



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	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 ~ 5V
Data Converters	A/D 4x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	8-VDFN Exposed Pad
Supplier Device Package	8-DFN (3x3)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic12hv615-i-mf

6.0 TIMER0 MODULE

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

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

Figure 6-1 is a block diagram of the Timer0 module.

6.1 Timer0 Operation

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

6.1.1 8-BIT TIMER MODE

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

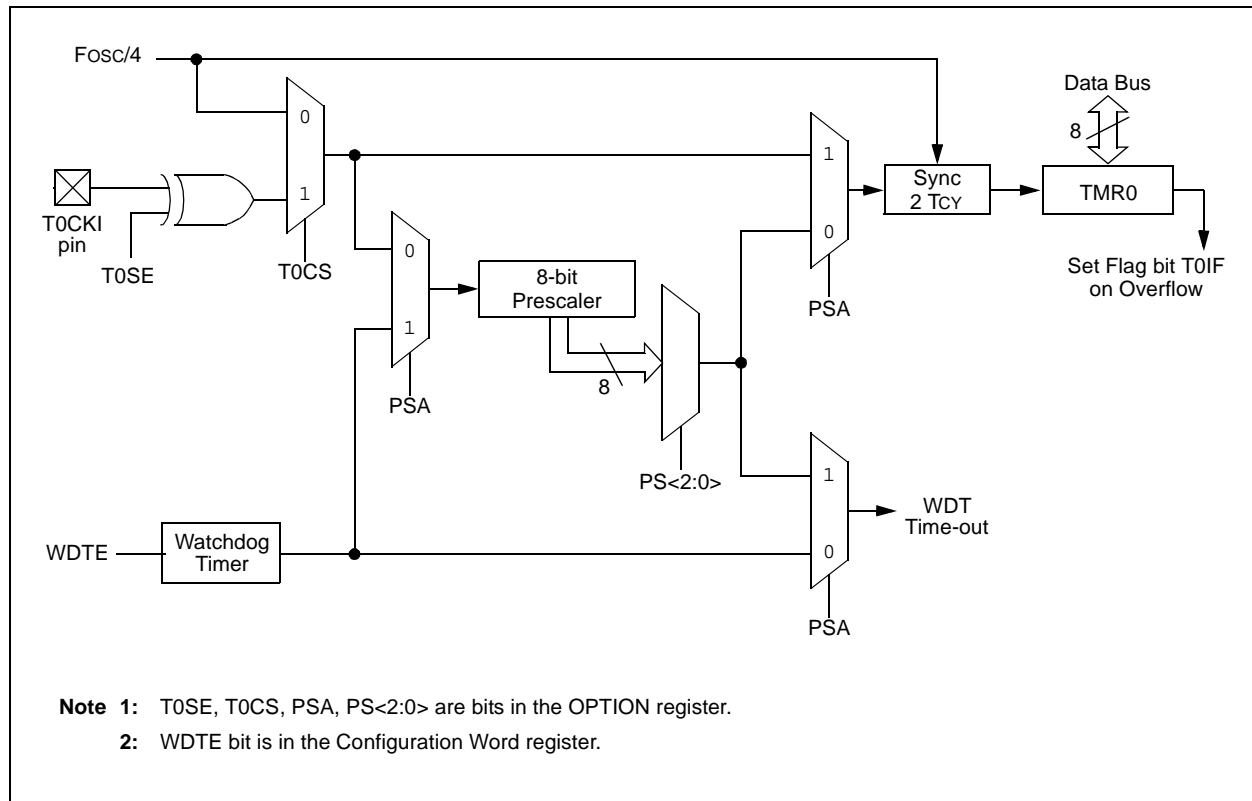
When TMR0 is written, the increment is inhibited for two instruction cycles immediately following the write.

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

6.1.2 8-BIT COUNTER MODE

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

FIGURE 6-1: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER



PIC12F609/615/617/12HV609/615

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
CLRWF      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”**.

PIC12F609/615/617/12HV609/615

REGISTER 9-3: VRCON: VOLTAGE REFERENCE CONTROL REGISTER

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

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7 **CMVREN:** Comparator Voltage Reference Enable bit^(1, 2)

1 = CVREF circuit powered on and routed to CVREF input of the Comparator

0 = 0.6 Volt constant reference routed to CVREF input of the Comparator

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

bit 5 **VRR:** CVREF Range Selection bit

1 = Low range

0 = High range

bit 4 **FVREN:** 0.6V Reference Enable bit⁽²⁾

1 = Enabled

0 = Disabled

bit 3-0 **VR<3:0>:** Comparator Voltage Reference CVREF Value Selection bits ($0 \leq VR<3:0> \leq 15$)

When VRR = 1: $CVREF = (VR<3:0>/24) * VDD$

When VRR = 0: $CVREF = VDD/4 + (VR<3:0>/32) * VDD$

Note 1: When CMVREN is low, the CVREF circuit is powered down and does not contribute to IDD current.

2: When CMVREN is low and the FVREN bit is low, the CVREF signal should provide Vss to the comparator.

PIC12F609/615/617/12HV609/615

REGISTER 10-2: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 0 (READ-ONLY)

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
ADRES9	ADRES8	ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 **ADRES<9:2>**: ADC Result Register bits
Upper 8 bits of 10-bit conversion result

REGISTER 10-3: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 0 (READ-ONLY)

R-x	R-x	U-0	U-0	U-0	U-0	U-0	U-0
ADRES1	ADRES0	—	—	—	—	—	—
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-6 **ADRES<1:0>**: ADC Result Register bits
Lower 2 bits of 10-bit conversion result

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

REGISTER 10-4: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 1 (READ-ONLY)

U-0	U-0	U-0	U-0	U-0	U-0	R-x	R-x
—	—	—	—	—	—	ADRES9	ADRES8
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

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

bit 1-0 **ADRES<9:8>**: ADC Result Register bits
Upper 2 bits of 10-bit conversion result

REGISTER 10-5: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 1 (READ-ONLY)

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2	ADRES1	ADRES0
bit 7							bit 0

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

bit 7-0 **ADRES<7:0>**: ADC Result Register bits
Lower 8 bits of 10-bit conversion result

PIC12F609/615/617/12HV609/615

FIGURE 11-6: EXAMPLE PWM (ENHANCED MODE) OUTPUT RELATIONSHIPS (ACTIVE-HIGH STATE)

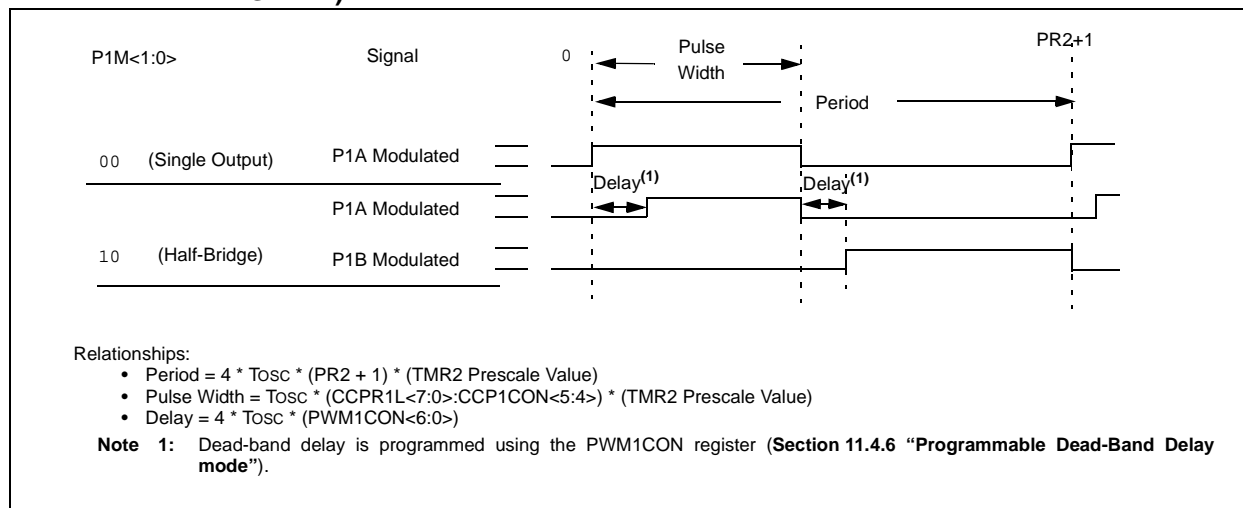
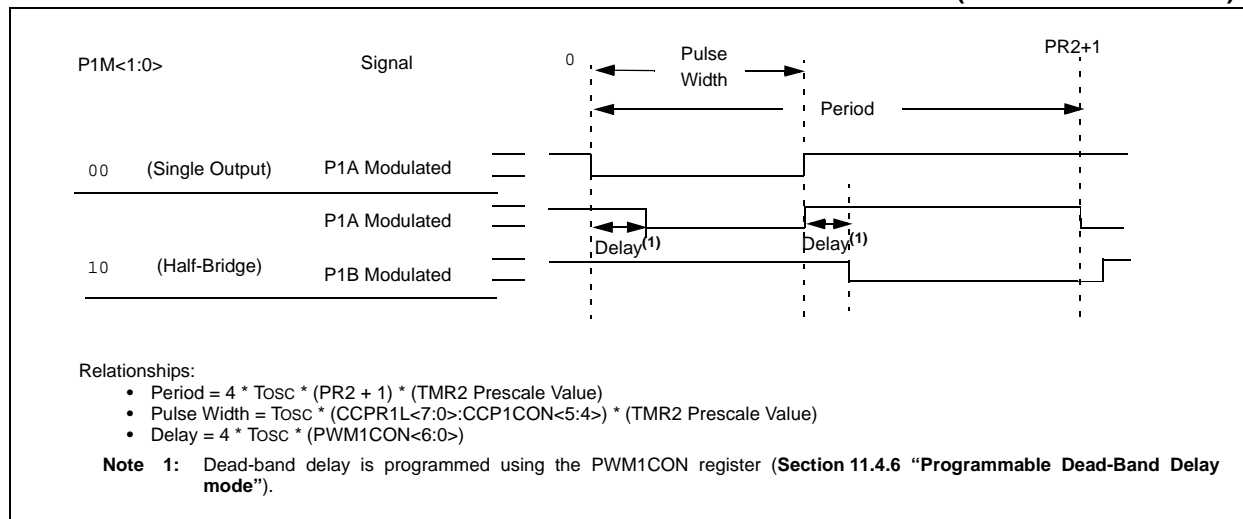


FIGURE 11-7: EXAMPLE ENHANCED PWM OUTPUT RELATIONSHIPS (ACTIVE-LOW STATE)



PIC12F609/615/617/12HV609/615

REGISTER 11-3: PWM1CON: ENHANCED PWM 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
PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0
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 **PRSEN:** PWM Restart Enable bit
 1 = Upon auto-shutdown, the ECCPASE bit clears automatically once the shutdown event goes away; the PWM restarts automatically
 0 = Upon auto-shutdown, ECCPASE must be cleared in software to restart the PWM

bit 6-0 **PDC<6:0>:** PWM Delay Count bits
 PDCn = Number of Fosc/4 (4 * Tosc) cycles between the scheduled time when a PWM signal **should** transition active and the **actual** time it transitions active

TABLE 11-7: SUMMARY OF REGISTERS ASSOCIATED WITH PWM

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
APFCON	—	—	—	T1GSEL	—	—	P1BSEL	P1ASEL	---0 --00	---0 --00
CCP1CON ⁽¹⁾	P1M	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0-00 0000	0-00 0000
CCPR1L ⁽¹⁾	Capture/Compare/PWM Register 1 Low Byte								xxxx xxxx	uuuu uuuu
CCPR1H ⁽¹⁾	Capture/Compare/PWM Register 1 High Byte								xxxx xxxx	uuuu uuuu
CMCON0	CMON	COUT	CMOE	CMPOL	—	CMR	—	CMCH	0000 -0-0	0000 -0-0
CMCON1	—	—	—	T1ACS	CMHYS	—	T1GSS	CMSYNC	---0 0-10	---0 0-10
ECCPAS ⁽¹⁾	ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0	0000 0000	0000 0000
PWM1CON	PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0	0000 0000	0000 0000
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
T2CON ⁽¹⁾	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
TMR2 ⁽¹⁾	Timer2 Module Register								0000 0000	0000 0000
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 PWM.

Note 1: For PIC12F615/617/HV615 only.

PIC12F609/615/617/12HV609/615

TABLE 12-4: INITIALIZATION CONDITION FOR REGISTERS (PIC12F609/HV609)

Register	Address	Power-on Reset	MCLR Reset WDT Reset Brown-out Reset ⁽¹⁾	Wake-up from Sleep through Interrupt Wake-up from Sleep through WDT Time-out
W	—	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	00h/80h	xxxx xxxx	xxxx xxxx	uuuu uuuu
TMR0	01h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02h/82h	0000 0000	0000 0000	PC + 1 ⁽³⁾
STATUS	03h/83h	0001 1xxx	000q quuu ⁽⁴⁾	uuuq quuu ⁽⁴⁾
FSR	04h/84h	xxxx xxxx	uuuu uuuu	uuuu uuuu
GPIO	05h	--x0 x000	--u0 u000	--uu uuuu
PCLATH	0Ah/8Ah	---0 0000	---0 0000	---u uuuu
INTCON	0Bh/8Bh	0000 0000	0000 0000	uuuu uuuu ⁽²⁾
PIR1	0Ch	----- 0--0	---- 0--0	---- u--u ⁽²⁾
TMR1L	0Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	0Fh	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	10h	0000 0000	uuuu uuuu	-uuu uuuu
VRCON	19h	0-00 0000	0-00 0000	u-uu uuuu
CMCON0	1Ah	0000 -0-0	0000 -0-0	uuuu -u-u
CMCON1	1Ch	---0 0-10	---0 0-10	---u u-qu
OPTION_REG	81h	1111 1111	1111 1111	uuuu uuuu
TRISIO	85h	--11 1111	--11 1111	--uu uuuu
PIE1	8Ch	----- 0--0	---- 0--0	---- u--u
PCON	8Eh	---- --0x	---- --uu ^(1, 5)	---- --uu
OSCTUNE	90h	---0 0000	---u uuuu	---u uuuu
WPU	95h	--11 -111	--11 -111	--uu -uuu
IOC	96h	--00 0000	--00 0000	--uu uuuu
ANSEL	9Fh	---- 1-11	---- 1-11	---- q-qq

Legend: u = unchanged, x = unknown, – = unimplemented bit, reads as '0', q = value depends on condition.

Note 1: If VDD goes too low, Power-on Reset will be activated and registers will be affected differently.

2: One or more bits in INTCON and/or PIR1 will be affected (to cause wake-up).

3: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

4: See Table 12-6 for Reset value for specific condition.

5: If Reset was due to brown-out, then bit 0 = 0. All other Resets will cause bit 0 = u.

PIC12F609/615/617/12HV609/615

TABLE 12-5: INITIALIZATION CONDITION FOR REGISTERS (PIC12F615/617/HV615)

Register	Address	Power-on Reset	MCLR Reset WDT Reset Brown-out Reset ⁽¹⁾	Wake-up from Sleep through Interrupt Wake-up from Sleep through WDT Time-out
W	—	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	00h/80h	xxxx xxxx	xxxx xxxx	uuuu uuuu
TMR0	01h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02h/82h	0000 0000	0000 0000	PC + 1 ⁽³⁾
STATUS	03h/83h	0001 1xxx	000q quuu ⁽⁴⁾	uuuq quuu ⁽⁴⁾
FSR	04h/84h	xxxx xxxx	uuuu uuuu	uuuu uuuu
GPIO	05h	--x0 x000	--u0 u000	--uu uuuu
PCLATH	0Ah/8Ah	---0 0000	---0 0000	---u uuuu
INTCON	0Bh/8Bh	0000 0000	0000 0000	uuuu uuuu ⁽²⁾
PIR1	0Ch	-000 0-00	-000 0-00	-uuu u-uu ⁽²⁾
TMR1L	0Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	0Fh	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	10h	0000 0000	uuuu uuuu	-uuu uuuu
TMR2 ⁽¹⁾	11h	0000 0000	0000 0000	uuuu uuuu
T2CON ⁽¹⁾	12h	-000 0000	-000 0000	-uuu uuuu
CCPR1L ⁽¹⁾	13h	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR1H ⁽¹⁾	14h	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP1CON ⁽¹⁾	15h	0-00 0000	0-00 0000	u-uu uuuu
PWM1CON ⁽¹⁾	16h	0000 0000	0000 0000	uuuu uuuu
ECCPAS ⁽¹⁾	17h	0000 0000	0000 0000	uuuu uuuu
VRCON	19h	0-00 0000	0-00 0000	u-uu uuuu
CMCON0	1Ah	0000 -0-0	0000 -0-0	uuuu -u-u
CMCON1	1Ch	---0 0-10	---0 0-10	---u u-qu
ADRESH ⁽¹⁾	1Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON0 ⁽¹⁾	1Fh	00-0 0000	00-0 0000	uu-u uuuu
OPTION_REG	81h	1111 1111	1111 1111	uuuu uuuu
TRISIO	85h	--11 1111	--11 1111	--uu uuuu
PIE1	8Ch	-00- 0-00	-00- 0-00	-uu- u-uu
PCON	8Eh	---- --0x	---- --uu ^(1, 5)	---- --uu
OSCTUNE	90h	---0 0000	---u uuuu	---u uuuu
PR2	92h	1111 1111	1111 1111	1111 1111
APFCON	93h	---0 --00	---0 --00	---u --uu
WPU	95h	--11 -111	--11 -111	--uu -uuu
IOC	96h	--00 0000	--00 0000	--uu uuuu
PMCON1 ⁽⁶⁾	98h	---- -000	---- -000	---- -uuu
PMCON2 ⁽⁶⁾	99h	---- ----	---- ----	---- ----
PMADRL ⁽⁶⁾	9Ah	0000 0000	0000 0000	uuuu uuuu

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as '0', q = value depends on condition.

Note 1: If VDD goes too low, Power-on Reset will be activated and registers will be affected differently.

2: One or more bits in INTCON and/or PIR1 will be affected (to cause wake-up).

3: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

4: See Table 12-6 for Reset value for specific condition.

5: If Reset was due to brown-out, then bit 0 = 0. All other Resets will cause bit 0 = u.

6: For PIC12F617 only.

12.7 Power-Down Mode (Sleep)

The Power-Down mode is entered by executing a `SLEEP` instruction.

If the Watchdog Timer is enabled:

- WDT will be cleared but keeps running.
- \overline{PD} bit in the STATUS register is cleared.
- \overline{TO} bit is set.
- Oscillator driver is turned off.
- I/O ports maintain the status they had before `SLEEP` was executed (driving high, low or high-impedance).

For lowest current consumption in this mode, all I/O pins should be either at V_{DD} or V_{SS} , with no external circuitry drawing current from the I/O pin and the comparators and $CVREF$ should be disabled. I/O pins that are high-impedance inputs should be pulled high or low externally to avoid switching currents caused by floating inputs. The $T0CKI$ input should also be at V_{DD} or V_{SS} for lowest current consumption. The contribution from on-chip pull-ups on GPIO should be considered.

The \overline{MCLR} pin must be at a logic high level.

Note: It should be noted that a Reset generated by a WDT time-out does not drive \overline{MCLR} pin low.

12.7.1 WAKE-UP FROM SLEEP

The device can wake-up from Sleep through one of the following events:

1. External Reset input on \overline{MCLR} pin.
2. Watchdog Timer wake-up (if WDT was enabled).
3. Interrupt from GP2/INT pin, GPIO change or a peripheral interrupt.

The first event will cause a device Reset. The two latter events are considered a continuation of program execution. The \overline{TO} and \overline{PD} bits in the STATUS register can be used to determine the cause of device Reset. The \overline{PD} bit, which is set on power-up, is cleared when Sleep is invoked. \overline{TO} bit is cleared if WDT wake-up occurred.

The following peripheral interrupts can wake the device from Sleep:

1. Timer1 interrupt. Timer1 must be operating as an asynchronous counter.
2. ECCP Capture mode interrupt.
3. A/D conversion (when A/D clock source is RC).
4. Comparator output changes state.
5. Interrupt-on-change.
6. External Interrupt from INT pin.

Other peripherals cannot generate interrupts since during Sleep, no on-chip clocks are present.

When the `SLEEP` instruction is being executed, the next instruction ($PC + 1$) is prefetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be set (enabled). Wake-up is regardless of the state of the GIE bit. If the GIE bit is clear (disabled), the device continues execution at the instruction after the `SLEEP` instruction. If the GIE bit is set (enabled), the device executes the instruction after the `SLEEP` instruction, then branches to the interrupt address (0004h). In cases where the execution of the instruction following `SLEEP` is not desirable, the user should have a `NOP` after the `SLEEP` instruction.

Note: If the global interrupts are disabled (GIE is cleared) and any interrupt source has both its interrupt enable bit and the corresponding interrupt flag bits set, the device will immediately wake-up from Sleep.

The WDT is cleared when the device wakes up from Sleep, regardless of the source of wake-up.

12.7.2 WAKE-UP USING INTERRUPTS

When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

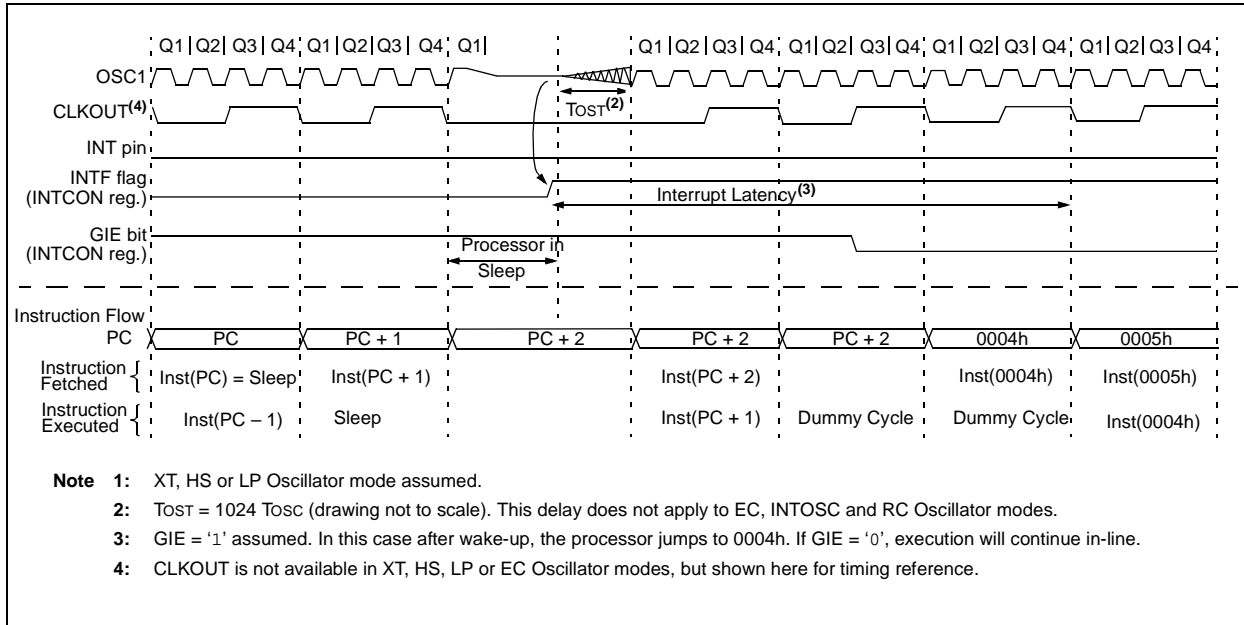
- If the interrupt occurs **before** the execution of a `SLEEP` instruction, the `SLEEP` instruction will complete as a `NOP`. Therefore, the WDT and WDT prescaler and postscaler (if enabled) will not be cleared, the \overline{TO} bit will not be set and the \overline{PD} bit will not be cleared.
- If the interrupt occurs **during or after** the execution of a `SLEEP` instruction, the device will immediately wake-up from Sleep. The `SLEEP` instruction is executed. Therefore, the WDT and WDT prescaler and postscaler (if enabled) will be cleared, the \overline{TO} bit will be set and the \overline{PD} bit will be cleared.

Even if the flag bits were checked before executing a `SLEEP` instruction, it may be possible for flag bits to become set before the `SLEEP` instruction completes. To determine whether a `SLEEP` instruction executed, test the \overline{PD} bit. If the \overline{PD} bit is set, the `SLEEP` instruction was executed as a `NOP`.

To ensure that the WDT is cleared, a `CLRWDT` instruction should be executed before a `SLEEP` instruction. See Figure 12-9 for more details.

PIC12F609/615/617/12HV609/615

FIGURE 12-9: WAKE-UP FROM SLEEP THROUGH INTERRUPT



12.8 Code Protection

If the code protection bit(s) have not been programmed, the on-chip program memory can be read out using ICSP™ for verification purposes.

Note: The entire Flash program memory will be erased when the code protection is turned off. See the *Memory Programming Specification* (DS41204) for more information.

12.9 ID Locations

Four memory locations (2000h-2003h) are designated as ID locations where the user can store checksum or other code identification numbers. These locations are not accessible during normal execution but are readable and writable during Program/Verify mode. Only the Least Significant 7 bits of the ID locations are used.

PIC12F609/615/617/12HV609/615

14.0 INSTRUCTION SET SUMMARY

The PIC12F609/615/617/12HV609/615 instruction set is highly orthogonal and is comprised of three basic categories:

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

Each PIC16 instruction is a 14-bit word divided into an **opcode**, which specifies the instruction type and one or more **operands**, which further specify the operation of the instruction. The formats for each of the categories is presented in Figure 14-1, while the various opcode fields are summarized in Table 14-1.

Table 14-2 lists the instructions recognized by the MPASM™ assembler.

For **byte-oriented** instructions, 'f' represents a file register designator and 'd' represents a destination designator. The file register designator specifies which file register is to be used by the instruction.

The destination designator specifies where the result of the operation is to be placed. If 'd' is zero, the result is placed in the W register. If 'd' is one, the result is placed in the file register specified in the instruction.

For **bit-oriented** instructions, 'b' represents a bit field designator, which selects the bit affected by the operation, while 'f' represents the address of the file in which the bit is located.

For **literal and control** operations, 'k' represents an 8-bit or 11-bit constant, or literal value.

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

All instruction examples use the format '0xhh' to represent a hexadecimal number, where 'h' signifies a hexadecimal digit.

14.1 Read-Modify-Write Operations

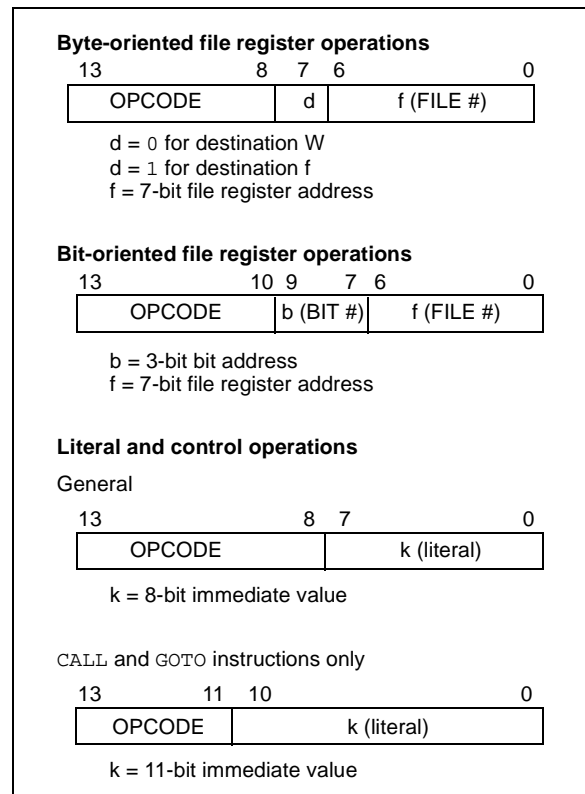
Any instruction that specifies a file register as part of the instruction performs a Read-Modify-Write (RMW) operation. The register is read, the data is modified, and the result is stored according to either the instruction or the destination designator 'd'. A read operation is performed on a register even if the instruction writes to that register.

For example, a CLRF GPIO instruction will read GPIO, clear all the data bits, then write the result back to GPIO. This example would have the unintended consequence of clearing the condition that set the GPIF flag.

TABLE 14-1: OPCODE FIELD DESCRIPTIONS

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1). The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1.
PC	Program Counter
TO	Time-out bit
C	Carry bit
DC	Digit carry bit
Z	Zero bit
PD	Power-down bit

FIGURE 14-1: GENERAL FORMAT FOR INSTRUCTIONS



PIC12F609/615/617/12HV609/615

RLF Rotate Left f through Carry

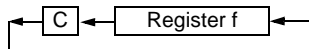
Syntax: [label] RLF f,d

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: See description below

Status Affected: C

Description: The contents of register 'f' are rotated one bit to the left through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is stored back in register 'f'.



Words: 1

Cycles: 1

Example: RLF REG1,0

Before Instruction

REG1 = 1110 0110
C = 0

After Instruction

REG1 = 1110 0110
W = 1100 1100
C = 1

SLEEP Enter Sleep mode

Syntax: [label] SLEEP

Operands: None

Operation: 00h → WDT,
0 → WDT prescaler,
1 → \overline{TO} ,
0 → \overline{PD}

Status Affected: \overline{TO} , \overline{PD}

Description: The power-down Status bit, \overline{PD} is cleared. Time-out Status bit, \overline{TO} is set. Watchdog Timer and its prescaler are cleared. The processor is put into Sleep mode with the oscillator stopped.

RRF Rotate Right f through Carry

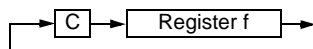
Syntax: [label] RRF f,d

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: See description below

Status Affected: C

Description: The contents of register 'f' are rotated one bit to the right through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.



SUBLW Subtract W from literal

Syntax: [label] SUBLW k

Operands: $0 \leq k \leq 255$

Operation: $k - (W) \rightarrow (W)$

Status Affected: C, DC, Z

Description: The W register is subtracted (2's complement method) from the eight-bit literal 'k'. The result is placed in the W register.

Result	Condition
C = 0	$W > k$
C = 1	$W \leq k$
DC = 0	$W<3:0> > k<3:0>$
DC = 1	$W<3:0> \leq k<3:0>$

15.2 MPLAB C Compilers for Various Device Families

The MPLAB C Compiler code development systems are complete ANSI C compilers for Microchip's PIC18, PIC24 and PIC32 families of microcontrollers and the dsPIC30 and dsPIC33 families of digital signal controllers. These compilers provide powerful integration capabilities, superior code optimization and ease of use.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

15.3 HI-TECH C for Various Device Families

The HI-TECH C Compiler code development systems are complete ANSI C compilers for Microchip's PIC family of microcontrollers and the dsPIC family of digital signal controllers. These compilers provide powerful integration capabilities, omniscient code generation and ease of use.

For easy source level debugging, the compilers provide symbol information that is optimized to the MPLAB IDE debugger.

The compilers include a macro assembler, linker, pre-processor, and one-step driver, and can run on multiple platforms.

15.4 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multi-purpose source files
- Directives that allow complete control over the assembly process

15.5 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler and the MPLAB C18 C Compiler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

15.6 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC devices. MPLAB C Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command line interface
- Rich directive set
- Flexible macro language
- MPLAB IDE compatibility

PIC12F609/615/617/12HV609/615

16.11 AC Characteristics: PIC12F609/615/617/12HV609/615 (Industrial, Extended)

FIGURE 16-4: CLOCK TIMING

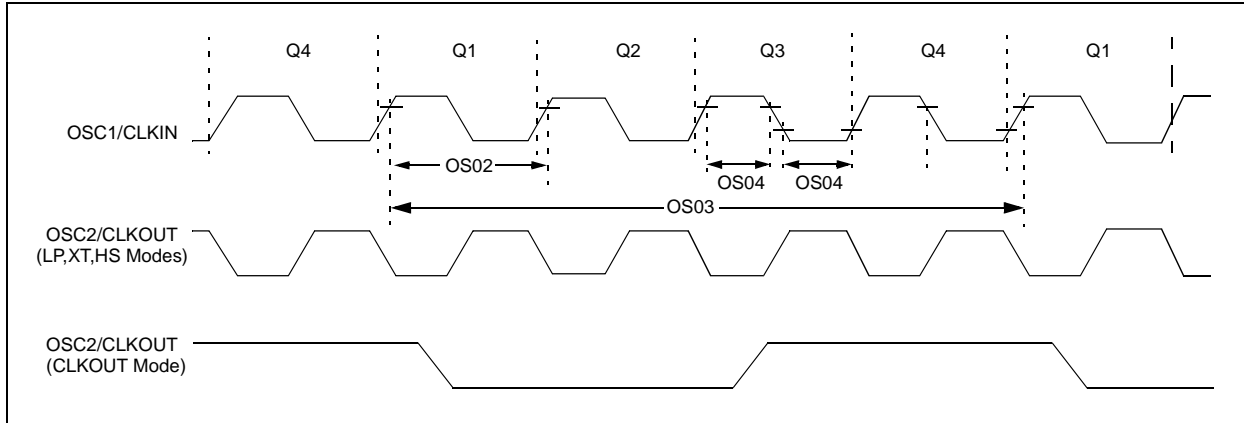


TABLE 16-1: CLOCK OSCILLATOR TIMING REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)							
Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$							
Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
OS01	Fosc	External CLKIN Frequency ⁽¹⁾	DC	—	37	kHz	LP Oscillator mode
			DC	—	4	MHz	XT Oscillator mode
			DC	—	20	MHz	HS Oscillator mode
			DC	—	20	MHz	EC Oscillator mode
		Oscillator Frequency ⁽¹⁾	—	32.768	—	kHz	LP Oscillator mode
OS02	Tosc	External CLKIN Period ⁽¹⁾	0.1	—	4	MHz	XT Oscillator mode
			1	—	20	MHz	HS Oscillator mode
			DC	—	4	MHz	RC Oscillator mode
			DC	—	4	MHz	RC Oscillator mode
		Oscillator Period ⁽¹⁾	—	30.5	—	μs	LP Oscillator mode
OS03	Tcy	Instruction Cycle Time ⁽¹⁾	250	—	10,000	ns	XT Oscillator mode
			50	—	1,000	ns	HS Oscillator mode
			250	—	—	ns	RC Oscillator mode
			250	—	—	ns	RC Oscillator mode
		Instruction Cycle Time ⁽¹⁾	—	Tcy	DC	ns	Tcy = 4/Fosc
OS04*	TosH, TosL	External CLKIN High, External CLKIN Low	2	—	—	μs	LP oscillator
			100	—	—	ns	XT oscillator
			20	—	—	ns	HS oscillator
OS05*	TosR, TosF	External CLKIN Rise, External CLKIN Fall	0	—	∞	ns	LP oscillator
			0	—	∞	ns	XT oscillator
			0	—	∞	ns	HS oscillator

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Instruction cycle period (Tcy) equals four times the input oscillator time base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min" values with an external clock applied to OSC1 pin. When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

PIC12F609/615/617/12HV609/615

FIGURE 16-8: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS

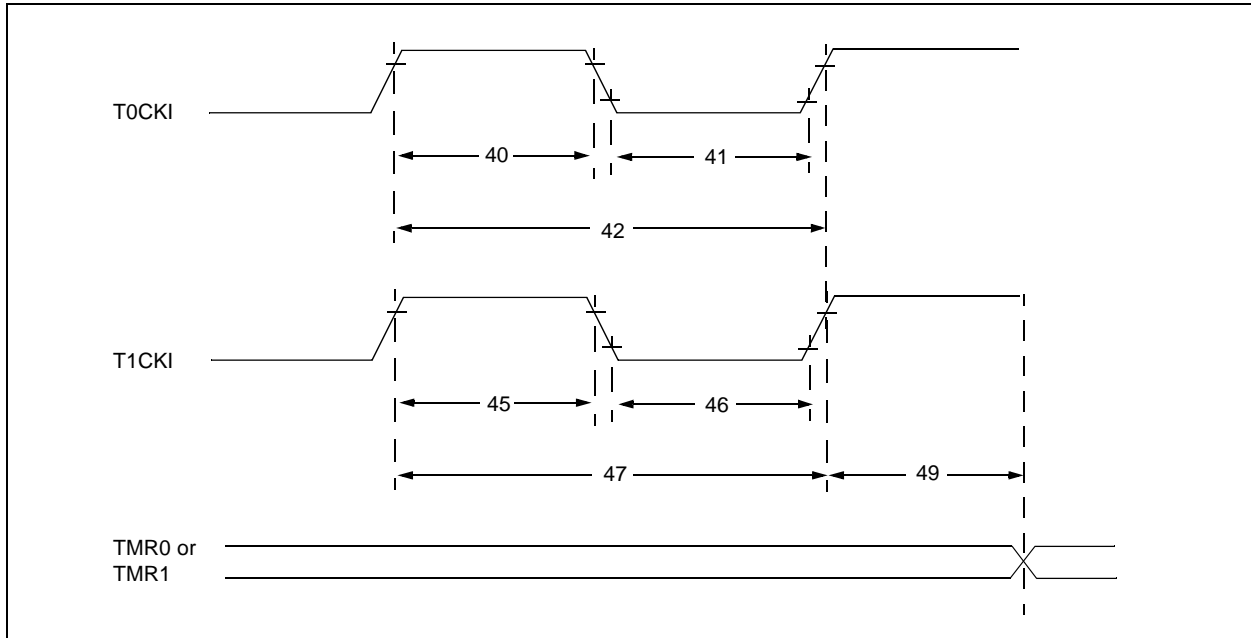


TABLE 16-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)								
Operating Temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$								
Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
40*	Tt0H	T0CKI High Pulse Width	No Prescaler	$0.5 T_{CY} + 20$	—	—	ns	
			With Prescaler	10	—	—	ns	
41*	Tt0L	T0CKI Low Pulse Width	No Prescaler	$0.5 T_{CY} + 20$	—	—	ns	
			With Prescaler	10	—	—	ns	
42*	Tt0P	T0CKI Period		Greater of: 20 or $\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (2, 4, ..., 256)
45*	Tt1H	T1CKI High Time	Synchronous, No Prescaler	$0.5 T_{CY} + 20$	—	—	ns	
			Synchronous, with Prescaler	15	—	—	ns	
			Asynchronous	30	—	—	ns	
46*	Tt1L	T1CKI Low Time	Synchronous, No Prescaler	$0.5 T_{CY} + 20$	—	—	ns	
			Synchronous, with Prescaler	15	—	—	ns	
			Asynchronous	30	—	—	ns	
47*	Tt1P	T1CKI Input Period	Synchronous	Greater of: 30 or $\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1, 2, 4, 8)
			Asynchronous	60	—	—	ns	
48	Ft1	Timer1 Oscillator Input Frequency Range (oscillator enabled by setting bit T1OSCEN)		—	32.768	—	kHz	
49*	TCKEZTMR1	Delay from External Clock Edge to Timer Increment		$2 T_{OSC}$	—	$7 T_{OSC}$	—	Timers in Sync mode

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

PIC12F609/615/617/12HV609/615

FIGURE 17-10: PIC12F609/615/617 IPD BASE vs. VDD

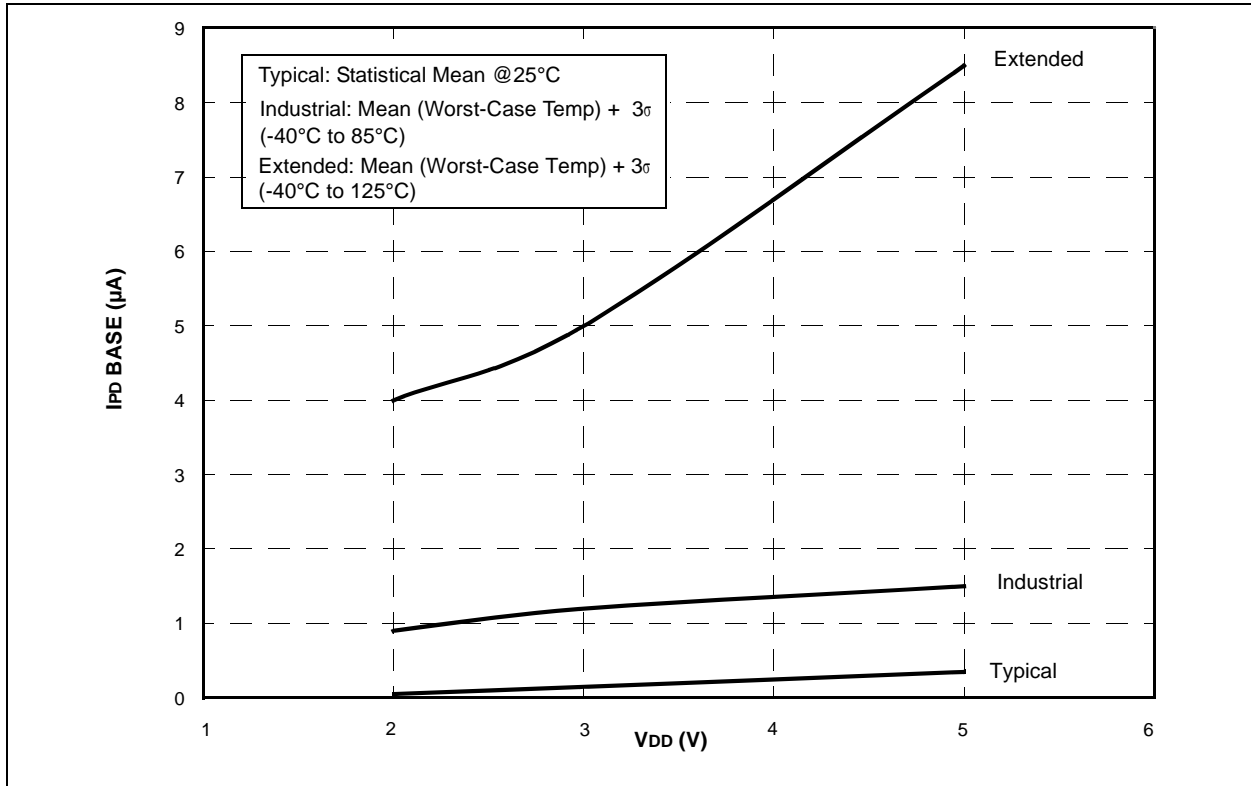
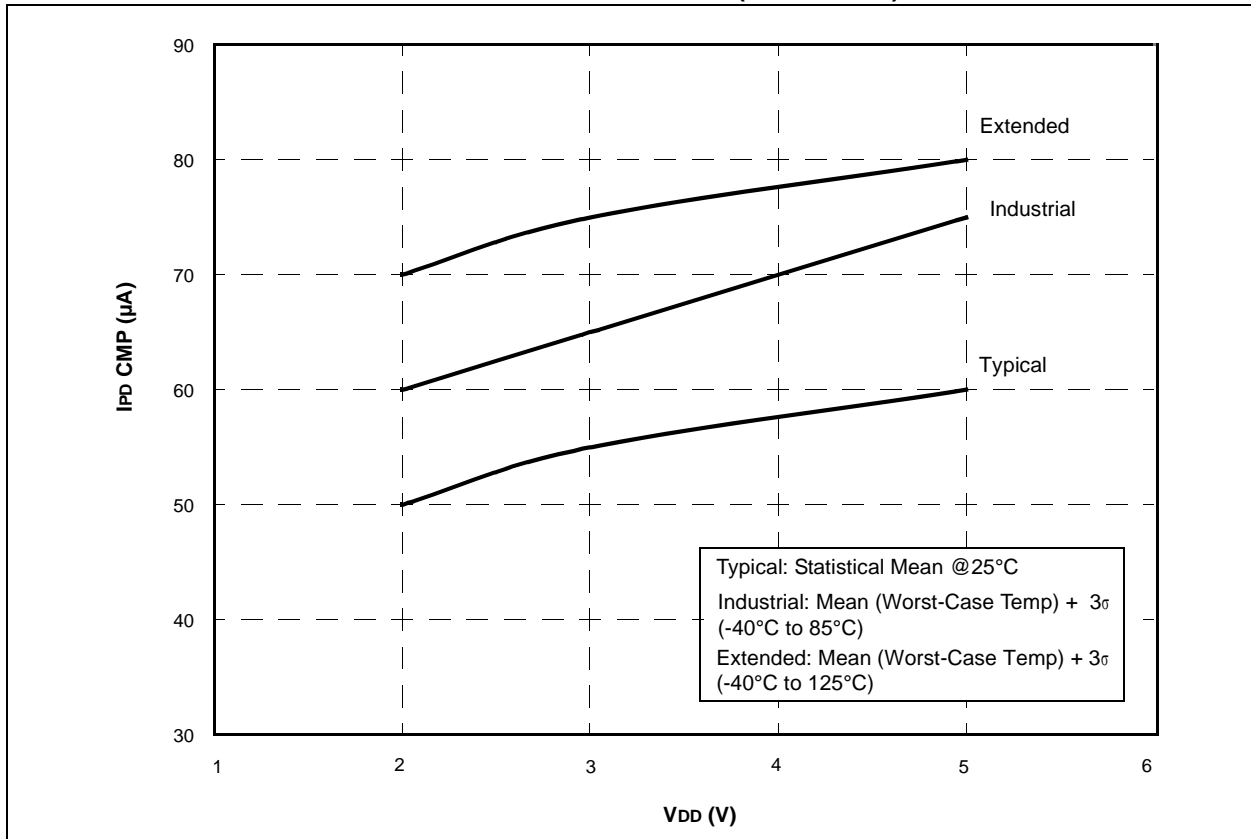


FIGURE 17-11: PIC12F609/615/617 IPD COMPARATOR (SINGLE ON) vs. VDD



PIC12F609/615/617/12HV609/615

FIGURE 17-33: PIC12HV615 IPD A/D vs. VDD

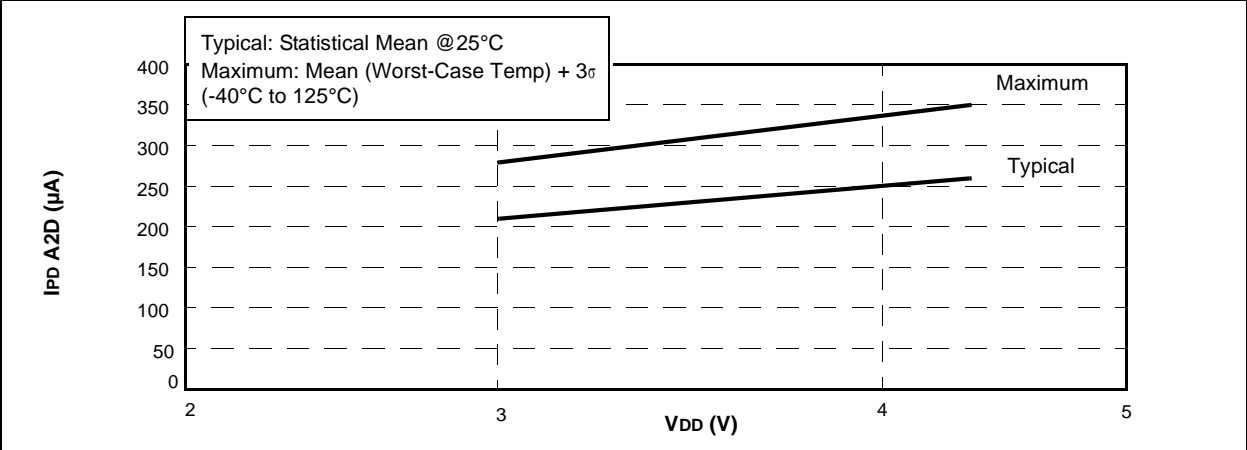
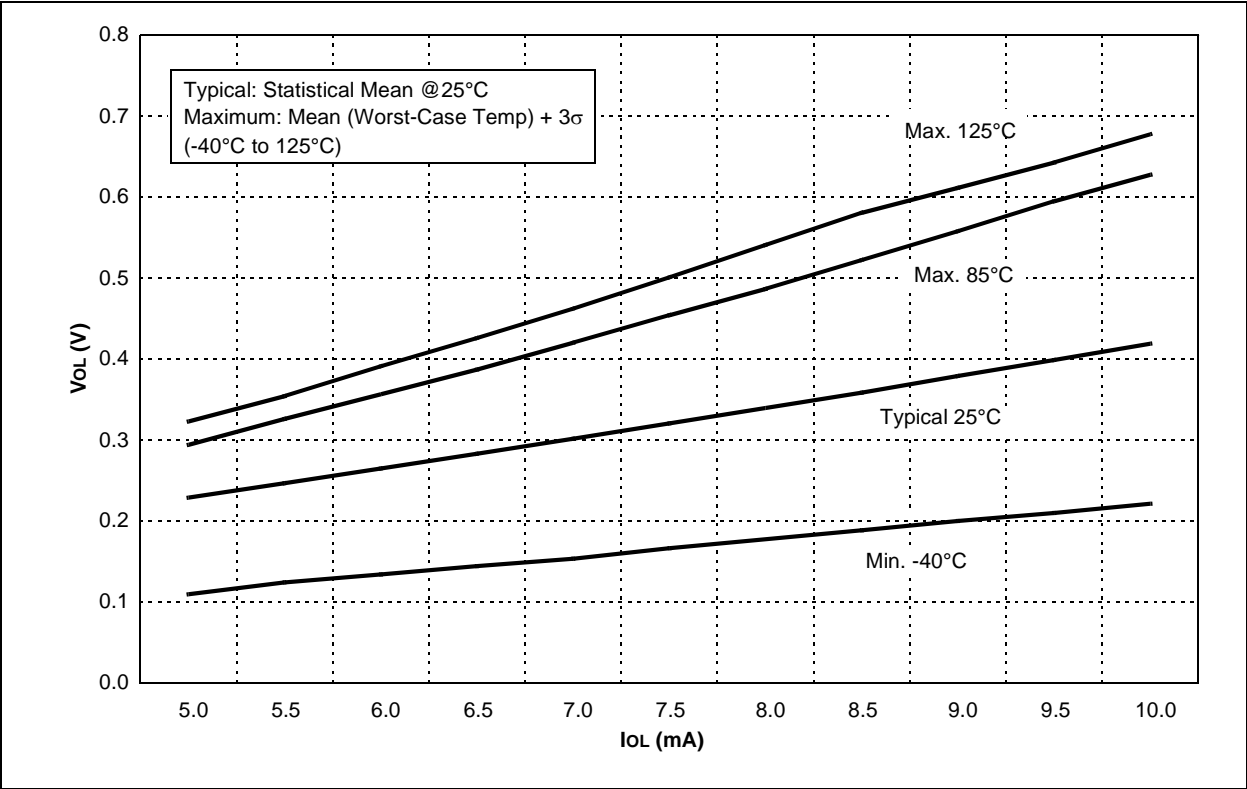


FIGURE 17-34: VOL vs. IOL OVER TEMPERATURE (VDD = 3.0V)



PIC12F609/615/617/12HV609/615

Initializing GPIO	43	Timer Resources	89
Saving Status and W Registers in RAM	121	Enhanced Capture/Compare/PWM	
Writing to Flash Program Memory	34	(PIC12F615/617/HV615 Only)	89
Code Protection	124	Errata	6
Comparator	67	F	
Associated registers	78	Firmware Instructions	129
Control	69	Flash Program Memory Self Read/Self Write	
Gating Timer1	73	Control (For PIC12F617 only)	27
Operation During Sleep	71	Fuses. See Configuration Bits	
Overview	67	G	
Response Time	69	General Purpose Register File	12
Synchronizing COUT w/Timer1	73	GPIO	43
Comparator Hysteresis	77	Additional Pin Functions	44
Comparator Voltage Reference (CVREF)	74	ANSEL Register	44
Effects of a Reset	71	Interrupt-on-Change	44
Comparator Voltage Reference (CVREF)		Weak Pull-Ups	44
Response Time	69	Associated registers	52
Comparator Voltage Reference (CVREF)		GP0	47
Specifications	163	GP1	47
Comparators		GP2	48
C2OUT as T1 Gate	60	GP3	49
Effects of a Reset	71	GP4	50
Specifications	162	GP5	51
Compare Module. See Enhanced Capture/Compare/ PWM (ECCP) (PIC12F615/617/HV615 only)		Pin Descriptions and Diagrams	47
CONFIG Register	108	Specifications	158
Configuration Bits	107	GPIO Register	43
CPU Features	107	H	
Customer Change Notification Service	209	High Temperature Operation	167
Customer Notification Service	209	I	
Customer Support	209	ID Locations	124
D		In-Circuit Debugger	125
Data EEPROM Memory		In-Circuit Serial Programming (ICSP)	125
Associated Registers	35	Indirect Addressing, INDF and FSR registers	25
Data Memory	11	Instruction Format	129
DC and AC Characteristics		Instruction Set	129
Graphs and Tables	171	ADDLW	131
DC Characteristics		ADDWF	131
Extended and Industrial	152	ANDLW	131
Industrial and Extended	145	ANDWF	131
Development Support	139	MOVF	134
Device Overview	7	BCF	131
E		BSF	131
ECCP. See Enhanced Capture/Compare/PWM		BTFSC	131
ECCPAS Register	102	BTFSS	132
EEDAT Register	28	CALL	132
EEDATH Register	28	CLRf	132
Effects of Reset		CLRw	132
PWM mode	96	CLRWDt	132
Electrical Specifications	143	COMF	132
Enhanced Capture/Compare/PWM (ECCP)		DECf	132
Enhanced PWM Mode	97	DECFSZ	133
Auto-Restart	103	GOTO	133
Auto-shutdown	101	INCF	133
Half-Bridge Application	99	INCFSZ	133
Half-Bridge Application Examples	104	IORLW	133
Half-Bridge Mode	99	IORWF	133
Output Relationships (Active-High and		MOVLW	134
Active-Low)	98	MOVWF	134
Output Relationships Diagram	98	NOP	134
Programmable Dead Band Delay	104	RETFIE	135
Shoot-through Current	104	RETLW	135
Start-up Considerations	100	RETURN	135
Specifications	162		

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com, click on Customer Change Notification and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support
- Development Systems Information Line

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: <http://support.microchip.com>

PIC12F609/615/617/12HV609/615

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
RE: Reader Response
From: Name _____
Company _____
Address _____
City / State / ZIP / Country _____
Telephone: (____) _____ - _____ FAX: (____) _____ - _____

Application (optional):

Would you like a reply? __Y__ __N__

Device: PIC12F609/615/617/12HV609/615

Literature Number: DS41302D

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?
