



Welcome to [E-XFL.COM](https://www.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	Obsolete
Core Processor	PIC
Core Size	8-Bit
Speed	4MHz
Connectivity	-
Peripherals	Brown-out Detect/Reset, POR, WDT
Number of I/O	13
Program Memory Size	1.75KB (1K x 14)
Program Memory Type	OTP
EEPROM Size	128 x 8
RAM Size	96 x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	-
Oscillator Type	External
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	18-SOIC (0.295", 7.50mm Width)
Supplier Device Package	18-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lce624-04e-so

PIC16CE62X

4.2.2.4 PIE1 REGISTER

This register contains the individual enable bit for the comparator interrupt.

REGISTER 4-4: PIE1 REGISTER (ADDRESS 8CH)

U-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
—	CMIE	—	—	—	—	—	—
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
-n = Value at POR reset
-x = Unknown at POR reset

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

bit 6: **CMIE:** Comparator Interrupt Enable bit
1 = Enables the Comparator interrupt
0 = Disables the Comparator interrupt

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

4.2.2.5 PIR1 REGISTER

This register contains the individual flag bit for the comparator interrupt.

Note: Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 4-5: PIR1 REGISTER (ADDRESS 0CH)

U-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
—	CMIF	—	—	—	—	—	—
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
-n = Value at POR reset
-x = Unknown at POR reset

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

bit 6: **CMIF:** Comparator Interrupt Flag bit
1 = Comparator input has changed
0 = Comparator input has not changed

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

4.2.2.6 PCON REGISTER

The PCON register contains flag bits to differentiate between a Power-on Reset, an external $\overline{\text{MCLR}}$ reset, WDT reset or a Brown-out Reset.

Note: $\overline{\text{BOD}}$ is unknown on Power-on Reset. It must then be set by the user and checked on subsequent resets to see if $\overline{\text{BOD}}$ is cleared, indicating a brown-out has occurred. The $\overline{\text{BOD}}$ status bit is a "don't care" and is not necessarily predictable if the brown-out circuit is disabled (by programming BODEN bit in the configuration word).

REGISTER 4-6: PCON REGISTER (ADDRESS 8Eh)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	POR	$\overline{\text{BOD}}$
bit7						bit0	

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

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

R = Readable bit
 W = Writable bit
 U = Unimplemented bit, read as '0'
 -n = Value at POR reset
 -x = Unknown at POR reset

6.0 EEPROM PERIPHERAL OPERATION

The PIC16CE623/624/625 each have 128 bytes of EEPROM data memory. The EEPROM data memory supports a bi-directional, 2-wire bus and data transmission protocol. These two-wires are serial data (SDA) and serial clock (SCL), and are mapped to bit1 and bit2, respectively, of the EEINTF register (SFR 90h). In addition, the power to the EEPROM can be controlled using bit0 (EEVDD) of the EEINTF register. For most applications, all that is required is calls to the following functions:

```
; Byte_Write: Byte write routine
;   Inputs: EEPROM Address      EEADDR
;           EEPROM Data         EEDATA
;   Outputs: Return 01 in W if OK, else
;           return 00 in W
;
; Read_Current: Read EEPROM at address
;               currently held by EE device.
;   Inputs: NONE
;   Outputs: EEPROM Data        EEDATA
;           Return 01 in W if OK, else
;           return 00 in W
;
; Read_Random: Read EEPROM byte at supplied
; address
;   Inputs: EEPROM Address      EEADDR
;   Outputs: EEPROM Data        EEDATA
;           Return 01 in W if OK,
;           else return 00 in W
;
```

The code for these functions is available on our web site (www.microchip.com). The code will be accessed by either including the source code FL62XINC.ASM or by linking FLASH62X.ASM. FLASH62.IMC provides external definition to the calling program.

6.0.1 SERIAL DATA

SDA is a bi-directional pin used to transfer addresses and data into and data out of the memory.

For normal data transfer, SDA is allowed to change only during SCL low. Changes during SCL high are reserved for indicating the START and STOP conditions.

6.0.2 SERIAL CLOCK

This SCL input is used to synchronize the data transfer to and from the memory.

6.0.3 EEINTF REGISTER

The EEINTF register (SFR 90h) controls the access to the EEPROM. Register 6-1 details the function of each bit. User code must generate the clock and data signals.

REGISTER 6-1: EEINTF REGISTER (ADDRESS 90h)

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1
—	—	—	—	—	EESCL	EESDA	EEVDD
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7-3: **Unimplemented:** Read as '0'

bit 2: **EESCL:** Clock line to the EEPROM
1 = Clock high
0 = Clock low

bit 1: **EESDA:** Data line to EEPROM
1 = Data line is high (pin is tri-stated, line is pulled high by a pull-up resistor)
0 = Data line is low

bit 0: **EEVDD:** VDD control bit for EEPROM
1 = VDD is turned on to EEPROM
0 = VDD is turned off to EEPROM (all pins are tri-stated and the EEPROM is powered down)

Note: EESDA, EESCL and EEVDD will read '0' if EEVDD is turned off.

6.1 Bus Characteristics

In this section, the term “processor” refers to the portion of the PIC16CE62X that interfaces to the EEPROM through software manipulating the EEINTF register. The following **bus protocol** is to be used with the EEPROM data memory.

- Data transfer may be initiated only when the bus is not busy.
- During data transfer, the data line must remain stable whenever the clock line is HIGH. Changes in the data line while the clock line is HIGH will be interpreted by the EEPROM as a START or STOP condition.

Accordingly, the following bus conditions have been defined (Figure 6-1).

6.1.1 BUS NOT BUSY (A)

Both data and clock lines remain HIGH.

6.1.2 START DATA TRANSFER (B)

A HIGH to LOW transition of the SDA line while the clock (SCL) is HIGH determines a START condition. All commands must be preceded by a START condition.

6.1.3 STOP DATA TRANSFER (C)

A LOW to HIGH transition of the SDA line while the clock (SCL) is HIGH determines a STOP condition. All operations must be ended with a STOP condition.

6.1.4 DATA VALID (D)

The state of the data line represents valid data when, after a START condition, the data line is stable for the duration of the HIGH period of the clock signal.

The data on the line must be changed during the LOW period of the clock signal. There is one bit of data per clock pulse.

Each data transfer is initiated with a START condition and terminated with a STOP condition. The number of the data bytes transferred between the START and STOP conditions is determined by the processor and is theoretically unlimited, although only the last sixteen will be stored when doing a write operation. When an overwrite does occur, it will replace data in a first-in, first-out fashion.

6.1.5 ACKNOWLEDGE

The EEPROM will generate an acknowledge after the reception of each byte. The processor must generate an extra clock pulse which is associated with this acknowledge bit.

Note: Acknowledge bits are not generated if an internal programming cycle is in progress.
--

When the EEPROM acknowledges, it pulls down the SDA line during the acknowledge clock pulse in such a way that the SDA line is stable LOW during the HIGH period of the acknowledge related clock pulse. Of course, setup and hold times must be taken into account. The processor must signal an end of data to the EEPROM by not generating an acknowledge bit on the last byte that has been clocked out of the EEPROM. In this case, the EEPROM must leave the data line HIGH to enable the processor to generate the STOP condition (Figure 6-2).

FIGURE 6-7: CURRENT ADDRESS READ

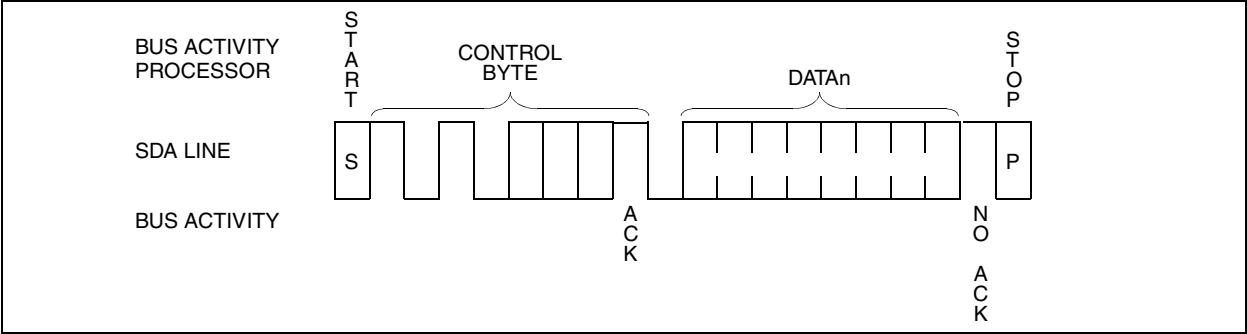


FIGURE 6-8: RANDOM READ

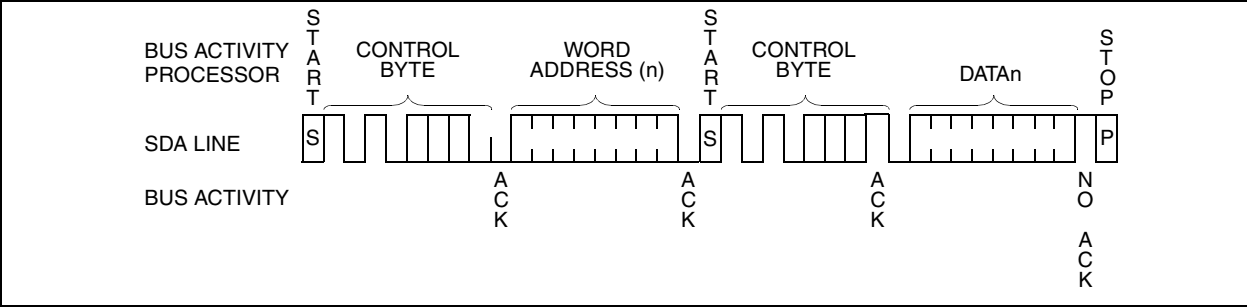


FIGURE 6-9: SEQUENTIAL READ

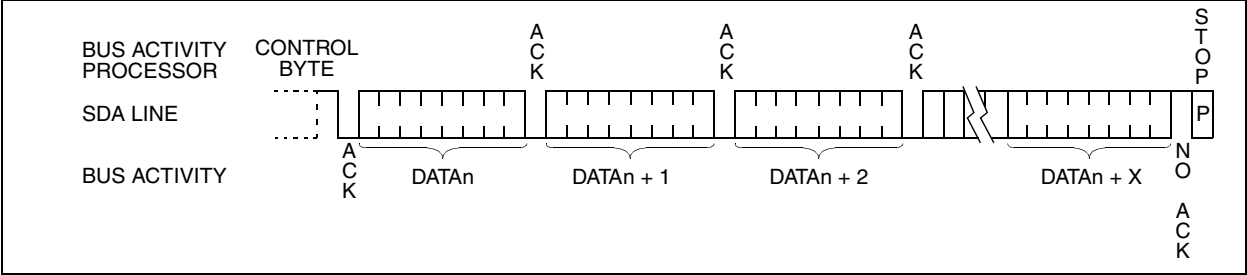


FIGURE 7-3: TIMER0 TIMING: INTERNAL CLOCK/PRESCALE 1:2

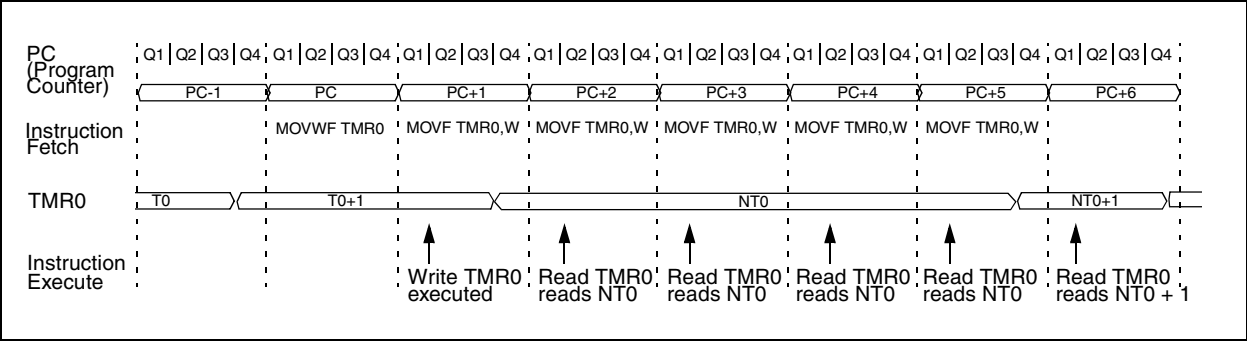
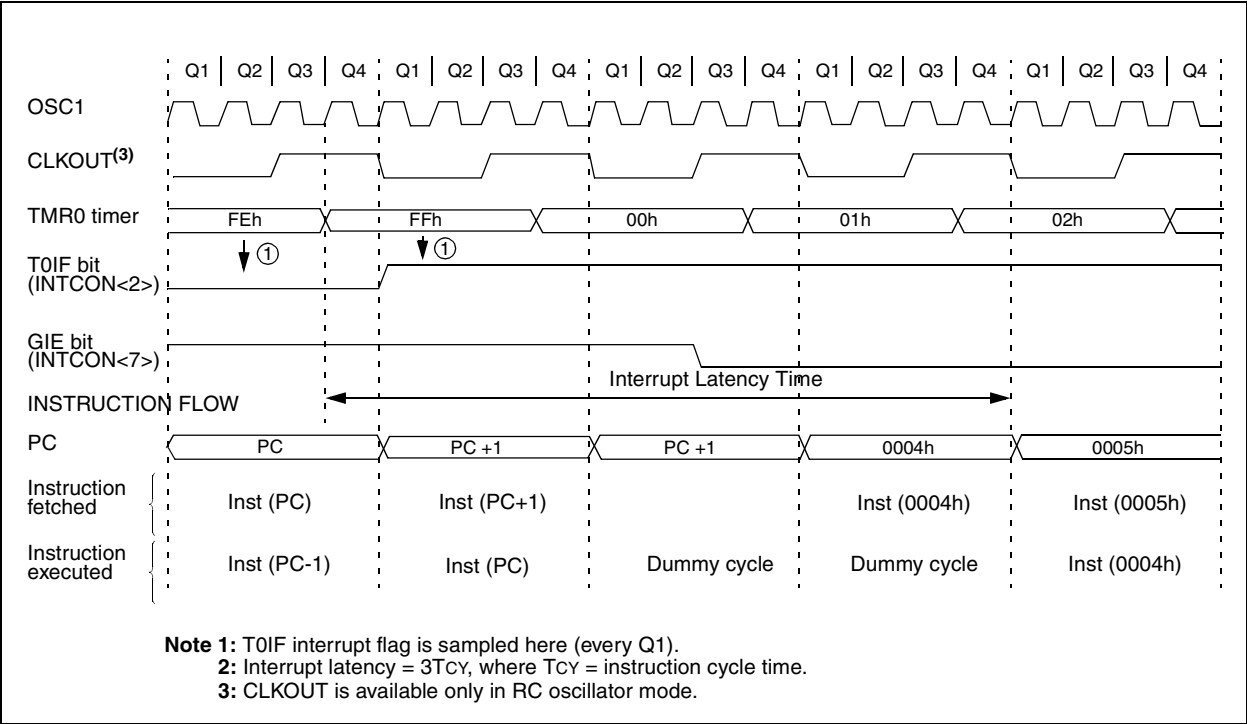


FIGURE 7-4: TIMER0 INTERRUPT TIMING



PIC16CE62X

NOTES:

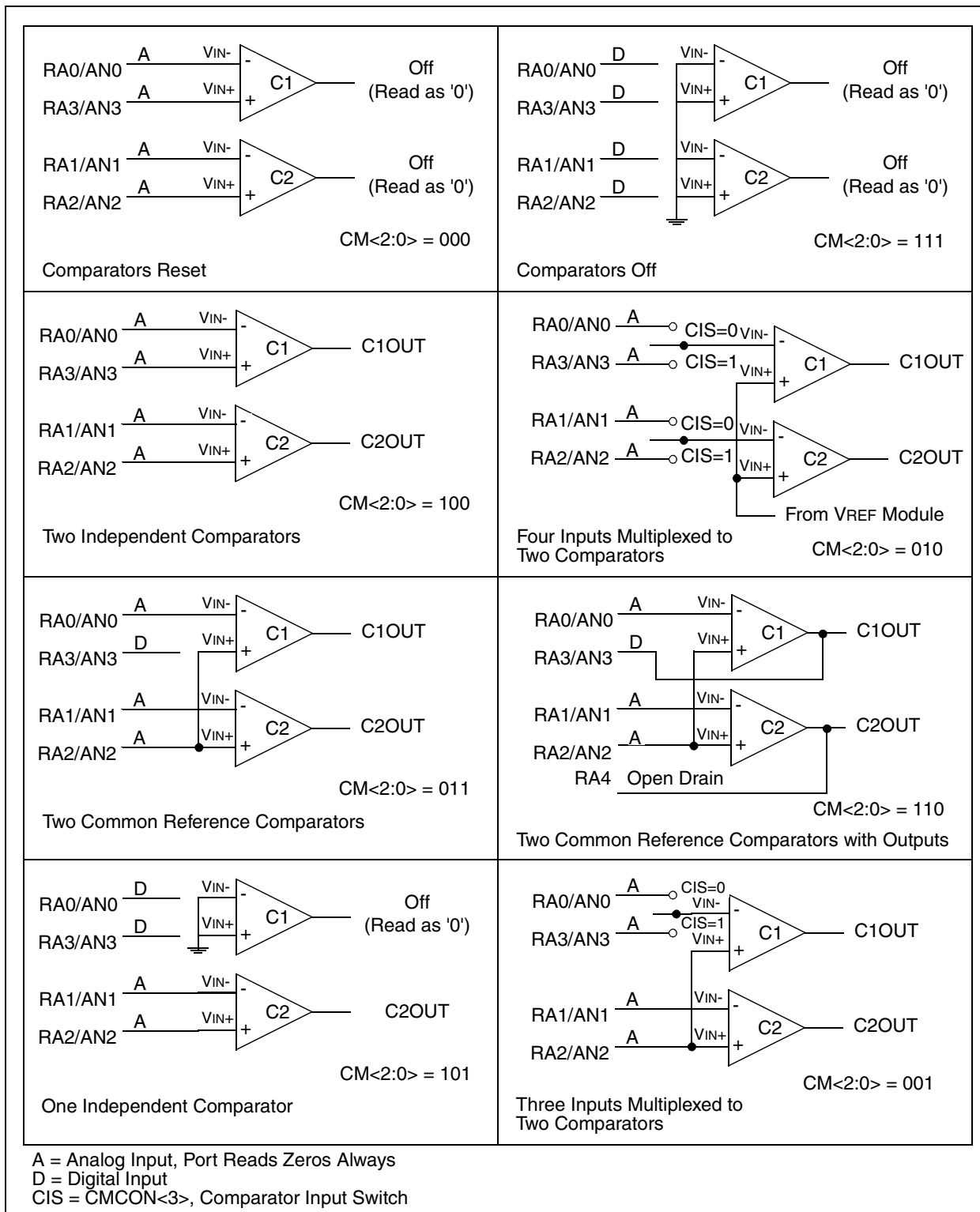
8.1 Comparator Configuration

There are eight modes of operation for the comparators. The CMCON register is used to select the mode. Figure 8-1 shows the eight possible modes. The TRISA register controls the data direction of the comparator pins for each mode. If the comparator

mode is changed, the comparator output level may not be valid for the specified mode change delay shown in Table 13-1.

Note: Comparator interrupts should be disabled during a comparator mode change, otherwise a false interrupt may occur.

FIGURE 8-1: COMPARATOR I/O OPERATING MODES



8.6 Comparator Interrupts

The comparator interrupt flag is set whenever there is a change in the output value of either comparator. Software will need to maintain information about the status of the output bits, as read from CMCON<7:6>, to determine the actual change that has occurred. The CMIF bit, PIR1<6>, is the comparator interrupt flag. The CMIF bit must be reset by clearing '0'. Since it is also possible to write a '1' to this register, a simulated interrupt may be initiated.

The CMIE bit (PIE1<6>) and the PEIE bit (INTCON<6>) must be set to enable the interrupt. In addition, the GIE bit must also be set. If any of these bits are clear, the interrupt is not enabled, though the CMIF bit will still be set if an interrupt condition occurs.

Note: If a change in the CMCON register (C1OUT or C2OUT) should occur when a read operation is being executed (start of the Q2 cycle), then the CMIF (PIR1<6>) interrupt flag may not get set.

The user, in the interrupt service routine, can clear the interrupt in the following manner:

- a) Any read or write of CMCON. This will end the mismatch condition.
- b) Clear flag bit CMIF.

A mismatch condition will continue to set flag bit CMIF. Reading CMCON will end the mismatch condition, and allow flag bit CMIF to be cleared.

8.7 Comparator Operation During SLEEP

When a comparator is active and the device is placed in SLEEP mode, the comparator remains active and the interrupt is functional if enabled. This interrupt will

wake-up the device from SLEEP mode when enabled. While the comparator is powered-up, higher sleep currents than shown in the power down current specification will occur. Each comparator that is operational will consume additional current as shown in the comparator specifications. To minimize power consumption while in SLEEP mode, turn off the comparators, CM<2:0> = 111, before entering sleep. If the device wakes-up from sleep, the contents of the CMCON register are not affected.

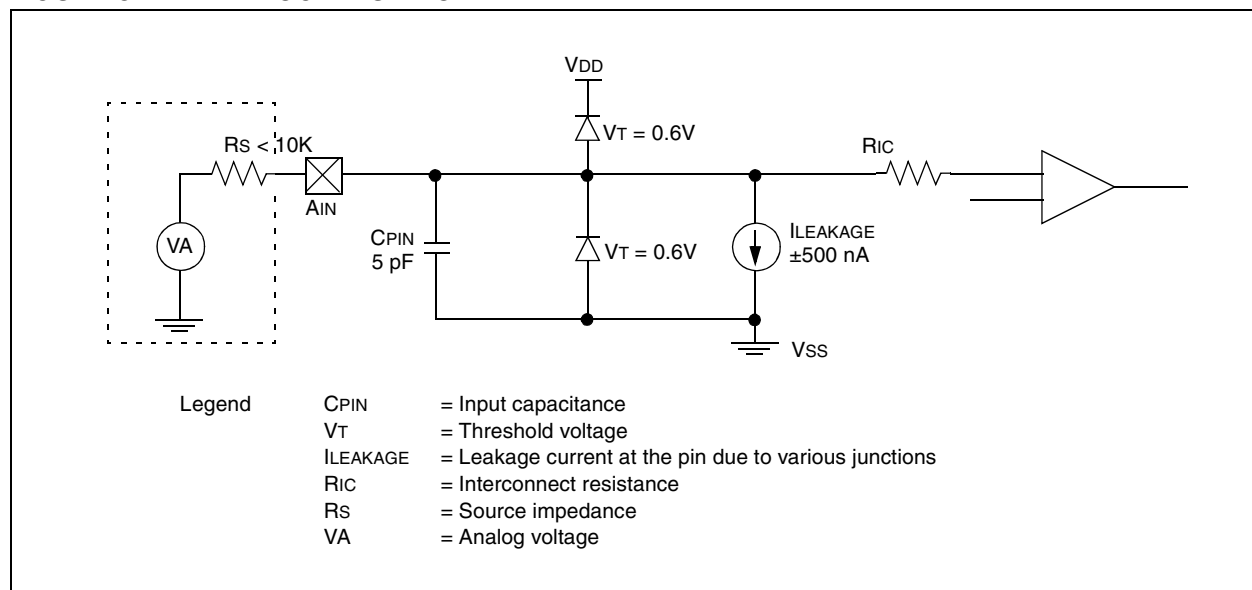
8.8 Effects of a RESET

A device reset forces the CMCON register to its reset state. This forces the comparator module to be in the comparator reset mode, CM<2:0> = 000. This ensures that all potential inputs are analog inputs. Device current is minimized when analog inputs are present at reset time. The comparators will be powered-down during the reset interval.

8.9 Analog Input Connection Considerations

A simplified circuit for an analog input is shown in Figure 8-4. Since the analog pins are connected to a digital output, they have reverse biased 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. Any external component connected to an analog input pin, such as a capacitor or a Zener diode, should have very little leakage current.

FIGURE 8-4: ANALOG INPUT MODEL



10.3 Reset

The PIC16CE62X differentiates between various kinds of reset:

- Power-on reset (POR)
- $\overline{\text{MCLR}}$ reset during normal operation
- $\overline{\text{MCLR}}$ reset during SLEEP
- WDT reset (normal operation)
- WDT wake-up (SLEEP)
- Brown-out Reset (BOD)

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, WDT reset and $\overline{\text{MCLR}}$ reset during SLEEP. 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 10-4. These bits are used in software to determine the nature of the reset. See Table 10-6 for a full description of reset states of all registers.

A simplified block diagram of the on-chip reset circuit is shown in Figure 10-6.

The $\overline{\text{MCLR}}$ reset path has a noise filter to detect and ignore small pulses. See Table 13-5 for pulse width specification.

FIGURE 10-6: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT

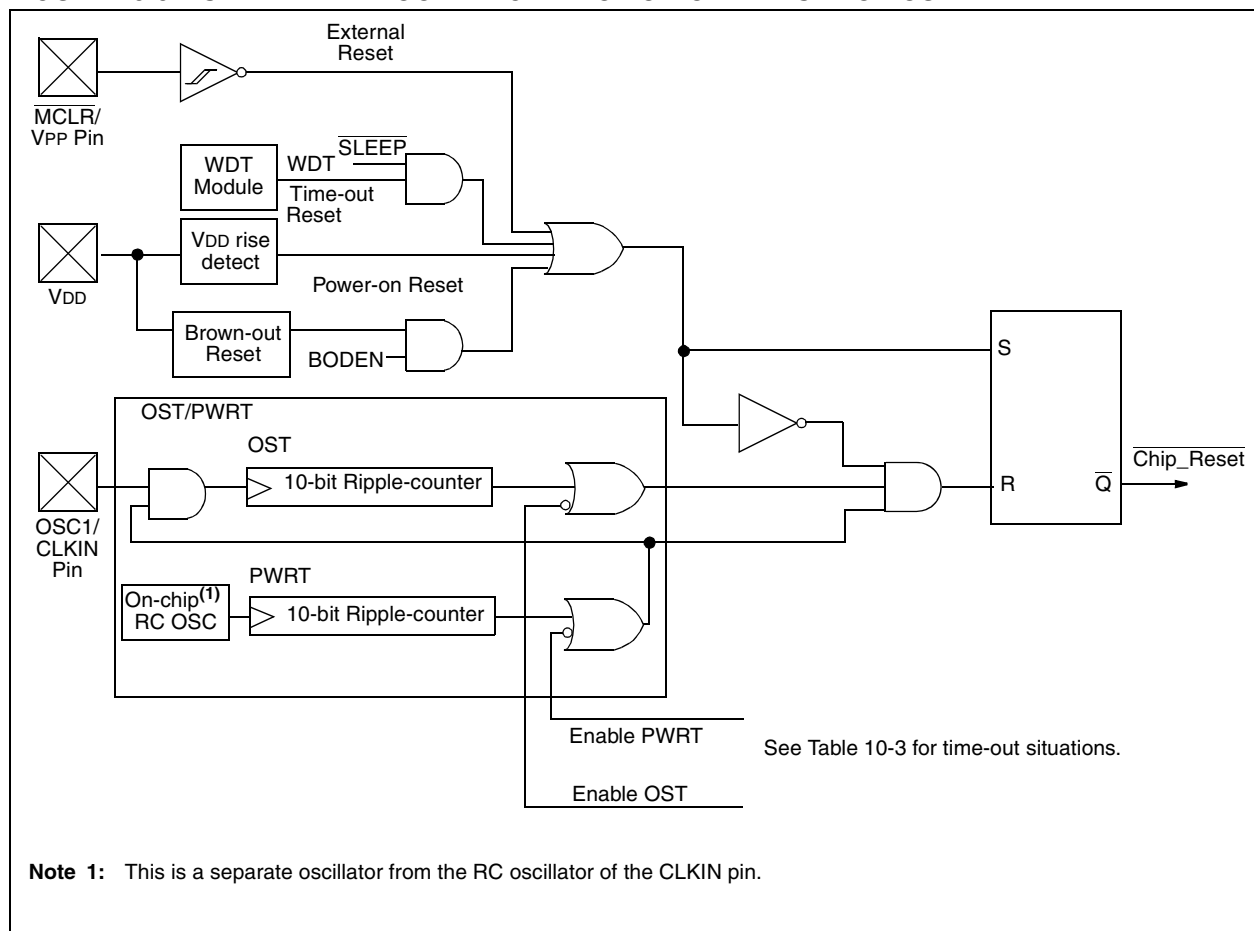


FIGURE 10-17: WATCHDOG TIMER BLOCK DIAGRAM

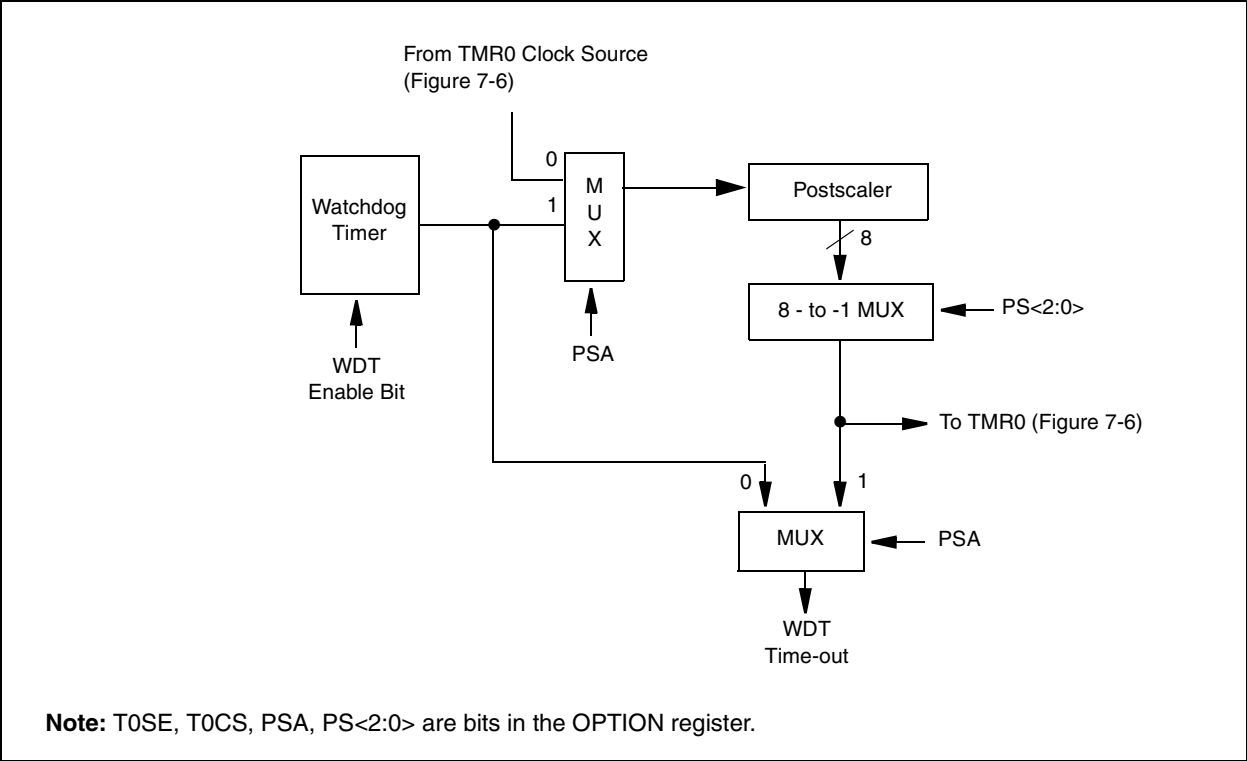


FIGURE 10-18: SUMMARY OF WATCHDOG TIMER REGISTERS

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2007h	Config. bits	—	BOREN	CP1	CP0	PWRTE	WDTE	FOSC1	FOSC0
81h	OPTION	RBP \overline{U}	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0

Legend: – = Unimplemented location, read as “0”, + = Reserved for future use

Note: Shaded cells are not used by the Watchdog Timer.

11.0 INSTRUCTION SET SUMMARY

Each PIC16CE62X 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 PIC16CE62X instruction set summary in Table 11-2 lists **byte-oriented**, **bit-oriented**, and **literal and control** operations. Table 11-1 shows the opcode field descriptions.

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 number of the bit affected by the operation, while 'f' represents the number of the file in which the bit is located.

For **literal and control** operations, 'k' represents an eight or eleven bit constant or literal value.

TABLE 11-1: 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
label	Label name
TOS	Top of Stack
PC	Program Counter
PCLATH	Program Counter High Latch
GIE	Global Interrupt Enable bit
WDT	Watchdog Timer/Counter
TO	Time-out bit
PD	Power-down bit
dest	Destination either the W register or the specified register file location
[]	Options
()	Contents
→	Assigned to
< >	Register bit field
∈	In the set of
<i>italics</i>	User defined term (font is courier)

The instruction set is highly orthogonal and is grouped into three basic categories:

- **Byte-oriented** operations
- **Bit-oriented** operations
- **Literal and control** operations

All instructions are executed within one single instruction cycle, unless a conditional test is true or the program counter is changed as a result of an instruction. In this case, the execution takes two instruction cycles with the second cycle executed as a NOP. One instruction cycle consists of four oscillator periods. Thus, for an oscillator frequency of 4 MHz, the normal instruction execution time is 1 μs. If a conditional test is true or the program counter is changed as a result of an instruction, the instruction execution time is 2 μs.

Table 11-1 lists the instructions recognized by the MPASM assembler.

Figure 11-1 shows the three general formats that the instructions can have.

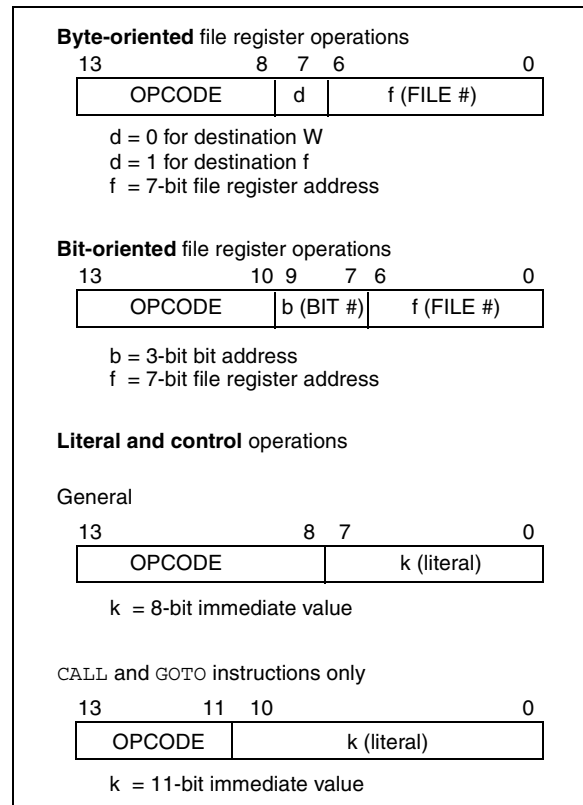
Note: To maintain upward compatibility with future PIC® MCU products, do not use the `OPTION` and `TRIS` instructions.

All examples use the following format to represent a hexadecimal number:

0xhh

where h signifies a hexadecimal digit.

FIGURE 11-1: GENERAL FORMAT FOR INSTRUCTIONS



GOTO		Unconditional Branch							
Syntax:	[<i>label</i>] GOTO k								
Operands:	$0 \leq k \leq 2047$								
Operation:	$k \rightarrow PC<10:0>$ $PCLATH<4:3> \rightarrow PC<12:11>$								
Status Affected:	None								
Encoding:	<table><tr><td>10</td><td>1kkk</td><td>kkkk</td><td>kkkk</td></tr></table>					10	1kkk	kkkk	kkkk
10	1kkk	kkkk	kkkk						
Description:	<p>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>.</p> <p>GOTO is a two-cycle instruction.</p>								
Words:	1								
Cycles:	2								
Example	<pre> GOTO THERE After Instruction PC = Address THERE</pre>								

INCFSZ		Increment f, Skip if 0							
Syntax:	[<i>label</i>] INCFSZ f,d								
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$								
Operation:	$(f) + 1 \rightarrow (\text{dest})$, skip if result = 0								
Status Affected:	None								
Encoding:	<table border="1"><tr><td>00</td><td>1111</td><td>dfff</td><td>ffff</td></tr></table>					00	1111	dfff	ffff
00	1111	dfff	ffff						
Description:	<p>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 0, the next instruction, which is already fetched, is discarded. A NOP is executed instead making it a two-cycle instruction.</p>								
Words:	1								
Cycles:	1(2)								
Example	HERE INCFSZ CNT, 1								

Before Instruction
 PC = address HERE
 After Instruction
 CNT = CNT + 1
 if CNT= 0,
 PC = address CONTINUE
 if CNT≠ 0,
 PC = address HERE +1

INCF		Increment f								
Syntax:	[<i>label</i>] INCF f,d									
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$									
Operation:	$(f) + 1 \rightarrow (\text{dest})$									
Status Affected:	Z									
Encoding:	<table><tr><td>00</td><td>1010</td><td>dfff</td><td>ffff</td></tr></table>						00	1010	dfff	ffff
00	1010	dfff	ffff							
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'.									
Words:	1									
Cycles:	1									
Example	INCF CNT, 1									
	Before Instruction									
	CNT		=	0xFF						
	Z		=	0						
	After Instruction									
	CNT		=	0x00						
	Z		=	1						

IORLW		Inclusive OR Literal with W						
Syntax:	[<i>label</i>] IORLW k							
Operands:	$0 \leq k \leq 255$							
Operation:	(W) .OR. k \rightarrow (W)							
Status Affected:	Z							
Encoding:	<table border="1"><tr><td>11</td><td>1000</td><td>kkkk</td><td>kkkk</td></tr></table>				11	1000	kkkk	kkkk
11	1000	kkkk	kkkk					
Description:	The contents of the W register are OR'ed with the eight bit literal 'k'. The result is placed in the W register.							
Words:	1							
Cycles:	1							
Example	IORLW 0x35							
	Before Instruction							
	W = 0x9A							
	After Instruction							
	W = 0xBF							
	Z = 1							

IORWF		Inclusive OR W with f							
Syntax:	[<i>label</i>] IORWF f,d								
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$								
Operation:	(W) .OR. (f) \rightarrow (dest)								
Status Affected:	Z								
Encoding:	<table border="1"><tr><td>00</td><td>0100</td><td>dfff</td><td>ffff</td></tr></table>					00	0100	dfff	ffff
00	0100	dfff	ffff						
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'.								
Words:	1								
Cycles:	1								
Example	IORWF RESULT, 0								
	Before Instruction								
	RESULT	=	0x13						
	W	=	0x91						
	After Instruction								
	RESULT	=	0x13						
	W	=	0x93						
	Z	=	1						

MOVLW	Move Literal to W				
Syntax:	[<i>label</i>] MOVLW k				
Operands:	$0 \leq k \leq 255$				
Operation:	$k \rightarrow (W)$				
Status Affected:	None				
Encoding:	<table><tr><td>11</td><td>00xx</td><td>kkkk</td><td>kkkk</td></tr></table>	11	00xx	kkkk	kkkk
11	00xx	kkkk	kkkk		
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	<pre>MOVLW 0x5A</pre> <p>After Instruction</p> <p>W = 0x5A</p>				

MOVF	Move f				
Syntax:	[<i>label</i>] MOVF f,d				
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$				
Operation:	(f) → (dest)				
Status Affected:	Z				
Encoding:	<table><tr><td>00</td><td>1000</td><td>dfff</td><td>ffff</td></tr></table>	00	1000	dfff	ffff
00	1000	dfff	ffff		
Description:	The contents of register f are moved to a destination dependant 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:	[<i>label</i>] MOVWF f								
Operands:	$0 \leq f \leq 127$								
Operation:	(W) → (f)								
Status Affected:	None								
Encoding:	<table><tr><td>00</td><td>0000</td><td>1fff</td><td>ffff</td></tr></table>					00	0000	1fff	ffff
00	0000	1fff	ffff						
Description:	Move data from W register to register 'f'.								
Words:	1								
Cycles:	1								
Example	MOVWF OPTION								
	Before Instruction								
	OPTION = 0xFF								
	W = 0x4F								
	After Instruction								
	OPTION = 0x4F								
	W = 0x4F								

PIC16CE62X

NOTES:

TABLE 13-1: COMPARATOR SPECIFICATIONS

Operating Conditions: VDD range as described in Table 12-1, -40°C<TA<+125°C. .

Param No.	Characteristics	Sym	Min	Typ	Max	Units	Comments
D300	Input offset voltage	VIOFF		± 5.0	± 10	mV	
D301	Input common mode voltage	VICM	0		VDD - 1.5	V	
D302	CMRR	CMRR	+55*			db	
300	Response Time ⁽¹⁾	TRESP		150*	400*	ns	PIC16CE62X
301	Comparator Mode Change to Output Valid	TMC2OV			10*	µs	

* These parameters are characterized but not tested.

Note 1: Response time measured with one comparator input at (VDD - 1.5)/2 while the other input transitions from VSS to VDD.

TABLE 13-2: VOLTAGE REFERENCE SPECIFICATIONS

Operating Conditions: VDD range as described in Table 12-1, -40°C<TA<+125°C.

Param No.	Characteristics	Sym	Min	Typ	Max	Units	Comments
D310	Resolution	VRES	VDD/24		VDD/32	LSB	
D311	Absolute Accuracy	VRAA			±1/4 ±1/2	LSB LSB	Low Range (VRR=1) High Range (VRR=0)
D312	Unit Resistor Value (R)	VRUR		2K*		Ω	Figure 9-1
310	Settling Time ⁽¹⁾	TSET			10*	µs	

* These parameters are characterized but not tested.

Note 1: Settling time measured while VRR = 1 and VR<3:0> transitions from 0000 to 1111.

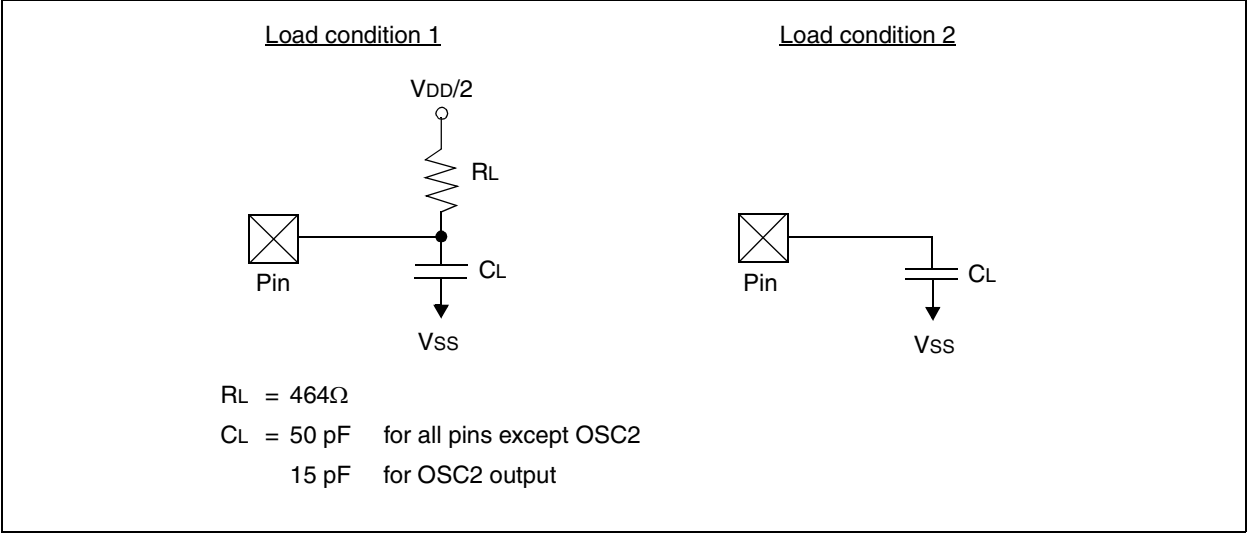
13.4 Timing Parameter Symbology

The timing parameter symbols have been created with one of the following formats:

- 1. TppS2ppS
- 2. TppS

T			
F	Frequency	T	Time
Lowercase subscripts (pp) and their meanings:			
pp			
ck	CLKOUT	osc	OSC1
io	I/O port	t0	T0CKI
mc	MCLR		
Uppercase letters and their meanings:			
S			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (Hi-impedance)	V	Valid
L	Low	Z	Hi-Impedance

FIGURE 13-4: LOAD CONDITIONS



PIC16CE62X

NOTES:

PIC16XXXXXX FAMILY

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: PIC16xxxxxx family

Literature Number: DS40182D

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?

Worldwide Sales and Service

AMERICAS

Corporate Office

2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
<http://www.microchip.com/support>
Web Address:
www.microchip.com

Atlanta

Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston

Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago

Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland

Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas

Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit

Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Indianapolis

Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

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: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu

Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing

Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hangzhou

Tel: 86-571-2819-3187
Fax: 86-571-2819-3189

China - Hong Kong SAR

Tel: 852-2943-5100
Fax: 852-2401-3431

China - Nanjing

Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao

Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai

Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang

Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen

Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

China - Wuhan

Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian

Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Xiamen

Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai

Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore

Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi

Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune

Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Osaka

Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

Japan - Tokyo

Tel: 81-3-6880-3770
Fax: 81-3-6880-3771

Korea - Daegu

Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul

Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur

Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang

Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila

Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore

Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu

Tel: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung

Tel: 886-7-213-7828
Fax: 886-7-330-9305

Taiwan - Taipei

Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

Thailand - Bangkok

Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels

Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen

Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris

Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - 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

11/29/12