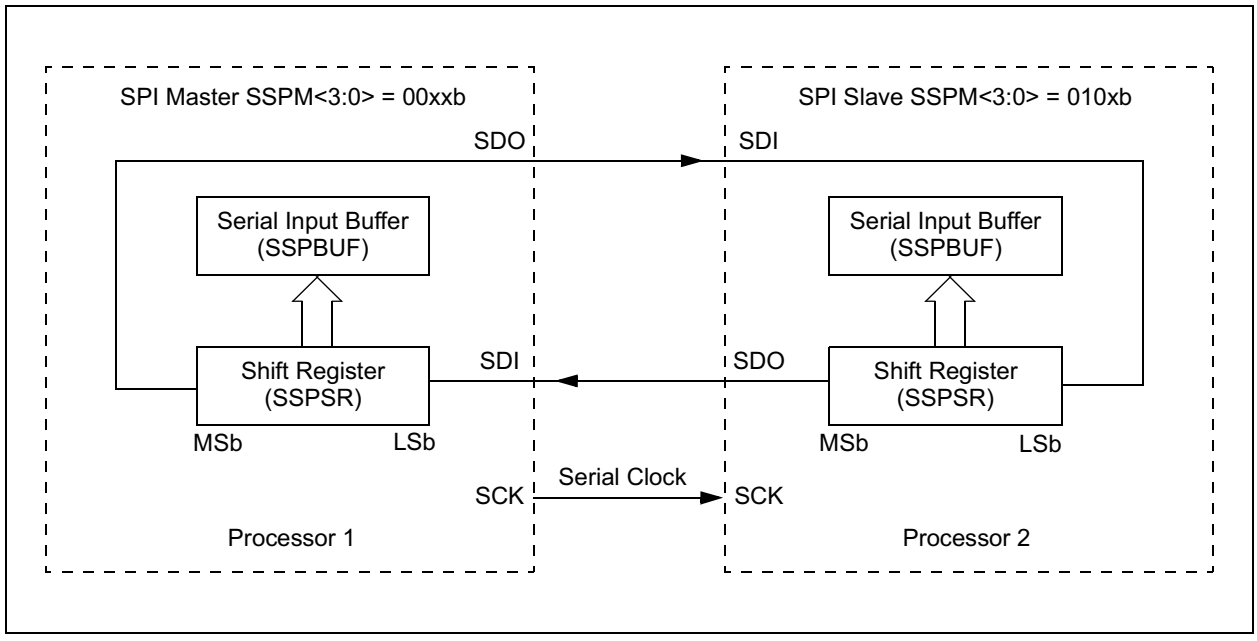
Any serial port function that is not desired may be overridden by programming the corresponding data direction (TRISA and TRISC) registers to the opposite value.

14.4 Typical Connection

Figure 14-2 shows a typical connection between two microcontrollers. The master controller (Processor 1) initiates the data transfer by sending the SCK signal. Data is shifted out of both shift registers on their programmed clock edge and latched on the opposite edge of the clock. Both processors should be programmed to the same Clock Polarity (CKP), then both controllers would send and receive data at the same time. Whether the data is meaningful (or dummy data) depends on the application software. This leads to three scenarios for data transmission:

- Master sends data – Slave sends dummy data
- Master sends data – Slave sends data
- Master sends dummy data – Slave sends data

FIGURE 14-2: SPI MASTER/SLAVE CONNECTION



14.5 Master Mode

The master can initiate the data transfer at any time because it controls the SCK. The master determines when the slave (Processor 2, Figure 14-2) is to broadcast data by the software protocol.

In Master mode, the data is transmitted/received as soon as the SSPBUF register is written to. If the SPI is only going to receive, the SDO output could be disabled (programmed as an input). The SSPSR register will continue to shift in the signal present on the SDI pin at the programmed clock rate. As each byte is received, it will be loaded into the SSPBUF register as if a normal received byte (interrupts and Status bits appropriately set). This could be useful in receiver applications as a Line Activity Monitor mode.

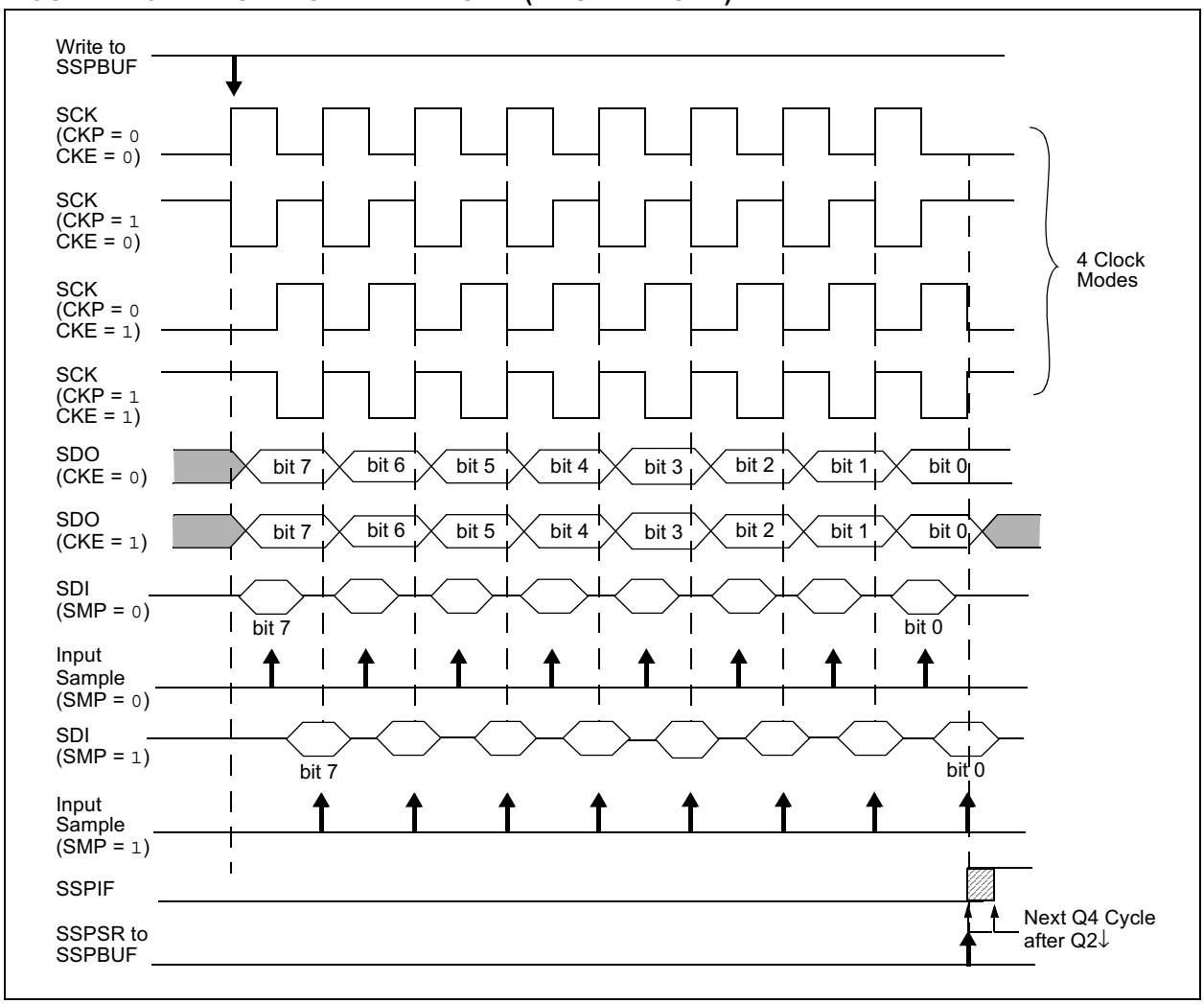
The clock polarity is selected by appropriately programming the CKP bit of the SSPCON register. This then, would give waveforms for SPI communication as shown in Figure 14-3, Figure 14-5 and Figure 14-6, where the MSB is transmitted first. In Master mode, the SPI clock rate (bit rate) is user programmable to be one of the following:

- FOSC/4 (or TCY)
- FOSC/16 (or 4 • TCY)
- FOSC/64 (or 16 • TCY)
- Timer2 output/2

This allows a maximum data rate (at 20 MHz) of 5 Mbps.

Figure 14-3 shows the waveforms for Master mode. When the CKE bit is set, the SDO data is valid before there is a clock edge on SCK. The change of the input sample is shown based on the state of the SMP bit. The time when the SSPBUF is loaded with the received data is shown.

FIGURE 14-3: SPI MODE WAVEFORM (MASTER MODE)



14.6 Slave Mode

In Slave mode, the data is transmitted and received as the external clock pulses appear on SCK. When the last bit is latched, the SSPIF interrupt flag bit is set.

While in Slave mode, the external clock is supplied by the external clock source on the SCK pin. This external clock must meet the minimum high and low times as specified in the electrical specifications.

While in Sleep mode, the slave can transmit/receive data. When a byte is received, the device will wake-up from Sleep.

14.7 Slave Select Synchronization

The $\overline{SS}$ pin allows a Synchronous Slave mode. The SPI must be in Slave mode with $\overline{SS}$ pin control enabled (SSPCON<3:0> = 0100). The pin must not be driven low for the $\overline{SS}$ pin to function as an input. The data latch must be high. When the $\overline{SS}$ pin is low, transmission and reception are enabled and the SDO pin is driven. When the $\overline{SS}$ pin goes high, the SDO pin is no longer driven, even if in the middle of a transmitted byte, and becomes a floating output. External pull-up/pull-down resistors may be desirable, depending on the application.

Note 1:	When the SPI is in Slave mode with $\overline{SS}$ pin control enabled (SSPCON<3:0> = 0100), the SPI module will reset if the $\overline{SS}$ pin is set to V_{DD}.
2:	If the SPI is used in Slave Mode with CKE set, then the $\overline{SS}$ pin control must be enabled.

When the SPI module resets, the bit counter is forced to 0. This can be done by either forcing the $\overline{SS}$ pin to a high level or clearing the SSPEN bit.

To emulate two-wire communication, the SDO pin can be connected to the SDI pin. When the SPI needs to operate as a receiver, the SDO pin can be configured as an input. This disables transmissions from the SDO. The SDI can always be left as an input (SDI function) since it cannot create a bus conflict.

FIGURE 14-4: SLAVE SYNCHRONIZATION WAVEFORM

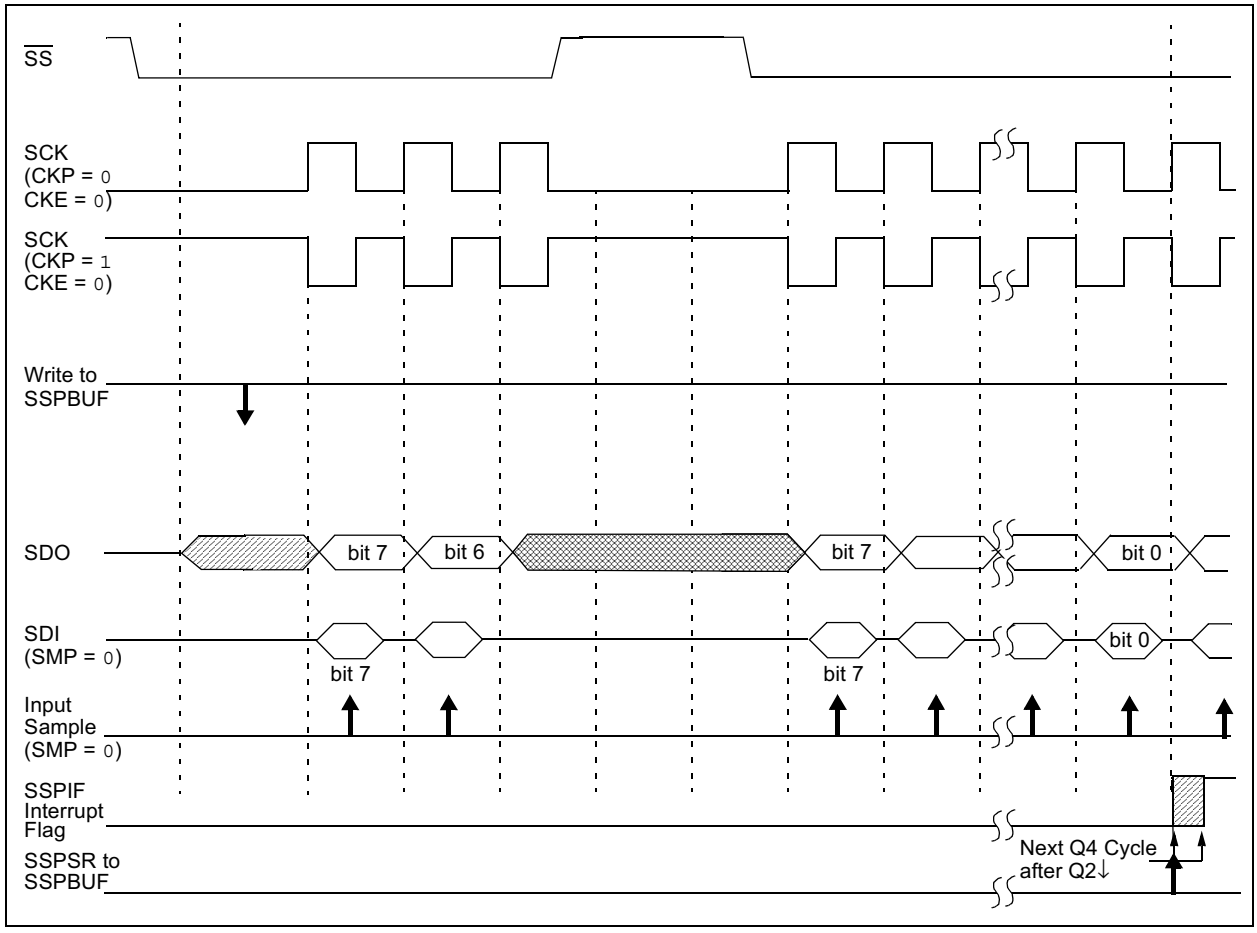


FIGURE 14-5: SPI MODE WAVEFORM (SLAVE MODE WITH CKE = 0)

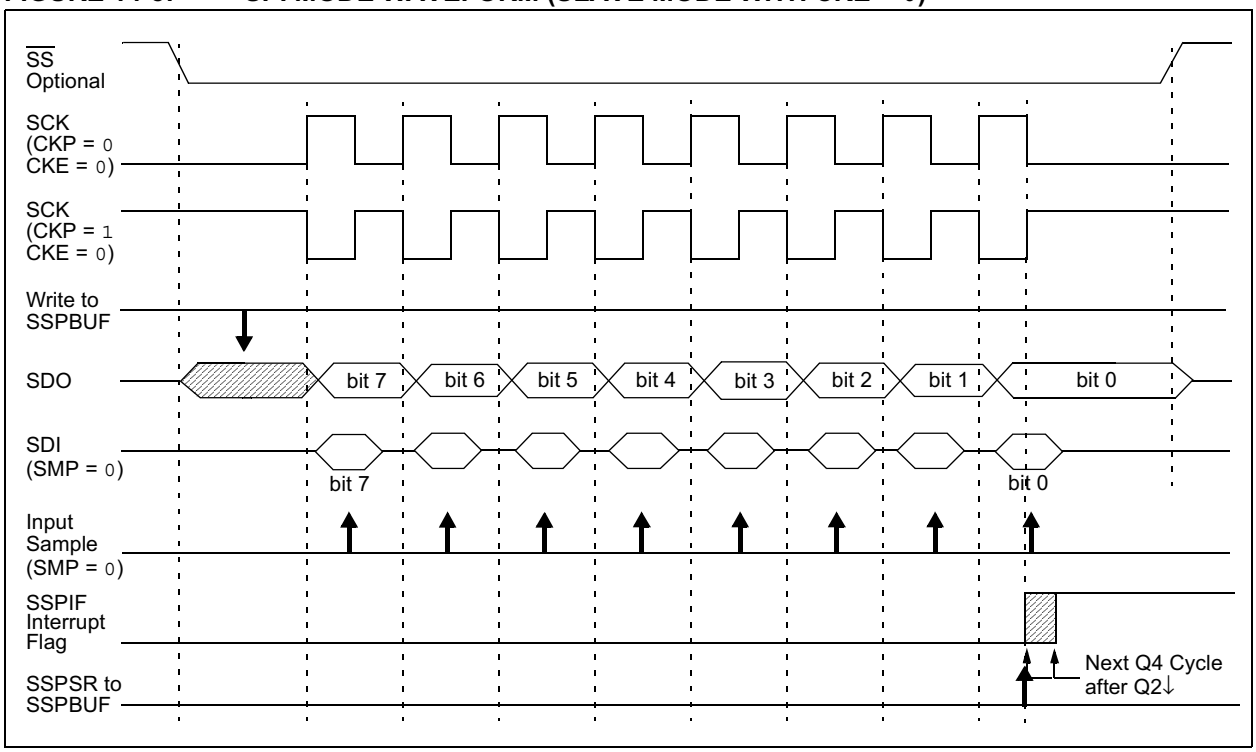
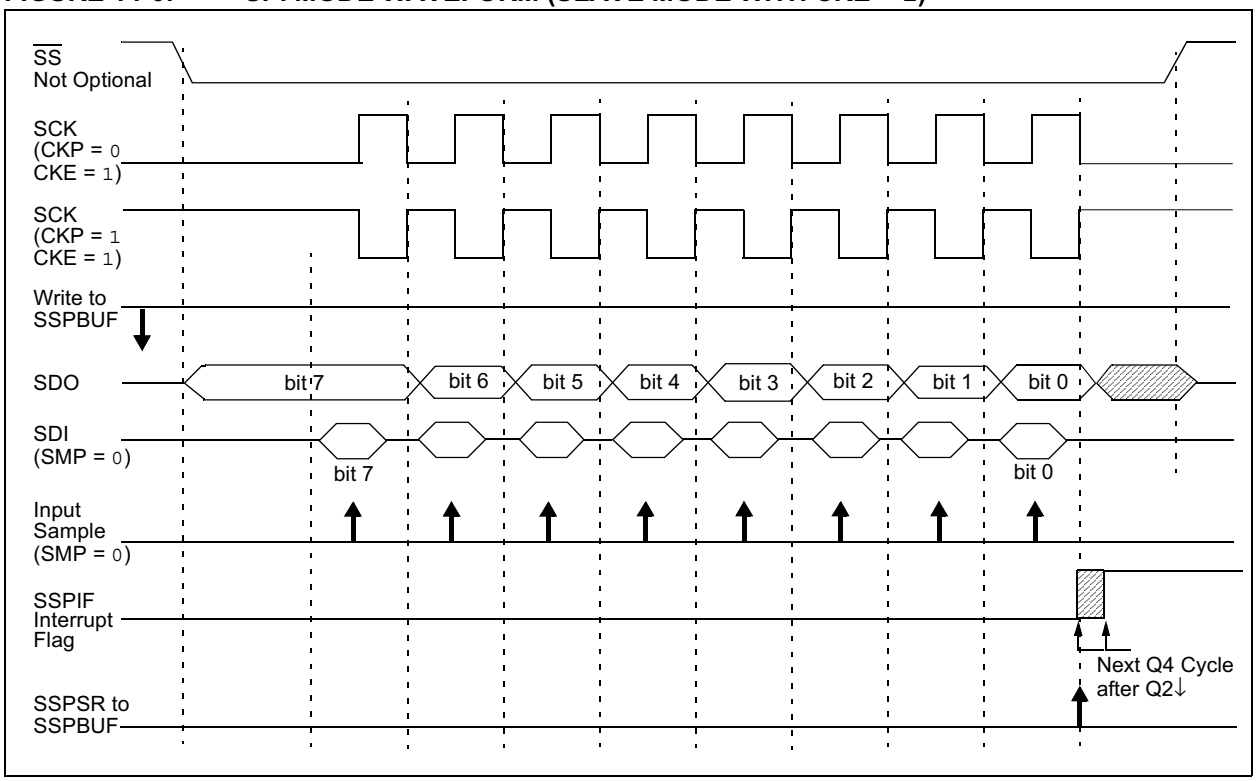


FIGURE 14-6: SPI MODE WAVEFORM (SLAVE MODE WITH CKE = 1)



14.8 Sleep Operation

In Master mode, all module clocks are halted and the transmission/reception will remain in that state until the device wakes from Sleep. After the device returns to Normal mode, the module will continue to transmit/receive data.

In Slave mode, the SPI Transmit/Receive Shift register operates asynchronously to the device. This allows the device to be placed in Sleep mode and data to be shifted into the SPI Transmit/Receive Shift register. When all 8 bits have been received, the SSP interrupt flag bit will be set and if enabled, will wake the device from Sleep.

14.9 Effects of a Reset

A Reset disables the SSP module and terminates the current transfer.

14.10 Bus Mode Compatibility

Table 14-1 shows the compatibility between the standard SPI modes and the states of the CKP and CKE control bits.

TABLE 14-1: SPI BUS MODES

Standard SPI Mode Terminology	Control Bits State	
	CKP	CKE
0, 0	0	1
0, 1	0	0
1, 0	1	1
1, 1	1	0

There is also a SMP bit which controls when the data is sampled.

TABLE 14-2: SUMMARY OF REGISTERS ASSOCIATED WITH SPI OPERATION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE0	SE7	SE6	SE5	SE4	SE3	SE2	SE1	SE0	0000 0000	0000 0000
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	0000 0000
PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
SSPSTAT	SMP	CKE	D/$\overline{\text{A}}$	P	S	R/$\overline{\text{W}}$	UA	BF	0000 0000	0000 0000
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, – = unimplemented, read as '0'. Shaded cells are not used by the SSP in SPI mode.

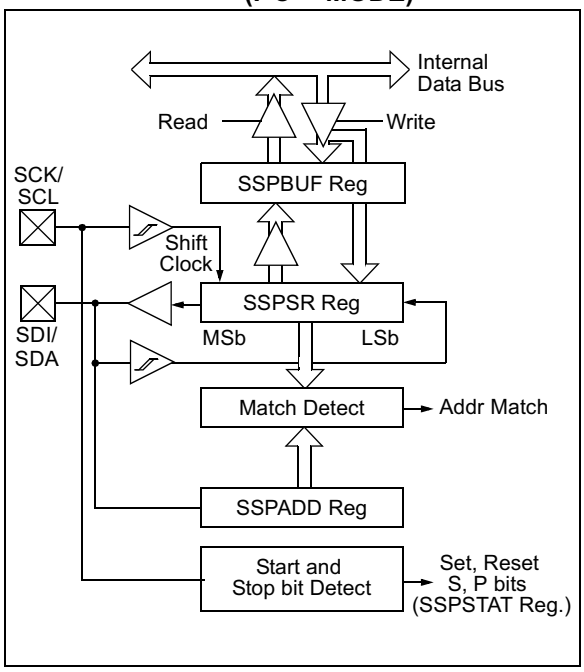
14.11 SSP I²C Operation

The SSP module in I²C mode, fully implements all slave functions, except general call support, and provides interrupts on Start and Stop bits in hardware to facilitate firmware implementations of the master functions. The SSP module implements the Standard mode specifications, as well as 7-bit and 10-bit addressing.

Two pins are used for data transfer. These are the RC6/TX/CK/SCK/SCL/SEG9 pin, which is the clock (SCL), and the RC7/RX/DT/SDI/SDA/SEG8 pin, which is the data (SDA).

The SSP module functions are enabled by setting SSP enable bit SSPEN (SSPCON<5>).

FIGURE 14-7: SSP BLOCK DIAGRAM (I²C™ MODE)



The SSP module has five registers for the I²C operation, which are listed below.

- SSP Control register (SSPCON)
- SSP STATUS register (SSPSTAT)
- Serial Receive/Transmit Buffer (SSPBUF)
- SSP Shift register (SSPSR) – Not directly accessible
- SSP Address register (SSPADD)

The SSPCON register allows control of the I²C operation. Four mode selection bits (SSPCON<3:0>) allow one of the following I²C modes to be selected:

- I²C Slave mode (7-bit address)
- I²C Slave mode (10-bit address)
- I²C Slave mode (7-bit address), with Start and Stop bit interrupts enabled to support Firmware Master mode
- I²C Slave mode (10-bit address), with Start and Stop bit interrupts enabled to support Firmware Master mode
- I²C Start and Stop bit interrupts enabled to support Firmware Master mode; Slave is idle

Selection of any I²C mode with the SSPEN bit set forces the SCL and SDA pins to be open drain, provided these pins are programmed to inputs by setting the appropriate TRISC bits. Pull-up resistors must be provided externally to the SCL and SDA pins for proper operation of the I²C module.

14.12 Slave Mode

In Slave mode, the SCL and SDA pins must be configured as inputs (TRISC<7,6> are set). The SSP module will override the input state with the output data when required (slave-transmitter).

When an address is matched, or the data transfer after an address match is received, the hardware automatically will generate the Acknowledge ($\overline{ACK}$) pulse, and then load the SSPBUF register with the received value currently in the SSPSR register.

There are certain conditions that will cause the SSP module not to give this $\overline{ACK}$ pulse. They include (either or both):

a) The Buffer Full bit BF of the SSPSTAT register was set before the transfer was received.

b) The overflow bit SSPOV of the SSPCON register was set before the transfer was received.

In this case, the SSPSR register value is not loaded into the SSPBUF, but bit SSPIF of the PIR1 register is set. Table 14-3 shows the results of when a data transfer byte is received, given the status of bits BF and SSPOV. The shaded cells show the condition where user software did not properly clear the overflow condition. Flag bit BF is cleared by reading the SSPBUF register, while bit SSPOV is cleared through software.

The SCL clock input must have a minimum high and low for proper operation. For high and low times of the I²C specification, as well as the requirements of the SSP module, see **Section 19.0 "Electrical Specifications"**.

14.12.1 ADDRESSING

Once the SSP module has been enabled, it waits for a Start condition to occur. Following the Start condition, the 8-bits are shifted into the SSPSR register. All incoming bits are sampled with the rising edge of the clock (SCL) line. The value of register SSPSR<7:1> is compared to the value of register SSPADD <7:1>. The address is compared on the falling edge of the eighth clock (SCL) pulse. If the addresses match, and the BF and SSPOV bits are clear, the following events occur:

a) The SSPSR register value is loaded into the SSPBUF register.
b) The buffer full bit, BF is set.
c) An $\overline{ACK}$ pulse is generated.
d) SSP interrupt flag bit, SSPIF of the PIR1 register is set (interrupt is generated if enabled) on the falling edge of the ninth SCL pulse.

In 10-bit Address mode, two address bytes need to be received by the slave (Figure 14-8). The five Most Significant bits (MSbs) of the first address byte specify if this is a 10-bit address. Bit R/$\overline{W}$ (SSPSTAT<2>) must specify a write so the slave device will receive the second address byte. For a 10-bit address, the first byte would equal '1111 0 A9 A8 0', where A9 and A8 are the two MSbs of the address.

The sequence of events for 10-bit address is as follows, with steps 7-9 for slave-transmitter:

1. Receive first (high) byte of address (bits SSPIF, BF and bit UA (SSPSTAT<1>) are set).
2. Update the SSPADD register with second (low) byte of address (clears bit UA and releases the SCL line).
3. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.
4. Receive second (low) byte of address (bits SSPIF, BF and UA are set).
5. Update the SSPADD register with the first (high) byte of address; if match releases SCL line, this will clear bit UA.
6. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.
7. Receive repeated Start condition.
8. Receive first (high) byte of address (bits SSPIF and BF are set).
9. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.

TABLE 14-3: DATA TRANSFER RECEIVED BYTE ACTIONS

Status Bits as Data Transfer is Received		SSPSR → SSPBUF	Generate $\overline{ACK}$ Pulse	Set bit SSPIF (SSP Interrupt occurs if enabled)
BF	SSPOV			
0	0	Yes	Yes	Yes
1	0	No	No	Yes
1	1	No	No	Yes
0	1	No	No	Yes

Note: Shaded cells show the conditions where the user software did not properly clear the overflow condition.

14.12.2 RECEPTION

When the R/$\overline{W}$ bit of the address byte is clear and an address match occurs, the R/$\overline{W}$ bit of the SSPSTAT register is cleared. The received address is loaded into the SSPBUF register.

When the address byte overflow condition exists, then no Acknowledge ($\overline{ACK}$) pulse is given. An overflow condition is defined as either bit BF of the SSPSTAT register is set, or bit SSPOV of the SSPCON register is set. This is an error condition due to the user's firmware.

An SSP interrupt is generated for each data transfer byte. Flag bit SSPIF of the PIR1 register must be cleared in software. The SSPSTAT register is used to determine the status of the byte.

FIGURE 14-8: I²C™ WAVEFORMS FOR RECEPTION (7-BIT ADDRESS)

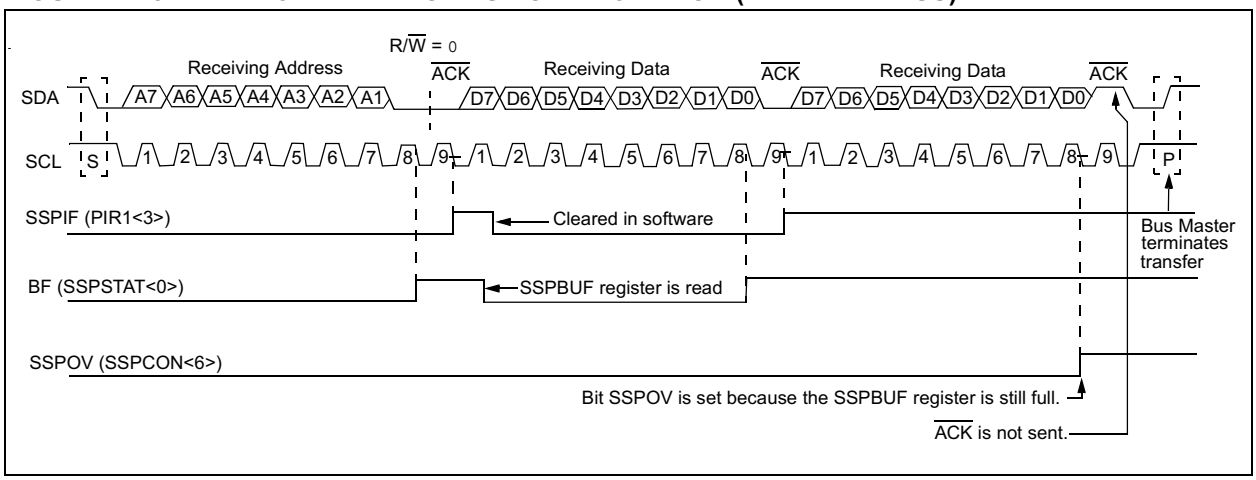
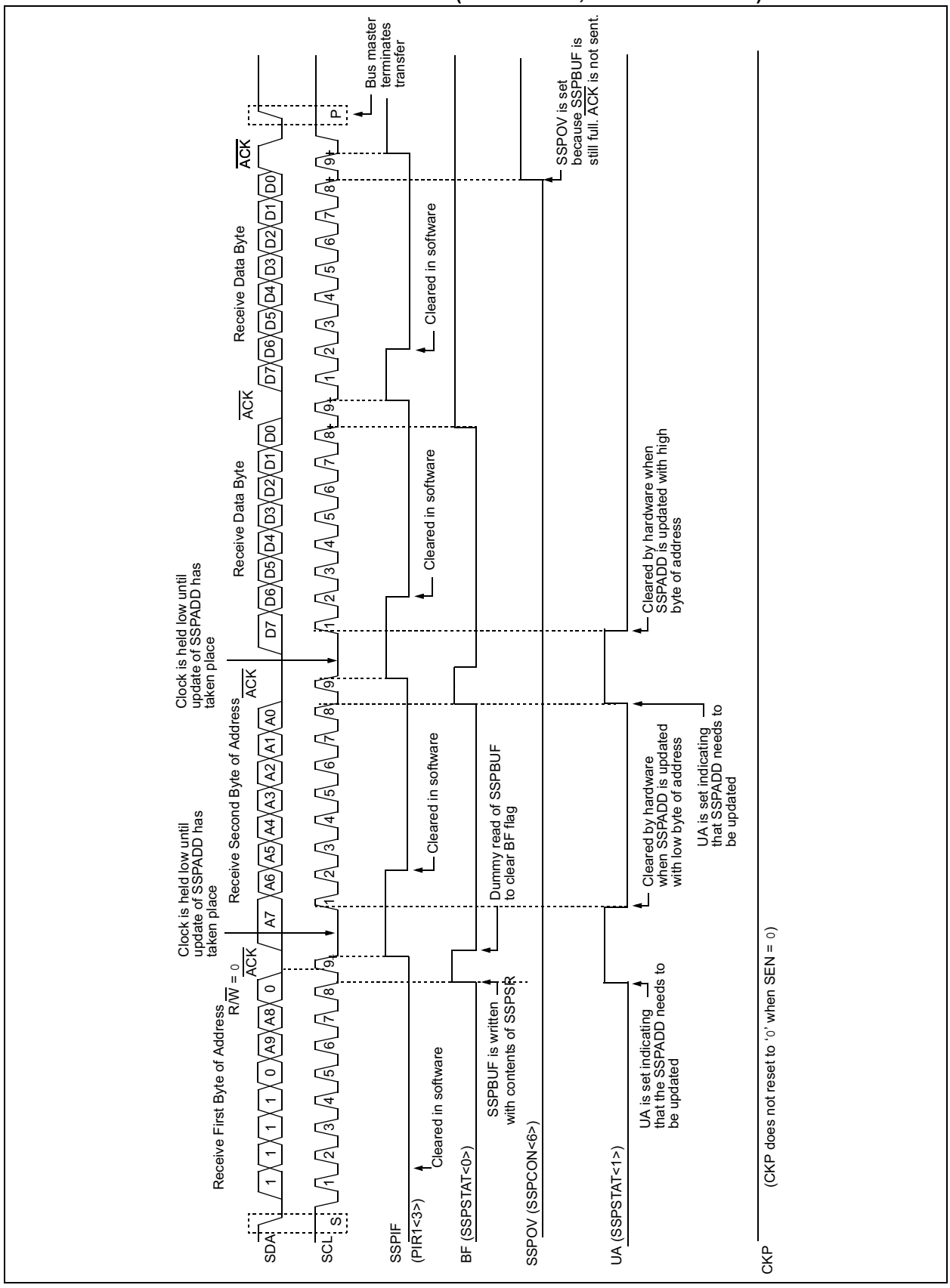


FIGURE 14-9: I²C™ SLAVE MODE TIMING (RECEPTION, 10-BIT ADDRESS)



14.12.3 TRANSMISSION

When the R/$\overline{W}$ bit of the incoming address byte is set and an address match occurs, the R/$\overline{W}$ bit of the SSPSTAT register is set. The received address is loaded into the SSPBUF register. The $\overline{ACK}$ pulse will be sent on the ninth bit, and pin RC6/TX/CK/SCK/SCL/SEG9 is held low. The transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. Then, pin RC6/TX/CK/SCK/SCL/SEG9 should be enabled by setting bit CKP of the SSPCON register. The master must monitor the SCL pin prior to asserting another clock pulse. The slave devices may be holding off the master by stretching the clock. The eight data bits are shifted out on the falling edge of the SCL input. This ensures that the SDA signal is valid during the SCL high time (Figure 14-10).

An SSP interrupt is generated for each data transfer byte. Flag bit SSPIF must be cleared in software, and the SSPSTAT register is used to determine the status of the byte. Flag bit SSPIF is set on the falling edge of the ninth clock pulse.

As a slave-transmitter, the $\overline{ACK}$ pulse from the master receiver is latched on the rising edge of the ninth SCL input pulse. If the SDA line was high (not $\overline{ACK}$), then the data transfer is complete. When the $\overline{ACK}$ is latched by the slave, the slave logic is reset (resets SSPSTAT register) and the slave then monitors for another occurrence of the Start bit. If the SDA line was low ($\overline{ACK}$), the transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. Then pin RC6/TX/CK/SCK/SCL/SEG9 should be enabled by setting bit CKP.

FIGURE 14-10: I²C™ WAVEFORMS FOR TRANSMISSION (7-BIT ADDRESS)

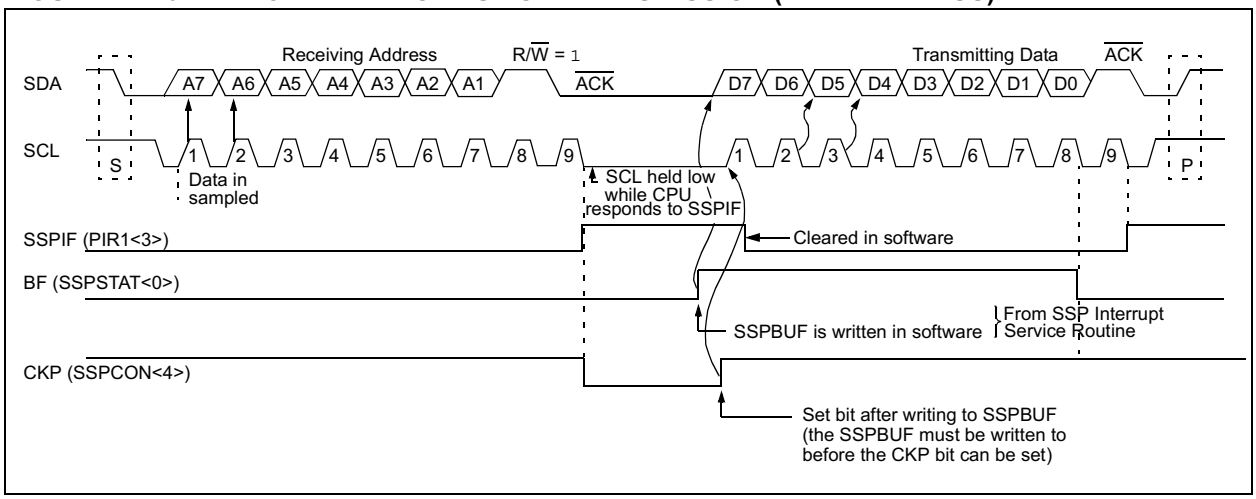
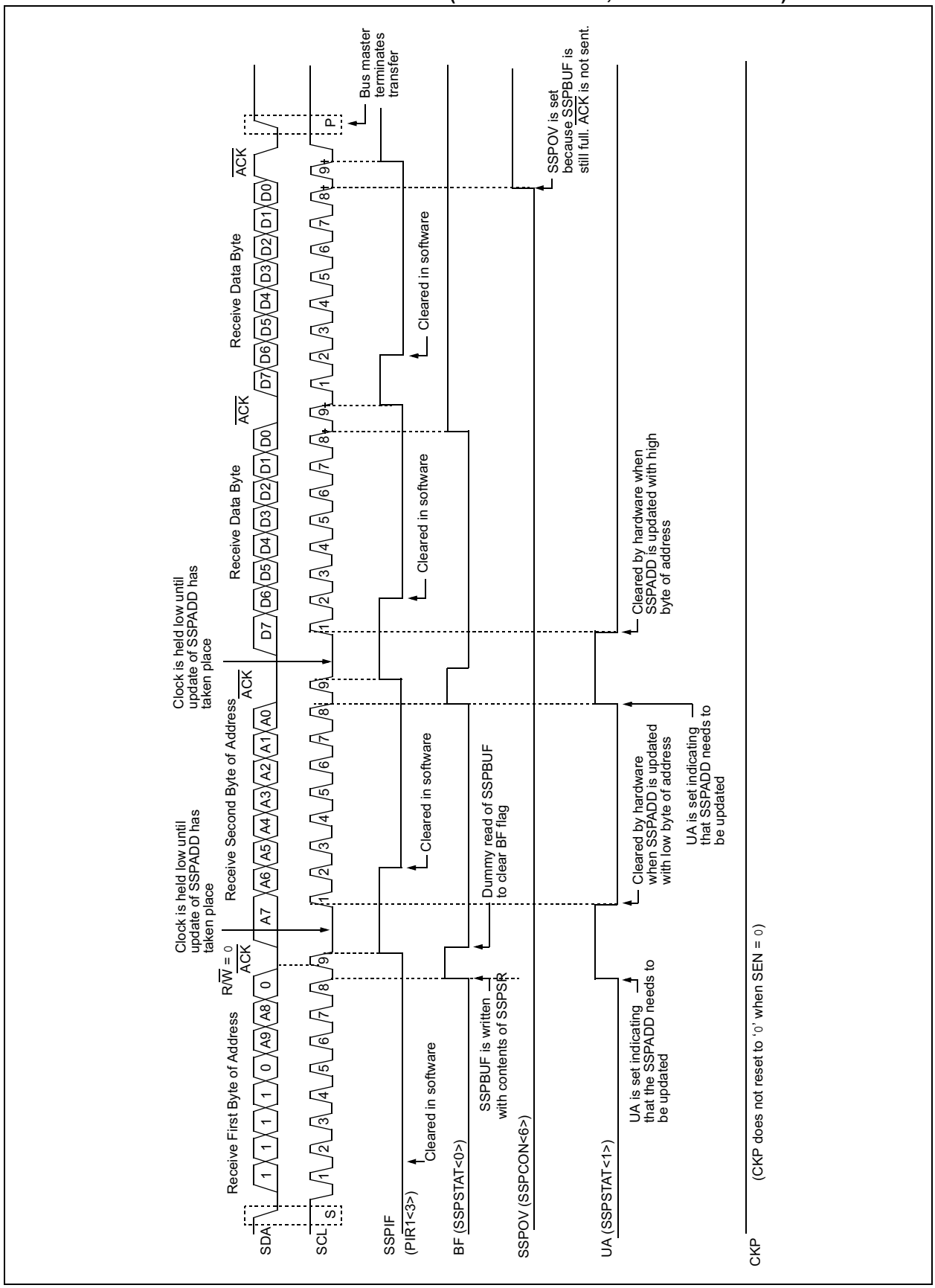


FIGURE 14-11: I²C™ SLAVE MODE TIMING (TRANSMISSION, 10-BIT ADDRESS)



14.13 Master Mode

Master mode of operation is supported in firmware using interrupt generation on the detection of the Start and Stop conditions. The Stop (P) and Start (S) bits are cleared from a Reset or when the SSP module is disabled. The Stop (P) and Start (S) bits will toggle based on the Start and Stop conditions. Control of the I^2C bus may be taken when the P bit is set or the bus is idle and both the S and P bits are clear.

In Master mode, the SCL and SDA lines are manipulated by clearing the corresponding TRISC<7,6> bit(s). The output level is always low, irrespective of the value(s) in PORTC<7,6>. So when transmitting data, a '1' data bit must have the TRISC<6> bit set (input) and a '0' data bit must have the TRISC<7> bit cleared (output). The same scenario is true for the SCL line with the TRISC<6> bit. Pull-up resistors must be provided externally to the SCL and SDA pins for proper operation of the I^2C module.

The following events will cause the SSP Interrupt Flag bit, SSPIF, to be set (SSP Interrupt will occur if enabled):

- Start condition
- Stop condition
- Data transfer byte transmitted/received

Master mode of operation can be done with either the Slave mode idle (SSPM<3:0> = 1011), or with the Slave active. When both Master and Slave modes are enabled, the software needs to differentiate the source(s) of the interrupt.

14.14 Multi-Master Mode

In Multi-Master mode, the interrupt generation on the detection of the Start and Stop conditions, allows the determination of when the bus is free. The Stop (P) and Start (S) bits are cleared from a Reset or when the SSP module is disabled. The Stop (P) and Start (S) bits will toggle based on the Start and Stop conditions. Control of the I^2C bus may be taken when bit P (SSPSTAT<4>) is set, or the bus is idle and both the S and P bits clear. When the bus is busy, enabling the SSP Interrupt will generate the interrupt when the Stop condition occurs.

In Multi-Master operation, the SDA line must be monitored to see if the signal level is the expected output level. This check only needs to be done when a high level is output. If a high level is expected and a low level is present, the device needs to release the SDA and SCL lines (set TRISC<7,6>). There are two stages where this arbitration can be lost, these are:

- Address Transfer
- Data Transfer

When the slave logic is enabled, the slave continues to receive. If arbitration was lost during the address transfer stage, communication to the device may be in progress. If addressed, an $\overline{\text{ACK}}$ pulse will be generated. If arbitration was lost during the data transfer stage, the device will need to re-transfer the data at a later time.

14.14.1 CLOCK SYNCHRONIZATION AND THE CKP BIT

When the CKP bit is cleared, the SCL output is forced to '0'; however, setting the CKP bit will not assert the SCL output low until the SCL output is already sampled low. Therefore, the CKP bit will not assert the SCL line until an external I^2C master device has already asserted the SCL line. The SCL output will remain low until the CKP bit is set and all other devices on the I^2C bus have deasserted SCL. This ensures that a write to the CKP bit will not violate the minimum high time requirement for SCL (see Figure 14-12).

FIGURE 14-12: CLOCK SYNCHRONIZATION TIMING

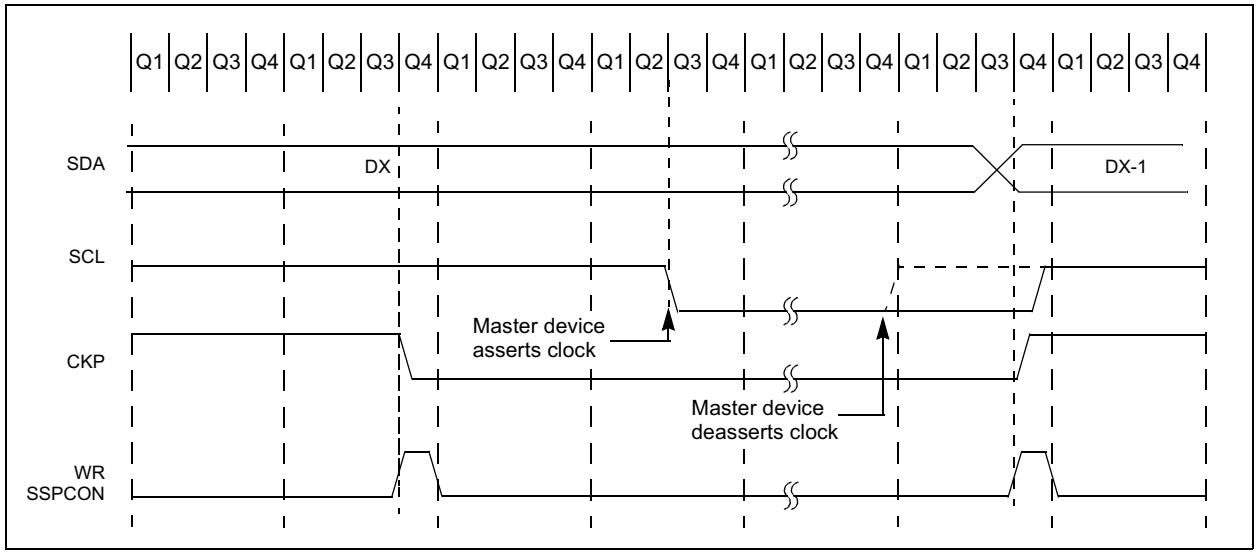


TABLE 14-4: SUMMARY OF REGISTERS ASSOCIATED WITH I²C™ OPERATION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	0000 0000
PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IF	TMR1IF	0000 0000	0000 0000
PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
SSPSTAT	SMP(1)	CKE(1)	D/$\overline{\text{A}}$	P	S	R/$\overline{\text{W}}$	UA	BF	0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111

Legend: – = Unimplemented locations, read as '0', u = unchanged, x = unknown. Shaded cells are not used by the SSP module.

Note 1: Maintain these bits clear.

NOTES:

15.0 CAPTURE/COMPARE/PWM (CCP) MODULE

The Capture/Compare/PWM module is a peripheral which allows the user to time and control different events. In Capture mode, the peripheral allows the timing of the duration of an event. The Compare mode allows the user to trigger an external event when a predetermined amount of time has expired. The PWM mode can generate a Pulse-Width Modulated signal of varying frequency and duty cycle.

The timer resources used by the module are shown in Table 15-1.

Additional information on CCP modules is available in the Application Note AN594, *"Using the CCP Modules"* (DS00594).

TABLE 15-1: CCP MODE – TIMER RESOURCES REQUIRED

CCP Mode	Timer Resource
Capture	Timer1
Compare	Timer1
PWM	Timer2

TABLE 15-2: INTERACTION OF TWO CCP MODULES

CCPx Mode	CCPy Mode	Interaction
Capture	Capture	Same TMR1 time base
Capture	Compare	Same TMR1 time base
Compare	Compare	Same TMR1 time base
PWM	PWM	The PWMs will have the same frequency and update rate (TMR2 interrupt). The rising edges will be aligned.
PWM	Capture	None
PWM	Compare	None

Note: CCPRx and CCPx throughout this document refer to CCPR1 or CCPR2 and CCP1 or CCP2, respectively.

REGISTER 15-1: CCPxCON: CCPx CONTROL REGISTER

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	CCPxX	CCPxY	CCP1M3	CCP1M2	CCP1M1	CCP1M0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

bit 7-6 **Unimplemented:** Read as '0'

bit 5-4 **CCPxX:CCPxY**: PWM Least Significant bits

Capture mode:

Unused

Compare mode:

Unused

PWM mode:

These bits are the two LSbs of the PWM duty cycle. The eight MSbs are found in CCPRxL.

bit 3-0 **CCPxM<3:0>:** CCP Mode Select bits

0000 = Capture/Compare/PWM off (resets CCP module)
0001 = Unused (reserved)
0010 = Unused (reserved)
0011 = Unused (reserved)
0100 = Capture mode, every falling edge
0101 = Capture mode, every rising edge
0110 = Capture mode, every 4th rising edge
0111 = Capture mode, every 16th rising edge
1000 = Compare mode, set output on match (CCPxIF bit is set)
1001 = Compare mode, clear output on match (CCPxIF bit is set)
1010 = Compare mode, generate software interrupt on match (CCPxIF bit is set, CCPx pin is unaffected)
1011 = Compare mode, trigger special event (CCPxIF bit is set, TMR1 is reset and A/D conversion is started if the ADC module is enabled. CCPx pin is unaffected.)
11xx = PWM mode.

15.1 Capture Mode

In Capture mode, CCPRxH:CCPRxL captures the 16-bit value of the TMR1 register when an event occurs on pin CCPx. An event is defined as one of the following and is configured by the CCPxM<3:0> bits of the CCPxCON register:

- Every falling edge
- Every rising edge
- Every 4th rising edge
- Every 16th rising edge

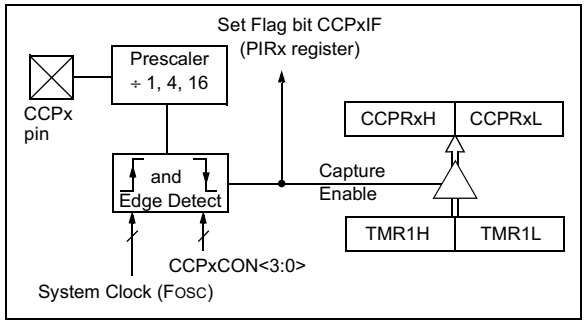
When a capture is made, the Interrupt Request Flag bit CCPxIF of the PIRx register is set. The interrupt flag must be cleared in software. If another capture occurs before the value in the CCPRxH, CCPRxL register pair is read, the old captured value is overwritten by the new captured value (see Figure 15-1).

15.1.1 CCPx PIN CONFIGURATION

In Capture mode, the CCPx pin should be configured as an input by setting the associated TRIS control bit.

Note:	If the CCPx pin is configured as an output, a write to the port can cause a capture condition.

FIGURE 15-1: CAPTURE MODE OPERATION BLOCK DIAGRAM



15.1.2 TIMER1 MODE SELECTION

Timer1 must be running in Timer mode or Synchronized Counter mode for the CCP module to use the capture feature. In Asynchronous Counter mode, the capture operation may not work.

15.1.3 SOFTWARE INTERRUPT

When the Capture mode is changed, a false capture interrupt may be generated. The user should keep the CCPxIE interrupt enable bit of the PIEx register clear to avoid false interrupts. Additionally, the user should clear the CCPxIF interrupt flag bit of the PIRx register following any change in operating mode.

15.1.4 CCP PRESCALER

There are four prescaler settings specified by the CCPxM<3:0> bits of the CCPxCON register. Whenever the CCP module is turned off, or the CCP module is not in Capture mode, the prescaler counter is cleared. Any Reset will clear the prescaler counter.

Switching from one capture prescaler to another does not clear the prescaler and may generate a false interrupt. To avoid this unexpected operation, turn the module off by clearing the CCPxCON register before changing the prescaler (see Example 15-1).

EXAMPLE 15-1: CHANGING BETWEEN CAPTURE PRESCALERS

```
BANKSEL CCP1CON     ;Set Bank bits to point
                    ;to CCP1CON
CLRF    CCP1CON     ;Turn CCP module off
MOVLW   NEW_CAPT_PS ;Load the W reg with
                    ; the new prescaler
                    ; move value and CCP ON
MOVWF   CCP1CON     ;Load CCP1CON with this
                    ; value
```

15.2 Compare Mode

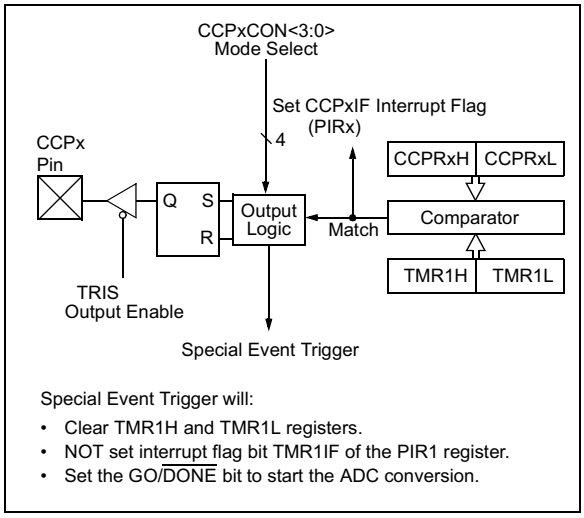
In Compare mode, the 16-bit CCPRx register value is constantly compared against the TMR1 register pair value. When a match occurs, the CCPx module may:

- Toggle the CCPx output.
- Set the CCPx output.
- Clear the CCPx output.
- Generate a Special Event Trigger.
- Generate a Software Interrupt.

The action on the pin is based on the value of the CCPxM<3:0> control bits of the CCPxCON register.

All Compare modes can generate an interrupt.

FIGURE 15-2: COMPARE MODE OPERATION BLOCK DIAGRAM



15.2.1 CCPx PIN CONFIGURATION

The user must configure the CCPx pin as an output by clearing the associated TRIS bit.

Note:	Clearing the CCPxCON register will force the CCPx compare output latch to the default low level. This is not the PORT I/O data latch.

15.2.2 TIMER1 MODE SELECTION

In Compare mode, Timer1 must be running in either Timer mode or Synchronized Counter mode. The compare operation may not work in Asynchronous Counter mode.

15.2.3 SOFTWARE INTERRUPT MODE

When Generate Software Interrupt mode is chosen (CCPxM<3:0> = `1010`), the CCPx module does not assert control of the CCPx pin (see the CCPxCON register).

15.2.4 SPECIAL EVENT TRIGGER

When Special Event Trigger mode is chosen (CCPxM<3:0> = `1011`), the CCPx module does the following:

- Resets Timer1
- Starts an ADC conversion if ADC is enabled

The CCPx module does not assert control of the CCPx pin in this mode (see the CCPxCON register).

The Special Event Trigger output of the CCP occurs immediately upon a match between the TMR1H, TMR1L register pair and the CCPRxH, CCPRxL register pair. The TMR1H, TMR1L register pair is not reset until the next rising edge of the Timer1 clock. This allows the CCPRxH, CCPRxL register pair to effectively provide a 16-bit programmable period register for Timer1.

Note 1:	The Special Event Trigger from the CCP module does not set interrupt flag bit TMRxIF of the PIR1 register.
2:	Removing the match condition by changing the contents of the CCPRxH and CCPRxL register pair, between the clock edge that generates the Special Event Trigger and the clock edge that generates the Timer1 Reset, will preclude the Reset from occurring.

15.3 PWM Mode

The PWM mode generates a Pulse-Width Modulated signal on the CCPx pin. The duty cycle, period and resolution are determined by the following registers:

- PR2
- T2CON
- CCPRxL
- CCPxCON

In Pulse-Width Modulation (PWM) mode, the CCP module produces up to a 10-bit resolution PWM output on the CCPx pin. Since the CCPx pin is multiplexed with the PORT data latch, the TRIS for that pin must be cleared to enable the CCPx pin output driver.

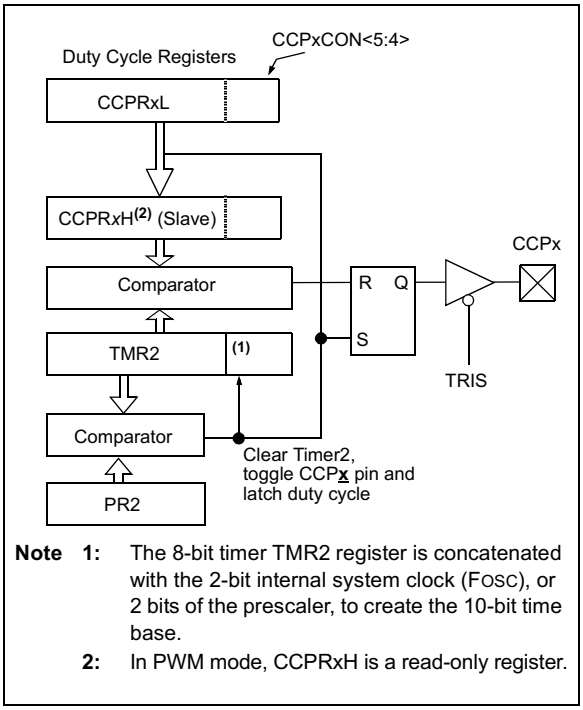
Note:	Clearing the CCPxCON register will relinquish CCPx control of the CCPx pin.

Figure 15-3 shows a simplified block diagram of PWM operation.

Figure 15-4 shows a typical waveform of the PWM signal.

For a step-by-step procedure on how to set up the CCP module for PWM operation, see **Section 15.3.7 "Setup for PWM Operation"**.

FIGURE 15-3: SIMPLIFIED PWM BLOCK DIAGRAM

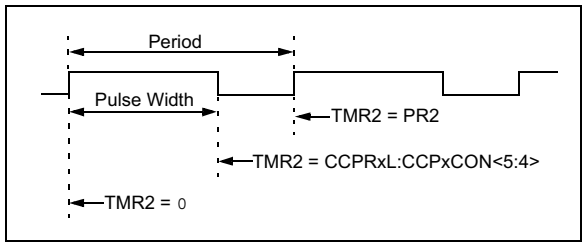


Note 1: The 8-bit timer TMR2 register is concatenated with the 2-bit internal system clock (FOSC), or 2 bits of the prescaler, to create the 10-bit time base.

2: In PWM mode, CCPRxH is a read-only register.

The PWM output (Figure 15-2) has a time base (period) and a time that the output stays high (duty cycle).

FIGURE 15-4: CCP PWM OUTPUT



15.3.1 PWM PERIOD

The PWM period is specified by the PR2 register of Timer2. The PWM period can be calculated using the formula of Equation 15-1.

EQUATION 15-1: PWM PERIOD

$$PWM\ Period = [(PR2) + 1] \bullet 4 \bullet T_{OSC} \bullet (TMR2\ Prescale\ Value)$$

Note: TOSC = 1/FOSC

When TMR2 is equal to PR2, the following three events occur on the next increment cycle:

- TMR2 is cleared
- The CCPx pin is set. (Exception: If the PWM duty cycle = 0%, the pin will not be set.)
- The PWM duty cycle is latched from CCPRxL into CCPRxH.

Note: The Timer2 postscaler (see **Section 7.1 "Timer2 Operation"**) is not used in the determination of the PWM frequency.

15.3.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing a 10-bit value to multiple registers: CCPRxL register and CCPx<1:0> bits of the CCPxCON register. The CCPRxL contains the eight MSbs and the CCPx<1:0> bits of the CCPxCON register contain the two LSbs. CCPRxL and CCPx<1:0> bits of the CCPxCON register can be written to at any time. The duty cycle value is not latched into CCPRxH until after the period completes (i.e., a match between PR2 and TMR2 registers occurs). While using the PWM, the CCPRxH register is read-only.

Equation 15-2 is used to calculate the PWM pulse width.

Equation 15-3 is used to calculate the PWM duty cycle ratio.

EQUATION 15-2: PULSE WIDTH

$$Pulse\ Width = (CCPRxL:CCPxCON<5:4>) \bullet T_{OSC} \bullet (TMR2\ Prescale\ Value)$$

EQUATION 15-3: DUTY CYCLE RATIO

$$Duty\ Cycle\ Ratio = \frac{(CCPRxL:CCPxCON<5:4>)}{4(PR2 + 1)}$$

The CCPRxH register and a 2-bit internal latch are used to double buffer the PWM duty cycle. This double buffering is essential for glitchless PWM operation.

The 8-bit timer TMR2 register is concatenated with either the 2-bit internal system clock (FOSC), or 2 bits of the prescaler, to create the 10-bit time base. The system clock is used if the Timer2 prescaler is set to 1:1.

When the 10-bit time base matches the CCPRxH and 2-bit latch, then the CCPx pin is cleared (see Figure 15-3).

15.3.3 PWM RESOLUTION

The resolution determines the number of available duty cycles for a given period. For example, a 10-bit resolution will result in 1024 discrete duty cycles, whereas an 8-bit resolution will result in 256 discrete duty cycles.

The maximum PWM resolution is 10 bits when PR2 is 255. The resolution is a function of the PR2 register value as shown by Equation 15-4.

EQUATION 15-4: PWM RESOLUTION

$$Resolution = \frac{log[4(PR2 + 1)]}{log(2)} \ bits$$

Note: If the pulse width value is greater than the period the assigned PWM pin(s) will remain unchanged.

TABLE 15-3: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS (FOSC = 20 MHz)

PWM Frequency	1.22 kHz	4.88 kHz	19.53 kHz	78.12 kHz	156.3 kHz	208.3 kHz
Timer Prescale (1, 4, 16)	16	4	1	1	1	1
PR2 Value	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
Maximum Resolution (bits)	10	10	10	8	7	6.6

TABLE 15-4: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS (FOSC = 8 MHz)

PWM Frequency	1.22 kHz	4.90 kHz	19.61 kHz	76.92 kHz	153.85 kHz	200.0 kHz
Timer Prescale (1, 4, 16)	16	4	1	1	1	1
PR2 Value	0x65	0x65	0x65	0x19	0x0C	0x09
Maximum Resolution (bits)	8	8	8	6	5	5

15.3.4 OPERATION IN SLEEP MODE

In Sleep mode, the TMR2 register will not increment and the state of the module will not change. If the CCPx pin is driving a value, it will continue to drive that value. When the device wakes up, TMR2 will continue from its previous state.

15.3.5 CHANGES IN SYSTEM CLOCK FREQUENCY

The PWM frequency is derived from the system clock frequency. Any changes in the system clock frequency will result in changes to the PWM frequency. See **Section 4.0 "Oscillator Module (With Fail-Safe Clock Monitor)"** for additional details.

15.3.6 EFFECTS OF RESET

Any Reset will force all ports to Input mode and the CCP registers to their Reset states.

15.3.7 SETUP FOR PWM OPERATION

The following steps should be taken when configuring the CCP module for PWM operation:

1. Disable the PWM pin (CCPx) output drivers by setting the associated TRIS bit.
2. Set the PWM period by loading the PR2 register.
3. Configure the CCP module for the PWM mode by loading the CCPxCON register with the appropriate values.
4. Set the PWM duty cycle by loading the CCPRxL register and CCPx bits of the CCPxCON register.
5. Configure and start Timer2:
   - Clear the TMR2IF interrupt flag bit of the PIR1 register.
   - Set the Timer2 prescale value by loading the T2CKPS bits of the T2CON register.
   - Enable Timer2 by setting the TMR2ON bit of the T2CON register.
6. Enable PWM output after a new PWM cycle has started:
   - Wait until Timer2 overflows (TMR2IF bit of the PIR1 register is set).
   - Enable the CCPx pin output driver by clearing the associated TRIS bit.

TABLE 15-5: SUMMARY OF REGISTERS ASSOCIATED WITH CAPTURE, COMPARE AND PWM

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
CCPxCON	—	—	CCPxX	CCPxY	CCPxM3	CCPxM2	CCPxM1	CCPxM0	--00 0000	--00 0000
CCPRxL	Capture/Compare/PWM Register X Low Byte								xxxx xxxx	uuuu uuuu
CCPRxH	Capture/Compare/PWM Register X High Byte								xxxx xxxx	uuuu uuuu
CMCON1	—	—	—	—	—	—	T1GSS	C2SYNC	---- --10	---- --10
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
LCDCON	LCDEN	$\overline{\text{SLPEN}}$	WERR	VLCDEN	CS1	CS0	LMUX1	LMUX0	0001 0011	0001 0011
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE9	SE8	0000 0000	0000 0000
PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIE2	OSFIE	C2IE	C1IE	LCDIE	—	LVDIE	—	CCP2IE	0000 -0-0	0000 -0-0
PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIR2	OSFIF	C2IF	C1IF	LCDIF	—	LVDIF	—	CCP2IF	0000 -0-0	0000 -0-0
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
T1CON	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	0000 0000	uuuu uuuu
T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	uuuu uuuu
TMR2	Timer2 Module Register								0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
TRISD(1)	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	1111 1111	1111 1111

Legend: - = Unimplemented locations, read as '0', u = unchanged, x = unknown. Shaded cells are not used by the Capture, Compare and PWM.

Note 1: PIC16F914/917 and PIC16F946 only.

16.0 SPECIAL FEATURES OF THE CPU

The PIC16F91X/946 has a host of features intended to maximize system reliability, minimize cost through elimination of external components, provide power-saving features and offer code protection.

These features are:

- Reset
  - Power-on Reset (POR)
  - Power-up Timer (PWRT)
  - Oscillator Start-up Timer (OST)
  - Brown-out Reset (BOR)
- Interrupts
- Watchdog Timer (WDT)
- Oscillator Selection
- Sleep
- Code Protection
- ID Locations
- In-Circuit Serial Programming™

The PIC16F91X/946 has two timers that offer necessary delays on power-up. One is the Oscillator Start-up Timer (OST), intended to keep the chip in Reset until the crystal oscillator is stable. The other is the Power-up Timer (PWRT), which provides a fixed delay of 64 ms (nominal) on power-up only, designed to keep the part in Reset while the power supply stabilizes. There is also circuitry to reset the device if a brown-out occurs, which can use the Power-up Timer to provide at least a 64 ms Reset. With these three functions-on-chip, most applications need no external Reset circuitry.

The Sleep mode is designed to offer a very low-current Power-down mode. The user can wake-up from Sleep through:

- External Reset
- Watchdog Timer Wake-up
- An interrupt

Several oscillator options are also made available to allow the part to fit the application. The INTOSC option saves system cost, while the LP crystal option saves power. A set of Configuration bits are used to select various options (see Register 16-1).

16.1 Configuration Bits

The Configuration bits can be programmed (read as '0'), or left unprogrammed (read as '1') to select various device configurations as shown in Register 16-1. These bits are mapped in program memory location 2007h.

Note: Address 2007h is beyond the user program memory space. It belongs to the special configuration memory space (2000h-3FFFh), which can be accessed only during programming. See "*PIC16F91X/946 Memory Programming Specification*" (DS41244) for more information.

REGISTER 16-1: CONFIG1: CONFIGURATION WORD REGISTER 1

—	—	—	$\overline{\text{DEBUG}}$	FCMEN	IESO	BOREN1	BOREN0
bit 15							bit 8

$\overline{\text{CPD}}$	$\overline{\text{CP}}$	MCLRE	$\overline{\text{PWRTE}}$	WDTE	FOSC2	FOSC1	FOSC0
bit 7							bit 0

bit 15-13 **Unimplemented**: Read as '1'

bit 12 **$\overline{\text{DEBUG}}$:** In-Circuit Debugger Mode bit
1 = In-Circuit Debugger disabled, RB6/ICSPCLK and RB7/ICSPDAT are general purpose I/O pins
0 = In-Circuit Debugger enabled, RB6/ICSPCLK and RB7/ICSPDAT are dedicated to the debugger

bit 11 **FCMEN:** Fail-Safe Clock Monitor Enabled bit
1 = Fail-Safe Clock Monitor is enabled
0 = Fail-Safe Clock Monitor is disabled

bit 10 **IESO:** Internal External Switchover bit
1 = Internal/External Switchover mode is enabled
0 = Internal/External Switchover mode is disabled

bit 9-8 **BOREN<1:0>:** Brown-out Reset Selection bits(1)
11 = BOR enabled
10 = BOR enabled during operation and disabled in Sleep
01 = BOR controlled by SBOREN bit of the PCON register
00 = BOR disabled

bit 7 **$\overline{\text{CPD}}$:** Data Code Protection bit(2)
1 = Data memory code protection is disabled
0 = Data memory code protection is enabled

bit 6 **$\overline{\text{CP}}$:** Code Protection bit(3)
1 = Program memory code protection is disabled
0 = Program memory code protection is enabled

bit 5 **MCLRE:** RE3/$\overline{\text{MCLR}}$ pin function select bit(4)
1 = RE3/$\overline{\text{MCLR}}$ pin function is $\overline{\text{MCLR}}$
0 = RE3/$\overline{\text{MCLR}}$ pin function is digital input, $\overline{\text{MCLR}}$ internally tied to V_{DD}

bit 4 **$\overline{\text{PWRTE}}$:** Power-up Timer Enable bit
1 = PWRT disabled
0 = PWRT enabled

bit 3 **WDTE:** Watchdog Timer Enable bit
1 = WDT enabled
0 = WDT disabled and can be enabled by SWDTEN bit of the WDTCON register

bit 2-0 **FOSC<2:0>:** Oscillator Selection bits
111 = RC oscillator: CLKOUT function on RA6/OSC2/CLKOUT/T1OSO pin, RC on RA7/OSC1/CLKIN/T1OSI
110 = RCIO oscillator: I/O function on RA6/OSC2/CLKOUT/T1OSO pin, RC on RA7/OSC1/CLKIN/T1OSI
101 = INTOSC oscillator: CLKOUT function on RA6/OSC2/CLKOUT/T1OSO pin, I/O function on RA7/OSC1/CLKIN/T1OSI
100 = INTOSCIO oscillator: I/O function on RA6/OSC2/CLKOUT/T1OSO pin, I/O function on RA7/OSC1/CLKIN/T1OSI
011 = EC: I/O function on RA6/OSC2/CLKOUT/T1OSO pin, CLKIN on RA7/OSC1/CLKIN/T1OSI
010 = HS oscillator: High-speed crystal/resonator on RA6/OSC2/CLKOUT/T1OSO and RA7/OSC1/CLKIN/T1OSI
001 = XT oscillator: Crystal/resonator on RA6/OSC2/CLKOUT/T1OSO and RA7/OSC1/CLKIN/T1OSI
000 = LP oscillator: Low-power crystal on RA6/OSC2/CLKOUT/T1OSO and RA7/OSC1/CLKIN/T1OSI

Note 1: Enabling Brown-out Reset does not automatically enable Power-up Timer.
2: The entire data EEPROM will be erased when the code protection is turned off.
3: The entire program memory will be erased when the code protection is turned off.
4: When $\overline{\text{MCLR}}$ is asserted in INTOSC or RC mode, the internal clock oscillator is disabled.

16.2 Resets

The PIC16F91X/946 differentiates between various kinds of Reset:

a) Power-on Reset (POR)
b) WDT Reset during normal operation
c) WDT Reset during Sleep
d) $\overline{\text{MCLR}}$ Reset during normal operation
e) $\overline{\text{MCLR}}$ Reset during Sleep
f) Brown-out Reset (BOR)

Some registers are not affected in any Reset condition; their status is unknown on POR and unchanged in any other Reset. Most other registers are reset to a "Reset state" on:

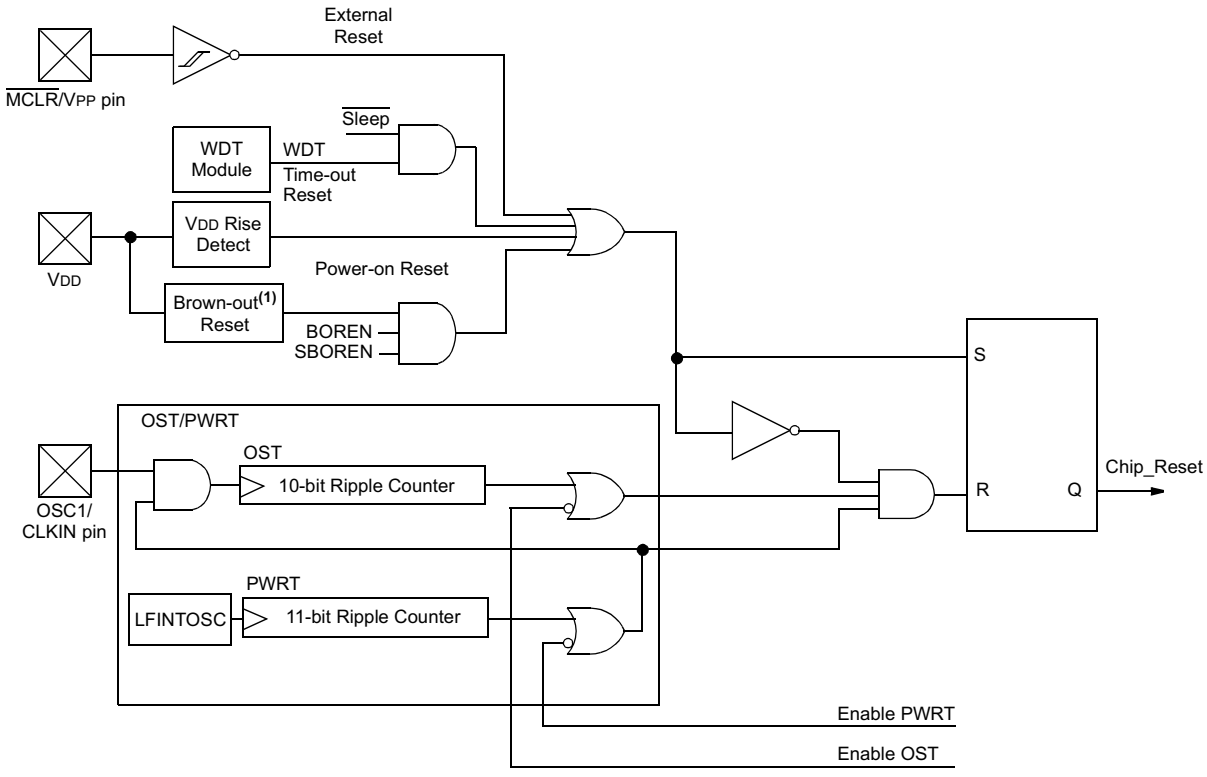
- Power-on Reset
- $\overline{\text{MCLR}}$ Reset
- $\overline{\text{MCLR}}$ Reset during Sleep
- WDT Reset
- Brown-out Reset (BOR)

They are not affected by a WDT wake-up since this is viewed as the resumption of normal operation. $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits are set or cleared differently in different Reset situations, as indicated in Table 16-2. These bits are used in software to determine the nature of the Reset. See Table 16-5 for a full description of Reset states of all registers.

A simplified block diagram of the On-Chip Reset Circuit is shown in Figure 16-1.

The $\overline{\text{MCLR}}$ Reset path has a noise filter to detect and ignore small pulses. See **Section 19.0 "Electrical Specifications"** for pulse width specifications.

FIGURE 16-1: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



Note 1: Refer to the Configuration Word register (Register 16-1).

16.2.1 POWER-ON RESET (POR)

The on-chip POR circuit holds the chip in Reset until VDD has reached a high enough level for proper operation. To take advantage of the POR, simply connect the MCLR pin through a resistor to VDD. This will eliminate external RC components usually needed to create Power-on Reset. A maximum rise time for VDD is required. See **Section 19.0 "Electrical Specifications"** for details. If the BOR is enabled, the maximum rise time specification does not apply. The BOR circuitry will keep the device in Reset until VDD reaches VBOR (see **Section 16.2.4 "Brown-Out Reset (BOR)"**).

Note:	The POR circuit does not produce an internal Reset when VDD declines. To re-enable the POR, VDD must reach Vss for a minimum of 100 µs.

When the device starts normal operation (exits the Reset condition), device operating parameters (i.e., voltage, frequency, temperature, etc.) must be met to ensure operation. If these conditions are not met, the device must be held in Reset until the operating conditions are met.

For additional information, refer to Application Note AN607, *"Power-up Trouble Shooting"* (DS00607).

16.2.2 MCLR

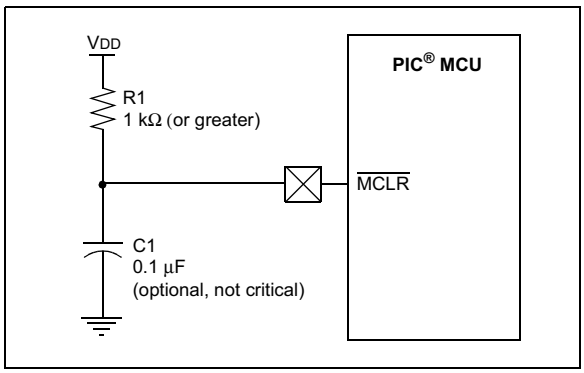
PIC16F91X/946 has a noise filter in the MCLR Reset path. The filter will detect and ignore small pulses.

It should be noted that a WDT Reset does not drive MCLR pin low.

Voltages applied to the pin that exceed its specification can result in both MCLR Resets and excessive current beyond the device specification during the ESD event. For this reason, Microchip recommends that the MCLR pin no longer be tied directly to VDD. The use of an RC network, as shown in Figure 16-2, is suggested.

An internal MCLR option is enabled by clearing the MCLRE bit in the Configuration Word register. When MCLRE = 0, the Reset signal to the chip is generated internally. When the MCLRE = 1, the RE3/MCLR pin becomes an external Reset input. In this mode, the RE3/MCLR pin has a weak pull-up to VDD. In-Circuit Serial Programming is not affected by selecting the internal MCLR option.

FIGURE 16-2: RECOMMENDED MCLR CIRCUIT



16.2.3 POWER-UP TIMER (PWRT)

The Power-up Timer provides a fixed 64 ms (nominal) time-out on power-up only, from POR or Brown-out Reset. The Power-up Timer operates from the 31 kHz LFINTOSC oscillator. For more information, see **Section 4.5 "Internal Clock Modes"**. The chip is kept in Reset as long as PWRT is active. The PWRT delay allows the VDD to rise to an acceptable level. A Configuration bit, PWRTE, can disable (if set) or enable (if cleared or programmed) the Power-up Timer. The Power-up Timer should be enabled when Brown-out Reset is enabled, although it is not required.

The Power-up Timer delay will vary from chip-to-chip and vary due to:

- VDD variation
- Temperature variation
- Process variation

See DC parameters for details (**Section 19.0 "Electrical Specifications"**).

16.2.4 BROWN-OUT RESET (BOR)

The BOREN0 and BOREN1 bits in the Configuration Word register selects one of four BOR modes. Two modes have been added to allow software or hardware control of the BOR enable. When BOREN<1:0> = 01, the SBOREN bit of the PCON register enables/disables the BOR allowing it to be controlled in software. By selecting BOREN<1:0>, the BOR is automatically disabled in Sleep to conserve power and enabled on wake-up. In this mode, the SBOREN bit is disabled. See Register 16-1 for the Configuration Word definition.

If VDD falls below VBOR for greater than parameter (TBOR) (see **Section 19.0 "Electrical Specifications"**), the Brown-out situation will reset the device. This will occur regardless of VDD slew rate. A Reset is not insured to occur if VDD falls below VBOR for less than parameter (TBOR).

On any Reset (Power-on, Brown-out Reset, Watchdog Timer, etc.), the chip will remain in Reset until VDD rises above VBOR (see Figure 16-3). The Power-up Timer will now be invoked, if enabled and will keep the chip in Reset an additional 64 ms.

Note: The Power-up Timer is enabled by the $\overline{\text{PWRTE}}$ bit in the Configuration Word.

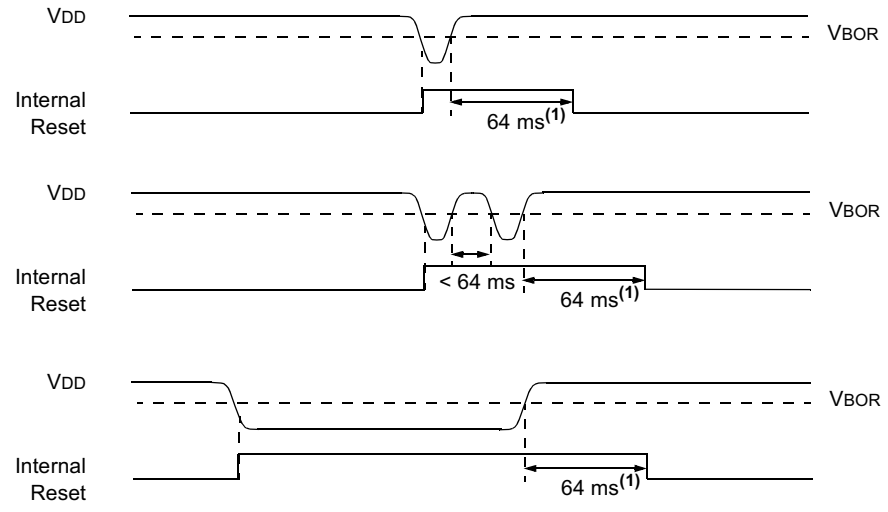
If VDD drops below VBOR while the Power-up Timer is running, the chip will go back into a Brown-out Reset and the Power-up Timer will be re-initialized. Once VDD rises above VBOR, the Power-up Timer will execute a 64 ms Reset.

16.2.5 BOR CALIBRATION

The PIC16F91X/946 stores the BOR calibration values in fuses located in the Calibration Word (2008h). The Calibration Word is not erased when using the specified bulk erase sequence in the "*PIC16F91X/946 Memory Programming Specification*" (DS41244) and thus, does not require reprogramming.

Address 2008h is beyond the user program memory space. It belongs to the special configuration memory space (2000h-3FFFh), which can be accessed only during programming. See "*PIC16F91X/946 Memory Programming Specification*" (DS41244) for more information.

FIGURE 16-3: BROWN-OUT SITUATIONS



Note 1: 64 ms delay only if $\overline{\text{PWRTE}}$ bit is programmed to '0'.

16.2.6 TIME-OUT SEQUENCE

On power-up, the time-out sequence is as follows: first, PWRT time-out is invoked after POR has expired, then OST is activated after the PWRT time-out has expired. The total time-out will vary based on oscillator configuration and $\overline{PWRTE}$ bit status. For example, in EC mode with $\overline{PWRTE}$ bit erased (PWRT disabled), there will be no time-out at all. Figure 16-4, Figure 16-5 and Figure 16-6 depict time-out sequences. The device can execute code from the INTOSC while OST is active, by enabling Two-Speed Start-up or Fail-Safe Monitor (see **Section 4.7.2 "Two-Speed Start-up Sequence"** and **Section 4.8 "Fail-Safe Clock Monitor"**).

Since the time-outs occur from the POR pulse, if $\overline{MCLR}$ is kept low long enough, the time-outs will expire. Then, bringing $\overline{MCLR}$ high will begin execution immediately (see Figure 16-5). This is useful for testing purposes or to synchronize more than one PIC16F91X/946 device operating in parallel.

Table 16-5 shows the Reset conditions for some special registers, while Table 16-5 shows the Reset conditions for all the registers.

16.2.7 POWER CONTROL (PCON) REGISTER

The Power Control (PCON) register (address 8Eh) has two Status bits to indicate what type of Reset that last occurred.

Bit 0 is $\overline{BOR}$ (Brown-out Reset). $\overline{BOR}$ is unknown on Power-on Reset. It must then be set by the user and checked on subsequent Resets to see if $\overline{BOR}$ = 0, indicating that a Brown-out has occurred. The $\overline{BOR}$ Status bit is a "don't care" and is not necessarily predictable if the brown-out circuit is disabled (BOREN<1:0> = 00 in the Configuration Word register).

Bit 1 is $\overline{POR}$ (Power-on Reset). It is a '0' on Power-on Reset and unaffected otherwise. The user must write a '1' to this bit following a Power-on Reset. On a subsequent Reset, if $\overline{POR}$ is '0', it will indicate that a Power-on Reset has occurred (i.e., V_{DD} may have gone too low).

For more information, see **Section 16.2.4 "Brown-Out Reset (BOR)"**.

TABLE 16-1: TIME-OUT IN VARIOUS SITUATIONS

Oscillator Configuration	Power-up		Brown-out Reset		Wake-up from Sleep
	$\overline{PWRTE}$ = 0	$\overline{PWRTE}$ = 1	$\overline{PWRTE}$ = 0	$\overline{PWRTE}$ = 1	
XT, HS, LP(1)	TPWRT + 1024 • TOSC	1024 • TOSC	TPWRT + 1024 • TOSC	1024 • TOSC	1024 • TOSC
RC, EC, INTOSC	TPWRT	—	TPWRT	—	—

Note 1: LP mode with T1OSC disabled.

TABLE 16-2: PCON BITS AND THEIR SIGNIFICANCE

$\overline{POR}$	$\overline{BOR}$	$\overline{TO}$	$\overline{PD}$	Condition
0	u	1	1	Power-on Reset
1	0	1	1	Brown-out Reset
u	u	0	u	WDT Reset
u	u	0	0	WDT Wake-up
u	u	u	u	$\overline{MCLR}$ Reset during normal operation
u	u	1	0	$\overline{MCLR}$ Reset during Sleep

Legend: u = unchanged, x = unknown

TABLE 16-3: SUMMARY OF REGISTERS ASSOCIATED WITH BROWN-OUT

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets(1)
STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
PCON	—	—	—	SBOREN	—	—	$\overline{POR}$	$\overline{BOR}$	--01 --qq	--0u --uu

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0', q = value depends on condition. Shaded cells are not used by BOR.

Note 1: Other (non Power-up) Resets include $\overline{MCLR}$ Reset and Watchdog Timer Reset during normal operation.

FIGURE 16-4: TIME-OUT SEQUENCE ON POWER-UP (DELAYED $\overline{MCLR}$): CASE 1

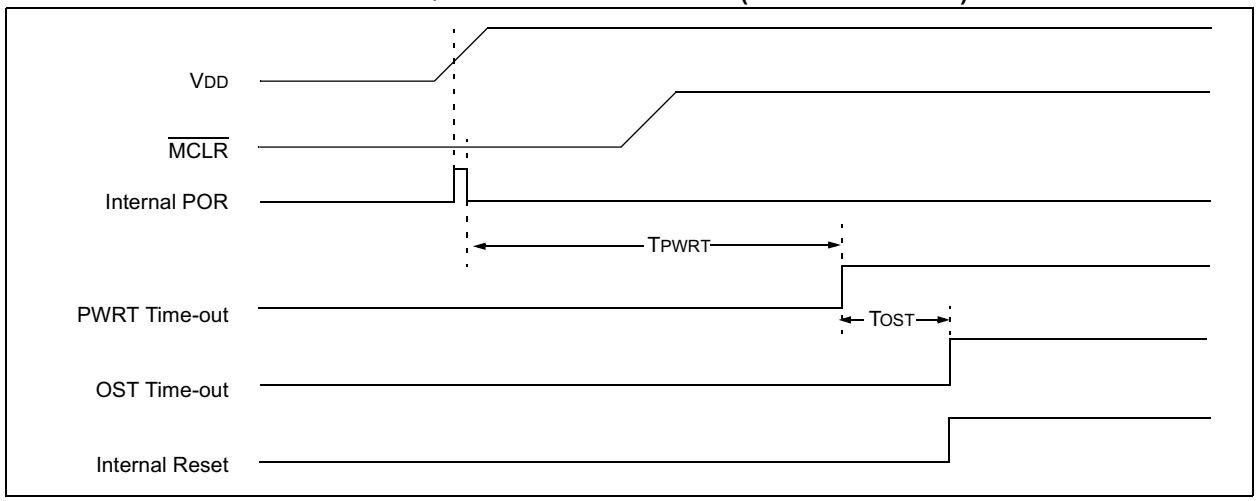


FIGURE 16-5: TIME-OUT SEQUENCE ON POWER-UP (DELAYED $\overline{MCLR}$): CASE 2

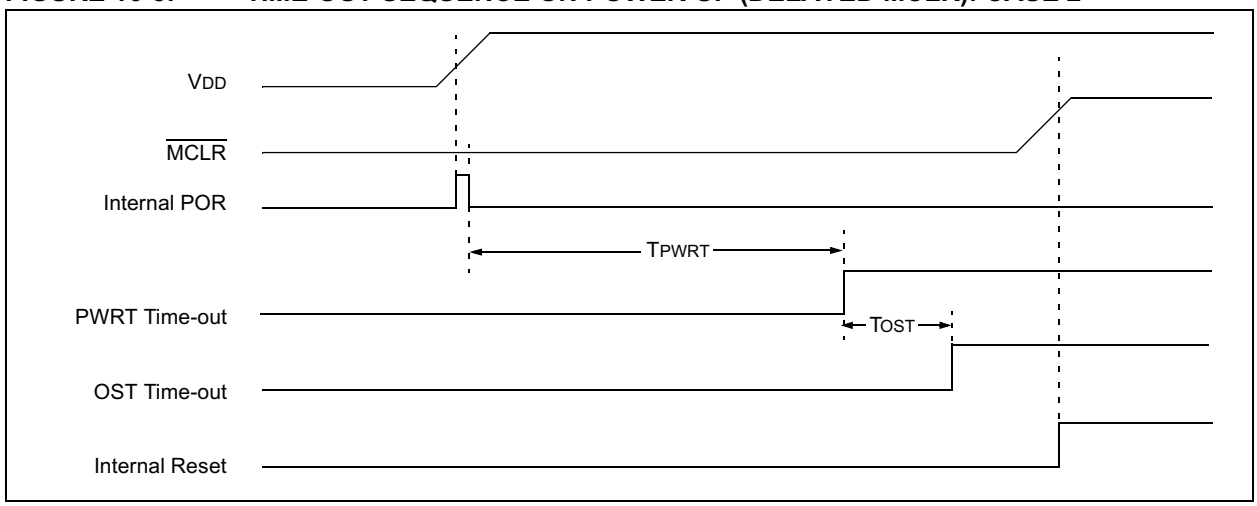


FIGURE 16-6: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ WITH VDD): CASE 3

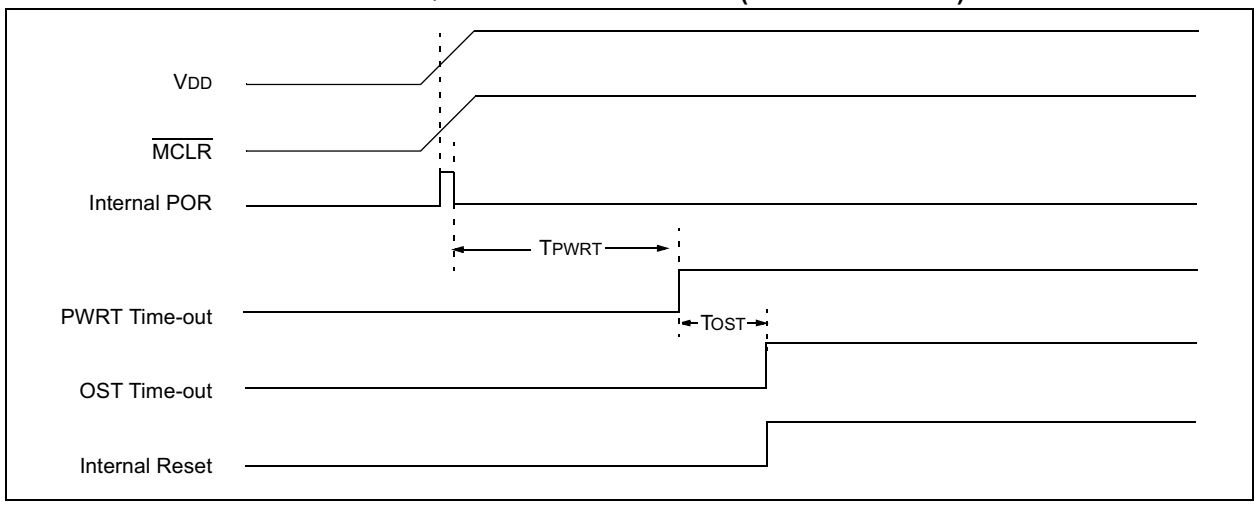


TABLE 16-4: INITIALIZATION CONDITION FOR REGISTERS

Register	Address	Power-on Reset	• $\overline{\text{MCLR}}$ Reset • WDT Reset • Brown-out Reset(1)	• Wake-up from Sleep through interrupt • Wake-up from Sleep through WDT time-out
W	—	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	00h/80h/ 100h/180h	xxxx xxxx	xxxx xxxx	uuuu uuuu
TMR0	01h/101h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02h/82h/ 102h/182h	0000 0000	0000 0000	PC + 1(3)
STATUS	03h/83h/ 103h/183h	0001 1xxx	000q quuu(4)	uuuq quuu(4)
FSR	04h/84h/ 104h/184h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA	05h	xxxx xxxx	xxxx xxxx	uuuu uuuu
PORTB	06h/106h	xxxx xxxx	xxxx xxxx	uuuu uuuu
PORTC	07h	xxxx xxxx	xxxx xxxx	uuuu uuuu
PORTD(6)	08h	xxxx xxxx	xxxx xxxx	uuuu uuuu
PORTE	09h	---- xxxx xxxx xxxx(7)	---- xxxx xxxx xxxx(7)	---- uuuu uuuu uuuu(7)
PCLATH	0Ah/8Ah/ 10Ah/18Ah	---0 0000	---0 0000	---u uuuu
INTCON	0Bh/8Bh/ 10Bh/18Bh	0000 000x	0000 000x	uuuu uuuu(2)
PIR1	0Ch	0000 0000	0000 0000	uuuu uuuu(2)
PIR2	0Dh	0000 -0-0	0000 -0-0	uuuu -u-u
TMR1L	0Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	0Fh	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	10h	0000 0000	uuuu uuuu	uuuu uuuu
TMR2	11h	0000 0000	0000 0000	uuuu uuuu
T2CON	12h	-000 0000	-000 0000	-uuu uuuu
SSPBUF	13h	xxxx xxxx	xxxx xxxx	uuuu uuuu
SSPCON	14h	0000 0000	0000 0000	uuuu uuuu
CCPR1L	15h	xxxx xxxx	xxxx xxxx	uuuu uuuu
CCPR1H	16h	xxxx xxxx	xxxx xxxx	uuuu uuuu
CCP1CON	17h	--00 0000	--00 0000	--uu uuuu
RCSTA	18h	---0 1000	---0 1000	---u uuuu
TXREG	19h	0000 0000	0000 0000	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0', q = value depends on condition.

Note 1: If VDD goes too low, Power-on Reset will be activated and registers will be affected differently.

2: One or more bits in INTCON and/or PIR1 will be affected (to cause wake-up).

3: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

4: See Table 16-5 for Reset value for specific condition.

5: If Reset was due to brown-out, then bit 0 = 0. All other Resets will cause bit 0 = u.

6: PIC16F914/917 and PIC16F946 only.

7: PIC16F946 only.

TABLE 16-4: INITIALIZATION CONDITION FOR REGISTERS (CONTINUED)

Register	Address	Power-on Reset	• $\overline{\text{MCLR}}$ Reset • WDT Reset • Brown-out Reset(1)	• Wake-up from Sleep through interrupt • Wake-up from Sleep through WDT time-out
RCREG	1Ah	0000 0000	0000 0000	uuuu uuuu
CCPR2L(6)	1Bh	xxxx xxxx	xxxx xxxx	uuuu uuuu
CCPR2H(6)	1Ch	xxxx xxxx	xxxx xxxx	uuuu uuuu
CCP2CON(6)	1Dh	--00 0000	--00 0000	--uu uuuu
ADRESH	1Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON0	1Fh	0000 0000	0000 0000	uuuu uuuu
OPTION_REG	81h/181h	1111 1111	1111 1111	uuuu uuuu
TRISA	85h	1111 1111	1111 1111	uuuu uuuu
TRISB	86h/186h	1111 1111	1111 1111	uuuu uuuu
TRISC	87h	1111 1111	1111 1111	uuuu uuuu
TRISD(6)	88h	1111 1111	1111 1111	uuuu uuuu
TRISE	89h	---- 1111 1111 1111(7)	---- 1111 1111 1111(7)	---- uuuu uuuu uuuu(7)
PIE1	8Ch	0000 0000	0000 0000	uuuu uuuu
PIE2	8Dh	0000 -0-0	0000 -0-0	uuuu -u-u
PCON	8Eh	--01 --0x	--0u --uu(1,5)	--uu --uu
OSCCON	8Fh	-110 q000	-110 x000	-uuu uuuu
OSCTUNE	90h	---0 0000	---u uuuu	---u uuuu
ANSEL	91h	1111 1111	1111 1111	uuuu uuuu
PR2	92h	1111 1111	1111 1111	1111 1111
SSPADD	93h	0000 0000	0000 0000	uuuu uuuu
SSPSTAT	94h	0000 0000	0000 0000	uuuu uuuu
WPUB	95h	1111 1111	1111 1111	uuuu uuuu
IOCB	96h	0000 ----	0000 ----	uuuu ----
CMCON1	97h	---- --10	---- --10	---- --uu
TXSTA	98h	0000 -010	0000 -010	uuuu -uuu
SPBRG	99h	0000 0000	0000 0000	uuuu uuuu
CMCON0	9Ch	0000 0000	0000 0000	uuuu uuuu
VRCON	9Dh	0-0- 0000	0-0- 0000	u-u- uuuu
ADRESL	9Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON1	9Fh	-000 ----	-000 ----	-uuu ----
WDTCON	105h	---0 1000	---0 1000	---u uuuu
LCDCON	107h	0001 0011	0001 0011	uuuu uuuu
LCDPS	108h	0000 0000	0000 0000	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0', q = value depends on condition.

Note 1: If VDD goes too low, Power-on Reset will be activated and registers will be affected differently.

2: One or more bits in INTCON and/or PIR1 will be affected (to cause wake-up).

3: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

4: See Table 16-5 for Reset value for specific condition.

5: If Reset was due to brown-out, then bit 0 = 0. All other Resets will cause bit 0 = u.

6: PIC16F914/917 and PIC16F946 only.

7: PIC16F946 only.

TABLE 16-4: INITIALIZATION CONDITION FOR REGISTERS (CONTINUED)

Register	Address	Power-on Reset	• $\overline{\text{MCLR}}$ Reset • WDT Reset • Brown-out Reset(1)	• Wake-up from Sleep through interrupt • Wake-up from Sleep through WDT time-out
LVDCON	109h	--00 -100	--00 -100	--uu -uuu
EEDATL	10Ch	0000 0000	0000 0000	uuuu uuuu
EEADRL	10Dh	0000 0000	0000 0000	uuuu uuuu
EEDATH	10Eh	--00 0000	0000 0000	uuuu uuuu
EEADRH	10Fh	---0 0000	0000 0000	uuuu uuuu
LCDDATA0	110h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA1	111h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA2(6)	112h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA3	113h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA4	114h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA5(6)	115h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA6	116h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA7	117h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA8(6)	118h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA9	119h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA10	11Ah	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA11(6)	11Bh	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDSE0	11Ch	0000 0000	uuuu uuuu	uuuu uuuu
LCDSE1	11Dh	0000 0000	uuuu uuuu	uuuu uuuu
LCDSE2(6)	11Eh	0000 0000	uuuu uuuu	uuuu uuuu
TRISF(7)	185h	1111 1111	1111 1111	uuuu uuuu
TRISG(7)	187h	--11 1111	--11 1111	--uu uuuu
PORTF(7)	188h	xxxx xxxx	0000 0000	uuuu uuuu
PORTG(7)	189h	--xx xxxx	--00 0000	--uu uuuu
LCDDATA12(7)	190h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA13(7)	191h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA14(7)	192h	---- --xx	---- --uu	---- --uu
LCDDATA15(7)	193h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA16(7)	194h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA17(7)	195h	---- --xx	---- --uu	---- --uu
LCDDATA18(7)	196h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA19(7)	197h	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA20(7)	198h	---- --xx	---- --uu	---- --uu
LCDDATA21(7)	199h	xxxx xxxx	uuuu uuuu	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0', q = value depends on condition.

Note 1: If VDD goes too low, Power-on Reset will be activated and registers will be affected differently.

2: One or more bits in INTCON and/or PIR1 will be affected (to cause wake-up).

3: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

4: See Table 16-5 for Reset value for specific condition.

5: If Reset was due to brown-out, then bit 0 = 0. All other Resets will cause bit 0 = u.

6: PIC16F914/917 and PIC16F946 only.

7: PIC16F946 only.

TABLE 16-4: INITIALIZATION CONDITION FOR REGISTERS (CONTINUED)

Register	Address	Power-on Reset	• $\overline{\text{MCLR}}$ Reset • WDT Reset • Brown-out Reset(1)	• Wake-up from Sleep through interrupt • Wake-up from Sleep through WDT time-out
LCDDATA22(7)	19Ah	xxxx xxxx	uuuu uuuu	uuuu uuuu
LCDDATA23(7)	19Bh	---- --xx	---- --uu	---- --uu
LCDSE3(7)	19Ch	0000 0000	uuuu uuuu	uuuu uuuu
LCDSE4(7)	19Dh	0000 0000	uuuu uuuu	uuuu uuuu
LCDSE5(7)	19Eh	---- --00	---- --uu	---- --uu
EECON1	18Ch	x--- x000	u--- q000	u--- uuuu

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0', q = value depends on condition.

Note 1: If VDD goes too low, Power-on Reset will be activated and registers will be affected differently.

2: One or more bits in INTCON and/or PIR1 will be affected (to cause wake-up).

3: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

4: See Table 16-5 for Reset value for specific condition.

5: If Reset was due to brown-out, then bit 0 = 0. All other Resets will cause bit 0 = u.

6: PIC16F914/917 and PIC16F946 only.

7: PIC16F946 only.

TABLE 16-5: INITIALIZATION CONDITION FOR SPECIAL REGISTERS

Condition	Program Counter	STATUS Register	PCON Register
Power-on Reset	0000h	0001 1xxx	---1 --0x
$\overline{\text{MCLR}}$ Reset during normal operation	0000h	000u uuuu	---u --uu
$\overline{\text{MCLR}}$ Reset during Sleep	0000h	0001 0uuu	---u --uu
WDT Reset	0000h	0000 uuuu	---u --uu
WDT Wake-up	PC + 1	uuu0 0uuu	---u --uu
Brown-out Reset	0000h	0001 1uuu	---1 --10
Interrupt Wake-up from Sleep	PC + 1(1)	uuu1 0uuu	---u --uu

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0'.

Note 1: When the wake-up is due to an interrupt and Global Interrupt Enable bit, GIE, is set, the PC is loaded with the interrupt vector (0004h) after execution of PC + 1.

16.3 Interrupts

The PIC16F91X/946 has multiple sources of interrupt:

- External Interrupt RB0/INT/SEG0
- TMR0 Overflow Interrupt
- PORTB Change Interrupts
- 2 Comparator Interrupts
- A/D Interrupt
- Timer1 Overflow Interrupt
- EEPROM Data Write Interrupt
- Fail-Safe Clock Monitor Interrupt
- LCD Interrupt
- PLVD Interrupt
- USART Receive and Transmit interrupts
- CCP1 and CCP2 Interrupts
- Timer2 Interrupt

The Interrupt Control (INTCON), Peripheral Interrupt Request 1 (PIR1) and Peripheral Interrupt Request 2 (PIR2) registers record individual interrupt requests in flag bits. The INTCON register also has individual and global interrupt enable bits.

A Global Interrupt Enable bit, GIE of the INTCON register, enables (if set) all unmasked interrupts, or disables (if cleared) all interrupts. Individual interrupts can be disabled through their corresponding enable bits in the INTCON, PIE1 and PIE2 registers. GIE is cleared on Reset.

The Return from Interrupt instruction, `RETFIE`, exits the interrupt routine, as well as sets the GIE bit, which re-enables unmasked interrupts.

The following interrupt flags are contained in the INTCON register:

- INT Pin Interrupt
- PORTB Change Interrupt
- TMR0 Overflow Interrupt

The peripheral interrupt flags are contained in the special registers, PIR1 and PIR2. The corresponding interrupt enable bit are contained in the special registers, PIE1 and PIE2.

The following interrupt flags are contained in the PIR1 register:

- EEPROM Data Write Interrupt
- A/D Interrupt
- USART Receive and Transmit Interrupts
- Timer1 Overflow Interrupt
- CCP1 Interrupt
- SSP Interrupt
- Timer2 Interrupt

The following interrupt flags are contained in the PIR2 register:

- Fail-Safe Clock Monitor Interrupt
- Comparator 1 and 2 Interrupts
- LCD Interrupt
- PLVD Interrupt
- CCP2 Interrupt

When an interrupt is serviced:

- The GIE is cleared to disable any further interrupt.
- The return address is pushed onto the stack.
- The PC is loaded with 0004h.

For external interrupt events, such as the INT pin or PORTB change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends upon when the interrupt event occurs (see Figure 16-8). The latency is the same for one or two-cycle instructions. Once in the Interrupt Service Routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid multiple interrupt requests.

Note 1:	Individual interrupt flag bits are set, regardless of the status of their corresponding mask bit or the GIE bit.
2:	When an instruction that clears the GIE bit is executed, any interrupts that were pending for execution in the next cycle are ignored. The interrupts, which were ignored, are still pending to be serviced when the GIE bit is set again.

For additional information on how a module generates an interrupt, refer to the respective peripheral section.

Note:	The ANSEL and CMCON0 registers must be initialized to configure an analog channel as a digital input. Pins configured as analog inputs will read '0'. Also, if a LCD output function is active on an external interrupt pin, that interrupt function will be disabled.

16.3.1 RB0/INT/SEG0 INTERRUPT

External interrupt on RB0/INT/SEG0 pin is edge-triggered; either rising if the INTEDG bit of the OPTION register is set, or falling, if the INTEDG bit is clear. When a valid edge appears on the RB0/INT/SEG0 pin, the INTF bit of the INTCON register is set. This interrupt can be disabled by clearing the INTE control bit of the INTCON register. The INTF bit must be cleared in software in the Interrupt Service Routine before re-enabling this interrupt. The RB0/INT/SEG0 interrupt can wake-up the processor from Sleep if the INTE bit was set prior to going into Sleep. The status of the GIE bit decides whether or not the processor branches to the interrupt vector following wake-up (0004h). See **Section 16.5 "Power-Down Mode (Sleep)"** for details on Sleep and Figure 16-10 for timing of wake-up from Sleep through RB0/INT/SEG0 interrupt.

16.3.2 TMR0 INTERRUPT

An overflow (FFh → 00h) in the TMR0 register will set the T0IF bit of the INTCON register. The interrupt can be enabled/disabled by setting/clearing T0IE bit of the INTCON register. See **Section 5.0 "Timer0 Module"** for operation of the Timer0 module.

16.3.3 PORTB INTERRUPT

An input change on PORTB change sets the RBIF bit of the INTCON register. The interrupt can be enabled/disabled by setting/clearing the RBIE bit of the INTCON register. Plus, individual pins can be configured through the IOCB register.

Note:	If a change on the I/O pin should occur when the read operation is being executed (start of the Q2 cycle), then the RBIF interrupt flag may not get set.

FIGURE 16-7: INTERRUPT LOGIC

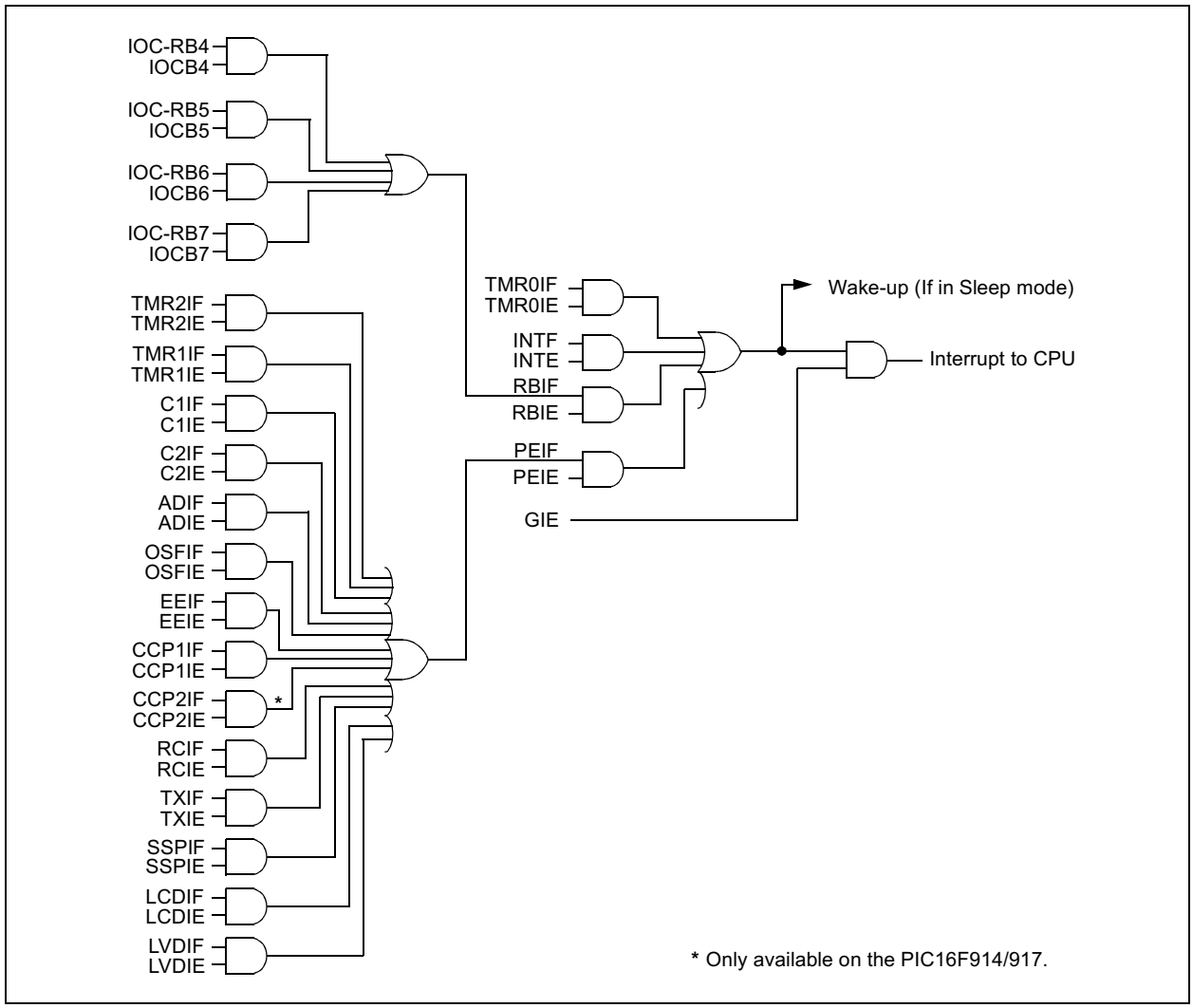
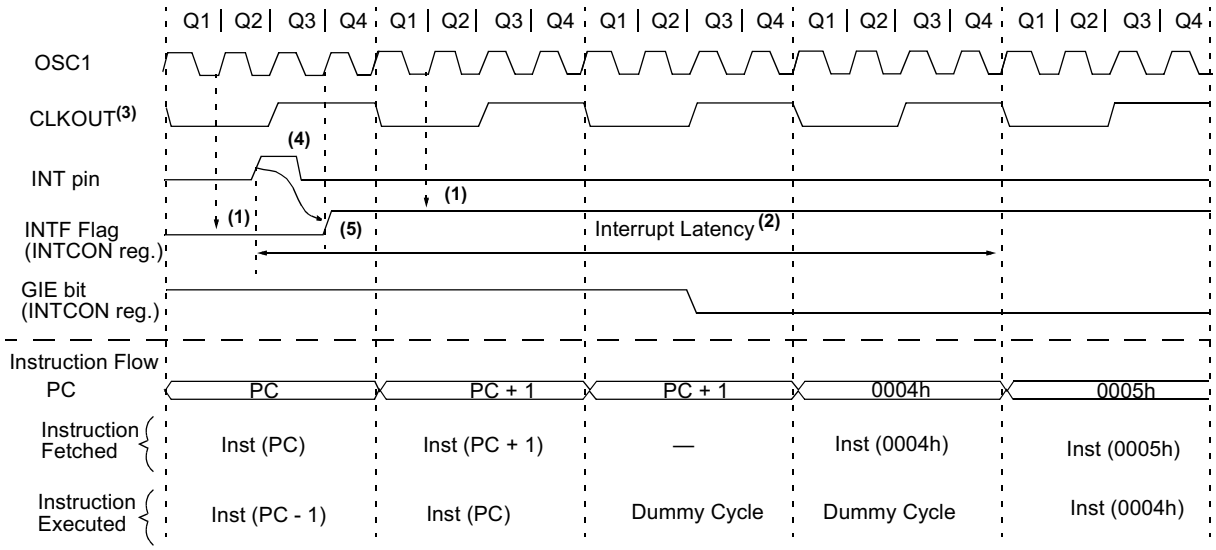


FIGURE 16-8: INT PIN INTERRUPT TIMING



Note 1: INTF flag is sampled here (every Q1).

2: Asynchronous interrupt latency = 3-4 TCY. Synchronous latency = 3 TCY, where TCY = instruction cycle time. Latency is the same whether Inst (PC) is a single cycle or a 2-cycle instruction.

3: CLKOUT is available only in INTOSC and RC Oscillator modes.

4: For minimum width of INT pulse, refer to AC specifications in **Section 19.0 "Electrical Specifications"**.

5: INTF is enabled to be set any time during the Q4-Q1 cycles.

TABLE 16-6: SUMMARY OF INTERRUPT REGISTERS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIR1	EEIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIR2	OSFIF	C2IF	C1IF	LCDIF	—	LVDIF	—	CCP2IF	0000 -0-0	0000 -0-0
PIE1	EEIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIE2	OSFIE	C2IE	C1IE	LCDIE	—	LVDIE	—	CCP2IE	0000 -0-0	0000 -0-0

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by the Interrupt Module.

16.3.4 CONTEXT SAVING DURING INTERRUPTS

During an interrupt, only the return PC value is saved on the stack. Typically, users may wish to save key registers during an interrupt (e.g., W and STATUS registers). This must be implemented in software.

Since the lower 16 bytes of all banks are common in the PIC16F91X/946 (see Figure 2-3), temporary holding registers, W_TEMP and STATUS_TEMP, should be placed in here. These 16 locations do not require banking and therefore, make it easier to context save and restore. The same code shown in Example 16-1 can be used to:

- Store the W register
- Store the STATUS register
- Execute the ISR code
- Restore the STATUS register (Bank Select bits)
- Restore the W register

Note:	The microcontroller does not normally require saving the PCLATH register unless it is modified in code either directly or via the `pagesel` macro. Then, the PCLATH register must be saved at the beginning of the ISR, managed for `CALL`s and `GOTO`s in the ISR and restored when the ISR is complete to ensure correct program flow.

EXAMPLE 16-1: SAVING STATUS AND W REGISTERS IN RAM

```
MOVWF   W_TEMP              ;Copy W to TEMP register
SWAPF   STATUS,W            ;Swap status to be saved into W
CLRF    STATUS              ;bank 0, regardless of current bank, Clears IRP,RP1,RP0
MOVWF   STATUS_TEMP         ;Save status to bank zero STATUS_TEMP register
:
:(ISR)                      ;Insert user code here
:
SWAPF   STATUS_TEMP,W       ;Swap STATUS_TEMP register into W
                            ;(sets bank to original state)
MOVWF   STATUS              ;Move W into STATUS register
SWAPF   W_TEMP,F            ;Swap W_TEMP
SWAPF   W_TEMP,W            ;Swap W_TEMP into W
```

16.4 Watchdog Timer (WDT)

For PIC16F91X/946, the WDT has been modified from previous PIC16F devices. The new WDT is code and functionally compatible with previous PIC16F WDT modules and adds a 16-bit prescaler to the WDT. This allows the user to have a scaled value for the WDT and TMR0 at the same time. In addition, the WDT time-out value can be extended to 268 seconds. WDT is cleared under certain conditions described in Table 16-7.

16.4.1 WDT OSCILLATOR

The WDT derives its time base from the 31 kHz LFINTOSC. The LTS bit does not reflect that the LFINTOSC is enabled.

The value of WDTCON is '---0 1000' on all Resets. This gives a nominal time base of 16 ms, which is compatible with the time base generated with previous PIC16F microcontroller versions.

Note:	When the Oscillator Start-up Timer (OST) is invoked, the WDT is held in Reset, because the WDT Ripple Counter is used by the OST to perform the oscillator delay count. When the OST count has expired, the WDT will begin counting (if enabled).

A new prescaler has been added to the path between the INTOSC and the multiplexers used to select the path for the WDT. This prescaler is 16 bits and can be programmed to divide the INTOSC by 32 to 65536, giving the WDT a nominal range of 1 ms to 268s.

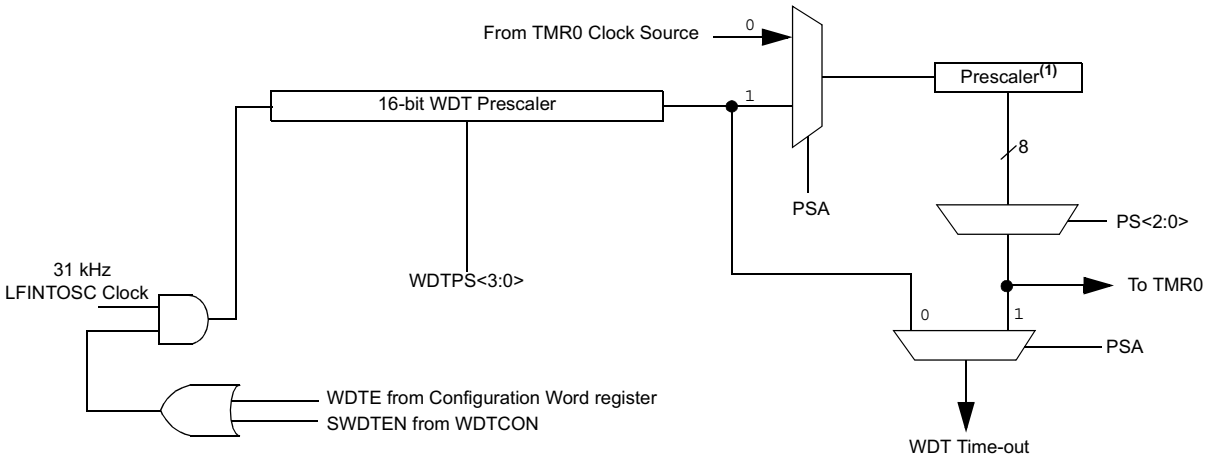
16.4.2 WDT CONTROL

The WDTE bit is located in the Configuration Word register. When set, the WDT runs continuously.

When the WDTE bit in the Configuration Word register is set, the SWDTEN bit of the WDTCON register has no effect. If WDTE is clear, then the SWDTEN bit can be used to enable and disable the WDT. Setting the bit will enable it and clearing the bit will disable it.

The PSA and PS<2:0> bits of the OPTION register have the same function as in previous versions of the PIC16F family of microcontrollers. See **Section 5.0 "Timer0 Module"** for more information.

FIGURE 16-9: WATCHDOG TIMER BLOCK DIAGRAM



Note 1: This is the shared Timer0/WDT prescaler. See **Section 5.4 "Prescaler"** for more information.

TABLE 16-7: WDT STATUS

Conditions	WDT
WDTE = 0	Cleared
CLRWDT Command	
Oscillator Fail Detected	
Exit Sleep + System Clock = T1OSC, EXTRC, INTOSC, EXTCLK	
Exit Sleep + System Clock = XT, HS, LP	Cleared until the end of OST

REGISTER 16-2: WDTCON – WATCHDOG TIMER CONTROL REGISTER (ADDRESS: 105h)

U-0	U-0	U-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0
—	—	—	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN
bit 7							bit 0

bit 7-5 **Unimplemented:** Read as '0'

bit 4-1 **WDTPS<3:0>:** Watchdog Timer Period Select bits

Bit Value = Prescale Rate

0000 = 1:32
0001 = 1:64
0010 = 1:128
0011 = 1:256
0100 = 1:512 (Reset value)
0101 = 1:1024
0110 = 1:2048
0111 = 1:4096
1000 = 1:8192
1001 = 1:16384
1010 = 1:32768
1011 = 1:65536
1100 = reserved
1101 = reserved
1110 = reserved
1111 = reserved

bit 0 **SWDTEN:** Software Enable or Disable the Watchdog Timer bit(1)

1 = WDT is turned on
0 = WDT is turned off (Reset value)

Note 1: If WDTE Configuration bit = 1, then WDT is always enabled, irrespective of this control bit. If WDTE Configuration bit = 0, then it is possible to turn WDT on/off with this control bit.

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
- n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

TABLE 16-8: SUMMARY OF WATCHDOG TIMER REGISTERS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CONFIG	CPD	CP	MCLRE	$\overline{\text{PWRTE}}$	WDTE	FOSC2	FOSC1	FOSC0
OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
WDTCON	—	—	—	WDTPS3	WDTPS2	WSTPS1	WDTPS0	SWDTEN

Legend: Shaded cells are not used by the Watchdog Timer.

Note 1: See Register 16-1 for operation of all Configuration Word register bits.

16.5 Power-Down Mode (Sleep)

The Power-down mode is entered by executing a SLEEP instruction.

If the Watchdog Timer is enabled:

- WDT will be cleared but keeps running.
- $\overline{PD}$ bit in the STATUS register is cleared.
- $\overline{TO}$ bit is set.
- Oscillator driver is turned off.
- Timer1 oscillator is unaffected
- I/O ports maintain the status they had before SLEEP was executed (driving high, low or high-impedance).

For lowest current consumption in this mode, all I/O pins should be either at VDD or VSS, with no external circuitry drawing current from the I/O pin, and the comparators and CVREF should be disabled. I/O pins that are high-impedance inputs should be pulled high or low externally to avoid switching currents caused by floating inputs. The T0CKI input should also be at VDD or VSS for lowest current consumption. The contribution from on-chip pull-ups on PORTB should be considered.

The $\overline{MCLR}$ pin must be at a logic high level.

Note:	It should be noted that a Reset generated by a WDT time-out does not drive $\overline{MCLR}$ pin low.

16.5.1 WAKE-UP FROM SLEEP

The device can wake-up from Sleep through one of the following events:

1. External Reset input on $\overline{MCLR}$ pin.
2. Watchdog Timer wake-up (if WDT was enabled).
3. Interrupt from RB0/INT/SEG0 pin, PORTB change or a peripheral interrupt.

The first event will cause a device Reset. The two latter events are considered a continuation of program execution. The $\overline{TO}$ and $\overline{PD}$ bits in the STATUS register can be used to determine the cause of device Reset. The $\overline{PD}$ bit, which is set on power-up, is cleared when Sleep is invoked. $\overline{TO}$ bit is cleared if WDT wake-up occurred.

The following peripheral interrupts can wake the device from Sleep:

1. TMR1 Interrupt. Timer1 must be operating as an asynchronous counter.
2. USART Receive Interrupt (Sync Slave mode only)
3. A/D conversion (when A/D clock source is RC)
4. EEPROM write operation completion
5. Comparator output changes state
6. Interrupt-on-change
7. External Interrupt from INT pin
8. PLVD Interrupt
9. LCD Interrupt (if running during Sleep)

Other peripherals cannot generate interrupts since during Sleep, no on-chip clocks are present.

When the SLEEP instruction is being executed, the next instruction (PC + 1) is pre-fetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be set (enabled). Wake-up is regardless of the state of the GIE bit. If the GIE bit is clear (disabled), the device continues execution at the instruction after the SLEEP instruction. If the GIE bit is set (enabled), the device executes the instruction after the SLEEP instruction, then branches to the interrupt address (0004h). In cases where the execution of the instruction following SLEEP is not desirable, the user should have a NOP after the SLEEP instruction.

Note:	If the global interrupts are disabled (GIE is cleared), but any interrupt source has both its interrupt enable bit and the corresponding interrupt flag bits set, the device will immediately wake-up from Sleep. The SLEEP instruction is completely executed.

The WDT is cleared when the device wakes up from Sleep, regardless of the source of wake-up.

16.5.2 WAKE-UP USING INTERRUPTS

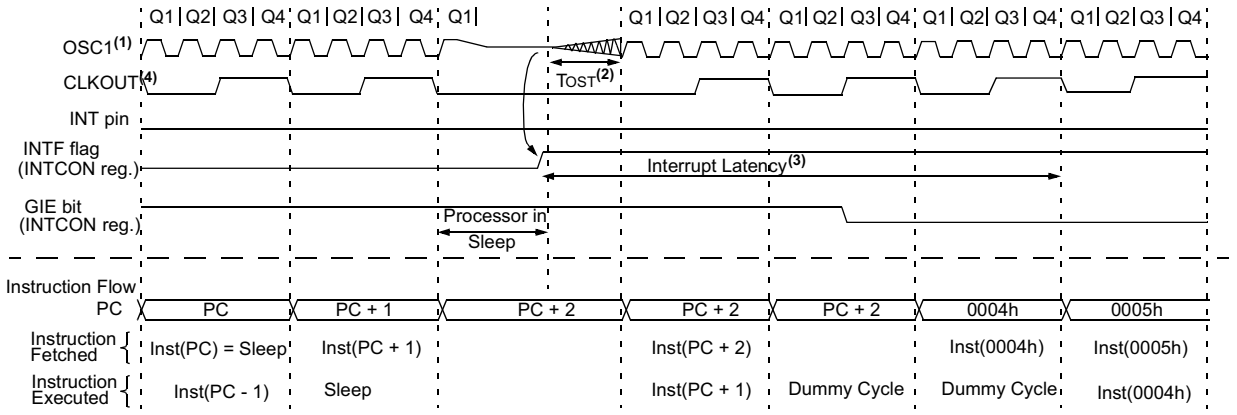
When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the execution of a SLEEP instruction, the SLEEP instruction will complete as a NOP. Therefore, the WDT and WDT prescaler and postscaler (if enabled) will not be cleared, the $\overline{TO}$ bit will not be set and the $\overline{PD}$ bit will not be cleared.
- If the interrupt occurs **during or after** the execution of a SLEEP instruction, the device will immediately wake-up from Sleep. The SLEEP instruction will be completely executed before the wake-up. Therefore, the WDT and WDT prescaler and postscaler (if enabled) will be cleared, the $\overline{TO}$ bit will be set and the $\overline{PD}$ bit will be cleared.

Even if the flag bits were checked before executing a SLEEP instruction, it may be possible for flag bits to become set before the SLEEP instruction completes. To determine whether a SLEEP instruction executed, test the $\overline{PD}$ bit. If the $\overline{PD}$ bit is set, the SLEEP instruction was executed as a NOP.

To ensure that the WDT is cleared, a CLRWDT instruction should be executed before a SLEEP instruction.

FIGURE 16-10: WAKE-UP FROM SLEEP THROUGH INTERRUPT



Note 1: XT, HS or LP Oscillator mode assumed.
2: TOST = 1024 TOSC (drawing not to scale). This delay does not apply to EC and RC Oscillator modes.
3: GIE = 1 assumed. In this case after wake-up, the processor jumps to 0004h. If GIE = 0, execution will continue in-line.
4: CLKOUT is not available in XT, HS, LP or EC Oscillator modes, but shown here for timing reference.

16.6 Code Protection

If the code protection bit(s) have not been programmed, the on-chip program memory can be read out using ICSP for verification purposes.

Note:	The entire data EEPROM and Flash program memory will be erased when the code protection is turned off. See the "*PIC16F91X/946 Memory Programming Specification*" (DS41244) for more information.

16.7 ID Locations

Four memory locations (2000h-2003h) are designated as ID locations where the user can store checksum or other code identification numbers. These locations are not accessible during normal execution, but are readable and writable during Program/Verify mode. Only the Least Significant 7 bits of the ID locations are used.

16.8 In-Circuit Serial Programming

The PIC16F91X/946 microcontrollers can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data and three other lines for:

- power
- ground
- programming voltage

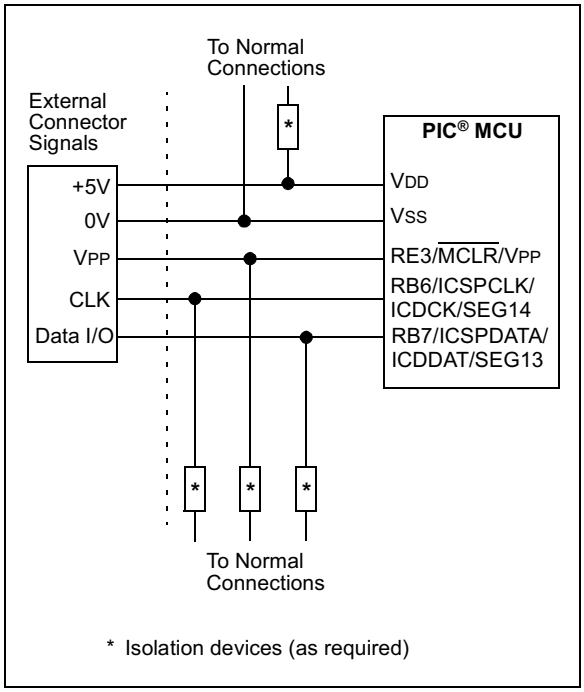
This allows customers to manufacture boards with unprogrammed devices and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

The device is placed into a Program/Verify mode by holding the RB7/ICSPDAT/ICDDAT/SEG13 and RB6/ICSPCLK/ICDCK/SEG14 pins low, while raising the $\overline{\text{MCLR}}$ (V_{PP}) pin from V_{IL} to V_{IHH}. See "*PIC16F91X/946 Memory Programming Specification*" (DS41244) for more information. RB7 becomes the programming data and the RB6 becomes the programming clock. Both RB7 and RB6 are Schmitt Trigger inputs in this mode.

After Reset, to place the device into Program/Verify mode, the Program Counter (PC) is at location 0000h. A 6-bit command is then supplied to the device. Depending on the command, 14 bits of program data are then supplied to or from the device, depending on whether the command was a load or a read. For complete details of serial programming, please refer to the "*PIC16F91X/946 Memory Programming Specification*" (DS41244).

A typical In-Circuit Serial Programming connection is shown in Figure 16-11.

FIGURE 16-11: TYPICAL IN-CIRCUIT SERIAL PROGRAMMING CONNECTION



16.9 In-Circuit Debugger

When the debug bit in the Configuration Word register is programmed to a '0', the In-Circuit Debugger functionality is enabled. This function allows simple debugging functions when used with MPLAB® ICD 2. When the microcontroller has this feature enabled, some of the resources are not available for general use. See Table 16-9 for more detail.

Note: The user's application must have the circuitry required to support ICD functionality. Once the ICD circuitry is enabled, normal device pin functions on RB6/ICSPCLK/ICDCK/SEG14 and RB7/ICSPDAT/ICDDAT/SEG13 will not be usable. The ICD circuitry uses these pins for communication with the ICD2 external debugger.

For more information, see "*Using MPLAB® ICD 2*" (DS51265), available on Microchip's web site (www.microchip.com).

16.9.1 ICD PINOUT

The devices in the PIC16F91X/946 family carry the circuitry for the In-Circuit Debugger on-chip and on existing device pins. This eliminates the need for a separate die or package for the ICD device. The pinout for the ICD device is the same as the devices (see **Section 1.0 "Device Overview"** for complete pinout and pin descriptions). Table 16-9 shows the location and function of the ICD related pins on the 28 and 40-pin devices.

TABLE 16-9: PIC16F91X/946-ICD PIN DESCRIPTIONS

Pin Numbers			Name	Type	Pull-up	Description
PDIP		TQFP				
PIC16F914/917	PIC16F913/916	PIC16F946				
40	28	24	ICDDATA	TTL	—	In Circuit Debugger Bidirectional data
39	27	23	ICDCLK	ST	—	In Circuit Debugger Bidirectional clock
1	1	36	$\overline{MCLR}$/VPP	HV	—	Programming voltage
11,32	20	10, 19, 38, 51	VDD	P	—	Power
12,31	8,19	9, 20, 41, 56	VSS	P	—	Ground
—	—	26	AVDD	P	—	Analog power
—	—	25	AVSS	P	—	Analog ground

Legend: TTL = TTL input buffer, ST = Schmitt Trigger input buffer, P = Power, HV = High Voltage

NOTES:

17.0 INSTRUCTION SET SUMMARY

The PIC16F913/914/916/917/946 instruction set is highly orthogonal and is comprised of three basic categories:

- **Byte-oriented** operations
- **Bit-oriented** operations
- **Literal and control** operations

Each PIC16 instruction is a 14-bit word divided into an **opcode,** which specifies the instruction type and one or more **operands,** which further specify the operation of the instruction. The formats for each of the categories is presented in Figure 17-1, while the various opcode fields are summarized in Table 17-1.

Table 17-2 lists the instructions recognized by the MPASM™ assembler.

For **byte-oriented** instructions, 'f' represents a file register designator and 'd' represents a destination designator. The file register designator specifies which file register is to be used by the instruction.

The destination designator specifies where the result of the operation is to be placed. If 'd' is zero, the result is placed in the W register. If 'd' is one, the result is placed in the file register specified in the instruction.

For **bit-oriented** instructions, 'b' represents a bit field designator, which selects the bit affected by the operation, while 'f' represents the address of the file in which the bit is located.

For **literal and control** operations, 'k' represents an 8-bit or 11-bit constant, or literal value.

One instruction cycle consists of four oscillator periods; for an oscillator frequency of 4 MHz, this gives a nominal instruction execution time of 1 µs. All instructions are executed within a single instruction cycle, unless a conditional test is true, or the program counter is changed as a result of an instruction. When this occurs, the execution takes two instruction cycles, with the second cycle executed as a NOP.

All instruction examples use the format '0xhh' to represent a hexadecimal number, where 'h' signifies a hexadecimal digit.

17.1 Read-Modify-Write Operations

Any instruction that specifies a file register as part of the instruction performs a Read-Modify-Write (R-M-W) operation. The register is read, the data is modified, and the result is stored according to either the instruction, or the destination designator 'd'. A read operation is performed on a register even if the instruction writes to that register.

For example, a CLRF PORTA instruction will read PORTA, clear all the data bits, then write the result back to PORTA. This example would have the unintended consequence of clearing the condition that set the RBIF flag.

TABLE 17-1: OPCODE FIELD DESCRIPTIONS

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1). The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1.
PC	Program Counter
$\overline{TO}$	Time-out bit
C	Carry bit
DC	Digit carry bit
Z	Zero bit
$\overline{PD}$	Power-down bit

FIGURE 17-1: GENERAL FORMAT FOR INSTRUCTIONS

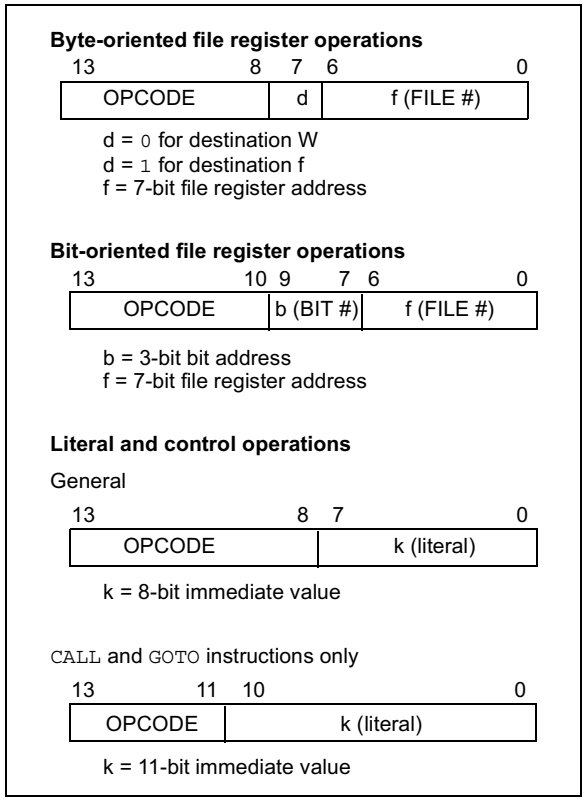


TABLE 17-2: PIC16F913/914/916/917/946 INSTRUCTION SET

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb LSb	Status Affected	Notes
BYTE-ORIENTED FILE REGISTER OPERATIONS						
ADDWF	f, d	Add W and f	1	00 0111 dfff ffff	C, DC, Z	**1, 2**
ANDWF	f, d	AND W with f	1	00 0101 dfff ffff	Z	**1, 2**
CLRF	f	Clear f	1	00 0001 lfff ffff	Z	**2**
CLRW	–	Clear W	1	00 0001 0xxx xxxx	Z	
COMF	f, d	Complement f	1	00 1001 dfff ffff	Z	**1, 2**
DECF	f, d	Decrement f	1	00 0011 dfff ffff	Z	**1, 2**
DECFSZ	f, d	Decrement f, Skip if 0	1**(2)**	00 1011 dfff ffff		**1, 2, 3**
INCF	f, d	Increment f	1	00 1010 dfff ffff	Z	**1, 2**
INCFSZ	f, d	Increment f, Skip if 0	1**(2)**	00 1111 dfff ffff		**1, 2, 3**
IORWF	f, d	Inclusive OR W with f	1	00 0100 dfff ffff	Z	**1, 2**
MOVF	f, d	Move f	1	00 1000 dfff ffff	Z	**1, 2**
MOVWF	f	Move W to f	1	00 0000 lfff ffff		
NOP	–	No Operation	1	00 0000 0xx0 0000		
RLF	f, d	Rotate Left f through Carry	1	00 1101 dfff ffff	C	**1, 2**
RRF	f, d	Rotate Right f through Carry	1	00 1100 dfff ffff	C	**1, 2**
SUBWF	f, d	Subtract W from f	1	00 0010 dfff ffff	C, DC, Z	**1, 2**
SWAPF	f, d	Swap nibbles in f	1	00 1110 dfff ffff		**1, 2**
XORWF	f, d	Exclusive OR W with f	1	00 0110 dfff ffff	Z	**1, 2**
BIT-ORIENTED FILE REGISTER OPERATIONS						
BCF	f, b	Bit Clear f	1	01 00bb bfff ffff		**1, 2**
BSF	f, b	Bit Set f	1	01 01bb bfff ffff		**1, 2**
BTFSC	f, b	Bit Test f, Skip if Clear	1 **(2)**	01 10bb bfff ffff		**3**
BTFSS	f, b	Bit Test f, Skip if Set	1 **(2)**	01 11bb bfff ffff		**3**
LITERAL AND CONTROL OPERATIONS						
ADDLW	k	Add literal and W	1	11 111x kkkk kkkk	C, DC, Z	
ANDLW	k	AND literal with W	1	11 1001 kkkk kkkk	Z	
CALL	k	Call Subroutine	2	10 0kkk kkkk kkkk		
CLRWDT	–	Clear Watchdog Timer	1	00 0000 0110 0100	$\overline{TO}$, $\overline{PD}$	
GOTO	k	Go to address	2	10 1kkk kkkk kkkk		
IORLW	k	Inclusive OR literal with W	1	11 1000 kkkk kkkk	Z	
MOVLW	k	Move literal to W	1	11 00xx kkkk kkkk		
RETFIE	–	Return from interrupt	2	00 0000 0000 1001		
RETLW	k	Return with literal in W	2	11 01xx kkkk kkkk		
RETURN	–	Return from Subroutine	2	00 0000 0000 1000		
SLEEP	–	Go into Standby mode	1	00 0000 0110 0011	$\overline{TO}$, $\overline{PD}$	
SUBLW	k	Subtract W from literal	1	11 110x kkkk kkkk	C, DC, Z	
XORLW	k	Exclusive OR literal with W	1	11 1010 kkkk kkkk	Z	

Note 1: When an I/O register is modified as a function of itself (e.g., `MOVF GPIO, 1`), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

2: If this instruction is executed on the TMR0 register (and where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 module.

3: If the Program Counter (PC) is modified, or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

17.2 Instruction Descriptions

ADDLW **Add literal and W**

Syntax:	[*label*] ADDLW k
Operands:	0 ≤ k ≤ 255
Operation:	(W) + k → (W)
Status Affected:	C, DC, Z
Description:	The contents of the W register are added to the eight-bit literal 'k' and the result is placed in the W register.

ADDWF **Add W and f**

Syntax:	[*label*] ADDWF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(W) + (f) → (destination)
Status Affected:	C, DC, Z
Description:	Add the contents of the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

ANDLW **AND literal with W**

Syntax:	[*label*] ANDLW k
Operands:	0 ≤ k ≤ 255
Operation:	(W) .AND. (k) → (W)
Status Affected:	Z
Description:	The contents of W register are AND'ed with the eight-bit literal 'k'. The result is placed in the W register.

ANDWF **AND W with f**

Syntax:	[*label*] ANDWF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(W) .AND. (f) → (destination)
Status Affected:	Z
Description:	AND the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

BCF **Bit Clear f**

Syntax:	[*label*] BCF f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b ≤ 7
Operation:	0 → (f<b>)
Status Affected:	None
Description:	Bit 'b' in register 'f' is cleared.

BSF **Bit Set f**

Syntax:	[*label*] BSF f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b ≤ 7
Operation:	1 → (f<b>)
Status Affected:	None
Description:	Bit 'b' in register 'f' is set.

BTFSC **Bit Test f, Skip if Clear**

Syntax:	[*label*] BTFSC f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b ≤ 7
Operation:	skip if (f<b>) = 0
Status Affected:	None
Description:	If bit 'b' in register 'f' is '1', the next instruction is executed. If bit 'b', in register 'f', is '0', the next instruction is discarded, and a NOP is executed instead, making this a 2-cycle instruction.

BTFSS **Bit Test f, Skip if Set**

Syntax:	[*label*] BTFSS f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b < 7
Operation:	skip if (f<b>) = 1
Status Affected:	None
Description:	If bit 'b' in register 'f' is '0', the next instruction is executed. If bit 'b' is '1', then the next instruction is discarded and a NOP is executed instead, making this a 2-cycle instruction.

CALL **Call Subroutine**

Syntax:	[*label*] CALL k
Operands:	0 ≤ k ≤ 2047
Operation:	(PC)+ 1→ TOS, k → PC<10:0>, (PCLATH<4:3>) → PC<12:11>
Status Affected:	None
Description:	Call Subroutine. First, return address (PC + 1) is pushed onto the stack. The eleven-bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a two-cycle instruction.

CLRF **Clear f**

Syntax:	[*label*] CLRF f
Operands:	0 ≤ f ≤ 127
Operation:	00h → (f) 1 → Z
Status Affected:	Z
Description:	The contents of register 'f' are cleared and the Z bit is set.

CLRW **Clear W**

Syntax:	[*label*] CLRW
Operands:	None
Operation:	00h → (W) 1 → Z
Status Affected:	Z
Description:	W register is cleared. Zero bit (Z) is set.

CLRWDT **Clear Watchdog Timer**

Syntax:	[*label*] CLRWDT
Operands:	None
Operation:	00h → WDT 0 → WDT prescaler, 1 → $\overline{TO}$ 1 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	CLRWDT instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set.

COMF **Complement f**

Syntax:	[*label*] COMF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	($\bar{f}$) → (destination)
Status Affected:	Z
Description:	The contents of register 'f' are complemented. If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f'.

DECF **Decrement f**

Syntax:	[*label*] DECF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f) - 1 → (destination)
Status Affected:	Z
Description:	Decrement register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

DECFSZ **Decrement f, Skip if 0**

Syntax: [*label*] DECFSZ f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) - 1 → (destination);
skip if result = 0

Status Affected: None

Description: The contents of register 'f' are decremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.
If the result is '1', the next instruction is executed. If the result is '0', then a NOP is executed instead, making it a 2-cycle instruction.

GOTO **Unconditional Branch**

Syntax: [*label*] GOTO k

Operands: $0 \le k \le 2047$

Operation: k → PC<10:0>
PCLATH<4:3> → PC<12:11>

Status Affected: None

Description: GOTO is an unconditional branch. The eleven-bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. GOTO is a two-cycle instruction.

INCF **Increment f**

Syntax: [*label*] INCF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) + 1 → (destination)

Status Affected: Z

Description: The contents of register 'f' are incremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

INCFSZ **Increment f, Skip if 0**

Syntax: [*label*] INCFSZ f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) + 1 → (destination),
skip if result = 0

Status Affected: None

Description: The contents of register 'f' are incremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.
If the result is '1', the next instruction is executed. If the result is '0', a NOP is executed instead, making it a 2-cycle instruction.

IORLW **Inclusive OR literal with W**

Syntax: [*label*] IORLW k

Operands: $0 \le k \le 255$

Operation: (W) .OR. k → (W)

Status Affected: Z

Description: The contents of the W register are OR'ed with the eight-bit literal 'k'. The result is placed in the W register.

IORWF **Inclusive OR W with f**

Syntax: [*label*] IORWF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (W) .OR. (f) → (destination)

Status Affected: Z

Description: Inclusive OR the W register with register 'f'. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

MOVF **Move f**

Syntax: [*label*] MOVF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) → (dest)

Status Affected: Z

Description: The contents of register f is moved to a destination dependent upon the status of d. If d = 0, destination is W register. If d = 1, the destination is file register f itself. d = 1 is useful to test a file register since status flag Z is affected.

Words: 1

Cycles: 1

Example:
```
MOVF    FSR, 0
```
After Instruction
W = value in FSR register
Z = 1

MOVWF **Move W to f**

Syntax: [*label*] MOVWF f

Operands: $0 \le f \le 127$

Operation: (W) → (f)

Status Affected: None

Description: Move data from W register to register 'f'.

Words: 1

Cycles: 1

Example:
```
MOVW    OPTION
F
```
Before Instruction
OPTION = 0xFF
W = 0x4F
After Instruction
OPTION = 0x4F
W = 0x4F

MOVLW **Move literal to W**

Syntax: [*label*] MOVLW k

Operands: $0 \le k \le 255$

Operation: k → (W)

Status Affected: None

Description: The eight-bit literal 'k' is loaded into W register. The "don't cares" will assemble as '0's.

Words: 1

Cycles: 1

Example:
```
MOVLW   0x5A
```
After Instruction
W = 0x5A

NOP **No Operation**

Syntax: [*label*] NOP

Operands: None

Operation: No operation

Status Affected: None

Description: No operation.

Words: 1

Cycles: 1

Example:
```
NOP
```

RETFIE	Return from Interrupt
Syntax:	[*label*] RETFIE
Operands:	None
Operation:	TOS → PC, 1 → GIE
Status Affected:	None
Description:	Return from Interrupt. Stack is POPed and Top-of-Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a two-cycle instruction.
Words:	1
Cycles:	2
Example:	RETFIE After Interrupt PC = TOS GIE = 1

RETLW	Return with literal in W
Syntax:	[*label*] RETLW k
Operands:	0 ≤ k ≤ 255
Operation:	k → (W); TOS → PC
Status Affected:	None
Description:	The W register is loaded with the eight bit literal ‘k’. The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.
Words:	1
Cycles:	2
Example:	CALL TABLE;W contains table ;offset value TABLE ;W now has table value • • ADDWF PC ;W = offset RETLW k1 ;Begin table RETLW k2 ; • • • RETLW kn ; End of table Before Instruction W = 0x07 After Instruction W = value of k8

RETURN	Return from Subroutine
Syntax:	[*label*] RETURN
Operands:	None
Operation:	TOS → PC
Status Affected:	None
Description:	Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two-cycle instruction.

RLF **Rotate Left f through Carry**

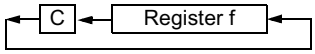
Syntax: [*label*] RLF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: See description below

Status Affected: C

Description: The contents of register 'f' are rotated one bit to the left through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is stored back in register 'f'.



Words: 1

Cycles: 1

Example:

```
RLF     REG1,0
```

Before Instruction

```
REG1    =   1110 0110
C       =   0
```

After Instruction

```
REG1    =   1110 0110
W       =   1100 1100
C       =   1
```

RRF **Rotate Right f through Carry**

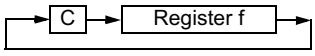
Syntax: [*label*] RRF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: See description below

Status Affected: C

Description: The contents of register 'f' are rotated one bit to the right through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.



SLEEP **Enter Sleep mode**

Syntax: [*label*] SLEEP

Operands: None

Operation: 00h → WDT,
0 → WDT prescaler,
1 → $\overline{TO}$,
0 → $\overline{PD}$

Status Affected: $\overline{TO}$, $\overline{PD}$

Description: The power-down Status bit, $\overline{PD}$ is cleared. Time-out Status bit, $\overline{TO}$ is set. Watchdog Timer and its prescaler are cleared.
The processor is put into Sleep mode with the oscillator stopped.

SUBLW **Subtract W from literal**

Syntax: [*label*] SUBLW k

Operands: $0 \le k \le 255$

Operation: k - (W) → (W)

Status Affected: C, DC, Z

Description: The W register is subtracted (2's complement method) from the eight-bit literal 'k'. The result is placed in the W register.

C = 0	W > k
C = 1	W ≤ k
DC = 0	W<3:0> > k<3:0>
DC = 1	W<3:0> ≤ k<3:0>

SUBWF **Subtract W from f**

Syntax: [*label*] SUBWF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) - (W) → (destination)

Status Affected: C, DC, Z

Description: Subtract (2's complement method) W register from register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f.

C = 0	W > f
C = 1	W ≤ f
DC = 0	W<3:0> > f<3:0>
DC = 1	W<3:0> ≤ f<3:0>

SWAPF **Swap Nibbles in f**

Syntax: [*label*] SWAPF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f<3:0>) → (destination<7:4>),
(f<7:4>) → (destination<3:0>)

Status Affected: None

Description: The upper and lower nibbles of register 'f' are exchanged. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed in register 'f'.

XORLW **Exclusive OR literal with W**

Syntax: [*label*] XORLW k

Operands: $0 \le k \le 255$

Operation: (W) .XOR. k → (W)

Status Affected: Z

Description: The contents of the W register are XOR'ed with the eight-bit literal 'k'. The result is placed in the W register.

XORWF **Exclusive OR W with f**

Syntax: [*label*] XORWF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (W) .XOR. (f) → (destination)

Status Affected: Z

Description: Exclusive OR the contents of the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

NOTES:

18.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers are supported with a full range of hardware and software development tools:

- Integrated Development Environment
  - MPLAB® IDE Software
- Assemblers/Compilers/Linkers
  - MPASM™ Assembler
  - MPLAB C18 and MPLAB C30 C Compilers
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
  - MPLAB ASM30 Assembler/Linker/Library
- Simulators
  - MPLAB SIM Software Simulator
- Emulators
  - MPLAB ICE 2000 In-Circuit Emulator
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debugger
  - MPLAB ICD 2
- Device Programmers
  - PICSTART® Plus Development Programmer
  - MPLAB PM3 Device Programmer
  - PICkit™ 2 Development Programmer
- Low-Cost Demonstration and Development Boards and Evaluation Kits

18.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8/16-bit microcontroller market. The MPLAB IDE is a Windows® operating system-based application that contains:

- A single graphical interface to all debugging tools
  - Simulator
  - Programmer (sold separately)
  - Emulator (sold separately)
  - In-Circuit Debugger (sold separately)
- A full-featured editor with color-coded context
- A multiple project manager
- Customizable data windows with direct edit of contents
- High-level source code debugging
- Visual device initializer for easy register initialization
- Mouse over variable inspection
- Drag and drop variables from source to watch windows
- Extensive on-line help
- Integration of select third party tools, such as HI-TECH Software C Compilers and IAR C Compilers

The MPLAB IDE allows you to:

- Edit your source files (either assembly or C)
- One touch assemble (or compile) and download to PIC MCU emulator and simulator tools (automatically updates all project information)
- Debug using:
  - Source files (assembly or C)
  - Mixed assembly and C
  - Machine code

MPLAB IDE supports multiple debugging tools in a single development paradigm, from the cost-effective simulators, through low-cost in-circuit debuggers, to full-featured emulators. This eliminates the learning curve when upgrading to tools with increased flexibility and power.

18.2 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for all PIC MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multi-purpose source files
- Directives that allow complete control over the assembly process

18.3 MPLAB C18 and MPLAB C30 C Compilers

The MPLAB C18 and MPLAB C30 Code Development Systems are complete ANSI C compilers for Microchip's PIC18 and PIC24 families of microcontrollers and the dsPIC30 and dsPIC33 family of digital signal controllers. These compilers provide powerful integration capabilities, superior code optimization and ease of use not found with other compilers.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

18.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler and the MPLAB C18 C Compiler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

18.5 MPLAB ASM30 Assembler, Linker and Librarian

MPLAB ASM30 Assembler produces relocatable machine code from symbolic assembly language for dsPIC30F devices. MPLAB C30 C Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire dsPIC30F instruction set
- Support for fixed-point and floating-point data
- Command line interface
- Rich directive set
- Flexible macro language
- MPLAB IDE compatibility

18.6 MPLAB SIM Software Simulator

The MPLAB SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC® DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB SIM Software Simulator fully supports symbolic debugging using the MPLAB C18 and MPLAB C30 C Compilers, and the MPASM and MPLAB ASM30 Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

18.7 MPLAB ICE 2000 High-Performance In-Circuit Emulator

The MPLAB ICE 2000 In-Circuit Emulator is intended to provide the product development engineer with a complete microcontroller design tool set for PIC microcontrollers. Software control of the MPLAB ICE 2000 In-Circuit Emulator is advanced by the MPLAB Integrated Development Environment, which allows editing, building, downloading and source debugging from a single environment.

The MPLAB ICE 2000 is a full-featured emulator system with enhanced trace, trigger and data monitoring features. Interchangeable processor modules allow the system to be easily reconfigured for emulation of different processors. The architecture of the MPLAB ICE 2000 In-Circuit Emulator allows expansion to support new PIC microcontrollers.

The MPLAB ICE 2000 In-Circuit Emulator system has been designed as a real-time emulation system with advanced features that are typically found on more expensive development tools. The PC platform and Microsoft® Windows® 32-bit operating system were chosen to best make these features available in a simple, unified application.

18.8 MPLAB REAL ICE In-Circuit Emulator System

MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC® and MCU devices. It debugs and programs PIC® and dsPIC® Flash microcontrollers with the easy-to-use, powerful graphical user interface of the MPLAB Integrated Development Environment (IDE), included with each kit.

The MPLAB REAL ICE probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with the popular MPLAB ICD 2 system (RJ11) or with the new high speed, noise tolerant, low-voltage differential signal (LVDS) interconnection (CAT5).

MPLAB REAL ICE is field upgradeable through future firmware downloads in MPLAB IDE. In upcoming releases of MPLAB IDE, new devices will be supported, and new features will be added, such as software breakpoints and assembly code trace. MPLAB REAL ICE offers significant advantages over competitive emulators including low-cost, full-speed emulation, real-time variable watches, trace analysis, complex breakpoints, a ruggedized probe interface and long (up to three meters) interconnection cables.

18.9 MPLAB ICD 2 In-Circuit Debugger

Microchip's In-Circuit Debugger, MPLAB ICD 2, is a powerful, low-cost, run-time development tool, connecting to the host PC via an RS-232 or high-speed USB interface. This tool is based on the Flash PIC MCUs and can be used to develop for these and other PIC MCUs and dsPIC DSCs. The MPLAB ICD 2 utilizes the in-circuit debugging capability built into the Flash devices. This feature, along with Microchip's In-Circuit Serial Programming™ (ICSP™) protocol, offers cost-effective, in-circuit Flash debugging from the graphical user interface of the MPLAB Integrated Development Environment. This enables a designer to develop and debug source code by setting breakpoints, single stepping and watching variables, and CPU STATUS and peripheral registers. Running at full speed enables testing hardware and applications in real time. MPLAB ICD 2 also serves as a development programmer for selected PIC devices.

18.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at V_{DDMIN} and V_{DDMAX} for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages and a modular, detachable socket assembly to support various package types. The ICSP™ cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices and incorporates an SD/MMC card for file storage and secure data applications.

18.11 PICSTART Plus Development Programmer

The PICSTART Plus Development Programmer is an easy-to-use, low-cost, prototype programmer. It connects to the PC via a COM (RS-232) port. MPLAB Integrated Development Environment software makes using the programmer simple and efficient. The PICSTART Plus Development Programmer supports most PIC devices in DIP packages up to 40 pins. Larger pin count devices, such as the PIC16C92X and PIC17C76X, may be supported with an adapter socket. The PICSTART Plus Development Programmer is CE compliant.

18.12 PICkit 2 Development Programmer

The PICkit™ 2 Development Programmer is a low-cost programmer and selected Flash device debugger with an easy-to-use interface for programming many of Microchip's baseline, mid-range and PIC18F families of Flash memory microcontrollers. The PICkit 2 Starter Kit includes a prototyping development board, twelve sequential lessons, software and HI-TECH's PICC™ Lite C compiler, and is designed to help get up to speed quickly using PIC® microcontrollers. The kit provides everything needed to program, evaluate and develop applications using Microchip's powerful, mid-range Flash memory family of microcontrollers.

18.13 Demonstration, Development and Evaluation Boards

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart® battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Check the Microchip web page (www.microchip.com) and the latest *"Product Selector Guide"* (DS00148) for the complete list of demonstration, development and evaluation kits.

19.0 ELECTRICAL SPECIFICATIONS

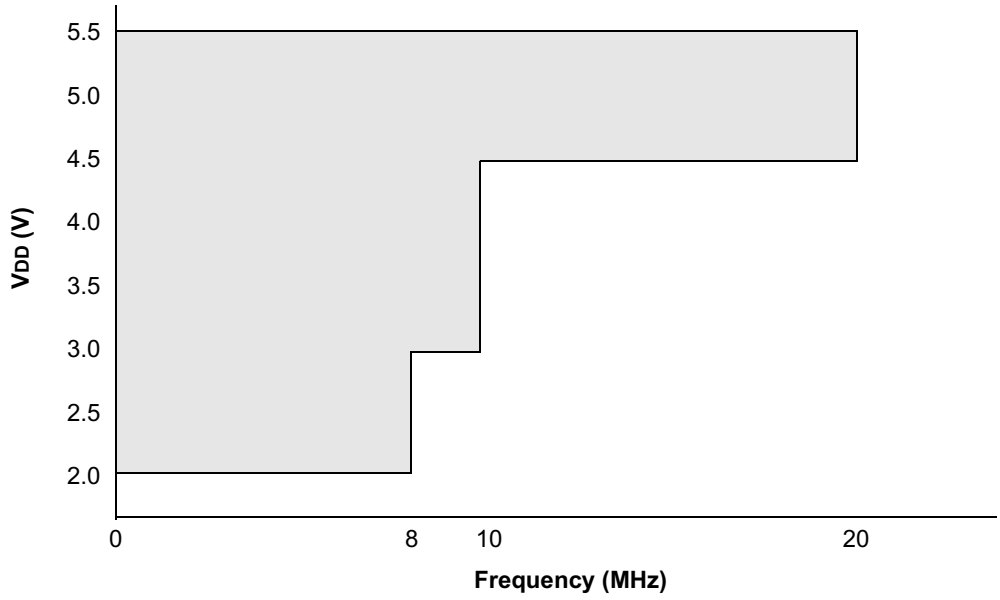
Absolute Maximum Ratings(†)

Ambient temperature under bias ..... -40° to +125°C
Storage temperature ..... -65°C to +150°C
Voltage on V_{DD} with respect to V_{SS} ..... -0.3V to +6.5V
Voltage on $\overline{MCLR}$ with respect to Vss ..... -0.3V to +13.5V
Voltage on all other pins with respect to V_{SS} ..... -0.3V to (V_{DD} + 0.3V)
Total power dissipation(1) ..... 800 mW
Maximum current out of V_{SS} pin ..... 95 mA
Maximum current into V_{DD} pin ..... 95 mA
Input clamp current, I_{IK} (V_I < 0 or V_I > V_{DD}) ..... ±20 mA
Output clamp current, I_{OK} (Vo < 0 or Vo > V_{DD}) ..... ±20 mA
Maximum output current sunk by any I/O pin ..... 25 mA
Maximum output current sourced by any I/O pin ..... 25 mA
Maximum current sourced by all ports (combined) ..... 90 mA
Maximum current sunk by all ports (combined) ..... 90 mA

Note 1: Power dissipation is calculated as follows: $P_{DIS} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum(V_{OL} \times I_{OL})$.
2: PORTD and PORTE are not implemented in PIC16F913/916 devices.

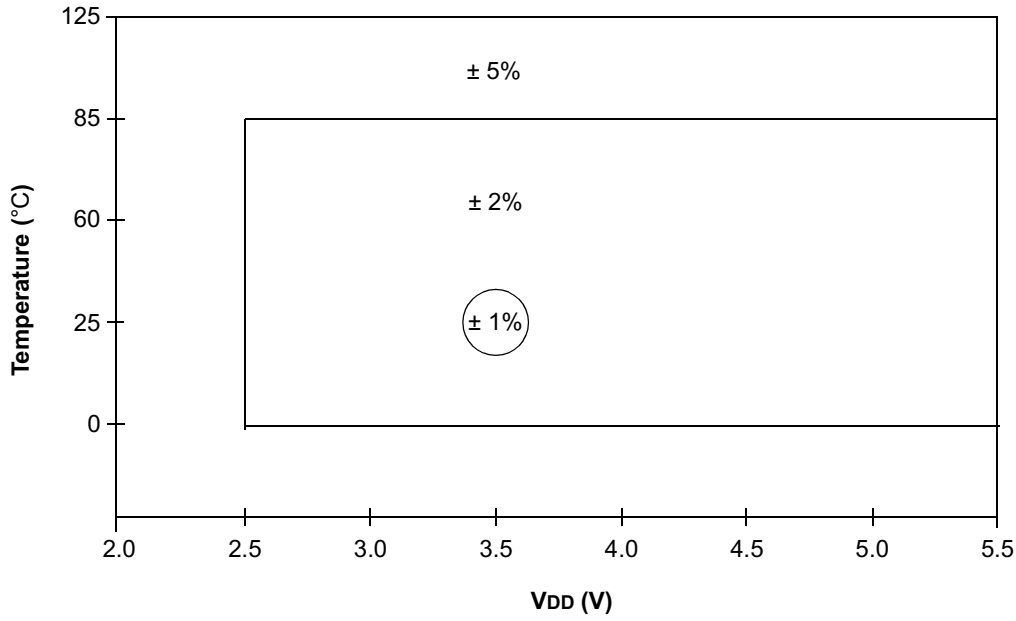
† NOTICE: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

FIGURE 19-1: PIC16F913/914/916/917/946 VOLTAGE-FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C



Note 1: The shaded region indicates the permissible combinations of voltage and frequency.

FIGURE 19-2: HFINTOSC FREQUENCY ACCURACY OVER DEVICE VDD AND TEMPERATURE



19.1 DC Characteristics: PIC16F913/914/916/917/946-I (Industrial) PIC16F913/914/916/917/946-E (Extended)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended				
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
	VDD	**Supply Voltage**	2.0	—	5.5	V	FOSC < = 8 MHz: HFINTOSC, EC
D001			2.0	—	5.5	V	FOSC < = 4 MHz
D001C			3.0	—	5.5	V	FOSC < = 10 MHz
D001D			4.5	—	5.5	V	FOSC < = 20 MHz
D002*	VDR	**RAM Data Retention Voltage(1)**	1.5	—	—	V	Device in Sleep mode
D003	VPOR	**VDD Start Voltage** to ensure internal Power-on Reset signal	—	VSS	—	V	See **Section 16.2.1 "Power-on Reset (POR)"** for details.
D004*	SVDD	**VDD Rise Rate** to ensure internal Power-on Reset signal	0.05	—	—	V/ms	See **Section 16.2.1 "Power-on Reset (POR)"** for details.

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: This is the limit to which VDD can be lowered in Sleep mode without losing RAM data.

19.2 DC Characteristics: PIC16F913/914/916/917/946-I (Industrial) PIC16F913/914/916/917/946-E (Extended)

DC CHARACTERISTICS		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended					
Param No.	Device Characteristics	Min.	Typ†	Max.	Units	Conditions	
						VDD	Note
D010	Supply Current (IDD)(1, 2)	—	13	19	µA	2.0	FOSC = 32 kHz LP Oscillator mode
		—	22	30	µA	3.0	
		—	33	60	µA	5.0	
D011*		—	180	250	µA	2.0	FOSC = 1 MHz XT Oscillator mode
		—	290	400	µA	3.0	
		—	490	650	µA	5.0	
D012		—	280	380	µA	2.0	FOSC = 4 MHz XT Oscillator mode
		—	480	670	µA	3.0	
		—	0.9	1.4	mA	5.0	
D013*		—	170	295	µA	2.0	FOSC = 1 MHz EC Oscillator mode
		—	280	480	µA	3.0	
		—	470	690	µA	5.0	
D014		—	290	450	µA	2.0	FOSC = 4 MHz EC Oscillator mode
		—	490	720	µA	3.0	
		—	0.85	1.3	mA	5.0	
D015		—	8	20	µA	2.0	FOSC = 31 kHz LFINTOSC mode
		—	16	40	µA	3.0	
		—	31	65	µA	5.0	
D016*		—	416	520	µA	2.0	FOSC = 4 MHz HFINTOSC mode
		—	640	840	µA	3.0	
		—	1.13	1.6	mA	5.0	
D017		—	0.65	0.9	mA	2.0	FOSC = 8 MHz HFINTOSC mode
		—	1.01	1.3	mA	3.0	
		—	1.86	2.3	mA	5.0	
D018		—	340	580	µA	2.0	FOSC = 4 MHz EXTRC mode(3)
		—	550	900	µA	3.0	
		—	0.92	1.4	mA	5.0	
D019		—	3.8	4.7	mA	4.5	FOSC = 20 MHz HS Oscillator mode
		—	4.0	4.8	mA	5.0	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: The test conditions for all IDD measurements in active operation mode are: OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD; $\overline{MCLR}$ = VDD; WDT disabled.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature, also have an impact on the current consumption.

3: For RC oscillator configurations, current through REXT is not included. The current through the resistor can be extended by the formula IR = VDD/2REXT (mA) with REXT in kΩ.

19.3 DC Characteristics: PIC16F913/914/916/917/946-I (Industrial)

DC CHARACTERISTICS		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial					
Param No.	Device Characteristics	Min.	Typ†	Max.	Units	Conditions	
						VDD	Note
D020	**Power-down Base Current(IPD)(2)**	—	0.05	1.2	µA	2.0	WDT, BOR, Comparators, VREF and T1OSC disabled
		—	0.15	1.5	µA	3.0	
		—	0.35	1.8	µA	5.0	
		—	150	500	nA	3.0	-40°C ≤ TA ≤ +25°C
D021		—	1.0	2.2	µA	2.0	WDT Current(1)
		—	2.0	4.0	µA	3.0	
		—	3.0	7.0	µA	5.0	
D022A		—	42	60	µA	3.0	BOR Current(1)
		—	85	122	µA	5.0	
D022B		—	22	28	µA	2.0	PLVD Current
		—	25	35	µA	3.0	
		—	33	45	µA	5.0	
D023		—	32	45	µA	2.0	Comparator Current(1), both comparators enabled
		—	60	78	µA	3.0	
		—	120	160	µA	5.0	
D024		—	30	36	µA	2.0	CVREF Current(1) (high range)
		—	45	55	µA	3.0	
		—	75	95	µA	5.0	
D025*		—	39	47	µA	2.0	CVREF Current(1) (low range)
		—	59	72	µA	3.0	
		—	98	124	µA	5.0	
D026		—	2.0	5.0	µA	2.0	T1OSC Current(1), 32.768 kHz
		—	2.5	5.5	µA	3.0	
		—	3.0	7.0	µA	5.0	
D027		—	0.30	1.6	µA	3.0	A/D Current(1), no conversion in progress
		—	0.36	1.9	µA	5.0	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: The peripheral current is the sum of the base IDD or IPD and the additional current consumed when this peripheral is enabled. The peripheral Δ current can be determined by subtracting the base IDD or IPD current from this limit. Max values should be used when calculating total current consumption.

2: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to VDD.

19.4 DC Characteristics: PIC16F913/914/916/917/946-E (Extended)

DC CHARACTERISTICS		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C for extended					
Param No.	Device Characteristics	Min.	Typ†	Max.	Units	Conditions	
						VDD	Note
D020E	**Power-down Base Current (IPD)(2)**	—	0.05	9	µA	2.0	WDT, BOR, Comparators, VREF and T1OSC disabled
		—	0.15	11	µA	3.0	
		—	0.35	15	µA	5.0	
D021E		—	1	28	µA	2.0	WDT Current(1)
		—	2	30	µA	3.0	
		—	3	35	µA	5.0	
D022E		—	42	65	µA	3.0	BOR Current(1)
		—	85	127	µA	5.0	
D022B		—	22	48	µA	2.0	PLVD Current
		—	25	55	µA	3.0	
		—	33	65	µA	5.0	
D023E		—	32	45	µA	2.0	Comparator Current(1), both comparators enabled
		—	60	78	µA	3.0	
		—	120	160	µA	5.0	
D024E		—	30	70	µA	2.0	CVREF Current(1) (high range)
		—	45	90	µA	3.0	
		—	75	120	µA	5.0	
D025E*		—	39	91	µA	2.0	CVREF Current(1) (low range)
		—	59	117	µA	3.0	
		—	98	156	µA	5.0	
D026E		—	3.5	18	µA	2.0	T1OSC Current(1), 32.768 kHz
		—	4	21	µA	3.0	
		—	5	24	µA	5.0	
D027E		—	0.30	12	µA	3.0	A/D Current(1), no conversion in progress
		—	0.36	16	µA	5.0	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: The peripheral current is the sum of the base IDD or IPD and the additional current consumed when this peripheral is enabled. The peripheral ∆ current can be determined by subtracting the base IDD or IPD current from this limit. Max values should be used when calculating total current consumption.

2: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to VDD.

19.5 DC Characteristics: PIC16F913/914/916/917/946-I (Industrial) PIC16F913/914/916/917/946-E (Extended)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended				
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
	VIL	**Input Low Voltage**					
		I/O Port:					
D030		with TTL buffer	Vss	—	0.8	V	4.5V ≤ VDD ≤ 5.5V
D030A			Vss	—	0.15 VDD	V	2.0V ≤ VDD ≤ 4.5V
D031		with Schmitt Trigger buffer	Vss	—	0.2 VDD	V	2.0V ≤ VDD ≤ 5.5V
D032		$\overline{MCLR}$, OSC1 (RC mode)(1)	Vss	—	0.2 VDD	V	
D033		OSC1 (XT mode)	Vss	—	0.3	V	
D033A		OSC1 (HS mode)	Vss	—	0.3 VDD	V	
	VIH	Input High Voltage					
		I/O ports:		—			
D040		with TTL buffer	2.0	—	VDD	V	4.5V ≤ VDD ≤ 5.5V
D040A			0.25 VDD + 0.8	—	VDD	V	2.0V ≤ VDD ≤ 4.5V
D041		with Schmitt Trigger buffer	0.8 VDD	—	VDD	V	2.0V ≤ VDD ≤ 5.5V
D042		$\overline{MCLR}$	0.8 VDD	—	VDD	V	
D043		OSC1 (XT mode)	1.6	—	VDD	V	
D043A		OSC1 (HS mode)	0.7 VDD	—	VDD	V	
D043B		OSC1 (RC mode)	0.9 VDD	—	VDD	V	**(Note 1)**
	IIL	Input Leakage Current(2)					
D060		I/O ports	—	± 0.1	± 1	µA	VSS ≤ VPIN ≤ VDD, Pin at high-impedance
D061		$\overline{MCLR}$(3)	—	± 0.1	± 5	µA	VSS ≤ VPIN ≤ VDD
D063		OSC1	—	± 0.1	± 5	µA	VSS ≤ VPIN ≤ VDD, XT, HS and LP oscillator configuration
D070*	IPUR	PORTB Weak Pull-up Current	50	250	400	µA	VDD = 5.0V, VPIN = VSS
	VOL	Output Low Voltage(5)					
D080		I/O ports	—	—	0.6	V	IOL = 8.5 mA, VDD = 4.5V (Ind.)
	VOH	Output High Voltage(5)					
D090		I/O ports	VDD – 0.7	—	—	V	IOH = -3.0 mA, VDD = 4.5V (Ind.)

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended to use an external clock in RC mode.

2: Negative current is defined as current sourced by the pin.

3: The leakage current on the $\overline{MCLR}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

4: See **Section 13.0 "Data EEPROM and Flash Program Memory Control"** for additional information.

5: Including OSC2 in CLKOUT mode.

19.5 DC Characteristics: PIC16F913/914/916/917/946-I (Industrial) PIC16F913/914/916/917/946-E (Extended) (Continued)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended				
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
		Capacitive Loading Specs on Output Pins					
D101*	COSC2	OSC2 pin	—	—	15	pF	In XT, HS and LP modes when external clock is used to drive OSC1
D101A*	CIO	All I/O pins	—	—	50	pF	
		Data EEPROM Memory					
D120	ED	Byte Endurance	100K	1M	—	E/W	-40°C ≤ TA ≤ +85°C
D120A	ED	Byte Endurance	10K	100K	—	E/W	+85°C ≤ TA ≤ +125°C
D121	VDRW	VDD for Read/Write	VMIN	—	5.5	V	Using EECON1 to read/write VMIN = Minimum operating voltage
D122	TDEW	Erase/Write Cycle Time	—	5	6	ms	
D123	TRETD	Characteristic Retention	40	—	—	Year	Provided no other specifications are violated
D124	TREF	Number of Total Erase/Write Cycles before Refresh(4)	1M	10M	—	E/W	-40°C ≤ TA ≤ +85°C
		Program Flash Memory					
D130	EP	Cell Endurance	10K	100K	—	E/W	-40°C ≤ TA ≤ +85°C
D130A	ED	Cell Endurance	1K	10K	—	E/W	+85°C ≤ TA ≤ +125°C
D131	VPR	VDD for Read	VMIN	—	5.5	V	VMIN = Minimum operating voltage
D132	VPEW	VDD for Erase/Write	4.5	—	5.5	V	
D133	TPEW	Erase/Write cycle time	—	—	3	ms	
D134	TRETD	Characteristic Retention	40	—	—	Year	Provided no other specifications are violated

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended to use an external clock in RC mode.

2: Negative current is defined as current sourced by the pin.

3: The leakage current on the $\overline{MCLR}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

4: See **Section 13.0 "Data EEPROM and Flash Program Memory Control"** for additional information.

5: Including OSC2 in CLKOUT mode.

19.6 Thermal Considerations

Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C					
Param No.	**Symbol**	**Characteristic**	**Typ.**	**Units**	**Conditions**
TH01	θJA	Thermal Resistance Junction to Ambient	60.0	°C/W	28-pin PDIP package
			80.0	°C/W	28-pin SOIC package
			90.0	°C/W	28-pin SSOP package
			27.5	°C/W	28-pin QFN 6x6 mm package
			47.2	°C/W	40-pin PDIP package
			46.0	°C/W	44-pin TQFP package
			24.4	°C/W	44-pin QFN 8x8 mm package
			77.0	°C/W	64-pin TQFP package
TH02	θJC	Thermal Resistance Junction to Case	31.4	°C/W	28-pin PDIP package
			24.0	°C/W	28-pin SOIC package
			24.0	°C/W	28-pin SSOP package
			20.0	°C/W	28-pin QFN 6x6 mm package
			24.7	°C/W	40-pin PDIP package
			14.5	°C/W	44-pin TQFP package
			20.0	°C/W	44-pin QFN 8x8 mm package
			24.4	°C/W	64-pin TQFP package
TH03	TJ	Junction Temperature	150	°C	For derated power calculations
TH04	PD	Power Dissipation	—	W	PD = PINTERNAL + PI/O
TH05	PINTERNAL	Internal Power Dissipation	—	W	PINTERNAL = IDD x VDD **(NOTE 1)**
TH06	PI/O	I/O Power Dissipation	—	W	PI/O = Σ (IOL * VOL) + Σ (IOH * (VDD - VOH))
TH07	PDER	Derated Power	—	W	PDER = (TJ - TA)/θJA **(NOTE 2, 3)**

Note 1: IDD is current to run the chip alone without driving any load on the output pins.

2: TA = Ambient Temperature.

3: Maximum allowable power dissipation is the lower value of either the absolute maximum total power dissipation or derated power (PDER).

19.7 Timing Parameter Symbology

The timing parameter symbols have been created with one of the following formats:

1. TppS2ppS
2. TppS

T			
F	Frequency	T	Time

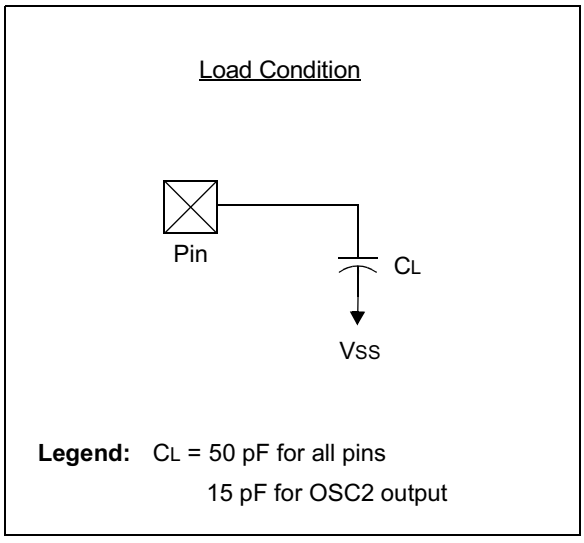
Lowercase letters (pp) and their meanings:

pp			
cc	CCP1	osc	OSC1
ck	CLKOUT	rd	$\overline{RD}$
cs	$\overline{CS}$	rw	$\overline{RD}$ or $\overline{WR}$
di	SDI	sc	SCK
do	SDO	ss	$\overline{SS}$
dt	Data in	t0	T0CKI
io	I/O port	t1	T1CKI
mc	$\overline{MCLR}$	wr	$\overline{WR}$

Uppercase letters and their meanings:

S			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (High-impedance)	V	Valid
L	Low	Z	High-impedance

FIGURE 19-3: LOAD CONDITIONS



Legend: CL = 50 pF for all pins
15 pF for OSC2 output

19.8 AC Characteristics: PIC16F913/914/916/917/946 (Industrial, Extended)

FIGURE 19-4: CLOCK TIMING

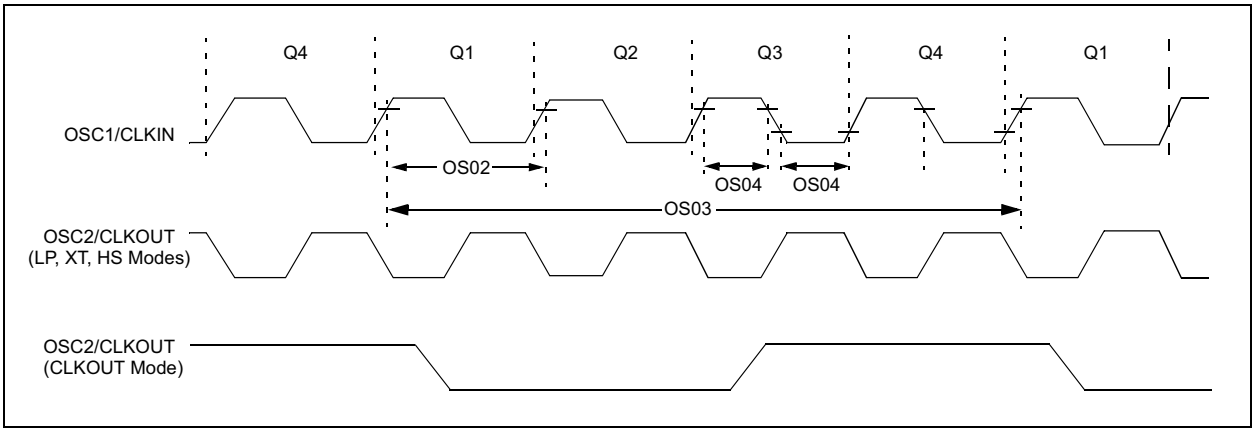


TABLE 19-1: CLOCK OSCILLATOR TIMING REQUIREMENTS

Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C							
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
OS01	FOSC	External CLKIN Frequency(1)	DC	—	37	kHz	LP Oscillator mode
			DC	—	4	MHz	XT Oscillator mode
			DC	—	20	MHz	HS Oscillator mode
			DC	—	20	MHz	EC Oscillator mode
		Oscillator Frequency(1)	—	32.768	—	kHz	LP Oscillator mode
			0.1	—	4	MHz	XT Oscillator mode
			1	—	20	MHz	HS Oscillator mode
			DC	—	4	MHz	RC Oscillator mode
OS02	TOSC	External CLKIN Period(1)	27	—	∞	µs	LP Oscillator mode
			250	—	∞	ns	XT Oscillator mode
			50	—	∞	ns	HS Oscillator mode
			50	—	∞	ns	EC Oscillator mode
		Oscillator Period(1)	—	30.5	—	µs	LP Oscillator mode
			250	—	10,000	ns	XT Oscillator mode
			50	—	1,000	ns	HS Oscillator mode
			250	—	—	ns	RC Oscillator mode
OS03	TCY	Instruction Cycle Time(1)	200	TCY	DC	ns	TCY = 4/FOSC
OS04*	TosH, TosL	External CLKIN High, External CLKIN Low	2	—	—	µs	LP oscillator
			100	—	—	ns	XT oscillator
			20	—	—	ns	HS oscillator
OS05*	TosR, TosF	External CLKIN Rise, External CLKIN Fall	0	—	∞	ns	LP oscillator
			0	—	∞	ns	XT oscillator
			0	—	∞	ns	HS oscillator

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Instruction cycle period (TCY) equals four times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min" values with an external clock applied to OSC1 pin. When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

TABLE 19-2: OSCILLATOR PARAMETERS

Standard Operating Conditions (unless otherwise stated) Operating Temperature -40°C ≤ TA ≤ +125°C								
Param No.	**Sym.**	**Characteristic**	**Freq. Tolerance**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
OS06	TWARM	Internal Oscillator Switch when running(3)	—	—	—	2	TOSC	Slowest clock
OS07	TSC	Fail-Safe Sample Clock Period(1)	—	—	21	—	ms	LFINTOSC/64
OS08	HFOSC	Internal Calibrated HFINTOSC Frequency(2)	±1%	7.92	8.0	8.08	MHz	VDD = 3.5V, 25°C
			±2%	7.84	8.0	8.16	MHz	2.5V ≤ VDD ≤ 5.5V, 0°C ≤ TA ≤ +85°C
			±5%	7.60	8.0	8.40	MHz	2.0V ≤ VDD ≤ 5.5V, -40°C ≤ TA ≤ +85°C (Ind.), -40°C ≤ TA ≤ +125°C (Ext.)
OS09*	LFOSC	Internal Uncalibrated LFINTOSC Frequency	—	15	31	45	kHz	
OS10*	TIOSC ST	HFINTOSC Oscillator Wake-up from Sleep Start-up Time	—	5.5	12	24	µs	VDD = 2.0V, -40°C to +85°C
			—	3.5	7	14	µs	VDD = 3.0V, -40°C to +85°C
			—	3	6	11	µs	VDD = 5.0V, -40°C to +85°C

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Instruction cycle period (TCY) equals four times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min" values with an external clock applied to the OSC1 pin. When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

2: To ensure these oscillator frequency tolerances, VDD and VSS must be capacitively decoupled as close to the device as possible. 0.1 µF and 0.01 µF values in parallel are recommended.

3: By design.

FIGURE 19-5: CLKOUT AND I/O TIMING

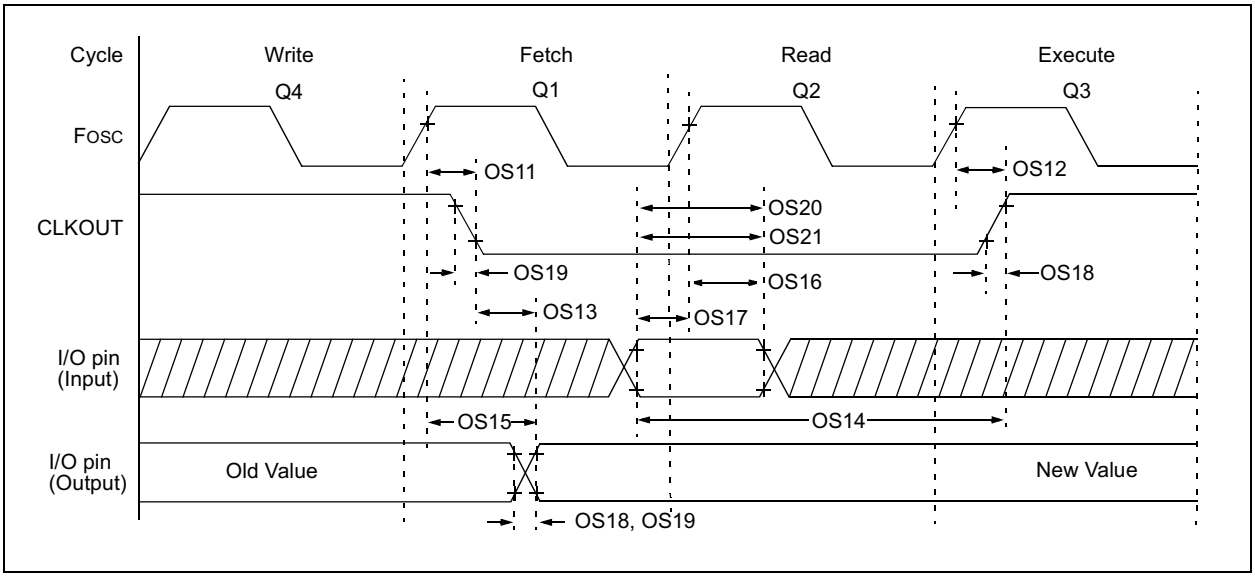


TABLE 19-3: CLKOUT AND I/O TIMING PARAMETERS

Standard Operating Conditions (unless otherwise stated) Operating Temperature -40°C ≤ TA ≤ +125°C							
Param No.	**Symbol**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
OS11	TOSH2CKL	FOSC↑ to CLKOUT↓ (1)	—	—	70	ns	VDD = 5.0V
OS12	TOSH2CKH	FOSC↑ to CLKOUT↑ (1)	—	—	72	ns	VDD = 5.0V
OS13	TCKL2IOV	CLKOUT↓ to Port out valid(1)	—	—	20	ns	
OS14	TIOV2CKH	Port input valid before CLKOUT↑(1)	TOSC + 200 ns	—	—	ns	
OS15*	TOSH2IOV	FOSC↑ (Q1 cycle) to Port out valid	—	50	70	ns	VDD = 5.0V
OS16	TOSH2IOI	FOSC↑ (Q2 cycle) to Port input invalid (I/O in hold time)	50	—	—	ns	VDD = 5.0V
OS17	TIOV2OSH	Port input valid to FOSC↑ (Q2 cycle) (I/O in setup time)	20	—	—	ns	
OS18	TIOR	Port output rise time(2)	— —	15 40	72 32	ns	VDD = 2.0V VDD = 5.0V
OS19	TIOF	Port output fall time(2)	— —	28 15	55 30	ns	VDD = 2.0V VDD = 5.0V
OS20*	TINP	INT pin input high or low time	25	—	—	ns	
OS21*	TRAP	PORTA interrupt-on-change new input level time	TCY	—	—	ns	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated.

Note 1: Measurements are taken in RC mode where CLKOUT output is 4 x TOSC.

2: Includes OSC2 in CLKOUT mode.

FIGURE 19-6: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING

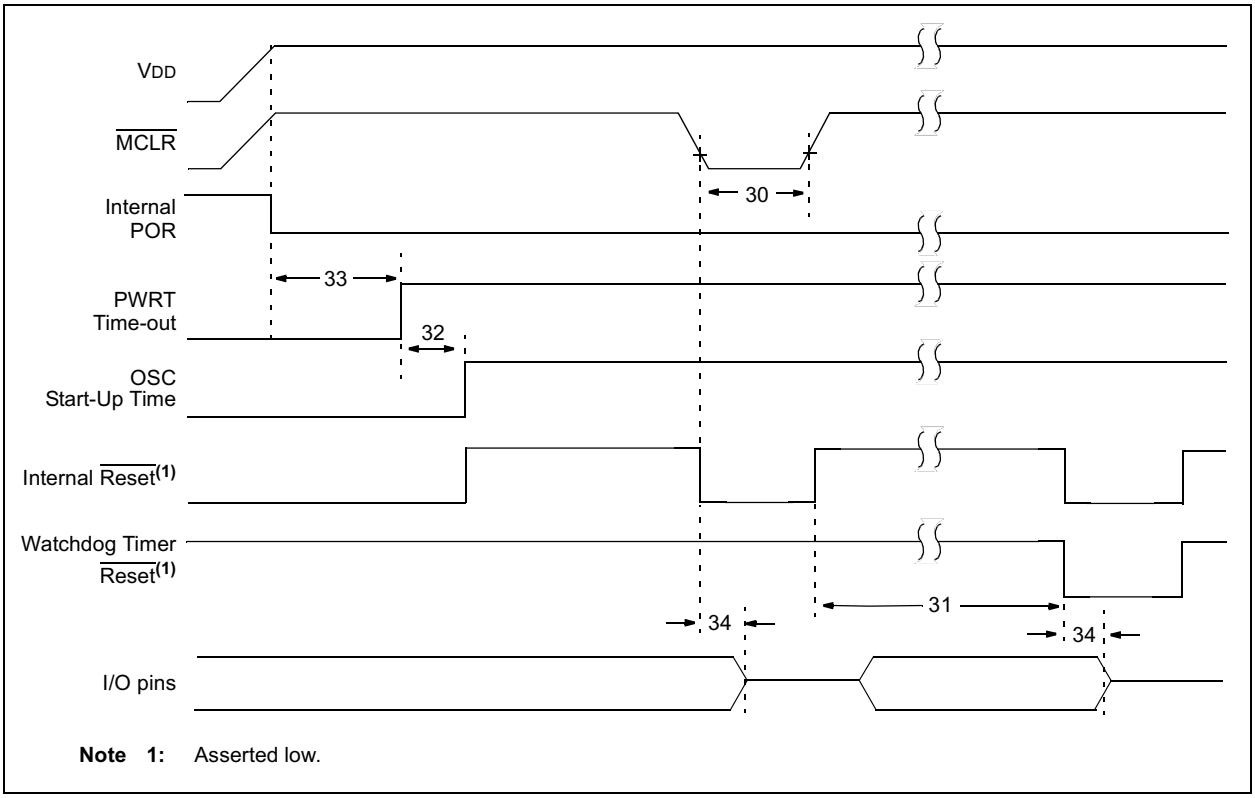


FIGURE 19-7: BROWN-OUT RESET TIMING AND CHARACTERISTICS

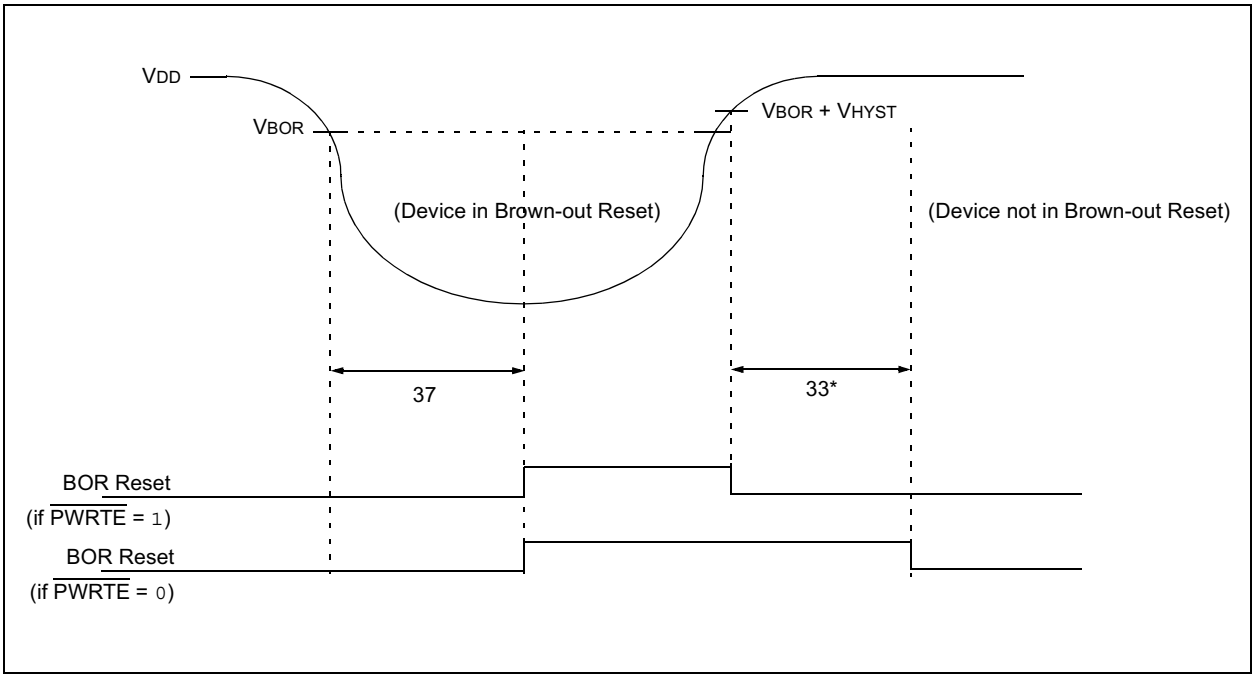


TABLE 19-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER AND BROWN-OUT RESET PARAMETERS

Standard Operating Conditions (unless otherwise stated) Operating Temperature -40°C ≤ TA ≤ +125°C							
Param No.	**Symbol**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
30	TMCL	$\overline{\text{MCLR}}$ Pulse Width (low)	2 5	— —	— —	µs µs	VDD = 5V, -40°C to +85°C VDD = 5V
31	TWDT	Watchdog Timer Time-out Period (No Prescaler)	10 10	16 16	29 31	ms ms	VDD = 5V, -40°C to +85°C VDD = 5V
32	TOST	Oscillation Start-up Timer Period**(1, 2)**	—	1024	—	TOSC	**(NOTE 3)**
33*	TPWRT	Power-up Timer Period	40	65	140	ms	
34*	TIOZ	I/O High-impedance from $\overline{\text{MCLR}}$ Low or Watchdog Timer Reset	—	—	2.0	µs	
35	VBOR	Brown-out Reset Voltage	2.0 2.0	— —	2.2 2.25	V V	-40°C to +85°C, **(NOTE 4)** -40°C to +125°C, **(NOTE 4)**
36*	VHYST	Brown-out Reset Hysteresis	—	50	—	mV	
37*	TBOR	Brown-out Reset Minimum Detection Period	100	—	—	µs	VDD ≤ VBOR

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Instruction cycle period (TCY) equals four times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min" values with an external clock applied to the OSC1 pin. When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

2: By design.

3: Period of the slower clock.

4: To ensure these voltage tolerances, VDD and VSS must be capacitively decoupled as close to the device as possible. 0.1 µF and 0.01 µF values in parallel are recommended.

FIGURE 19-8: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS

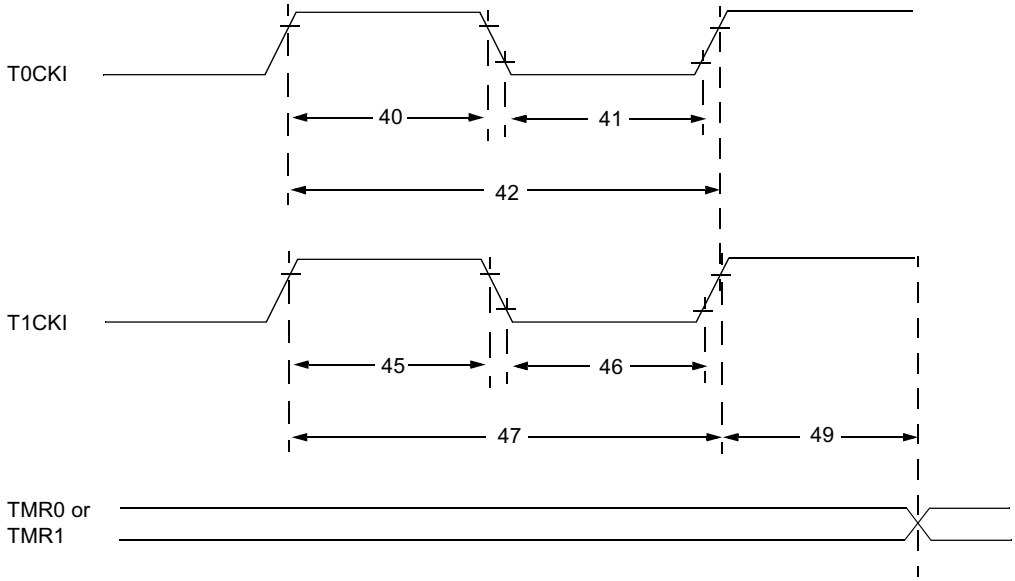


TABLE 19-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

Standard Operating Conditions (unless otherwise stated) Operating Temperature -40°C ≤ TA ≤ +125°C								
Param No.	**Symbol**	**Characteristic**		**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
40*	TT0H	T0CKI High Pulse Width	No Prescaler	0.5 TCY + 20	—	—	ns	
			With Prescaler	10	—	—	ns	
41*	TT0L	T0CKI Low Pulse Width	No Prescaler	0.5 TCY + 20	—	—	ns	
			With Prescaler	10	—	—	ns	
42*	TT0P	T0CKI Period		Greater of: 20 or TCY + 40 N	—	—	ns	N = prescale value (2, 4, ..., 256)
45*	TT1H	T1CKI High Time	Synchronous, No Prescaler	0.5 TCY + 20	—	—	ns	
			Synchronous, with Prescaler	15	—	—	ns	
			Asynchronous	30	—	—	ns	
46*	TT1L	T1CKI Low Time	Synchronous, No Prescaler	0.5 TCY + 20	—	—	ns	
			Synchronous, with Prescaler	15	—	—	ns	
			Asynchronous	30	—	—	ns	
47*	TT1P	T1CKI Input Period	Synchronous	Greater of: 30 or TCY + 40 N	—	—	ns	N = prescale value (1, 2, 4, 8)
			Asynchronous	60	—	—	ns	
48	FT1	Timer1 Oscillator Input Frequency Range (oscillator enabled by setting bit T1OSCEN)		—	32.768	—	kHz	
49*	TCKEZTMR1	Delay from External Clock Edge to Timer Increment		2 TOSC	—	7 TOSC	—	Timers in Sync mode

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

TABLE 19-6: COMPARATOR SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated) Operating Temperature -40°C ≤ TA ≤ +125°C								
Param No.	**Symbol**	**Characteristics**		**Min.**	**Typ†**	**Max.**	**Units**	**Comments**
CM01	VOS	Input Offset Voltage		—	± 5.0	± 10	mV	(VDD - 1.5)/2
CM02	VCM	Input Common Mode Voltage		0	—	VDD – 1.5	V	
CM03*	CMRR	Common Mode Rejection Ratio		+55	—	—	dB	
CM04*	TRT	Response Time	Falling	—	150	600	ns	**(NOTE 1)**
			Rising	—	200	1000	ns	
CM05*	TMC2COV	Comparator Mode Change to Output Valid		—	—	10	µs	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Response time is measured with one comparator input at (VDD - 1.5)/2 - 100 mV to (VDD - 1.5)/2 + 20 mV.

TABLE 19-7: COMPARATOR VOLTAGE REFERENCE (CVREF) SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C							
Param No.	**Symbol**	**Characteristics**	**Min.**	**Typ†**	**Max.**	**Units**	**Comments**
CV01*	CLSB	Step Size(2)	— —	VDD/24 VDD/32	— —	V V	Low Range (VRR = 1) High Range (VRR = 0)
CV02*	CACC	Absolute Accuracy	— —	— —	± 1/2 ± 1/2	LSb LSb	Low Range (VRR = 1) High Range (VRR = 0)
CV03*	CR	Unit Resistor Value (R)	—	2k	—	Ω	
CV04*	CST	Settling Time(1)	—	—	10	µs	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Settling time measured while VRR = 1 and VR<3:0> transitions from '0000' to '1111'.

2: See **Section 8.10 "Comparator Voltage Reference"** for more information.

TABLE 19-8: PIC16F913/914/916/917/946 A/D CONVERTER (ADC) CHARACTERISTICS

Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C							
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
AD01	NR	Resolution	—	—	10 bits	bit	
AD02	EIL	Integral Error	—	—	±1	LSb	VREF = 5.12V
AD03	EDL	Differential Error	—	—	±1	LSb	No missing codes to 10 bits VREF = 5.12V
AD04	EOFF	Offset Error	—	—	±1	LSb	VREF = 5.12V
AD07	EGN	Gain Error	—	—	±1	LSb	VREF = 5.12V
AD06 AD06A	VREF	Reference Voltage(1)	2.2 2.7	—	VDD VDD	V	 Absolute minimum to ensure 1 LSb accuracy
AD07	VAIN	Full-Scale Range	VSS	—	VREF	V	
AD08	ZAIN	Recommended Impedance of Analog Voltage Source	—	—	10	kΩ	
AD09*	IREF	VREF Input Current(1)	10	—	1000	µA	During VAIN acquisition. Based on differential of VHOLD to VAIN.
			—	—	50	µA	During A/D conversion cycle.

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: ADC VREF is from external VREF or VDD pin, whichever is selected as reference input.

TABLE 19-9: PIC16F913/914/916/917/946 A/D CONVERSION REQUIREMENTS

Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C							
Param No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
AD130*	TAD	A/D Clock Period	1.6	—	9.0	µs	TOSC-based, VREF ≥ 3.0V
			3.0	—	9.0	µs	TOSC-based, VREF full range
		A/D Internal RC Oscillator Period					ADCS<1:0> = 11 (ADRC mode)
			3.0	6.0	9.0	µs	At VDD = 2.5V
			1.6	4.0	6.0	µs	At VDD = 5.0V
AD131	TCNV	Conversion Time (not including Acquisition Time)(1)	—	11	—	TAD	Set GO/$\overline{\text{DONE}}$ bit to new data in A/D Result register
AD132*	TACQ	Acquisition Time		11.5	—	µs	
AD133*	TAMP	Amplifier Settling Time	—	—	5	µs	
AD134	TGO	Q4 to A/D Clock Start	—	TOSC/2	—	—	
			—	TOSC/2 + TCY	—	—	If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

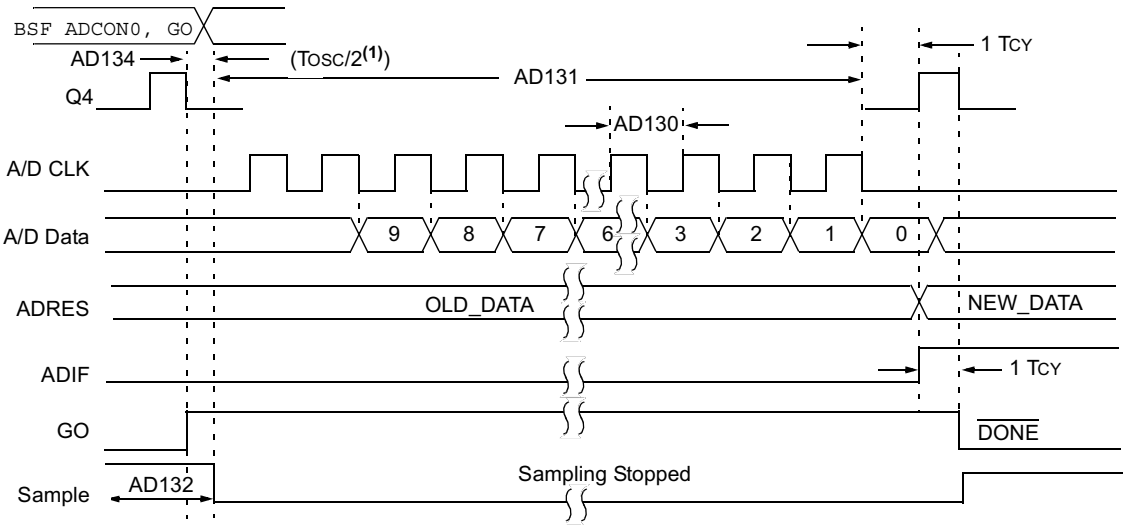
* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: ADRESH and ADRESL registers may be read on the following TCY cycle.

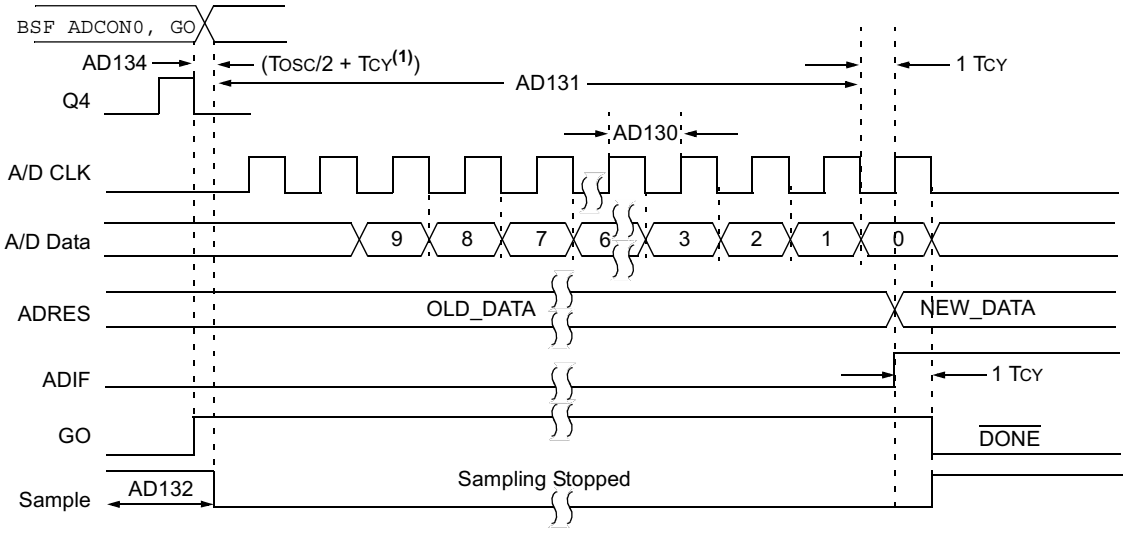
2: See **Section 12.3 "A/D Acquisition Requirements"** for minimum conditions.

FIGURE 19-9: PIC16F913/914/916/917/946 A/D CONVERSION TIMING (NORMAL MODE)



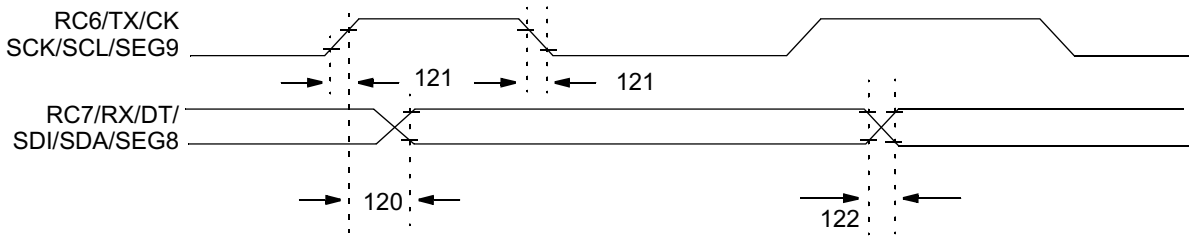
Note 1: If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

FIGURE 19-10: PIC16F913/914/916/917/946 A/D CONVERSION TIMING (SLEEP MODE)



Note 1: If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

FIGURE 19-11: USART SYNCHRONOUS TRANSMISSION (MASTER/SLAVE) TIMING

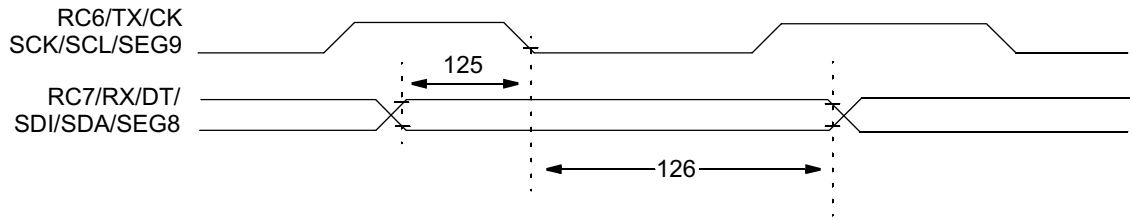


Note: Refer to Figure 19-3 for load conditions.

TABLE 19-10: USART SYNCHRONOUS TRANSMISSION REQUIREMENTS

Standard Operating Conditions (unless otherwise stated) Operating Temperature -40°C ≤ TA ≤ +125°C							
Param. No.	**Symbol**	**Characteristic**		**Min.**	**Max.**	**Units**	**Conditions**
120	TCKH2DTV	SYNC XMIT (Master and Slave) Clock high to data-out valid	3.0-5.5V	—	80	ns	
			2.0-5.5V	—	100	ns	
121	TCKRF	Clock out rise time and fall time (Master mode)	3.0-5.5V	—	45	ns	
			2.0-5.5V	—	50	ns	
122	TDTRF	Data-out rise time and fall time	3.0-5.5V	—	45	ns	
			2.0-5.5V	—	50	ns	

FIGURE 19-12: USART SYNCHRONOUS RECEIVE (MASTER/SLAVE) TIMING

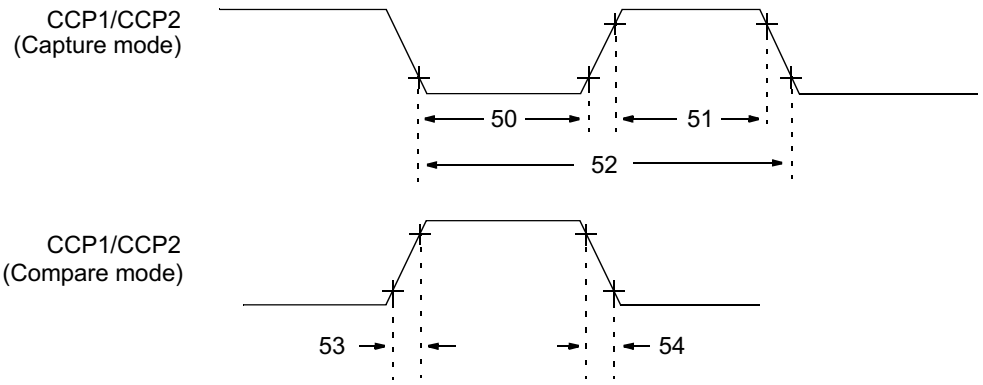


Note: Refer to Figure 19-3 for load conditions.

TABLE 19-11: USART SYNCHRONOUS RECEIVE REQUIREMENTS

Standard Operating Conditions (unless otherwise stated) Operating Temperature -40°C ≤ TA ≤ +125°C						
Param. No.	**Symbol**	**Characteristic**	**Min.**	**Max.**	**Units**	**Conditions**
125	TDTV2CKL	SYNC RCV (Master and Slave) Data-hold before CK ↓ (DT hold time)	10	—	ns	
126	TCKL2DTL	Data-hold after CK ↓ (DT hold time)	15	—	ns	

FIGURE 19-13: CAPTURE/COMPARE/PWM TIMINGS



Note: Refer to Figure 19-3 for load conditions.

TABLE 19-12: CAPTURE/COMPARE/PWM (CCP) REQUIREMENTS

Param. No.	Sym.	Characteristic			Min.	Typ†	Max.	Units	Conditions
50*	TccL	CCPx input low time	No Prescaler		0.5TCY + 5	—	—	ns	
			With Prescaler	3.0-5.5V	10	—	—	ns	
				2.0-5.5V	20	—	—	ns	
51*	TccH	CCPx input high time	No Prescaler		0.5TCY + 5	—	—	ns	
			With Prescaler	3.0-5.5V	10	—	—	ns	
				2.0-5.5V	20	—	—	ns	
52*	TccP	CCPx input period			$\frac{3T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1,4 or 16)
53*	TccR	CCPx output fall time		3.0-5.5V	—	10	25	ns	
				2.0-5.5V	—	25	50	ns	
54*	TccF	CCPx output fall time		3.0-5.5V	—	10	25	ns	
				2.0-5.5V	—	25	45	ns	

* These parameters are characterized but not tested.

† Data in “Typ” column is at 5.0V, 25°C unless otherwise stated. Parameters are for design guidance only and are not tested.

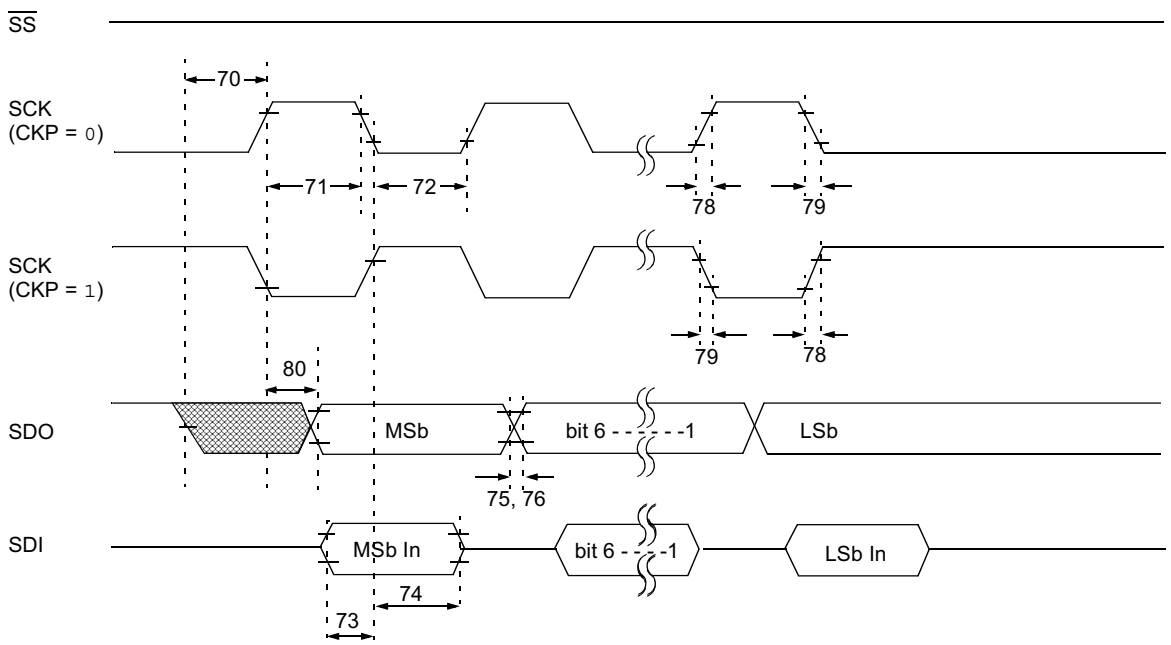
TABLE 19-13: PIC16F913/914/916/917/946 PLVD CHARACTERISTICS:

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating Temperature -40°C ≤ TA ≤ +125°C Operating Voltage VDD Range 2.0V-5.5V					
Sym.	**Characteristic**		**Min.**	**Typ†**	**Max. (85°C)**	**Max. (125°C)**	**Units**	**Conditions**
VPLVD	PLVD Voltage	LVDL<2:0> = 001	1.900	2.0	2.100	2.125	V	
		LVDL<2:0> = 010	2.000	2.1	2.200	2.225	V	
		LVDL<2:0> = 011	2.100	2.2	2.300	2.325	V	
		LVDL<2:0> = 100	2.200	2.3	2.400	2.425	V	
		LVDL<2:0> = 101	3.825	4.0	4.175	4.200	V	
		LVDL<2:0> = 110	4.025	4.2	4.375	4.400	V	
		LVDL<2:0> = 111	4.425	4.5	4.675	4.700	V	
*TPLVDS	PLVD Settling time		—	50 25	—	—	µs	VDD = 5.0V VDD = 3.0V

* These parameters are characterized but not tested

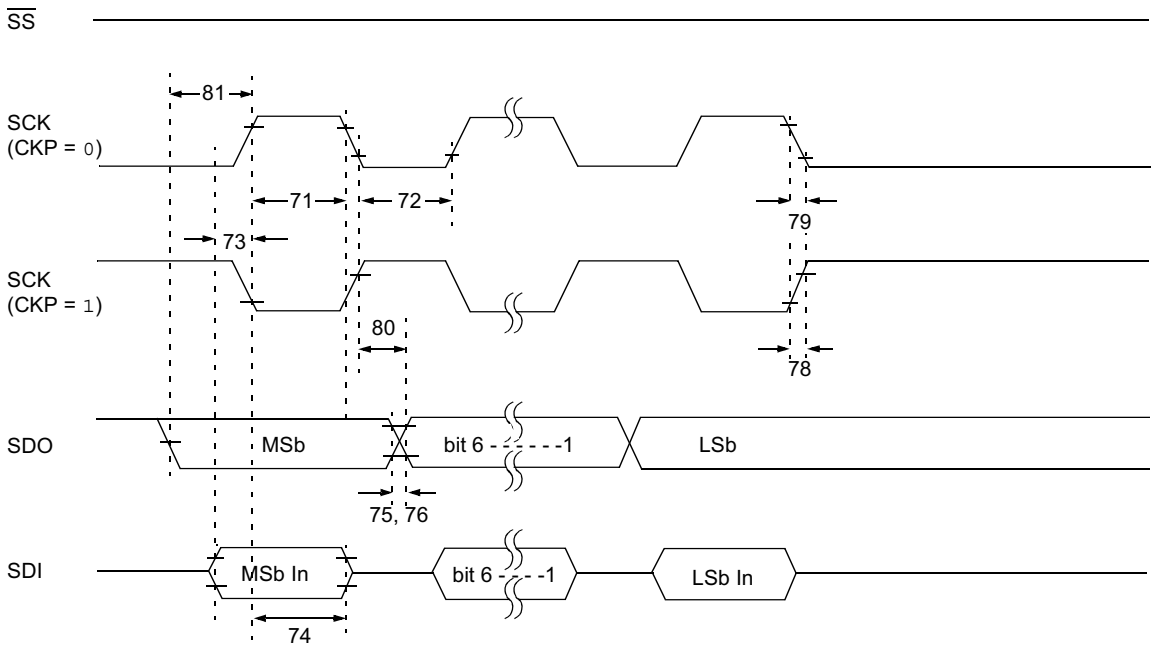
† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 19-14: SPI MASTER MODE TIMING (CKE = 0, SMP = 0)



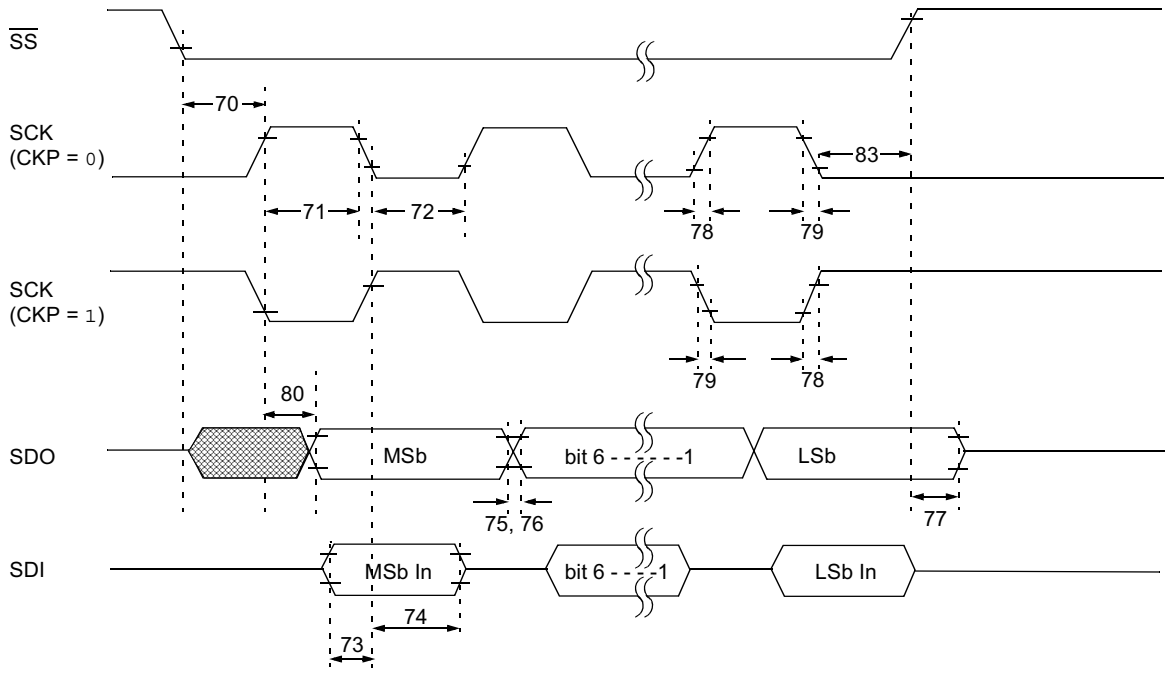
Note: Refer to Figure 19-3 for load conditions.

FIGURE 19-15: SPI MASTER MODE TIMING (CKE = 1, SMP = 1)



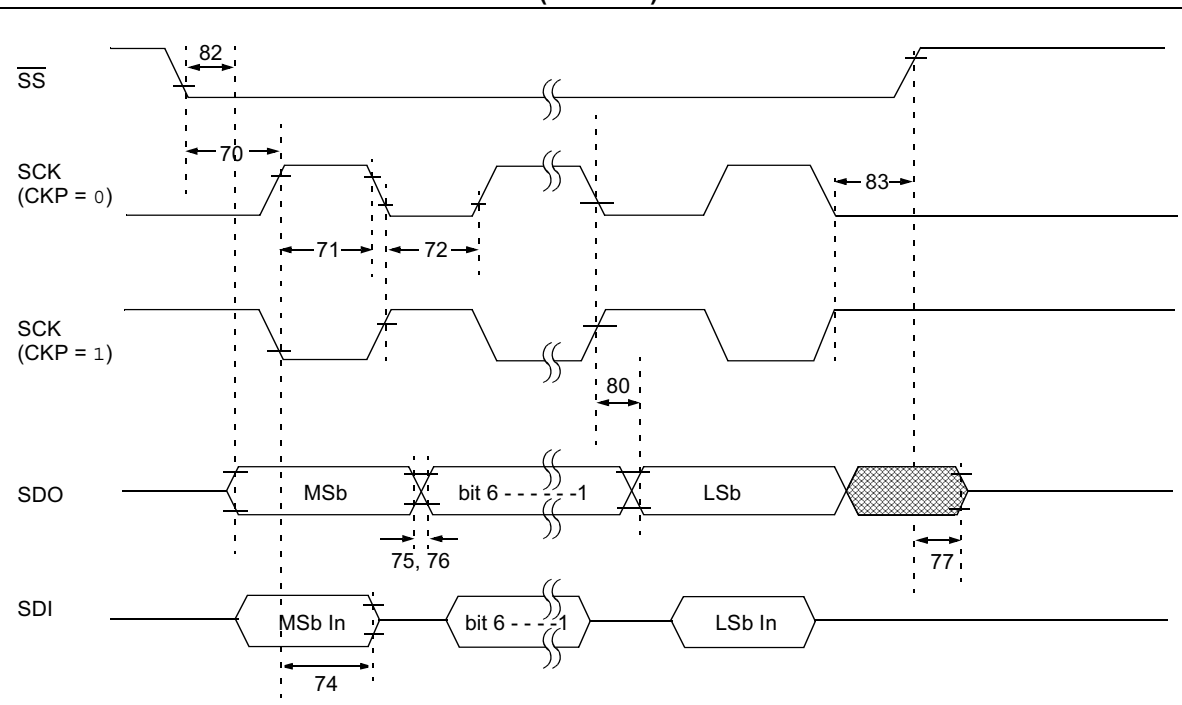
Note: Refer to Figure 19-3 for load conditions.

FIGURE 19-16: SPI SLAVE MODE TIMING (CKE = 0)



Note: Refer to Figure 19-3 for load conditions.

FIGURE 19-17: SPI SLAVE MODE TIMING (CKE = 1)



Note: Refer to Figure 19-3 for load conditions.

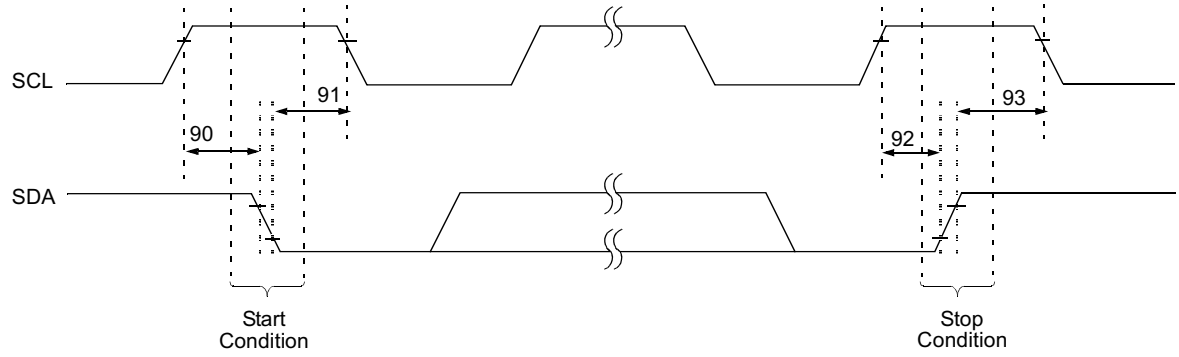
TABLE 19-14: SPI MODE REQUIREMENTS

Param No.	Symbol	Characteristic		Min.	Typ†	Max.	Units	Conditions
70*	TssL2scH, TssL2scL	$\overline{SS}\downarrow$ to SCK$\downarrow$ or SCK$\uparrow$ input		Tcy	—	—	ns	
71*	TscH	SCK input high time (Slave mode)		Tcy + 20	—	—	ns	
72*	TscL	SCK input low time (Slave mode)		Tcy + 20	—	—	ns	
73*	TdiV2scH, TdiV2scL	Setup time of SDI data input to SCK edge		100	—	—	ns	
74*	TscH2diL, TscL2diL	Hold time of SDI data input to SCK edge		100	—	—	ns	
75*	TdoR	SDO data output rise time	3.0-5.5V	—	10	25	ns	
			2.0-5.5V	—	25	50	ns	
76*	TdoF	SDO data output fall time		—	10	25	ns	
77*	TssH2doZ	$\overline{SS}\uparrow$ to SDO output high-impedance		10	—	50	ns	
78*	TscR	SCK output rise time (Master mode)	3.0-5.5V	—	10	25	ns	
			2.0-5.5V	—	25	50	ns	
79*	TscF	SCK output fall time (Master mode)		—	10	25	ns	
80*	TscH2doV, TscL2doV	SDO data output valid after SCK edge	3.0-5.5V	—	—	50	ns	
			2.0-5.5V	—	—	145	ns	
81*	TdoV2scH, TdoV2scL	SDO data output setup to SCK edge		Tcy	—	—	ns	
82*	TssL2doV	SDO data output valid after $\overline{SS}\downarrow$ edge		—	—	50	ns	
83*	TscH2ssH, TscL2ssH	$\overline{SS}\uparrow$ after SCK edge		1.5Tcy + 40	—	—	ns	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 19-18: I^2C™ BUS START/STOP BITS TIMING



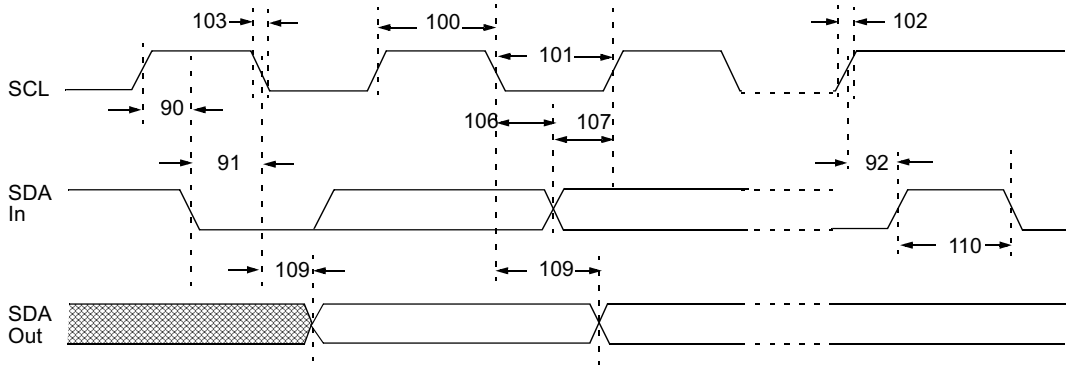
Note: Refer to Figure 19-3 for load conditions.

TABLE 19-15: I²C™ BUS START/STOP BITS REQUIREMENTS

Param No.	Symbol	Characteristic		Min.	Typ.	Max.	Units	Conditions
90*	TSU:STA	Start condition Setup time	400 kHz mode	600	—	—	ns	Only relevant for Repeated Start condition
91*	THD:STA	Start condition Hold time	400 kHz mode	600	—	—	ns	After this period, the first clock pulse is generated
92*	TSU:STO	Stop condition Setup time	400 kHz mode	600	—	—	ns	
93	THD:STO	Stop condition Hold time	400 kHz mode	600	—	—	ns	

* These parameters are characterized but not tested.

FIGURE 19-19: I²C™ BUS DATA TIMING



Note: Refer to Figure 19-3 for load conditions.

TABLE 19-16: I²C™ BUS DATA REQUIREMENTS

Param. No.	Symbol	Characteristic		Min.	Max.	Units	Conditions
100*	THIGH	Clock high time	400 kHz mode	0.6	—	µs	Device must operate at a minimum of 10 MHz
			SSP Module	1.5TCY	—		
101*	TLOW	Clock low time	400 kHz mode	1.3	—	µs	Device must operate at a minimum of 10 MHz
			SSP Module	1.5TCY	—		
102*	TR	SDA and SCL rise time	400 kHz mode	20 + 0.1CB	250	ns	CB is specified to be from 10-400 pF
103*	TF	SDA and SCL fall time	400 kHz mode	20 + 0.1CB	250	ns	CB is specified to be from 10-400 pF
90*	TSU:STA	Start condition setup time	400 kHz mode	1.3	—	µs	Only relevant for Repeated Start condition
91*	THD:STA	Start condition hold time	400 kHz mode	0.6	—	µs	After this period the first clock pulse is generated
106*	THD:DAT	Data input hold time	400 kHz mode	0	0.9	µs	
107*	TSU:DAT	Data input setup time	400 kHz mode	100	—	ns	**(Note 2)**
92*	TSU:STO	Stop condition setup time	400 kHz mode	0.6	—	µs	
109*	TAA	Output valid from clock	400 kHz mode	—	—	ns	**(Note 1)**
110*	TBUF	Bus free time	400 kHz mode	1.3	—	µs	Time the bus must be free before a new transmission can start
	CB	Bus capacitive loading		—	400	pF	

* These parameters are characterized but not tested.

Note 1: As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCL to avoid unintended generation of Start or Stop conditions.

2: A Fast mode (400 kHz) I²C bus device can be used in a Standard mode (100 kHz) I²C bus system, but the requirement TSU:DAT ≥ 250 ns must then be met. This will automatically be the case if the device does not stretch the low period of the SCL signal. If such a device does stretch the low period of the SCL signal, it must output the next data bit to the SDA line TR max. + TSU:DAT = 1000 + 250 = 1250 ns (according to the Standard mode I²C bus specification), before the SCL line is released.

20.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES

The graphs and tables provided in this section are for **design guidance** and are **not tested**.

In some graphs or tables, the data presented are **outside specified operating range** (i.e., outside specified VDD range). This is for **information only** and devices are ensured to operate properly only within the specified range.

Note:	The graphs and tables provided following this note are a statistical summary based on a limited number of samples and are provided for informational purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs or tables, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore, outside the warranted range.

"Typical" represents the mean of the distribution at 25°C. "Maximum" or "minimum" represents (mean + 3σ) or (mean - 3σ) respectively, where σ is a standard deviation, over each temperature range.

FIGURE 20-1: TYPICAL IDD vs. FOSC OVER VDD (EC MODE)

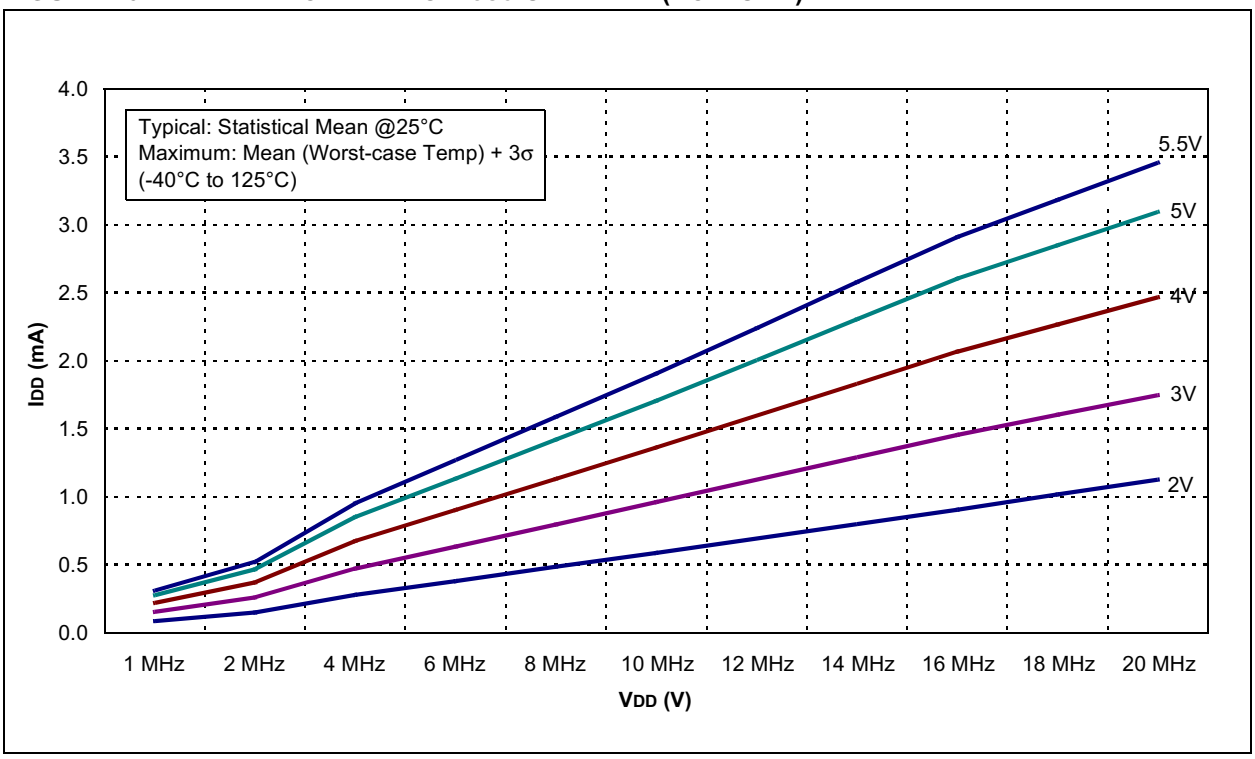


FIGURE 20-2: MAXIMUM IDD vs. FOSC OVER VDD (EC MODE)

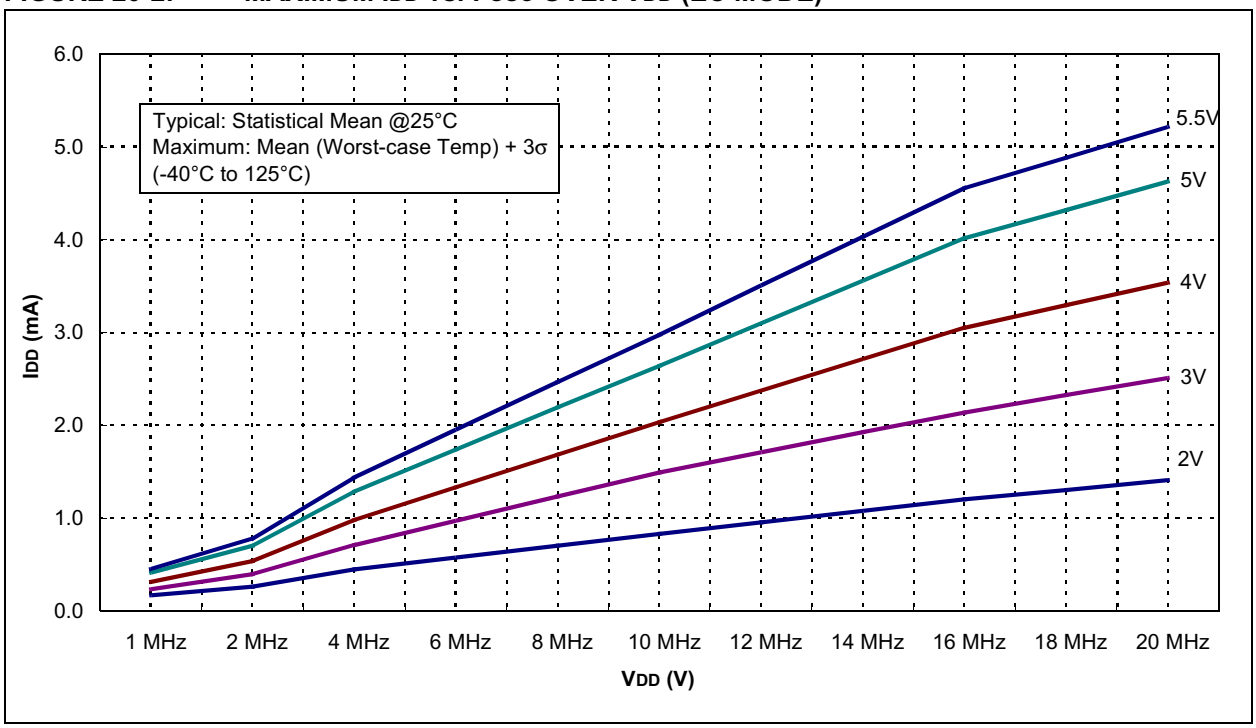


FIGURE 20-3: TYPICAL IDD vs. FOSC OVER VDD (HS MODE)

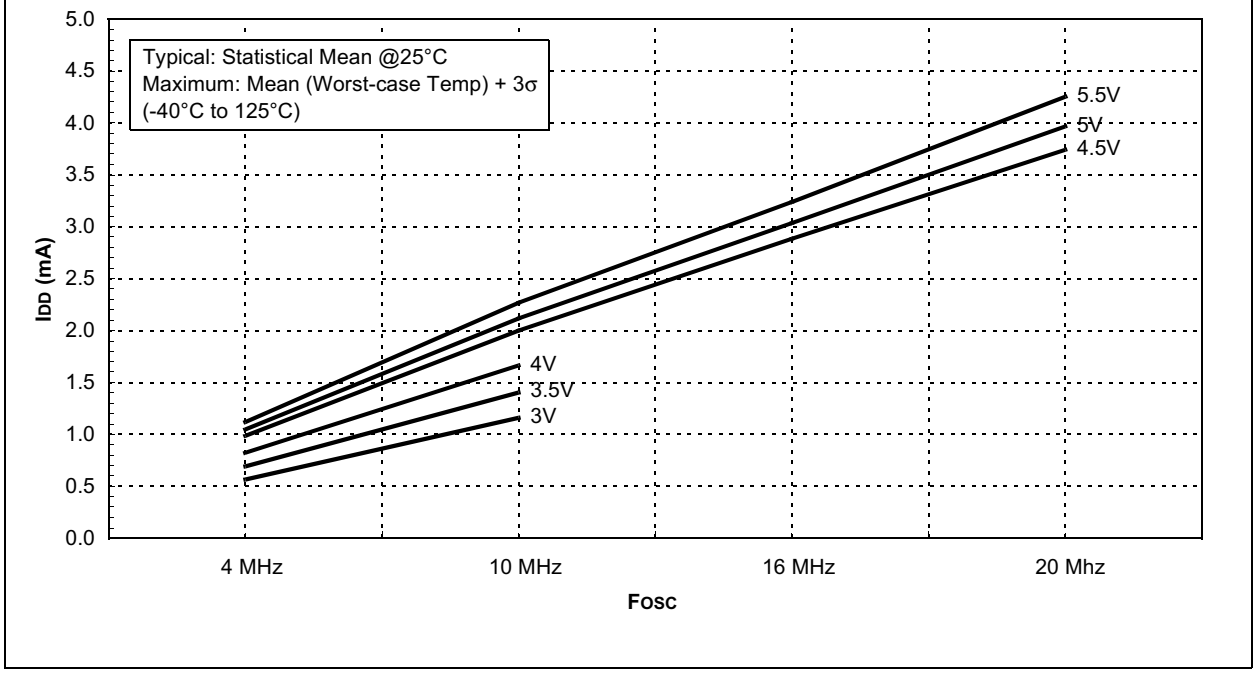


FIGURE 20-4: MAXIMUM IDD vs. FOSC OVER VDD (HS MODE)

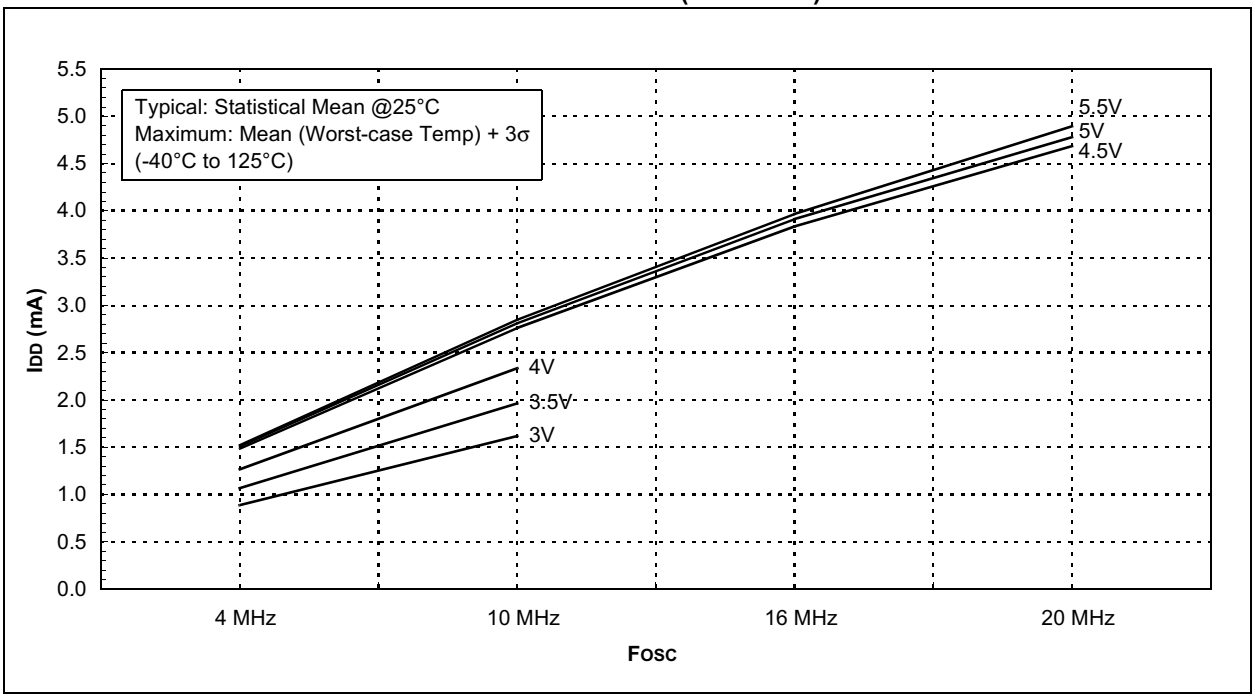


FIGURE 20-5: TYPICAL IDD vs. VDD OVER FOSC (XT MODE)

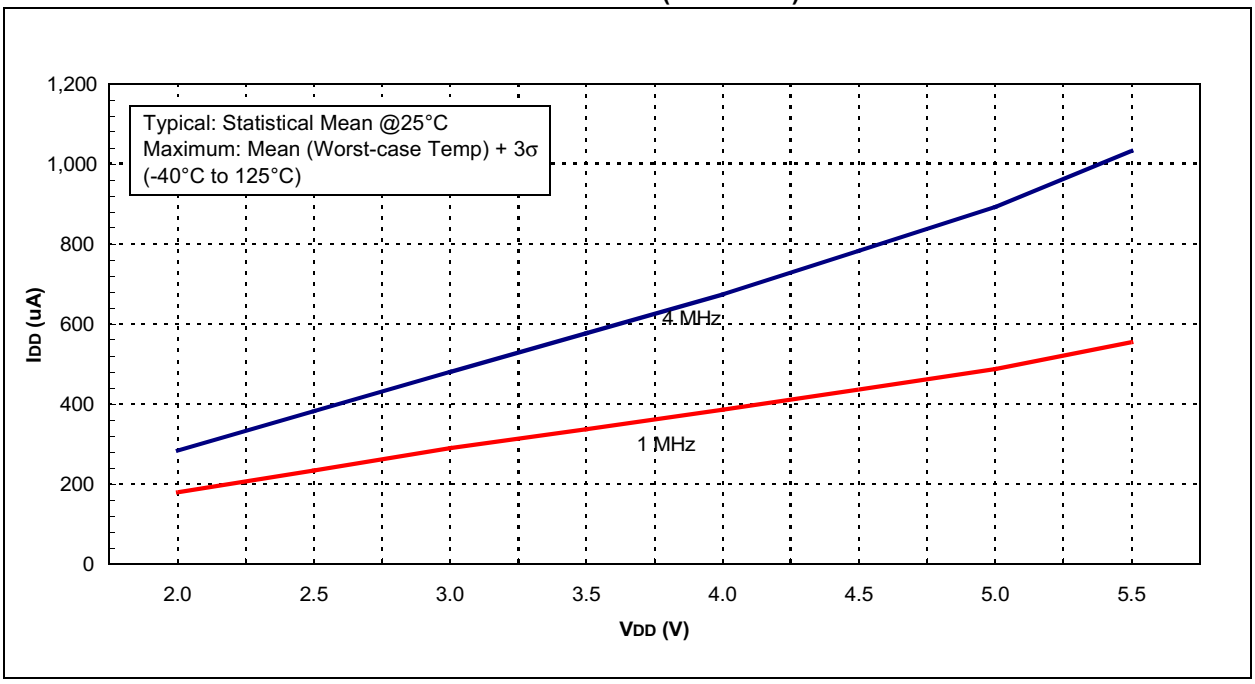


FIGURE 20-6: MAXIMUM IDD vs. VDD OVER FOSC (XT MODE)

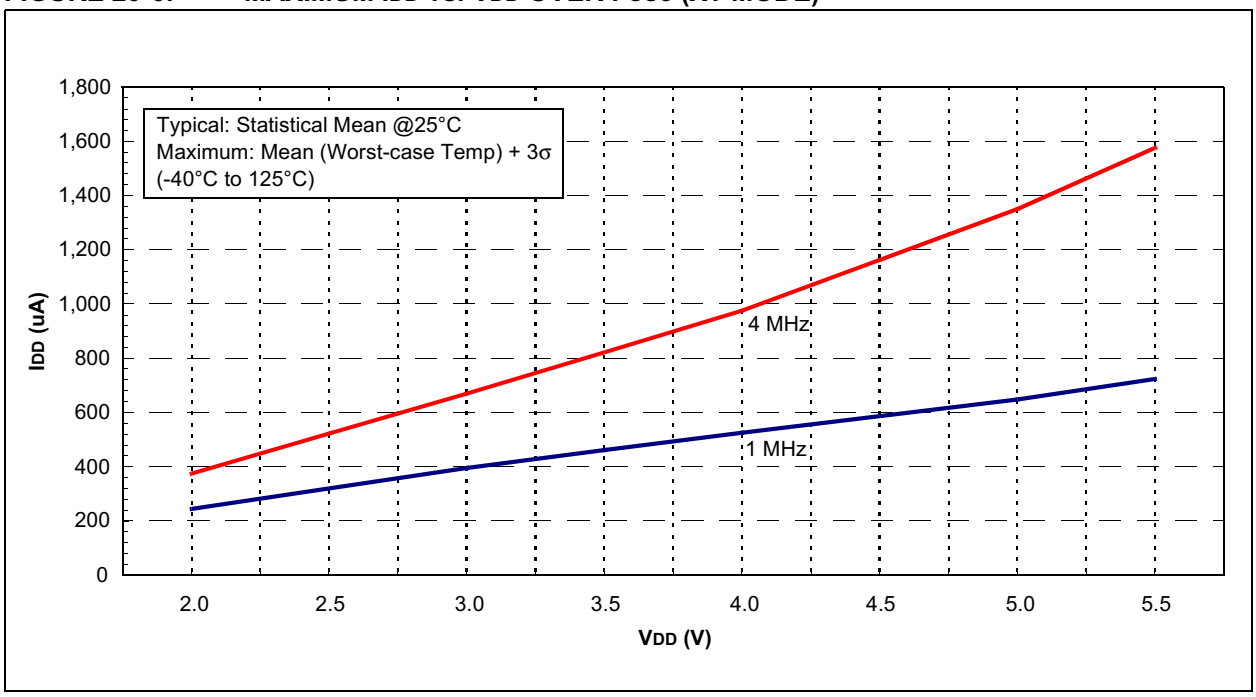


FIGURE 20-7: TYPICAL IDD vs. VDD OVER FOSC (EXTRC MODE)

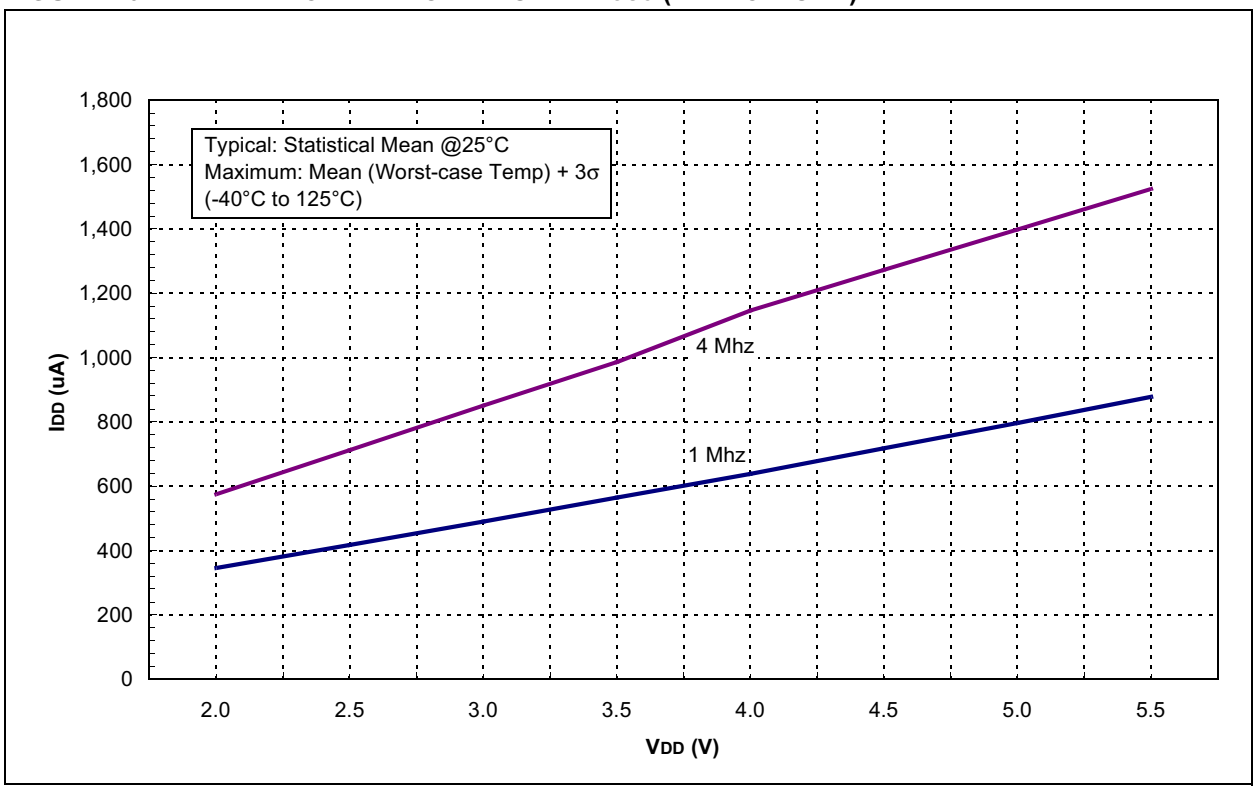


FIGURE 20-8: MAXIMUM IDD vs. VDD (EXTRC MODE)

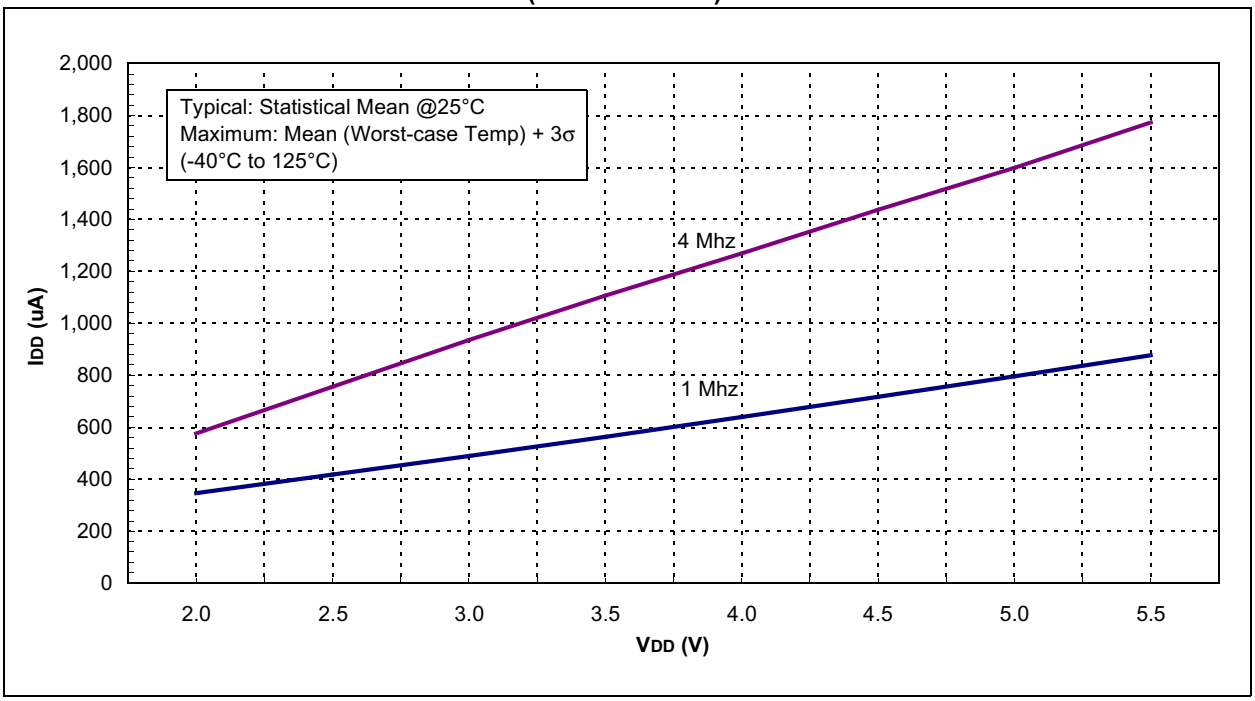


FIGURE 20-9: IDD vs. VDD OVER FOSC (LFINTOSC MODE, 31 kHz)

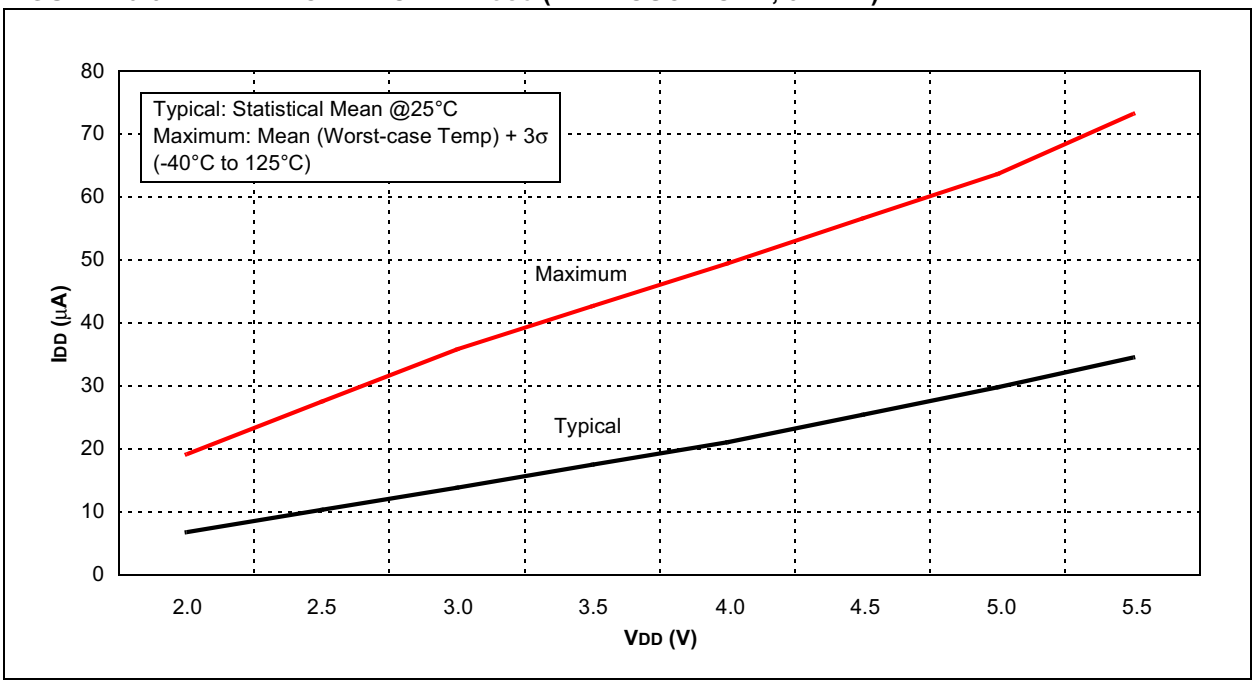


FIGURE 20-10: IDD vs. VDD (LP MODE)

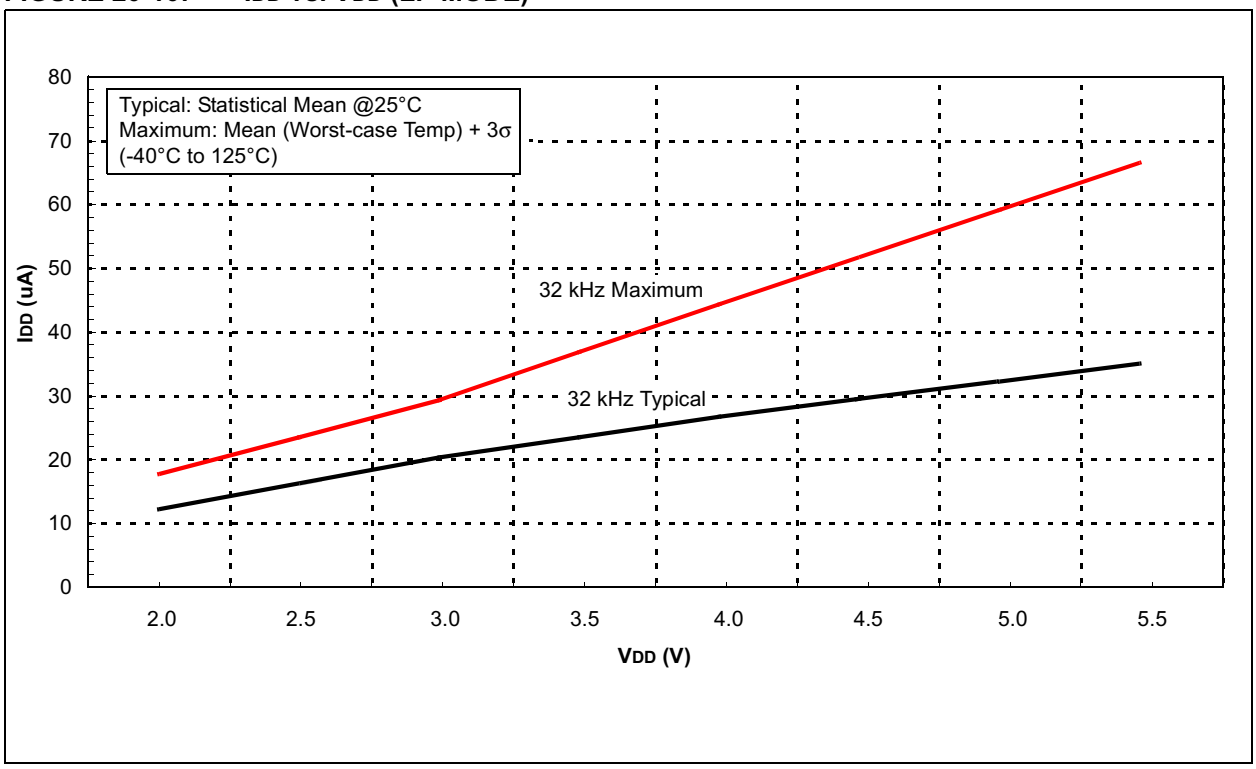


FIGURE 20-11: TYPICAL IDD vs. FOSC OVER VDD (HFINTOSC MODE)

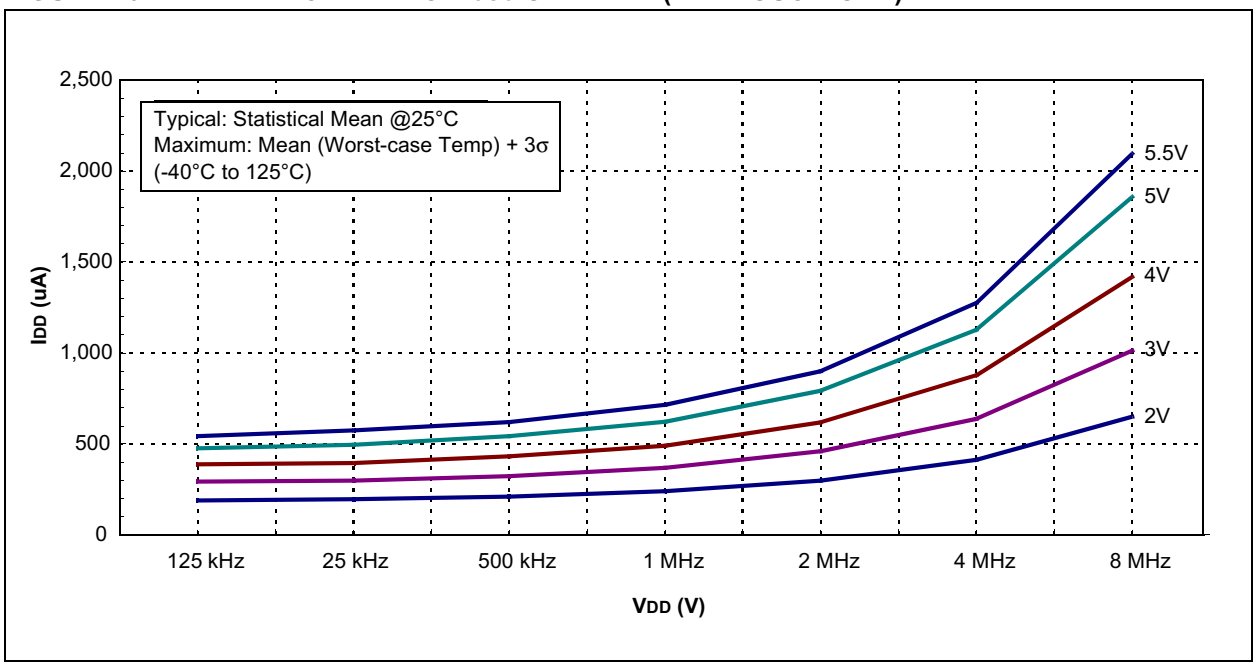


FIGURE 20-12: MAXIMUM IDD vs. FOSC OVER VDD (HFINTOSC MODE)

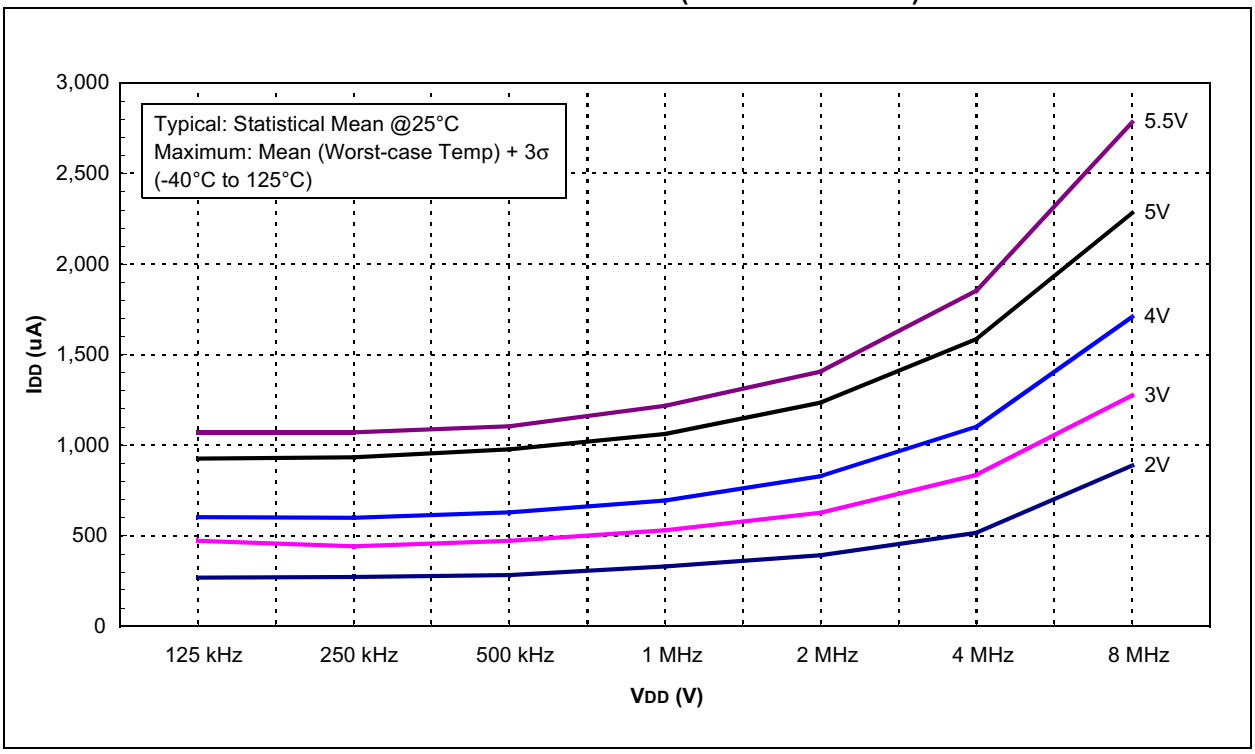


FIGURE 20-13: TYPICAL IPD vs. VDD (SLEEP MODE, ALL PERIPHERALS DISABLED)

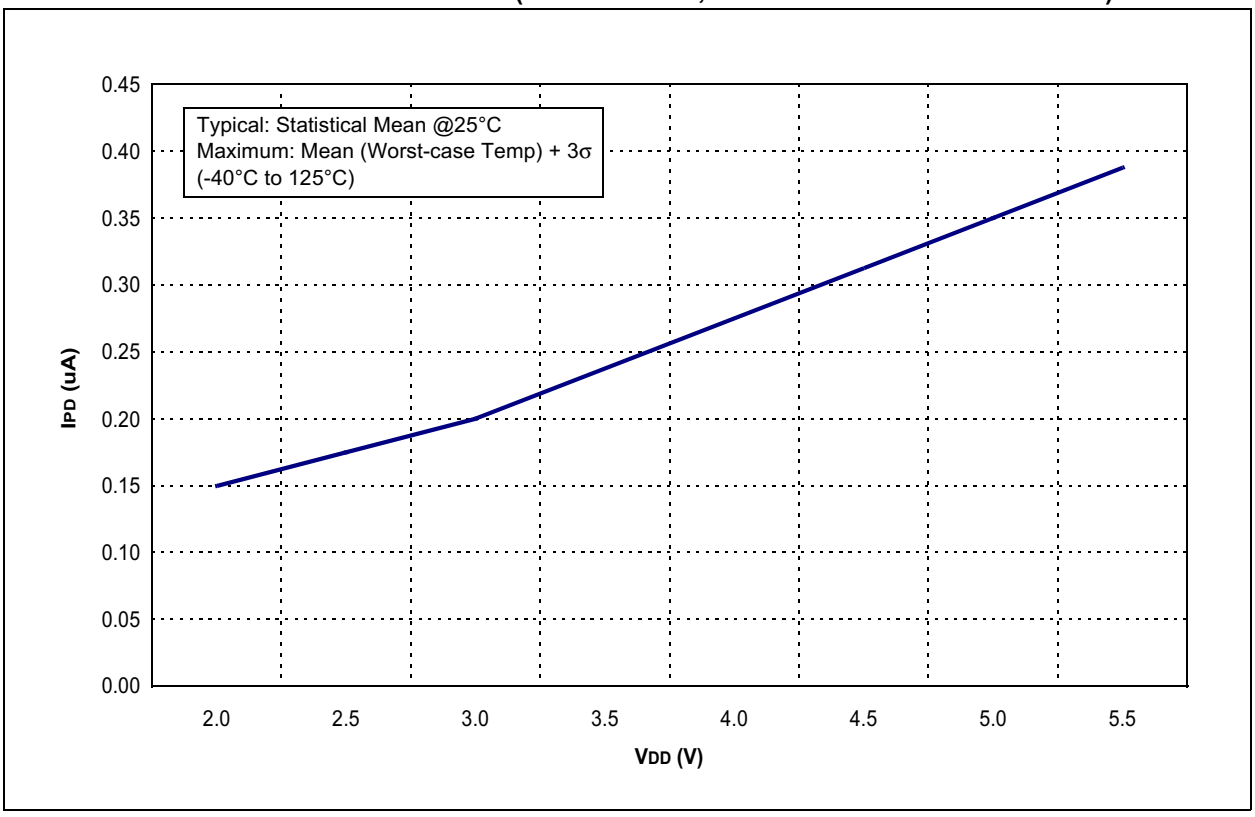


FIGURE 20-14: MAXIMUM IPD vs. VDD (SLEEP MODE, ALL PERIPHERALS DISABLED)

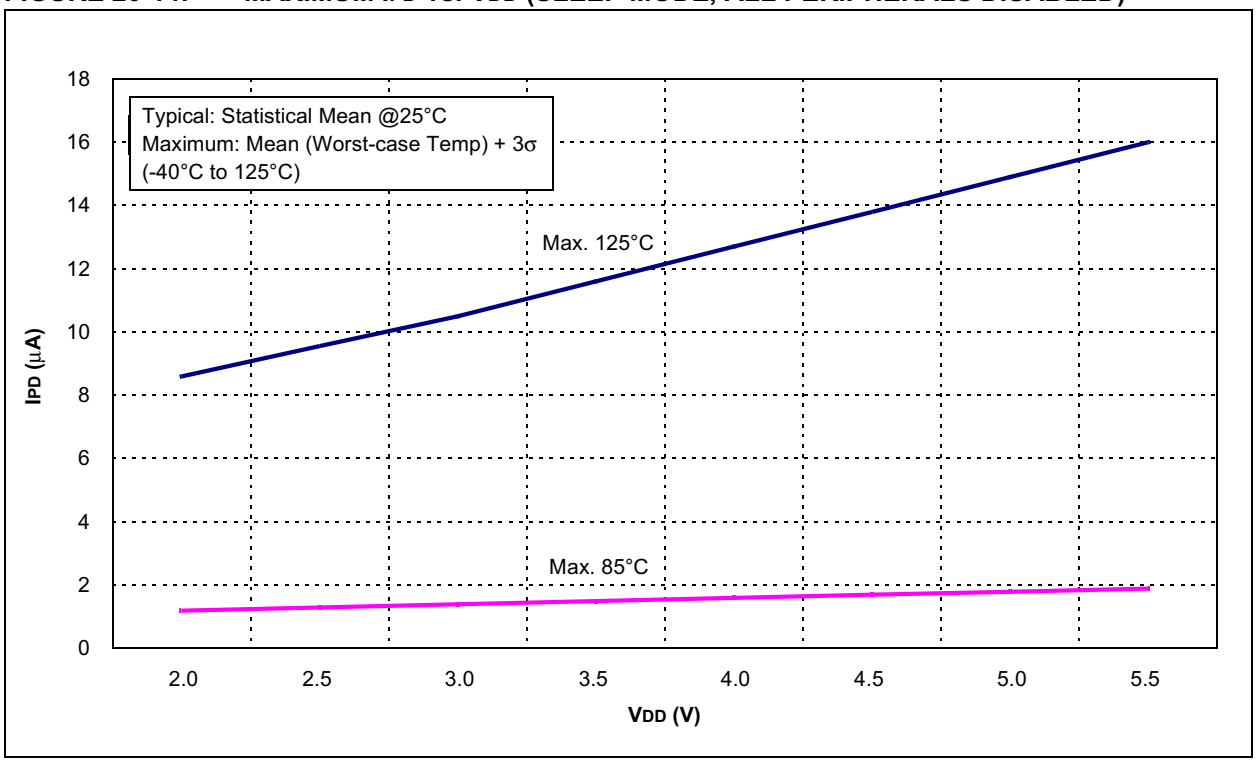


FIGURE 20-15: COMPARATOR IPD vs. VDD (BOTH COMPARATORS ENABLED)

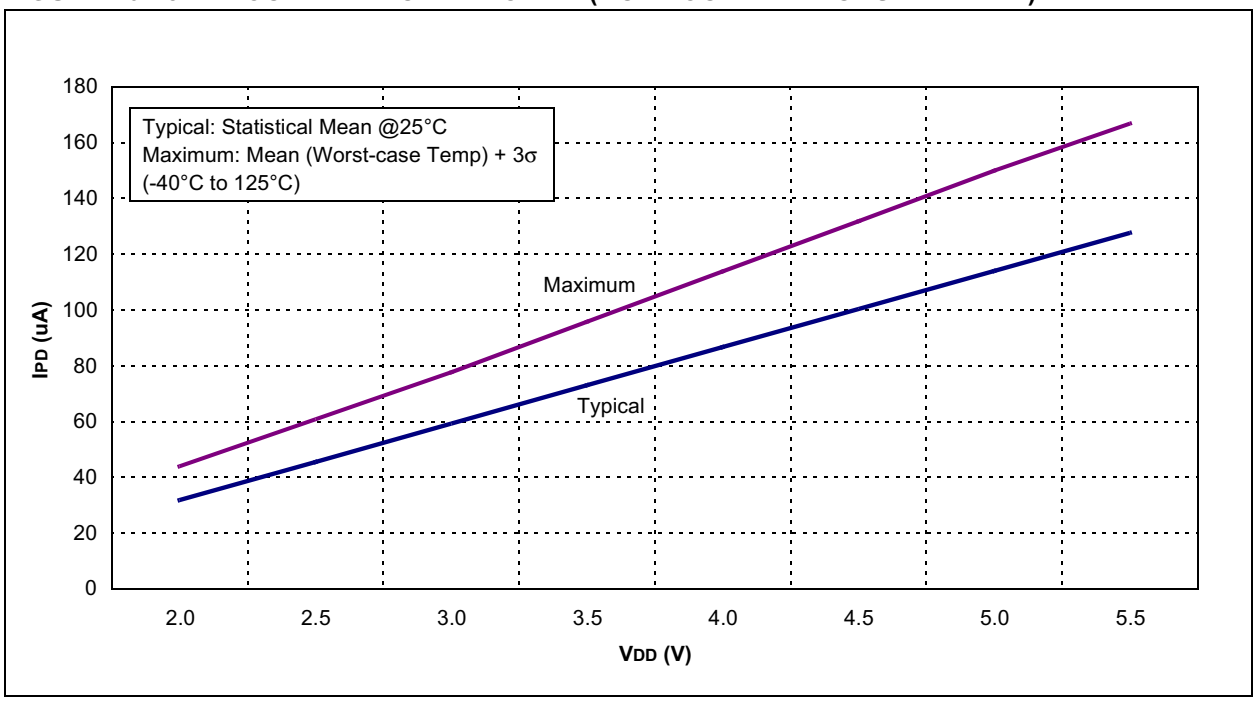


FIGURE 20-16: BOR IPD vs. VDD OVER TEMPERATURE

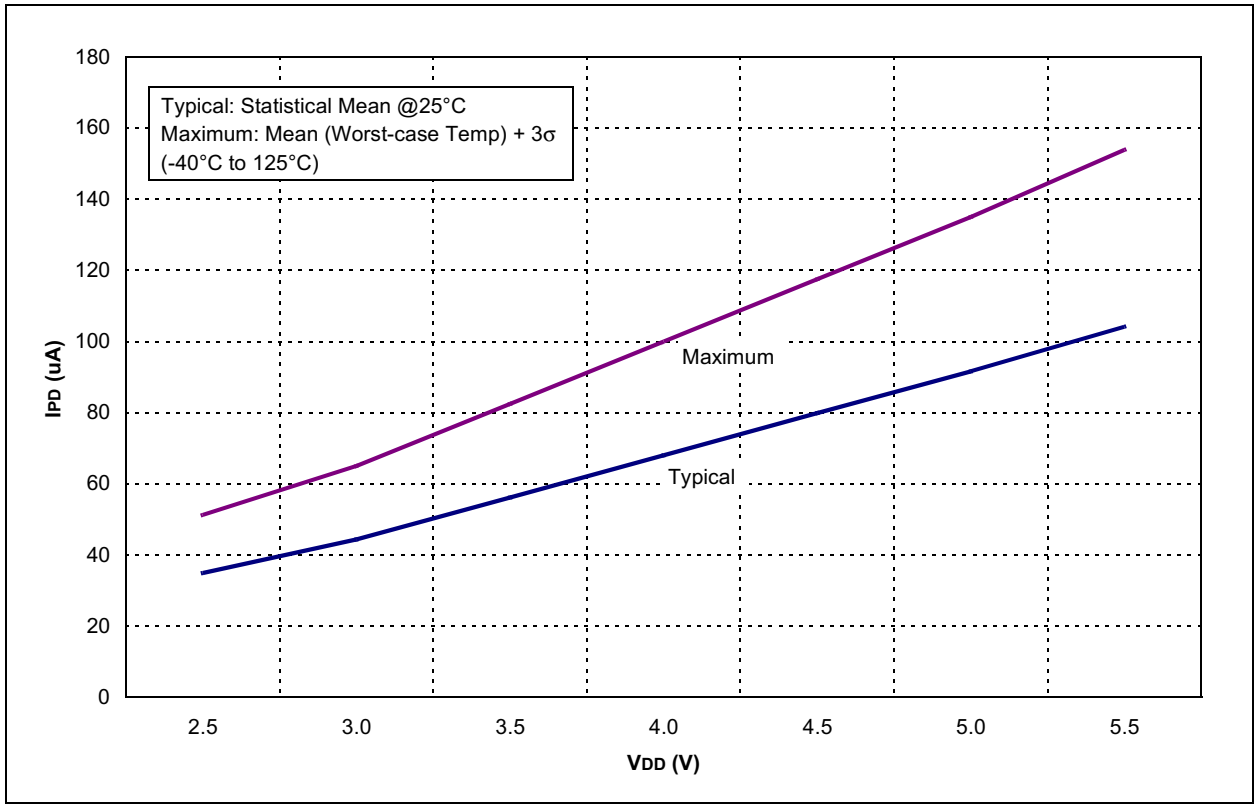


FIGURE 20-17: TYPICAL WDT IPD vs. VDD (25°C)

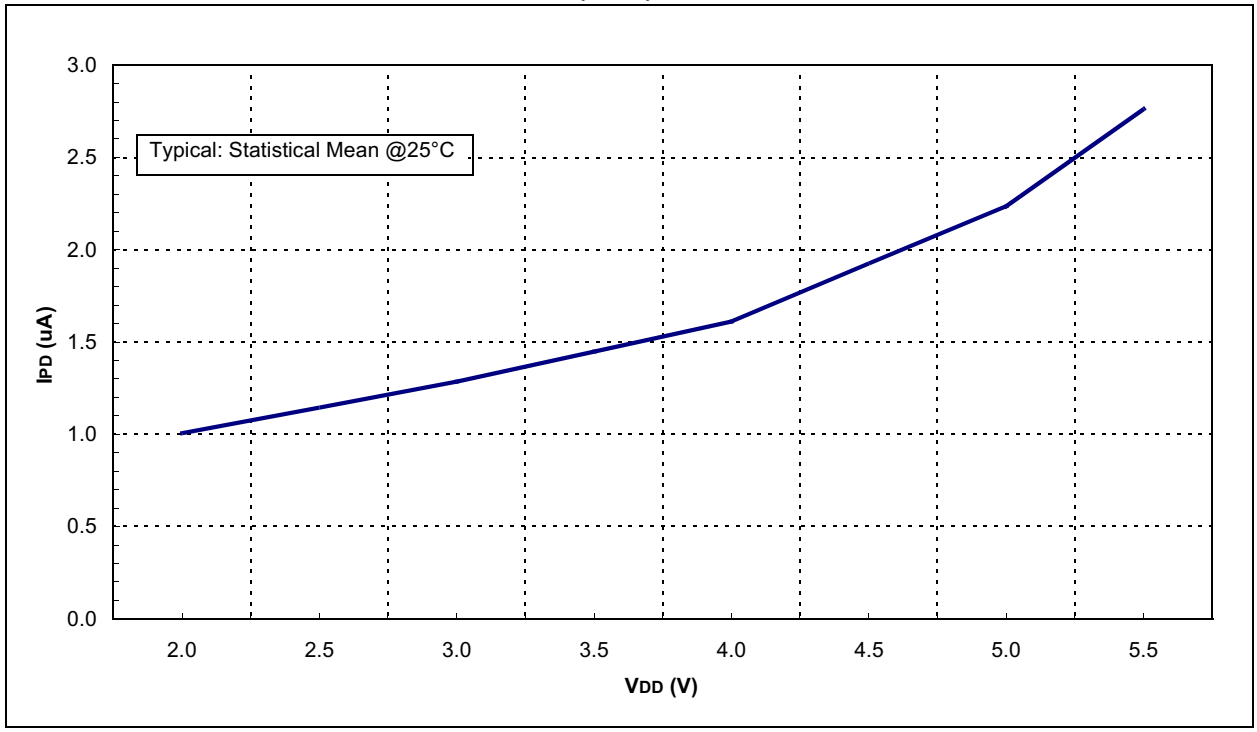


FIGURE 20-18: MAXIMUM WDT IPD vs. VDD OVER TEMPERATURE

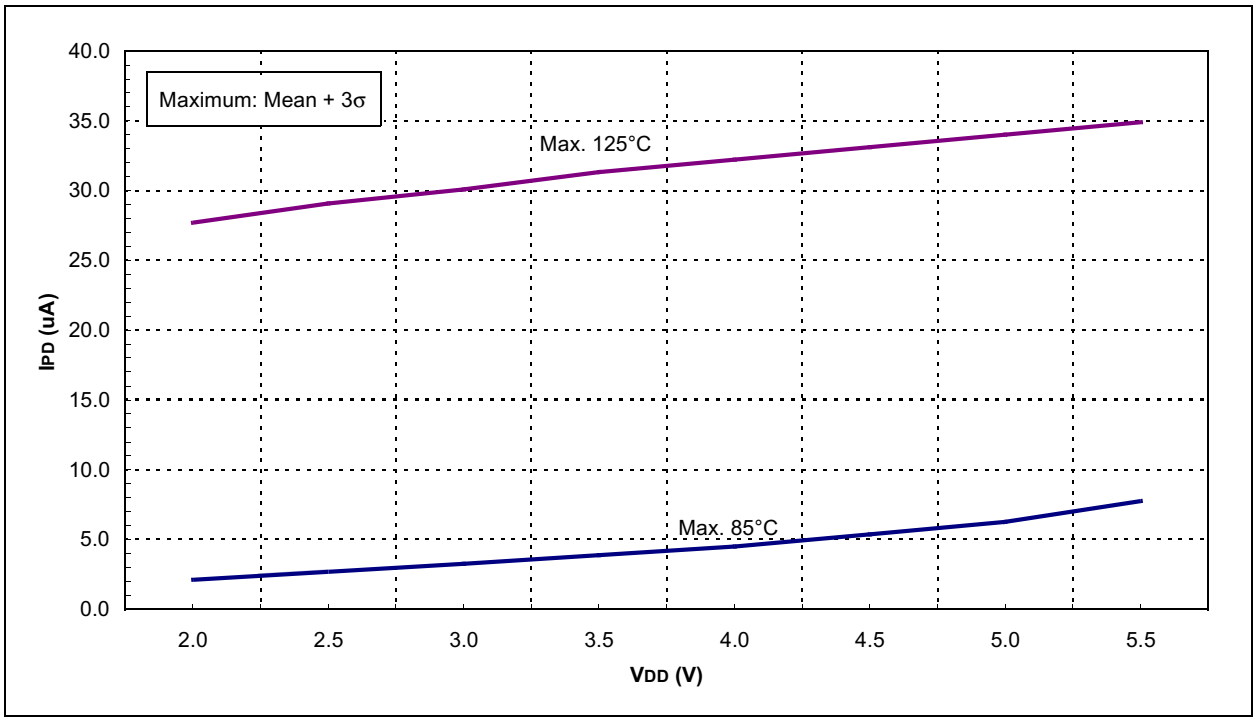


FIGURE 20-19: WDT PERIOD vs. VDD OVER TEMPERATURE

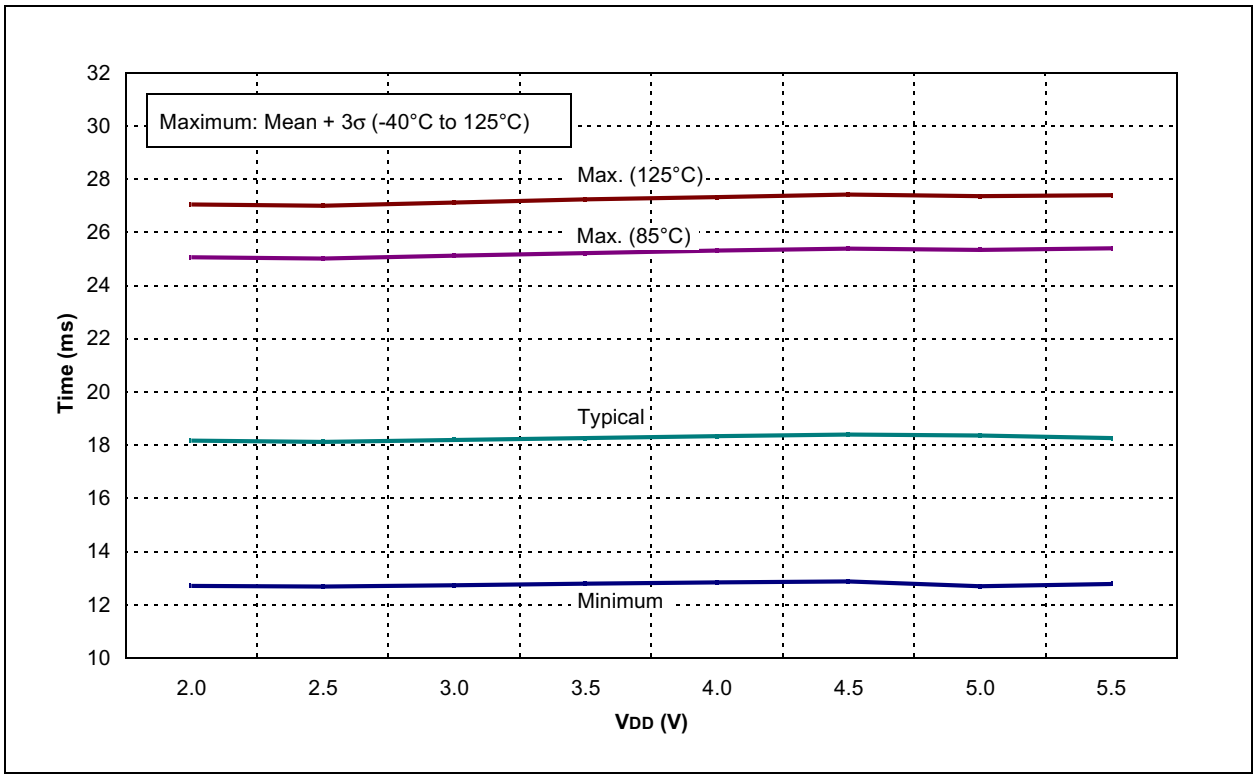


FIGURE 20-20: WDT PERIOD vs. TEMPERATURE (V_{DD} = 5.0V)

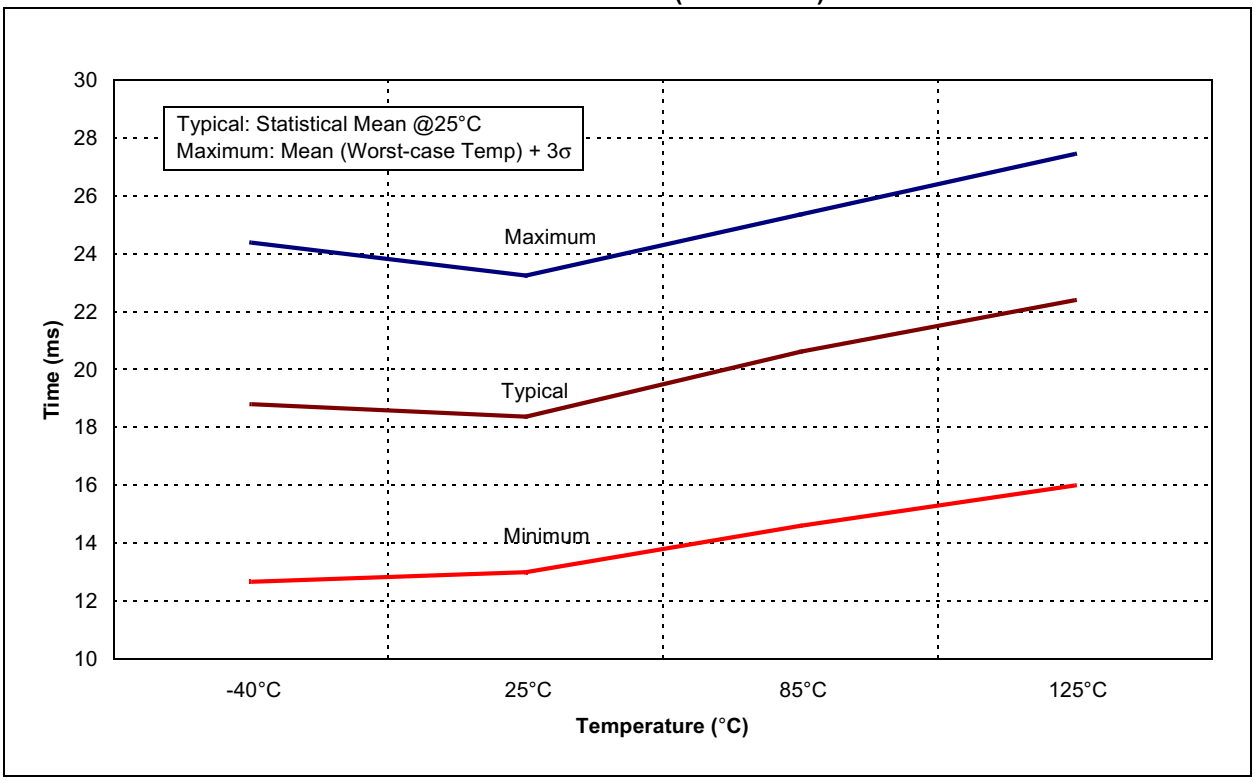


FIGURE 20-21: CV_{REF} I_{PD} vs. V_{DD} OVER TEMPERATURE (HIGH RANGE)

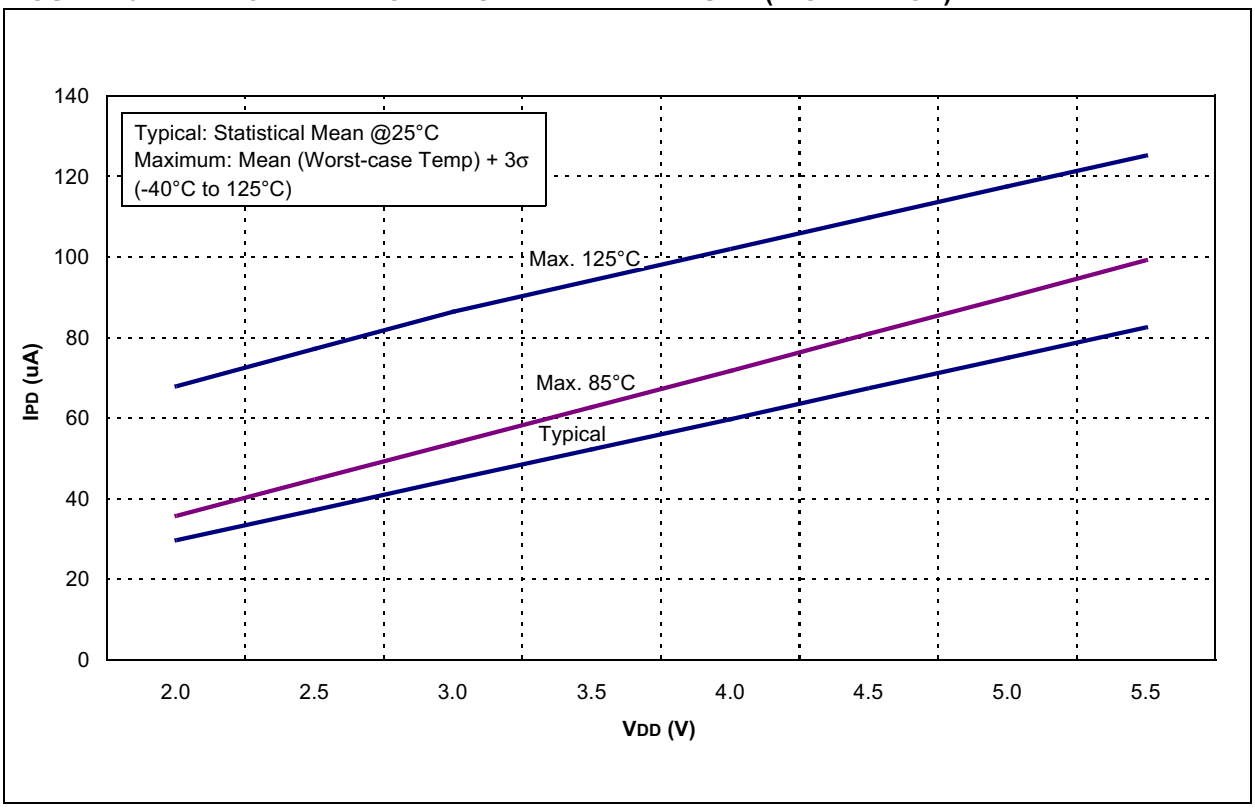


FIGURE 20-22: CVREF IPD vs. VDD OVER TEMPERATURE (LOW RANGE)

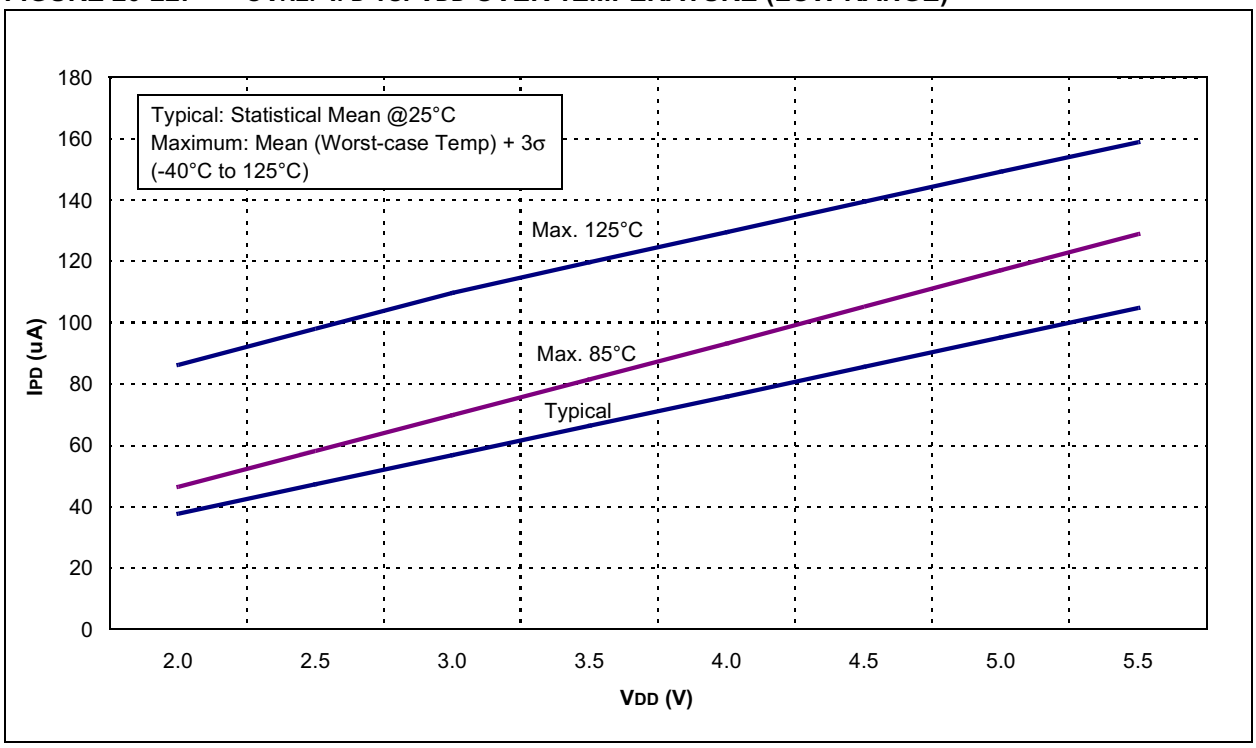


FIGURE 20-23: LVD IPD vs. VDD OVER TEMPERATURE

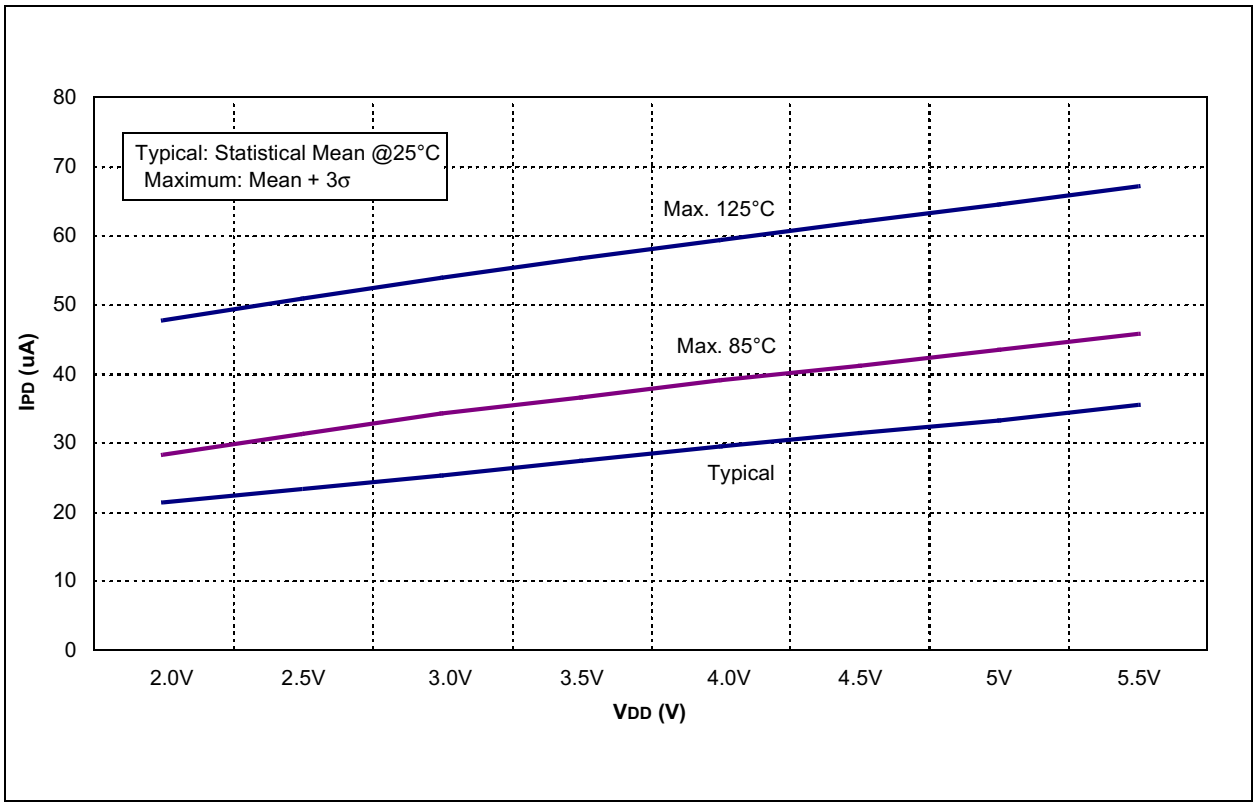


FIGURE 20-24: T1OSC IPD vs. VDD OVER TEMPERATURE (32 kHz)

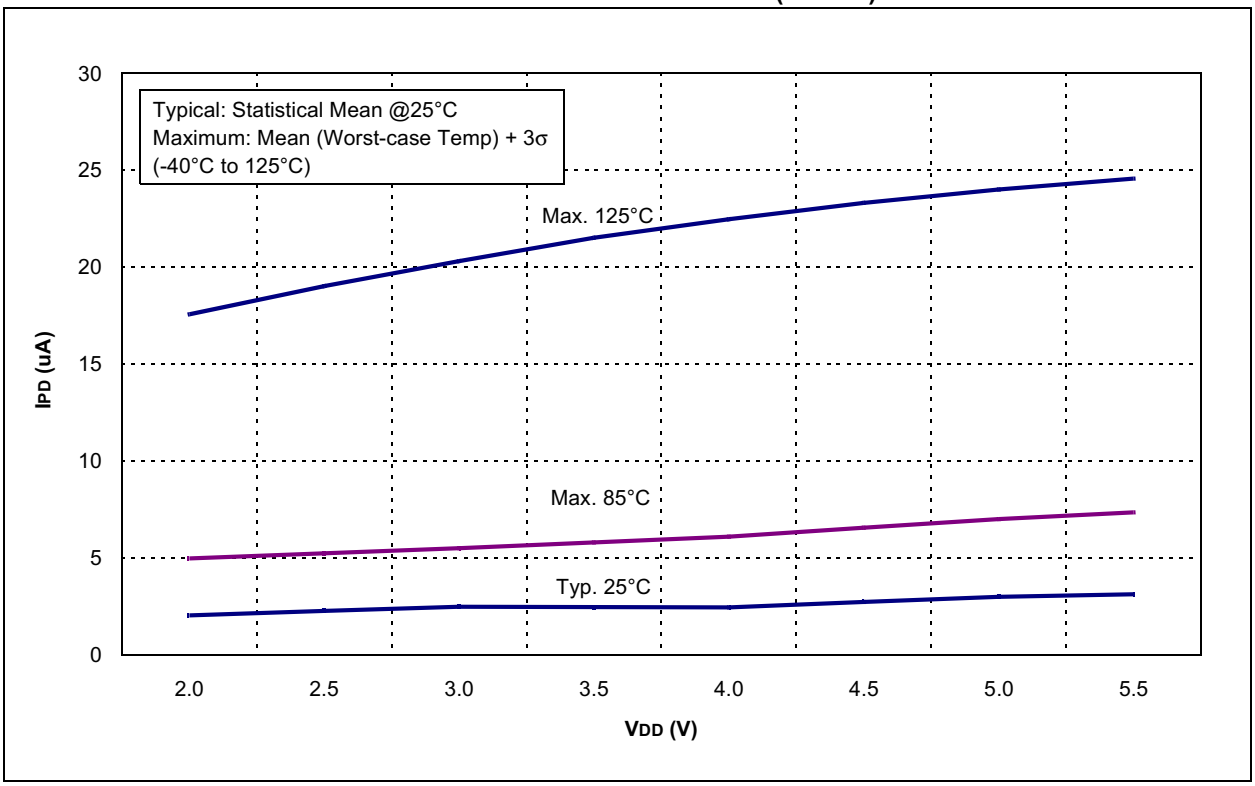


FIGURE 20-25: VOL vs. IOL OVER TEMPERATURE (VDD = 3.0V)

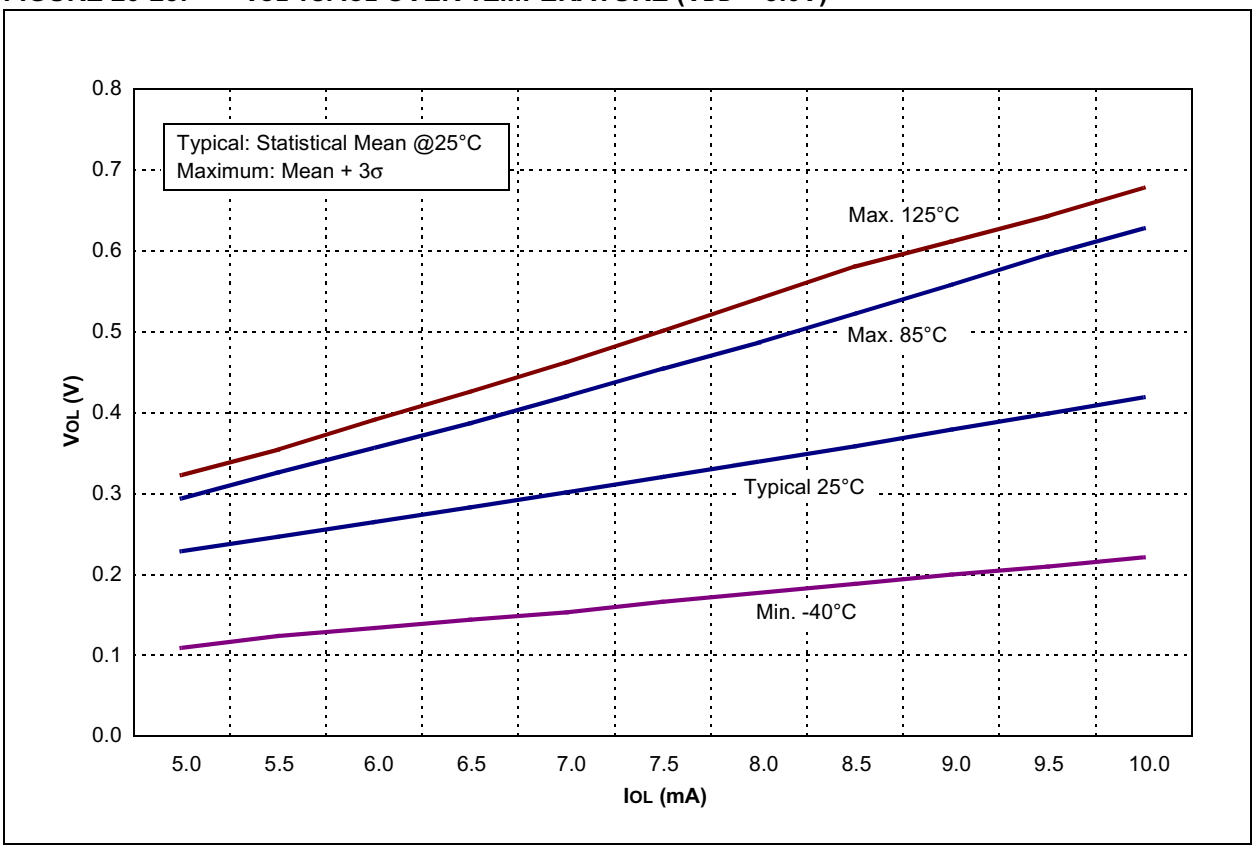


FIGURE 20-26: V_{OL} vs. I_{OL} OVER TEMPERATURE (V_{DD} = 5.0V)

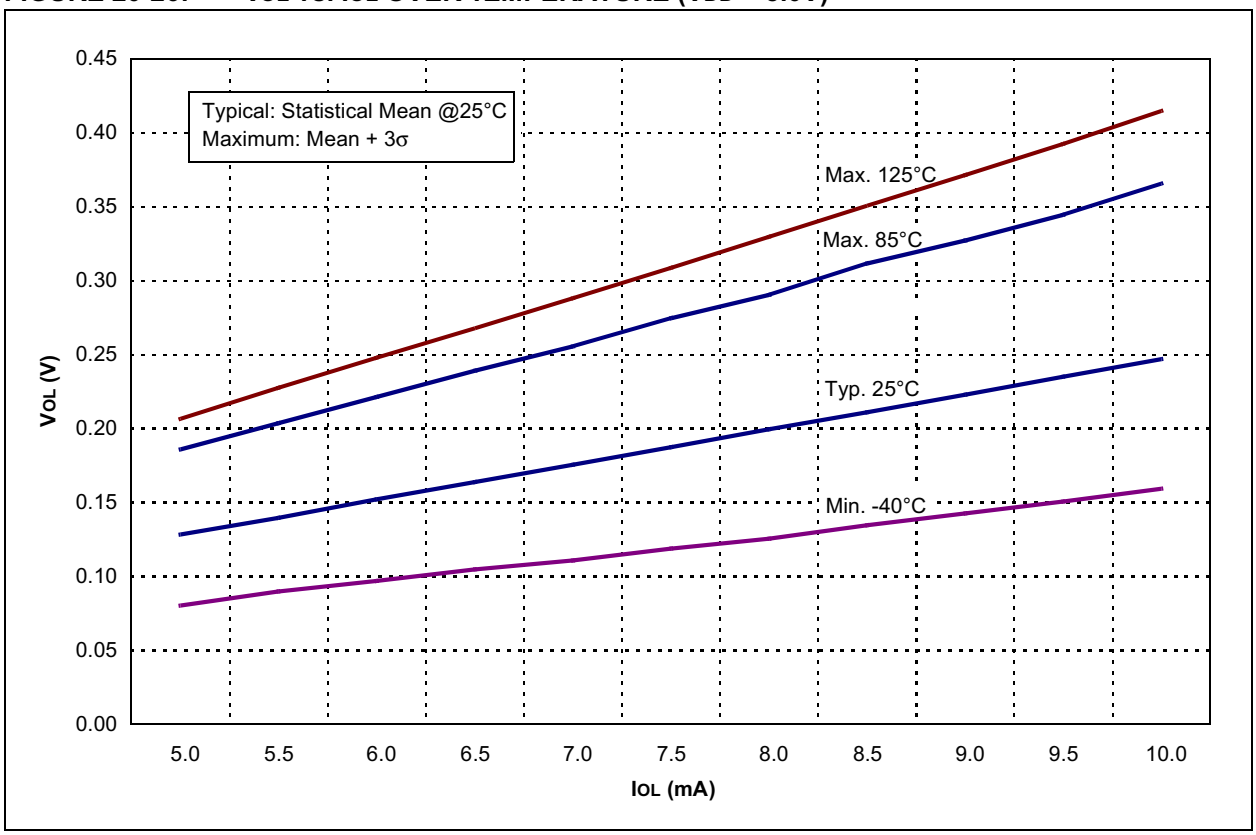


FIGURE 20-27: V_{OH} vs. I_{OH} OVER TEMPERATURE (V_{DD} = 3.0V)

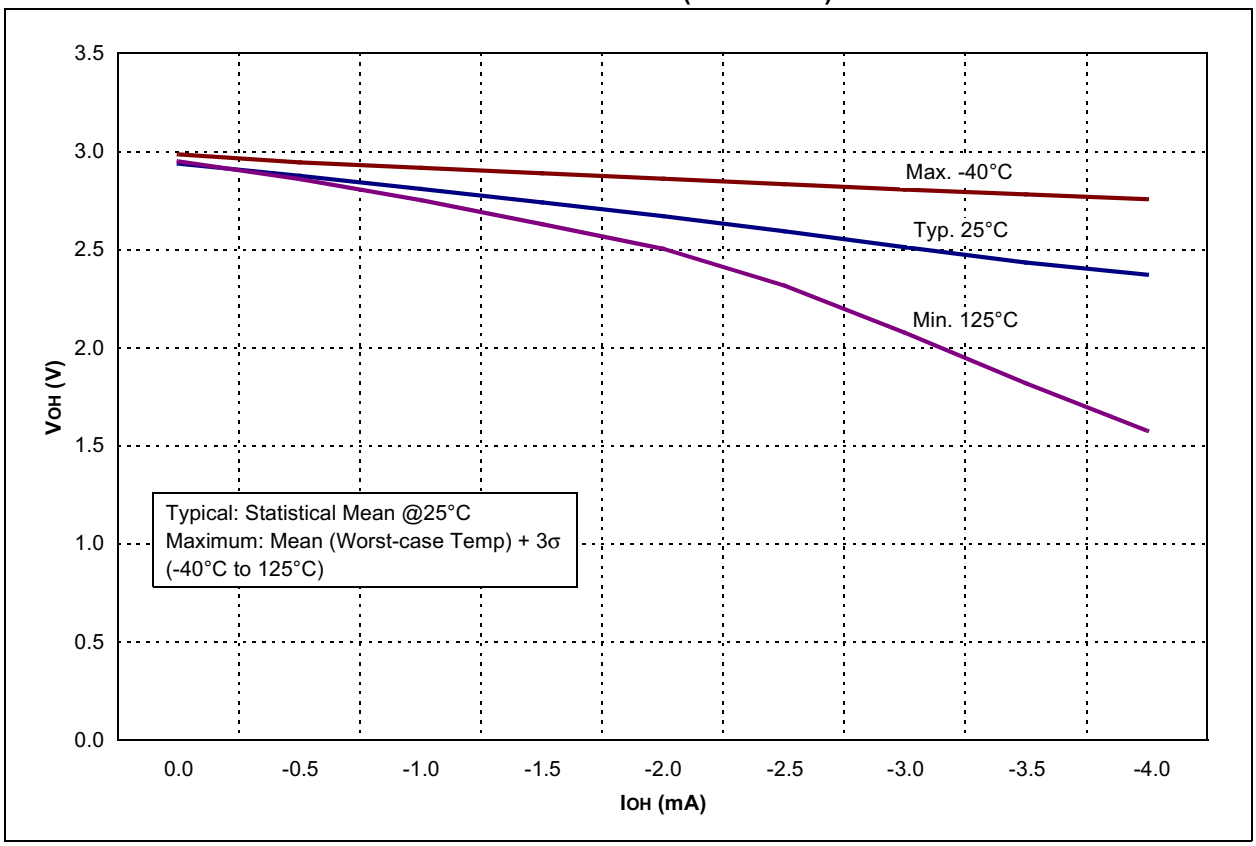


FIGURE 20-28: VOH vs. IOH OVER TEMPERATURE (VDD = 5.0V)

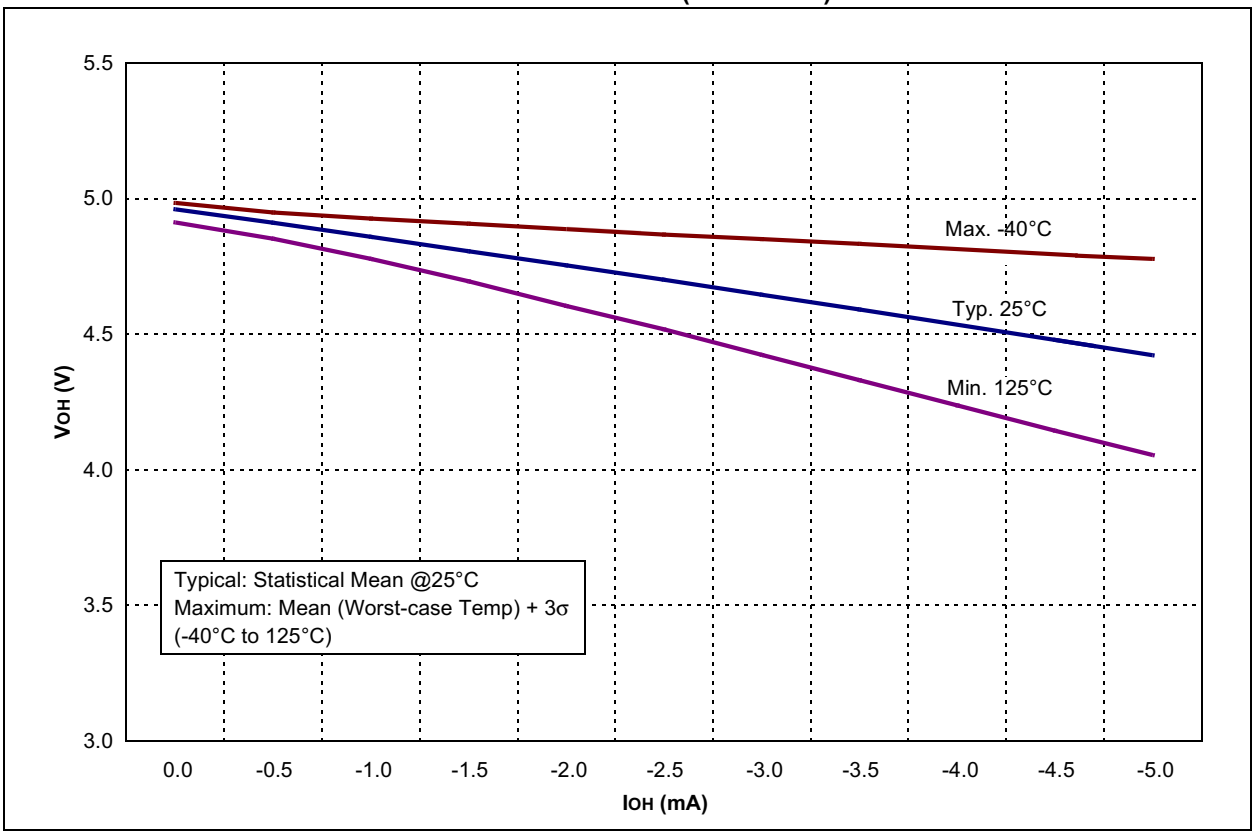


FIGURE 20-29: TTL INPUT THRESHOLD VIN vs. VDD OVER TEMPERATURE

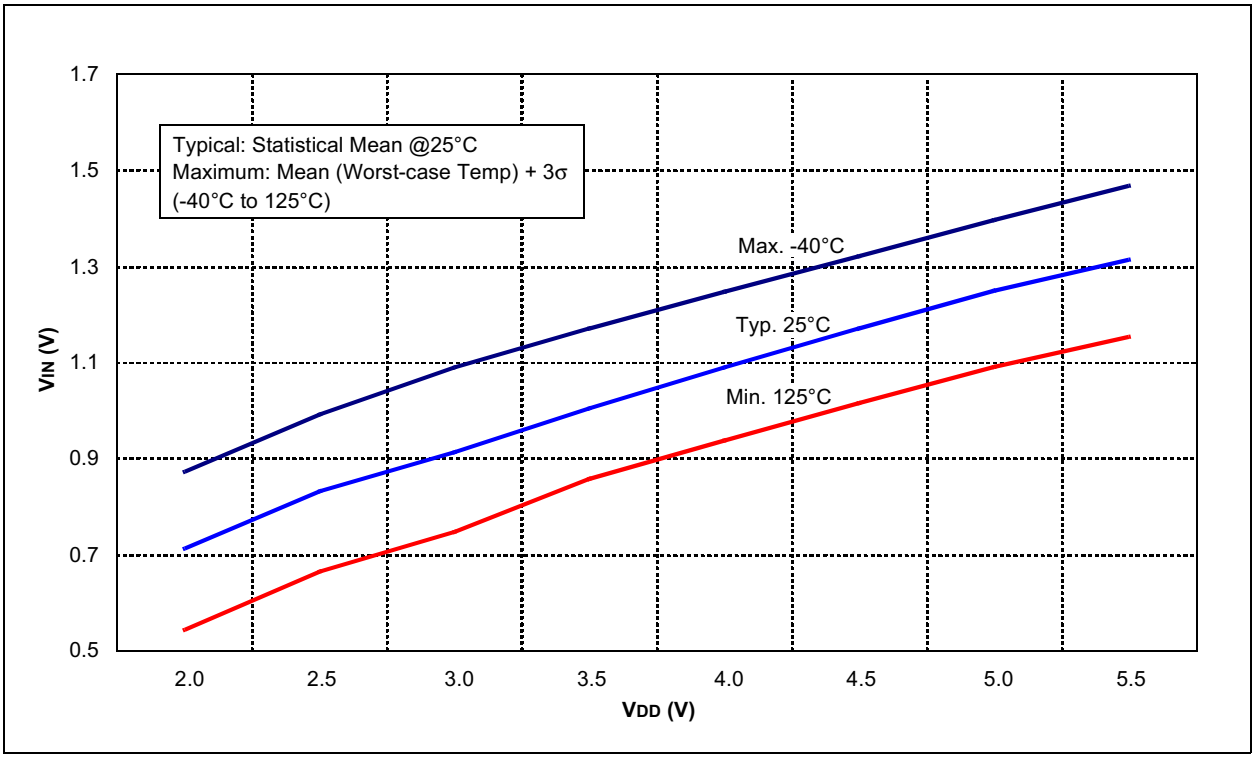


FIGURE 20-30: SCHMITT TRIGGER INPUT THRESHOLD VIN vs. VDD OVER TEMPERATURE

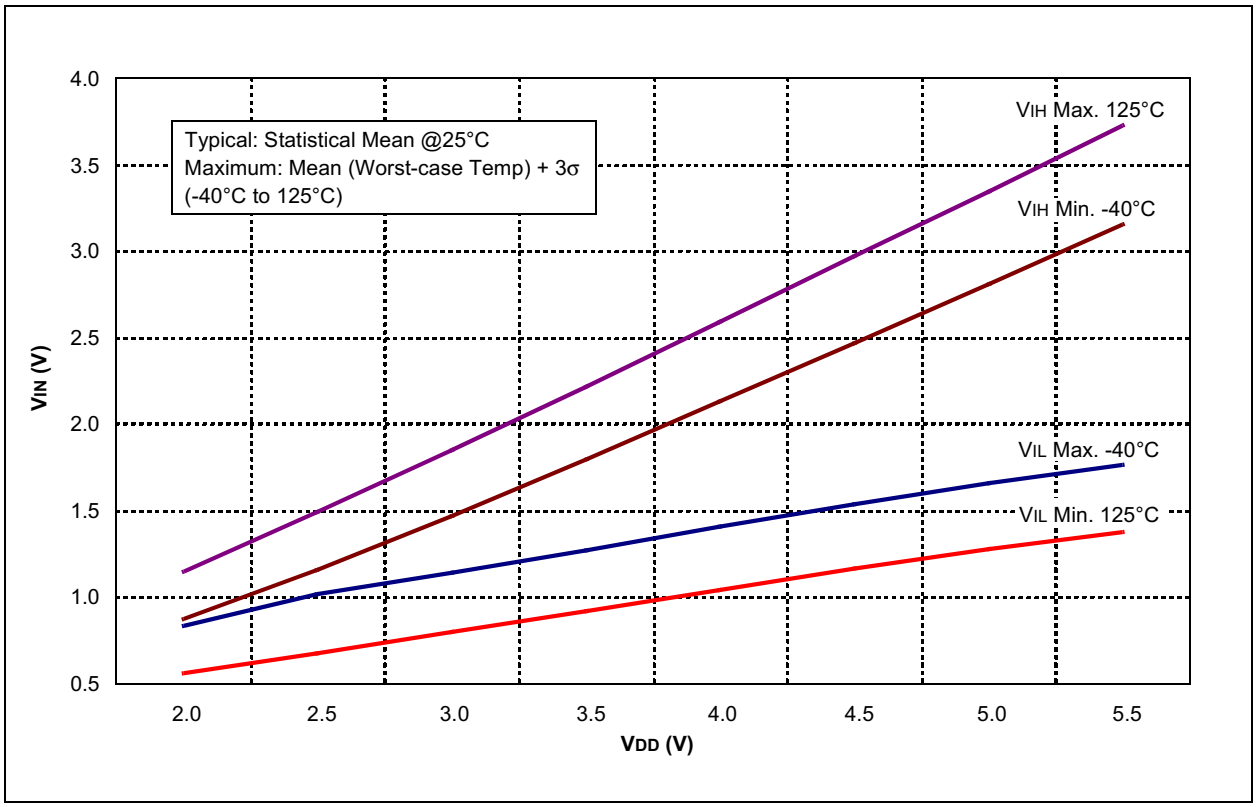


FIGURE 20-31: COMPARATOR RESPONSE TIME (RISING EDGE)

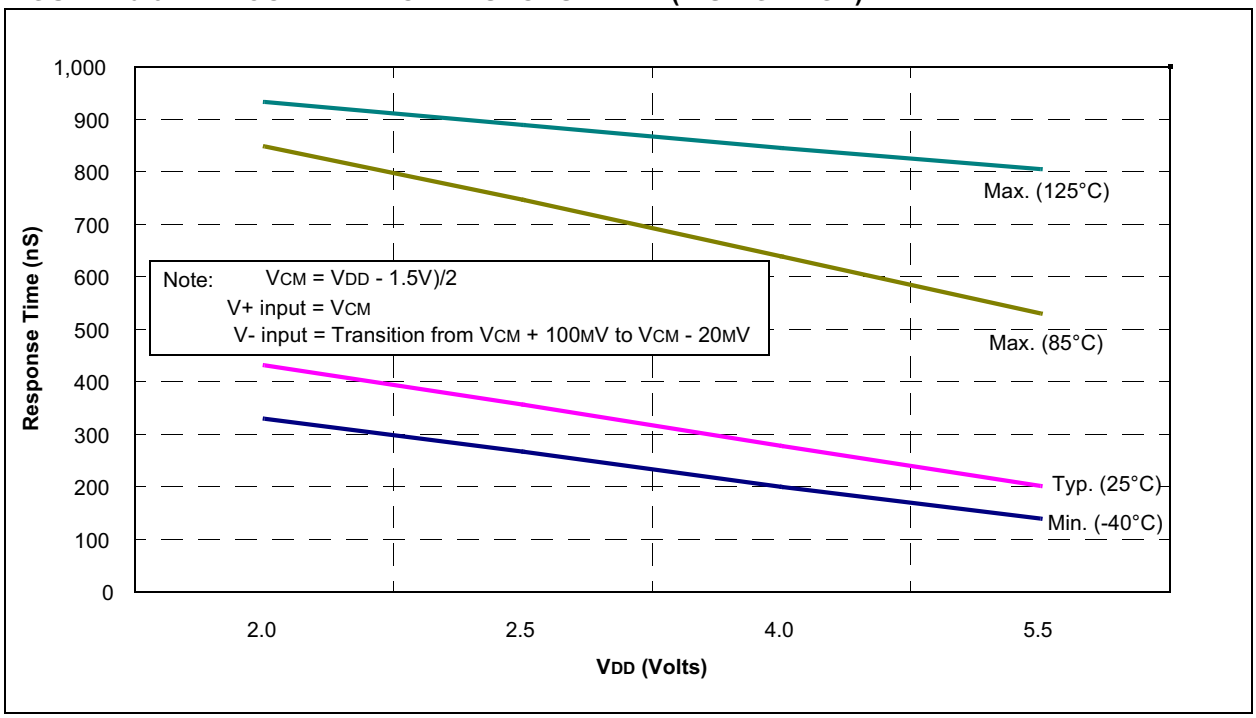


FIGURE 20-32: COMPARATOR RESPONSE TIME (FALLING EDGE)

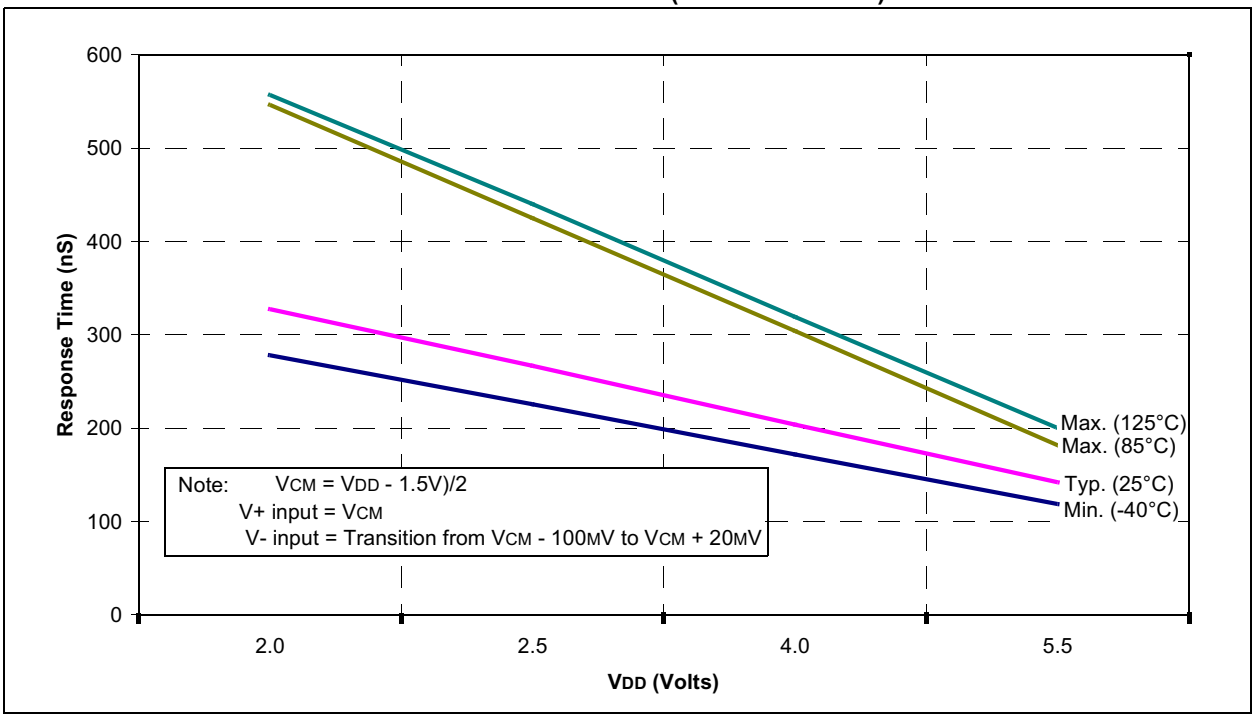


FIGURE 20-33: LFINTOSC FREQUENCY vs. VDD OVER TEMPERATURE (31 kHz)

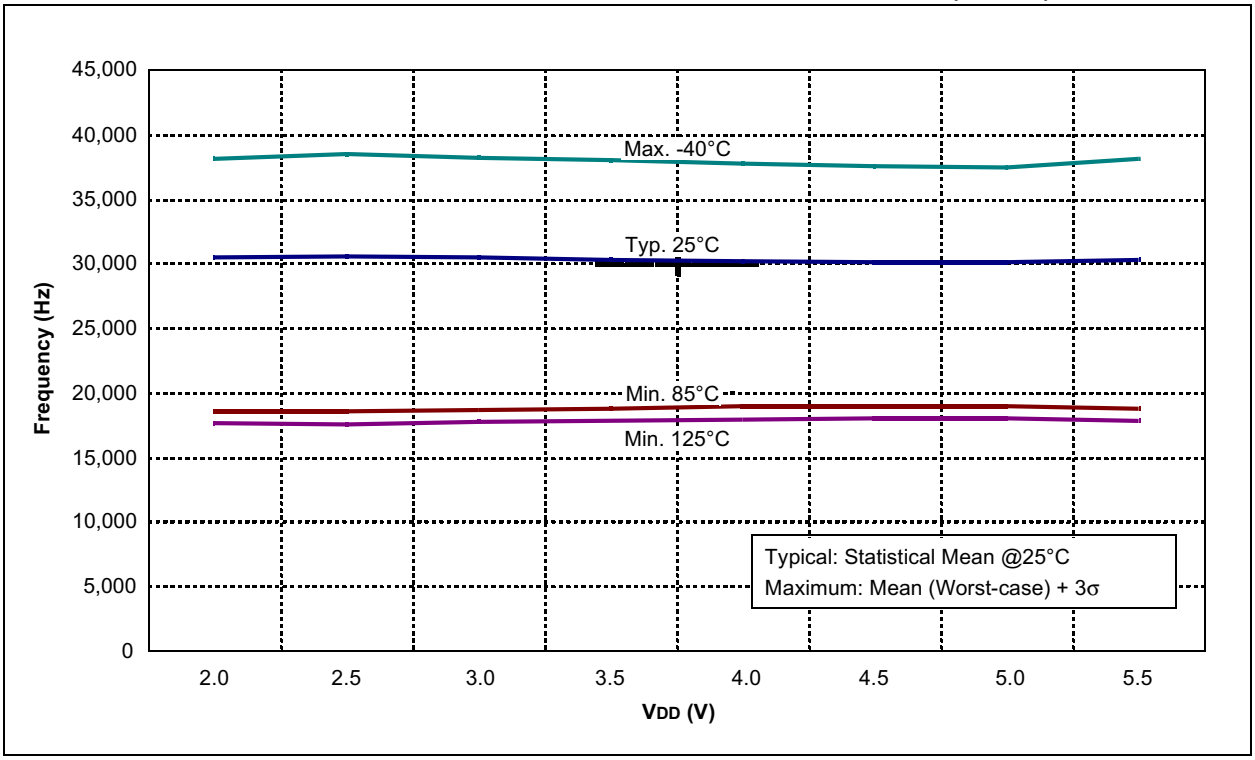


FIGURE 20-34: ADC CLOCK PERIOD vs. VDD OVER TEMPERATURE

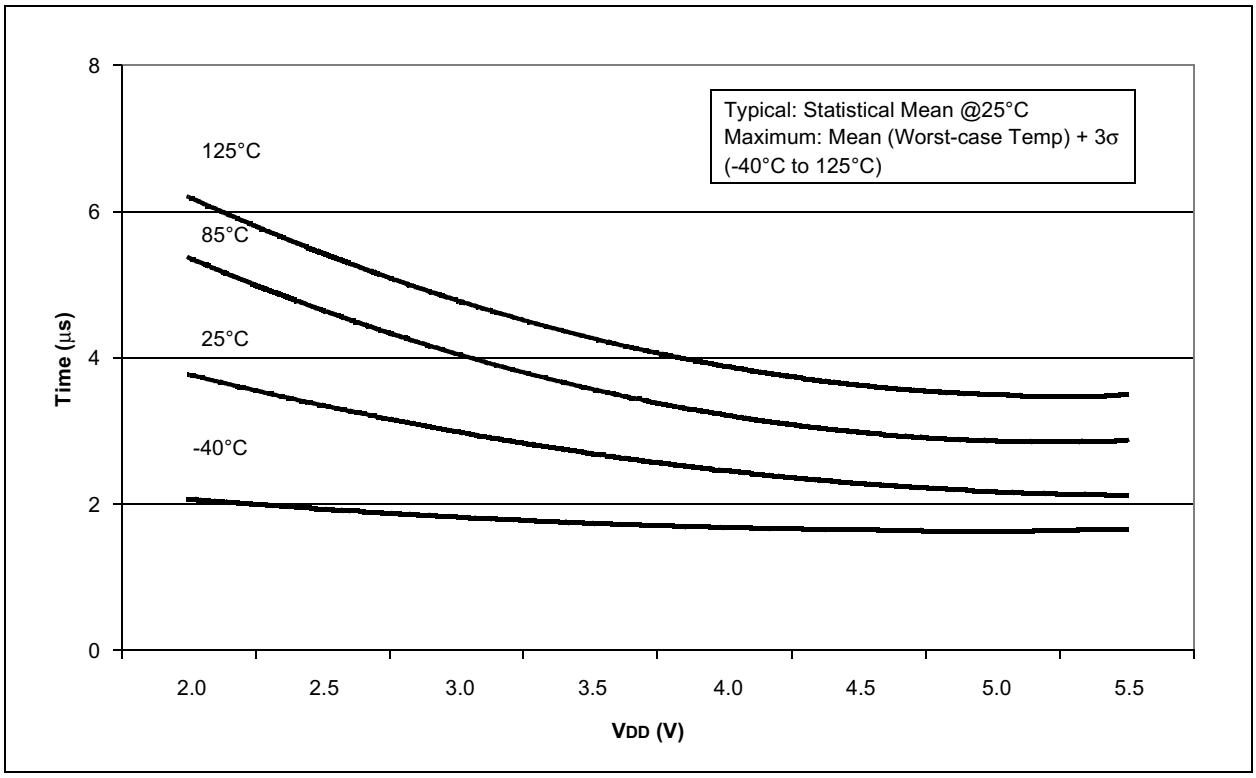


FIGURE 20-35: TYPICAL HFINTOSC START-UP TIMES vs. VDD OVER TEMPERATURE

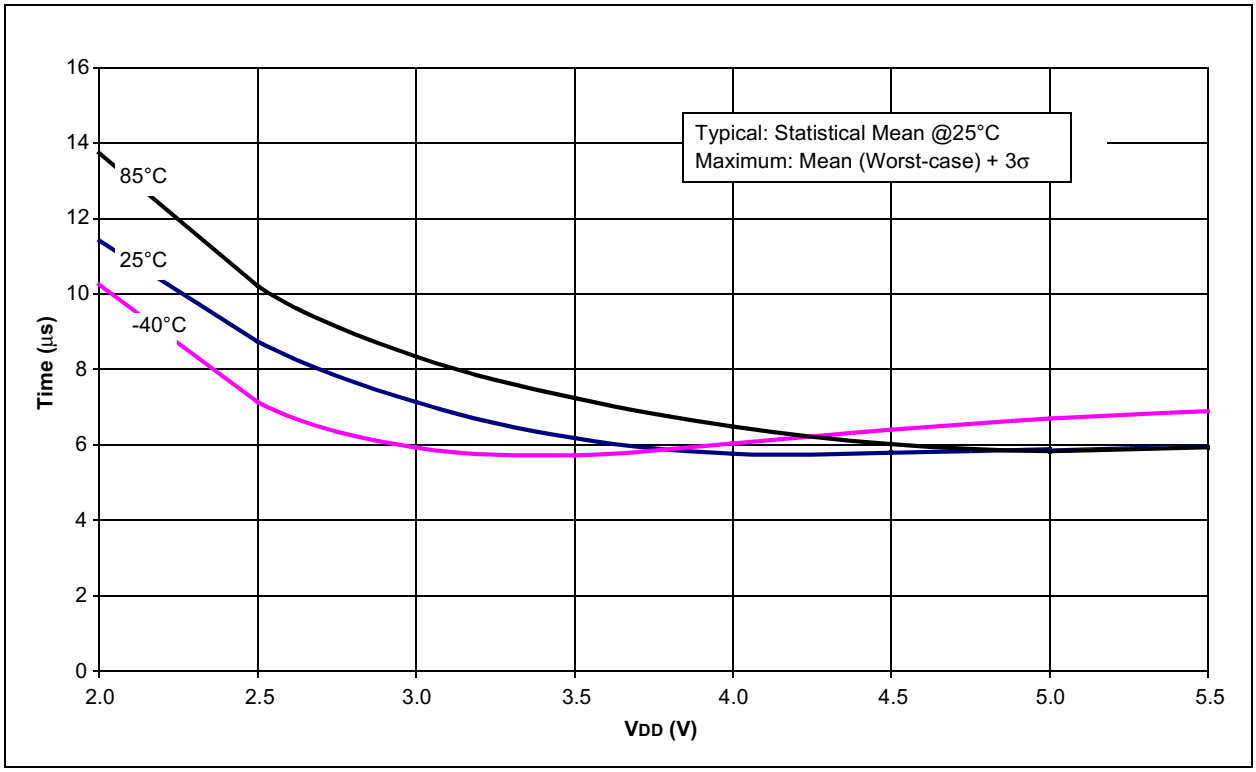


FIGURE 20-36: MAXIMUM HFINTOSC START-UP TIMES vs. VDD OVER TEMPERATURE

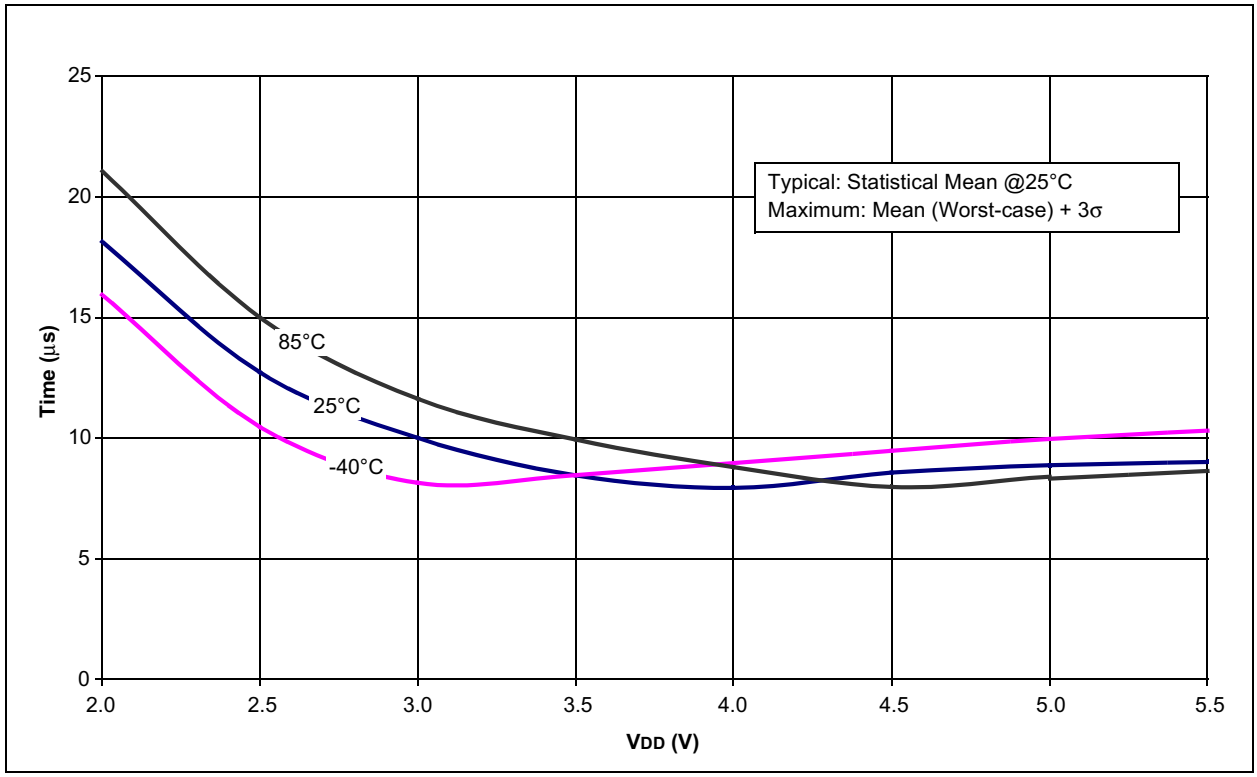


FIGURE 20-37: MINIMUM HFINTOSC START-UP TIMES vs. VDD OVER TEMPERATURE

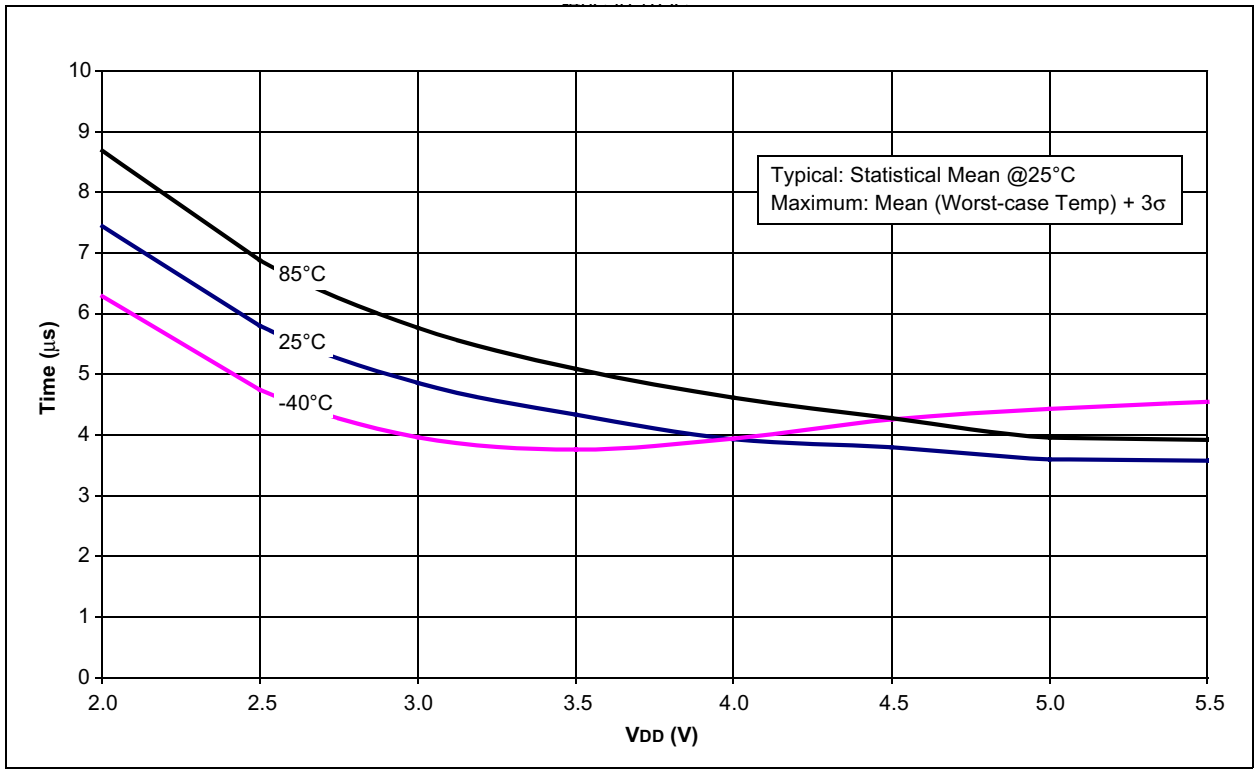


FIGURE 20-38: TYPICAL HFINTOSC FREQUENCY CHANGE vs. VDD (25°C)

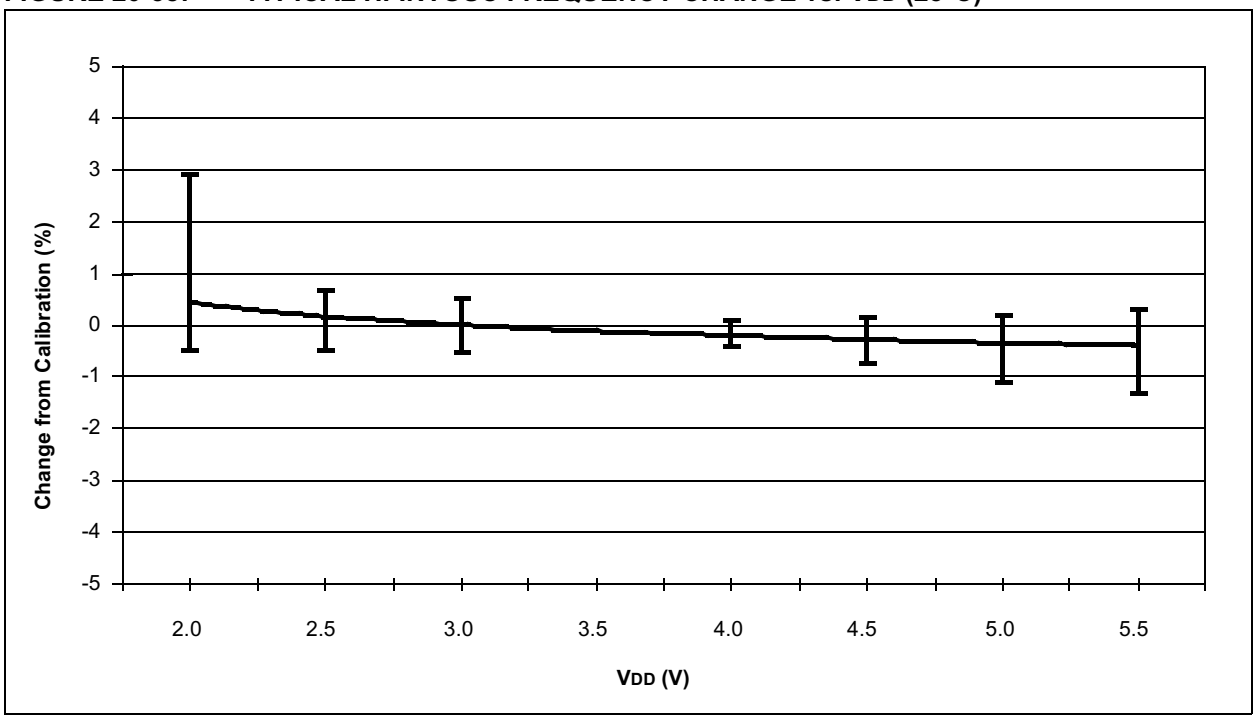


FIGURE 20-39: TYPICAL HFINTOSC FREQUENCY CHANGE OVER DEVICE VDD (85°C)

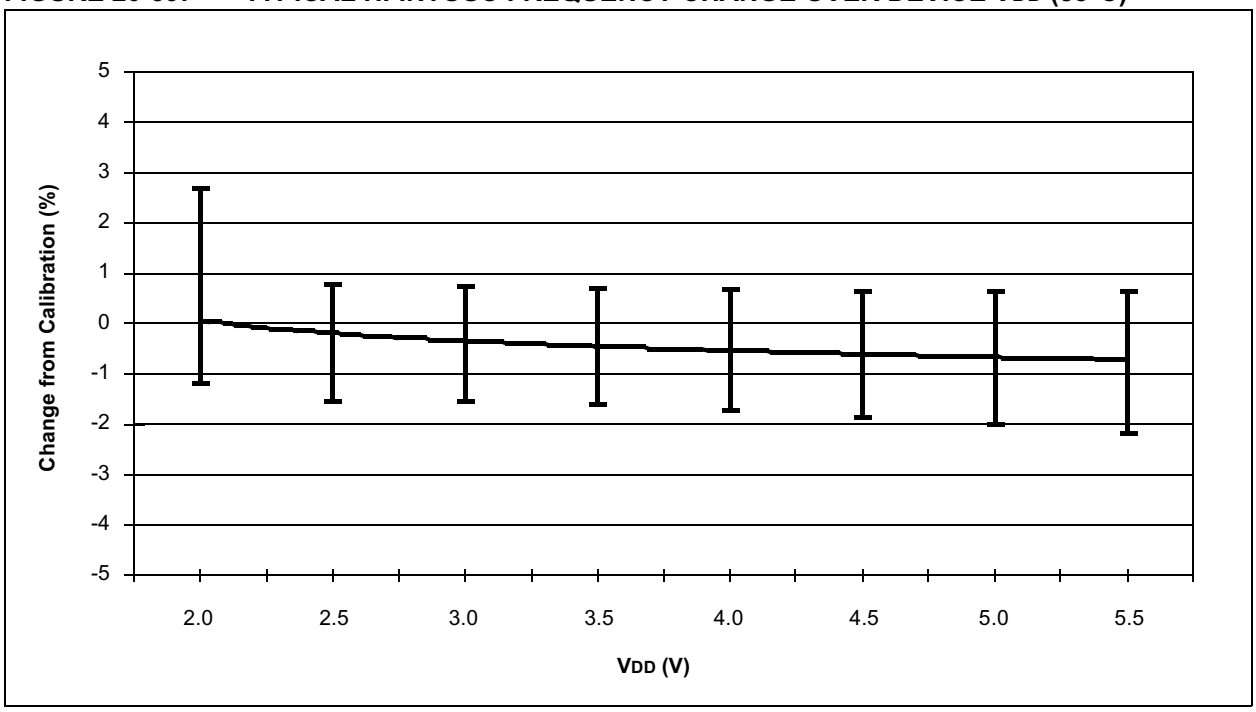


FIGURE 20-40: TYPICAL HFINTOSC FREQUENCY CHANGE vs. VDD (125°C)

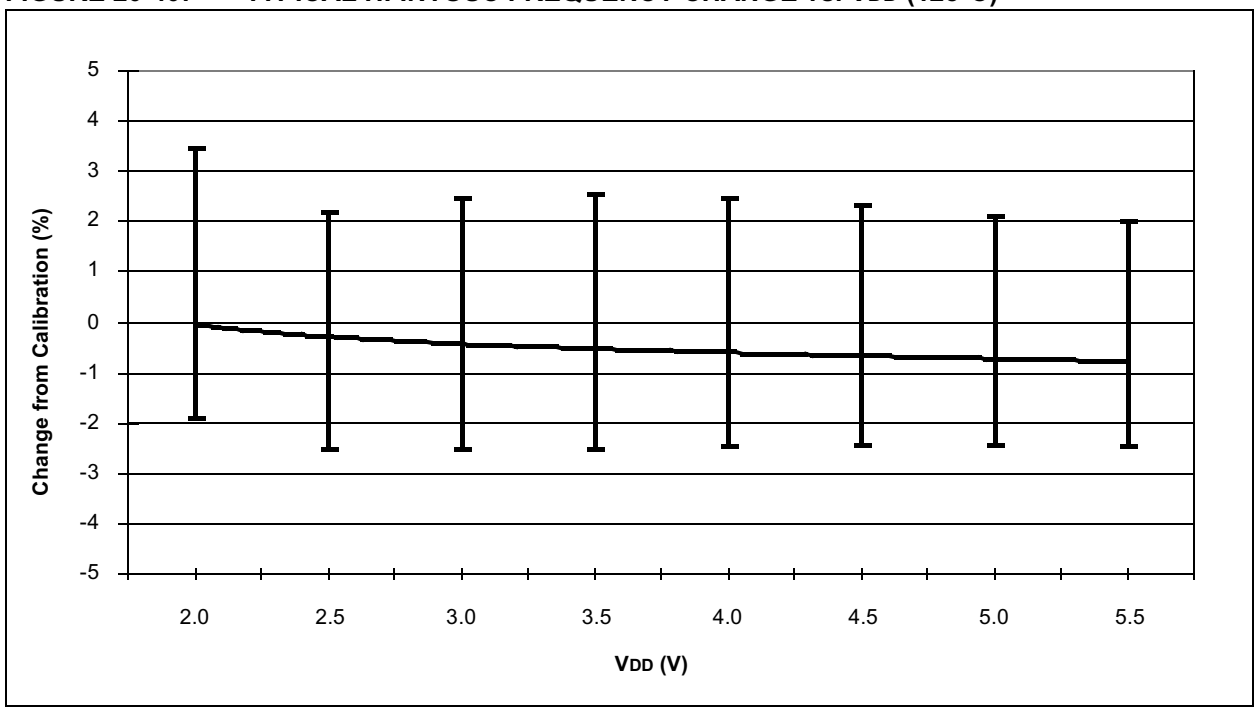
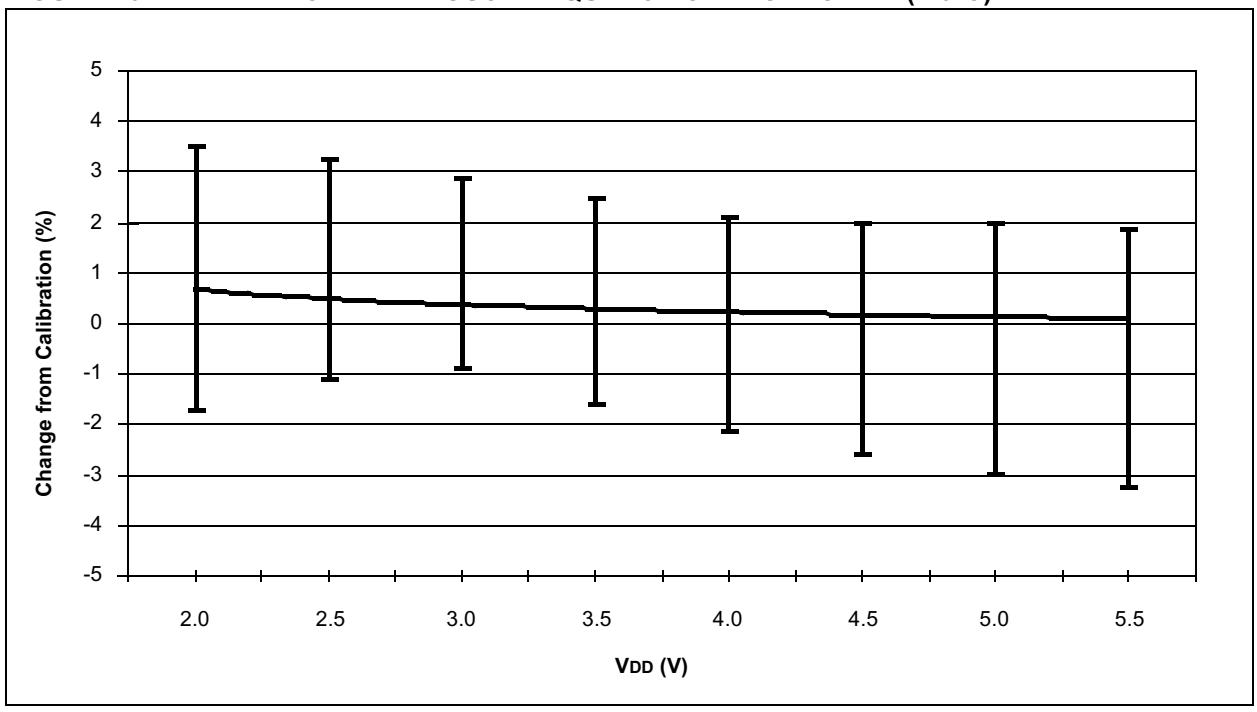


FIGURE 20-41: TYPICAL HFINTOSC FREQUENCY CHANGE vs. VDD (-40°C)

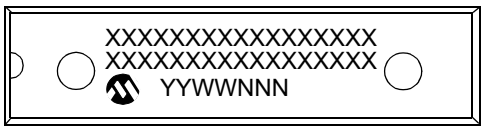


NOTES:

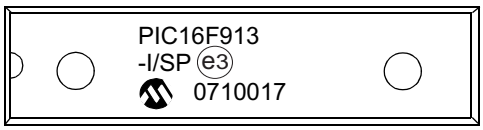
21.0 PACKAGING INFORMATION

21.1 Package Marking Information

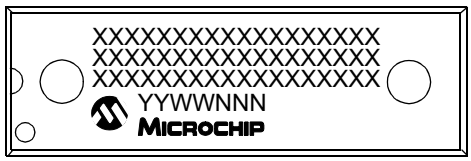
28-Lead SPDIP



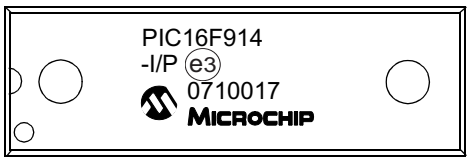
Example



40-Lead PDIP



Example



28-Lead QFN

Example

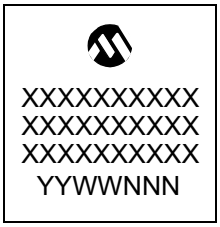
Legend:	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week ‘01’)
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator ((e3)) can be found on the outer packaging for this package.

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

* Standard PIC® device marking consists of Microchip part number, year code, week code and traceability code. For PIC® device marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

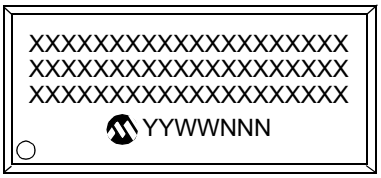
Package Marking Information (Continued)

44-Lead QFN

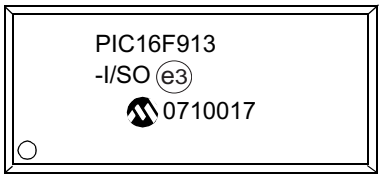


Example

28-Lead SOIC



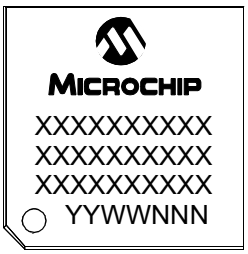
Example



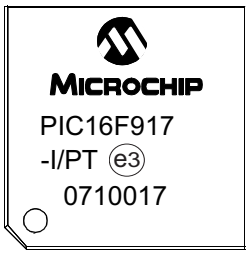
28-Lead SSOP

Example

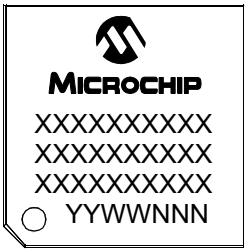
44-Lead TQFP



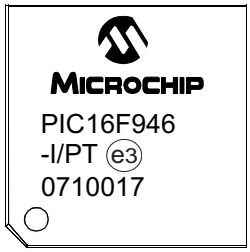
Example



64-Lead TQFP (10x10x1mm)



Example

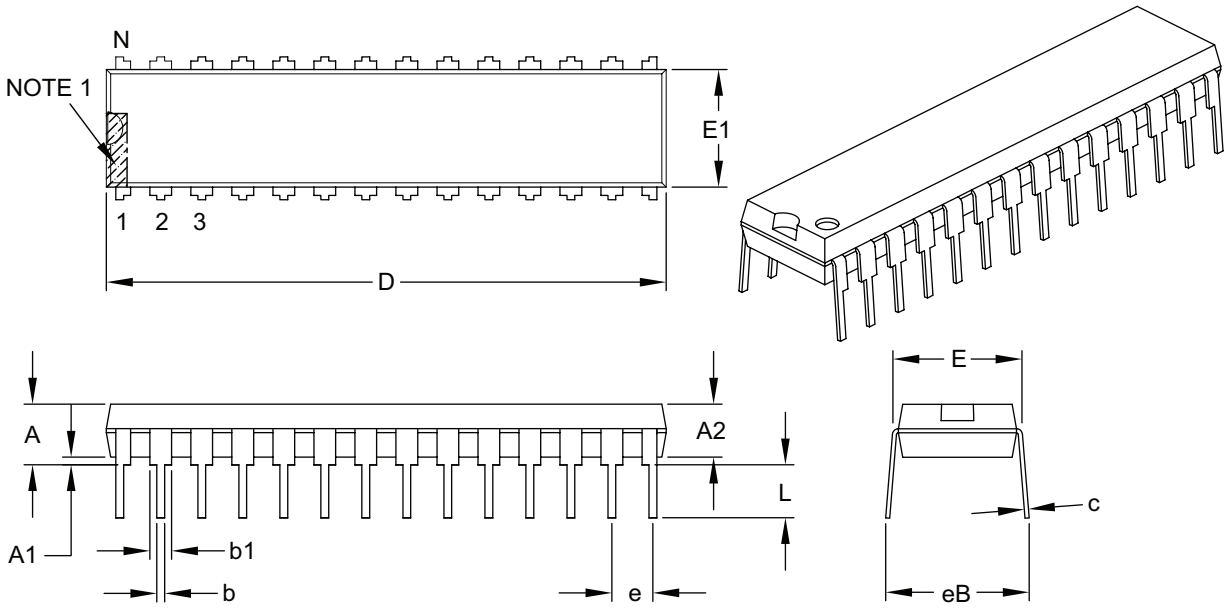


21.2 Package Details

The following sections give the technical details of the packages.

28-Lead Skinny Plastic Dual In-Line (SP) – 300 mil Body [SPDIP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		28	
Pitch	e		.100 BSC	
Top to Seating Plane	A	–	–	.200
Molded Package Thickness	A2	.120	.135	.150
Base to Seating Plane	A1	.015	–	–
Shoulder to Shoulder Width	E	.290	.310	.335
Molded Package Width	E1	.240	.285	.295
Overall Length	D	1.345	1.365	1.400
Tip to Seating Plane	L	.110	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.040	.050	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	–	–	.430

Notes:

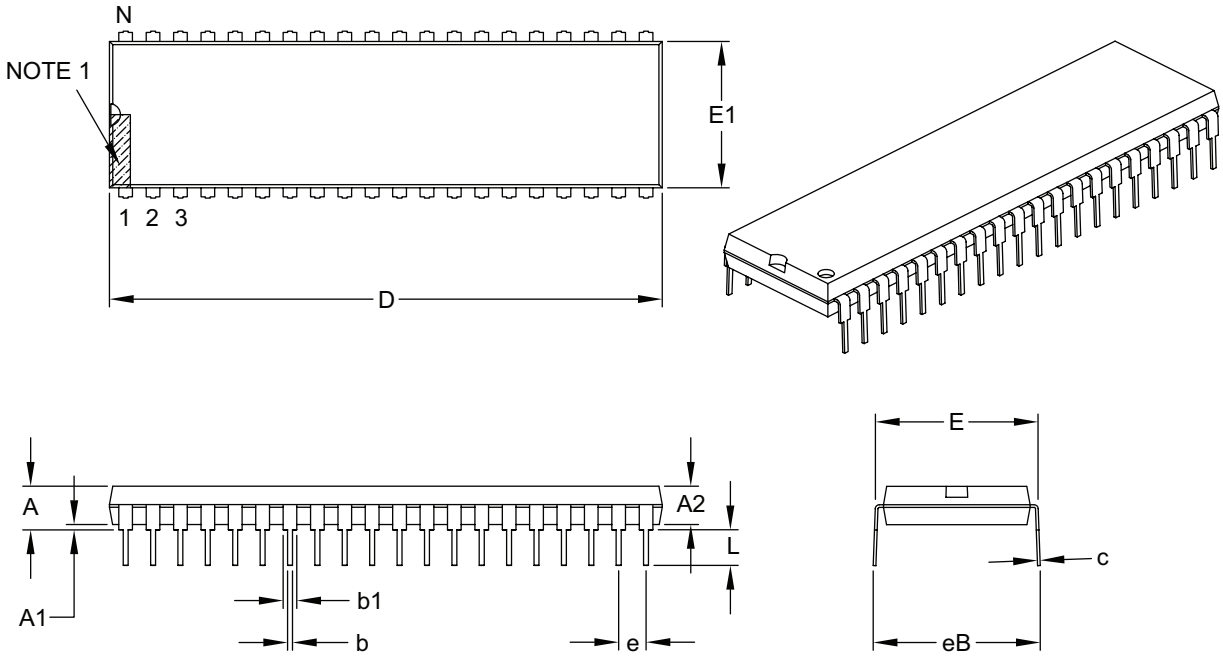
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-070B

40-Lead Plastic Dual In-Line (P) – 600 mil Body [PDIP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		40	
Pitch	e		.100 BSC	
Top to Seating Plane	A	–	–	.250
Molded Package Thickness	A2	.125	–	.195
Base to Seating Plane	A1	.015	–	–
Shoulder to Shoulder Width	E	.590	–	.625
Molded Package Width	E1	.485	–	.580
Overall Length	D	1.980	–	2.095
Tip to Seating Plane	L	.115	–	.200
Lead Thickness	c	.008	–	.015
Upper Lead Width	b1	.030	–	.070
Lower Lead Width	b	.014	–	.023
Overall Row Spacing §	eB	–	–	.700

Notes:

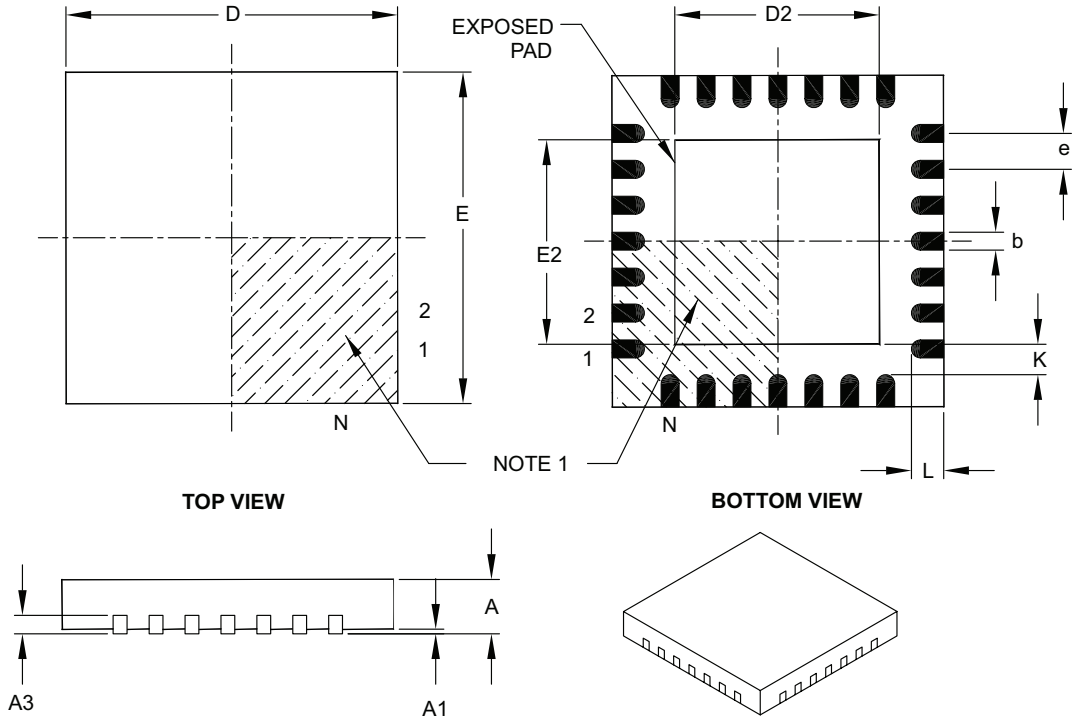
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-016B

28-Lead Plastic Quad Flat, No Lead Package (ML) – 6x6 mm Body [QFN] with 0.55 mm Contact Length

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 Pins	N		28	
Pitch	e		0.65 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3		0.20 REF	
Overall Width	E		6.00 BSC	
Exposed Pad Width	E2	3.65	3.70	4.20
Overall Length	D		6.00 BSC	
Exposed Pad Length	D2	3.65	3.70	4.20
Contact Width	b	0.23	0.30	0.35
Contact Length	L	0.50	0.55	0.70
Contact-to-Exposed Pad	K	0.20	–	–

Notes:

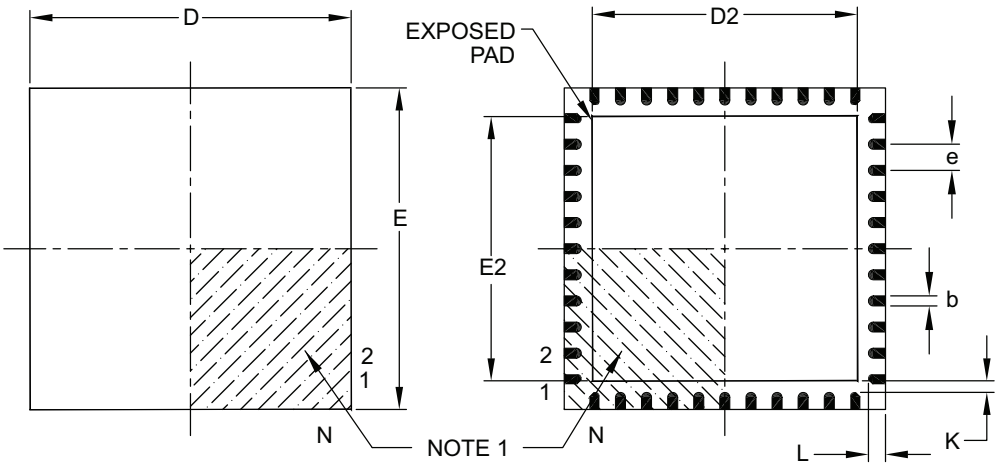
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

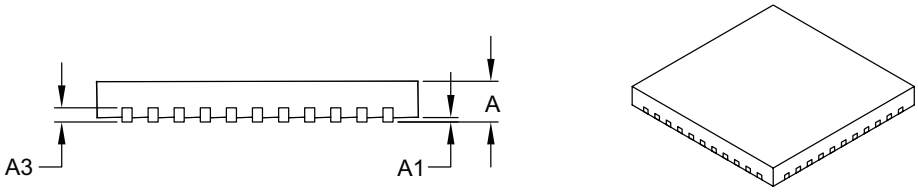
Microchip Technology Drawing C04-105B

44-Lead Plastic Quad Flat, No Lead Package (ML) – 8x8 mm Body [QFN]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



TOP VIEW **BOTTOM VIEW**



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		44	
Pitch	e		0.65 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3		0.20 REF	
Overall Width	E		8.00 BSC	
Exposed Pad Width	E2	6.30	6.45	6.80
Overall Length	D		8.00 BSC	
Exposed Pad Length	D2	6.30	6.45	6.80
Contact Width	b	0.25	0.30	0.38
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	–	–

Notes:

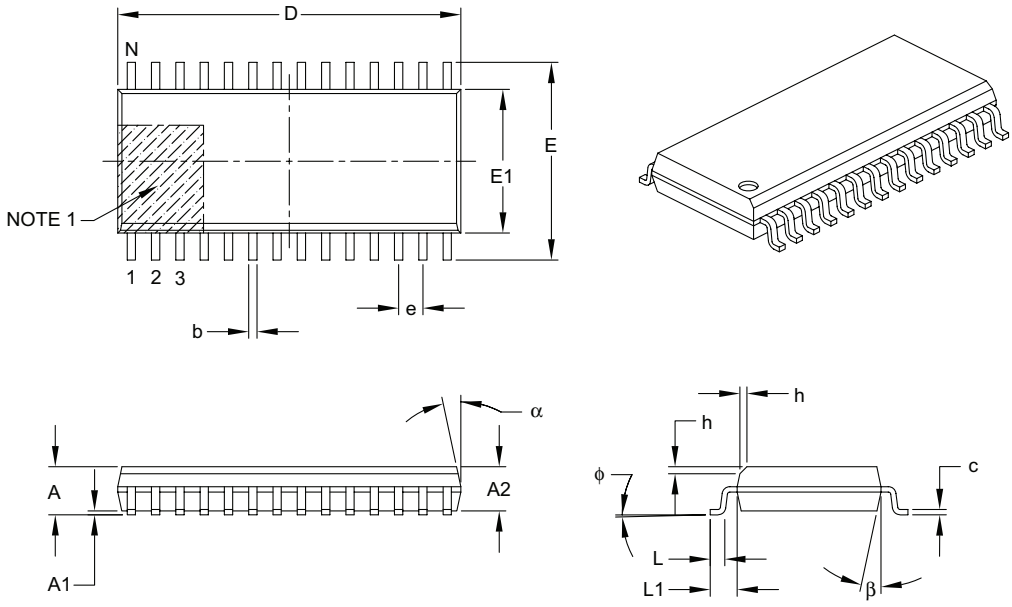
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated.
3. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-103B

28-Lead Plastic Small Outline (SO) – Wide, 7.50 mm Body [SOIC]

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 Pins	N		28	
Pitch	e		1.27 BSC	
Overall Height	A	–	–	2.65
Molded Package Thickness	A2	2.05	–	–
Standoff §	A1	0.10	–	0.30
Overall Width	E		10.30 BSC	
Molded Package Width	E1		7.50 BSC	
Overall Length	D		17.90 BSC	
Chamfer (optional)	h	0.25	–	0.75
Foot Length	L	0.40	–	1.27
Footprint	L1		1.40 REF	
Foot Angle Top	ϕ	0°	–	8°
Lead Thickness	c	0.18	–	0.33
Lead Width	b	0.31	–	0.51
Mold Draft Angle Top	α	5°	–	15°
Mold Draft Angle Bottom	β	5°	–	15°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

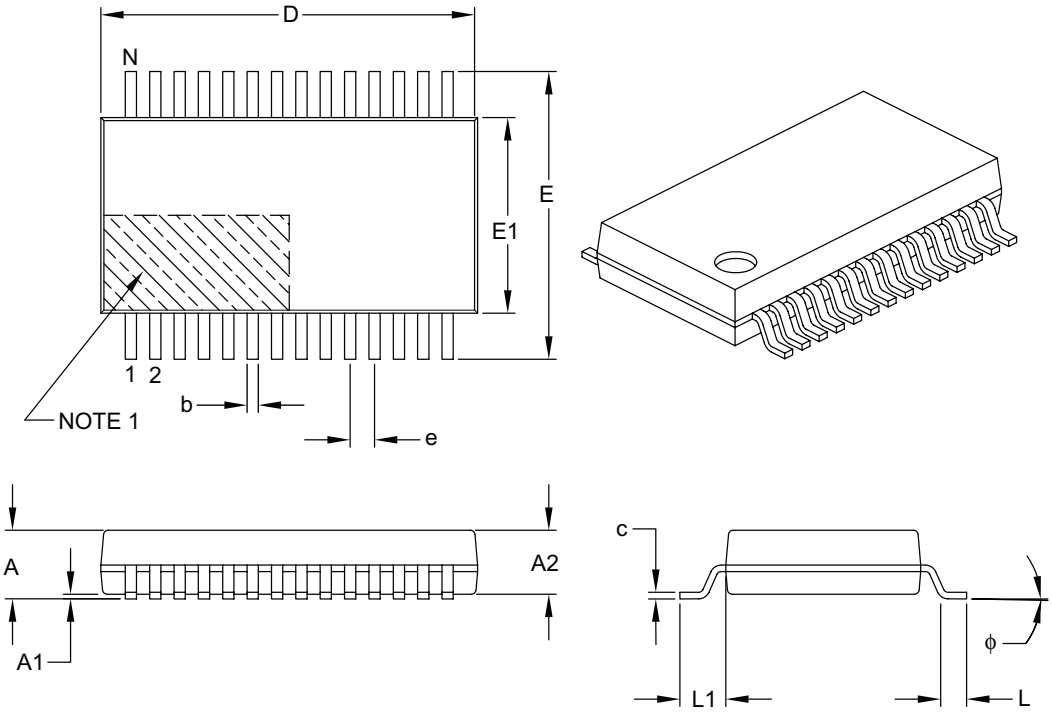
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-052B

28-Lead Plastic Shrink Small Outline (SS) – 5.30 mm Body [SSOP]

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 Pins	N		28	
Pitch	e		0.65 BSC	
Overall Height	A	–	–	2.00
Molded Package Thickness	A2	1.65	1.75	1.85
Standoff	A1	0.05	–	–
Overall Width	E	7.40	7.80	8.20
Molded Package Width	E1	5.00	5.30	5.60
Overall Length	D	9.90	10.20	10.50
Foot Length	L	0.55	0.75	0.95
Footprint	L1		1.25 REF	
Lead Thickness	c	0.09	–	0.25
Foot Angle	φ	0°	4°	8°
Lead Width	b	0.22	–	0.38

Notes:

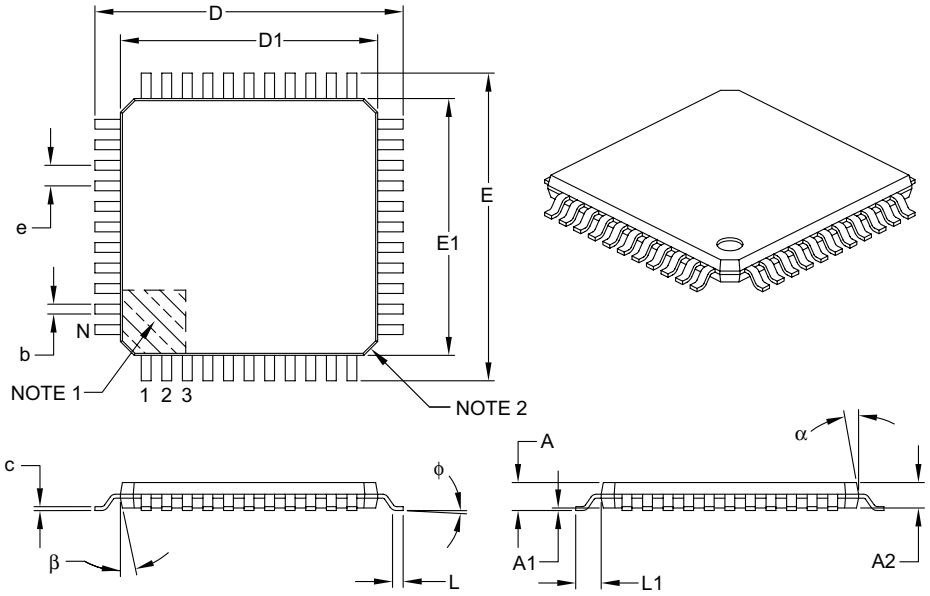
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.20 mm per side.
3. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-073B

44-Lead Plastic Thin Quad Flatpack (PT) – 10x10x1 mm Body, 2.00 mm Footprint [TQFP]

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 Leads	N		44	
Lead Pitch	e		0.80 BSC	
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	–	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1		1.00 REF	
Foot Angle	φ	0°	3.5°	7°
Overall Width	E		12.00 BSC	
Overall Length	D		12.00 BSC	
Molded Package Width	E1		10.00 BSC	
Molded Package Length	D1		10.00 BSC	
Lead Thickness	c	0.09	–	0.20
Lead Width	b	0.30	0.37	0.45
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

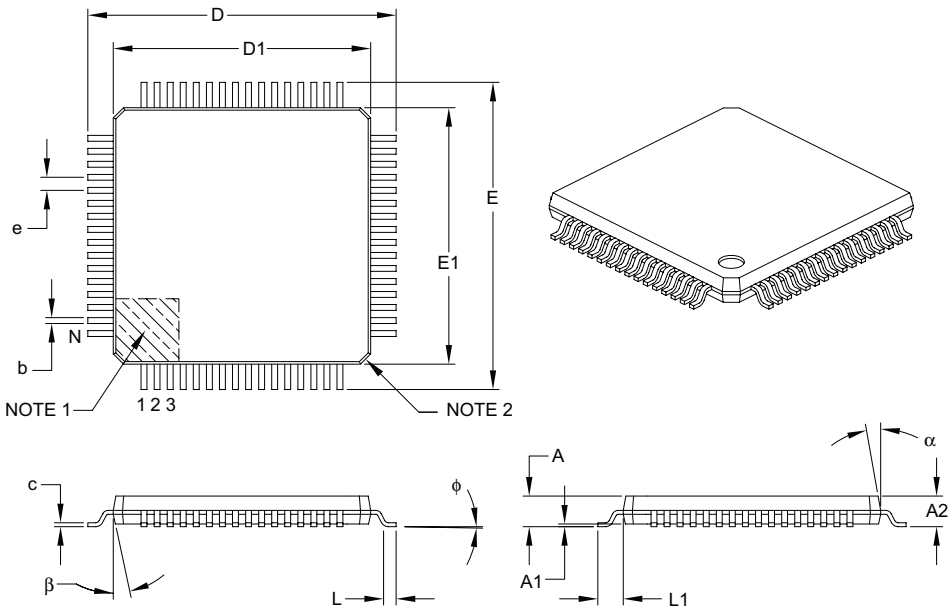
   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-076B

64-Lead Plastic Thin Quad Flatpack (PT) – 10x10x1 mm Body, 2.00 mm Footprint [TQFP]

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 Leads	N		64	
Lead Pitch	e		0.50 BSC	
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	–	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1		1.00 REF	
Foot Angle	φ	0°	3.5°	7°
Overall Width	E		12.00 BSC	
Overall Length	D		12.00 BSC	
Molded Package Width	E1		10.00 BSC	
Molded Package Length	D1		10.00 BSC	
Lead Thickness	c	0.09	–	0.20
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Chamfers at corners are optional; size may vary.
3. Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
4. 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 C04-085B

APPENDIX A: DATA SHEET REVISION HISTORY

Revision A

This is a new data sheet.

Revision B

Updated Peripheral Features.
Page 2, Table: Corrected I/O numbers.
Figure 8-3: Revised Comparator I/O operating modes.
Register 9-1, Table: Corrected max. number of pixels.

Revision C

Correction to Pin Description Table.
Correction to IPD base and T1OSC.

Revision D

Revised references 31.25 kHz to 31 kHz.
Revised Standby Current to 100 nA.
Revised 9.1: internal RC oscillator to internal LF oscillator.

Revision E

Removed “Advance Information” from Section 19.0 Electrical Specifications. Removed 28-Lead Plastic Quad Flat No Lead Package (ML) (QFN-S) package.

Revision F

Updates throughout document. Removed “Preliminary” from Data Sheet. Added Characterization Data chapter. Update Electrical Specifications chapter. Added PIC16F946 device.

APPENDIX B: MIGRATING FROM OTHER PIC® DEVICES

This discusses some of the issues in migrating from other PIC® devices to the PIC16F91X/946 family of devices.

B.1 PIC16F676 to PIC16F91X/946

TABLE B-1: FEATURE COMPARISON

Feature	PIC16F676	PIC16F91X/946
Max. Operating Speed	20 MHz	20 MHz
Max. Program Memory (Words)	1K	8K
Max. SRAM (Bytes)	64	352
A/D Resolution	10-bit	10-bit
Data EEPROM (bytes)	128	256
Timers (8/16-bit)	1/1	2/1
Oscillator Modes	8	8
Brown-out Reset	Y	Y
Internal Pull-ups	RB0/1/2/4/5	RB<7:0>
Interrupt-on-change	RB0/1/2/3/4/5	RB<7:4>
Comparator	1	2
USART	N	Y
Extended WDT	N	Y
Software Control Option of WDT/BOR	N	Y
INTOSC Frequencies	4 MHz	32 kHz - 8 MHz
Clock Switching	N	Y

APPENDIX C: CONVERSION CONSIDERATIONS

Considerations for converting from previous versions of devices to the ones listed in this data sheet are listed in Table C-1.

TABLE C-1: CONVERSION CONSIDERATIONS

Characteristic	PIC16F91X/946	PIC16F87X	PIC16F87XA
Pins	28/40/64	28/40	28/40
Timers	3	3	3
Interrupts	11 or 12	13 or 14	14 or 15
Communication	USART, SSP(1) (SPI, I²C™ Slave)	PSP, USART, SSP (SPI, I²C Master/Slave)	PSP, USART, SSP (SPI, I²C Master/Slave)
Frequency	20 MHz	20 MHz	20 MHz
Voltage	2.0V-5.5V	2.2V-5.5V	2.0V-5.5V
A/D	10-bit, 7 conversion clock selects	10-bit, 4 conversion clock selects	10-bit, 7 conversion clock selects
CCP	2	2	2
Comparator	2	—	2
Comparator Voltage Reference	Yes	—	Yes
Program Memory	4K, 8K Flash	4K, 8K Flash (Erase/Write on single-word)	4K, 8K Flash (Erase/Write on four-word blocks)
RAM	256, 336, 352 bytes	192, 368 bytes	192, 368 bytes
EEPROM Data	256 bytes	128, 256 bytes	128, 256 bytes
Code Protection	On/Off	Segmented, starting at end of program memory	On/Off
Program Memory Write Protection	—	On/Off	Segmented, starting at beginning of program memory
LCD Module	16, 24 segment drivers, 4 commons	—	—
Other	In-Circuit Debugger, Low-Voltage Programming	In-Circuit Debugger, Low-Voltage Programming	In-Circuit Debugger, Low-Voltage Programming

Note 1: SSP aand USART share the same pins on the PIC16F91X.

INDEX

A

B

U

V

W

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, 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
- Development Systems Information Line

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://support.microchip.com

READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

Please list the following information, and use this outline to provide us with your comments about this document.

To: Technical Publications Manager Total Pages Sent _____

RE: Reader Response

From: Name _____

Company _____

Address _____

City / State / ZIP / Country _____

Telephone: (_____) _____ - _____ FAX: (_____) _____ - _____

Application (optional):

Would you like a reply? ___Y ___N

Device: PIC16F913/914/916/917/946 Literature Number: DS41250F

Questions:

1. What are the best features of this document?

2. How does this document meet your hardware and software development needs?

3. Do you find the organization of this document easy to follow? If not, why?

4. What additions to the document do you think would enhance the structure and subject?

5. What deletions from the document could be made without affecting the overall usefulness?

6. Is there any incorrect or misleading information (what and where)?

7. How would you improve this document?

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	/XX	XXX
Device	Temperature Range	Package	Pattern

Device:	PIC16F913, PIC16F913T(1) PIC16F914, PIC16F914T(1) PIC16F916, PIC16F916T(1) PIC16F917, PIC16F917T(1) PIC16F946, PIC16F946T(1)	
Temperature Range:	I E	= -40°C to +85°C = -40°C to +125°C
Package:	ML P PT SO SP SS	= Micro Lead Frame (QFN) = Plastic DIP = TQFP (Thin Quad Flatpack) = SOIC = Skinny Plastic DIP = SSOP
Pattern:	3-Digit Pattern Code for QTP (blank otherwise)	

Examples:

a) PIC16F913-E/SP 301 = Extended Temp., skinny PDIP package, 20 MHz, QTP pattern #301

b) PIC16F913-I/SO = Industrial Temp., SOIC package, 20 MHz

Note 1: T = In tape and reel.

* JW Devices are UV erasable and can be programmed to any device configuration. JW Devices meet the electrical requirement of each oscillator type.

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://support.microchip.com
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Kokomo
Kokomo, IN
Tel: 765-864-8360
Fax: 765-864-8387

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
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: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Fuzhou
Tel: 86-591-8750-3506
Fax: 86-591-8750-3521

China - Hong Kong SAR
Tel: 852-2401-1200
Fax: 852-2401-3431

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: 86-755-8203-2660
Fax: 86-755-8203-1760

China - Shunde
Tel: 86-757-2839-5507
Fax: 86-757-2839-5571

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7250
Fax: 86-29-8833-7256

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-4182-8400
Fax: 91-80-4182-8422

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Yokohama
Tel: 81-45-471- 6166
Fax: 81-45-471-6122

Korea - Gumi
Tel: 82-54-473-4301
Fax: 82-54-473-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Penang
Tel: 60-4-646-8870
Fax: 60-4-646-5086

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: 886-3-572-9526
Fax: 886-3-572-6459

Taiwan - Kaohsiung
Tel: 886-7-536-4818
Fax: 886-7-536-4803

Taiwan - Taipei
Tel: 886-2-2500-6610
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 - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

12/08/06