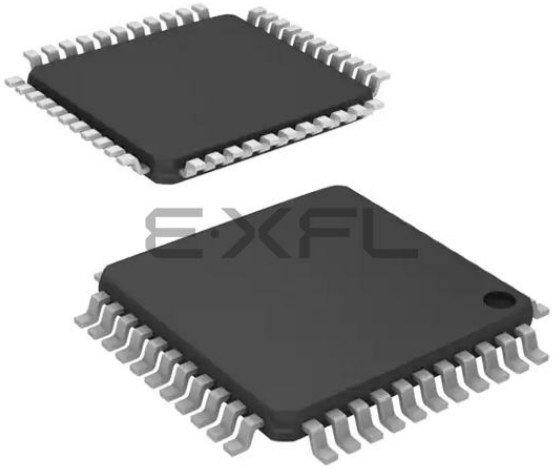


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	64MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, HLVD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 35x12b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f46k42t-i-pt

5.2 Register Definitions: Configuration Words

REGISTER 5-1: CONFIGURATION WORD 1L (30 0000h)

U-1	R/W-1	R/W-1	R/W-1	U-1	R/W-1	R/W-1	R/W-1
—	RSTOSC<2:0>			—	FEXTOSC<2:0>		
bit 7							bit 0

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

bit 7 **Unimplemented:** Read as '1'

bit 6-4 **RSTOSC<2:0>:** Power-up Default Value for COSC bits

111 = EXTOSC operating per FEXTOSC<2:0> bits
110 = HFINTOSC with HFFRQ = 4 MHz and CDIV = 4:1
101 = LFINTOSC
100 = SOSC
011 = Reserved
010 = EXTOSC with 4x PLL, with EXTOSC operating per FEXTOSC<2:0> bits
001 = Reserved
000 = HFINTOSC with HFFRQ = 64 MHz and CDIV = 1:1; resets COSC/NOSC to 3'b110

bit 3 **Unimplemented:** Read as '1'

bit 2-0 **FEXTOSC<2:0>:** FEXTOSC External Oscillator Mode Selection bits

111 = ECH (External Clock High Power)(1)
110 = ECM (External Clock Medium Power)(1)
101 = ECL (External Clock Low Power)(1)
100 = Oscillator is not enabled
011 = Reserved (do not use)
010 = HS (crystal oscillator) above 8 MHz
001 = XT (crystal oscillator) above 500 kHz, below 8 MHz
000 = LP (crystal oscillator) optimized for 32.768 kHz

Note 1: Refer to Table 44-9 for External Clock/Oscillator Timing Requirements.

5.7.4 FIXED VOLTAGE REFERENCE DATA

The DIA stores measured FVR voltages for this device in mV for the different buffer settings of 1x, 2x or 4x at Program Memory locations 3F0030h to 3F003Bh. For more information on the FVR, refer to **Section 34.0 "Fixed Voltage Reference (FVR)"**.

- FVRA1X stores the value of ADC FVR1 Output voltage for 1x setting (in mV)
- FVRA2X stores the value of ADC FVR1 Output Voltage for 2x setting (in mV)
- FVRA4X stores the value of ADC FVR1 Output Voltage for 4x setting (in mV)
- FVRC1X stores the value of Comparator FVR2 output voltage for 2x setting (in mV)
- FVRC2X stores the value of Comparator FVR2 output voltage for 2x setting (in mV)
- FVRC4X stores the value of Comparator FVR2 output voltage for 4x setting (in mV)

5.8 Device Configuration Information

The Device Configuration Information (DCI) is a dedicated region in the Program memory space mapped from 3FFF00h to 3FFF09h. The data stored in these locations is read-only and cannot be erased.

Refer to Table 5-4: Device Configuration Information for PIC18(L)F26/27/45/55/46/47/56/57K42 for the complete DCI table address and description. The DCI holds information about the device which is useful for programming and Bootloader applications.

The erase size is the minimum erasable unit in the PFM, expressed as rows. The total device Flash memory capacity is (Row Size * Number of rows)

TABLE 5-4: DEVICE CONFIGURATION INFORMATION FOR PIC18(L)F26/27/45/55/46/47/56/57K42

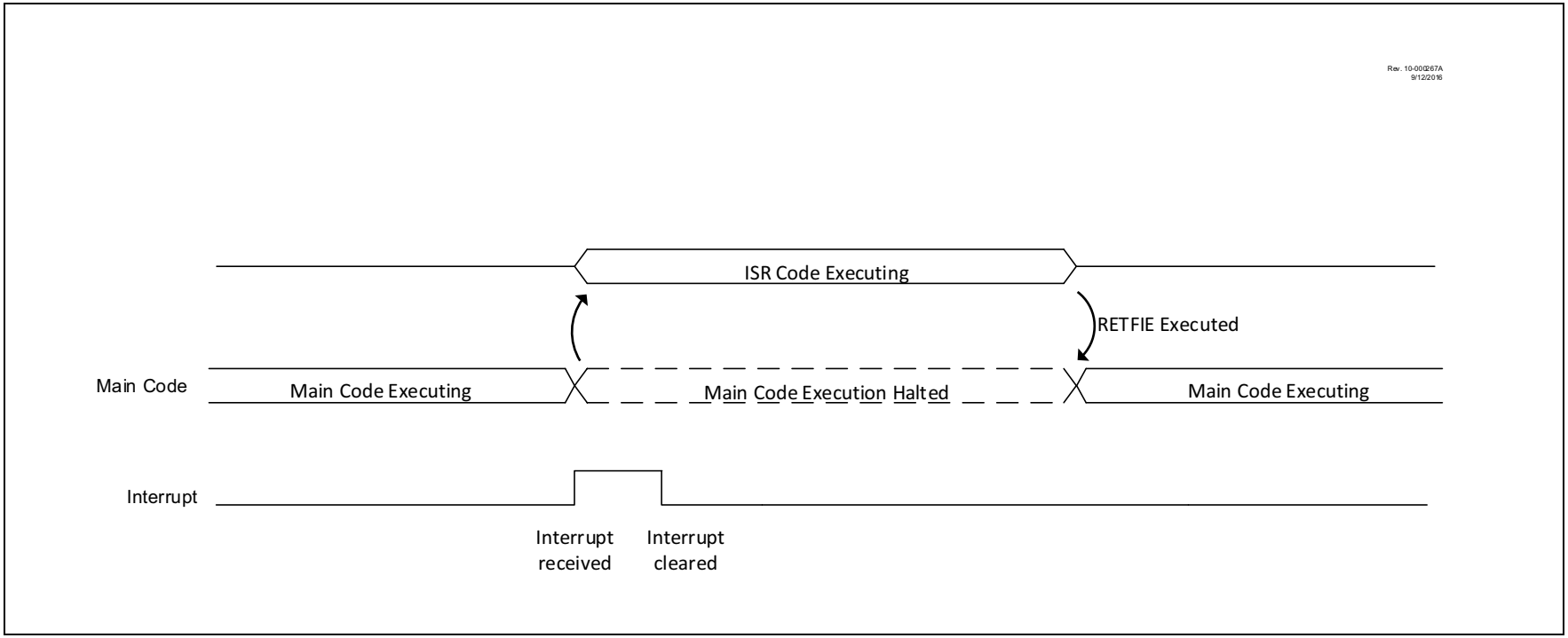
ADDRESS	Name	DESCRIPTION	VALUE			UNITS
			PIC18(L)F45/55K42	PIC18(L)F26/46/56K42	PIC18(L)F27/47/57K42	
3F FF00h-3F FF01h	ERSIZ	Erase Row Size	64	64	64	Words
3F FF02h-3F FF03h	WLSIZ	Number of write latches per row	128	128	128	Bytes
3F FF04h-3F FF05h	URSIZ	Number of User Rows	256	512	1024	Rows
3F FF06h-3F FF07h	EESIZ	Data EEPROM memory size	256	1024	1024	Bytes
3F FF08h-3F FF09h	PCNT	Pin Count	40(1)/48	28/40(1)/48	28/40(1)/48	Pins

Note 1: Pin count of 40 is also used for 44-pin part.

9.4.1 SERVING A HIGH OR LOW PRIORITY INTERRUPT WHEN MAIN ROUTINE CODE IS EXECUTING

When a high or low priority interrupt is requested when the main routine code is executing, the main routine execution is halted and the ISR is addressed, see Figure 9-2. Upon a return from the ISR (by executing the RETFIE instruction), the main routine resumes execution.

FIGURE 9-2: INTERRUPT EXECUTION: HIGH/LOW PRIORITY INTERRUPT WHEN EXECUTING MAIN ROUTINE



REGISTER 9-9: PIR6: PERIPHERAL INTERRUPT REGISTER 6(1)

R/W/HS-0/0	R/W/HS-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0
TMR3GIF	TMR3IF	U2IF(2)	U2EIF(3)	U2TXIF(4)	U2RXIF(4)	I2C2EIF(5)	I2C2IF(6)
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	HS = Bit is set in hardware

bit 7 **TMR3GIF:** TMR3 Gate Interrupt Flag bit
1 = Interrupt has occurred (must be cleared by software)
0 = Interrupt event has not occurred

bit 6 **TMR3IF:** TMR3 Interrupt Flag bit
1 = Interrupt has occurred (must be cleared by software)
0 = Interrupt event has not occurred

bit 5 **U2IF:** UART2 Interrupt Flag bit(2)
1 = Interrupt has occurred
0 = Interrupt event has not occurred

bit 4 **U2EIF:** UART2 Framing Error Interrupt Flag bit(3)
1 = Interrupt has occurred
0 = Interrupt event has not occurred

bit 3 **U2TXIF:** UART2 Transmit Interrupt Flag bit(4)
1 = Interrupt has occurred
0 = Interrupt event has not occurred

bit 2 **U2RXIF:** UART2 Receive Interrupt Flag bit(4)
1 = Interrupt has occurred
0 = Interrupt event has not occurred

bit 1 **I2C2EIF:** I^2C2 Error Interrupt Flag bit(5)
1 = Interrupt has occurred
0 = Interrupt event has not occurred

bit 0 **I2C2IF:** I^2C2 Interrupt Flag bit(6)
1 = Interrupt has occurred
0 = Interrupt event has not occurred

Note 1: Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit, or the global enable bit. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.
2: UxIF is a read-only bit. To clear the interrupt condition, all bits in the UxUIR register must be cleared.
3: UxEIF is a read-only bit. To clear the interrupt condition, all bits in the UxERRIR register must be cleared.
4: UxTXIF and UxRXIF are read-only bits and cannot be set/cleared by the software.
5: I2CxEIF is a read-only bit. To clear the interrupt condition, all bits in the I2CxERR register must be cleared.
6: I2CxIF is a read-only bit. To clear the interrupt condition, all bits in the I2CxPIR register must be cleared.

13.1.1 TABLE READS AND TABLE WRITES

In order to read and write program memory, there are two operations that allow the processor to move bytes between the program memory space and the data RAM:

- Table Read (TBLRD)
- Table Write (TBLWT)

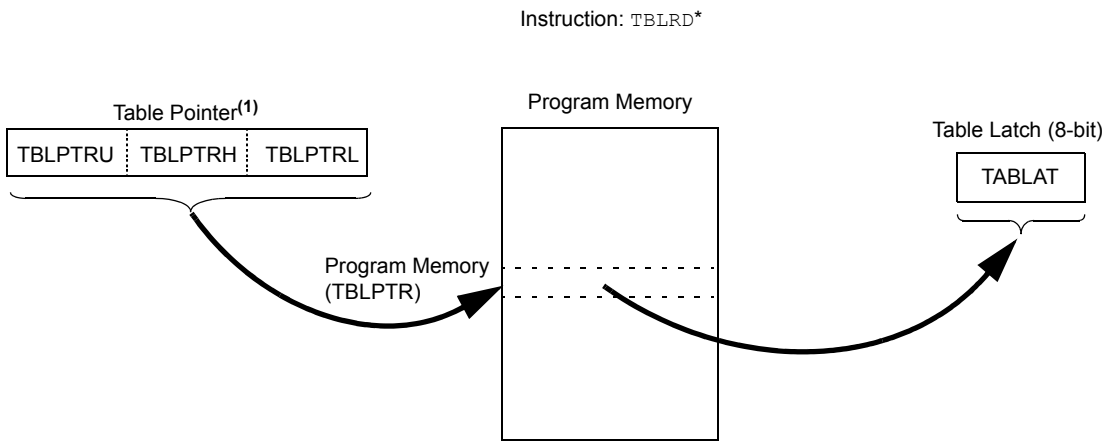
The program memory space is 16 bits wide, while the data RAM space is eight bits wide. Table reads and table writes move data between these two memory spaces through an 8-bit register (TABLAT).

The table read operation retrieves one byte of data directly from program memory and places it into the TABLAT register. Figure 13-1 shows the operation of a table read.

The table write operation stores one byte of data from the TABLAT register into a write block holding register. The procedure to write the contents of the holding registers into program memory is detailed in **Section 13.1.6 "Writing to Program Flash Memory"**. Figure 13-2 shows the operation of a table write with program memory.

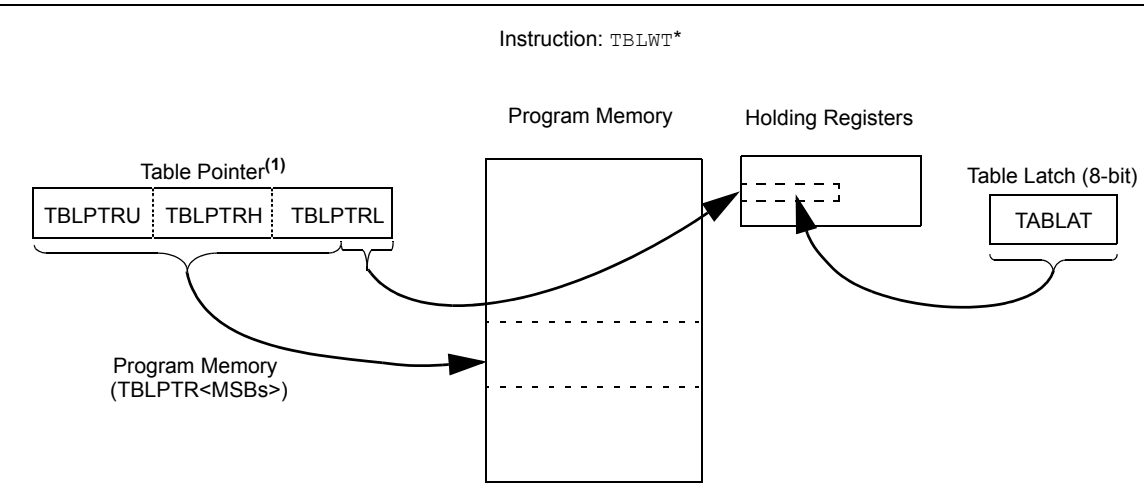
Table operations work with byte entities. Tables containing data, rather than program instructions, are not required to be word aligned. Therefore, a table can start and end at any byte address. If a table write is being used to write executable code into program memory, program instructions will need to be word aligned.

FIGURE 13-1: TABLE READ OPERATION



Note 1: Table Pointer register points to a byte in program memory.

FIGURE 13-2: TABLE WRITE OPERATION

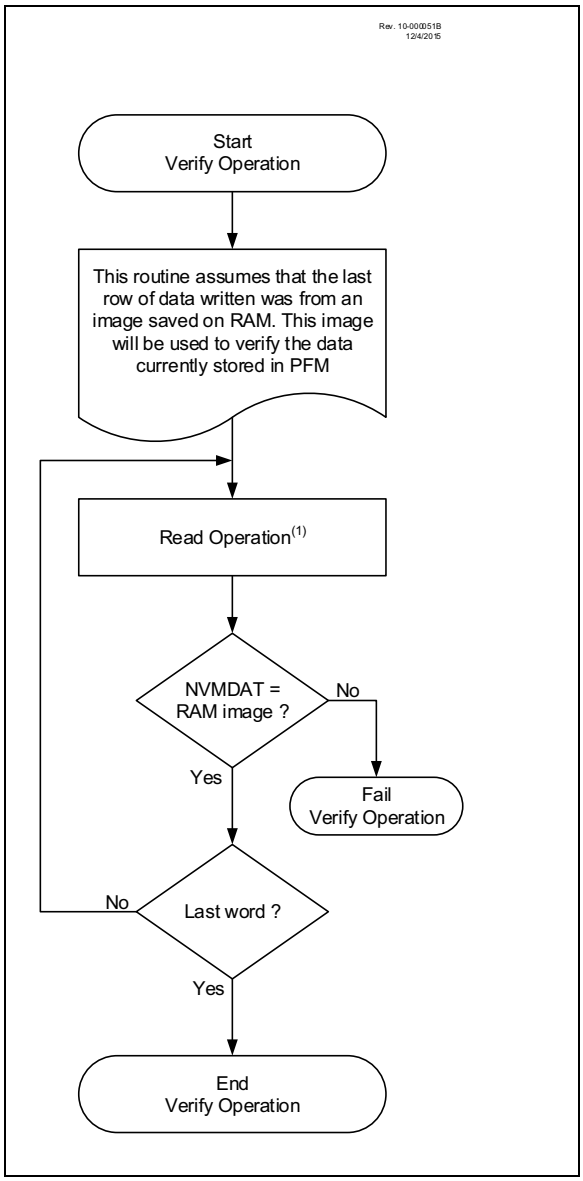


Note 1: During table writes the Table Pointer does not point directly to program memory. The LSBs of TBLPRTL actually point to an address within the write block holding registers. The MSBs of the Table Pointer determine where the write block will eventually be written. The process for writing the holding registers to the program memory array is discussed in **Section 13.1.6 "Writing to Program Flash Memory"**.

13.1.6.2 Write Verify

Depending on the application, good programming practice may dictate that the value written to the memory should be verified against the original value. This should be used in applications where excessive writes can stress bits near the specification limit. Since program memory is stored as a full page, the stored program memory contents are compared with the intended data stored in RAM after the last write is complete.

FIGURE 13-10: PROGRAM FLASH MEMORY VERIFY FLOWCHART



13.1.6.3 Unexpected Termination of Write Operation

If a write is terminated by an unplanned event, such as loss of power or an unexpected Reset, the memory location just programmed should be verified and reprogrammed if needed. If the write operation is interrupted by a MCLR Reset or a WDT Time-out Reset during normal operation, the WRERR bit will be set which the user can check to decide whether a rewrite of the location(s) is needed.

13.1.6.4 Protection Against Spurious Writes

A write sequence is valid only when both the following conditions are met, this prevents spurious writes which might lead to data corruption.

1. The WR bit is gated through the WREN bit. It is suggested to have the WREN bit cleared at all times except during memory writes. This prevents memory writes if the WR bit gets set accidentally.
2. The NVM unlock sequence must be performed each time before a write operation.

13.2 Device Information Area, Device Configuration Area, User ID, Device ID and Configuration Word Access

When REG<1:0> = 0b01 or 0b11 in the NVMCON1 register, the Device Information Area, the Device Configuration Area, the User IDs, Device ID/ Revision ID and Configuration Words can be accessed. Different access may exist for reads and writes (see Table 13-1).

13.2.1 Reading Access

The user can read from these blocks by setting the REG bits to 0b01 or 0b11. The user needs to load the address into the TBLPTR registers. Executing a TBLRD after that moves the byte pointed to the TABLAT register. The CPU operation is suspended during the read and resumes after. When read access is initiated on an address outside the parameters listed in Table 13-1, the TABLAT register is cleared, reading back ‘0’s.

13.2.2 Writing Access

The WREN bit in NVMCON1 must be set to enable writes. This prevents accidental writes to the CONFIG words due to errant (unexpected) code execution. The WREN bit should be kept clear at all times, except when updating the CONFIG words. The WREN bit is not cleared by hardware. The WR bit will be inhibited from being set unless the WREN bit is set.

The user needs to load the TBLPTR and TABLAT register with the address and data byte respectively before executing the write command. An unlock sequence needs to be followed for writing to the USER IDs/ DEVICE IDs/CONFIG words (Section 13.1.4, NVM Unlock Sequence). If WRTC = 0 or if TBLPTR points an invalid address location (see Table 13-1), WR bit is cleared without any effect and WRERR is set.

A single CONFIG word byte is written at once and the operation includes an implicit erase cycle for that byte (it is not necessary to set FREE). CPU execution is stalled and at the completion of the write cycle, the WR bit is cleared in hardware and the NVM Interrupt Flag bit (NVMIF) is set. The new CONFIG value takes effect when the CPU resumes operation.

TABLE 13-3: DIA, DCI, USER ID, DEV/REV ID AND CONFIGURATION WORD ACCESS (REG<1:0> = x1)

Address	Function	Read Access	Write Access
20 0000h-20 000Fh	User IDs	Yes	Yes
30 0000h-30 0009h	Configuration Words	Yes	Yes
3F 0000h-3F 003Fh	DIA	Yes	No
3F FF00h-3F FF09h	DCI	Yes	No
3F FFFCh-3F FFFFh	Revision ID/Device ID	Yes	No

18.0 INTERRUPT-ON-CHANGE

PORTA, PORTB, PORTC and pin RE3 of PORTE can be configured to operate as Interrupt-on-Change (IOC) pins on PIC18(L)F26/27/45/46/47/55/56/57K42 family devices. An interrupt can be generated by detecting a signal that has either a rising edge or a falling edge. Any individual port pin, or combination of port pins, can be configured to generate an interrupt. The interrupt-on-change module has the following features:

- Interrupt-on-Change enable (Master Switch)
- Individual pin configuration
- Rising and falling edge detection
- Individual pin interrupt flags

Figure 18-1 is a block diagram of the IOC module.

18.1 Enabling the Module

To allow individual port pins to generate an interrupt, the IOCIE bit of the PIEx register must be set. If the IOCIE bit is disabled, the edge detection on the pin will still occur, but an interrupt will not be generated.

18.2 Individual Pin Configuration

For each port pin, a rising edge detector and a falling edge detector are present. To enable a pin to detect a rising edge, the associated bit of the IOCxP register is set. To enable a pin to detect a falling edge, the associated bit of the IOCxN register is set.

A pin can be configured to detect rising and falling edges simultaneously by setting both associated bits of the IOCxP and IOCxN registers, respectively.

18.3 Interrupt Flags

The IOCAFx, IOCBFx, IOCCFx and IOCEF3 bits located in the IOCAF, IOCBF, IOCCF and IOCEF registers respectively, are status flags that correspond to the interrupt-on-change pins of the associated port. If an expected edge is detected on an appropriately enabled pin, then the status flag for that pin will be set, and an interrupt will be generated if the IOCIE bit is set. The IOCIF bit of the PIR0 register reflects the status of all IOCAFx, IOCBFx, IOCCFx and IOCEF3 bits.

18.4 Clearing Interrupt Flags

The individual status flags, (IOCAFx, IOCBFx, IOCCFx and IOCEF3 bits), can be cleared by resetting them to zero. If another edge is detected during this clearing operation, the associated status flag will be set at the end of the sequence, regardless of the value actually being written.

In order to ensure that no detected edge is lost while clearing flags, only AND operations masking out known changed bits should be performed. The following sequence is an example of what should be performed.

EXAMPLE 18-1: CLEARING INTERRUPT FLAGS (PORTA EXAMPLE)

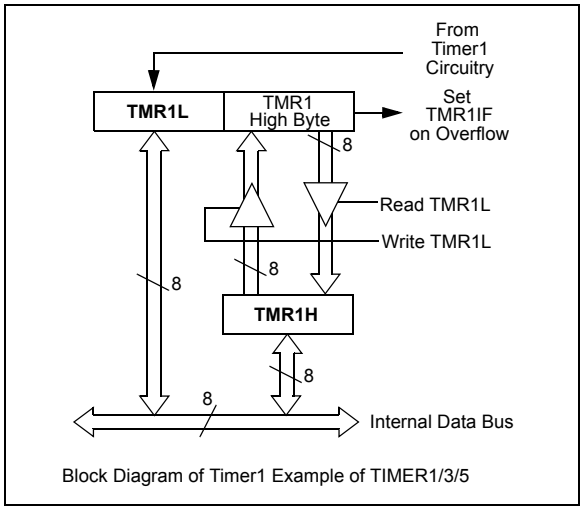
```
MOVLW   0xff
XORWF   IOCAF, W
ANDWF   IOCAF, F
```

18.5 Operation in Sleep

The interrupt-on-change interrupt sequence will wake the device from Sleep mode, if the IOCIE bit is set.

If an edge is detected while in Sleep mode, the IOCxF register will be updated prior to the first instruction executed out of Sleep.

FIGURE 21-2: TIMER1/3/5 16-BIT READ/ WRITE MODE BLOCK DIAGRAM



Block Diagram of Timer1 Example of TIMER1/3/5

21.6 Timer1/3/5 Gate

Timer1/3/5 can be configured to count freely or the count can be enabled and disabled using Timer1/3/5 gate circuitry. This is also referred to as Timer1/3/5 gate enable.

Timer1/3/5 gate can also be driven by multiple selectable sources.

21.6.1 TIMER1/3/5 GATE ENABLE

The Timer1/3/5 Gate Enable mode is enabled by setting the TMRxGE bit of the TxGCON register. The polarity of the Timer1/3/5 Gate Enable mode is configured using the TxGPOL bit of the TxGCON register.

When Timer1/3/5 Gate Enable mode is enabled, Timer1/3/5 will increment on the rising edge of the Timer1/3/5 clock source. When Timer1/3/5 Gate signal is inactive, the timer will not increment and hold the current count. See Figure 21-4 for timing details.

TABLE 21-2: TIMER1/3/5 GATE ENABLE SELECTIONS

TMRxCLK	TxGPOL	TxG	Timer1/3/5 Operation
↑	1	1	Counts
↑	1	0	Holds Count
↑	0	1	Holds Count
↑	0	0	Counts

23.2.3 SOFTWARE INTERRUPT MODE

When the Capture mode is changed, a false capture interrupt may be generated. The user should keep the CCPxIE Interrupt Priority bit of the respective PIE register clear to avoid false interrupts. Additionally, the user should clear the CCPxIF interrupt flag bit of the respective PIR register following any change in Operating mode.

23.2.4 CAPTURE DURING SLEEP

Capture mode depends upon the Timer1 module for proper operation. There are two options for driving the Timer1 module in Capture mode. It can be driven by the instruction clock (FOSC/4), or by an external clock source.

When Timer1 is clocked by FOSC/4, Timer1 will not increment during Sleep. When the device wakes from Sleep, Timer1 will continue from its previous state.

Capture mode will operate during Sleep as long as the clock source for Timer1 is active in Sleep.

23.3 Compare Mode

Compare mode makes use of the 16-bit Timer1 resource. The 16-bit value of the CCPRxH:CCPRxL register pair is constantly compared against the 16-bit value of the TMRxH:TMRxL register pair. When a match occurs, one of the following events can occur:

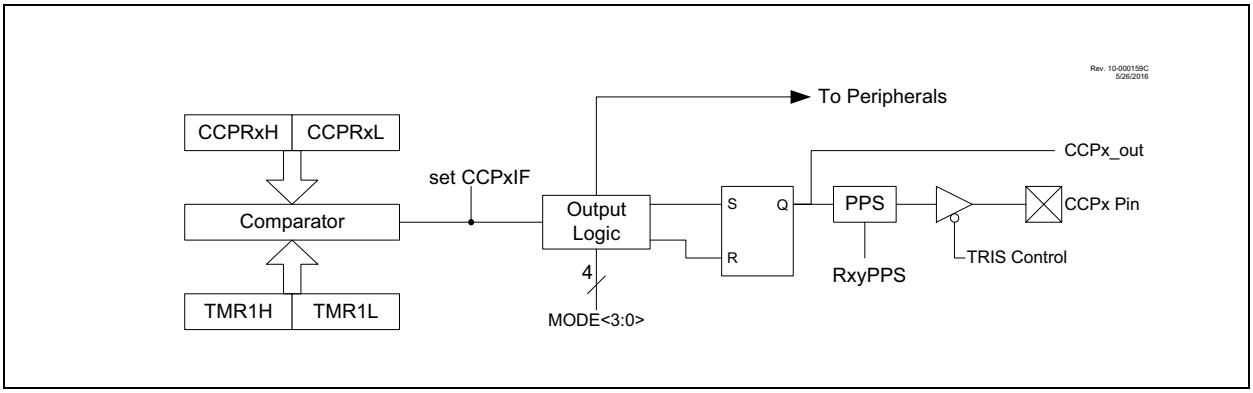
- Toggle the CCPx output, clear TMRx
- Toggle the CCPx output
- Set the CCPx output
- Clear the CCPx output
- Pulse output
- Pulse output, clear TMRx

The action on the pin is based on the value of the MODE<3:0> control bits of the CCPxCON register. At the same time, the interrupt flag CCPxIF bit is set, and an ADC conversion can be triggered, if selected.

All Compare modes can generate an interrupt and trigger an ADC conversion. When MODE = `0b0001` or `0b1011`, the CCP resets the TMR register pair.

Figure 23-2 shows a simplified diagram of the compare operation.

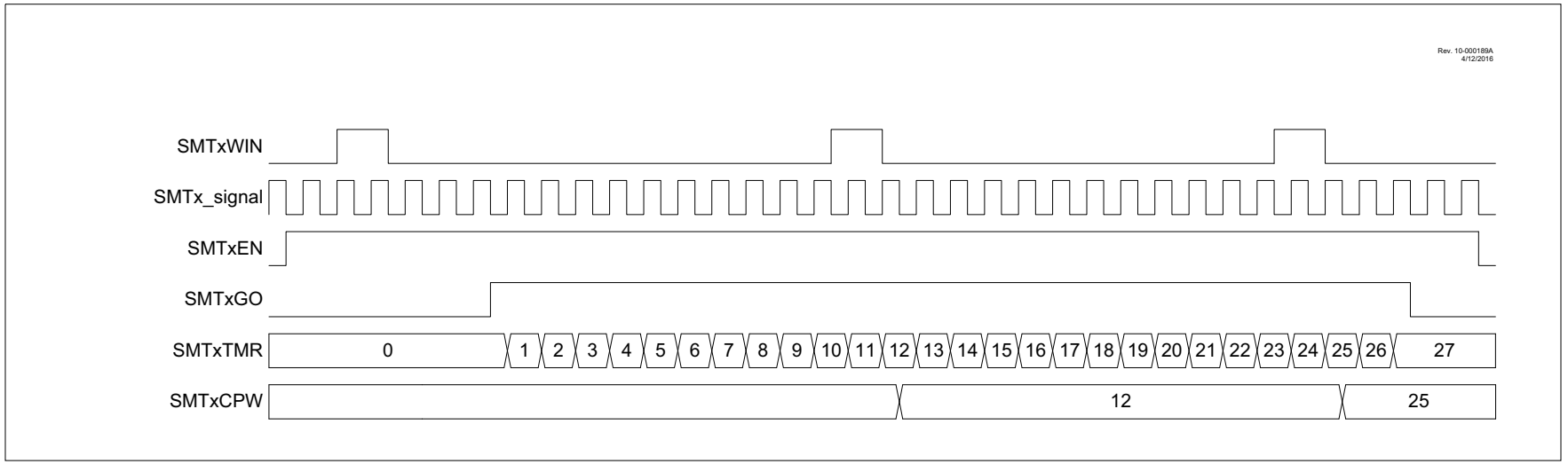
FIGURE 23-2: COMPARE MODE OPERATION BLOCK DIAGRAM



25.6.9 COUNTER MODE

This mode increments the timer on each pulse of the SMT1_signal input. This mode is asynchronous to the SMT clock and uses the SMT1_signal as a time source. The SMT1CPW register will be updated with the current SMT1TMR value on the rising edge of the SMT1WIN input. See Figure 25-18.

FIGURE 25-18: COUNTER MODE TIMING DIAGRAM



REGISTER 25-6: SMT1SIG: SMT1 SIGNAL INPUT SELECT REGISTER

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	SSEL<4:0>				
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	q = Value depends on condition

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

bit 4-0 **SSEL<4:0>:** SMT1 Signal Selection bits

11111 = Reserved
•
•
•
11010 = Reserved
11001 = CLC4_out
11000 = CLC3_out
10111 = CLC2_out
10110 = CLC1_out
10101 = ZCD1_out
10100 = CMP2_out
10011 = CMP1_out
10010 = NCO1_out
10001 = Reserved
10000 = Reserved
01111 = PWM8_out
01110 = PWM7_out
01101 = PWM6_out
01100 = PWM5_out
01011 = CCP4_out
01010 = CCP3_out
01001 = CCP2_out
01000 = CCP1_out
00111 = TMR6_postscaled
00110 = TMR5_postscaled
00101 = TMR4_postscaled
00100 = TMR3_postscaled
00011 = TMR2_postscaled
00010 = TMR1_postscaled
00001 = TMR0_overflow
00000 = SMTxSIGPPS

REGISTER 27-7: CLCxGLS0: GATE 0 LOGIC SELECT REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
G1D4T	G1D4N	G1D3T	G1D3N	G1D2T	G1D2N	G1D1T	G1D1N
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **G1D4T:** Gate 0 Data 4 True (non-inverted) bit
1 = CLCIN3 (true) is gated into CLCx Gate 0
0 = CLCIN3 (true) is not gated into CLCx Gate 0

bit 6 **G1D4N:** Gate 0 Data 4 Negated (inverted) bit
1 = CLCIN3 (inverted) is gated into CLCx Gate 0
0 = CLCIN3 (inverted) is not gated into CLCx Gate 0

bit 5 **G1D3T:** Gate 0 Data 3 True (non-inverted) bit
1 = CLCIN2 (true) is gated into CLCx Gate 0
0 = CLCIN2 (true) is not gated into CLCx Gate 0

bit 4 **G1D3N:** Gate 0 Data 3 Negated (inverted) bit
1 = CLCIN2 (inverted) is gated into CLCx Gate 0
0 = CLCIN2 (inverted) is not gated into CLCx Gate 0

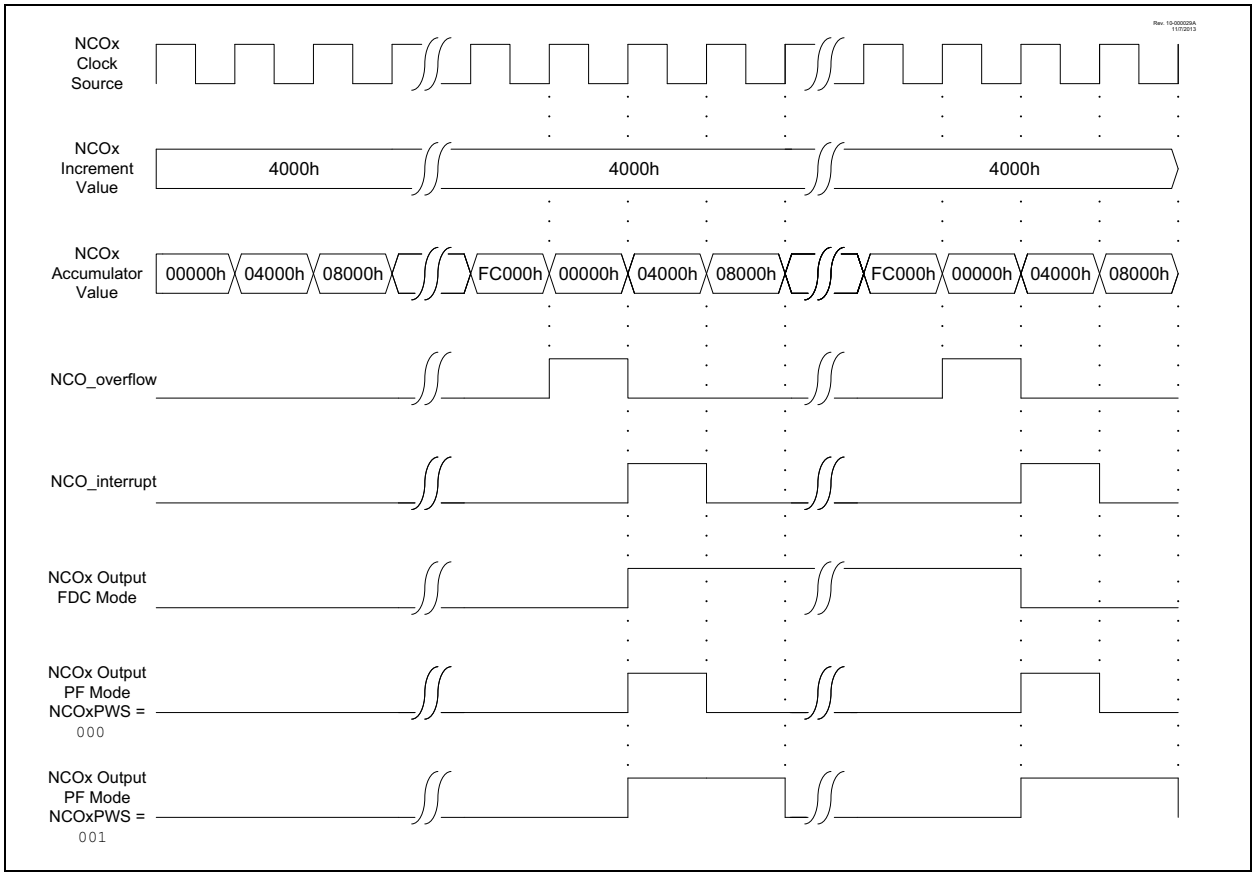
bit 3 **G1D2T:** Gate 0 Data 2 True (non-inverted) bit
1 = CLCIN1 (true) is gated into CLCx Gate 0
0 = CLCIN1 (true) is not gated into l CLCx Gate 0

bit 2 **G1D2N:** Gate 0 Data 2 Negated (inverted) bit
1 = CLCIN1 (inverted) is gated into CLCx Gate 0
0 = CLCIN1 (inverted) is not gated into CLCx Gate 0

bit 1 **G1D1T:** Gate 0 Data 1 True (non-inverted) bit
1 = CLCIN0 (true) is gated into CLCx Gate 0
0 = CLCIN0 (true) is not gated into CLCx Gate 0

bit 0 **G1D1N:** Gate 0 Data 1 Negated (inverted) bit
1 = CLCIN0 (inverted) is gated into CLCx Gate 0
0 = CLCIN0 (inverted) is not gated into CLCx Gate 0

FIGURE 28-2: FDC OUTPUT MODE OPERATION DIAGRAM



REGISTER 31-3: UxCON2: UART CONTROL REGISTER 2

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
RUNOVF	RXPOL	STP<1:0>		C0EN	TXPOL	FLO<1:0>	
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **RUNOVF:** Run During Overflow Control bit
1 = RX input shifter continues to synchronize with Start bits after overflow condition
0 = RX input shifter stops all activity on receiver overflow condition

bit 6 **RXPOL**: Receive Polarity Control bit
1 = Invert RX polarity, Idle state is low
0 = RX polarity is not inverted, Idle state is high

bit 5-4 **STP<1:0>:** Stop Bit Mode Control bits(1)
11 = Transmit 2 Stop bits, receiver verifies first Stop bit
10 = Transmit 2 Stop bits, receiver verifies first and second Stop bits
01 = Transmit 1.5 Stop bits, receiver verifies first Stop bit
00 = Transmit 1 Stop bit, receiver verifies first Stop bit

bit 3 **C0EN**: Checksum Mode Select bit(2)
LIN mode:
1 = Checksum Mode 1, enhanced LIN checksum includes PID in sum
0 = Checksum Mode 0, legacy LIN checksum does not include PID in sum
Other modes:
1 = Add all TX and RX characters
0 = Checksums disabled

bit 2 **TXPOL:** Transmit Polarity Control bit
1 = Output data is inverted, TX output is low in Idle state
0 = Output data is not inverted, TX output is high in Idle state

bit 1-0 **FLO<1:0>:** Handshake Flow Control bits
11 = Reserved
10 = RTS/CTS and TXDE Hardware flow control
01 = XON/XOFF Software flow control
00 = Flow control is off

Note 1: All modes transmit selected number of Stop bits. Only DMX and DALI receivers verify selected number of Stop bits and all others verify only the first Stop bit.
2: UART1 only.

REGISTER 31-10: UxRXB: UART RECEIVE REGISTER

R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0
RXB<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **RXB<7:0>:** Top of Receive Buffer

REGISTER 31-11: UxTXB: UART TRANSMIT REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
TXB<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **TXB<7:0>:** Bottom of Transmit Buffer

REGISTER 32-10: SPIxSTATUS: SPI STATUS REGISTER

R/C/HS-0/0	U-0	R-1/1	U-0	R/C/HS-0/0	S-0/0	U-0	R-0/0
TXWE	—	TXBE	—	RXRE	CLRBF	—	RXBF
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0' C = Clearable bit S = Settable bit HS = Bit can be set by hardware

bit 7 **TXWE**: Transmit Buffer Write Error bit

1 = SPIxTxB was written while TxFIFO was full

0 = No error has occurred

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

bit 5 **TXBE**: Transmit Buffer Empty bit (read-only)

1 = Transmit buffer TxFIFO is empty

0 = Transmit buffer is not empty

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

bit 3 **RXRE**: Receive Buffer Read Error bit

1 = SPIxRB was read while RxFIFO was empty

0 = No error has occurred

bit 2 **CLRBF**: Clear Buffer Control bit (write only)

1 = Reset the receive and transmit buffers, making both buffers empty

0 = Take no action

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

bit 0 **RXBF**: Receive Buffer Full bit (read-only)

1 = Receive buffer is full

0 = Receive buffer is not full

REGISTER 32-11: SPIxRxB: SPI READ BUFFER REGISTER

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
RXB7	RXB6	RXB5	RXB4	RXB3	RXB2	RXB1	RXB0
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'

bit 7-0 **RXB<7:0>**: Receiver Buffer bits (read-only)

If RX buffer is not empty:

Contains the top-most byte of RXFIFO, and reading this register will remove the top-most byte RXFIFO and decrease the occupancy of the RXFIFO

If RX buffer is empty:

Reading this register will read as '0', leave the occupancy unchanged, and set the RXRE bit of SPIxSTATUS

REGISTER 33-6: I2CxSTAT0: I²C STATUS REGISTER 0

R-0	R-0	R-0	R-0	R-0	U-0	U-0	U-0
BFRE(3)	SMA	MMA	R(1, 2)	D	—	—	—
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	HS = Hardware set HC = Hardware clear

bit 7 **BFRE:** Bus Free Status bit(3)
- 1 = Indicates the I²C bus is idle
  Both SCL and SDA have been high for time-out selected by I2CCON2<BFRET<1:0>> bits.
  I2CCLK must select a valid clock source for this bit to function.
- 0 = Bus not idle (When no I2CCLK is selected, this bit remains clear)

bit 6 **SMA:** Slave Module Active Status bit
- 1 = Set after the 8th falling SCL edge of a received matching 7-bit slave address
  Set after the 8th falling SCL edge of a received matching 10-bit slave **low** address
  Set after the 8th falling SCL edge of a received matching 10-bit slave **high** w/ read address, only after a previous matching high and low w/ write.
- 0 = Cleared by any Restart/Stop detected on the bus
  Cleared by BTOIF and BCLIF conditions

bit 5 **MMA:** Master Module Active Status bit
- 1 = Master Mode state machine is active
  Set when master state machine asserts a Start on bus
- 0 = Master state machine is idle
  Cleared when BCLIF is set
  Cleared when Stop is shifted out by master.
  Cleared for BTOIF condition, after the master successfully shifts out a Stop condition.

bit 4 **R:** Read Information bit (1, 2)
- 1 = Indicates the last matching received (high) address was a Read request
- 0 = Indicates the last matching received (high) address was a Write

bit 3 **D:** Data bit
- 1 = Indicates the last byte received or transmitted was data
- 0 = Indicates the last byte received or transmitted was an address

bit 2-0 **Unimplemented:** Read as 1'b0

Note 1: This bit holds the R bit information following the last received address match. Addresses transmitted by the Master or appearing on the bus without a match do not affect this bit.

2: Clock requests and input from I2CxCLK register are disabled in Slave modes.

3: Software must use the EN bit to force Master or Slave hardware to idle.

38.3 Comparator Hysteresis

A selectable amount of separation voltage can be added to the input pins of each comparator to provide a hysteresis function to the overall operation. Hysteresis is enabled by setting the HYS bit of the CMxCON0 register.

See Comparator Specifications in Table 44-17 for more information.

38.3.1 COMPARATOR OUTPUT SYNCHRONIZATION

The output from a comparator can be synchronized with Timer1 by setting the SYNC bit of the CMxCON0 register.

Once enabled, the comparator output is latched on the falling edge of the Timer1 source clock. If a prescaler is used, the CxOUT bit is synchronized with the timer, so that the software sees no ambiguity due to timing. See the Comparator Block Diagram (Figure 38-2) and the Timer1 Block Diagram (Figure 21-1) for more information.

38.4 Comparator Interrupt

An interrupt can be generated for every rising or falling edge of the comparator output.

When either edge detector is triggered and its associated enable bit is set (INTP and/or INTN bits of the CMxCON1 register), the Corresponding Interrupt Flag bit (CxIF bit of the respective PIR register) will be set.

To enable the interrupt, you must set the following bits:

- EN bit of the CMxCON0 register
- CxIE bit of the respective PIE register
- INTP bit of the CMxCON1 register (for a rising edge detection)
- INTN bit of the CMxCON1 register (for a falling edge detection)
- GIE bit of the INTCON0 register

The associated interrupt flag bit, CxIF bit of the respective PIR register, must be cleared in software. If another edge is detected while this flag is being cleared, the flag will still be set at the end of the sequence.

Note:	Although a comparator is disabled, an interrupt can be generated by changing the output polarity with the POL bit of the CMxCON0 register, or by switching the comparator on or off with the EN bit of the CMxCON0 register.

38.5 Comparator Positive Input Selection

Configuring the PCH<2:0> bits of the CMxPCH register directs an internal voltage reference or an analog pin to the non-inverting input of the comparator:

- CxIN0+, CxIN1+ analog pin
- DAC output
- FVR (Fixed Voltage Reference)
- VSS (Ground)

See **Section 34.0 “Fixed Voltage Reference (FVR)”** for more information on the Fixed Voltage Reference module.

See **Section 37.0 “5-Bit Digital-to-Analog Converter (DAC) Module”** for more information on the DAC input signal.

Any time the comparator is disabled (EN = 0), all comparator inputs are disabled.

38.6 Comparator Negative Input Selection

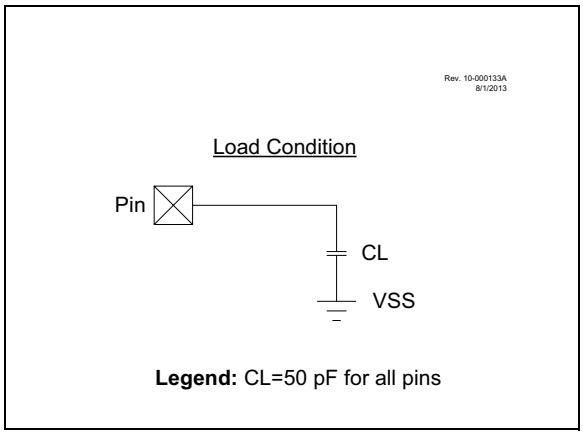
The NCH<2:0> bits of the CMxNCH register direct an analog input pin and internal reference voltage or analog ground to the inverting input of the comparator:

- CxIN0-, CxIN1-, CxIN2-, CxIN3- analog pin
- FVR (Fixed Voltage Reference)
- Analog Ground

Note:	To use CxINy+ and CxINy- pins as analog input, the appropriate bits must be set in the ANSEL register and the corresponding TRIS bits must also be set to disable the output drivers.

44.4 AC Characteristics

FIGURE 44-4: LOAD CONDITIONS



Legend: CL=50 pF for all pins