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	20MHz
Connectivity	-
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	5
Program Memory Size	1.75KB (1K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	64 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	8-TSSOP, 8-MSOP (0.118", 3.00mm Width)
Supplier Device Package	8-MSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic12f609t-i-ms

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 (RAM)

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 the **Section 14.0 "Instruction Set Summary"**.

Note 1:	Bits IRP and RP1 of the STATUS register are not used by the PIC12F609/615/617/12HV609/615 and should be maintained as clear. Use of these bits is not recommended, since this may affect upward compatibility with future products.
2:	The C and DC bits operate as a $\overline{Borrow}$ and Digit $\overline{Borrow}$ out bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

REGISTER 2-1: STATUS: STATUS REGISTER

Reserved	Reserved	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	C
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:** This bit is reserved and should be maintained as '0'

bit 6 **RP1:** This bit is reserved and should be maintained as '0'

bit 5 **RP0:** Register Bank Select bit (used for direct addressing)
1 = Bank 1 (80h – FFh)
0 = Bank 0 (00h – 7Fh)

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), For $\overline{Borrow}$, the polarity is reversed.
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 = 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.

FIGURE 3-2: BLOCK WRITES TO 2K FLASH PROGRAM MEMORY

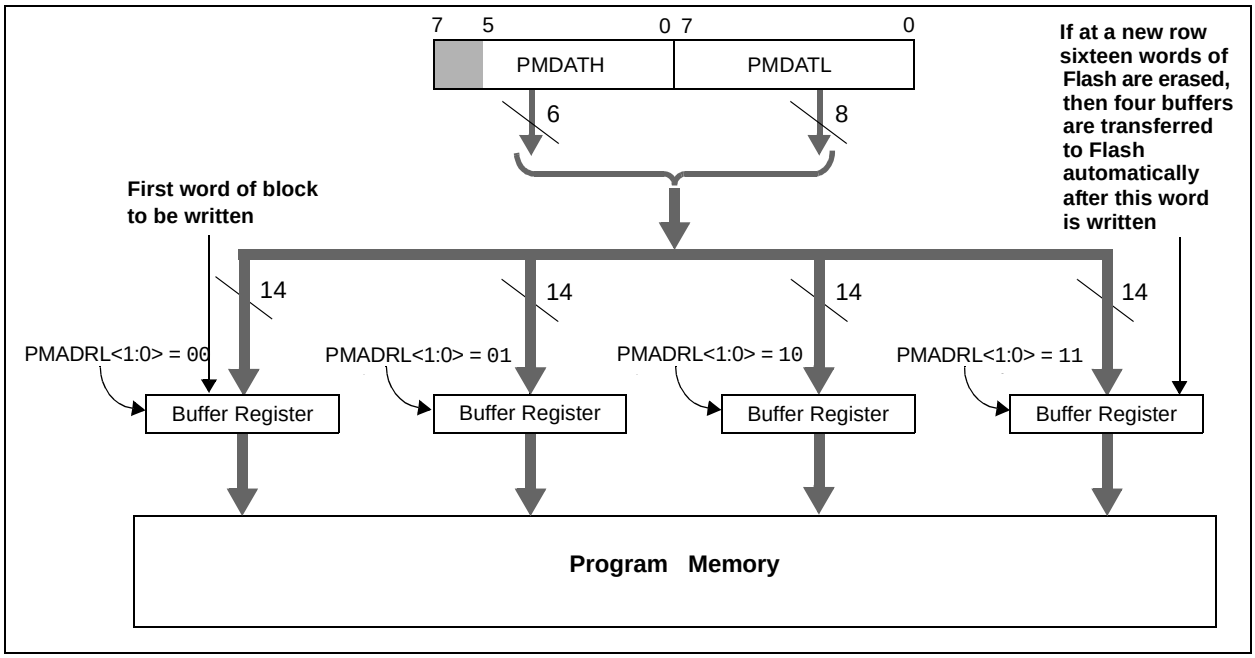
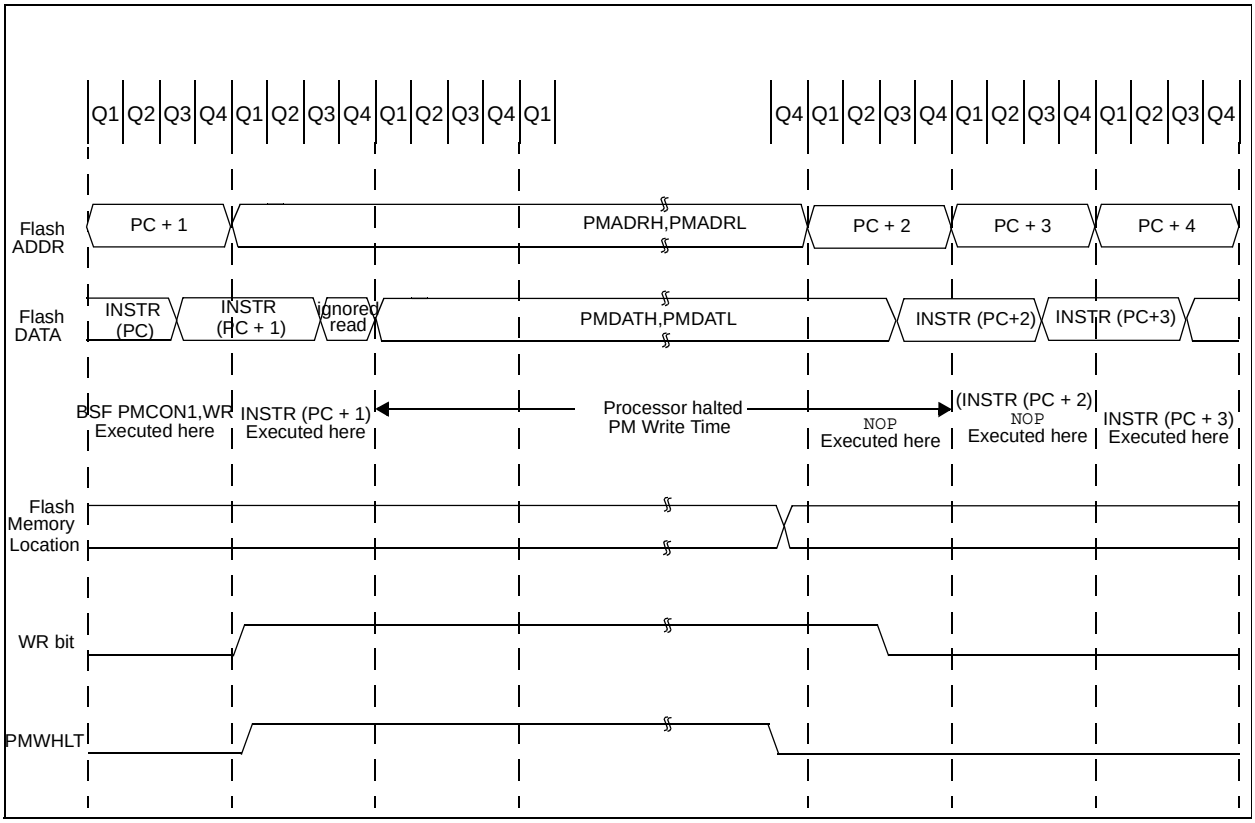


FIGURE 3-3: FLASH PROGRAM MEMORY LONG WRITE CYCLE EXECUTION



NOTES:

6.1.3 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.

6.1.3.1 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 6-1, must be executed.

EXAMPLE 6-1: 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 6-2).

EXAMPLE 6-2: 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    ;
```

6.1.4 TIMER0 INTERRUPT

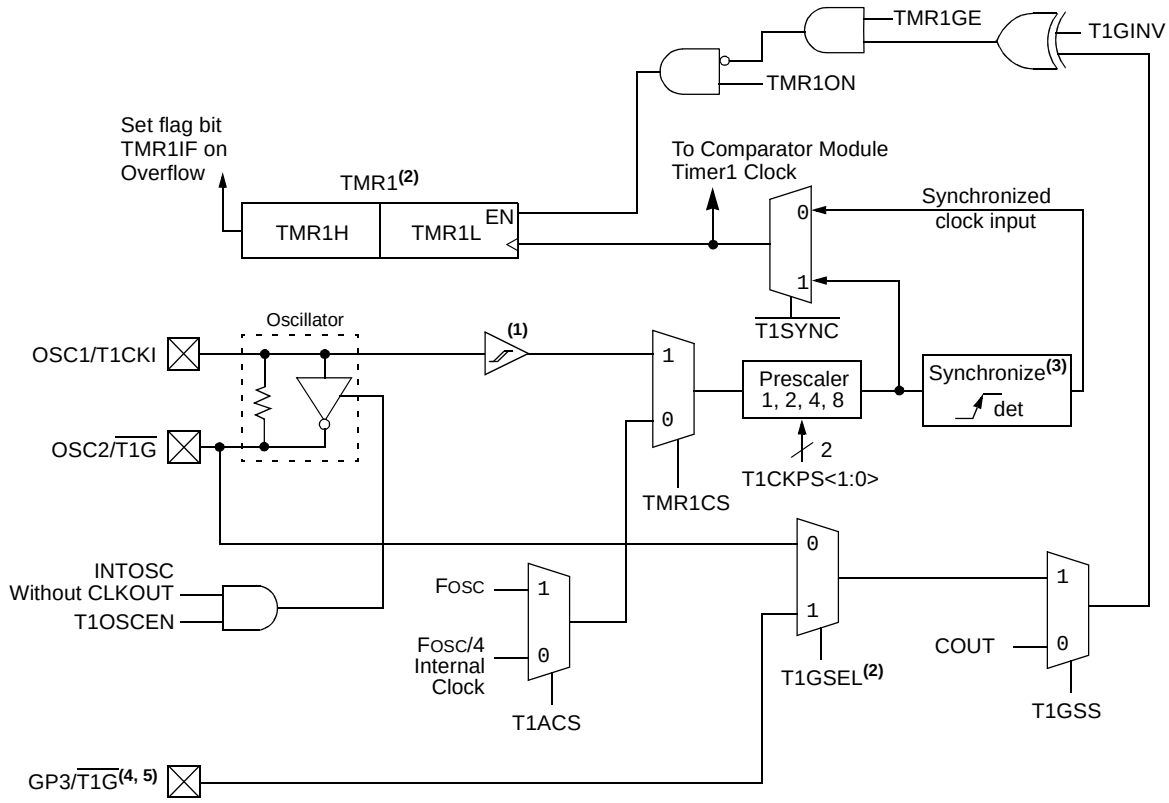
Timer0 will generate an interrupt when the TMR0 register overflows from FFh to 00h. The T0IF interrupt flag bit of the INTCON register is set every time the TMR0 register overflows, regardless of whether or not the Timer0 interrupt is enabled. The T0IF bit must be cleared in software. The Timer0 interrupt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

6.1.5 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 16.0 "Electrical Specifications"**.

FIGURE 7-1: 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: Alternate pin function.
5: PIC12F615/617/HV615 only.

7.12 Timer1 Control Register

The Timer1 Control register (T1CON), shown in Register 7-1, is used to control Timer1 and select the various features of the Timer1 module.

REGISTER 7-1: 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 is on if Timer1 gate is active
- 0 = Timer1 is on

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

For all other system clock modes:
This bit is ignored. LP oscillator is disabled.

bit 2 **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) or system clock (FOSC)(3)

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 COUT, as selected by the T1GSS bit of the CMCON1 register, as a Timer1 gate source.

3: See T1ACS bit in CMCON1 register.

TABLE 7-1: 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
APFCON(1)	—	—	—	T1GSEL	—	—	P1BSEL	P1ASEL	---0 --00	---0 --00
CMCON0	CMON	COUT	CMOE	CMPOL	—	CMR	—	CMCH	0000 -0-0	0000 -0-0
CMCON1	—	—	—	T1ACS	CMHYS	—	T1GSS	CMSYNC	---0 0-10	---0 0-10
INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 000x	0000 000x
PIE1	—	ADIE(1)	CCP1IE(1)	—	CMIE	—	TMR2IE(1)	TMR1IE	-00- 0-00	-00- 0-00
PIR1	—	ADIF(1)	CCP1IF(1)	—	CMIF	—	TMR2IF(1)	TMR1IF	-00- 0-00	-00- 0-00
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.

Note 1: PIC12F615/617/HV615 only.

9.0 COMPARATOR MODULE

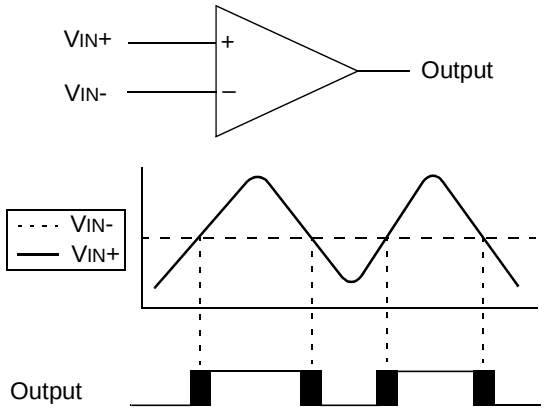
The comparator can be used to interface analog circuits to a digital circuit by comparing two analog voltages and providing a digital indication of their relative magnitudes. The comparator is a very useful mixed signal building block because it provides analog functionality independent of the program execution. The Analog Comparator module includes the following features:

- Programmable input section
- Comparator output is available internally/externally
- Programmable output polarity
- Interrupt-on-change
- Wake-up from Sleep
- PWM shutdown
- Timer1 gate (count enable)
- Output synchronization to Timer1 clock input
- Programmable voltage reference
- User-enable Comparator Hysteresis

9.1 Comparator Overview

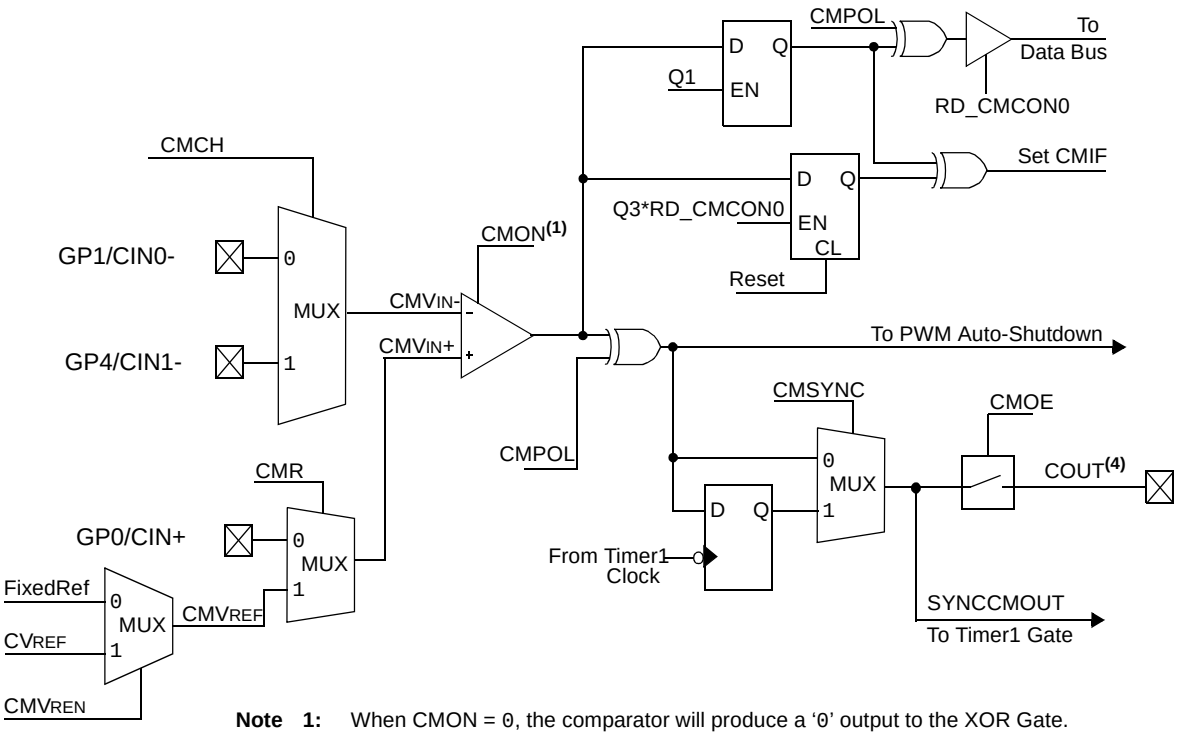
The comparator is shown in Figure 9-1 along with the relationship between the analog input levels and the digital output. When the analog voltage at VIN+ is less than the analog voltage at VIN-, the output of the comparator is a digital low level. When the analog voltage at VIN+ is greater than the analog voltage at VIN-, the output of the comparator is a digital high level.

FIGURE 9-1:SINGLE COMPARATOR



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

FIGURE 9-2: COMPARATOR SIMPLIFIED BLOCK DIAGRAM



Note 1: When CMON = 0, the comparator will produce a '0' output to the XOR Gate.
2: Q1 and Q3 are phases of the four-phase system clock (FOSC).
3: Q1 is held high during Sleep mode.
4: Output shown for reference only. See I/O port pin diagram for more details.

9.5 Comparator Interrupt Operation

The comparator interrupt flag can be 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 9-4 and Figure 9-5). 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. At this point the two mismatch latches have opposite output levels which is detected by the exclusive-or gate and fed to the interrupt circuitry. The mismatch condition persists until either the CMCON0 register is read or the comparator output returns to the previous state.

Note 1:	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.
2:	Comparator interrupts will operate correctly regardless of the state of CMOE.

The comparator interrupt is set by the mismatch edge and not the mismatch level. This means that the interrupt flag can be reset without the additional step of reading or writing the CMCON0 register to clear the mismatch registers. When the mismatch registers are cleared, an interrupt will occur upon the comparator's return to the previous state, otherwise no interrupt will be generated.

Software will need to maintain information about the status of the comparator output, as read from the CMCON1 register, to determine the actual change that has occurred.

The CMIF bit of the PIR1 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, an interrupt can be generated.

The CMIE bit of the PIE1 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 CMIF bit of the PIR1 register will still be set if an interrupt condition occurs.

FIGURE 9-4: COMPARATOR INTERRUPT TIMING W/O CMCON0 READ

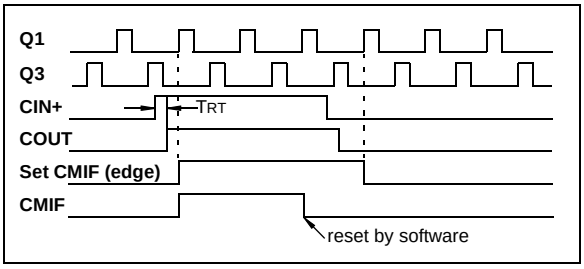
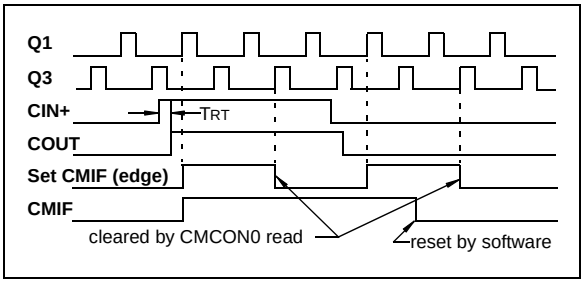


FIGURE 9-5: COMPARATOR INTERRUPT TIMING WITH CMCON0 READ



Note 1:	If a change in the CMCON0 register (COUT) should occur when a read operation is being executed (start of the Q2 cycle), then the CMIF of the PIR1 register interrupt flag may not get set.
2:	When a 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.

REGISTER 9-1: CMCON0: COMPARATOR CONTROL REGISTER 0

R/W-0	R-0	R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0
CMON	COUT	CMOE	CMPOL	—	CMR	—	CMCH
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 **CMON:** Comparator Enable bit
1 = Comparator is enabled
0 = Comparator is disabled

bit 6 **COUT:** Comparator Output bit
If C1POL = 1 (inverted polarity):
COUT = 0 when $CMV_{IN}+ > CMV_{IN}-$
COUT = 1 when $CMV_{IN}+ < CMV_{IN}-$
If C1POL = 0 (non-inverted polarity):
COUT = 1 when $CMV_{IN}+ > CMV_{IN}-$
COUT = 0 when $CMV_{IN}+ < CMV_{IN}-$

bit 5 **CMOE:** Comparator Output Enable bit
1 = COUT is present on the COUT pin(1)
0 = COUT is internal only

bit 4 **CMPOL:** Comparator Output Polarity Select bit
1 = COUT logic is inverted
0 = COUT logic is not inverted

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

bit 2 **CMR:** Comparator Reference Select bit (non-inverting input)
1 = $CMV_{IN}+$ connects to CMV_{REF} output
0 = $CMV_{IN}+$ connects to CIN+ pin

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

bit 0 **CMCH:** Comparator C1 Channel Select bit
0 = $CMV_{IN}-$ pin of the Comparator connects to CIN0-
1 = $CMV_{IN}-$ pin of the Comparator connects to CIN1-

Note 1: Comparator output requires the following three conditions: CMOE = 1, CMON = 1 and corresponding port TRIS bit = 0.

10.3 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 10-4. 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 10-4. **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 10-1 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 10-1: ACQUISITION TIME EXAMPLE

Assumptions: *Temperature = 50°C and external impedance of 10kΩ 5.0V VDD*

$$\begin{aligned} T_{ACQ} &= \text{Amplifier Settling Time} + \text{Hold Capacitor Charging Time} + \text{Temperature Coefficient} \\ &= T_{AMP} + T_C + T_{COFF} \\ &= 2\mu s + T_C + [(\text{Temperature} - 25°C)(0.05\mu s/°C)] \end{aligned}$$

The value for TC can be approximated with the following equations:

$$V_{APPLIED}\left(1 - \frac{1}{2047}\right) = V_{CHOLD} \quad ;[1]\ V_{CHOLD}\ \text{charged to within 1/2 lsb}$$

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{RC}}\right) = V_{CHOLD} \quad ;[2]\ V_{CHOLD}\ \text{charge response to}\ V_{APPLIED}$$

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{RC}}\right) = V_{APPLIED}\left(1 - \frac{1}{2047}\right) \quad ;\text{combining [1] and [2]}$$

Solving for TC:

$$\begin{aligned} T_C &= -C_{HOLD}(R_{IC} + R_{SS} + R_S)\ ln(1/2047) \\ &= -10pF(1k\Omega + 7k\Omega + 10k\Omega)\ ln(0.0004885) \\ &= 1.37\mu s \end{aligned}$$

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.

TABLE 11-2: SUMMARY OF REGISTERS ASSOCIATED WITH CAPTURE

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	P1M	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0-00 0000	0-00 0000
CCPR1L	Capture/Compare/PWM Register 1 Low Byte								xxxx xxxx	uuuu uuuu
CCPR1H	Capture/Compare/PWM Register 1 High Byte								xxxx xxxx	uuuu uuuu
INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	0000 0000
PIE1	—	ADIE(1)	CCP1IE(1)	—	CMIE	—	TMR2IE(1)	TMR1IE	-00- 0-00	-00- 0-00
PIR1	—	ADIF(1)	CCP1IF(1)	—	CMIF	—	TMR2IF(1)	TMR1IF	-00- 0-00	-00- 0-00
T1CON	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	0000 0000	uuuu uuuu
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
TRISIO	—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0	--11 1111	--11 1111

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

Note 1: For PIC12F615/617/HV615 only.

11.3 PWM Mode

The PWM mode generates a Pulse-Width Modulated signal on the CCP1 pin. The duty cycle, period and resolution are determined by the following registers:

- PR2
- T2CON
- CCPR1L
- CCP1CON

In Pulse-Width Modulation (PWM) mode, the CCP module produces up to a 10-bit resolution PWM output on the CCP1 pin. Since the CCP1 pin is multiplexed with the PORT data latch, the TRIS for that pin must be cleared to enable the CCP1 pin output driver.

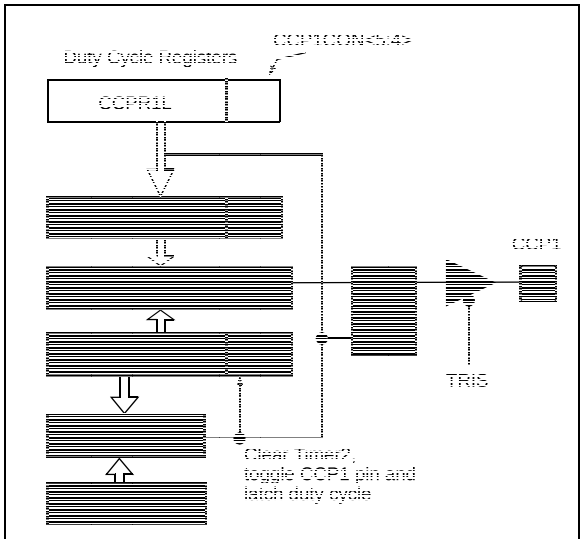
Note:	Clearing the CCP1CON register will relinquish CCP1 control of the CCP1 pin.

Figure 11-3 shows a simplified block diagram of PWM operation.

Figure 11-4 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 11.3.7 "Setup for PWM Operation"**.

FIGURE 11-3: SIMPLIFIED PWM BLOCK DIAGRAM

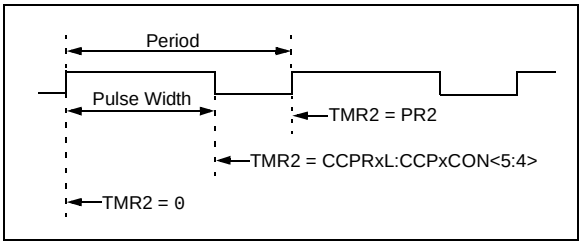


Note 1: The 8-bit timer TMR2 register is concatenated with the 2-bit internal system clock (F_{OSC}), or 2 bits of the prescaler, to create the 10-bit time base.

2: In PWM mode, CCPR1H is a read only register.

The PWM output (Figure 11-4) has a time base (period) and a time that the output stays high (duty cycle).

FIGURE 11-4: CCP PWM OUTPUT



REGISTER 12-1: CONFIG: CONFIGURATION WORD REGISTER (ADDRESS: 2007h) FOR PIC12F609/615/HV609/615 ONLY

U-1	U-1	U-1	U-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	—	—	BOREN1(1)	BOREN0(1)	IOSCFS	$\overline{CP}$(2)	MCLRE(3)	$\overline{PWRTE}$	WDTE	FOSC2	FOSC1	FOSC0
bit 13													bit 0

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

bit 13-10 **Unimplemented**: Read as ‘1’

bit 9-8 **BOREN<1:0>:** Brown-out Reset Selection bits(1)
11 = BOR enabled
10 = BOR enabled during operation and disabled in Sleep
0x = BOR disabled

bit 7 **IOSCFS:** Internal Oscillator Frequency Select bit
1 = 8 MHz
0 = 4 MHz

bit 6 **$\overline{CP}$:** Code Protection bit(2)
1 = Program memory code protection is disabled
0 = Program memory code protection is enabled

bit 5 **MCLRE:** $\overline{MCLR}$ Pin Function Select bit(3)
1 = $\overline{MCLR}$ pin function is $\overline{MCLR}$
0 = $\overline{MCLR}$ pin function is digital input, $\overline{MCLR}$ internally tied to V_{DD}

bit 4 **$\overline{PWRTE}$:** Power-up Timer Enable bit
1 = PWRT disabled
0 = PWRT enabled

bit 3 **WDTE:** Watchdog Timer Enable bit
1 = WDT enabled
0 = WDT disabled

bit 2-0 **FOSC<2:0>:** Oscillator Selection bits
111 =RC oscillator: CLKOUT function on GP4/OSC2/CLKOUT pin, RC on GP5/OSC1/CLKIN
110 =RCIO oscillator: I/O function on GP4/OSC2/CLKOUT pin, RC on GP5/OSC1/CLKIN
101 =INTOSC oscillator: CLKOUT function on GP4/OSC2/CLKOUT pin, I/O function on GP5/OSC1/CLKIN
100 = INTOSCIO oscillator: I/O function on GP4/OSC2/CLKOUT pin, I/O function on GP5/OSC1/CLKIN
011 =EC: I/O function on GP4/OSC2/CLKOUT pin, CLKIN on GP5/OSC1/CLKIN
010 =HS oscillator: High-speed crystal/resonator on GP4/OSC2/CLKOUT and GP5/OSC1/CLKIN
001 = XT oscillator: Crystal/resonator on GP4/OSC2/CLKOUT and GP5/OSC1/CLKIN
000 = LP oscillator: Low-power crystal on GP4/OSC2/CLKOUT and GP5/OSC1/CLKIN

Note 1: Enabling Brown-out Reset does not automatically enable Power-up Timer.
2: The entire program memory will be erased when the code protection is turned off.
3: When $\overline{MCLR}$ is asserted in INTOSC or RC mode, the internal clock oscillator is disabled.

REGISTER 12-2: CONFIG – CONFIGURATION WORD (ADDRESS: 2007h) FOR PIC12F617 ONLY

U-1	U-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	—	WRT1	WRT0	BOREN1	BOREN0	IOSCFS	$\overline{CP}$	MCLRE	$\overline{PWRTE}$	WDTE	FOSC2	F0SC1	F0SC0
bit 13													bit 0

bit 13-12 **Unimplemented:** Read as '1'

bit 11-10 **WRT<1:0>:** Flash Program Memory Self Write Enable bits
11 = Write protection off
10 = 000h to 1FFh write protected, 200h to 7FFh may be modified by PMCON1 control
01 = 000h to 3FFh write protected, 400h to 7FFh may be modified by PMCON1 control
00 = 000h to 7FFh write protected, entire program memory is write protected.

bit 9-8 **BOREN<1:0>:** Brown-out Reset Enable bits
11 = BOR enabled
10 = BOR disabled during Sleep and enabled during operation
0X = BOR disabled

bit 7 **IOSCFS:** Internal Oscillator Frequency Select
1 = 8 MHz
0 = 4 MHz

bit 6 **$\overline{CP}$:** Code Protection
1 = Program memory is not code protected
0 = Program memory is external read and write protected

bit 5 **MCLRE:** $\overline{MCLR}$ Pin Function Select
1 = $\overline{MCLR}$ pin is $\overline{MCLR}$ function and weak internal pull-up is enabled
0 = $\overline{MCLR}$ pin is alternate function, $\overline{MCLR}$ function is internally disabled

bit 4 **$\overline{PWRTE}$:** Power-up Timer Enable bit(1)
1 = PWRT disabled
0 = PWRT enabled

bit 3 **WDTE:** Watchdog Timer Enable bit
1 = WDT enabled
0 = WDT disabled

bit 2-0 **FOSC<2:0>:** Oscillator Selection bits
000 =LP oscillator: Low-power crystal on RA5/T1CKI/OSC1/CLKIN and RA4/AN3/$\overline{T1G}$/OSC2/CLKOUT
001 =XT oscillator: Crystal/resonator on RA5/T1CKI/OSC1/CLKIN and RA4/AN3/$\overline{T1G}$/OSC2/CLKOUT
010 =HS oscillator: High-speed crystal/resonator on RA5/T1CKI/OSC1/CLKIN and RA4/AN3/$\overline{T1G}$/OSC2/CLKOUT
011 =EC: I/O function on RA4/AN3/$\overline{T1G}$/OSC2/CLKOUT, CLKIN on RA5/T1CKI/OSC1/CLKIN
100 =INTOSCIO oscillator: I/O function on RA4/AN3/$\overline{T1G}$/OSC2/CLKOUT, I/O function on RA5/T1CKI/OSC1/CLKIN
101 =INTOSC oscillator: CLKOUT function on RA4/AN3/$\overline{T1G}$/OSC2/CLKOUT, I/O function on RA5/T1CKI/OSC1/CLKIN
110 =EXTRCIO oscillator: I/O function on RA4/AN3/$\overline{T1G}$/OSC2/CLKOUT, RC on RA5/T1CKI/OSC1/CLKIN
111 =EXTRC oscillator: CLKOUT function on RA4/AN3/$\overline{T1G}$/OSC2/CLKOUT, RC on RA5/T1CKI/OSC1/CLKIN

Note 1: Enabling Brown-out Reset does not automatically enable the Power-up Timer (PWRT).

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

12.4.2 TIMER0 INTERRUPT

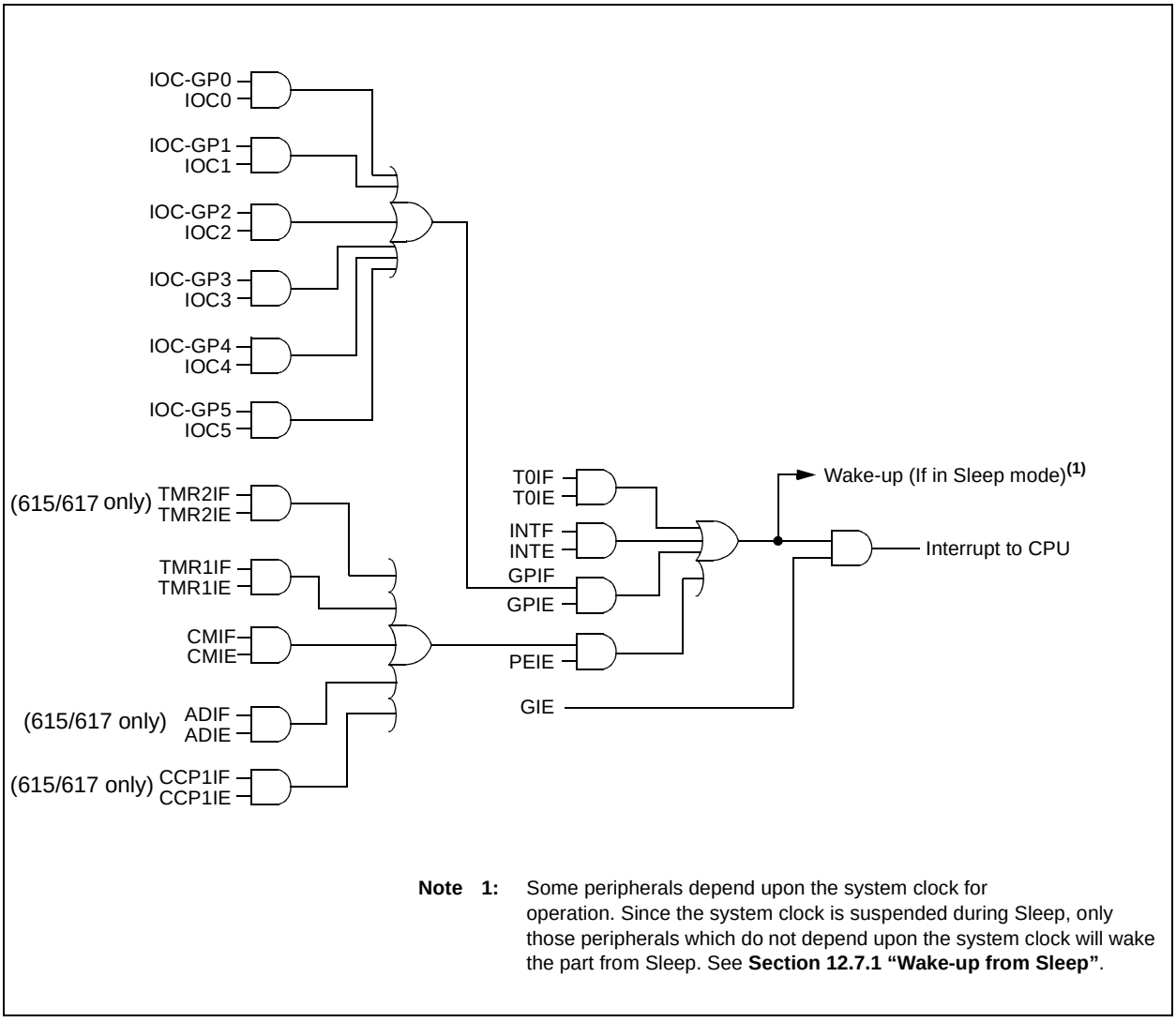
An overflow (FFh → 00h) in the TMR0 register will set the T0IF bit of the INTCON register. The interrupt can be enabled/disabled by setting/clearing T0IE bit of the INTCON register. See **Section 6.0 "Timer0 Module"** for operation of the Timer0 module.

12.4.3 GPIO INTERRUPT-ON-CHANGE

An input change on GPIO sets the GPIF bit of the INTCON register. The interrupt can be enabled/disabled by setting/clearing the GPIE bit of the INTCON register. Plus, individual pins can be configured through the IOC register.

Note:	If a change on the I/O pin should occur when any GPIO operation is being executed, then the GPIF interrupt flag may not get set.

FIGURE 12-7: INTERRUPT LOGIC



Note 1: Some peripherals depend upon the system clock for operation. Since the system clock is suspended during Sleep, only those peripherals which do not depend upon the system clock will wake the part from Sleep. See **Section 12.7.1 "Wake-up from Sleep"**.

15.11 PICkit 2 Development Programmer/Debugger and PICkit 2 Debug Express

The PICkit™ 2 Development Programmer/Debugger is a low-cost development tool with an easy to use interface for programming and debugging Microchip's Flash families of microcontrollers. The full featured Windows® programming interface supports baseline (PIC10F, PIC12F5xx, PIC16F5xx), midrange (PIC12F6xx, PIC16F), PIC18F, PIC24, dsPIC30, dsPIC33, and PIC32 families of 8-bit, 16-bit, and 32-bit microcontrollers, and many Microchip Serial EEPROM products. With Microchip's powerful MPLAB Integrated Development Environment (IDE) the PICkit™ 2 enables in-circuit debugging on most PIC® microcontrollers. In-Circuit-Debugging runs, halts and single steps the program while the PIC microcontroller is embedded in the application. When halted at a breakpoint, the file registers can be examined and modified.

The PICkit 2 Debug Express include the PICkit 2, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

15.12 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 MMC card for file storage and data applications.

15.13 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

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.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

16.9 Thermal Considerations

Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C					
Param No.	**Sym**	**Characteristic**	**Typ**	**Units**	**Conditions**
TH01	θJA	Thermal Resistance Junction to Ambient	84.6*	C/W	8-pin PDIP package
			149.5*	C/W	8-pin SOIC package
			211*	C/W	8-pin MSOP package
			60*	C/W	8-pin DFN 3x3mm package
			44*	C/W	8-pin DFN 4x4mm package
TH02	θJC	Thermal Resistance Junction to Case	41.2*	C/W	8-pin PDIP package
			39.9*	C/W	8-pin SOIC package
			39*	C/W	8-pin MSOP package
			9*	C/W	8-pin DFN 3x3mm package
			3.0*	C/W	8-pin DFN 4x4mm package
TH03	TDIE	Die Temperature	150*	C	
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 = PDMAX (TDIE - TA)/θJA **(NOTE 2)**

* These parameters are characterized but not tested.

Note 1: IDD is current to run the chip alone without driving any load on the output pins.

2: T_A = Ambient temperature.

FIGURE 17-41: MAXIMUM HFINTOSC START-UP TIMES vs. VDD OVER TEMPERATURE

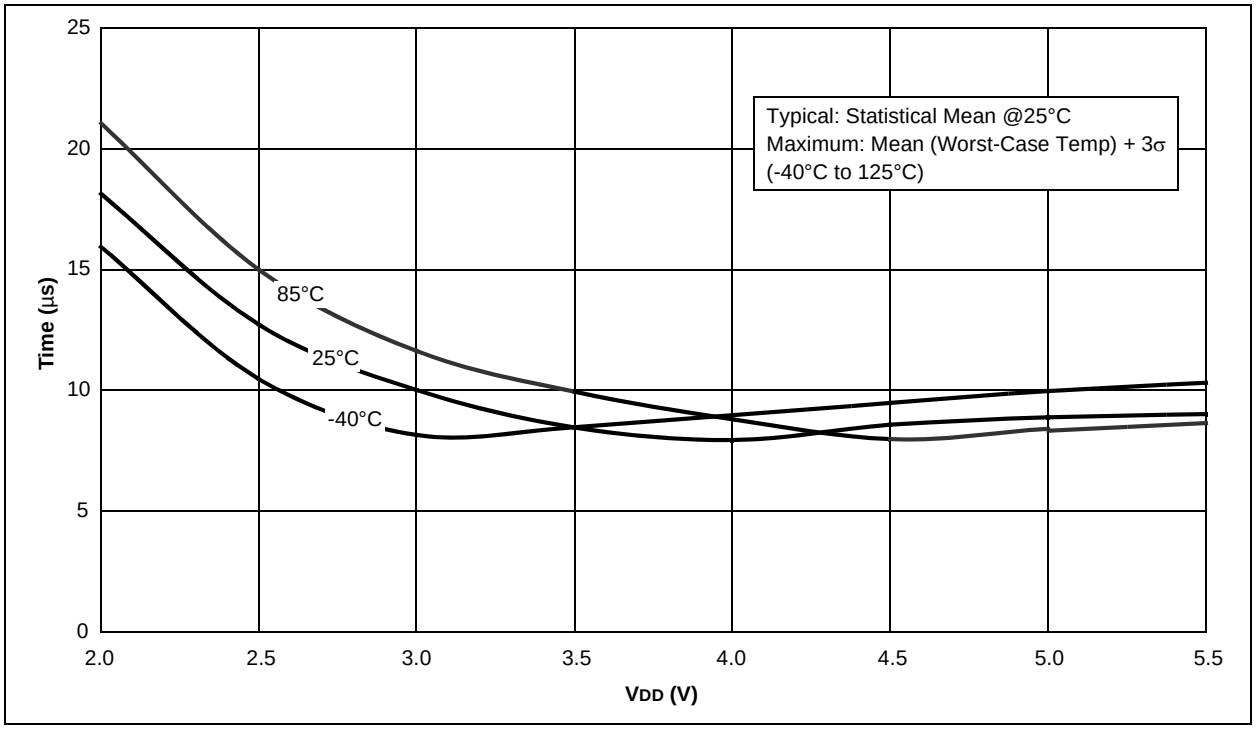
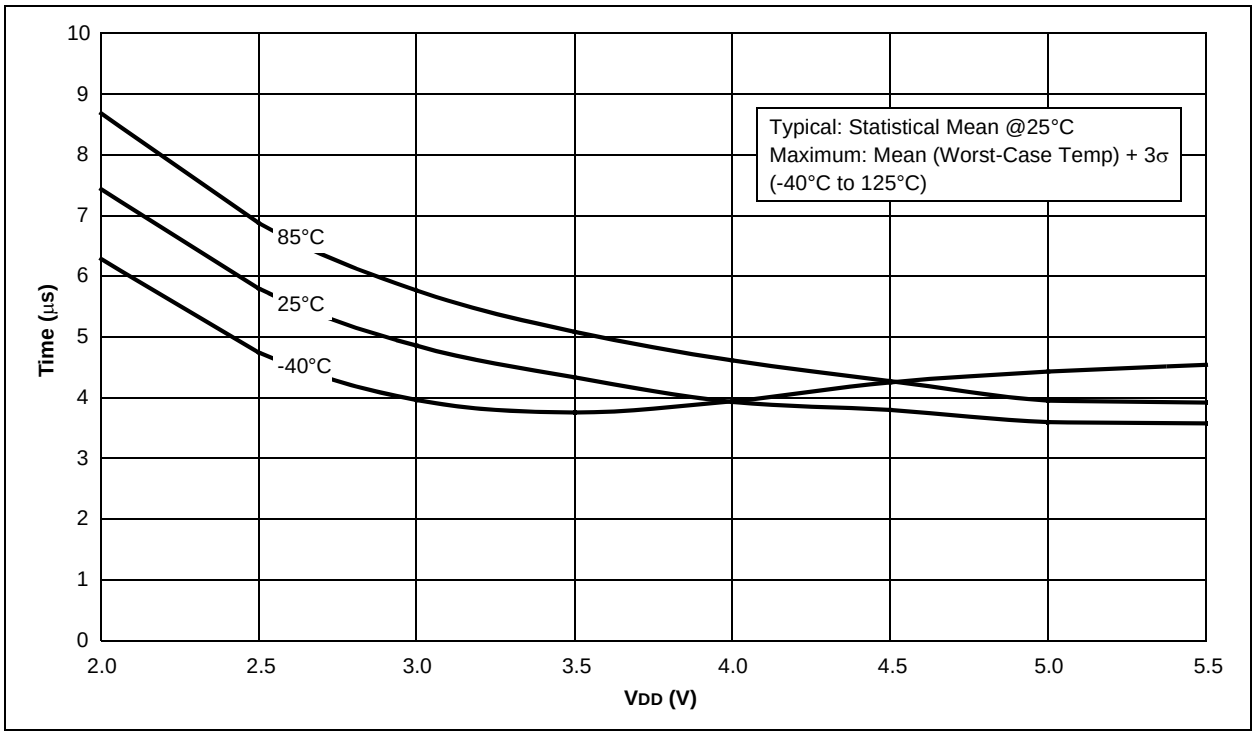


FIGURE 17-42: MINIMUM HFINTOSC START-UP TIMES vs. VDD OVER TEMPERATURE



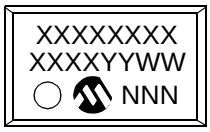
18.0 PACKAGING INFORMATION

18.1 Package Marking Information

8-Lead PDIP (.300”)

Example

8-Lead SOIC (.150”)



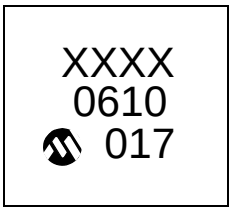
Example

8-Lead MSOP

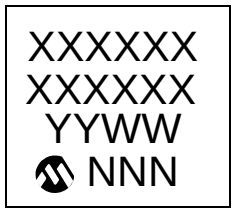
Example

8-Lead DFN (3x3 mm)

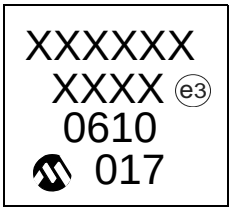
Example



8-Lead DFN (4x4 mm) (for PIC12F609/615/HV609/615 devices only)



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.