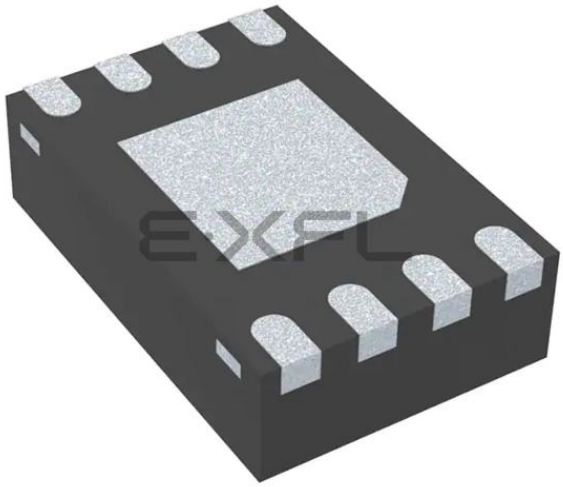


**Welcome to E-XFL.COM**

## 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.

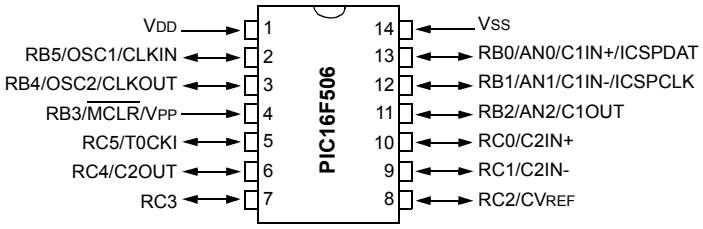
## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Active |
| Core Processor | PIC |
| Core Size | 8-Bit |
| Speed | 8MHz |
| Connectivity | - |
| Peripherals | POR, WDT |
| Number of I/O | 5 |
| Program Memory Size | 1.5KB (1K x 12) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 38 x 8 |
| Voltage - Supply (Vcc/Vdd) | 2V ~ 5.5V |
| Data Converters | A/D 4x8b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 125°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 8-VFDFN Exposed Pad |
| Supplier Device Package | 8-DFN (2x3) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic12f510-e-mc |

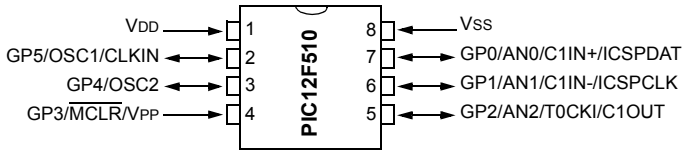
| Device | Program Memory | Data Memory | I/O | Timers 8-bit |
|---|---|---|---|---|
| | Flash (words) | SRAM (bytes) | | |
| PIC16F506 | 1024 | 67 | 12 | 1 |
| PIC12F510 | 1024 | 38 | 6 | 1 |

## Pin Diagrams

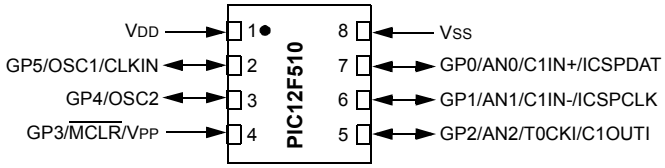
**PDIP, SOIC and TSSOP**




**PDIP, SOIC, MSOP**




**DFN**





© 2007 Microchip Technology Inc.

**TABLE 3-3: PIN DESCRIPTIONS – PIC16F506**

| Name | Function | Input Type | Output Type | Description |
|---|---|---|---|---|
| RB0/AN0/C1IN+/ICSPDAT | RB0 | TTL | CMOS | Bidirectional I/O port. Can be software programmed for internal weak pull-up and wake-up from Sleep on pin change. |
| | AN0 | AN | — | ADC channel input. |
| | C1IN+ | AN | — | Comparator 1 input. |
| | ICSPDAT | ST | CMOS | In-Circuit Serial Programming data pin. |
| RB1/AN1/C1IN-/ICSPCLK | RB1 | TTL | CMOS | Bidirectional I/O port. Can be software programmed for internal weak pull-up and wake-up from Sleep on pin change. |
| | AN1 | AN | — | ADC channel input. |
| | C1IN- | AN | — | Comparator 1 input. |
| | ICSPCLK | ST | — | In-Circuit Serial Programming clock pin. |
| RB2/AN2/C1OUT | RB2 | TTL | CMOS | Bidirectional I/O port. |
| | AN2 | AN | — | ADC channel input. |
| | C1OUT | — | CMOS | Comparator 1 output. |
| RB3/$\overline{\text{MCLR}}$/$V_{PP}$ | RB3 | TTL | — | Standard TTL input. Can be software programmed for internal weak pull-up and wake-up from Sleep on pin change. |
| | $\overline{\text{MCLR}}$ | ST | — | $\overline{\text{MCLR}}$ input – weak pull-up always enabled in this mode. |
| | $V_{PP}$ | HV | — | Programming voltage input. |
| RB4/OSC2/CLKOUT | RB4 | TTL | CMOS | Bidirectional I/O port. Can be software programmed for internal weak pull-up and wake-up from Sleep on pin change. |
| | OSC2 | — | XTAL | XTAL oscillator output pin. |
| | CLKOUT | — | CMOS | EXTRC/INTOSC CLKOUT pin ($F_{OSC}$/4). |
| RB5/OSC1/CLKIN | RB5 | TTL | CMOS | Bidirectional I/O port. |
| | OSC1 | XTAL | — | XTAL oscillator input pin. |
| | CLKIN | ST | — | EXTRC/EC Schmitt Trigger input. |
| RC0/C2IN+ | RC0 | TTL | CMOS | Bidirectional I/O port. |
| | C2IN+ | AN | — | Comparator 2 input. |
| RC1/C2IN- | RC1 | TTL | CMOS | Bidirectional I/O port. |
| | C2IN- | AN | — | Comparator 2 input. |
| RC2/$CV_{REF}$ | RC2 | TTL | CMOS | Bidirectional I/O port. |
| | $CV_{REF}$ | — | AN | Programmable Voltage Reference output. |
| RC3 | RC3 | TTL | CMOS | Bidirectional I/O port. |
| RC4/C2OUT | RC4 | TTL | CMOS | Bidirectional I/O port. |
| | C2OUT | — | CMOS | Comparator 2 output. |
| RC5/T0CKI | RC5 | TTL | CMOS | Bidirectional I/O port. |
| | T0CKI | ST | — | Timer0 clock input. |
| $V_{DD}$ | $V_{DD}$ | P | — | Positive supply for logic and I/O pins. |
| $V_{SS}$ | $V_{SS}$ | P | — | Ground reference for logic and I/O pins. |

**Legend:** I = input, O = output, I/O = input/output, P = power, — = Not Used, TTL = TTL input, ST = Schmitt Trigger input, AN = Analog Voltage, HV = High Voltage

## 3.1 Clocking Scheme/Instruction Cycle

The clock input (OSC1/CLKIN pin) is internally divided by four to generate four non-overlapping quadrature clocks, namely Q1, Q2, Q3 and Q4. Internally, the PC is incremented every Q1 and the instruction is fetched from program memory and latched into the instruction register in Q4. It is decoded and executed during the following Q1 through Q4. The clocks and instruction execution flow is shown in Figure 3-3 and Example 3-1.

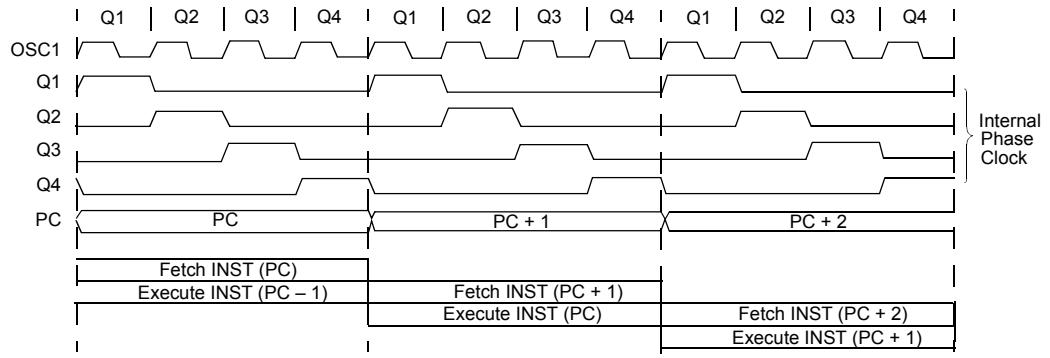
## 3.2 Instruction Flow/Pipelining

An instruction cycle consists of four Q cycles (Q1, Q2, Q3 and Q4). The instruction fetch and execute are pipelined such that fetch takes one instruction cycle, while decode and execute take another instruction cycle. However, due to the pipelining, each instruction effectively executes in one cycle. If an instruction causes the PC to change (e.g., `GOTO`), then two cycles are required to complete the instruction (Example 3-1).

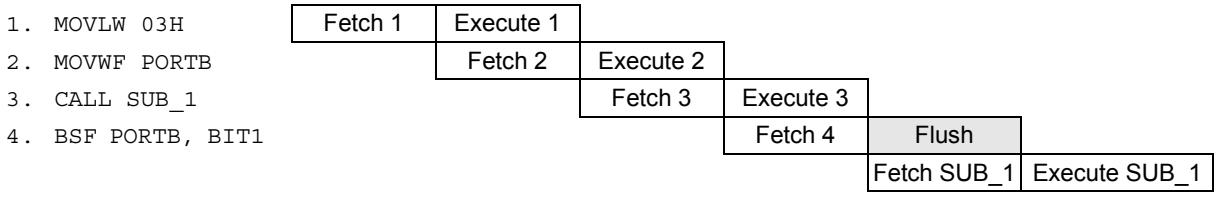
A fetch cycle begins with the PC incrementing in Q1.

In the execution cycle, the fetched instruction is latched into the Instruction Register (IR) in cycle Q1. This instruction is then decoded and executed during the Q2, Q3 and Q4 cycles. Data memory is read during Q2 (operand read) and written during Q4 (destination write).

**FIGURE 3-3: CLOCK/INSTRUCTION CYCLE**




**EXAMPLE 3-1: INSTRUCTION PIPELINE FLOW**




All instructions are single cycle, except for any program branches. These take two cycles, since the fetch instruction is “flushed” from the pipeline, while the new instruction is being fetched and then executed.


© 2007 Microchip Technology Inc.

**REGISTER 4-1: STATUS: STATUS REGISTER (PIC12F510)**

| R/W-0 | R/W-0 | R/W-0 | R-1 | R-1 | R/W-x | R/W-x | R/W-x |
|---|---|---|---|---|---|---|---|
| GPWUF | CWUF | PA0 | $\overline{TO}$ | $\overline{PD}$ | 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 **GPWUF**: GPIO Reset bit
1 = Reset due to wake-up from Sleep on pin change
0 = After power-up or other Reset

bit 6 **CWUF**: Comparator Reset bit
1 = Reset due to wake-up from Sleep on comparator change
0 = After power-up or other Reset

bit 5 **PA0**: Program Page Preselect bit
1 = Page 1 (200h-3FFh)
0 = Page 0 (000h-1FFh)
Each page is 512 bytes.
Using the PA0 bit as a general purpose read/write bit in devices which do not use it for program page preselect is not recommended, since this may affect upward compatibility with future products.

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 (for ADDWF and SUBWF instructions)
ADDWF:
1 = A carry from the 4th low-order bit of the result occurred
0 = A carry from the 4th low-order bit of the result did not occur
SUBWF:
1 = A borrow from the 4th low-order bit of the result did not occur
0 = A borrow from the 4th low-order bit of the result occurred

bit 0 **C**: Carry/$\overline{Borrow}$ bit (for ADDWF, SUBWF and RRF, RLF instructions)

| ADDWF: | SUBWF: | RRF or RLF: |
|---|---|---|
| 1 = A carry occurred | 1 = A borrow did not occur | Load bit with LSb or MSb, respectively |
| 0 = A carry did not occur | 0 = A borrow occurred | |

## 4.4 OPTION Register

The OPTION register is a 8-bit wide, write-only register, that contains various control bits to configure the Timer0/WDT prescaler and Timer0.

By executing the OPTION instruction, the contents of the W register will be transferred to the OPTION register. A Reset sets the OPTION<7:0> bits.

| | |
|---|---|
| **Note 1:** | If TRIS bit is set to '0', the wake-up on change and pull-up functions are disabled for that pin (i.e., note that TRIS overrides Option control of $\overline{\text{GPPU}}$/RBPU and $\overline{\text{GPWU}}$/$\overline{\text{RBWU}}$). |
| **2:** | If the T0CS bit is set to '1', it will override the TRIS function on the T0CKI pin. |

**REGISTER 4-3: OPTION_REG: OPTION REGISTER (PIC12F510)**

| W-1 | W-1 | W-1 | W-1 | W-1 | W-1 | W-1 | W-1 |
|---|---|---|---|---|---|---|---|
| $\overline{\text{GPWU}}$ | $\overline{\text{GPPU}}$ | T0CS | T0SE | PSA | PS2 | PS1 | PS0 |
| bit 7 | | | | | | | bit 0 |

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

bit 7 **$\overline{\text{GPWU}}$:** Enable Wake-up On Pin Change bit (GP0, GP1, GP3)
1 = Disabled
0 = Enabled

bit 6 **$\overline{\text{GPPU}}$:** Enable Weak Pull-Ups bit (GP0, GP1, GP3)
1 = Disabled
0 = Enabled

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

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

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

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

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

## 5.0 I/O PORT

As with any other register, the I/O register(s) can be written and read under program control. However, read instructions (e.g., `MOVF PORTB, W`) always read the I/O pins independent of the pin's Input/Output modes. On Reset, all I/O ports are defined as input (inputs are at high-impedance) since the I/O control registers are all set.

| | |
|---|---|
| **Note:** | On the PIC12F510, I/O PORTB is referenced as GPIO. On the PIC16F506, I/O PORTB is referenced as PORTB. |

### 5.1 PORTB/GPIO

PORTB/GPIO is an 8-bit I/O register. Only the low-order 6 bits are used (RB/GP<5:0>). Bits 7 and 6 are unimplemented and read as '0's. Please note that RB3/GP3 is an input only pin. The Configuration Word can set several I/O's to alternate functions. When acting as alternate functions, the pins will read as '0' during a port read. Pins RB0/GP0, RB1/GP1, RB3/GP3 and RB4 (PIC16F506 only) can be configured with weak pull-up and also for wake-up on change. The wake-up on change and weak pull-up functions are not pin selectable. If RB3/GP3/$\overline{MCLR}$ is configured as $\overline{MCLR}$, weak pull-up is always on and wake-up on change for this pin is not enabled.

### 5.2 PORTC (PIC16F506 Only)

PORTC is an 8-bit I/O register. Only the low-order 6 bits are used (RC<5:0>). Bits 7 and 6 are unimplemented and read as '0's.

### 5.3 TRIS Registers

The Output Driver Control register is loaded with the contents of the W register by executing the `TRIS f` instruction. A '1' from a TRIS register bit puts the corresponding output driver in a High-Impedance mode. A '0' puts the contents of the output data latch on the selected pins, enabling the output buffer. The exception is RB3/GP3, which are input only, and the T0CKI pin, which may be controlled by the OPTION register. See Register 4-3.

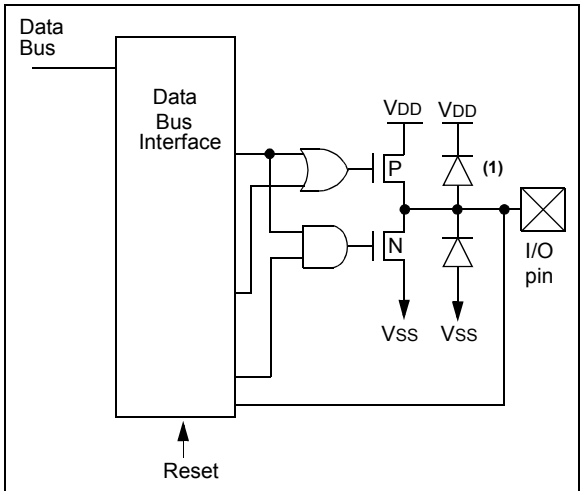
| | |
|---|---|
| **Note:** | A read of the port reads the pins, not the output data latches. That is, if an output driver on a pin is enabled and driven high but the external system is holding it low, a read of the port will indicate that the pin is low. |

| | |
|---|---|
| **Note:** | The TRIS registers are write-only and are set (output drivers disabled) upon Reset. |

### 5.4 I/O Interfacing

The equivalent circuit for an I/O port pin is shown in Figure 5-1. All port pins, except RB3/GP3 which is input only, may be used for both input and output operations. For input operations, these ports are non-latching. Any input must be present until read by an input instruction (e.g., `MOVF PORTB, W`). The outputs are latched and remain unchanged until the output latch is rewritten. To use a port pin as output, the corresponding direction control bit in TRIS must be cleared (= `0`). For use as an input, the corresponding TRIS bit must be set. Any I/O pin (except RB3/GP3) can be programmed individually as input or output.

**FIGURE 5-1: PIC12F510/16F506 EQUIVALENT CIRCUIT FOR PIN DRIVE(2)**




**Note 1:** GP3/RB3 has protection diode to VSS only.

**2:** For pin specific information, see Figure 5-2 through Figure 5-13.

## 6.0 TMR0 MODULE AND TMR0 REGISTER

The Timer0 module has the following features:

- 8-bit timer/counter register, TMR0
- Readable and writable
- 8-bit software programmable prescaler
- Internal or external clock select:
  - Edge select for external clock
  - External clock from either the T0CKI pin or from the output of the comparator

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

Timer mode is selected by clearing the T0CS bit (OPTION<5>). In Timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If TMR0 register is written, the increment is inhibited for the following two cycles (Figure 6-2 and Figure 6-3). The user can work around this by writing an adjusted value to the TMR0 register.

There are two types of Counter mode. The first Counter mode uses the T0CKI pin to increment Timer0. It is selected by setting the T0CKI bit (OPTION<5>), setting the $\overline{\text{C1T0CS}}$ bit (CM1CON0<4>) and setting the $\overline{\text{C1OUTEN}}$ bit (CM1CON0<6>). In this mode, Timer0 will increment either on every rising or falling edge of pin T0CKI. The T0SE bit (OPTION<4>) determines the source edge. Clearing the T0SE bit selects the rising edge. Restrictions on the external clock input are discussed in detail in **Section 6.1 "Using Timer0 With An External Clock"**.

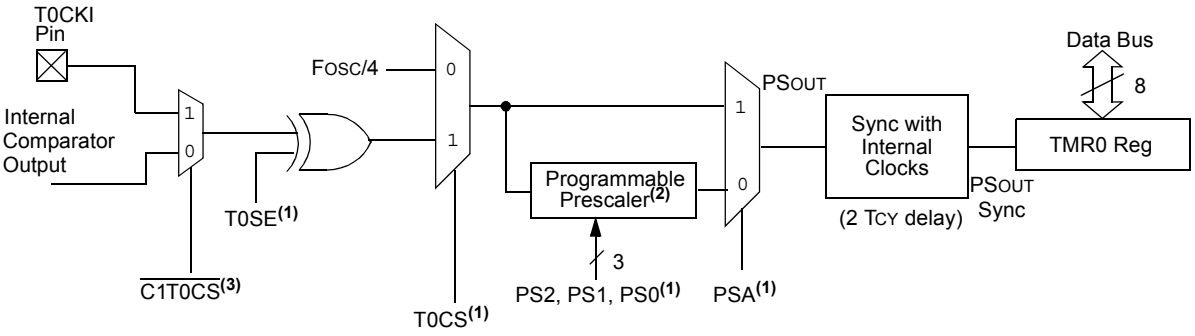
The second Counter mode uses the output of the comparator to increment Timer0. It can be entered in two different ways. The first way is selected by setting the T0CS bit (OPTION<5>), and clearing the $\overline{\text{C1T0CS}}$ bit (CM1CON0<4>) ($\overline{\text{C1OUTEN}}$ [CM1CON0<6>] does not affect this mode of operation). This enables an internal connection between the comparator and the Timer0.

The second way is selected by setting the T0CS bit (OPTION<5>), setting the $\overline{\text{C1T0CS}}$ bit (CM1CON0) and clearing the $\overline{\text{C1OUTEN}}$ bit (CM1CON0<6>). This allows the output of the comparator onto the T0CKI pin, while keeping the T0CKI input active. Therefore, any comparator change on the COUT pin is fed back into the T0CKI input. The T0SE bit (OPTION<4>) determines the source edge. Clearing the T0SE bit selects the rising edge. Restrictions on the external clock input as discussed in **Section 6.1 "Using Timer0 With An External Clock"**.

The prescaler may be used by either the Timer0 module or the Watchdog Timer, but not both. The prescaler assignment is controlled in software by the control bit PSA (OPTION<3>). Clearing the PSA bit will assign the prescaler to Timer0. The prescaler is not readable or writable. When the prescaler is assigned to the Timer0 module, prescale values of 1:2, 1:4,..., 1:256 are selectable. **Section 6.2 "Prescaler"** details the operation of the prescaler.

A summary of registers associated with the Timer0 module is found in Table 6-1.

**FIGURE 6-1: TIMER0 BLOCK DIAGRAM**




**Note 1:** Bits T0CS, T0SE, PSA, PS2, PS1 and PS0 are located in the OPTION register.
**2:** The prescaler is shared with the Watchdog Timer (Figure 6-5).
**3:** Bit $\overline{\text{C1T0CS}}$ is located in the CM1CON0 register, CM1CON0<4>.

**REGISTER 7-3: CM2CON0: COMPARATOR C2 CONTROL REGISTER (PIC16F506)**

| R-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| C2OUT | $\overline{\text{C2OUTEN}}$ | C2POL | C2PREF2 | C2ON | C2NREF | C2PREF1 | $\overline{\text{C2WU}}$ |
| bit 7 | | | | | | | bit 0 |

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

bit 7 **C2OUT:** Comparator Output bit
1 = $V_{IN+} > V_{IN-}$
0 = $V_{IN+} < V_{IN-}$

bit 6 **$\overline{\text{C2OUTEN}}$:** Comparator Output Enable bit(1), (2)
1 = Output of comparator is NOT placed on the C2OUT pin
0 = Output of comparator is placed in the C2OUT pin

bit 5 **C2POL:** Comparator Output Polarity bit(2)
1 = Output of comparator not inverted
0 = Output of comparator inverted

bit 4 **C2PREF2:** Comparator Positive Reference Select bit(2)
1 = C1IN+ pin
0 = C2IN- pin

bit 3 **C2ON:** Comparator Enable bit
1 = Comparator is on
0 = Comparator is off

bit 2 **C2NREF:** Comparator Negative Reference Select bit(2)
1 = C2IN- pin
0 = $CV_{REF}$

bit 1 **C2PREF1:** Comparator Positive Reference Select bit(2)
1 = C2IN+ pin
0 = C2PREF2 controls analog input selection

bit 0 **$\overline{\text{C2WU}}$:** Comparator Wake-up on Change Enable bit(2)
1 = Wake-up on Comparator change is disabled
0 = Wake-up on Comparator change is enabled.

**Note 1:** Overrides TOCS bit for TRIS control of RC4.

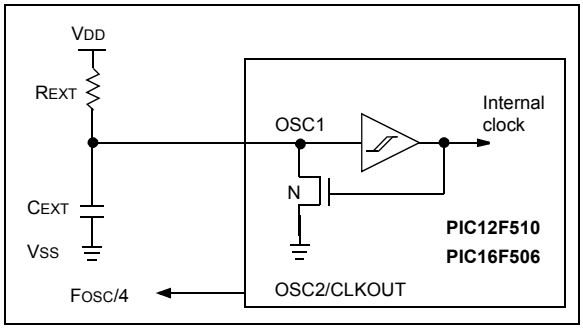
**2:** When comparator is turned on, these control bits assert themselves. Otherwise, the other registers have precedence.

Although the oscillator will operate with no external capacitor ($C_{EXT}$ = 0 pF), we recommend using values above 20 pF for noise and stability reasons. With no capacitance or small external capacitance, the oscillation frequency can vary dramatically due to changes in external capacitances, such as PCB trace capacitance or package lead frame capacitance.

**Section 13.0 "Electrical Characteristics"**, shows RC frequency variation from part-to-part due to normal process variation. The variation is larger for larger values of R (since leakage current variation will affect RC frequency more for large R) and for smaller values of C (since variation of input capacitance will affect RC frequency more).

Also, see the Electrical Specifications section for variation of oscillator frequency due to $V_{DD}$ for given $R_{EXT}/C_{EXT}$ values, as well as frequency variation due to operating temperature for given R, C and $V_{DD}$ values.

**FIGURE 10-5: EXTERNAL RC OSCILLATOR MODE**




### 10.2.5 INTERNAL 4/8 MHz RC OSCILLATOR

The internal RC oscillator provides a fixed 4/8 MHz (nominal) system clock (see **Section 13.0 "Electrical Characteristics"** for information on variation over voltage and temperature).

In addition, a calibration instruction is programmed into the last address of memory, which contains the calibration value for the internal RC oscillator. This location is always uncode protected, regardless of the code-protect settings. This value is programmed as a `MOVLW XX` instruction where `XX` is the calibration value, and is placed at the Reset vector. This will load the W register with the calibration value upon Reset and the PC will then roll over to the users program at address 0x000. The user then has the option of writing the value to the OSCCAL Register (05h) or ignoring it.

OSCCAL, when written to with the calibration value, will "trim" the internal oscillator to remove process variation from the oscillator frequency.

| **Note:** | Erasing the device will also erase the pre-programmed internal calibration value for the internal oscillator. The calibration value must be read prior to erasing the part so it can be reprogrammed correctly later. |
|---|---|

For the PIC12F510/16F506 devices, only bits <7:1> of OSCCAL are used for calibration. See Register 4-5 for more information.

| **Note:** | The 0 bit of OSCCAL is unimplemented and should be written as '0' when modifying OSCCAL for compatibility with future devices. |
|---|---|


© 2007 Microchip Technology Inc.

## 10.7 Time-out Sequence, Power-down and Wake-up from Sleep Status Bits (TO, PD, GPWUF/RBWUF)

The $\overline{TO}$, $\overline{PD}$ and (GPWUF/RBWUF) bits in the STATUS register can be tested to determine if a Reset condition has been caused by a power-up condition, a $\overline{MCLR}$ or Watchdog Timer (WDT) Reset.

**TABLE 10-8: $\overline{TO}$/$\overline{PD}$/(GPWUF/RBWUF) STATUS AFTER RESET**

| CWUF | GPWUF/ RBWUF | $\overline{TO}$ | $\overline{PD}$ | Reset Caused By |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | WDT wake-up from Sleep |
| 0 | 0 | 0 | u | WDT time-out (not from Sleep) |
| 0 | 0 | 1 | 0 | $\overline{MCLR}$ wake-up from Sleep |
| 0 | 0 | 1 | 1 | Power-up |
| 0 | 0 | u | u | $\overline{MCLR}$ not during Sleep |
| 0 | 1 | 1 | 0 | Wake-up from Sleep on pin change |
| 1 | 0 | 1 | 0 | Wake-up from Sleep on comparator change |

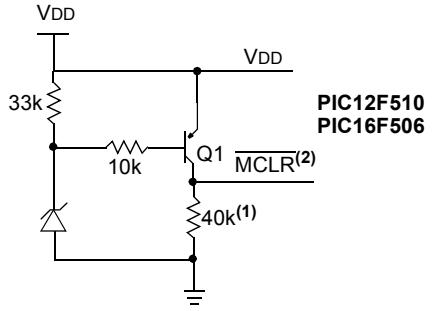
**Legend:** u = unchanged

## 10.8 Reset on Brown-out

A brown-out is a condition where device power ($V_{DD}$) dips below its minimum value, but not to zero, and then recovers. The device should be reset in the event of a brown-out.

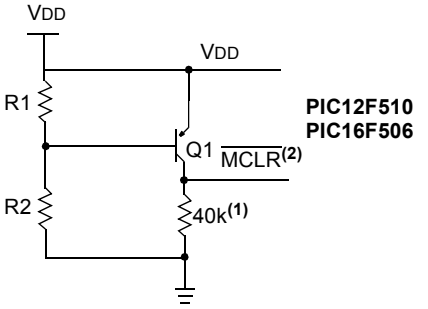
To reset PIC12F510/16F506 devices when a brown-out occurs, external brown-out protection circuits may be built, as shown in Figure 10-12 and Figure 10-13.

**FIGURE 10-12: BROWN-OUT PROTECTION CIRCUIT 1**




**Note 1:** This circuit will activate Reset when $V_{DD}$ goes below Vz + 0.7V (where Vz = Zener voltage).
**2:** Pin must be configured as $\overline{MCLR}$.

**FIGURE 10-13: BROWN-OUT PROTECTION CIRCUIT 2**

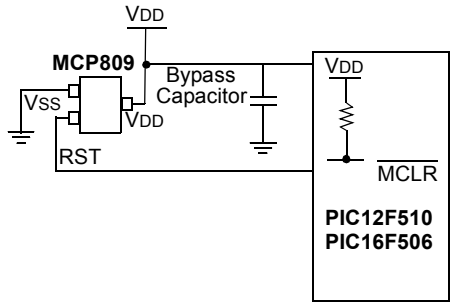



**Note 1:** This brown-out circuit is less expensive, although less accurate. Transistor Q1 turns off when $V_{DD}$ is below a certain level such that:

$$V_{DD} \cdot \frac{R1}{R1 + R2} = 0.7V$$

**2:** Pin must be configured as $\overline{MCLR}$.

**FIGURE 10-14: BROWN-OUT PROTECTION CIRCUIT 3**




**Note:** This brown-out protection circuit employs Microchip Technology's MCP809 microcontroller supervisor. There are 7 different trip point selections to accommodate 5V to 3V systems.


© 2007 Microchip Technology Inc.

## 10.9 Power-Down Mode (Sleep)

A device may be powered down (Sleep) and later powered up (wake-up from Sleep Reset).

### 10.9.1 SLEEP

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

If enabled, the Watchdog Timer will be cleared but keeps running, the $\overline{TO}$ bit (STATUS<4>) is set, the $\overline{PD}$ bit (STATUS<3>) is cleared and the oscillator driver is turned off. The I/O ports maintain the status they had before the SLEEP instruction was executed (driving high, driving low or high-impedance).

| | |
|---|---|
| **Note:** | A device Reset generated by a WDT time-will not drive the $\overline{MCLR}$ pin low. |

For lowest current consumption while powered down, all input pins should be at $V_{DD}$ or $V_{SS}$ and (GP3/RB3)/$\overline{MCLR}$/$V_{PP}$ pin must be at a logic high level if $\overline{MCLR}$ is enabled.

### 10.9.2 WAKE-UP FROM SLEEP RESET

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

1. An external Reset input on (GP3/RB3)/$\overline{MCLR}$/$V_{PP}$ pin when configured as $\overline{MCLR}$.
2. A Watchdog Timer Time-out Reset (if WDT was enabled).
3. A change-on-input pin GP0/RB0, GP1/RB1, GP3/RB3 or RB4 when wake-up on change is enabled.
4. A change in the comparator ouput bits, C1OUT and C2OUT (if comparator wake-up is enabled).

These events cause a device Reset. The $\overline{TO}$, $\overline{PD}$, CWUF and GPWUF/RBWUF bits can be used to determine the cause of device Reset. The $\overline{TO}$ bit is cleared if a WDT time-out occurred (and caused wake-up). The $\overline{PD}$ bit, which is set on power-up, is cleared when SLEEP is invoked. The CWUF bit indicates a change in comparator output state while the device was in Sleep. The GPWUF/RBWUF bit indicates a change in state while in Sleep at pins GP0/RB0, GP1/RB1, GP3/RB3 or RB4 (since the last file or bit operation on GP/RB port).

| | |
|---|---|
| **Note:** | **Caution:** Right before entering Sleep, read the input pins. When in Sleep, wake-up occurs when the values at the pins change from the state they were in at the last reading. If a wake-up on change occurs and the pins are not read before reentering Sleep, a wake-up will occur immediately even if no pins change while in Sleep mode. |

| | |
|---|---|
| **Note 1:** | **Caution:** Right before entering Sleep, read the comparator Configuration register(s) CM1CON0 and CM2CON0. When in Sleep, wake-up occurs when the comparator output bit C1OUT and C2OUT change from the state they were in at the last reading. If a wake-up on comparator change occurs and the pins are not read before re-entering Sleep, a wake-up will occur immediately, even if no pins change while in Sleep mode. |
| **2:** | For 16F506 only. |

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

## 10.10 Program Verification/Code Protection

If the code protection bit has not been programmed, the on-chip program memory can be read out for verification purposes.

The first 64 locations and the last location (OSCCAL) can be read, regardless of the code protection bit setting.

The last memory location can be read regardless of the code protection bit setting on the PIC12F510/16F506 devices.

## 10.11 ID Locations

Four memory locations 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.

Use only the lower 4 bits of the ID locations and always set the upper 4 bits as '1's. The upper 4 bits are unimplemented.

These locations can be read regardless of the code protect setting.

**TABLE 11-2: INSTRUCTION SET SUMMARY**

| Mnemonic, Operands | | Description | Cycles | 12-Bit Opcode | | | Status Affected | Notes |
|---|---|---|---|---|---|---|---|---|
| | | | | MSb | | LSb | | |
| ADDWF | f, d | Add W and f | 1 | 0001 | 11df | ffff | C, DC, Z | **1, 2, 4** |
| ANDWF | f, d | AND W with f | 1 | 0001 | 01df | ffff | Z | **2, 4** |
| CLRF | f | Clear f | 1 | 0000 | 011f | ffff | Z | **4** |
| CLRW | – | Clear W | 1 | 0000 | 0100 | 0000 | Z | |
| COMF | f, d | Complement f | 1 | 0010 | 01df | ffff | Z | |
| DECF | f, d | Decrement f | 1 | 0000 | 11df | ffff | Z | **2, 4** |
| DECFSZ | f, d | Decrement f, Skip if 0 | 1**(2)** | 0010 | 11df | ffff | None | **2, 4** |
| INCF | f, d | Increment f | 1 | 0010 | 10df | ffff | Z | **2, 4** |
| INCFSZ | f, d | Increment f, Skip if 0 | 1**(2)** | 0011 | 11df | ffff | None | **2, 4** |
| IORWF | f, d | Inclusive OR W with f | 1 | 0001 | 00df | ffff | Z | **2, 4** |
| MOVF | f, d | Move f | 1 | 0010 | 00df | ffff | Z | **2, 4** |
| MOVWF | f | Move W to f | 1 | 0000 | 001f | ffff | None | **1, 4** |
| NOP | – | No Operation | 1 | 0000 | 0000 | 0000 | None | |
| RLF | f, d | Rotate left f through Carry | 1 | 0011 | 01df | ffff | C | **2, 4** |
| RRF | f, d | Rotate right f through Carry | 1 | 0011 | 00df | ffff | C | **2, 4** |
| SUBWF | f, d | Subtract W from f | 1 | 0000 | 10df | ffff | C, DC, Z | **1, 2, 4** |
| SWAPF | f, d | Swap f | 1 | 0011 | 10df | ffff | None | **2, 4** |
| XORWF | f, d | Exclusive OR W with f | 1 | 0001 | 10df | ffff | Z | **2, 4** |
| **BIT-ORIENTED FILE REGISTER OPERATIONS** | | | | | | | | |
| BCF | f, b | Bit Clear f | 1 | 0100 | bbbf | ffff | None | **2, 4** |
| BSF | f, b | Bit Set f | 1 | 0101 | bbbf | ffff | None | **2, 4** |
| BTFSC | f, b | Bit Test f, Skip if Clear | 1**(2)** | 0110 | bbbf | ffff | None | |
| BTFSS | f, b | Bit Test f, Skip if Set | 1**(2)** | 0111 | bbbf | ffff | None | |
| **LITERAL AND CONTROL OPERATIONS** | | | | | | | | |
| ANDLW | k | AND literal with W | 1 | 1110 | kkkk | kkkk | Z | |
| CALL | k | Call Subroutine | 2 | 1001 | kkkk | kkkk | None | **1** |
| CLRWDT | – | Clear Watchdog Timer | 1 | 0000 | 0000 | 0100 | $\overline{TO}$, $\overline{PD}$ | |
| GOTO | k | Unconditional branch | 2 | 101k | kkkk | kkkk | None | |
| IORLW | k | Inclusive OR literal with W | 1 | 1101 | kkkk | kkkk | Z | |
| MOVLW | k | Move literal to W | 1 | 1100 | kkkk | kkkk | None | |
| OPTION | – | Load OPTION register | 1 | 0000 | 0000 | 0010 | None | |
| RETLW | k | Return, place literal in W | 2 | 1000 | kkkk | kkkk | None | |
| SLEEP | – | Go into Standby mode | 1 | 0000 | 0000 | 0011 | $\overline{TO}$, $\overline{PD}$ | |
| TRIS | f | Load TRIS register | 1 | 0000 | 0000 | 0fff | None | **3** |
| XORLW | k | Exclusive OR literal to W | 1 | 1111 | kkkk | kkkk | Z | |

**Note 1:** The 9th bit of the Program Counter will be forced to a '0' by any instruction that writes to the PC except for GOTO. See **Section 4.6 "Program Counter"**.

**2:** When an I/O register is modified as a function of itself (e.g. MOVF PORTB, 1), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

**3:** The instruction TRIS f, where f = 6, causes the contents of the W register to be written to the tri-state latches of PORTB. A '1' forces the pin to a high-impedance state and disables the output buffers.

**4:** If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared (if assigned to TMR0).


© 2007 Microchip Technology Inc.

| IORWF | Inclusive OR W with f |
|---|---|
| Syntax: | [ *label* ] IORWF f,d |
| Operands: | 0 ≤ f ≤ 31<br>d ∈ [0,1] |
| Operation: | (W).OR. (f) → (dest) |
| Status Affected: | Z |
| Description: | Inclusive OR the W register with register 'f'. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'. |

| MOVF | Move f |
|---|---|
| Syntax: | [ *label* ] MOVF f,d |
| Operands: | 0 ≤ f ≤ 31<br>d ∈ [0,1] |
| Operation: | (f) → (dest) |
| Status Affected: | Z |
| Description: | The contents of register 'f' are moved to destination 'd'. If 'd' is '0', destination is the W register. If 'd' is '1', the destination is file register 'f'. 'd' = 1 is useful as a test of a file register, since Status flag Z is affected. |

| MOVLW | Move Literal to W |
|---|---|
| Syntax: | [ *label* ] MOVLW k |
| Operands: | 0 ≤ k ≤ 255 |
| Operation: | k → (W) |
| Status Affected: | None |
| Description: | The eight-bit literal 'k' is loaded into the W register. The "don't cares" will be assembled as '0's. |

| MOVWF | Move W to f |
|---|---|
| Syntax: | [ *label* ] MOVWF f |
| Operands: | 0 ≤ f ≤ 31 |
| Operation: | (W) → (f) |
| Status Affected: | None |
| Description: | Move data from the W register to register 'f'. |

| NOP | No Operation |
|---|---|
| Syntax: | [ *label* ] NOP |
| Operands: | None |
| Operation: | No operation |
| Status Affected: | None |
| Description: | No operation. |

| OPTION | Load OPTION Register |
|---|---|
| Syntax: | [ *label* ] Option |
| Operands: | None |
| Operation: | (W) → Option |
| Status Affected: | None |
| Description: | The content of the W register is loaded into the OPTION register. |


© 2007 Microchip Technology Inc.

| TRIS | Load TRIS Register |
|---|---|
| Syntax: | [ *label* ] TRIS f |
| Operands: | f = 6 |
| Operation: | (W) → TRIS register f |
| Status Affected: | None |
| Description: | TRIS register 'f' (f = 6 or 7) is loaded with the contents of the W register |

| XORLW | Exclusive OR literal with W |
|---|---|
| Syntax: | [*label* ] XORLW k |
| Operands: | 0 ≤ k ≤ 255 |
| Operation: | (W) .XOR. k → (W) |
| Status Affected: | Z |
| Description: | The contents of the W register are XOR'ed with the eight-bit literal 'k'. The result is placed in the W register. |

| XORWF | Exclusive OR W with f |
|---|---|
| Syntax: | [ *label* ] XORWF f,d |
| Operands: | 0 ≤ f ≤ 31<br>d ∈ [0,1] |
| Operation: | (W) .XOR. (f) → (dest) |
| Status Affected: | Z |
| Description: | Exclusive OR the contents of the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'. |


© 2007 Microchip Technology Inc.

## 12.11 PICSTART Plus Development Programmer

The PICSTART Plus Development Programmer is an easy-to-use, low-cost, prototype programmer. It connects to the PC via a COM (RS-232) port. MPLAB Integrated Development Environment software makes using the programmer simple and efficient. The PICSTART Plus Development Programmer supports most PIC devices in DIP packages up to 40 pins. Larger pin count devices, such as the PIC16C92X and PIC17C76X, may be supported with an adapter socket. The PICSTART Plus Development Programmer is CE compliant.

## 12.12 PICkit 2 Development Programmer

The PICkit™ 2 Development Programmer is a low-cost programmer and selected Flash device debugger with an easy-to-use interface for programming many of Microchip's baseline, mid-range and PIC18F families of Flash memory microcontrollers. The PICkit 2 Starter Kit includes a prototyping development board, twelve sequential lessons, software and HI-TECH's PICC™ Lite C compiler, and is designed to help get up to speed quickly using PIC® microcontrollers. The kit provides everything needed to program, evaluate and develop applications using Microchip's powerful, mid-range Flash memory family of microcontrollers.

## 12.13 Demonstration, Development and Evaluation Boards

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart® battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Check the Microchip web page (www.microchip.com) and the latest *"Product Selector Guide"* (DS00148) for the complete list of demonstration, development and evaluation kits.


© 2007 Microchip Technology Inc.

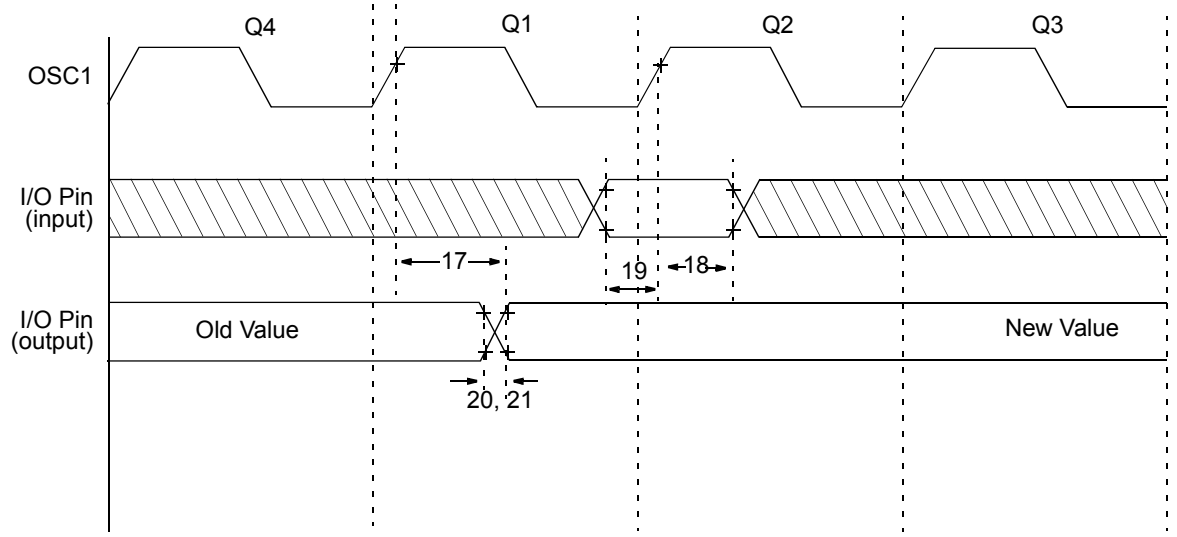
**TABLE 13-5: CALIBRATED INTERNAL RC FREQUENCIES**

| AC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise specified)<br>Operating Temperature -40°C ≤ TA ≤ +85°C (industrial),<br>-40°C ≤ TA ≤ +125°C (extended) | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym** | **Characteristic** | **Freq. Tolerance** | **Min** | **Typ(1)** | **Max*** | **Units** | **Conditions** |
| F10 | FOSC | Internal Calibrated INTOSC Frequency(1) | ±1%<br>±2%<br><br>±5% | 7.92<br>7.84<br><br>7.60 | 8.00<br>8.00<br><br>8.00 | 8.08<br>8.16<br><br>8.40 | MHz<br>MHz<br><br>MHz | VDD = 3.5V TA = 25°C<br>2.5V ≤ VDD ≤ 5.5V<br>0°C ≤ TA ≤ +85°C<br>2.0V ≤ VDD ≤ 5.5V<br>-40°C ≤ TA ≤ +85°C (Ind.)<br>-40°C ≤ TA ≤ +125°C (Ext.) |

* These parameters are characterized but not tested.

**Note 1:** Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**FIGURE 13-7: I/O TIMING**




**Note:** All tests must be done with specified capacitive loads (see data sheet) 50 pF on I/O pins and CLKOUT.


© 2007 Microchip Technology Inc.

**FIGURE 14-12: TTL INPUT THRESHOLD VIN vs. VDD**

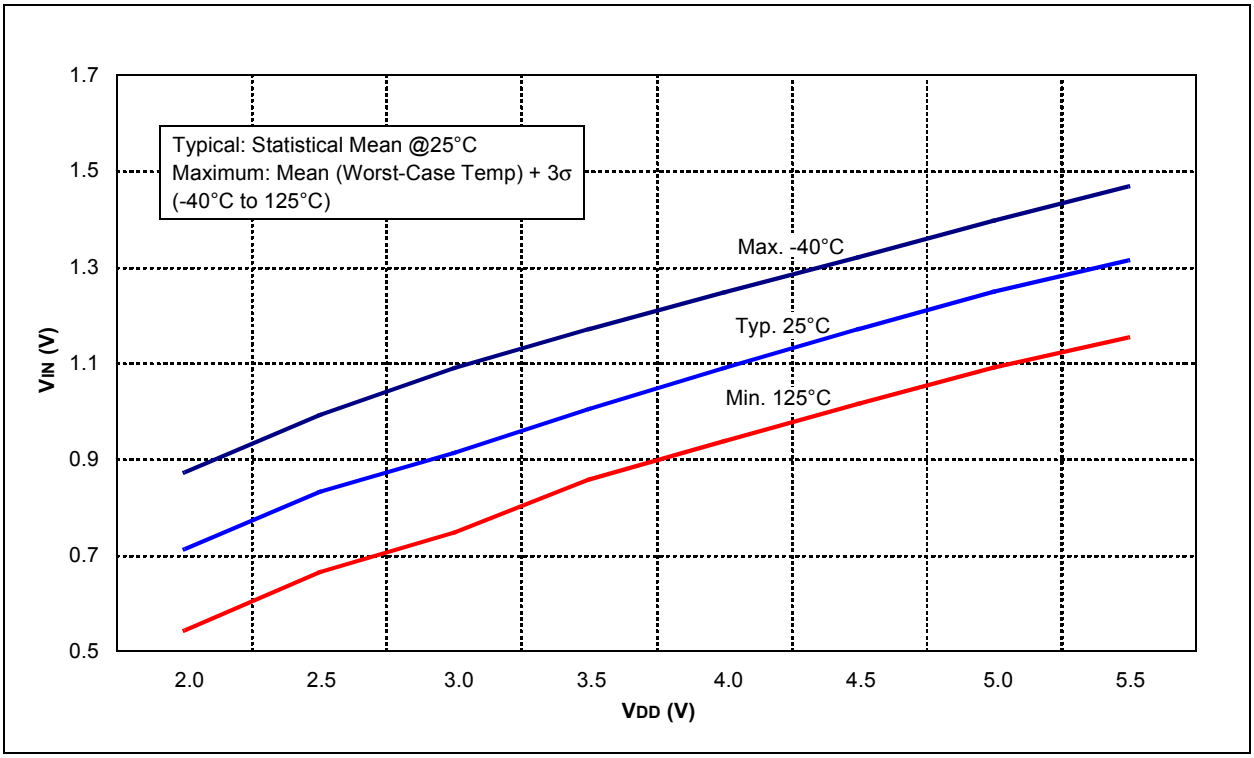
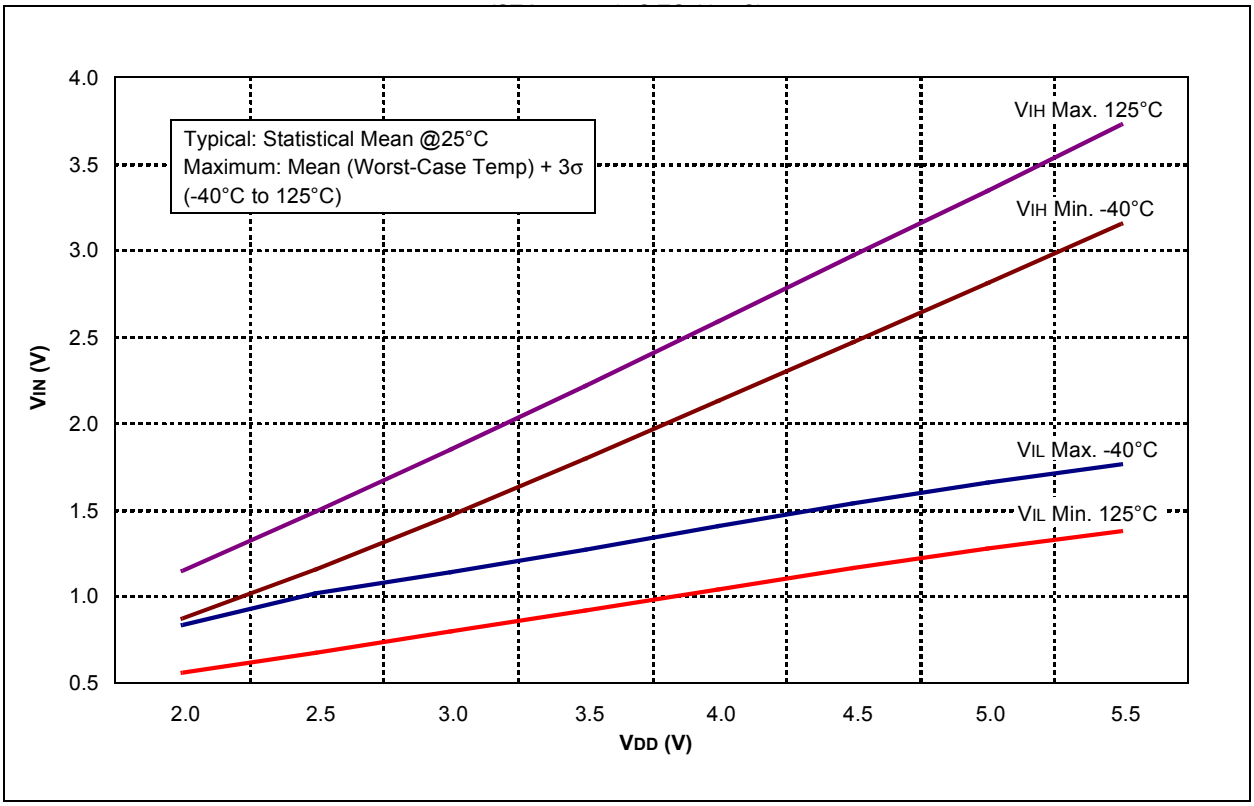


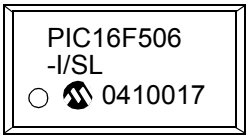

**FIGURE 14-13: SCHMITT TRIGGER INPUT THRESHOLD VIN vs. VDD**



### 15.2 Package Marking Information (Cont'd)

14-Lead SOIC (3.90 mm)

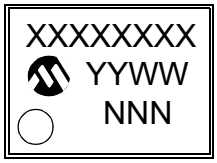

Example



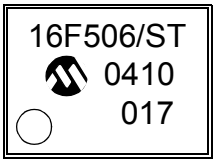

8-Lead MSOP


Example


14-Lead TSSOP (4.4 mm)




Example





© 2007 Microchip Technology Inc.

## T



## W



## Z




© 2007 Microchip Technology Inc.

## 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**