# PIC16C712/716

## 8-Bit CMOS Microcontrollers with A/D Converter and Capture/Compare/PWM

### Devices included in this Data Sheet:

- PIC16C712    • PIC16C716
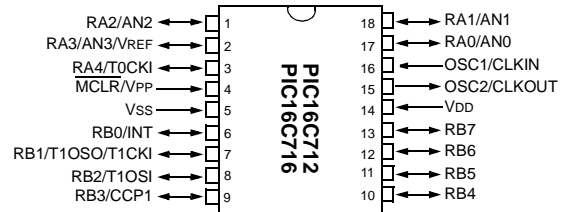
### Microcontroller Core Features:

- High-performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches which are two cycle
- Operating speed: DC - 20 MHz clock input
  DC - 200 ns instruction cycle

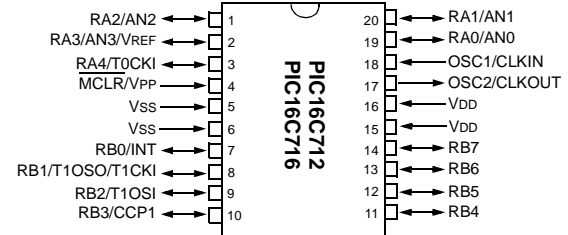| Device | Program Memory | Data Memory |
|--------|----------------|-------------|
| PIC16C712 | 1K | 128 |
| PIC16C716 | 2K | 128 |

- Interrupt capability
  (up to 7 internal/external interrupt sources)
- Eight level deep hardware stack
- Direct, indirect and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and
  Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Brown-out detection circuitry for
  Brown-out Reset (BOR)
- Programmable code-protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low-power, high-speed CMOS EPROM technology
- Fully static design
- In-Circuit Serial Programming™ (ICSP)
- Wide operating voltage range: 2.5V to 5.5V
- High Sink/Source Current 25/25 mA
- Commercial, Industrial and Extended temperature ranges
- Low-power consumption:
  - < 2 mA @ 5V, 4 MHz
  - 22.5 µA typical @ 3V, 32 kHz
  - < 1 µA typical standby current

### Pin Diagrams

#### 18-pin PDIP, SOIC, Windowed CERDIP



PIC16C712 / PIC16C716

| | Pin | | Pin | |
|---|---|---|---|---|
| RA2/AN2 | 1 | | 18 | RA1/AN1 |
| RA3/AN3/VREF | 2 | | 17 | RA0/AN0 |
| RA4/T0CKI | 3 | | 16 | OSC1/CLKIN |
| MCLR/VPP | 4 | | 15 | OSC2/CLKOUT |
| VSS | 5 | | 14 | VDD |
| RB0/INT | 6 | | 13 | RB7 |
| RB1/T1OSO/T1CKI | 7 | | 12 | RB6 |
| RB2/T1OSI | 8 | | 11 | RB5 |
| RB3/CCP1 | 9 | | 10 | RB4 |

#### 20-pin SSOP

PIC16C712 / PIC16C716

| | Pin | | Pin | |
|---|---|---|---|---|
| RA2/AN2 | 1 | | 20 | RA1/AN1 |
| RA3/AN3/VREF | 2 | | 19 | RA0/AN0 |
| RA4/T0CKI | 3 | | 18 | OSC1/CLKIN |
| MCLR/VPP | 4 | | 17 | OSC2/CLKOUT |
| VSS | 5 | | 16 | VDD |
| VSS | 6 | | 15 | VDD |
| RB0/INT | 7 | | 14 | RB7 |
| RB1/T1OSO/T1CKI | 8 | | 13 | RB6 |
| RB2/T1OSI | 9 | | 12 | RB5 |
| RB3/CCP1 | 10 | | 11 | RB4 |

### Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler can be incremented during sleep via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Capture, Compare, PWM module
- Capture is 16-bit, max. resolution is 12.5 ns, Compare is 16-bit, max. resolution is 200 ns, PWM maximum resolution is 10-bit
- 8-bit multi-channel Analog-to-Digital converter

# PIC16C712/716

| Key Features PICmicro™ Mid-Range Reference Manual (DS33023) | PIC16C712 | PIC16C716 |
|---|---|---|
| Operating Frequency | DC - 20 MHz | DC - 20 MHz |
| Resets (and Delays) | POR, BOR (PWRT, OST) | POR, BOR (PWRT, OST) |
| Program Memory (14-bit words) | 1K | 2K |
| Data Memory (bytes) | 128 | 128 |
| Interrupts | 7 | 7 |
| I/O Ports | Ports A,B | Ports A,B |
| Timers | 3 | 3 |
| Capture/Compare/PWM modules | 1 | 1 |
| 8-bit Analog-to-Digital Module | 4 input channels | 4 input channels |

## PIC16C7XX FAMILY OF DEVICES

| | | PIC16C710 | PIC16C71 | PIC16C711 | PIC16C712 | PIC16C715 | PIC16C716 | PIC16C72A | PIC16C73B |
|---|---|---|---|---|---|---|---|---|---|
| Clock | Maximum Frequency of Operation (MHz) | 20 | 20 | 20 | 20 | 20 | 20 | 20 | 20 |
| Memory | EPROM Program Memory (x14 words) | 512 | 1K | 1K | 1K | 2K | 2K | 2K | 4K |
| | Data Memory (bytes) | 36 | 36 | 68 | 128 | 128 | 128 | 128 | 192 |
| Peripherals | Timer Module(s) | TMR0 | TMR0 | TMR0 | TMR0 TMR1 TMR2 | TMR0 | TMR0 TMR1 TMR2 | TMR0 TMR1 TMR2 | TMR0 TMR1 TMR2 |
| | Capture/Compare/ PWM Module(s) | — | — | — | 1 | — | 1 | 1 | 2 |
| | Serial Port(s) (SPI/I$^2$C, USART) | — | — | — | — | — | — | SPI/I$^2$C | SPI/I$^2$C, USART |
| | A/D Converter (8-bit) Channels | 4 | 4 | 4 | 4 | 4 | 4 | 5 | 5 |
| Features | Interrupt Sources | 4 | 4 | 4 | 7 | 4 | 7 | 8 | 11 |
| | I/O Pins | 13 | 13 | 13 | 13 | 13 | 13 | 22 | 22 |
| | Voltage Range (Volts) | 2.5-6.0 | 3.0-6.0 | 2.5-6.0 | 2.5-5.5 | 2.5-5.5 | 2.5-5.5 | 2.5-5.5 | 2.5-5.5 |
| | In-Circuit Serial Programming | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| | Brown-out Reset | Yes | — | Yes | Yes | Yes | Yes | Yes | Yes |
| | Packages | 18-pin DIP, SOIC; 20-pin SSOP | 18-pin DIP, SOIC | 18-pin DIP, SOIC; 20-pin SSOP | 18-pin DIP, SOIC; 20-pin SSOP | 18-pin DIP, SOIC; 20-pin SSOP | 18-pin DIP, SOIC; 20-pin SSOP | 28-pin SDIP, SOIC, SSOP | 28-pin SDIP, SOIC |

**Preliminary**

**Table of Contents**

## To Our Valued Customers

**Most Current Data Sheet**

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number. e.g., DS30000A is version A of document DS30000.

**New Customer Notification System**

Register on our web site (www.microchip.com/cn) to receive the most current information on our products.

**Errata**

An errata sheet may exist for current devices, describing minor operational differences (from the data sheet) and recommended workarounds. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

• Microchip's Worldwide Web site; http://www.microchip.com
• Your local Microchip sales office (see last page)
• The Microchip Corporate Literature Center; U.S. FAX: (480) 786-7277

When contacting a sales office or the literature center, please specify which device, revision of silicon and data sheet (include literature number) you are using.

**Corrections to this Data Sheet**

We constantly strive to improve the quality of all our products and documentation. We have spent a great deal of time to ensure that this document is correct. However, we realize that we may have missed a few things. If you find any information that is missing or appears in error, please:

• Fill out and mail in the reader response form in the back of this data sheet.
• E-mail us at webmaster@microchip.com.

We appreciate your assistance in making this a better document.

**NOTES:**

Preliminary

## 1.0    DEVICE OVERVIEW

This document contains device-specific information. Additional information may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023), which may be obtained from your local Microchip Sales Representative or downloaded from the Microchip website. The Reference Manual should be considered a complementary document to this data sheet, and is highly recommended reading for a better understanding of the device architecture and operation of the peripheral modules.

There are two devices (PIC16C712, PIC16C716) covered by this datasheet.

Figure 1-1 is the block diagram for both devices. The pinouts are listed in Table 1-1.

**FIGURE 1-1:    PIC16C712/716 BLOCK DIAGRAM**



**Note 1:** Higher order bits are from the STATUS register.
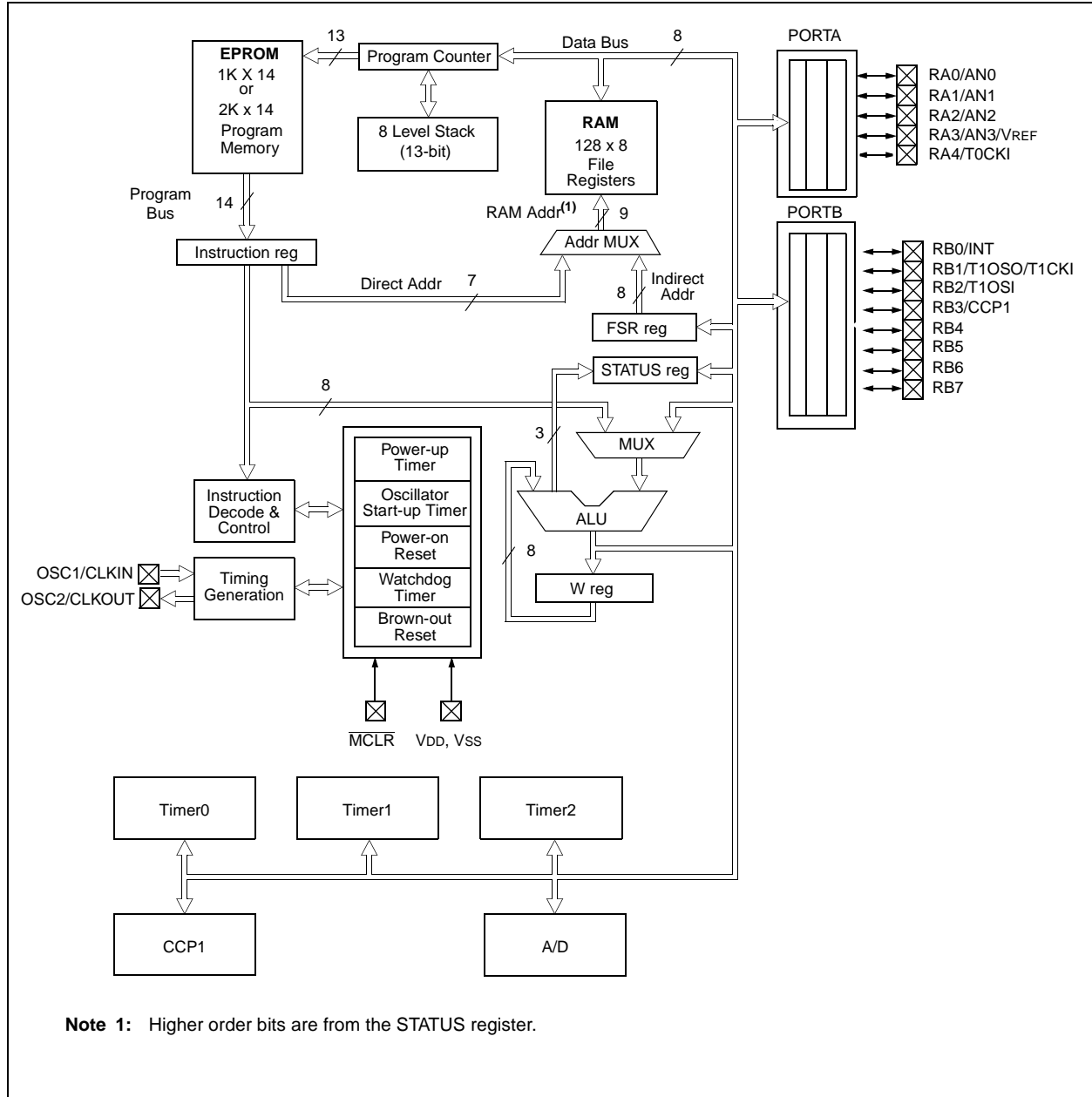
# PIC16C712/716

**TABLE 1-1    PIC16C712/716 PINOUT DESCRIPTION**

| Pin Name | PIC16C712/716 DIP, SOIC | SSOP | Pin Type | Buffer Type | Description |
|---|---|---|---|---|---|
| MCLR/VPP | 4 | 4 | | | |
| MCLR | | | I | ST | Master clear (reset) input. This pin is an active low reset to the device. |
| VPP | | | P | | Programming voltage input |
| OSC1/CLKIN | 16 | 18 | | | |
| OSC1 | | | I | ST | Oscillator crystal input or external clock source input. ST buffer when configured in RC mode. CMOS otherwise. |
| CLKIN | | | I | CMOS | External clock source input. |
| OSC2/CLKOUT | 15 | 17 | | | |
| OSC2 | | | O | — | Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. |
| CLKOUT | | | O | — | In RC mode, OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate. |
| | | | | | PORTA is a bi-directional I/O port. |
| RA0/AN0 | 17 | 19 | | | |
| RA0 | | | I/O | TTL | Digital I/O |
| AN0 | | | I | Analog | Analog input 0 |
| RA1/AN1 | 18 | 20 | | | |
| RA1 | | | I/O | TTL | Digital I/O |
| AN1 | | | I | Analog | Analog input 1 |
| RA2/AN2 | 1 | 1 | | | |
| RA2 | | | I/O | TTL | Digital I/O |
| AN2 | | | I | Analog | Analog input 2 |
| RA3/AN3/VREF | 2 | 2 | | | |
| RA3 | | | I/O | TTL | Digital I/O |
| AN3 | | | I | Analog | Analog input 3 |
| VREF | | | I | Analog | A/D Reference Voltage input. |
| RA4/T0CKI | 3 | 3 | | | |
| RA4 | | | I/O | ST/OD | Digital I/O. Open drain when configured as output. |
| T0CKI | | | I | ST | Timer0 external clock input |

Legend:  TTL = TTL-compatible input        CMOS = CMOS compatible input or output
         ST = Schmitt Trigger input with CMOS levels
         OD = Open drain output
         SM = SMBus compatible input. An external resistor is required if this pin is used as an output
         NPU = N-channel pull-up        PU = Weak internal pull-up
         No-P diode = No P-diode to VDD        AN = Analog input or output
         I = input        O = output
         P = Power        L = LCD Driver

**Preliminary**

**TABLE 1-1     PIC16C712/716 PINOUT DESCRIPTION (Cont.'d)**

| Pin Name | PIC16C712/716 DIP, SOIC | PIC16C712/716 SSOP | Pin Type | Buffer Type | Description |
|---|---|---|---|---|---|
| | | | | | PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-ups on all inputs. |
| RB0/INT | 6 | 7 | | | |
| RB0 | | | I/O | TTL | Digital I/O |
| INT | | | I | ST | External Interrupt |
| RB1/T1OSO/T1CKI | 7 | 8 | | | |
| RB1 | | | | | |
| T1OSO | | | I/O | TTL | Digital I/O |
| | | | O | — | Timer1 oscillator output. Connects to crystal in oscillator mode. |
| T1CKI | | | I | ST | Timer1 external clock input. |
| RB2/T1OSI | 8 | 9 | | | |
| RB2 | | | I/O | TTL | Digital I/O |
| T1OSI | | | I | — | Timer1 oscillator input. Connects to crystal in oscillator mode. |
| RB3/CCP1 | 9 | 10 | | | |
| RB3 | | | I/O | TTL | Digital I/O |
| CCP1 | | | I/O | ST | Capture1 input, Compare1 output, PWM1 output. |
| RB4 | 10 | 12 | I/O | TTL | Digital I/O Interrupt on change pin. |
| RB5 | 11 | 12 | I/O | TTL | Digital I/O Interrupt on change pin. |
| RB6 | 12 | 13 | I/O | TTL | Digital I/O Interrupt on change pin. |
| | | | I | ST | ICSP programming clock. |
| RB7 | 13 | 14 | I/O | TTL | Digital I/O Interrupt on change pin. |
| | | | I/O | ST | ICSP programming data. |
| VSS | 5 | 5, 6 | P | — | Ground reference for logic and I/O pins. |
| VDD | 14 | 15, 16 | P | — | Positive supply for logic and I/O pins. |

Legend:  TTL = TTL-compatible input          CMOS = CMOS compatible input or output
ST = Schmitt Trigger input with CMOS levels
OD = Open drain output
SM = SMBus compatible input. An external resistor is required if this pin is used as an output
NPU = N-channel pull-up                  PU = Weak internal pull-up
No-P diode = No P-diode to VDD           AN = Analog input or output
I = input                                O = output
P = Power                                L = LCD Driver

NOTES:

Preliminary

## 2.0    MEMORY ORGANIZATION

There are two memory blocks in each of these PICmicro® microcontroller devices. Each block (Program Memory and Data Memory) has its own bus so that concurrent access can occur.

Additional information on device memory may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023).

### 2.1    Program Memory Organization

The PIC16C712/716 has a 13-bit program counter capable of addressing an 8K x 14 program memory space. PIC16C712 has 1K x 14 words of program memory and PIC16C716 has 2K x 14 words of program memory. Accessing a location above the physically implemented address will cause a wraparound.

The reset vector is at 0000h and the interrupt vector is at 0004h.

FIGURE 2-1:    PROGRAM MEMORY MAP
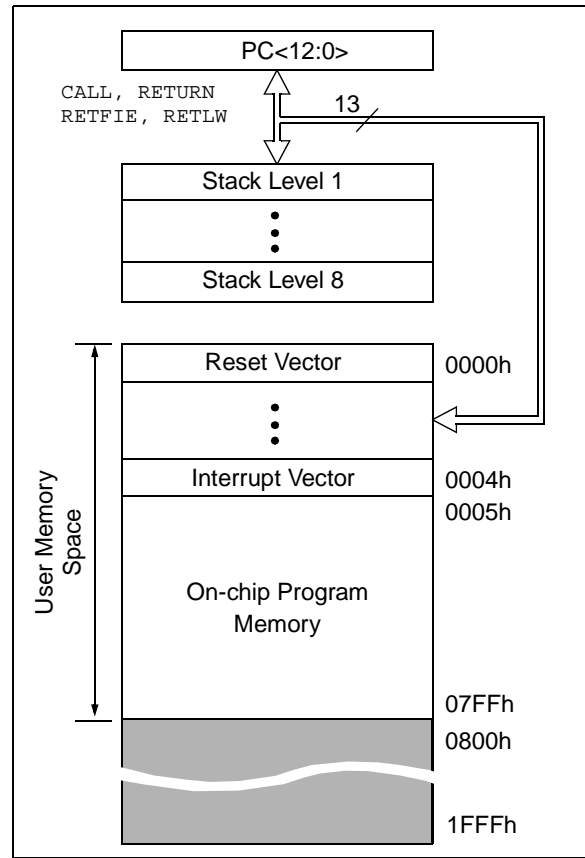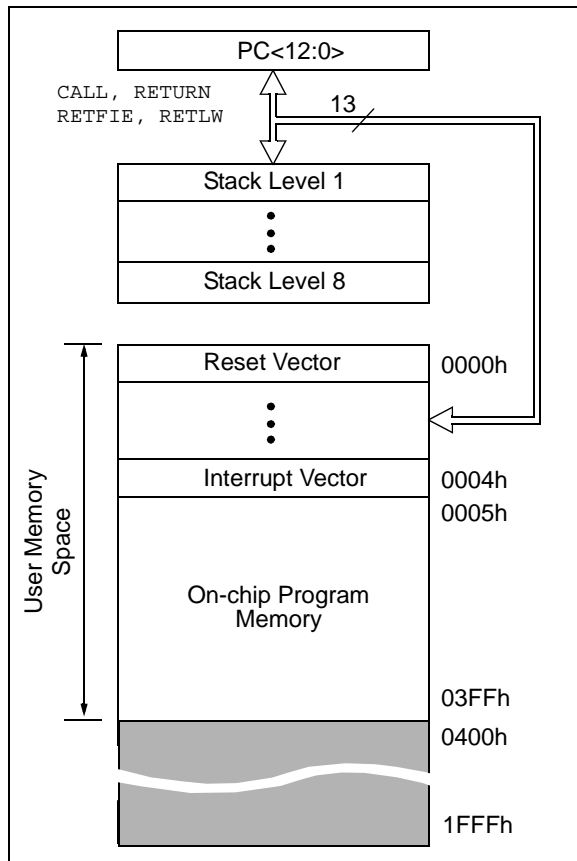                       AND STACK OF THE
                       PIC16C712



FIGURE 2-2:    PROGRAM MEMORY MAP
                       AND STACK OF PIC16C716

## 2.2 Data Memory Organization

The data memory is partitioned into multiple banks which contain the General Purpose Registers and the Special Function Registers. Bits RP1 and RP0 are the bank select bits.

| RP1[1] | RP0 | (STATUS<6:5>) |
|--------|-----|----------------|

= 00 → Bank0
= 01 → Bank1
= 10 → Bank2 (not implemented)
= 11 → Bank3 (not implemented)

**Note 1:** Maintain this bit clear to ensure upward compatibility with future products.

Each bank extends up to 7Fh (128 bytes). The lower locations of each bank are reserved for the Special Function Registers. Above the Special Function Registers are General Purpose Registers, implemented as static RAM. All implemented banks contain special function registers. Some "high use" special function registers from one bank may be mirrored in another bank for code reduction and quicker access.

### 2.2.1 GENERAL PURPOSE REGISTER FILE

The register file can be accessed either directly, or indirectly through the File Select Register FSR (Section 2.5).

**FIGURE 2-3: REGISTER FILE MAP**

| File Address | Bank 0 | Bank 1 | File Address |
|--------------|--------|--------|--------------|
| 00h | INDF[1] | INDF[1] | 80h |
| 01h | TMR0 | OPTION_REG | 81h |
| 02h | PCL | PCL | 82h |
| 03h | STATUS | STATUS | 83h |
| 04h | FSR | FSR | 84h |
| 05h | PORTA | TRISA | 85h |
| 06h | PORTB | TRISB | 86h |
| 07h | DATACCP | TRISCCP | 87h |
| 08h | | | 88h |
| 09h | | | 89h |
| 0Ah | PCLATH | PCLATH | 8Ah |
| 0Bh | INTCON | INTCON | 8Bh |
| 0Ch | PIR1 | PIE1 | 8Ch |
| 0Dh | | | 8Dh |
| 0Eh | TMR1L | PCON | 8Eh |
| 0Fh | TRM1H | | 8Fh |
| 10h | T1CON | | 90h |
| 11h | TRM2 | | 91h |
| 12h | T2CON | PR2 | 92h |
| 13h | | | 93h |
| 14h | | | 94h |
| 15h | CCPR1L | | 95h |
| 16h | CCPR1H | | 96h |
| 17h | CCP1CON | | 97h |
| 18h | | | 98h |
| 19h | | | 99h |
| 1Ah | | | 9Ah |
| 1Bh | | | 9Bh |
| 1Ch | | | 9Ch |
| 1Dh | | | 9Dh |
| 1Eh | ADRES | | 9Eh |
| 1Fh | ADCON0 | ADCON1 | 9Fh |
| 20h | | General Purpose Registers 32 Bytes | A0h |
| | General Purpose Registers 96 Bytes | | BFh |
| | | | C0h |
| 7Fh | | | FFh |

Unimplemented data memory locations, read as '0'.

**Note 1:** Not a physical register.

### 2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and Peripheral Modules for controlling the desired operation of the device. These registers are implemented as static RAM. A list of these registers is give in Table 2-1.

The special function registers can be classified into two sets; core (CPU) and peripheral. Those registers associated with the core functions are described in detail in this section. Those related to the operation of the peripheral features are described in detail in that peripheral feature section.

### TABLE 2-1 SPECIAL FUNCTION REGISTER SUMMARY

| Addr | 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 (4) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bank 0** | | | | | | | | | | | |
| 00h | INDF[1] | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | 0000 0000 | 0000 0000 |
| 01h | TMR0 | Timer0 module's register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 02h | PCL[1] | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 0000 0000 |
| 03h | STATUS[1] | IRP[4] | RP1[4] | RP0 | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C | rr01 1xxx | rr0q quuu |
| 04h | FSR[1] | Indirect data memory address pointer | | | | | | | | xxxx xxxx | uuuu uuuu |
| 05h | PORTA[5,6] | — | — | —[7] | PORTA Data Latch when written: PORTA pins when read | | | | | --xx xxxx | --xu uuuu |
| 06h | PORTB[5,6] | PORTB Data Latch when written: PORTB pins when read | | | | | | | | xxxx xxxx | uuuu uuuu |
| 07h | DATACCP | —[7] | —[7] | —[7] | —[7] | —[7] | DCCP | —[7] | DT1CK | xxxx xxxx | xxxx xuxu |
| 08h-09h | — | Unimplemented | | | | | | | | — | — |
| 0Ah | PCLATH[1,2] | — | — | — | Write Buffer for the upper 5 bits of the Program Counter | | | | | ---0 0000 | ---0 0000 |
| 0Bh | INTCON[1] | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | — | CCP1IF | TMR2IF | TMR1IF | -0-- 0000 | -0-- 0000 |
| 0Dh | — | Unimplemented | | | | | | | | — | — |
| 0Eh | TMR1L | Holding register for the Least Significant Byte of the 16-bit TMR1 register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 0Fh | TMR1H | Holding register for the Most Significant Byte of the 16-bit TMR1 register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 10h | T1CON | — | — | T1CKPS1 | T1CKPS0 | T1OSCEN | $\overline{T1SYNC}$ | TMR1CS | TMR1ON | --00 0000 | --uu uuuu |
| 11h | TMR2 | Timer2 module's register | | | | | | | | 0000 0000 | 0000 0000 |
| 12h | T2CON | — | TOUTPS3 | TOUTPS2 | TOUTPS1 | TOUTPS0 | TMR2ON | T2CKPS1 | T2CKPS0 | -000 0000 | -000 0000 |
| 13h-14h | | | | | | | | | | | |
| 15h | CCPR1L | Capture/Compare/PWM Register1 (LSB) | | | | | | | | xxxx xxxx | uuuu uuuu |
| 16h | CCPR1H | Capture/Compare/PWM Register1 (MSB) | | | | | | | | xxxx xxxx | uuuu uuuu |
| 17h | CCP1CON | — | — | DC1B1 | DC1B0 | CCP1M3 | CCP1M2 | CCP1M1 | CCP1M0 | --00 0000 | --00 0000 |
| 18h-1Dh | — | Unimplemented | | | | | | | | — | — |
| 1Eh | ADRES | A/D Result Register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 1Fh | ADCON0 | ADCS1 | ADCS0 | CHS2 | CHS1 | CHS0 | GO/$\overline{DONE}$ | — | ADON | 0000 00-0 | 0000 00-0 |

Legend: x = unknown, u = unchanged, q = value depends on condition, – = unimplemented, read as '0',
Shaded locations are unimplemented, read as '0'.

**Note 1:** These registers can be addressed from either bank.

**2:** The upper byte of the program counter is not directly accessible. PCLATH is a holding register for PC<12:8> whose contents are transferred to the upper byte of the program counter.

**3:** Other (non power-up) resets include: external reset through $\overline{MCLR}$ and the Watchdog Timer Reset.

**4:** The IRP and RP1 bits are reserved. Always maintain these bits clear.

**5:** On any device reset, these pins are configured as inputs.

**6:** This is the value that will be in the port output latch.

**7:** Reserved bits; Do Not Use.

# PIC16C712/716

**TABLE 2-1    SPECIAL FUNCTION REGISTER SUMMARY   (Cont.'d)**

| Addr | 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 (4) |
|------|------|-------|-------|-------|-------|-------|-------|-------|-------|--------------------|-------------------------------|
| **Bank 1** | | | | | | | | | | | |
| 80h | INDF[1] | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | 0000 0000 | 0000 0000 |
| 81h | OPTION_REG | RBPU | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |
| 82h | PCL[1] | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 0000 0000 |
| 83h | STATUS[1] | IRP[4] | RP1[4] | RP0 | TO | PD | Z | DC | C | rr01 1xxx | rr0q quuu |
| 84h | FSR[1] | Indirect data memory address pointer | | | | | | | | xxxx xxxx | uuuu uuuu |
| 85h | TRISA | — | — | —[7] | PORTA Data Direction Register | | | | | --x1 1111 | --x1 1111 |
| 86h | TRISB | PORTB Data Direction Register | | | | | | | | 1111 1111 | 1111 1111 |
| 87h | TRISCCP | —[7] | —[7] | —[7] | —[7] | —[7] | TCCP | —[7] | TT1CK | xxxx x1x1 | xxxx x1x1 |
| 88h-89h | — | Unimplemented | | | | | | | | — | — |
| 8Ah | PCLATH[1,2] | — | — | — | Write Buffer for the upper 5 bits of the Program Counter | | | | | ---0 0000 | ---0 0000 |
| 8Bh | INTCON[1] | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 8Ch | PIE1 | — | ADIE | — | — | — | CCP1IE | TMR2IE | TMR1IE | -0-- -000 | -0-- -000 |
| 8Dh | — | Unimplemented | | | | | | | | — | — |
| 8Eh | PCON | — | — | — | — | — | — | POR | BOR | ---- --qq | ---- --uu |
| 8Fh-91h | — | Unimplemented | | | | | | | | — | — |
| 92h | PR2 | Timer2 Period Register | | | | | | | | 1111 1111 | 1111 1111 |
| 93h-9Eh | — | Unimplemented | | | | | | | | — | — |
| 9Fh | ADCON1 | — | — | — | — | — | PCFG2 | PCFG1 | PCFG0 | ---- -000 | ---- -000 |

Legend:    x = unknown, u = unchanged, q = value depends on condition,  – = unimplemented, read as '0',
Shaded locations are unimplemented, read as '0'.

**Note 1:** These registers can be addressed from either bank.
**2:** The upper byte of the program counter is not directly accessible. PCLATH is a holding register for PC<12:8> whose contents are transferred to the upper byte of the program counter.
**3:** Other (non power-up) resets include: external reset through MCLR and the Watchdog Timer Reset.
**4:** The IRP and RP1 bits are reserved. Always maintain these bits clear.
**5:** On any device reset, these pins are configured as inputs.
**6:** This is the value that will be in the port output latch.
**7:** Reserved bits;  Do Not Use.

**Preliminary**

### 2.2.2.1 STATUS REGISTER

The STATUS register, shown in Figure 2-4, contains the arithmetic status of the ALU, the RESET status and the bank select bits for data memory.

The STATUS register can be the destination for any instruction, as with 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 uluu (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 the Z, C or DC bits from the STATUS register. For other instructions, not affecting any status bits, see the "Instruction Set Summary."

**Note 1:** These devices do not use bits IRP and RP1 (STATUS<7:6>). Maintain these bits clear to ensure upward compatibility with future products.

**Note 2:** The C and DC bits operate as a $\overline{\text{borrow}}$ and $\overline{\text{digit borrow}}$ bit, respectively, in subtraction. See the SUBLW and SUBWF instructions for examples.

### FIGURE 2-4: STATUS REGISTER (ADDRESS 03h, 83h)

| R/W-0 | R/W-0 | 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 |

bit7 ............ bit0

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

bit 7: **IRP**: Register Bank Select bit (used for indirect addressing)
1 = Bank 2, 3 (100h - 1FFh) - not implemented, maintain clear
0 = Bank 0, 1 (00h - FFh) - not implemented, maintain clear

bit 6-5: **RP1:RP0**: Register Bank Select bits (used for direct addressing)
01 = Bank 1 (80h - FFh)
00 = Bank 0 (00h - 7Fh)
Each bank is 128 bytes
**Note:** RP1 = not implemented, maintain clear

bit 4: $\overline{\text{TO}}$: Time-out bit
1 = After power-up, CLRWDT instruction, or SLEEP instruction
0 = A WDT time-out occurred

bit 3: $\overline{\text{PD}}$: 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/$\overline{\text{borrow}}$ bit (ADDWF, ADDLW, SUBLW, SUBWF instructions) (for $\overline{\text{borrow}}$ 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/$\overline{\text{borrow}}$ bit (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:** For $\overline{\text{borrow}}$ 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 or low order bit of the source register.
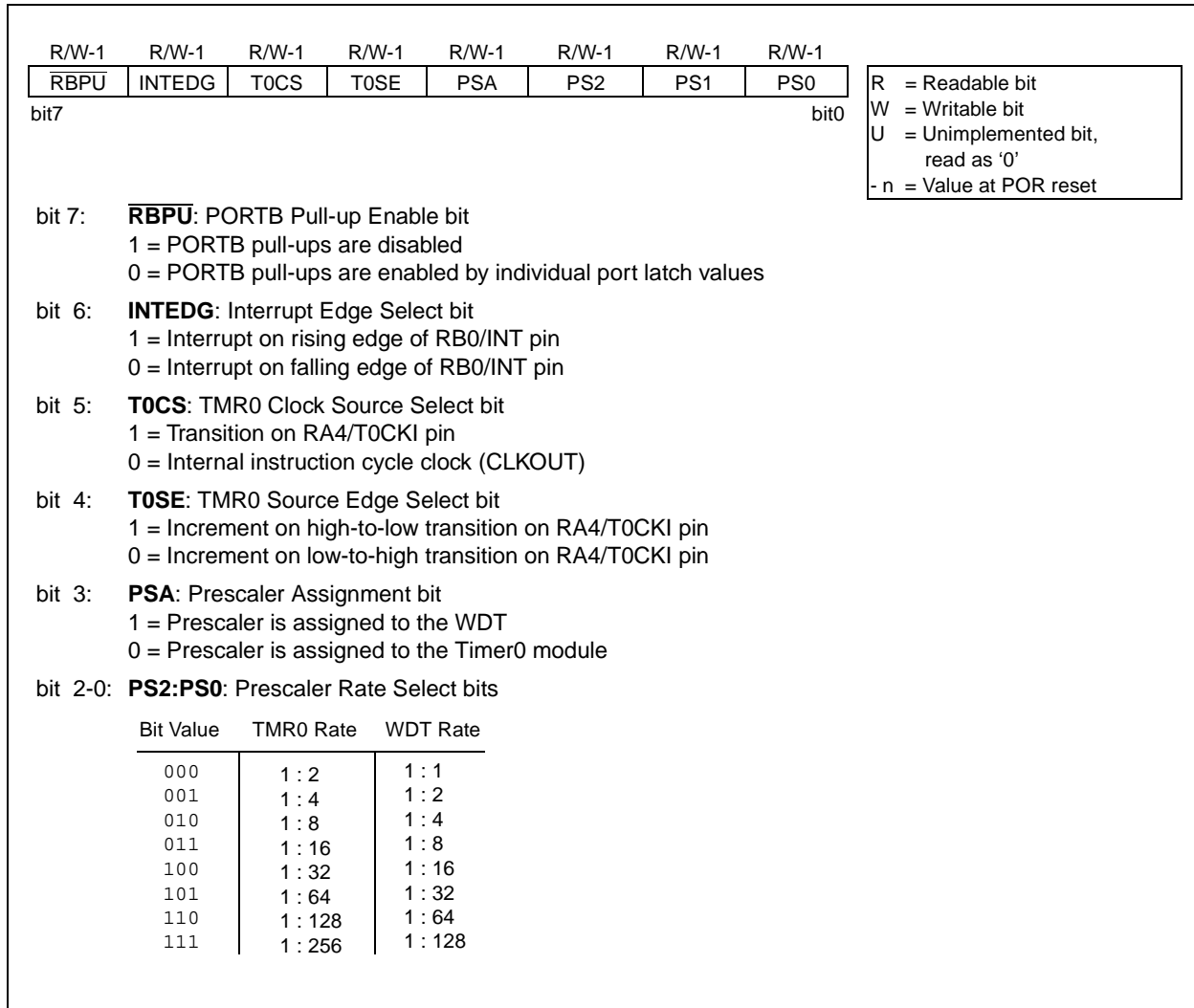
# PIC16C712/716

### 2.2.2.2 OPTION_REG REGISTER

The OPTION_REG register is a readable and writable register, which contains various control bits to configure the TMR0 prescaler/WDT postscaler (single assignable register known also as the prescaler), the External INT Interrupt, TMR0 and the weak pull-ups on PORTB.

| **Note:** | To achieve a 1:1 prescaler assignment for the TMR0 register, assign the prescaler to the Watchdog Timer. |
|---|---|

**FIGURE 2-5:    OPTION_REG REGISTER (ADDRESS 81h)**

| R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| RBPU | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 |

bit7                                                                                          bit0

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

bit 7:  **RBPU**: PORTB Pull-up Enable bit
1 = PORTB pull-ups are disabled
0 = PORTB pull-ups are enabled by individual port latch values

bit 6:  **INTEDG**: Interrupt Edge Select bit
1 = Interrupt on rising edge of RB0/INT pin
0 = Interrupt on falling edge of RB0/INT pin

bit 5:  **T0CS**: TMR0 Clock Source Select bit
1 = Transition on RA4/T0CKI pin
0 = Internal instruction cycle clock (CLKOUT)

bit 4:  **T0SE**: TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on RA4/T0CKI pin
0 = Increment on low-to-high transition on RA4/T0CKI pin

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

bit 2-0:  **PS2:PS0**: Prescaler Rate Select bits

| Bit Value | TMR0 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 |

**Preliminary**                                    © 1999 Microchip Technology Inc.

2.2.2.3    INTCON REGISTER

The INTCON Register is a readable and writable register which contains various enable and flag bits for the TMR0 register overflow, RB Port change and External RB0/INT pin interrupts.

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

**FIGURE 2-6:    INTCON REGISTER (ADDRESS 0Bh, 8Bh)**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-x |
|---|---|---|---|---|---|---|---|
| GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF |

bit7                                                                                                      bit0

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

bit 7:    **GIE:** Global Interrupt Enable bit
1 = Enables all un-masked interrupts
0 = Disables all interrupts

bit 6:    **PEIE:** Peripheral Interrupt Enable bit
1 = Enables all un-masked peripheral interrupts
0 = Disables all peripheral interrupts

bit 5:    **T0IE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt

bit 4:    **IINTE:** RB0/INT External Interrupt Enable bit
1 = Enables the RB0/INT external interrupt
0 = Disables the RB0/INT external interrupt

bit 3:    **RBIE:** RB Port Change Interrupt Enable bit
1 = Enables the RB port change interrupt
0 = Disables the RB port change interrupt

bit 2:    **T0IF:** TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1:    **INTF:** RB0/INT External Interrupt Flag bit
1 = The RB0/INT external interrupt occurred (must be cleared in software)
0 = The RB0/INT external interrupt did not occur

bit 0:    **RBIF:** RB Port Change Interrupt Flag bit
1 = At least one of the RB7:RB4 pins changed state (must be cleared in software)
0 = None of the RB7:RB4 pins have changed state

### 2.2.2.4    PIE1 REGISTER

This register contains the individual enable bits for the peripheral interrupts.

> **Note:**    Bit PEIE (INTCON<6>) must be set to enable any peripheral interrupt.

**FIGURE 2-7:    PIE1 REGISTER (ADDRESS 8Ch)**

| U-0 | R/W-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|-----|-------|-----|-----|-------|-------|-------|-------|
| — | ADIE | — | — | — | CCP1IE | TMR2IE | TMR1IE |

bit7                                                                 bit0

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

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

bit 6:    **ADIE**: A/D Converter Interrupt Enable bit
1 = Enables the A/D interrupt
0 = Disables the A/D interrupt

bit 5-3:    **Unimplemented**: Read as '0'

bit 2:    **CCP1IE**: CCP1 Interrupt Enable bit
1 = Enables the CCP1 interrupt
0 = Disables the CCP1 interrupt

bit 1:    **TMR2IE**: TMR2 to PR2 Match Interrupt Enable bit
1 = Enables the TMR2 to PR2 match interrupt
0 = Disables the TMR2 to PR2 match interrupt

bit 0:    **TMR1IE**: TMR1 Overflow Interrupt Enable bit
1 = Enables the TMR1 overflow interrupt
0 = Disables the TMR1 overflow interrupt

### 2.2.2.5    PIR1 REGISTER

This register contains the individual flag bits for the peripheral interrupts.

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

**FIGURE 2-8:    PIR1 REGISTER (ADDRESS 0Ch)**

| U-0 | R/W-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | ADIF | — | — | — | CCP1IF | TMR2IF | TMR1IF |

bit7                                                                                       bit0

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

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

bit 6:    **ADIF**: A/D Converter Interrupt Flag bit
1 = An A/D conversion completed (must be cleared in software)
0 = The A/D conversion is not complete

bit 5-3:  **Unimplemented**: Read as '0'

bit 2:    **CCP1IF**: CCP1 Interrupt Flag bit
Capture Mode
1 = A TMR1 register capture occurred (must be cleared in software)
0 = No TMR1 register capture occurred
Compare Mode
1 = A TMR1 register compare match occurred (must be cleared in software)
0 = No TMR1 register compare match occurred
PWM Mode
Unused in this mode

bit 1:    **TMR2IF**: TMR2 to PR2 Match Interrupt Flag bit
1 = TMR2 to PR2 match occurred (must be cleared in software)
0 = No TMR2 to PR2 match occurred

bit 0:    **TMR1IF**: TMR1 Overflow Interrupt Flag bit
1 = TMR1 register overflowed (must be cleared in software)
0 = TMR1 register did not overflow

2.2.2.6    PCON REGISTER

The Power Control (PCON) register contains a flag bit to allow differentiation between a Power-on Reset (POR) to an external $\overline{\text{MCLR}}$ Reset or WDT Reset. These devices contain an additional bit to differentiate a Brown-out Reset condition from a Power-on Reset condition.

| Note: | If the BODEN configuration bit is set, $\overline{\text{BOR}}$ is '1' on Power-on Reset. If the BODEN configuration bit is clear, $\overline{\text{BOR}}$ is unknown on Power-on Reset. |
| --- | --- |
| | The $\overline{\text{BOR}}$ status bit is a "don't care" and is not necessarily predictable if the brown-out circuit is disabled (the BODEN configuration bit is clear). $\overline{\text{BOR}}$ must then be set by the user and checked on subsequent resets to see if it is clear, indicating a brown-out has occurred. |

**FIGURE 2-9:    PCON REGISTER (ADDRESS 8Eh)**

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-q |
| --- | --- | --- | --- | --- | --- | --- | --- |
| — | — | — | — | — | — | $\overline{\text{POR}}$ | $\overline{\text{BOR}}$ |

bit7                                                                                                      bit0

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

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

bit 1:    $\overline{\text{POR}}$: Power-on Reset Status bit
1 = No Power-on Reset occurred
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

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

## 2.3    PCL and PCLATH

The program counter (PC) specifies the address of the instruction to fetch for execution. The PC is 13 bits wide. The low byte is called the PCL register. This register is readable and writable. The high byte is called the PCH register. This register contains the PC<12:8> bits and is not directly readable or writable. All updates to the PCH register go through the PCLATH register.

### 2.3.1    STACK

The stack allows a combination of up to 8 program calls and interrupts to occur. The stack contains the return address from this branch in program execution.

Midrange devices have an 8 level deep x 13-bit wide hardware stack. The stack space is not part of either program or data space and the stack pointer is not readable or writable. The PC is PUSHed onto the stack when a CALL instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not modified when the stack is PUSHed or POPed.

After the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

## 2.4    Program Memory Paging

The CALL and GOTO instructions provide 11 bits of address to allow branching within any 2K program memory page. When doing a CALL or GOTO instruction, the upper bit of the address is provided by PCLATH<3>. When doing a CALL or GOTO instruction, the user must ensure that the page select bit is programmed so that the desired program memory page is addressed. If a return from a CALL instruction (or interrupt) is executed, the entire 13-bit PC is pushed onto the stack. Therefore, manipulation of the PCLATH<3> bit is not required for the return instructions (which POPs the address from the stack).

## 2.5 Indirect Addressing, INDF and FSR Registers

The INDF register is not a physical register. Addressing INDF actually addresses the register whose address is contained in the FSR register (FSR is a *pointer*). This is indirect addressing.

### EXAMPLE 2-1: INDIRECT ADDRESSING

- Register file 05 contains the value 10h
- Register file 06 contains the value 0Ah
- Load the value 05 into the FSR register
- A read of the INDF register will return the value of 10h
- Increment the value of the FSR register by one (FSR = 06)
- A read of the INDR register now will return the value of 0Ah.

Reading INDF itself indirectly (FSR = 0) will produce 00h. Writing to the INDF register indirectly results in a no-operation (although STATUS bits may be affected).

A simple program to clear RAM locations 20h-2Fh using indirect addressing is shown in Example 2-2.

### EXAMPLE 2-2: HOW TO CLEAR RAM USING INDIRECT ADDRESSING

```
          movlw  0x20  ;initialize pointer
          movwf  FSR   ;  to RAM
NEXT      clrf   INDF  ;clear INDF register
          incf   FSR   ;inc pointer
          btfss  FSR,4 ;all done?
          goto   NEXT  ;NO, clear next
CONTINUE
          :            ;YES, continue
```

An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit (STATUS<7>), as shown in Figure 2-10. However, IRP is not used in the PIC16C712/716.

### FIGURE 2-10: DIRECT/INDIRECT ADDRESSING



**Note 1:** For register file map detail see Figure 2-3.

    **2:** Maintain clear for upward compatibility with future products.

    **3:** Not implemented.

    **Preliminary**    
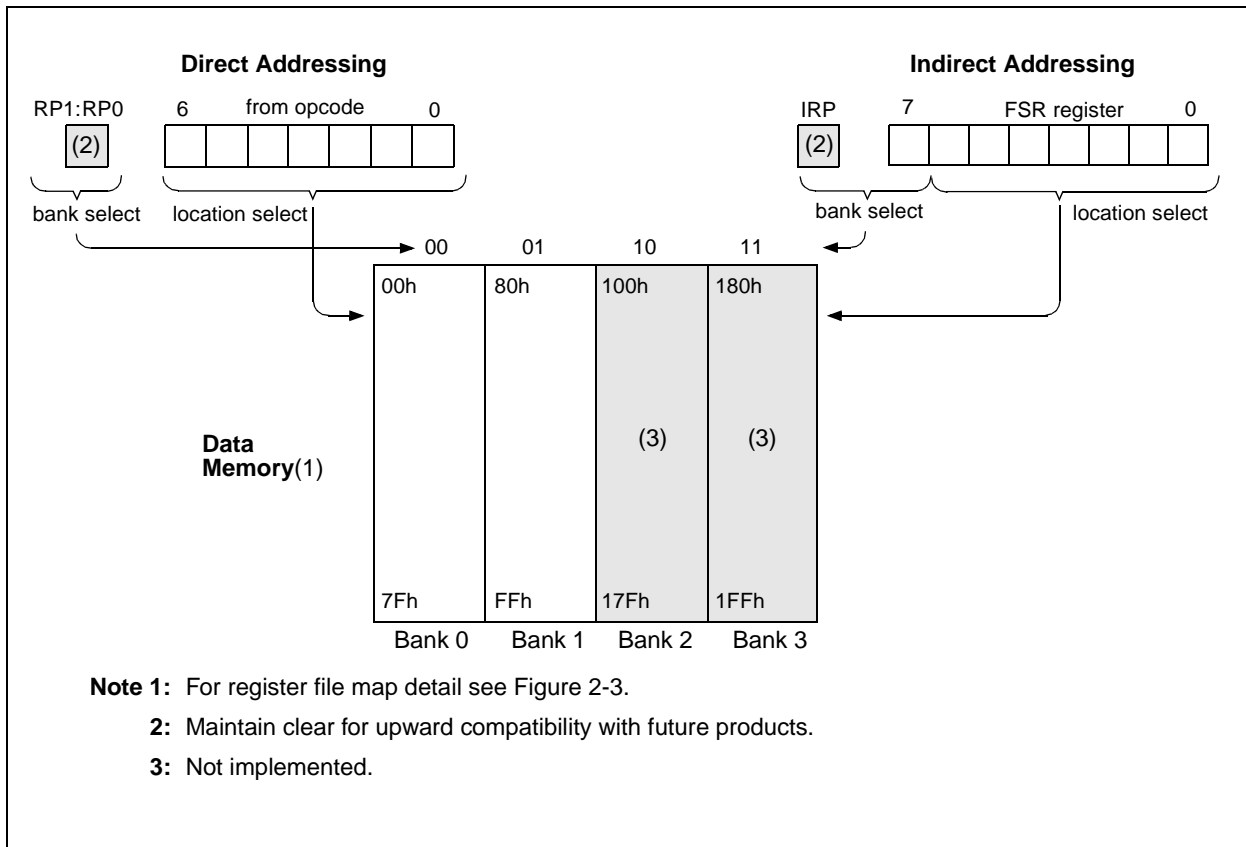
# 3.0    I/O PORTS

Some pins for these I/O ports are multiplexed with an alternate function for the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

Additional information on I/O ports may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023).

## 3.1    PORTA and the TRISA Register

PORTA is a 5-bit wide bi-directional port. The corresponding data direction register is TRISA. Setting a TRISA bit (=1) will make the corresponding PORTA pin an input, (i.e., put the corresponding output driver in a hi-impedance mode). Clearing a TRISA bit (=0) will make the corresponding PORTA pin an output, (i.e., put the contents of the output latch on the selected pin).

Reading the PORTA register reads the status of the pins whereas writing to it will write to the port latch. All write operations are read-modify-write operations. Therefore a write to a port implies that the port pins are read, the value is modified, and then written to the port data latch.

Pin RA4 is multiplexed with the Timer0 module clock input to become the RA4/T0CKI pin. The RA4/T0CKI pin is a Schmitt Trigger input and an open drain output. All other RA port pins have TTL input levels and full CMOS output drivers.

PORTA pins, RA3:0, are multiplexed with analog inputs and analog VREF input. The operation of each pin is selected by clearing/setting the control bits in the ADCON1 register (A/D Control Register1).

| Note: | On a Power-on Reset, these pins are configured as analog inputs and read as '0'. |
|---|---|

The TRISA register controls the direction of the RA pins, even when they are being used as analog inputs. The user must ensure the bits in the TRISA register are maintained set when using them as analog inputs.

### EXAMPLE 3-1:    INITIALIZING PORTA

```
BCF    STATUS, RP0  ;
CLRF   PORTA        ; Initialize PORTA by
                    ; clearing output
                    ; data latches
BSF    STATUS, RP0  ; Select Bank 1
MOVLW  0xEF         ; Value used to
                    ; initialize data
                    ; direction
MOVWF  TRISA        ; Set RA<3:0> as inputs
                    ; RA<4> as outputs
BCF    STATUS, RP0  ; Return to Bank 0
```

### FIGURE 3-1:    BLOCK DIAGRAM OF RA3:RA0
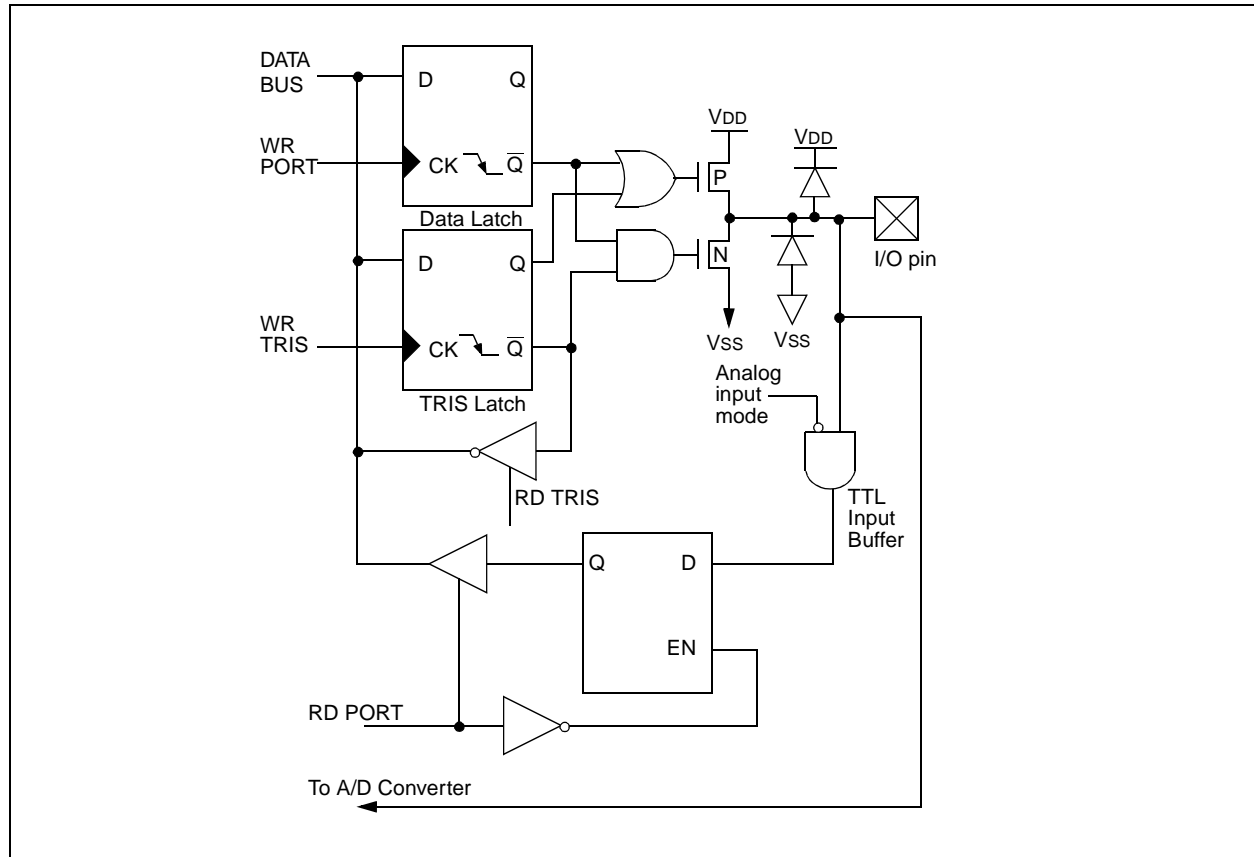
# PIC16C712/716

**FIGURE 3-2:** **BLOCK DIAGRAM OF RA4/T0CKI PIN**
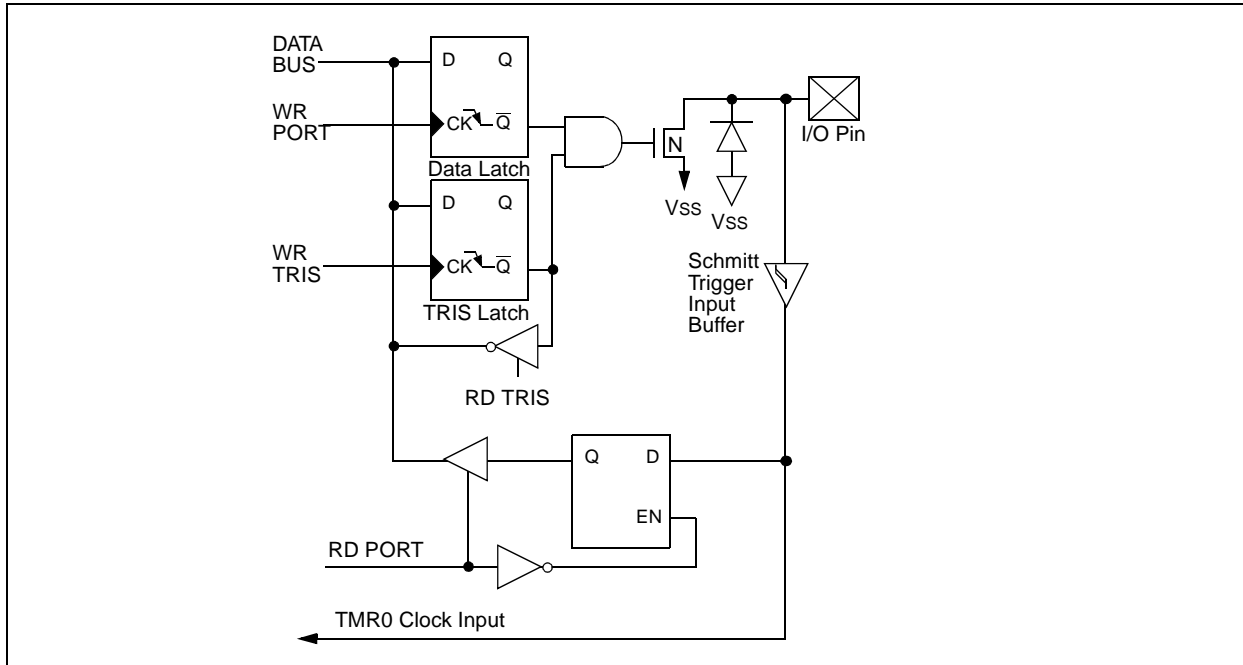


**TABLE 3-1** **PORTA FUNCTIONS**

| Name | Bit# | Buffer | Function |
|------|------|--------|----------|
| RA0/AN0 | bit0 | TTL | Input/output or analog input |
| RA1/AN1 | bit1 | TTL | Input/output or analog input |
| RA2/AN2 | bit2 | TTL | Input/output or analog input |
| RA3/AN3/VREF | bit3 | TTL | Input/output or analog input or VREF |
| RA4/T0CKI | bit4 | ST | Input/output or external clock input for Timer0 Output is open drain type |

Legend:  TTL = TTL input, ST = Schmitt Trigger input

**TABLE 3-2** **SUMMARY OF REGISTERS ASSOCIATED WITH PORTA**

| Address | 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 |
|---------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------------------|---------------------------|
| 05h | PORTA | — | — | —(1) | RA4 | RA3 | RA2 | RA1 | RA0 | --xx xxxx | --xu uuuu |
| 85h | TRISA | — | — | —(1) | PORTA Data Direction Register | | | | | --11 1111 | --11 1111 |
| 9Fh | ADCON1 | — | — | — | — | — | PCFG2 | PCFG1 | PCFG0 | ---- -000 | ---- -000 |

Legend:  x = unknown, u = unchanged, – = unimplemented locations read as '0'. Shaded cells are not used by PORTA.

**Note 1:** Reserved bits; Do Not Use.

### 3.2 PORTB and the TRISB Register

PORTB is an 8-bit wide bi-directional port. The corresponding data direction register is TRISB. Setting a TRISB bit (=1) will make the corresponding PORTB pin an input, (i.e., put the corresponding output driver in a hi-impedance mode). Clearing a TRISB bit (=0) will make the corresponding PORTB pin an output, (i.e., put the contents of the output latch on the selected pin).

Each of the PORTB pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is performed by clearing bit $\overline{\text{RBPU}}$ (OPTION_REG<7>). The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset.

### EXAMPLE 3-1: INITIALIZING PORTB

```
BCF    STATUS, RP0  ;
CLRF   PORTB        ; Initialize PORTB by
                    ; clearing output
                    ; data latches
BSF    STATUS, RP0  ; Select Bank 1
MOVLW  0xCF         ; Value used to
                    ; initialize data
                    ; direction
MOVWF  TRISB        ; Set RB<3:0> as inputs
                    ; RB<5:4> as outputs
                    ; RB<7:6> as inputs
```

### FIGURE 3-3: BLOCK DIAGRAM OF RB0 PIN



Note 1: To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the RBPU bit (OPTION_REG<7>).
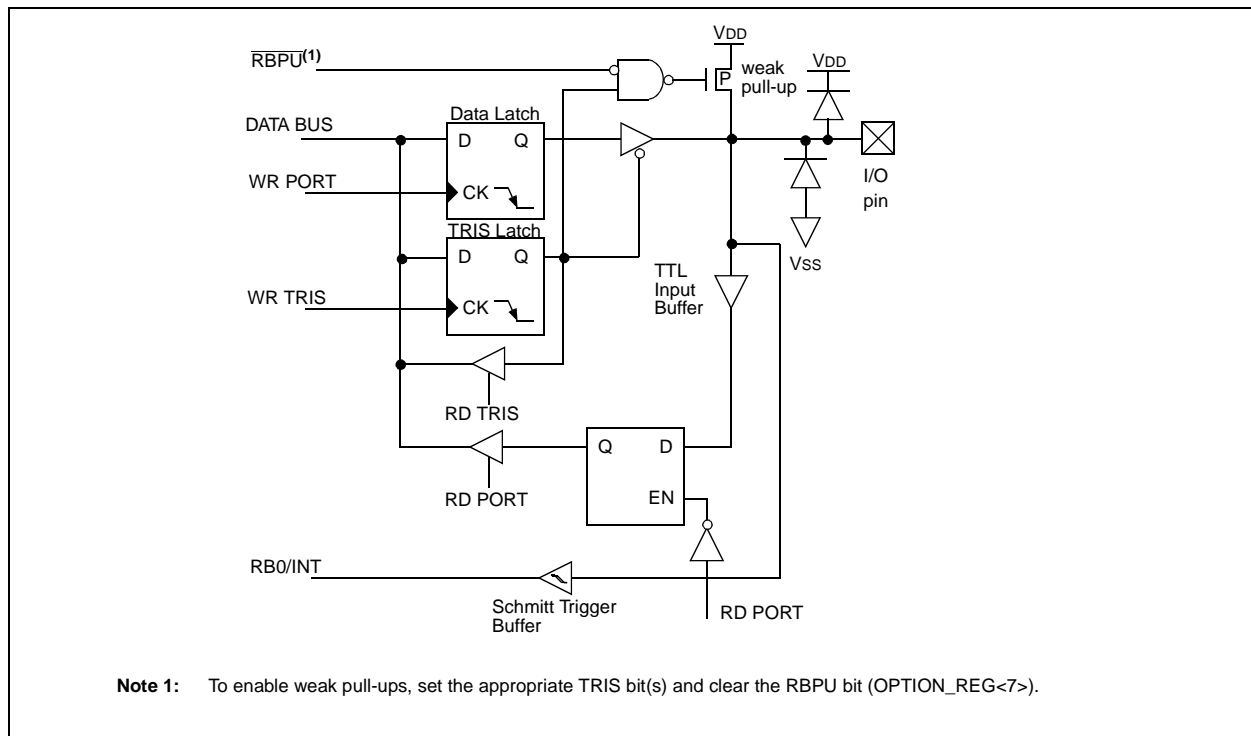
PORTB pins RB3:RB1 are multiplexed with several peripheral functions (Table 3-3). PORTB pins RB3:RB0 have Schmitt Trigger input buffers.

When enabling peripheral functions, care should be taken in defining TRIS bits for each PORTB pin. Some peripherals override the TRIS bit to make a pin an output, while other peripherals override the TRIS bit to make a pin an input. Since the TRIS bit override is in effect while the peripheral is enabled, read-modify-write instructions (BSF, BCF, XORWF) with TRISB as destination should be avoided. The user should refer to the corresponding peripheral section for the correct TRIS bit settings.

Four of PORTB's pins, RB7:RB4, have an interrupt on change feature. Only pins configured as inputs can cause this interrupt to occur (i.e. any RB7:RB4 pin configured as an output is excluded from the interrupt on change comparison). The input pins, RB7:RB4, are compared with the old value latched on the last read of PORTB. The "mismatch" outputs of RB7:RB4 are OR'ed together to generate the RB Port Change Interrupt with flag bit RBIF (INTCON<0>).

This interrupt can wake the device from SLEEP. The user, in the interrupt service routine, can clear the interrupt in the following manner:

a)   Any read or write of PORTB will end the mismatch condition.

b)   Clear flag bit RBIF.

A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition and allow flag bit RBIF to be cleared.

The interrupt on change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt on change feature. Polling of PORTB is not recommended while using the interrupt on change feature.

**FIGURE 3-4:     BLOCK DIAGRAM OF RB1/T1OSO/T1CKI PIN**



**Note 1:**   To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the RBPU bit (OPTION_REG<7>).
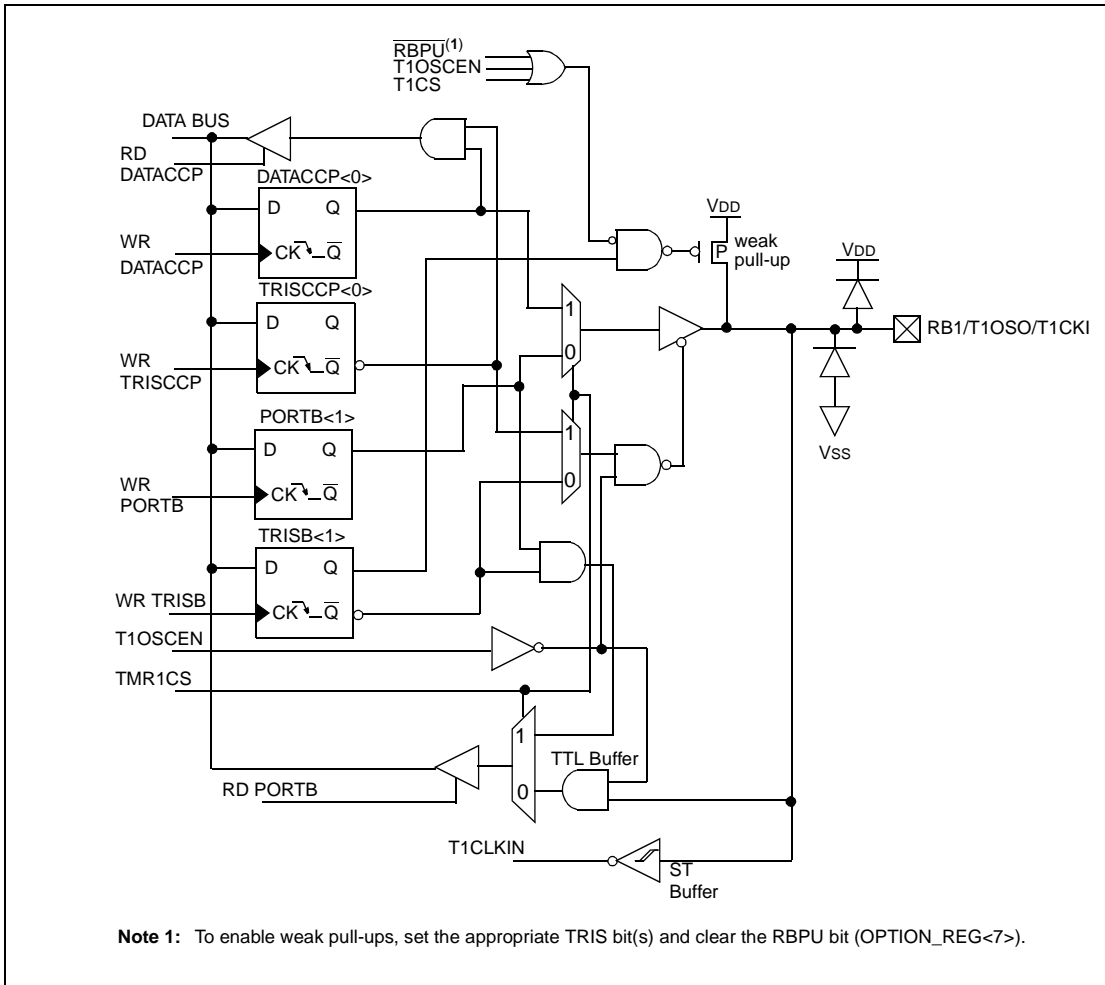
**FIGURE 3-5:    BLOCK DIAGRAM OF RB2/T1OSI PIN**



Note 1:   To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the RBPU bit (OPTION_REG<7>).
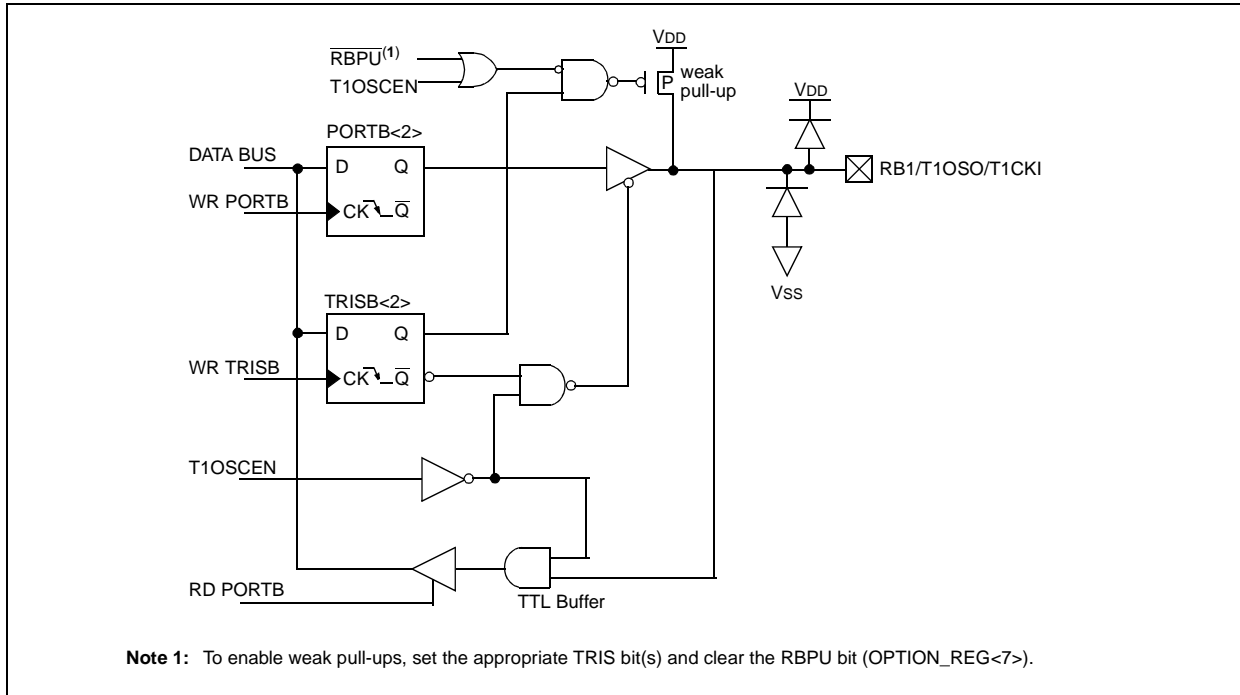
**FIGURE 3-6:    BLOCK DIAGRAM OF RB3/CCP1 PIN**



Note 1:   To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the RBPU bit (OPTION_REG<7>).
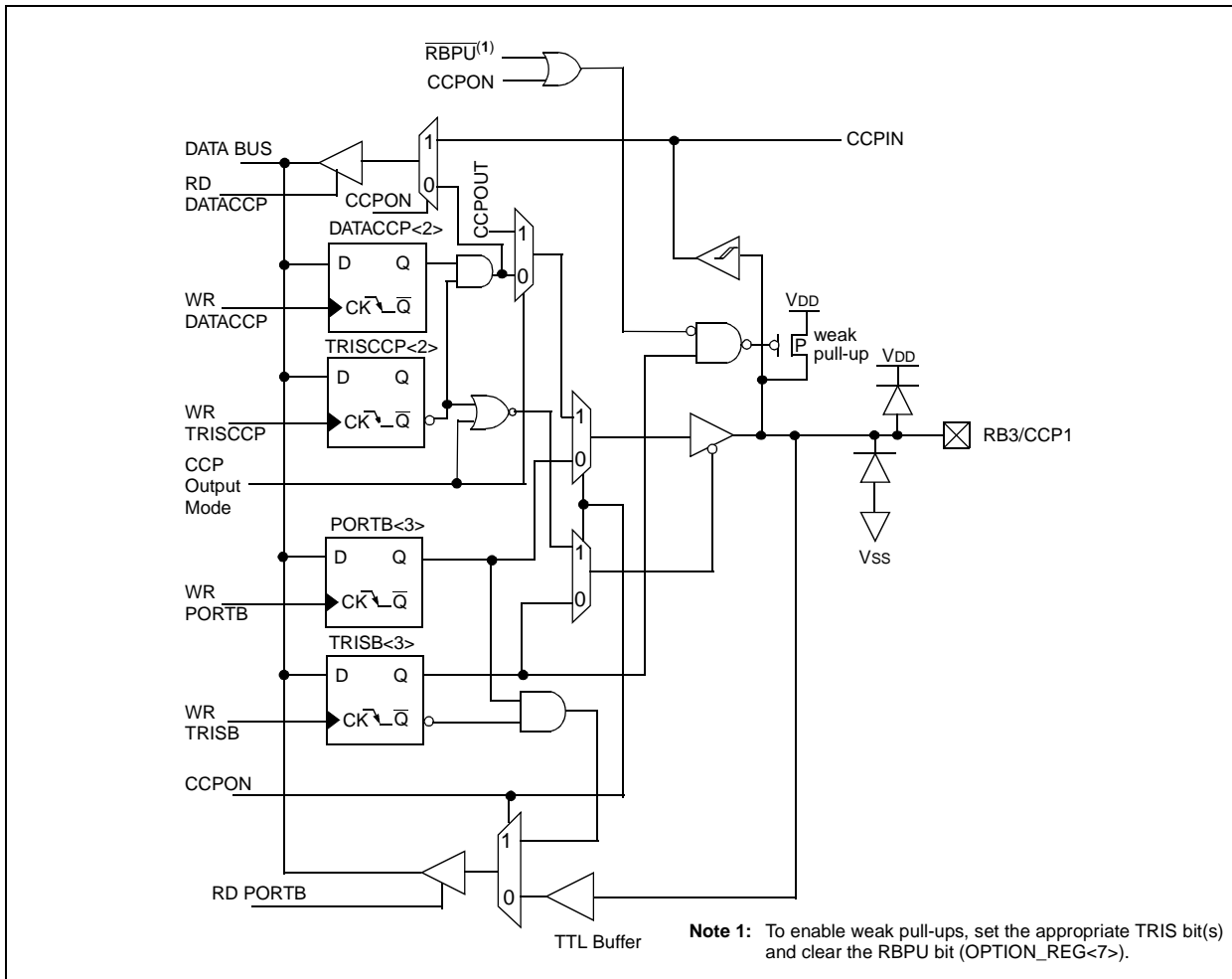
**FIGURE 3-7: BLOCK DIAGRAM OF RB7:RB4 PINS**



**Note 1:** To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the RBPU bit (OPTION_REG<7>).
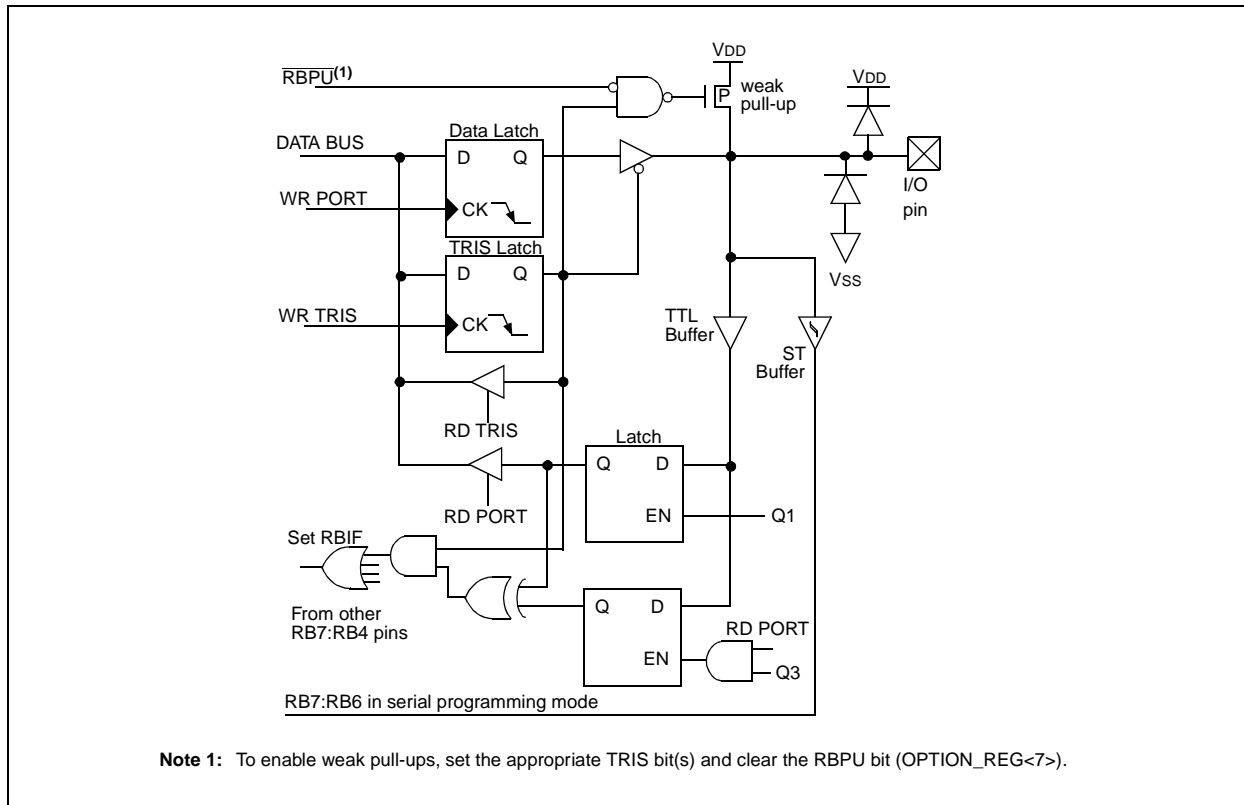
**TABLE 3-3      PORTB FUNCTIONS**

| Name | Bit# | Buffer | Function |
|------|------|--------|----------|
| RB0/INT | bit0 | TTL/ST[1] | Input/output pin or external interrupt input. Internal software programmable weak pull-up. |
| RB1/T1OS0/ T1CKI | bit1 | TTL/ST[1] | Input/output pin or Timer 1 oscillator output, or Timer 1 clock input.  Internal software programmable weak pull-up.  See Timer1 section for detailed operation. |
| RB2/T1OSI | bit2 | TTL/ST[1] | Input/output pin or Timer 1 oscillator input.  Internal software programmable weak pull-up.  See Timer1 section for detailed operation. |
| RB3/CCP1 | bit3 | TTL/ST[1] | Input/output pin or Capture 1 input, or Compare 1 output, or PWM1 output. Internal software programmable weak pull-up.  See CCP1 section for detailed operation. |
| RB4 | bit4 | TTL | Input/output pin (with interrupt on change). Internal software programmable weak pull-up. |
| RB5 | bit5 | TTL | Input/output pin (with interrupt on change). Internal software programmable weak pull-up. |
| RB6 | bit6 | TTL/ST[2] | Input/output pin (with interrupt on change). Internal software programmable weak pull-up. Serial programming clock. |
| RB7 | bit7 | TTL/ST[2] | Input/output pin (with interrupt on change). Internal software programmable weak pull-up. Serial programming data. |

Legend:  TTL = TTL input, ST = Schmitt Trigger input
**Note 1:** This buffer is a Schmitt Trigger input when configured as the external interrupt or peripheral input.
    **2:** This buffer is a Schmitt Trigger input when used in serial programming mode.

**TABLE 3-4    SUMMARY OF REGISTERS ASSOCIATED WITH PORTB**

| Address | 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 |
|---------|------|-------|-------|-------|-------|-------|-------|-------|-------|--------------------|---------------------------|
| 06h | PORTB | RB7 | RB6 | RB5 | RB4 | RB3 | RB2 | RB1 | RB0 | xxxx xxxx | uuuu uuuu |
| 86h | TRISB | PORTB Data Direction Register | | | | | | | | 1111 1111 | 1111 1111 |
| 81h | OPTION_REG | $\overline{\text{RBPU}}$ | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |

Legend:   x = unknown, u = unchanged. Shaded cells are not used by PORTB.

**NOTES:**

# 4.0 TIMER0 MODULE

The Timer0 module timer/counter has the following features:

• 8-bit timer/counter
• Readable and writable
• Internal or external clock select
• Edge select for external clock
• 8-bit software programmable prescaler
• Interrupt on overflow from FFh to 00h

Figure 4-1 is a simplified block diagram of the Timer0 module.

Additional information on timer modules is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

## 4.1 Timer0 Operation

Timer0 can operate as a timer or as a counter.

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

Counter mode is selected by setting bit T0CS (OPTION_REG<5>). In counter mode, Timer0 will increment on every rising or falling edge of pin RA4/T0CKI. The incrementing edge is determined by the Timer0 Source Edge Select bit T0SE (OPTION_REG<4>). Clearing bit T0SE selects the rising edge. Restrictions on the external clock input are discussed below.

When an external clock input is used for Timer0, it must meet certain requirements. The requirements ensure the external clock can be synchronized with the internal phase clock (TOSC). Also, there is a delay in the actual incrementing of Timer0 after synchronization.

Additional information on external clock requirements is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

## 4.2 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module or as a postscaler for the Watchdog Timer, respectively (Figure 4-2). For simplicity, this counter is being referred to as "prescaler" throughout this data sheet. Note that there is only one prescaler available, which is mutually exclusively shared between the Timer0 module and the Watchdog Timer. Thus, a prescaler assignment for the Timer0 module means that there is no prescaler for the Watchdog Timer and vice-versa.

The prescaler is not readable or writable.

The PSA and PS2:PS0 bits (OPTION_REG<3:0>) determine the prescaler assignment and prescale ratio.

Clearing bit PSA will assign the prescaler to the Timer0 module. When the prescaler is assigned to the Timer0 module, prescale values of 1:2, 1:4, ..., 1:256 are selectable.

Setting bit PSA will assign the prescaler to the Watchdog Timer (WDT). When the prescaler is assigned to the WDT, prescale values of 1:1, 1:2, ..., 1:128 are selectable.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g. CLRF 1, MOVWF 1, BSF 1,x....etc.) will clear the prescaler. When assigned to WDT, a CLRWDT instruction will clear the prescaler along with the WDT.

> **Note:** Writing to TMR0 when the prescaler is assigned to Timer0 will clear the prescaler count, but will not change the prescaler assignment.

## FIGURE 4-1: TIMER0 BLOCK DIAGRAM



Note 1: T0CS, T0SE, PSA, PS2:PS0 (OPTION_REG<5:0>).
2: The prescaler is shared with Watchdog Timer (refer to Figure 4-2 for detailed block diagram).
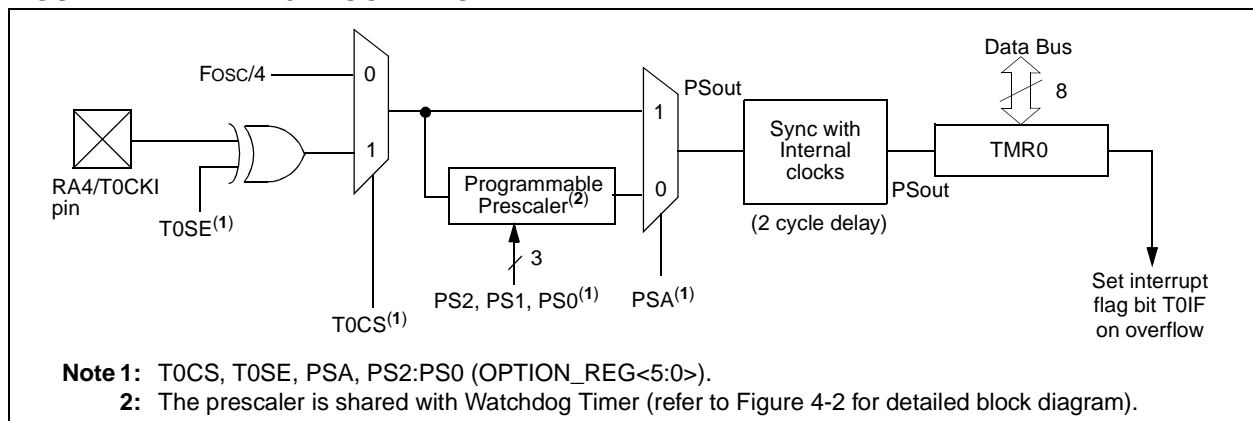
### 4.2.1 SWITCHING PRESCALER ASSIGNMENT

The prescaler assignment is fully under software control, i.e., it can be changed "on the fly" during program execution.

| **Note:** | To avoid an unintended device RESET, a specific instruction sequence (shown in the PICmicro™ Mid-Range Reference Manual, DS33023) must be executed when changing the prescaler assignment from Timer0 to the WDT. This sequence must be followed even if the WDT is disabled. |
|---|---|

### 4.3 Timer0 Interrupt

The TMR0 interrupt is generated when the TMR0 register overflows from FFh to 00h. This overflow sets bit T0IF (INTCON<2>). The interrupt can be masked by clearing bit T0IE (INTCON<5>). Bit T0IF must be cleared in software by the Timer0 module interrupt service routine before re-enabling this interrupt. The TMR0 interrupt cannot awaken the processor from SLEEP since the timer is shut off during SLEEP.

**FIGURE 4-2: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER**



**Note:** T0CS, T0SE, PSA, PS2:PS0 are (OPTION_REG<5:0>).
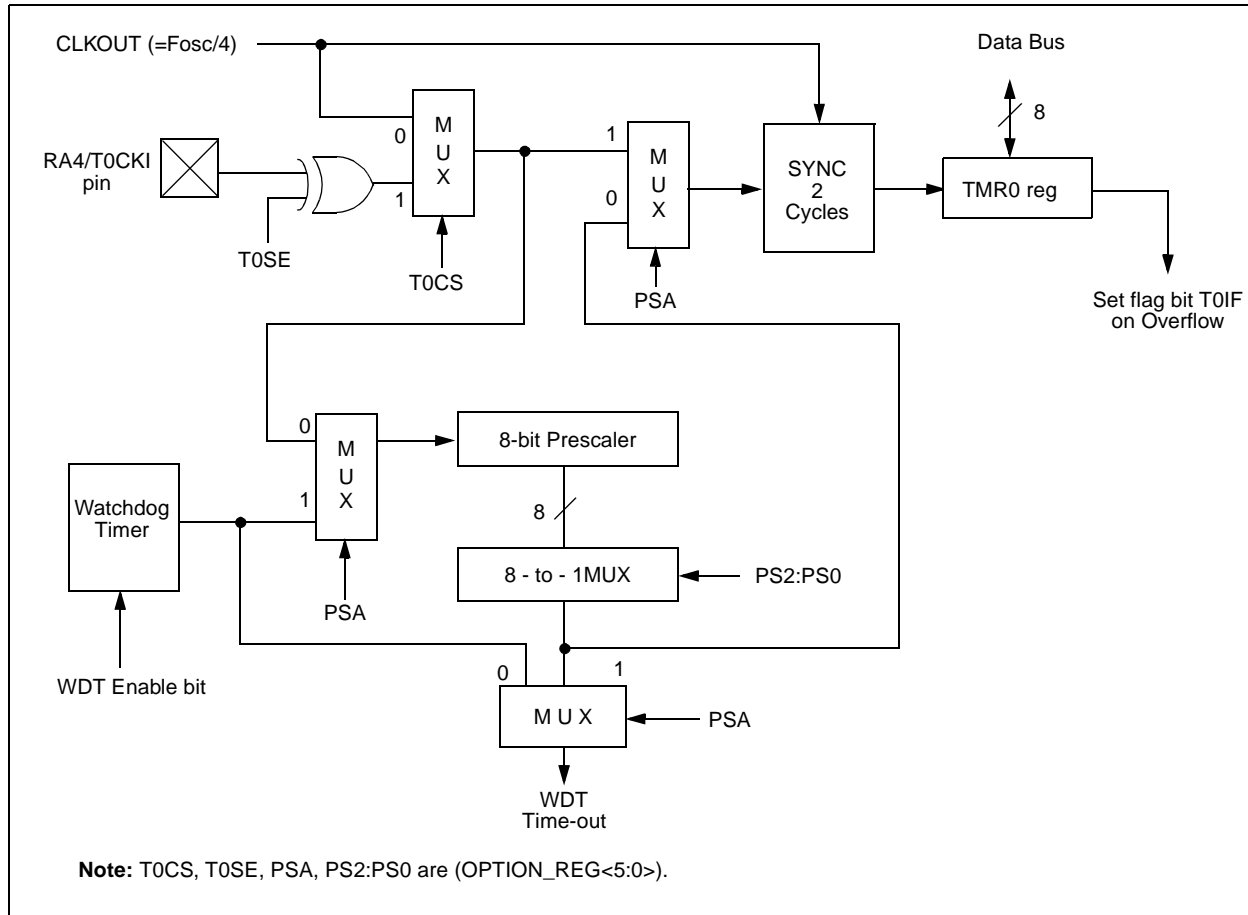
**TABLE 4-1 REGISTERS ASSOCIATED WITH TIMER0**

| Address | 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 01h | TMR0 | Timer0 module's register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 0Bh,8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 81h | OPTION_REG | RBPU | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |
| 85h | TRISA | — | — | —(1) | Bit 4 | PORTA Data Direction Register | | | | --11 1111 | --11 1111 |

Legend: x = unknown, u = unchanged, – = unimplemented locations read as '0'. Shaded cells are not used by Timer0.
**Note 1:** Reserved bit; Do Not Use.

# 5.0 TIMER1 MODULE

The Timer1 module timer/counter has the following features:

- 16-bit timer/counter
  (Two 8-bit registers; TMR1H and TMR1L)
- Readable and writable (Both registers)
- Internal or external clock select
- Interrupt on overflow from FFFFh to 0000h
- Reset from CCP module trigger

Timer1 has a control register, shown in Figure 5-1. Timer1 can be enabled/disabled by setting/clearing control bit TMR1ON (T1CON<0>).

Figure 5-2 is a simplified block diagram of the Timer1 module.

Additional information on timer modules is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

## 5.1 Timer1 Operation

Timer1 can operate in one of these modes:

- As a timer
- As a synchronous counter
- As an asynchronous counter

The operating mode is determined by the clock select bit, TMR1CS (T1CON<1>).

In timer mode, Timer1 increments every instruction cycle. In counter mode, it increments on every rising edge of the external clock input.

When the Timer1 oscillator is enabled (T1OSCEN is set), the RB2/T1OSI and RB1/T1OSO/T1CKI pins become inputs. That is, the TRISB<2:1> value is ignored.

Timer1 also has an internal "reset input". This reset can be generated by the CCP module (Section 7.0).

**FIGURE 5-1: T1CON: TIMER1 CONTROL REGISTER (ADDRESS 10h)**

| U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | T1CKPS1 | T1CKPS0 | T1OSCEN | T1SYNC | TMR1CS | TMR1ON |

bit7 ... bit0

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

bit 7-6: **Unimplemented:** Read as '0'

bit 5-4: **T1CKPS1:T1CKPS0**: 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**: Timer1 Oscillator Enable Control bit
1 = Oscillator is enabled
0 = Oscillator is shut off
Note: The oscillator inverter and feedback resistor are turned off to eliminate power drain

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 when TMR1CS = 0.

bit 1: **TMR1CS**: Timer1 Clock Source Select bit
1 = External clock from pin RB1/T1OSO/T1CKI (on the rising edge)
0 = Internal clock (FOSC/4)

bit 0: **TMR1ON**: Timer1 On bit
1 = Enables Timer1
0 = Stops Timer1

# PIC16C712/716

**FIGURE 5-2:** **TIMER1 BLOCK DIAGRAM**



**Note 1:** When the T1OSCEN bit is cleared, the inverter and feedback resistor are turned off. This eliminates power drain.
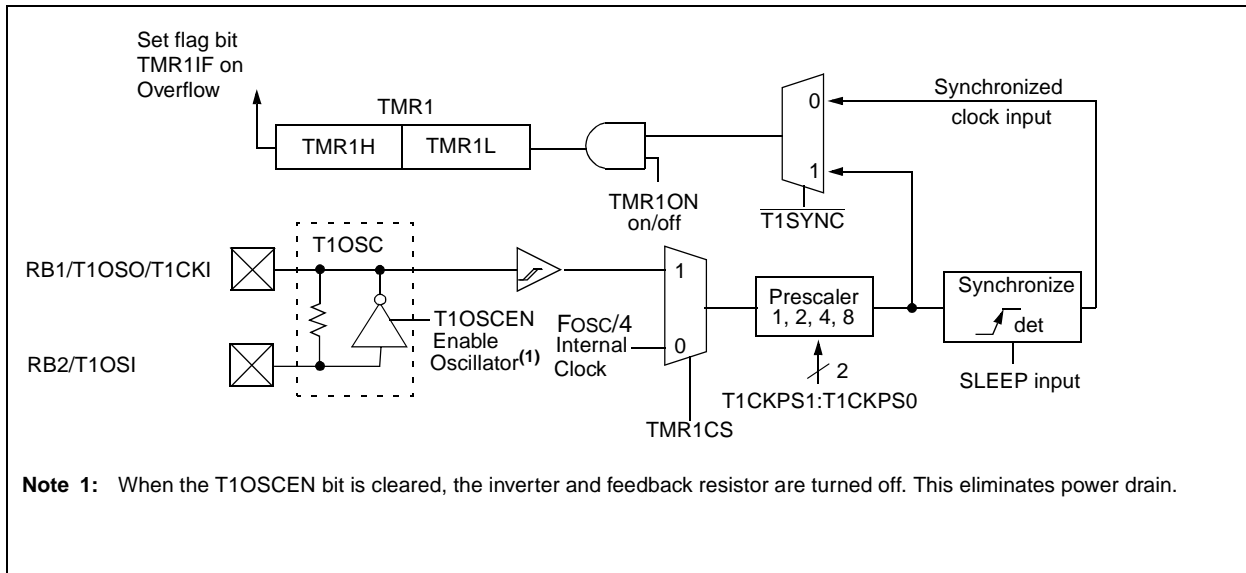
## 5.2 Timer1 Module and PORTB Operation

When Timer1 is configured as timer running from the main oscillator, PORTB<2:1> operate as normal I/O lines. When Timer1 is configured to function as a counter however, the clock source selection may affect the operation of PORTB<2:1>. Multiplexing details of the Timer1 clock selection on PORTB are shown in Figure 3-4 and Figure 3-5.

The clock source for Timer1 in the counter mode can be from one of the following:

1. External circuit connected to the RB1/T1OSO/T1CKI pin
2. Firmware controlled DATACCP<0> bit, DT1CKI
3. Timer1 oscillator

Table 5-1 shows the details of Timer1 mode selections, control bit settings, TMR1 and PORTB operations.

**TABLE 5-1      TMR1 MODULE AND PORTB OPERATION**

| TMR1 Module Mode | Clock Source | Control Bits | TMR1 Module Operation | PORTB<2:1> Operation |
|---|---|---|---|---|
| Off | N/A | T1CON = --xx 0x00 | Off | PORTB<2:1> function as normal I/O |
| Timer | Fosc/4 | T1CON = --xx 0x01 | TMR1 module uses the main oscillator as clock source. TMR1ON can turn on or turn off Timer1. | PORTB<2:1> function as normal I/O |
| Counter | External circuit | T1CON = --xx 0x11 TR1SCCP = ---- -x-1 | TMR1 module uses the external signal on the RB1/T1OSO/T1CKI pin as a clock source. TMR1ON can turn on or turn off Timer1. DT1CK can read the signal on the RB1/T1OSO/T1CKI pin. | PORTB<2> functions as normal I/O. PORTB<1> always reads 0 when configured as input . If PORTB<1> is configured as out-put, reading PORTB<1> will read the data latch. Writing to PORTB<1> will always store the result in the data latch, but not to the RB1/T1OSO/T1CKI pin. If the TMR1CS bit is cleared (TMR1 reverts to the timer mode), then pin PORTB<1> will be driven with the value in the data latch. |
| | Firmware | T1CON = --xx 0x11 TR1SCCP = ---- -x-0 | DATACCP<0> bit drives RB1/T1OSO/T1CKI and pro-duces the TMR1 clock source. TMR1ON can turn on or turn off Timer1. The DATACCP<0> bit, DT1CK, can read and write to the RB1/T1OSO/T1CKI pin. | |
| | Timer1 oscillator | T1CON = --xx 1x11 | RB1/T1OSO/T1CKI and RB2/T1OSI are configured as a 2 pin crystal oscillator. RB1/T1OSI/T1CKI is the clock input for TMR1. TMR1ON can turn on or turn off Timer1. DATACCP<1> bit, DT1CK, always reads 0 as input and can not write to the RB1/T1OSO/T1CK1 pin. | PORTB<2:1> always read 0 when configured as inputs. If PORTB<2:1> are configured as outputs, reading PORTB<2:1> will read the data latches. Writing to PORTB<2:1> will always store the result in the data latches, but not to the RB2/T1OSI and RB1/T1OSO/T1CKI pins. If the TMR1CS and T1OSCEN bits are cleared (TMR1 reverts to the timer mode and TMR1 oscillator is disabled), then pin PORTB<2:1> will be driven with the value in the data latches. |

## 5.3    Timer1 Oscillator

A crystal oscillator circuit is built in between pins T1OSI (input) and T1OSO (amplifier output). It is enabled by setting control bit T1OSCEN (T1CON<3>). The oscillator is a low power oscillator rated up to 200 kHz. It will continue to run during SLEEP. It is primarily intended for a 32 kHz crystal. Table 5-2 shows the capacitor selection for the Timer1 oscillator.

The Timer1 oscillator is identical to the LP oscillator. The user must provide a software time delay to ensure proper oscillator start-up.

### TABLE 5-2    CAPACITOR SELECTION FOR THE TIMER1 OSCILLATOR

| Osc Type | Freq | C1 | C2 |
|----------|------|-----|-----|
| LP | 32 kHz | 33 pF | 33 pF |
|  | 100 kHz | 15 pF | 15 pF |
|  | 200 kHz | 15 pF | 15 pF |

**These values are for design guidance only.**

Note 1:   Higher capacitance increases the stability of oscillator but also increases the start-up time.

2:   Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components.

## 5.4    Timer1 Interrupt

The TMR1 Register pair (TMR1H:TMR1L) increments from 0000h to FFFFh and rolls over to 0000h. The TMR1 interrupt, if enabled, is generated on overflow which is latched in interrupt flag bit TMR1IF (PIR1<0>). This interrupt can be enabled/disabled by setting/clearing TMR1 interrupt enable bit TMR1IE (PIE1<0>).

## 5.5    Resetting Timer1 using a CCP Trigger Output

If the CCP module is configured in compare mode to generate a "special event trigger" (CCP1M3:CCP1M0 = 1011), this signal will reset Timer1 and start an A/D conversion (if the A/D module is enabled).

| Note: | The special event triggers from the CCP1 module will not set interrupt flag bit TMR1IF (PIR1<0>). |
|-------|---|

Timer1 must be configured for either timer or synchronized counter mode to take advantage of this feature. If Timer1 is running in asynchronous counter mode, this reset operation may not work.

In the event that a write to Timer1 coincides with a special event trigger from CCP1, the write will take precedence.

In this mode of operation, the CCPR1H:CCPR1L registers pair effectively becomes the period register for Timer1.

### TABLE 5-3    REGISTERS ASSOCIATED WITH TIMER1 AS A TIMER/COUNTER

| Address | 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 |
|---------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------------------|---------------------------|
| 0Bh,8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | — | CCP1IF | TMR2IF | TMR1IF | -0-- -000 | -0-- -000 |
| 8Ch | PIE1 | — | ADIE | — | — | — | CCP1IE | TMR2IE | TMR1IE | -0-- -000 | -0-- -000 |
| 0Eh | TMR1L | Holding register for the Least Significant Byte of the 16-bit TMR1 register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 0Fh | TMR1H | Holding register for the Most Significant Byte of the 16-bit TMR1 register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 10h | T1CON | — | — | T1CKPS1 | T1CKPS0 | T1OSCEN | T1SYNC | TMR1CS | TMR1ON | --00 0000 | --uu uuuu |
| 07h | DATACCP | — | — | — | — | — | DCCP | — | DT1CK | ---- -x-x | ---- -u-u |
| 87h | TRISCCP | — | — | — | — | — | TCCP | — | TT1CK | ---- -1-1 | ---- -1-1 |

Legend:   x = unknown, u = unchanged, – = unimplemented read as '0'. Shaded cells are not used by the Timer1 module.

**NOTES:**

# PIC16C712/716

## 6.0 TIMER2 MODULE

The Timer2 module timer has the following features:

- 8-bit timer (TMR2 register)
- 8-bit period register (PR2)
- Readable and writable (Both registers)
- Software programmable prescaler (1:1, 1:4, 1:16)
- Software programmable postscaler (1:1 to 1:16)
- Interrupt on TMR2 match of PR2

Timer2 has a control register, shown in Figure 6-1. Timer2 can be shut off by clearing control bit TMR2ON (T2CON<2>) to minimize power consumption.

Figure 6-2 is a simplified block diagram of the Timer2 module.

Additional information on timer modules is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

### FIGURE 6-1: T2CON: TIMER2 CONTROL REGISTER (ADDRESS 12h)

| U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|------|---------|---------|---------|---------|--------|---------|---------|
| — | TOUTPS3 | TOUTPS2 | TOUTPS1 | TOUTPS0 | TMR2ON | T2CKPS1 | T2CKPS0 |

bit7                                                                          bit0

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

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

bit 6-3: **TOUTPS3:TOUTPS0**: Timer2 Output Postscale Select bits
0000 = 1:1 Postscale
0001 = 1:2 Postscale
0010 = 1:3 Postscale
0011 = 1:4 Postscale
0100 = 1:5 Postscale
0101 = 1:6 Postscale
0110 = 1:7 Postscale
0111 = 1:8 Postscale
1000 = 1:9 Postscale
1001 = 1:10 Postscale
1010 = 1:11 Postscale
1011 = 1:12 Postscale
1100 = 1:13 Postscale
1101 = 1:14 Postscale
1110 = 1:15 Postscale
1111 = 1:16 Postscale

bit 2: **TMR2ON**: Timer2 On bit
1 = Timer2 is on
0 = Timer2 is off

bit 1-0: **T2CKPS1:T2CKPS0**: Timer2 Clock Prescale Select bits
00 = Prescaler is 1
01 = Prescaler is 4
1x = Prescaler is 16

### FIGURE 6-2: TIMER2 BLOCK DIAGRAM
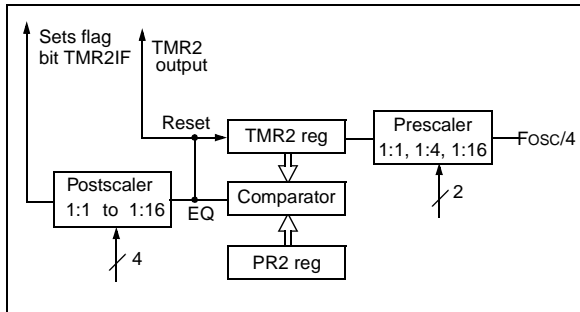
## 6.1 Timer2 Operation

Timer2 can be used as the PWM time-base for PWM mode of the CCP module.

The TMR2 register is readable and writable, and is cleared on any device reset.

The input clock (Fosc/4) has a prescale option of 1:1, 1:4 or 1:16, selected by control bits T2CKPS1:T2CKPS0 (T2CON<1:0>).

The match output of TMR2 goes through a 4-bit postscaler (which gives a 1:1 to 1:16 scaling inclusive) to generate a TMR2 interrupt (latched in flag bit TMR2IF, (PIR1<1>)).

The prescaler and postscaler counters are cleared when any of the following occurs:

- a write to the TMR2 register
- a write to the T2CON register
- any device reset (Power-on Reset, $\overline{\text{MCLR}}$ reset, Watchdog Timer reset, or Brown-out Reset)

TMR2 is not cleared when T2CON is written.

## 6.2 Timer2 Interrupt

The Timer2 module has an 8-bit period register PR2. Timer2 increments from 00h until it matches PR2 and then resets to 00h on the next increment cycle. PR2 is a readable and writable register. The PR2 register is initialized to FFh upon reset.

### TABLE 6-1 REGISTERS ASSOCIATED WITH TIMER2 AS A TIMER/COUNTER

| Address | 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 |
|---------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------------------|---------------------------|
| 0Bh,8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | — | CCP1IF | TMR2IF | TMR1IF | -00- -000 | 0000 -000 |
| 8Ch | PIE1 | — | ADIE | — | — | — | CCP1IE | TMR2IE | TMR1IE | -0-- -000 | 0000 -000 |
| 11h | TMR2 | Timer2 module's register | | | | | | | | 0000 0000 | 0000 0000 |
| 12h | T2CON | — | TOUTPS3 | TOUTPS2 | TOUTPS1 | TOUTPS0 | TMR2ON | T2CKPS1 | T2CKPS0 | -000 0000 | -000 0000 |
| 92h | PR2 | Timer2 Period Register | | | | | | | | 1111 1111 | 1111 1111 |

Legend:   x = unknown, u = unchanged, – = unimplemented read as '0'. Shaded cells are not used by the Timer2 module.

NOTES:

## 7.0 CAPTURE/COMPARE/PWM (CCP) MODULE(S)

Each CCP (Capture/Compare/PWM) module contains a 16-bit register, which can operate as a 16-bit capture register, as a 16-bit compare register or as a PWM master/slave Duty Cycle register. Table 7-1 shows the timer resources of the CCP module modes.

Capture/Compare/PWM Register 1 (CCPR1) is comprised of two 8-bit registers: CCPR1L (low byte) and CCPR1H (high byte). The CCP1CON register controls the operation of CCP1. All are readable and writable.

Additional information on the CCP module is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

### TABLE 7-1 CCP MODE - TIMER RESOURCE

| CCP Mode | Timer Resource |
|----------|----------------|
| Capture | Timer1 |
| Compare | Timer1 |
| PWM | Timer2 |

### FIGURE 7-1: CCP1CON REGISTER (ADDRESS 17h)

| U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|-----|-----|-------|-------|-------|-------|-------|-------|
| — | — | DC1B1 | DC1B0 | CCP1M3 | CCP1M2 | CCP1M1 | CCP1M0 |

bit7                                                                    bit0

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

bit 7-6: **Unimplemented:** Read as '0'

bit 5-4: **DC1B1:DC1B0**: PWM Least Significant bits
Capture Mode: Unused
Compare Mode: Unused
PWM Mode: These bits are the two LSbs of the PWM duty cycle. The eight MSbs are found in CCPR1L.

bit 3-0: **CCP1M3:CCP1M0**: CCP1 Mode Select bits
0000 = Capture/Compare/PWM off (resets CCP1 module)
0100 = Capture mode, every falling edge
0101 = Capture mode, every rising edge
0110 = Capture mode, every 4th rising edge
0111 = Capture mode, every 16th rising edge
1000 = Compare mode, set output on match (CCP1IF bit is set)
1001 = Compare mode, clear output on match (CCP1IF bit is set)
1010 = Compare mode, generate software interrupt on match (CCP1IF bit is set, CCP1 pin is unaffected)
1011 = Compare mode, trigger special event (CCP1IF bit is set; CCP1 resets TMR1 and starts an A/D conversion (if A/D module is enabled))
11xx = PWM mode

### FIGURE 7-2: TRISCCP Register (ADDRESS 87h)

| R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| — | — | — | — | — | TCCP | — | TT1CK |

bit7                                                                    bit0

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

bit 7-3: **Reserved bits; Do Not Use**

bit 2: **TCCP - Tri state control bit for CCP**
0 = Output pin driven
1 = Output pin tristated

bit 1: **Reserved bit; Do Not Use**

bit 0: **TT1CK - Tri state control bit for T1CKI pin**
0 = T1CKI pin is an output
1 = T1CKI pin is an input

## 7.1 Capture Mode

In Capture mode, CCPR1H:CCPR1L captures the 16-bit value of the TMR1 register when an event occurs on pin RB3/CCP1. An event is defined as:

- every falling edge
- every rising edge
- every 4th rising edge
- every 16th rising edge

An event is selected by control bits CCP1M3:CCP1M0 (CCP1CON<3:0>). When a capture is made, the interrupt request flag bit CCP1IF (PIR1<2>) is set. It must be cleared in software. If another capture occurs before the value in register CCPR1 is read, the old captured value will be lost.

**FIGURE 7-3: CAPTURE MODE OPERATION BLOCK DIAGRAM**
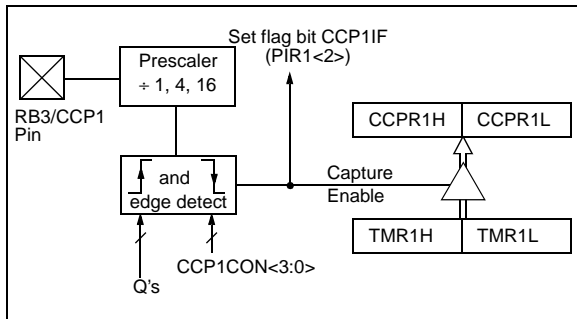


### 7.1.1 CCP PIN CONFIGURATION

In Capture mode, the CCP output must be disabled by setting the TRISCCP<2> bit.

> **Note:** If the RB3/CCP1 is configured as an output by clearing the TRISCCP<2> bit, a write to the DCCP bit can cause a capture condition.

### 7.1.2 TIMER1 MODE SELECTION

Timer1 must be running in timer mode or synchronized counter mode for the CCP module to use the capture feature. In asynchronous counter mode, the capture operation may not work.

### 7.1.3 SOFTWARE INTERRUPT

When the Capture mode is changed, a false capture interrupt may be generated. The user should keep bit CCP1IE (PIE1<2>) clear to avoid false interrupts and should clear the flag bit CCP1IF following any such change in operating mode.

### 7.1.4 CCP PRESCALER

There are four prescaler settings, specified by bits CCP1M3:CCP1M0. Whenever the CCP module is turned off, or the CCP module is not in capture mode, the prescaler counter is cleared. This means that any reset will clear the prescaler counter.

Switching from one capture prescaler to another may generate an interrupt. Also, the prescaler counter will not be cleared, therefore the first capture may be from a non-zero prescaler. Example 7-1 shows the recommended method for switching between capture prescalers. This example also clears the prescaler counter and will not generate the "false" interrupt.

**EXAMPLE 7-1: CHANGING BETWEEN CAPTURE PRESCALERS**

```
CLRF    CCP1CON       ;Turn CCP module off
MOVLW   NEW_CAPT_PS   ;Load the W reg with
                      ; the new prescaler
                      ; mode value and CCP ON
MOVWF   CCP1CON       ;Load CCP1CON with this
                      ; value
```

## 7.2    Compare Mode

In Compare mode, the 16-bit CCPR1 register value is constantly compared against the TMR1 register pair value. When a match occurs, the RB3/CCP1 pin is either:

• driven High
• driven Low
• remains Unchanged

The action on the pin is based on the value of control bits CCP1M3:CCP1M0 (CCP1CON<3:0>). At the same time, interrupt flag bit CCP1IF is set.

### FIGURE 7-4:    COMPARE MODE OPERATION BLOCK DIAGRAM
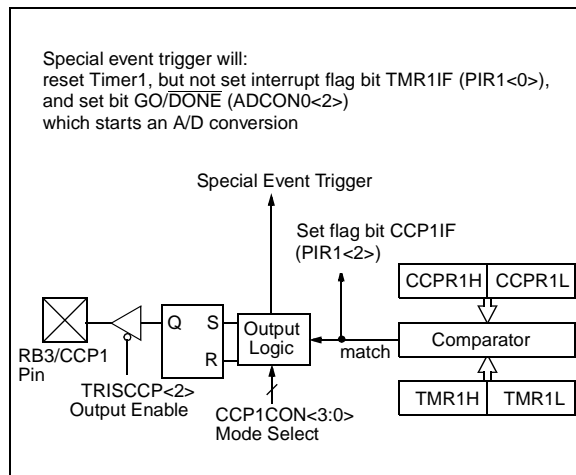


### 7.2.1    CCP PIN CONFIGURATION

The user must configure the RB3/CCP1 pin as the CCP output by clearing the TRISCCP<2> bit.

> **Note:**    Clearing the CCP1CON register will force the RB3/CCP1 compare output latch to the default low level. This is neither the PORTB I/O data latch nor the DATACCP latch.

### 7.2.2    TIMER1 MODE SELECTION

Timer1 must be running in Timer mode or Synchronized Counter mode if the CCP module is using the compare feature. In Asynchronous Counter mode, the compare operation may not work.

### 7.2.3    SOFTWARE INTERRUPT MODE

When generate software interrupt is chosen the CCP1 pin is not affected. Only a CCP interrupt is generated (if enabled).

### 7.2.4    SPECIAL EVENT TRIGGER

In this mode, an internal hardware trigger is generated which may be used to initiate an action.

The special event trigger output of CCP1 resets the TMR1 register pair. This allows the CCPR1 register to effectively be a 16-bit programmable period register for Timer1.

The special event trigger output of CCP1 also starts an A/D conversion (if the A/D module is enabled).

> **Note:**    The special event trigger from the CCP1 module will not set interrupt flag bit TMR1IF (PIR1<0>).

### TABLE 7-2    REGISTERS ASSOCIATED WITH CAPTURE, COMPARE, AND TIMER1

| Address | 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 |
|---------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------------------|---------------------------|
| 07h | DATACCP | — | — | — | — | — | DCCP | — | TT1CK | xxxx xxxx | xxxx xuxu |
| 0Bh,8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | — | CCP1IF | TMR2IF | TMR1IF | -0-- -000 | -0-- -000 |
| 0Eh | TMR1L | Holding register for the Least Significant Byte of the 16-bit TMR1 register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 0Fh | TMR1H | Holding register for the Most Significant Byte of the 16-bit TMR1register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 10h | T1CON | — | — | T1CKPS1 | T1CKPS0 | T1OSCEN | T1SYNC | TMR1CS | TMR1ON | --00 0000 | --uu uuuu |
| 15h | CCPR1L | Capture/Compare/PWM register1 (LSB) | | | | | | | | xxxx xxxx | uuuu uuuu |
| 16h | CCPR1H | Capture/Compare/PWM register1 (MSB) | | | | | | | | xxxx xxxx | uuuu uuuu |
| 17h | CCP1CON | — | — | DC1B1 | DC1B0 | CCP1M3 | CCP1M2 | CCP1M1 | CCP1M0 | --00 0000 | --00 0000 |
| 87h | TRISCCP | — | — | — | — | — | TCCP | — | TT1CK | xxxx x1x1 | xxxx x1x1 |
| 8Ch | PIE1 | — | ADIE | — | — | — | CCP1IE | TMR2IE | TMR1IE | -0-- -000 | -0-- -000 |

Legend:    x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by Capture and Timer1.

## 7.3 PWM Mode

In Pulse Width Modulation (PWM) mode, the CCP1 pin produces up to a 10-bit resolution PWM output. Since the CCP1 pin is multiplexed with the PORTB data latch, the TRISCCP<2> bit must be cleared to make the CCP1 pin an output.

> **Note:** Clearing the CCP1CON register will force the CCP1 PWM output latch to the default low level. This is neither the PORTB I/O data latch nor the DATACCP latch.

Figure 7-5 shows a simplified block diagram of the CCP module in PWM mode.

For a step by step procedure on how to set up the CCP module for PWM operation, see Section 7.3.3.

**FIGURE 7-5:** SIMPLIFIED PWM BLOCK DIAGRAM



**Note 1:** 8-bit timer is concatenated with 2-bit internal Q clock or 2 bits of the prescaler to create 10-bit time-base.
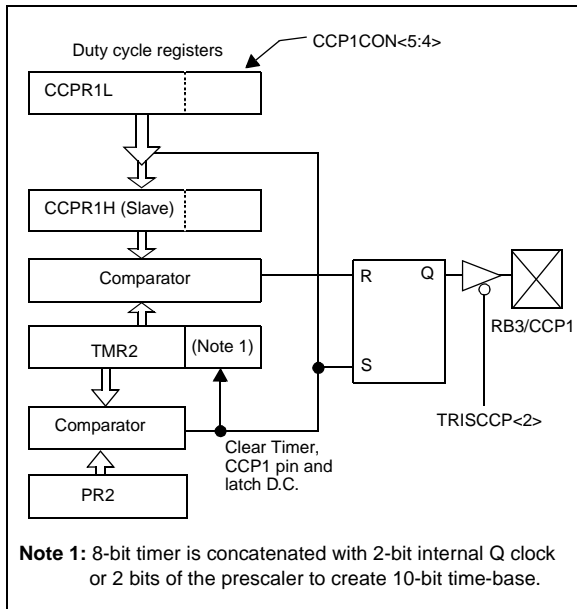
A PWM output (Figure 7-6) has a time base (period) and a time that the output stays high (duty cycle). The frequency of the PWM is the inverse of the period (1/period).

**FIGURE 7-6:** PWM OUTPUT
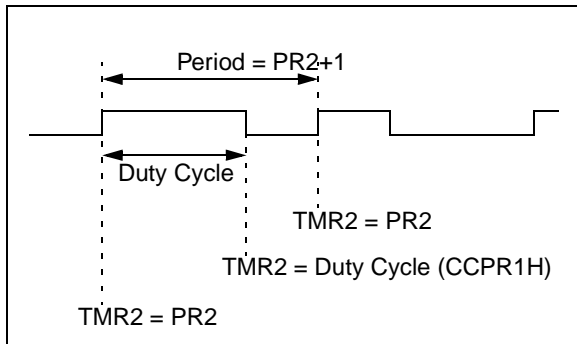


### 7.3.1 PWM PERIOD

The PWM period is specified by writing to the PR2 register. The PWM period can be calculated using the following formula:

$$\text{PWM period} = [(PR2) + 1] \cdot 4 \cdot \text{T}_{OSC} \cdot (\text{TMR2 prescale value})$$

PWM frequency is defined as 1 / [PWM period].

When TMR2 is equal to PR2, the following three events occur on the next increment cycle:

- TMR2 is cleared
- The CCP1 pin is set (exception: if PWM duty cycle = 0%, the CCP1 pin will not be set)
- The PWM duty cycle is latched from CCPR1L into CCPR1H

> **Note:** The Timer2 postscaler (see Section 6.0) is not used in the determination of the PWM frequency. The postscaler could be used to have a servo update rate at a different frequency than the PWM output.

### 7.3.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing to the CCPR1L register and to the CCP1CON<5:4> bits. Up to 10-bit resolution is available. The CCPR1L contains the eight MSbs and the CCP1CON<5:4> contains the two LSbs. This 10-bit value is represented by CCPR1L:CCP1CON<5:4>. The following equation is used to calculate the PWM duty cycle in time:

$$\text{PWM duty cycle} = (\text{CCPR1L:CCP1CON<5:4>}) \cdot \text{Tosc} \cdot (\text{TMR2 prescale value})$$

CCPR1L and CCP1CON<5:4> can be written to at any time, but the duty cycle value is not latched into CCPR1H until after a match between PR2 and TMR2 occurs (i.e., the period is complete). In PWM mode, CCPR1H is a read-only register.

The CCPR1H register and a 2-bit internal latch are used to double buffer the PWM duty cycle. This double buffering is essential for glitchless PWM operation.

When the CCPR1H and 2-bit latch match TMR2 concatenated with an internal 2-bit Q clock or 2 bits of the TMR2 prescaler, the CCP1 pin is cleared.

Maximum PWM resolution (bits) for a given PWM frequency:

$$= \frac{\log\left(\dfrac{\text{F}_{OSC}}{\text{F}_{PWM}}\right)}{\log(2)} \text{ bits}$$

> **Note:** If the PWM duty cycle value is longer than the PWM period the CCP1 pin will not be cleared.

For an example PWM period and duty cycle calculation, see the PICmicro™ Mid-Range Reference Manual, (DS33023).

### 7.3.3    SET-UP FOR PWM OPERATION

The following steps should be taken when configuring the CCP module for PWM operation:

1. Set the PWM period by writing to the PR2 register.
2. Set the PWM duty cycle by writing to the CCPR1L register and CCP1CON<5:4> bits.
3. Make the CCP1 pin an output by clearing the TRISCCP<2> bit.
4. Set the TMR2 prescale value and enable Timer2 by writing to T2CON.
5. Configure the CCP1 module for PWM operation.

**TABLE 7-3    EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 20 MHz**

| PWM Frequency | 1.22 kHz | 4.88 kHz | 19.53 kHz | 78.12 kHz | 156.3 kHz | 208.3 kHz |
|---|---|---|---|---|---|---|
| Timer Prescaler (1, 4, 16) | 16 | 4 | 1 | 1 | 1 | 1 |
| PR2 Value | 0xFF | 0xFF | 0xFF | 0x3F | 0x1F | 0x17 |
| Maximum Resolution (bits) | 10 | 10 | 10 | 8 | 7 | 5.5 |

**TABLE 7-4    REGISTERS ASSOCIATED WITH PWM AND TIMER2**

| Address | 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 07h | DATACCP | — | — | — | — | — | DCCP | — | DT1CK | xxxx xxxx | xxxx xuxu |
| 0Bh,8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | — | CCP1IF | TMR2IF | TMR1IF | -0-- -000 | -0-- -000 |
| 11h | TMR2 | Timer2 module's register | | | | | | | | 0000 0000 | 0000 0000 |
| 12h | T2CON | — | TOUTPS3 | TOUTPS2 | TOUTPS1 | TOUTPS0 | TMR2ON | T2CKPS1 | T2CKPS0 | -000 0000 | -000 0000 |
| 15h | CCPR1L | Capture/Compare/PWM register1 (LSB) | | | | | | | | xxxx xxxx | uuuu uuuu |
| 16h | CCPR1H | Capture/Compare/PWM register1 (MSB) | | | | | | | | xxxx xxxx | uuuu uuuu |
| 17h | CCP1CON | — | — | DC1B1 | DC1B0 | CCP1M3 | CCP1M2 | CCP1M1 | CCP1M0 | --00 0000 | --00 0000 |
| 87h | TRISCCP | — | — | — | — | — | TCCP | — | TT1CK | xxxx x1x1 | xxxx x1x1 |
| 8Ch | PIE1 | — | ADIE | — | — | — | CCP1IE | TMR2IE | TMR1IE | -0-- -000 | -0-- -000 |
| 92h | PR2 | Timer2 module's period register | | | | | | | | 1111 1111 | 1111 1111 |

Legend:    x = unknown, u = unchanged, – = unimplemented read as '0'. Shaded cells are not used by PWM and Timer2.

## 7.4    CCP1 Module and PORTB Operation

When the CCP module is disabled, PORTB<3> operates as a normal I/O pin.  When the CCP module is enabled, PORTB<3> operation is affected.   Multiplexing details of the CCP1 module are shown on PORTB<3>, refer to Figure 3.6.

Table 7-5 below shows the effects of the CCP module operation on PORTB<3>

**TABLE 7-5    CCP1 MODULE AND PORTB OPERATION**

| CCP1 Module Mode | Control Bits | CCP1 Module Operation | PORTB<3> Operation |
|---|---|---|---|
| Off | CCP1CON = --xx 0000 | Off | PORTB<3> functions as normal I/O. |
| Capture | CCP1CON = --xx 01xx TR1SCCP = ---- -1-x | The CCP1 module will capture an event on the RB3/CCP1 pin which is driven by an external circuit.  The DCCP bit can read the signal on the RB3/CCP1 pin. | PORTB<3> always reads 0 when configured as input.  If PORTB<3> is configured as output, reading PORTB<3> will read the data latch. Writing to PORTB<3> will always store the result in the data latch, but it does not drive the RB3/CCP1 pin. |
| | CCP1CON = --xx 01xx TR1SCCP = ---- -0-x | The CCP1 module will capture an event on the RB3/CCP1 pin which is driven by the DCCP bit.  The DCCP bit can read the signal on the RB3/CCP1 pin. | |
| Compare | CCP1CON = --xx 10xx TR1SCCP = ---- -0-x | The CCP1 module produces an output on the RB3/CCP1 pin when a compare event occurs.  The DCCP bit can read the signal on the RB3/CCP1 pin. | |
| PWM | CCP1CON = --xx 11xx TR1SCCP = ---- -0-x | The CCP1 module produces the PWM signal on the RB3/CCP1 pin.  The DCCP bit can read the signal on the RB3/CCP1 pin. | |

## 8.0 ANALOG-TO-DIGITAL CONVERTER (A/D) MODULE

The analog-to-digital (A/D) converter module has four inputs.

The A/D allows conversion of an analog input signal to a corresponding 8-bit digital number (refer to Application Note AN546 for use of A/D Converter). The output of the sample and hold is the input into the converter, which generates the result via successive approximation. The analog reference voltage is software selectable to either the device's positive supply voltage ($V_{DD}$) or the voltage level on the RA3/AN3/$V_{REF}$ pin.

The A/D converter has a unique feature of being able to operate while the device is in SLEEP mode. To operate in sleep, the A/D conversion clock must be derived from the A/D's internal RC oscillator.

Additional information on the A/D module is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

The A/D module has three registers. These registers are:

- A/D Result Register (ADRES)
- A/D Control Register 0 (ADCON0)
- A/D Control Register 1 (ADCON1)

A device reset forces all registers to their reset state. This forces the A/D module to be turned off, and any conversion is aborted.

The ADCON0 register, shown in Figure 8-1, controls the operation of the A/D module. The ADCON1 register, shown in Figure 8-2, configures the functions of the port pins. The port pins can be configured as analog inputs (RA3 can also be a voltage reference) or as digital I/O.

### FIGURE 8-1: ADCON0 REGISTER (ADDRESS 1Fh)

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | U-0 | R/W-0 |
|-------|-------|-------|-------|-------|-------|-----|-------|
| ADCS1 | ADCS0 | CHS2 | CHS1 | CHS0 | GO/$\overline{\text{DONE}}$ | — | $\overline{\text{ADON}}$ |

bit7                                       bit0

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

bit 7-6: **ADCS1:ADCS0:** A/D Conversion Clock Select bits
00 = $F_{OSC}$/2
01 = $F_{OSC}$/8
10 = $F_{OSC}$/32
11 = $F_{RC}$ (clock derived from the internal ADC RC oscillator)

bit 5-3: **CHS2:CHS0**: Analog Channel Select bits
000 = channel 0, (RA0/AN0)
001 = channel 1, (RA1/AN1)
010 = channel 2, (RA2/AN2)
011 = channel 3, (RA3/AN3)
1xx = reserved, do not use

bit 2: **GO/$\overline{\text{DONE}}$:** A/D Conversion Status bit

If ADON = 1
1 = A/D conversion in progress (setting this bit starts the A/D conversion)
0 = A/D conversion not in progress (This bit is automatically cleared by hardware when the A/D conversion is complete)

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

bit 0: **ADON**: A/D On bit
1 = A/D converter module is operating
0 = A/D converter module is shutoff and consumes no operating current

# PIC16C712/716

**FIGURE 8-2:** **ADCON1 REGISTER (ADDRESS 9Fh)**

| U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 |
|-----|-----|-----|-----|-----|-------|-------|-------|
| — | — | — | — | — | PCFG2 | PCFG1 | PCFG0 |

bit7                                                                           bit0

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

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

bit 2-0:   **PCFG2:PCFG0**: A/D Port Configuration Control bits

| PCFG2:PCFG0 | RA0 | RA1 | RA2 | RA3 | VREF |
|-------------|-----|-----|-----|------|------|
| 0x0 | A | A | A | A | VDD |
| 0x1 | A | A | A | VREF | RA3 |
| 100 | A | A | D | A | VDD |
| 101 | A | A | D | VREF | RA3 |
| 11x | D | D | D | D | VDD |

A = Analog input

D = Digital I/O

**Preliminary**
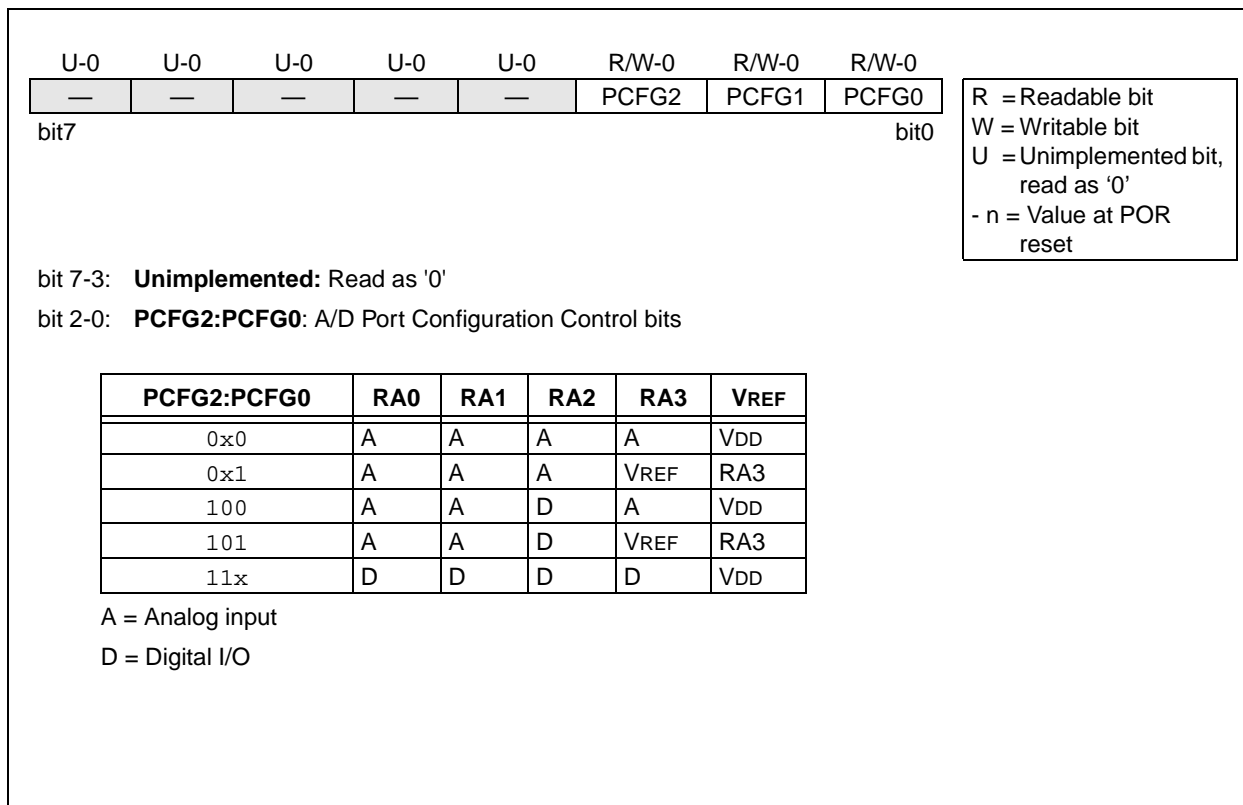
The ADRES register contains the result of the A/D conversion. When the A/D conversion is complete, the result is loaded into the ADRES register, the GO/$\overline{\text{DONE}}$ bit (ADCON0<2>) is cleared and the A/D interrupt flag bit ADIF is set. The block diagram of the A/D module is shown in Figure 8-3.

The value that is in the ADRES register is not modified for a Power-on Reset. The ADRES register will contain unknown data after a Power-on Reset.

After the A/D module has been configured as desired, the selected channel must be acquired before the conversion is started. The analog input channels must have their corresponding TRIS bits selected as an input. To determine acquisition time, see Section 8.1. After this acquisition time has elapsed, the A/D conversion can be started. The following steps should be followed for doing an A/D conversion:

1. Configure the A/D module:
   - Configure analog pins/voltage reference/ and digital I/O (ADCON1)
   - Select A/D input channel (ADCON0)
   - Select A/D conversion clock (ADCON0)
   - Turn on A/D module (ADCON0)
2. Configure A/D interrupt (if desired):
   - Clear ADIF bit
   - Set ADIE bit
   - Set GIE bit
3. Wait the required acquisition time.
4. Start conversion:
   - Set GO/$\overline{\text{DONE}}$ bit (ADCON0)
5. Wait for A/D conversion to complete, by either:
   - Polling for the GO/$\overline{\text{DONE}}$ bit to be cleared

   OR

   - Waiting for the A/D interrupt
6. Read A/D Result register (ADRES), clear bit ADIF if required.
7. For the next conversion, go to step 1 or step 2 as required. The A/D conversion time per bit is defined as $T_{AD}$. A minimum wait of $2T_{AD}$ is required before next acquisition starts.

**FIGURE 8-3: A/D BLOCK DIAGRAM**
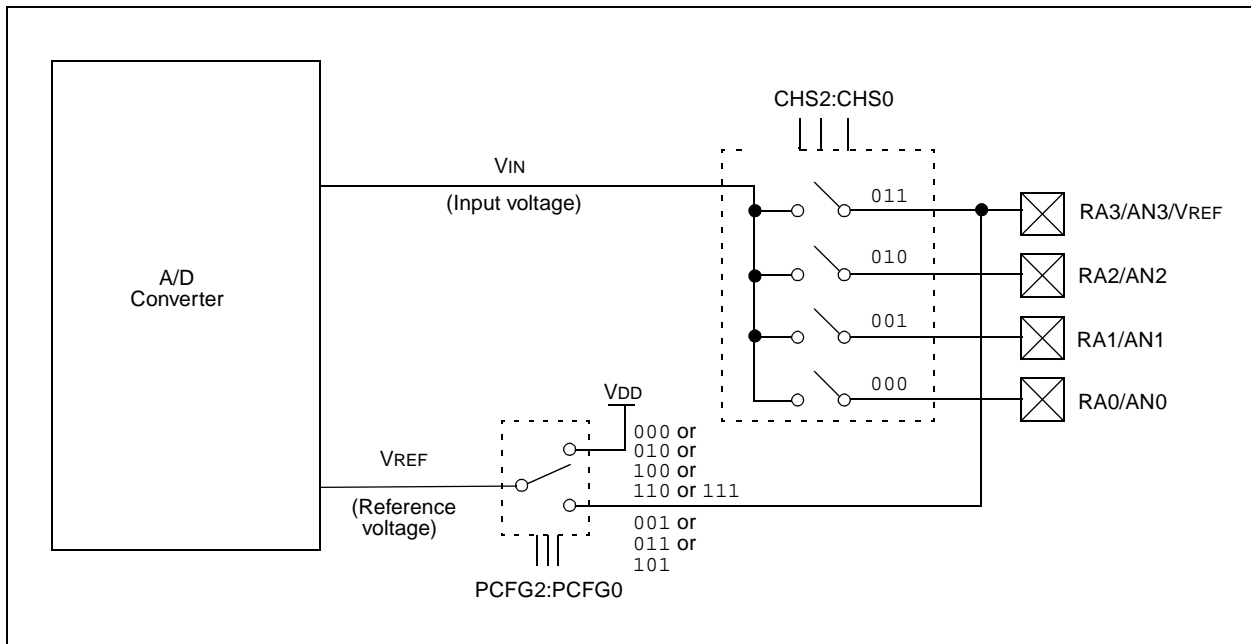
# PIC16C712/716

## 8.1    A/D Acquisition Requirements

For the A/D converter to meet its specified accuracy, the charge holding capacitor ($C_{HOLD}$) must be allowed to fully charge to the input channel voltage level. The analog input model is shown in Figure 8-4. The source impedance ($R_S$) and the internal sampling switch ($R_{SS}$) impedance directly affect the time required to charge the capacitor $C_{HOLD}$. The sampling switch ($R_{SS}$) impedance varies over the device voltage ($V_{DD}$). The source impedance affects the offset voltage at the analog input (due to pin leakage current). **The maximum recommended impedance for analog sources is 10 k$\Omega$.** After the analog input channel is selected (changed) this acquisition must be done before the conversion can be started.

To calculate the minimum acquisition time, $T_{ACQ}$, see the PICmicro™ Mid-Range Reference Manual, (DS33023). This equation calculates the acquisition time to within 1/2 LSb error (512 steps for the A/D). The 1/2 LSb error is the maximum error allowed for the A/D to meet its specified accuracy.

| **Note:** | When the conversion is started, the holding capacitor is disconnected from the input pin. |
|---|---|

**FIGURE 8-4:    ANALOG INPUT MODEL**



**Preliminary**                    © 1999 Microchip Technology Inc.
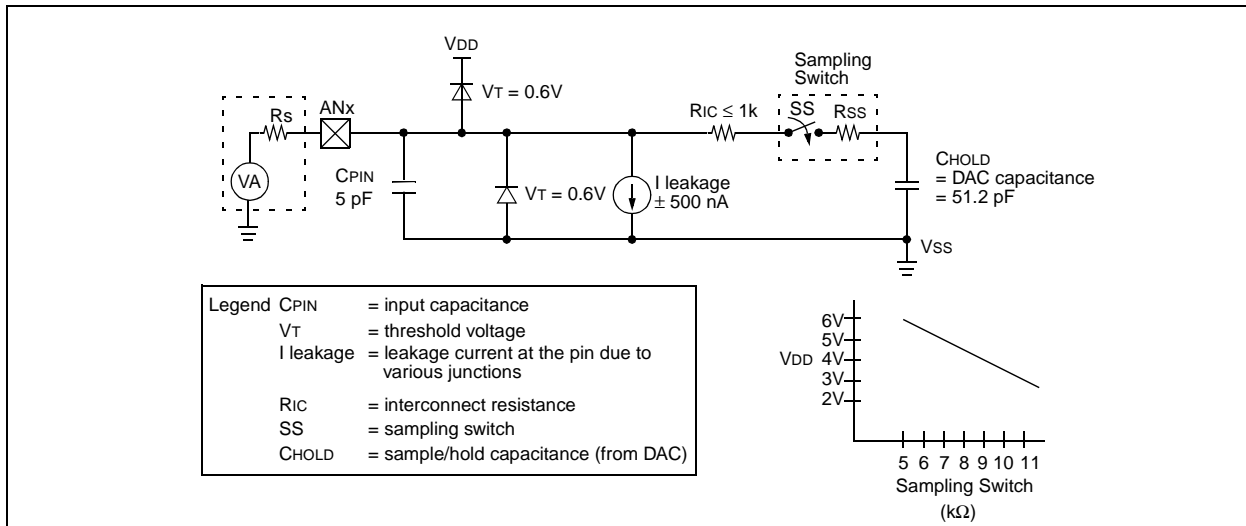
## 8.2 Selecting the A/D Conversion Clock

The A/D conversion time per bit is defined as $T_{AD}$. The A/D conversion requires $9.5 T_{AD}$ per 8-bit conversion. The source of the A/D conversion clock is software selectable. The four possible options for $T_{AD}$ are:

- $2T_{OSC}$
- $8T_{OSC}$
- $32T_{OSC}$
- Internal RC oscillator

For correct A/D conversions, the A/D conversion clock ($T_{AD}$) must be selected to ensure a minimum $T_{AD}$ time of 1.6 µs.

Table 8-1 shows the resultant $T_{AD}$ times derived from the device operating frequencies and the A/D clock source selected.

## 8.3 Configuring Analog Port Pins

The ADCON1 and TRISA registers control the operation of the A/D port pins. The port pins that are desired as analog inputs must have their corresponding TRIS bits set (input). If the TRIS bit is cleared (output), the digital output level ($V_{OH}$ or $V_{OL}$) will be converted.

The A/D operation is independent of the state of the CHS2:CHS0 bits and the TRIS bits.

> **Note 1:** When reading the port register, all pins configured as analog input channels will read as cleared (a low level). Pins configured as digital inputs, will convert an analog input. Analog levels on a digitally configured input will not affect the conversion accuracy.
>
> **Note 2:** Analog levels on any pin that is defined as a digital input (including the AN3:AN0 pins), may cause the input buffer to consume current that is out of the devices specification.

**TABLE 8-1    $T_{AD}$ vs. DEVICE OPERATING FREQUENCIES**

| AD Clock Source ($T_{AD}$) | | Device Frequency | | | |
|---|---|---|---|---|---|
| Operation | ADCS1:ADCS0 | 20 MHz | 5 MHz | 1.25 MHz | 333.33 kHz |
| $2T_{OSC}$ | 00 | 100 ns[2] | 400 ns[2] | 1.6 µs | 6 µs |
| $8T_{OSC}$ | 01 | 400 ns[2] | 1.6 µs | 6.4 µs | 24 µs[3] |
| $32T_{OSC}$ | 10 | 1.6 µs | 6.4 µs | 25.6 µs[3] | 96 µs[3] |
| RC[5] | 11 | 2 - 6 µs[1,4] | 2 - 6 µs[1,4] | 2 - 6 µs[1,4] | 2 - 6 µs[1] |

Legend: Shaded cells are outside of recommended range.
**Note 1:** The RC source has a typical $T_{AD}$ time of 4 µs.
　　　**2:** These values violate the minimum required $T_{AD}$ time.
　　　**3:** For faster conversion times, the selection of another clock source is recommended.
　　　**4:** When device frequency is greater than 1 MHz, the RC A/D conversion clock source is recommended for sleep operation only.
　　　**5:** For extended voltage devices (LC), please refer to Electrical Specifications section.

## 8.4    A/D Conversions

> **Note:** The GO/$\overline{\text{DONE}}$ bit should **NOT** be set in the same instruction that turns on the A/D.

## 8.5    Use of the CCP Trigger

An A/D conversion can be started by the "special event trigger" of the CCP1 module. This requires that the CCP1M3:CCP1M0 bits (CCP1CON<3:0>) be programmed as `1011` and that the A/D module is enabled (ADON bit is set). When the trigger occurs, the GO/$\overline{\text{DONE}}$ bit will be set, starting the A/D conversion, and the Timer1 counter will be reset to zero. Timer1 is reset to automatically repeat the A/D acquisition period with minimal software overhead (moving the ADRES to the desired location). The appropriate analog input channel must be selected and the minimum acquisition done before the "special event trigger" sets the GO/$\overline{\text{DONE}}$ bit (starts a conversion).

If the A/D module is not enabled (ADON is cleared), then the "special event trigger" will be ignored by the A/D module, but will still reset the Timer1 counter.

### TABLE 8-2    SUMMARY OF A/D REGISTERS

| Address | 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 |
|---------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------------------|---------------------------|
| 05h | PORTA | — | — | —(1) | RA4 | RA3 | RA2 | RA1 | RA0 | --xx xxxx | --xu uuuu |
| 0Bh,8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | — | CCP1IF | TMR2IF | TMR1IF | -0-- -000 | -0-- -000 |
| 1Eh | ADRES | A/D Result Register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 1Fh | ADCON0 | ADCS1 | ADCS0 | CHS2 | CHS1 | CHS0 | GO/$\overline{\text{DONE}}$ | — | ADON | 0000 00-0 | 0000 00-0 |
| 85h | TRISA | — | — | —(1) | PORTA Data Direction Register | | | | | ---1 1111 | ---1 1111 |
| 8Ch | PIE1 | — | ADIE | — | — | — | CCP1IE | TMR2IE | TMR1IE | -0-- -000 | -0-- 0000 |
| 9Fh | ADCON1 | — | — | — | — | — | PCFG2 | PCFG1 | PCFG0 | ---- -000 | ---- -000 |

Legend:   x = unknown, u = unchanged, – = unimplemented read as '0'. Shaded cells are not used for A/D conversion.
**Note 1:**  Reserved bits; Do Not Use.

**Preliminary**

## 9.0 SPECIAL FEATURES OF THE CPU

The PIC16C712/716 devices have a host of features intended to maximize system reliability, minimize cost through elimination of external components, provide power saving operating modes and offer code protection. These are:

• OSC Selection
• Reset
  - Power-on Reset (POR)
  - Power-up Timer (PWRT)
  - Oscillator Start-up Timer (OST)
  - Brown-out Reset (BOR)
• Interrupts
• Watchdog Timer (WDT)
• SLEEP
• Code protection
• ID locations
• In-Circuit Serial Programming™ (ICSP)

These devices have a Watchdog Timer, which can be shut off only through configuration bits. It runs off its own RC oscillator for added reliability. There are two timers that offer necessary delays on power-up. One is the Oscillator Start-up Timer (OST), intended to keep the chip in reset until the crystal oscillator is stable. The other is the Power-up Timer (PWRT), which provides a fixed delay on power-up only and is designed to keep the part in reset while the power supply stabilizes. With these two timers on-chip, most applications need no external reset circuitry.

SLEEP mode is designed to offer a very low current power-down mode. The user can wake-up from SLEEP through external reset, Watchdog Timer Wake-up, or through an interrupt. Several oscillator options are also made available to allow the part to fit the application. The RC oscillator option saves system cost, while the LP crystal option saves power. A set of configuration bits are used to select various options.

Additional information on special features is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

### 9.1 Configuration Bits

The configuration bits can be programmed (read as '0') or left unprogrammed (read as '1') to select various device configurations. These bits are mapped in program memory location 2007h.

The user will note that address 2007h is beyond the user program memory space. In fact, it belongs to the special test/configuration memory space (2000h - 3FFFh), which can be accessed only during programming.

# PIC16C712/716

## FIGURE 9-1: CONFIGURATION WORD

| CP1 | CP0 | CP1 | CP0 | CP1 | CP0 | — | BODEN | CP1 | CP0 | PWRTE | WDTE | FOSC1 | FOSC0 | Register:CONFIG Address2007h |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| bit13 | | | | | | | | | | | | | bit0 | |

bit 13-8, 5-4: **CP1:CP0**: Code Protection bits [2]

Code Protection for 2K Program memory (PIC16C716)
`11` = Programming code protection off
`10` = 0400h - 07FFh code protected
`01` = 0200h - 07FFh code protected
`00` = 0000h - 07FFh code protected

bit 13-8, 5-4:

Code Protection for 1K Program memory (PIC16C712)
`11` = Programming code protection off
`10` = Programming code protection off
`01` = 0200h - 03FFh code protected
`00` = 0000h - 03FFh code protected

bit 7:    **Unimplemented**: Read as '1'
bit 6:    **BODEN**: Brown-out Reset Enable bit [1]
`1` = BOR enabled
`0` = BOR disabled
bit 3:    **PWRTE**: Power-up Timer Enable bit [1]
`1` = PWRT disabled
`0` = PWRT enabled
bit 2:    **WDTE**: Watchdog Timer Enable bit
`1` = WDT enabled
`0` = WDT disabled
bit 1-0:    **FOSC1:FOSC0**: Oscillator Selection bits
`11` = RC oscillator
`10` = HS oscillator
`01` = XT oscillator
`00` = LP oscillator

**Note 1:** Enabling Brown-out Reset automatically enables Power-up Timer (PWRT) regardless of the value of bit PWRTE. Ensure the Power-up Timer is enabled anytime Brown-out Reset is enabled.
**2:** All of the CP1:CP0 pairs have to be given the same value to enable the code protection scheme listed.

## 9.2 Oscillator Configurations

### 9.2.1 OSCILLATOR TYPES

The PIC16CXXX can be operated in four different oscillator modes. The user can program two configuration bits (FOSC1 and FOSC0) to select one of these four modes:

• LP      Low Power Crystal
• XT      Crystal/Resonator
• HS      High Speed Crystal/Resonator
• RC      Resistor/Capacitor

### 9.2.2 CRYSTAL OSCILLATOR/CERAMIC RESONATORS

In XT, LP or HS modes, a crystal or ceramic resonator is connected to the OSC1/CLKIN and OSC2/CLKOUT pins to establish oscillation (Figure 9-2). The PIC16CXXX oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may give a frequency out of the crystal manufacturers specifications. When in XT, LP or HS modes, the device can have an external clock source to drive the OSC1/CLKIN pin (Figure 9-3).

**FIGURE 9-2: CRYSTAL/CERAMIC RESONATOR OPERATION (HS, XT OR LP OSC CONFIGURATION)**



Note 1: See Table 9-1 and Table 9-2 for recommended values of C1 and C2.
2: A series resistor (RS) may be required for AT strip cut crystals.
3: RF varies with the crystal chosen.
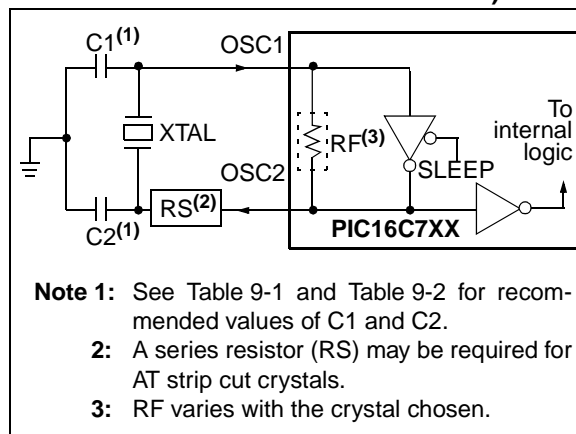
**FIGURE 9-3: EXTERNAL CLOCK INPUT OPERATION (HS, XT OR LP OSC CONFIGURATION)**
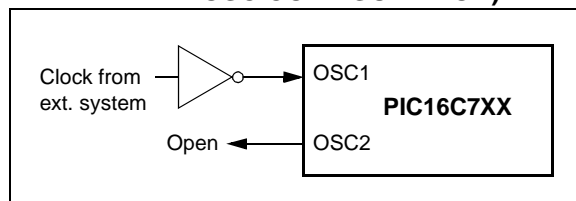


**TABLE 9-1 CERAMIC RESONATORS**

| Ranges Tested: | | | |
|---|---|---|---|
| **Mode** | **Freq** | **OSC1** | **OSC2** |
| XT | 455 kHz | 68 - 100 pF | 68 - 100 pF |
| | 2.0 MHz | 15 - 68 pF | 15 - 68 pF |
| | 4.0 MHz | 15 - 68 pF | 15 - 68 pF |
| HS | 8.0 MHz | 10 - 68 pF | 10 - 68 pF |
| | 16.0 MHz | 10 - 22 pF | 10 - 22 pF |

**These values are for design guidance only.** See notes at bottom of page.

**TABLE 9-2 CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR**

| Osc Type | Crystal Freq | Cap. Range C1 | Cap. Range C2 |
|---|---|---|---|
| LP | 32 kHz | 33 pF | 33 pF |
| | 200 kHz | 15 pF | 15 pF |
| XT | 200 kHz | 47-68 pF | 47-68 pF |
| | 1 MHz | 15 pF | 15 pF |
| | 4 MHz | 15 pF | 15 pF |
| HS | 4 MHz | 15 pF | 15 pF |
| | 8 MHz | 15-33 pF | 15-33 pF |
| | 20 MHz | 15-33 pF | 15-33 pF |

**These values are for design guidance only.** See notes at bottom of page.

Note 1: Recommended values of C1 and C2 are identical to the ranges tested (Table 9-1).
2: Higher capacitance increases the stability of the oscillator, but also increases the start-up time.
3: Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components.
4: Rs may be required in HS mode, as well as XT mode to avoid overdriving crystals with low drive level specification.

### 9.2.3    RC OSCILLATOR

For timing insensitive applications, the "RC" device option offers additional cost savings. The RC oscillator frequency is a function of the supply voltage, the resistor ($R_{EXT}$) and capacitor ($C_{EXT}$) values and the operating temperature. In addition to this, the oscillator frequency will vary from unit to unit due to normal process parameter variation. Furthermore, the difference in lead frame capacitance between package types will also affect the oscillation frequency, especially for low $C_{EXT}$ values. The user also needs to take into account variation due to tolerance of external R and C components used. Figure 9-4 shows how the R/C combination is connected to the PIC16CXXX.

**FIGURE 9-4:    RC OSCILLATOR MODE**
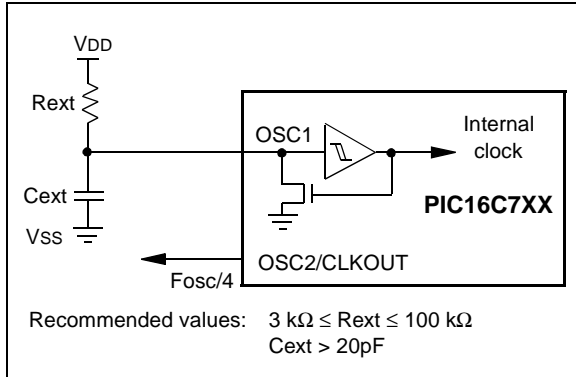


### 9.3    Reset

The PIC16CXXX differentiates between various kinds of reset:

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

Some registers are not affected in any reset condition; their status is unknown on POR and unchanged in any other reset. Most other registers are reset to a "reset state" on Power-on Reset (POR), on the $\overline{MCLR}$ and WDT Reset, on $\overline{MCLR}$ reset during SLEEP and Brown-out Reset (BOR). They are not affected by a WDT Wake-up, which is viewed as the resumption of normal operation. The $\overline{TO}$ and $\overline{PD}$ bits are set or cleared differently in different reset situations as indicated in Table 9-4. These bits are used in software to determine the nature of the reset. See Table 9-6 for a full description of reset states of all registers.

A simplified block diagram of the on-chip reset circuit is shown in Figure 9-6.

The PICmicro microcontrollers have a $\overline{MCLR}$ noise filter in the $\overline{MCLR}$ reset path. The filter will detect and ignore small pulses.

It should be noted that a WDT Reset does not drive $\overline{MCLR}$ pin low.

## 9.4    Power-On Reset (POR)

A Power-on Reset pulse is generated on-chip when VDD rise is detected (to a level of 1.5V - 2.1V). To take advantage of the POR, just tie the $\overline{MCLR}$ pin directly (or through a resistor) to VDD. This will eliminate external RC components usually needed to create a Power-on Reset. A maximum rise time for VDD is specified (parameter D004). For a slow rise time, see Figure 9-5.

When the device starts normal operation (exits the reset condition), device operating parameters (voltage, frequency, temperature,...) must be met to ensure operation. If these conditions are not met, the device must be held in reset until the operating conditions are met. Brown-out Reset may be used to meet the start-up conditions.

### FIGURE 9-5:    EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW VDD POWER-UP)



**Note 1:**  External Power-on Reset circuit is required only if VDD power-up slope is too slow. The diode D helps discharge the capacitor quickly when VDD powers down.

**2:**  R < 40 kΩ is recommended to make sure that voltage drop across R does not violate the device's electrical specification.

**3:**  R1 = 100Ω to 1 kΩ will limit any current flowing into $\overline{MCLR}$ from external capacitor C in the event of $\overline{MCLR}$/VPP pin breakdown due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS).

## 9.5    Power-up Timer (PWRT)

The Power-up Timer provides a fixed nominal time-out (parameter #33), on power-up only, from the POR. The Power-up Timer operates on an internal RC oscillator. The chip is kept in reset as long as the PWRT is active. The PWRT's time delay allows VDD to rise to an acceptable level. A configuration bit is provided to enable/disable the PWRT.

The power-up time delay will vary from chip to chip due to VDD, temperature, and process variation. See DC parameters for details.

## 9.6    Oscillator Start-up Timer (OST)

The Oscillator Start-up Timer (OST) provides a 1024 oscillator cycle (from OSC1 input) delay after the PWRT delay is over (parameter #32). This ensures that the crystal oscillator or resonator has started and stabilized.

The OST time-out is invoked only for XT, LP and HS modes and only on Power-on Reset or wake-up from SLEEP.

## 9.7    Brown-Out Reset (BOD)

The PIC16C712/716 members have on-chip Brown-out Reset circuitry. A configuration bit, BODEN, can disable (if clear/programmed) or enable (if set) the Brown-out Reset circuitry. If VDD falls below 4.0V, refer to VBOR parameter D005(VBOR) for a time greater than parameter (TBOR) in Table 12-6.  The brown-out situation will reset the chip. A reset is not guaranteed to occur if VDD falls below 4.0V for less than parameter (TBOR).

On any reset (Power-on, Brown-out, Watchdog, etc.) the chip will remain in Reset until VDD rises above VBOR. The Power-up Timer will now be invoked and will keep the chip in reset an additional 72 ms.

If VDD drops below VBOR while the Power-up Timer is running, the chip will go back into a Brown-out Reset and the Power-up Timer will be re-initialized. Once VDD rises above VBOR, the Power-Up Timer will execute a 72 ms reset. The Power-up Timer should always be enabled when Brown-out Reset is enabled. Figure 9-7 shows typical Brown-out situations.

For operations where the desired brown-out voltage is other than 4V, an external brown-out circuit must be used. Figure 9-8, 9-9 and 9-10 show examples of external brown-out protection circuits.

# PIC16C712/716

FIGURE 9-6: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



**Note 1:** This is a separate oscillator from the RC oscillator of the CLKIN pin.
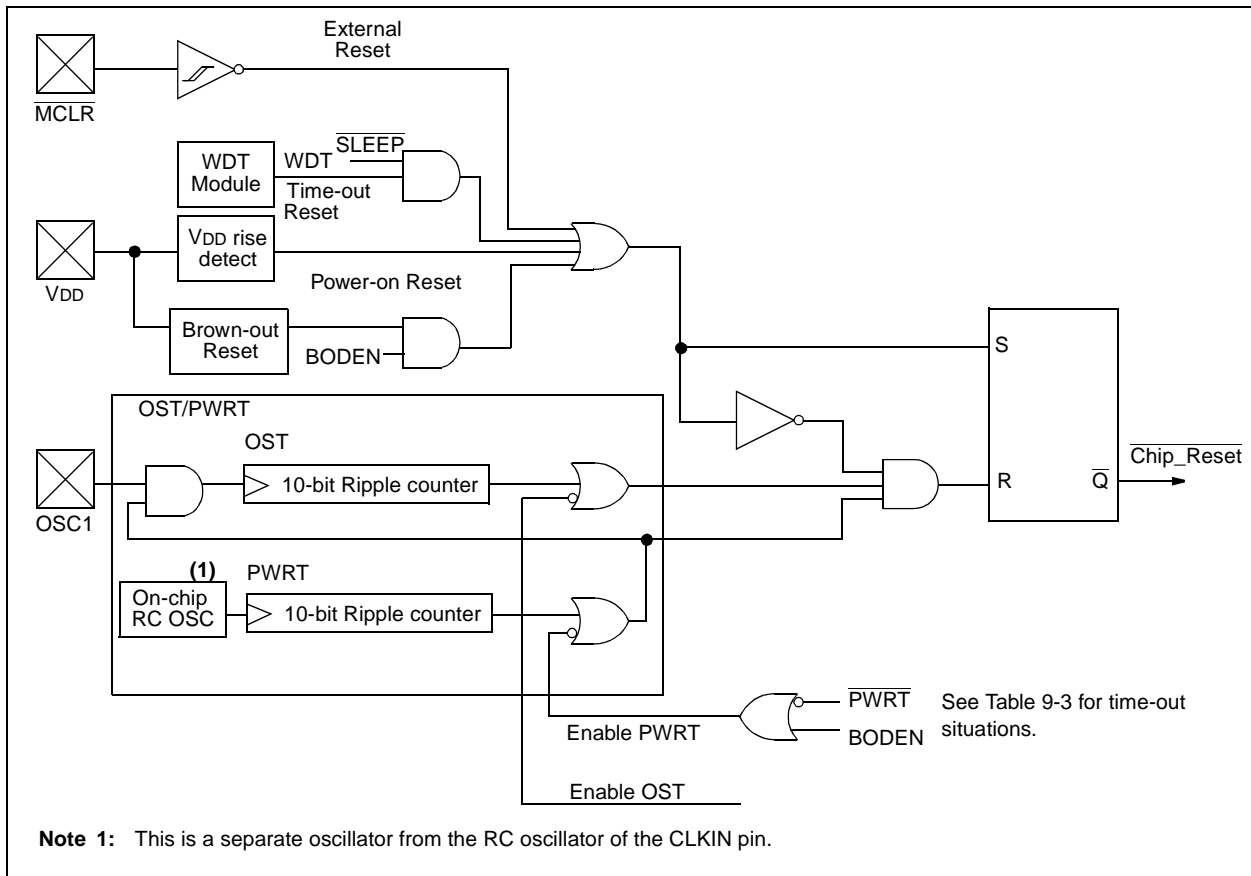
FIGURE 9-7: BROWN-OUT SITUATIONS



**Preliminary** © 1999 Microchip Technology Inc.
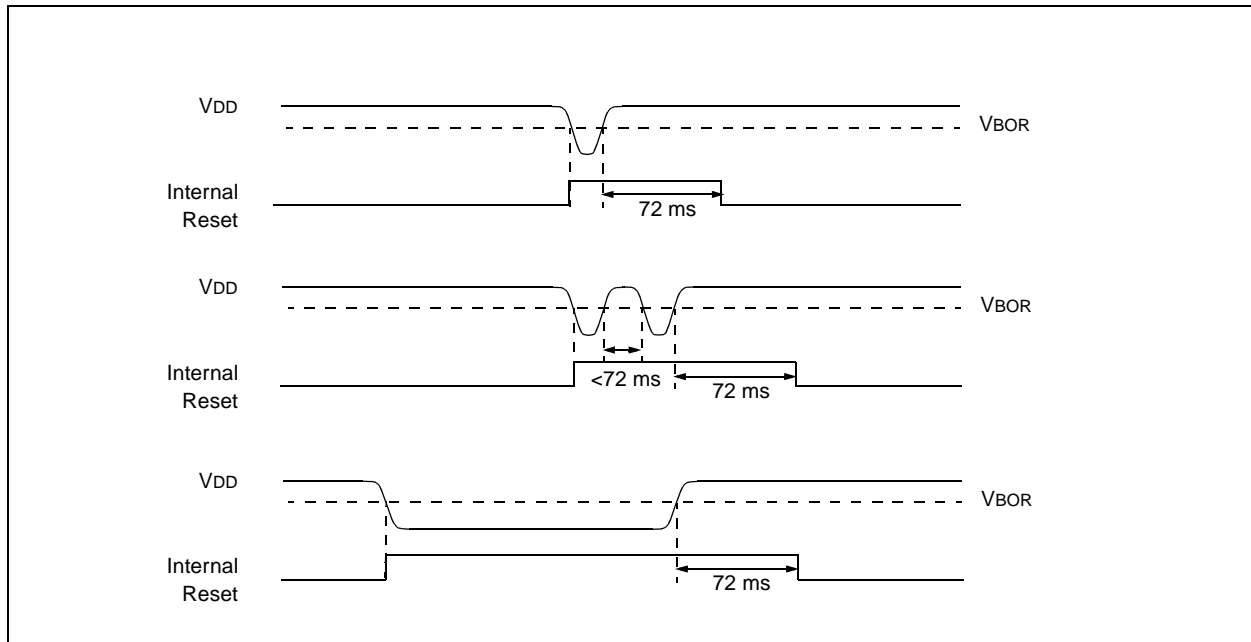
**FIGURE 9-8: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 1**
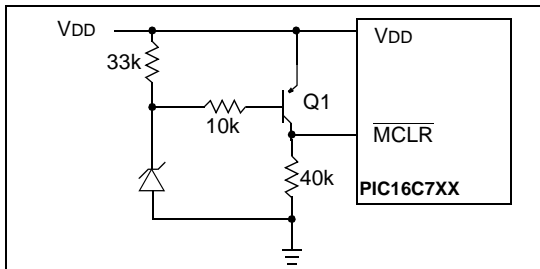


**Note 1:** This circuit will activate reset when VDD goes below (Vz + 0.7V) where Vz = Zener voltage.

**2:** Internal Brown-out Reset circuitry should be disabled when using this circuit.

**FIGURE 9-9: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 2**
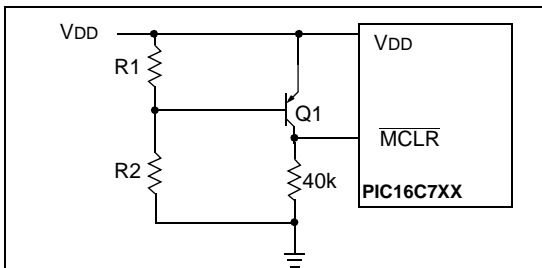


**Note 1:** This brown-out circuit is less expensive, albeit less accurate. Transistor Q1 turns off when VDD is below a certain level such that:

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

**2:** Internal brown-out reset should be disabled when using this circuit.

**3:** Resistors should be adjusted for the characteristics of the transistor.

**FIGURE 9-10: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 3**
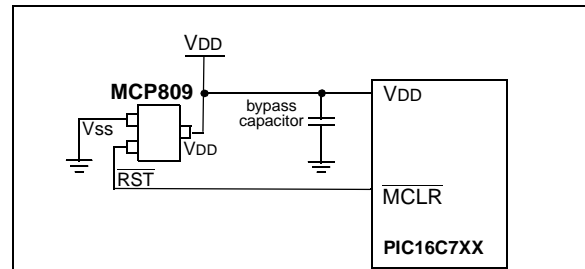


This brown-out protection circuit employs Microchip Technology's MCP809 microcontroller supervisor. The MCP8XX and MCP1XX families of supervisors provide push-pull and open collector outputs with both high and low active reset pins. There are 7 different trip point selections to accommodate 5V and 3V systems

## 9.8 Time-out Sequence

On power-up the time-out sequence is as follows: First PWRT time-out is invoked after the POR time delay has expired. Then OST is activated. The total time-out will vary based on oscillator configuration and the status of the PWRT. For example, in RC mode with the PWRT disabled, there will be no time-out at all. Figure 9-11, Figure 9-12, and Figure 9-13 depict time-out sequences on power-up.

Since the time-outs occur from the POR pulse, if MCLR is kept low long enough, the time-outs will expire. Then bringing MCLR high will begin execution immediately (Figure 9-13). This is useful for testing purposes or to synchronize more than one PIC16CXXX device operating in parallel.

Table 9-5 shows the reset conditions for some special function registers, while Table 9-6 shows the reset conditions for all the registers.

## 9.9 Power Control/Status Register (PCON)

The Power Control/Status Register, PCON has two bits.

Bit0 is Brown-out Reset Status bit, BOR. If the BODEN configuration bit is set, BOR is '1' on Power-on Reset. If the BODEN configuration bit is clear, BOR is unknown on Power-on Reset.

The BOR status bit is a "don't care" and is not necessarily predictable if the brown-out circuit is disabled (the BODEN configuration bit is clear). BOR must then be set by the user and checked on subsequent resets to see if it is clear, indicating a brown-out has occurred.

Bit1 is POR (Power-on Reset Status bit). It is cleared on a Power-on Reset and unaffected otherwise. The user must set this bit following a Power-on Reset.

**TABLE 9-3    TIME-OUT IN VARIOUS SITUATIONS**

| Oscillator Configuration | Power-up | | Brown-out | Wake-up from SLEEP |
|---|---|---|---|---|
| | $\overline{\text{PWRTE}}$ = 0 | $\overline{\text{PWRTE}}$ = 1 | | |
| XT, HS, LP | 72 ms + 1024T$_{OSC}$ | 1024T$_{OSC}$ | 72 ms + 1024T$_{OSC}$ | 1024T$_{OSC}$ |
| RC | 72 ms | — | 72 ms | — |

**TABLE 9-4    STATUS BITS AND THEIR SIGNIFICANCE**

| $\overline{\text{POR}}$ | $\overline{\text{BOR}}$ | $\overline{\text{TO}}$ | $\overline{\text{PD}}$ | |
|---|---|---|---|---|
| 0 | x | 1 | 1 | Power-on Reset |
| 0 | x | 0 | x | Illegal, $\overline{\text{TO}}$ is set on $\overline{\text{POR}}$ |
| 0 | x | x | 0 | Illegal, $\overline{\text{PD}}$ is set on $\overline{\text{POR}}$ |
| 1 | 0 | 1 | 1 | Brown-out Reset |
| 1 | 1 | 0 | 1 | WDT Reset |
| 1 | 1 | 0 | 0 | WDT Wake-up |
| 1 | 1 | u | u | $\overline{\text{MCLR}}$ Reset during normal operation |
| 1 | 1 | 1 | 0 | $\overline{\text{MCLR}}$ Reset during SLEEP or interrupt wake-up from SLEEP |

**TABLE 9-5    RESET CONDITION FOR SPECIAL REGISTERS**

| Condition | Program Counter | STATUS Register | PCON Register |
|---|---|---|---|
| Power-on Reset | 000h | 0001 1xxx | ---- --0x |
| $\overline{\text{MCLR}}$ Reset during normal operation | 000h | 000u uuuu | ---- --uu |
| $\overline{\text{MCLR}}$ Reset during SLEEP | 000h | 0001 0uuu | ---- --uu |
| WDT Reset | 000h | 0000 1uuu | ---- --uu |
| WDT Wake-up | PC + 1 | uuu0 0uuu | ---- --uu |
| Brown-out Reset | 000h | 0001 1uuu | ---- --u0 |
| Interrupt wake-up from SLEEP | PC + 1[1] | uuu1 0uuu | ---- --uu |

Legend: u = unchanged, x = unknown, – = unimplemented bit read as '0'.
**Note 1:** When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

**TABLE 9-6      INITIALIZATION CONDITIONS FOR ALL REGISTERS OF THE PIC16C712/716**

| Register | Power-on Reset, Brown-out Reset | $\overline{\text{MCLR}}$ Resets WDT Reset | Wake-up via WDT or Interrupt |
|---|---|---|---|
| W | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| INDF | N/A | N/A | N/A |
| TMR0 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PCL | 0000h | 0000h | PC + 1 **(2)** |
| STATUS | 0001 1xxx | 000q quuu **(3)** | uuuq quuu **(3)** |
| FSR | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PORTA **(4)** | --0x 0000 | --xx xxxx | --xu uuuu |
| PORTB **(5)** | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| DATACCP | ---- -x-x | ---- -u-u | ---- -u-u |
| PCLATH | ---0 0000 | ---0 0000 | ---u uuuu |
| INTCON | 0000 -00x | 0000 -00u | uuuu -uuu **(1)** |
| PIR1 | ---- 0000 | ---- 0000 | ---- uuuu **(1)** |
| | -0-- 0000 | -0-- 0000 | -u-- uuuu **(1)** |
| TMR1L | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| TMR1H | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| T1CON | --00 0000 | --uu uuuu | --uu uuuu |
| TMR2 | 0000 0000 | 0000 0000 | uuuu uuuu |
| T2CON | -000 0000 | -000 0000 | -uuu uuuu |
| CCPR1L | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| CCPR1H | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| CCP1CON | --00 0000 | --00 0000 | --uu uuuu |
| ADRES | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| ADCON0 | 0000 00-0 | 0000 00-0 | uuuu uu-u |
| OPTION_REG | 1111 1111 | 1111 1111 | uuuu uuuu |
| TRISA | --11 1111 | --11 1111 | --uu uuuu |
| TRISB | 1111 1111 | 1111 1111 | uuuu uuuu |
| TRISCCP | xxxx x1x1 | xxxx x1x1 | xxxx xuxu |
| PIE1 | ---- 0000 | ---- 0000 | ---- uuuu |
| | -0-- 0000 | -0-- 0000 | -u-- uuuu |
| PCON | ---- --0q | ---- --uq | ---- --uq |
| PR2 | 1111 1111 | 1111 1111 | 1111 1111 |
| ADCON1 | ---- -000 | ---- -000 | ---- -uuu |

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

**Note 1:** One or more bits in INTCON and/or PIR1 will be affected (to cause wake-up).
   **2:** When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).
   **3:** See Table 9-5 for reset value for specific condition.
   **4:** On any device reset, these pins are configured as inputs.
   **5:** This is the value that will be in the port output latch.

**FIGURE 9-11:** **TIME-OUT SEQUENCE ON POWER-UP (MCLR TIED TO V_DD)**
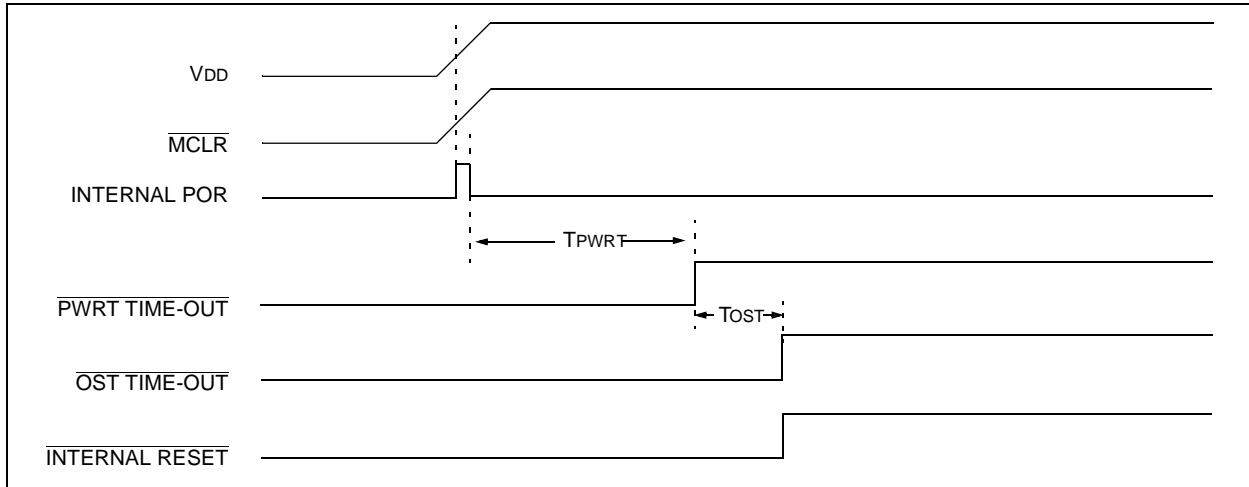


**FIGURE 9-12:** **TIME-OUT SEQUENCE ON POWER-UP (MCLR NOT TIED TO V_DD): CASE 1**
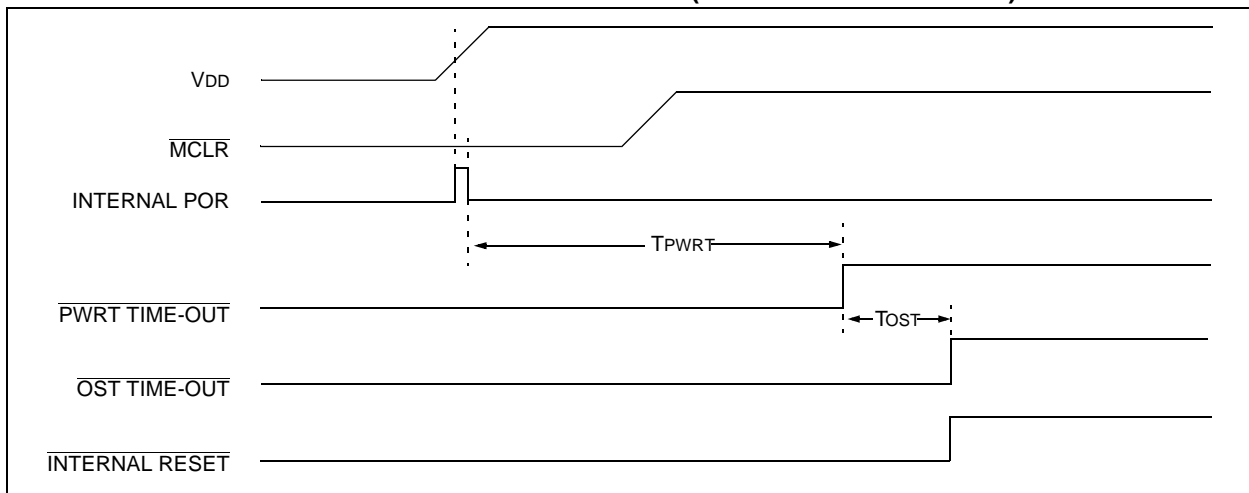


**FIGURE 9-13:** **TIME-OUT SEQUENCE ON POWER-UP (MCLR NOT TIED TO V_DD): CASE 2**
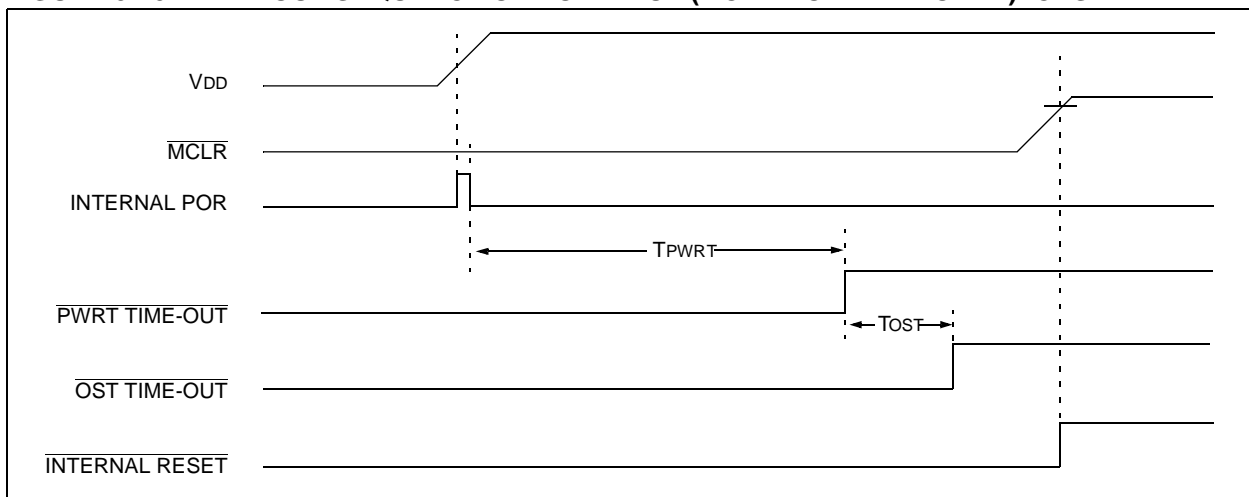
## 9.10 Interrupts

The PIC16C712/716 devices have up to 7 sources of interrupt. The interrupt control register (INTCON) records individual interrupt requests in flag bits. It also has individual and global interrupt enable bits.

> **Note:** Individual interrupt flag bits are set regardless of the status of their corresponding mask bit or the GIE bit.

A global interrupt enable bit, GIE (INTCON<7>) enables (if set) all un-masked interrupts or disables (if cleared) all interrupts. When bit GIE is enabled, and an interrupt's flag bit and mask bit are set, the interrupt will vector immediately. Individual interrupts can be disabled through their corresponding enable bits in various registers. Individual interrupt bits are set, regardless of the status of the GIE bit. The GIE bit is cleared on reset.

The "return from interrupt" instruction, RETFIE, exits the interrupt routine, as well as sets the GIE bit, which re-enables interrupts.

The RB0/INT pin interrupt, the RB port change interrupt and the TMR0 overflow interrupt flags are contained in the INTCON register.

The peripheral interrupt flags are contained in the special function registers, PIR1 and PIR2. The corresponding interrupt enable bits are contained in special function registers, PIE1 and PIE2, and the peripheral interrupt enable bit is contained in special function register, INTCON.

When an interrupt is responded to, the GIE bit is cleared to disable any further interrupt, the return address is pushed onto the stack and the PC is loaded with 0004h. Once in the interrupt service routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid recursive interrupts.

For external interrupt events, such as the INT pin or PORTB change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends when the interrupt event occurs. The latency is the same for one or two cycle instructions. Individual interrupt flag bits are set, regardless of the status of their corresponding mask bit or the GIE bit.

**FIGURE 9-14: INTERRUPT LOGIC**

# PIC16C712/716
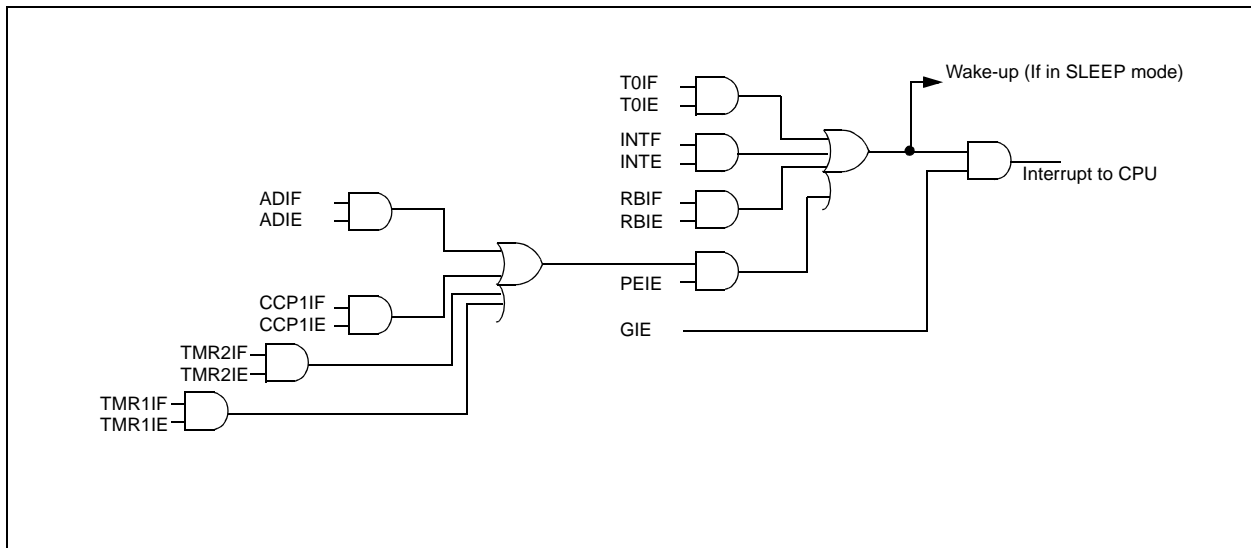
## 9.10.1 INT INTERRUPT

External interrupt on RB0/INT pin is edge triggered, either rising if bit INTEDG (OPTION_REG<6>) is set, or falling if the INTEDG bit is clear. When a valid edge appears on the RB0/INT pin, flag bit INTF (INTCON<1>) is set. This interrupt can be disabled by clearing enable bit INTE (INTCON<4>). Flag bit INTF must be cleared in software in the interrupt service routine before re-enabling this interrupt. The INT interrupt can wake-up the processor from SLEEP, if bit INTE was set prior to going into SLEEP. The status of global interrupt enable bit GIE decides whether or not the processor branches to the interrupt vector following wake-up. See Section 9.13 for details on SLEEP mode.

## 9.10.2 TMR0 INTERRUPT

An overflow (FFh $\rightarrow$ 00h) in the TMR0 register will set flag bit T0IF (INTCON<2>). The interrupt can be enabled/disabled by setting/clearing enable bit T0IE (INTCON<5>). (Section 4.0)

## 9.10.3 PORTB INTCON CHANGE

An input change on PORTB<7:4> sets flag bit RBIF (INTCON<0>). The interrupt can be enabled/disabled by setting/clearing enable bit RBIE (INTCON<4>). (Section 3.2)

## 9.11 Context Saving During Interrupts

During an interrupt, only the return PC value is saved on the stack. Typically, users may wish to save key registers during an interrupt, (i.e., W register and STATUS register). This will have to be implemented in software.

Example 9-1 stores and restores the W and STATUS registers. The register, W_TEMP, must be defined in each bank and must be defined at the same offset from the bank base address (i.e., if W_TEMP is defined at 0x20 in bank 0, it must also be defined at 0xA0 in bank 1).

The example:

a)  Stores the W register.
b)  Stores the STATUS register in bank 0.
c)  Stores the PCLATH register.
d)  Executes the interrupt service routine code (User-generated).
e)  Restores the STATUS register (and bank select bit).
f)  Restores the W and PCLATH registers.

**EXAMPLE 9-1:   SAVING STATUS, W, AND PCLATH REGISTERS IN RAM**

```
    MOVWF    W_TEMP           ;Copy W to TEMP register, could be bank one or zero
    SWAPF    STATUS,W         ;Swap status to be saved into W
    CLRF     STATUS           ;bank 0, regardless of current bank, Clears IRP,RP1,RP0
    MOVWF    STATUS_TEMP      ;Save status to bank zero STATUS_TEMP register
    MOVF     PCLATH, W        ;Only required if using pages 1, 2 and/or 3
    MOVWF    PCLATH_TEMP      ;Save PCLATH into W
    CLRF     PCLATH           ;Page zero, regardless of current page
    BCF      STATUS, IRP      ;Return to Bank 0
    MOVF     FSR, W           ;Copy FSR to W
    MOVWF    FSR_TEMP         ;Copy FSR from W to FSR_TEMP
    :
    :(ISR)
    :
    MOVF     PCLATH_TEMP, W   ;Restore PCLATH
    MOVWF    PCLATH           ;Move W into PCLATH
    SWAPF    STATUS_TEMP,W    ;Swap STATUS_TEMP register into W
                             ;(sets bank to original state)
    MOVWF    STATUS           ;Move W into STATUS register
    SWAPF    W_TEMP,F         ;Swap W_TEMP
    SWAPF    W_TEMP,W         ;Swap W_TEMP into W
```

## 9.12 Watchdog Timer (WDT)

The Watchdog Timer is as a free running, on-chip, RC oscillator which does not require any external components. This RC oscillator is separate from the RC oscillator of the OSC1/CLKIN pin. That means that the WDT will run, even if the clock on the OSC1/CLKIN and OSC2/CLKOUT pins of the device have been stopped, for example, by execution of a SLEEP instruction.

During normal operation, a WDT time-out generates a device RESET (Watchdog Timer Reset). If the device is in SLEEP mode, a WDT time-out causes the device to wake-up and continue with normal operation (Watchdog Timer Wake-up). The $\overline{TO}$ bit in the STATUS register will be cleared upon a Watchdog Timer time-out.

The WDT can be permanently disabled by clearing configuration bit WDTE (Section 9.1).

WDT time-out period values may be found in the Electrical Specifications section under TWDT (parameter #31). Values for the WDT prescaler (actually a postscaler, but shared with the Timer0 prescaler) may be assigned using the OPTION_REG register.

> **Note:** The CLRWDT and SLEEP instructions clear the WDT and the postscaler, if assigned to the WDT, and prevent it from timing out and generating a device RESET condition.

.

> **Note:** When a CLRWDT instruction is executed and the prescaler is assigned to the WDT, the prescaler count will be cleared, but the prescaler assignment is not changed.

### FIGURE 9-15: WATCHDOG TIMER BLOCK DIAGRAM
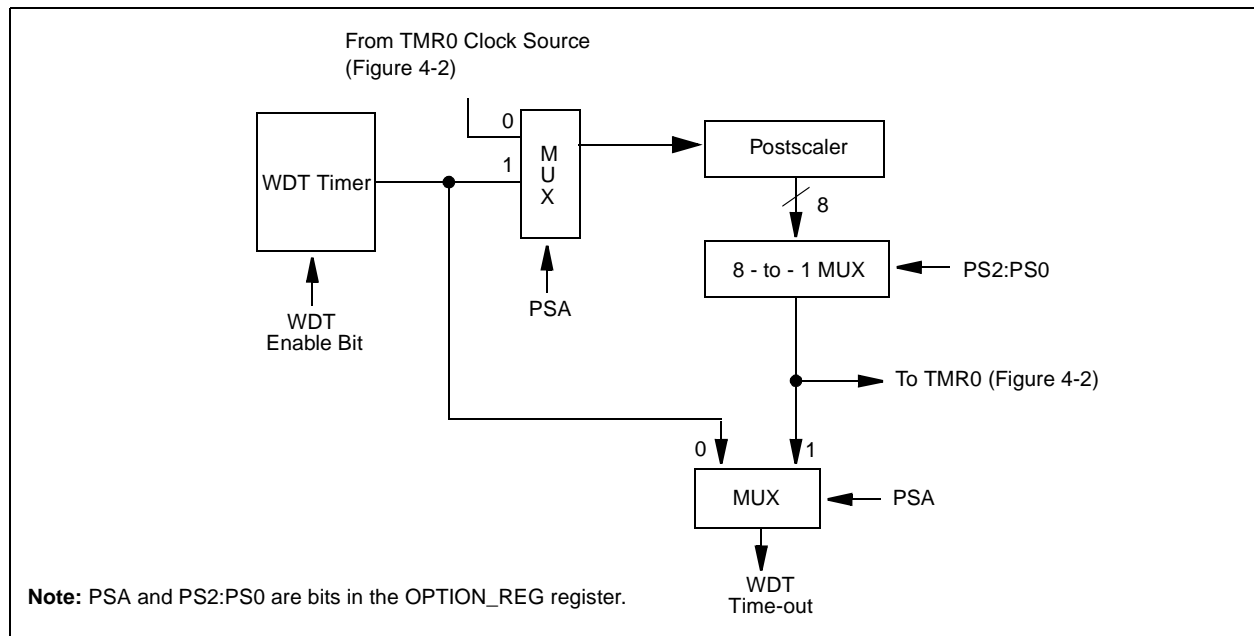


### FIGURE 9-16: SUMMARY OF WATCHDOG TIMER REGISTERS

| Address | Name | Bits 13:8 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---------|------|-----------|-------|-------|-------|-------|-------|-------|-------|-------|
| 2007h | Config. bits | **(1)** | — | BODEN[1] | $\overline{CP1}$ | $\overline{CP0}$ | $\overline{PWRTE}$[1] | WDTE | FOSC1 | FOSC0 |
| 81h | OPTION_REG | N/A | $\overline{RBPU}$ | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 |

Legend: Shaded cells are not used by the Watchdog Timer.
**Note 1:** See Figure 9-1 for operation of these bits.

## 9.13    Power-down Mode (SLEEP)

Power-down mode is entered by executing a SLEEP instruction.

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

For lowest current consumption in this mode, place all I/O pins at either VDD or VSS, ensure no external circuitry is drawing current from the I/O pin, power-down the A/D and the disable external clocks. Pull all I/O pins, that are hi-impedance inputs, high or low externally to avoid switching currents caused by floating inputs. The T0CKI input should also be at VDD or VSS for lowest current consumption. The contribution from on-chip pull-ups on PORTB should be considered.

The $\overline{MCLR}$ pin must be at a logic high level (VIHMC).

### 9.13.1    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 INT pin, RB port change, or some peripheral interrupts.

External $\overline{MCLR}$ Reset will cause a device reset. All other events are considered a continuation of program execution and cause a "wake-up". 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. The $\overline{TO}$ bit is cleared if a WDT time-out occurred (and caused wake-up).

The following peripheral interrupts can wake the device from SLEEP:

1.  TMR1 interrupt.  Timer1 must be operating as an asynchronous counter.
2.  CCP capture mode interrupt.
3.  Special event trigger (Timer1 in asynchronous mode using an external clock).

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 pre-fetched. 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 and 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.

### 9.13.2    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 postscaler will not be cleared, the $\overline{TO}$ bit will not be set and $\overline{PD}$ bits 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 will be completely executed before the wake-up. Therefore, the WDT and WDT postscaler 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.

**Preliminary**

**FIGURE 9-17: WAKE-UP FROM SLEEP THROUGH INTERRUPT**



**Note 1:** XT, HS or LP oscillator mode assumed.
**2:** T_OST = 1024T_OSC (drawing not to scale) This delay will not be there for RC osc mode.
**3:** GIE = '1' assumed. In this case after wake- up, the processor jumps to the interrupt routine. If GIE = '0', execution will continue in-line.
**4:** CLKOUT is not available in these osc modes, but shown here for timing reference.
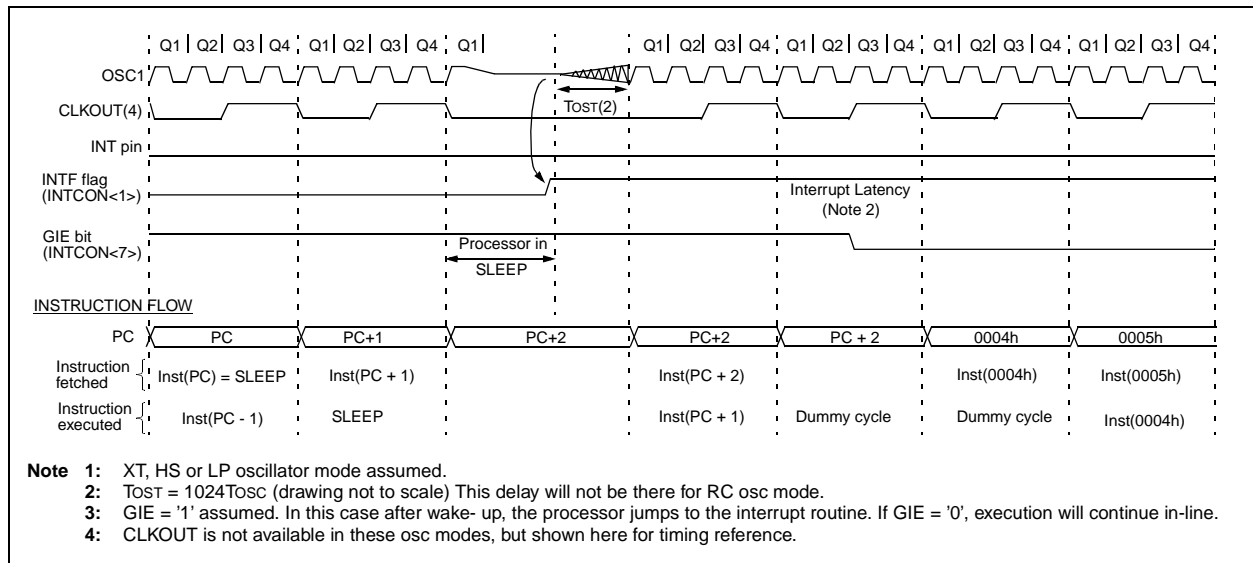
## 9.14    Program Verification/Code Protection

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

> **Note:** Microchip does not recommend code protecting windowed devices.

## 9.15    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. It is recommended that only the 4 least significant bits of the ID location are used.

For ROM devices, these values are submitted along with the ROM code.

## 9.16    In-Circuit Serial Programming™

PIC16CXXX microcontrollers can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data, and three other lines for power, ground and the programming voltage. This allows customers to manufacture boards with unprogrammed devices, and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

For complete details on serial programming, please refer to the In-Circuit Serial Programming (ICSP™) Guide, (DS30277).

**NOTES:**

**Preliminary**

## 10.0    INSTRUCTION SET SUMMARY

Each PIC16CXXX 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 PIC16CXXX instruction set summary in Table 10-2 lists **byte-oriented**, **bit-oriented**, and **literal and control** operations. Table 10-1 shows the opcode field descriptions.

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

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

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

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

### TABLE 10-1    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 |
| $\overline{TO}$ | Time-out bit |
| $\overline{PD}$ | Power-down bit |
| Z | Zero bit |
| DC | Digit Carry bit |
| C | Carry bit |

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

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

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

execution time is 1 μs. If a conditional test is true or the program counter is changed as a result of an instruction, the instruction execution time is 2 μs.

Table 10-2 lists the instructions recognized by the MPASM assembler.

Figure 10-1 shows the general formats that the instructions can have.

> **Note:** To maintain upward compatibility with future PIC16CXXX products, <u>do not use</u> the OPTION and TRIS instructions.

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

  0xhh

where h signifies a hexadecimal digit.

### FIGURE 10-1:    GENERAL FORMAT FOR INSTRUCTIONS

**Byte-oriented** file register operations

```
 13              8  7  6              0
┌──────────────┬────┬────────────────┐
│   OPCODE     │ d  │   f (FILE #)    │
└──────────────┴────┴────────────────┘
```

    d = 0 for destination W
    d = 1 for destination f
    f  = 7-bit file register address

**Bit-oriented** file register operations

```
 13           10 9    7  6             0
┌──────────────┬──────┬────────────────┐
│   OPCODE     │b(BIT#)│   f (FILE #)    │
└──────────────┴──────┴────────────────┘
```

    b = 3-bit bit address
    f  = 7-bit file register address

**Literal and control** operations

General

```
 13                  8  7              0
┌──────────────────┬──────────────────┐
│     OPCODE       │    k (literal)    │
└──────────────────┴──────────────────┘
```

    k  = 8-bit immediate value

CALL and GOTO instructions only

```
 13         11 10                     0
┌──────────────┬──────────────────────┐
│   OPCODE     │      k (literal)       │
└──────────────┴──────────────────────┘
```

    k  = 11-bit immediate value

A description of each instruction is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

# PIC16C712/716

## TABLE 10-2    PIC16CXXX INSTRUCTION SET

| Mnemonic, Operands | | Description | Cycles | 14-Bit Opcode MSb                LSb | Status Affected | Notes |
|---|---|---|---|---|---|---|
| **BYTE-ORIENTED FILE REGISTER OPERATIONS** | | | | | | |
| **ADDWF** | **f, d** | Add W and f | 1 | `00  0111 dfff ffff` | C,DC,Z | 1,2 |
| **ANDWF** | **f, d** | AND W with f | 1 | `00  0101 dfff ffff` | Z | 1,2 |
| **CLRF** | **f** | Clear f | 1 | `00  0001 1fff ffff` | Z | 2 |
| **CLRW** | **-** | Clear W | 1 | `00  0001 0000 0011` | Z | |
| **COMF** | **f, d** | Complement f | 1 | `00  1001 dfff ffff` | Z | 1,2 |
| **DECF** | **f, d** | Decrement f | 1 | `00  0011 dfff ffff` | Z | 1,2 |
| **DECFSZ** | **f, d** | Decrement f, Skip if 0 | 1(2) | `00  1011 dfff ffff` | | 1,2,3 |
| **INCF** | **f, d** | Increment f | 1 | `00  1010 dfff ffff` | Z | 1,2 |
| **INCFSZ** | **f, d** | Increment f, Skip if 0 | 1(2) | `00  1111 dfff ffff` | | 1,2,3 |
| **IORWF** | **f, d** | Inclusive OR W with f | 1 | `00  0100 dfff ffff` | Z | 1,2 |
| **MOVF** | **f, d** | Move f | 1 | `00  1000 dfff ffff` | Z | 1,2 |
| **MOVWF** | **f** | Move W to f | 1 | `00  0000 1fff ffff` | | |
| **NOP** | **-** | No Operation | 1 | `00  0000 0xx0 0000` | | |
| **RLF** | **f, d** | Rotate Left f through Carry | 1 | `00  1101 dfff ffff` | C | 1,2 |
| **RRF** | **f, d** | Rotate Right f through Carry | 1 | `00  1100 dfff ffff` | C | 1,2 |
| **SUBWF** | **f, d** | Subtract W from f | 1 | `00  0010 dfff ffff` | C,DC,Z | 1,2 |
| **SWAPF** | **f, d** | Swap nibbles in f | 1 | `00  1110 dfff ffff` | | 1,2 |
| **XORWF** | **f, d** | Exclusive OR W with f | 1 | `00  0110 dfff ffff` | Z | 1,2 |
| **BIT-ORIENTED FILE REGISTER OPERATIONS** | | | | | | |
| **BCF** | **f, b** | Bit Clear f | 1 | `01  00bb bfff ffff` | | 1,2 |
| **BSF** | **f, b** | Bit Set f | 1 | `01  01bb bfff ffff` | | 1,2 |
| **BTFSC** | **f, b** | Bit Test f, Skip if Clear | 1 (2) | `01  10bb bfff ffff` | | 3 |
| **BTFSS** | **f, b** | Bit Test f, Skip if Set | 1 (2) | `01  11bb bfff ffff` | | 3 |
| **LITERAL AND CONTROL OPERATIONS** | | | | | | |
| **ADDLW** | **k** | Add literal and W | 1 | `11  111x kkkk kkkk` | C,DC,Z | |
| **ANDLW** | **k** | AND literal with W | 1 | `11  1001 kkkk kkkk` | Z | |
| **CALL** | **k** | Call subroutine | 2 | `10  0kkk kkkk kkkk` | | |
| **CLRWDT** | **-** | Clear Watchdog Timer | 1 | `00  0000 0110 0100` | $\overline{TO},\overline{PD}$ | |
| **GOTO** | **k** | Go to address | 2 | `10  1kkk kkkk kkkk` | | |
| **IORLW** | **k** | Inclusive OR literal with W | 1 | `11  1000 kkkk kkkk` | Z | |
| **MOVLW** | **k** | Move literal to W | 1 | `11  00xx kkkk kkkk` | | |
| **RETFIE** | **-** | Return from interrupt | 2 | `00  0000 0000 1001` | | |
| **RETLW** | **k** | Return with literal in W | 2 | `11  01xx kkkk kkkk` | | |
| **RETURN** | **-** | Return from Subroutine | 2 | `00  0000 0000 1000` | | |
| **SLEEP** | **-** | Go into standby mode | 1 | `00  0000 0110 0011` | $\overline{TO},\overline{PD}$ | |
| **SUBLW** | **k** | Subtract W from literal | 1 | `11  110x kkkk kkkk` | C,DC,Z | |
| **XORLW** | **k** | Exclusive OR literal with W | 1 | `11  1010 kkkk kkkk` | Z | |

**Note 1:** 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'.

**2:** If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 Module.

**3:** If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

**Preliminary**

# 11.0 DEVELOPMENT SUPPORT

## 11.1 Development Tools

The PICmicro® microcontrollers are supported with a full range of hardware and software development tools:

• MPLAB™ -ICE Real-Time In-Circuit Emulator
• ICEPIC™ Low-Cost PIC16C5X and PIC16CXXX In-Circuit Emulator
• PRO MATE® II Universal Programmer
• PICSTART® Plus Entry-Level Prototype Programmer
• SIMICE
• PICDEM-1 Low-Cost Demonstration Board
• PICDEM-2 Low-Cost Demonstration Board
• PICDEM-3 Low-Cost Demonstration Board
• MPASM Assembler
• MPLAB™ SIM Software Simulator
• MPLAB-C17 (C Compiler)
• Fuzzy Logic Development System (*fuzzy*TECH®–MP)
• KEELOQ® Evaluation Kits and Programmer

## 11.2 MPLAB-ICE: High Performance Universal In-Circuit Emulator with MPLAB IDE

The MPLAB-ICE Universal In-Circuit Emulator is intended to provide the product development engineer with a complete microcontroller design tool set for PICmicro microcontrollers (MCUs). MPLAB-ICE is supplied with the MPLAB Integrated Development Environment (IDE), which allows editing, "make" and download, and source debugging from a single environment.

Interchangeable processor modules allow the system to be easily reconfigured for emulation of different processors. The universal architecture of the MPLAB-ICE allows expansion to support all new Microchip microcontrollers.

The MPLAB-ICE Emulator System has been designed as a real-time emulation system with advanced features that are generally found on more expensive development tools. The PC compatible 386 (and higher) machine platform and Microsoft Windows® 3.x or Windows 95 environment were chosen to best make these features available to you, the end user.

MPLAB-ICE is available in two versions. MPLAB-ICE 1000 is a basic, low-cost emulator system with simple trace capabilities. It shares processor modules with the MPLAB-ICE 2000. This is a full-featured emulator system with enhanced trace, trigger, and data monitoring features. Both systems will operate across the entire operating speed range of the PICmicro MCU.

## 11.3 ICEPIC: Low-Cost PICmicro In-Circuit Emulator

ICEPIC is a low-cost in-circuit emulator solution for the Microchip PIC12CXXX, PIC16C5X and PIC16CXXX families of 8-bit OTP microcontrollers.

ICEPIC is designed to operate on PC-compatible machines ranging from 386 through Pentium™ based machines under Windows 3.x, Windows 95, or Windows NT environment. ICEPIC features real time, non-intrusive emulation.

## 11.4 PRO MATE II: Universal Programmer

The PRO MATE II Universal Programmer is a full-featured programmer capable of operating in stand-alone mode as well as PC-hosted mode. PRO MATE II is CE compliant.

The PRO MATE II has programmable VDD and VPP supplies which allows it to verify programmed memory at VDD min and VDD max for maximum reliability. It has an LCD display for displaying error messages, keys to enter commands and a modular detachable socket assembly to support various package types. In stand-alone mode the PRO MATE II can read, verify or program PIC12CXXX, PIC14C000, PIC16C5X, PIC16CXXX and PIC17CXX devices. It can also set configuration and code-protect bits in this mode.

## 11.5 PICSTART Plus Entry Level Development System

The PICSTART programmer is an easy-to-use, low-cost prototype programmer. It connects to the PC via one of the COM (RS-232) ports. MPLAB Integrated Development Environment software makes using the programmer simple and efficient. PICSTART Plus is not recommended for production programming.

PICSTART Plus supports all PIC12CXXX, PIC14C000, PIC16C5X, PIC16CXXX and PIC17CXX devices with up to 40 pins. Larger pin count devices such as the PIC16C923, PIC16C924 and PIC17C756 may be supported with an adapter socket. PICSTART Plus is CE compliant.

## 11.6    SIMICE Entry-Level Hardware Simulator

SIMICE is an entry-level hardware development system designed to operate in a PC-based environment with Microchip's simulator MPLAB™-SIM. Both SIMICE and MPLAB-SIM run under Microchip Technology's MPLAB Integrated Development Environment (IDE) software. Specifically, SIMICE provides hardware simulation for Microchip's PIC12C5XX, PIC12CE5XX, and PIC16C5X families of PICmicro 8-bit microcontrollers. SIMICE works in conjunction with MPLAB-SIM to provide non-real-time I/O port emulation. SIMICE enables a developer to run simulator code for driving the target system. In addition, the target system can provide input to the simulator code. This capability allows for simple and interactive debugging without having to manually generate MPLAB-SIM stimulus files. SIMICE is a valuable debugging tool for entry-level system development.

## 11.7    PICDEM-1 Low-Cost PICmicro Demonstration Board

The PICDEM-1 is a simple board which demonstrates the capabilities of several of Microchip's microcontrollers. The microcontrollers supported are: PIC16C5X (PIC16C54 to PIC16C58A), PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 and PIC17C44. All necessary hardware and software is included to run basic demo programs. The users can program the sample microcontrollers provided with the PICDEM-1 board, on a PRO MATE II or PICSTART-Plus programmer, and easily test firmware. The user can also connect the PICDEM-1 board to the MPLAB-ICE emulator and download the firmware to the emulator for testing. Additional prototype area is available for the user to build some additional hardware and connect it to the microcontroller socket(s). Some of the features include an RS-232 interface, a potentiometer for simulated analog input, push-button switches and eight LEDs connected to PORTB.

## 11.8    PICDEM-2 Low-Cost PIC16CXX Demonstration Board

The PICDEM-2 is a simple demonstration board that supports the PIC16C62, PIC16C64, PIC16C65, PIC16C73 and PIC16C74 microcontrollers. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM-2 board, on a PRO MATE II programmer or PICSTART-Plus, and easily test firmware. The MPLAB-ICE emulator may also be used with the PICDEM-2 board to test firmware. Additional prototype area has been provided to the user for adding additional hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push-button switches, a potentiometer for simulated analog input, a Serial EEPROM to demonstrate usage of the $I^2C$ bus and separate headers for connection to an LCD module and a keypad.

## 11.9    PICDEM-3 Low-Cost PIC16CXXX Demonstration Board

The PICDEM-3 is a simple demonstration board that supports the PIC16C923 and PIC16C924 in the PLCC package. It will also support future 44-pin PLCC microcontrollers with a LCD Module. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM-3 board, on a PRO MATE II programmer or PICSTART Plus with an adapter socket, and easily test firmware. The MPLAB-ICE emulator may also be used with the PICDEM-3 board to test firmware. Additional prototype area has been provided to the user for adding hardware and connecting it to the microcontroller socket(s). Some of the features include an RS-232 interface, push-button switches, a potentiometer for simulated analog input, a thermistor and separate headers for connection to an external LCD module and a keypad. Also provided on the PICDEM-3 board is an LCD panel, with 4 commons and 12 segments, that is capable of displaying time, temperature and day of the week. The PICDEM-3 provides an additional RS-232 interface and Windows 3.1 software for showing the demultiplexed LCD signals on a PC. A simple serial interface allows the user to construct a hardware demultiplexer for the LCD signals.

## 11.10 MPLAB Integrated Development Environment Software

The MPLAB IDE Software brings an ease of software development previously unseen in the 8-bit microcontroller market. MPLAB is a windows based application which contains:

- A full featured editor
- Three operating modes
  - editor
  - emulator
  - simulator
- A project manager
- Customizable tool bar and key mapping
- A status bar with project information
- Extensive on-line help

MPLAB allows you to:

- Edit your source files (either assembly or 'C')
- One touch assemble (or compile) and download to PICmicro tools (automatically updates all project information)
- Debug using:
  - source files
  - absolute listing file

The ability to use MPLAB with Microchip's simulator allows a consistent platform and the ability to easily switch from the low cost simulator to the full featured emulator with minimal retraining due to development tools.

## 11.11 Assembler (MPASM)

The MPASM Universal Macro Assembler is a PC-hosted symbolic assembler. It supports all microcontroller series including the PIC12C5XX, PIC14000, PIC16C5X, PIC16CXXX, and PIC17CXX families.

MPASM offers full featured Macro capabilities, conditional assembly, and several source and listing formats. It generates various object code formats to support Microchip's development tools as well as third party programmers.

MPASM allows full symbolic debugging from MPLAB-ICE, Microchip's Universal Emulator System.

MPASM has the following features to assist in developing software for specific use applications.

- Provides translation of Assembler source code to object code for all Microchip microcontrollers.
- Macro assembly capability.
- Produces all the files (Object, Listing, Symbol, and special) required for symbolic debug with Microchip's emulator systems.
- Supports Hex (default), Decimal and Octal source and listing formats.

MPASM provides a rich directive language to support programming of the PICmicro. Directives are helpful in making the development of your assemble source code shorter and more maintainable.

## 11.12 Software Simulator (MPLAB-SIM)

The MPLAB-SIM Software Simulator allows code development in a PC host environment. It allows the user to simulate the PICmicro series microcontrollers on an instruction level. On any given instruction, the user may examine or modify any of the data areas or provide external stimulus to any of the pins. The input/output radix can be set by the user and the execution can be performed in; single step, execute until break, or in a trace mode.

MPLAB-SIM fully supports symbolic debugging using MPLAB-C17 and MPASM. The Software Simulator offers the low cost flexibility to develop and debug code outside of the laboratory environment making it an excellent multi-project software development tool.

## 11.13 MPLAB-C17 Compiler

The MPLAB-C17 Code Development System is a complete ANSI 'C' compiler and integrated development environment for Microchip's PIC17CXXX family of microcontrollers. The compiler provides powerful integration capabilities and ease of use not found with other compilers.

For easier source level debugging, the compiler provides symbol information that is compatible with the MPLAB IDE memory display.

## 11.14 Fuzzy Logic Development System (fuzzyTECH-MP)

fuzzyTECH-MP fuzzy logic development tool is available in two versions - a low cost introductory version, MP Explorer, for designers to gain a comprehensive working knowledge of fuzzy logic system design; and a full-featured version, fuzzyTECH-MP, Edition for implementing more complex systems.

Both versions include Microchip's fuzzyLAB™ demonstration board for hands-on experience with fuzzy logic systems implementation.

## 11.15 SEEVAL® Evaluation and Programming System

The SEEVAL SEEPROM Designer's Kit supports all Microchip 2-wire and 3-wire Serial EEPROMs. The kit includes everything necessary to read, write, erase or program special features of any Microchip SEEPROM product including Smart Serials™ and secure serials. The Total Endurance™ Disk is included to aid in trade-off analysis and reliability calculations. The total kit can significantly reduce time-to-market and result in an optimized system.

## 11.16 KEELOQ® Evaluation and Programming Tools

KEELOQ evaluation and programming tools support Microchips HCS Secure Data Products. The HCS evaluation kit includes an LCD display to show changing codes, a decoder to decode transmissions, and a programming interface to program test transmitters.

**Preliminary** © 1999 Microchip Technology Inc.

**TABLE 11-1    DEVELOPMENT TOOLS FROM MICROCHIP**

| | PIC12C5XX | PIC14000 | PIC16C5X | PIC16CXXX | PIC16C6X | PIC16C7XX | PIC16C8X | PIC16C9XX | PIC17C4X | PIC17C7XX | 24CXX 25CXX 93CXX | HCS200 HCS300 HCS301 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Emulator Products** — MPLAB™-ICE | ✓ | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | |
| ICEPIC™ Low-Cost In-Circuit Emulator | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | |
| **Software Tools** — MPLAB™ Integrated Development Environment | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | |
| MPLAB™ C17* Compiler | | | | | | | | | ✓ | ✓ | | |
| *fuzzy*TECH®-MP Explorer/Edition Fuzzy Logic Dev. Tool | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | |
| Total Endurance™ Software Model | | | | | | | | | | | ✓ | |
| **Programmers** — PICSTART® Plus Low-Cost Universal Dev. Kit | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | |
| PRO MATE® II Universal Programmer | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | |
| KEELOQ® Programmer | | | | | | | | | | | | ✓ |
| **Demo Boards** — SEEVAL® Designers Kit | | | | | | | | | | | ✓ | |
| SIMICE | ✓ | ✓ | ✓ | | | | | | | | | |
| PICDEM-14A | | ✓ | | | | | | | | | | |
| PICDEM-1 | | | ✓ | ✓ | ✓ | | ✓ | | ✓ | | | |
| PICDEM-2 | | | | | ✓ | ✓ | ✓ | | | | | |
| PICDEM-3 | | | | | | | | ✓ | | | | |
| KEELOQ® Evaluation Kit | | | | | | | | | | | | ✓ |
| KEELOQ Transponder Kit | | | | | | | | | | | | ✓ |

**NOTES:**

## 12.0   ELECTRICAL CHARACTERISTICS

**Absolute Maximum Ratings [†]**

Ambient temperature under bias ................................................................................................................. -55°C to +125°C

Storage temperature ............................................................................................................................ -65°C to +150°C

Voltage on any pin with respect to $V_{SS}$ (except $V_{DD}$, $\overline{MCLR}$, and RA4) ......................................... -0.3V to ($V_{DD}$ + 0.3V)

Voltage on $V_{DD}$ with respect to $V_{SS}$ ............................................................................................... -0.3V to +7.5V

Voltage on $\overline{MCLR}$ with respect to $V_{SS}$ (Note 2) ..................................................................... 0V to +13.25V

Voltage on RA4 with respect to $V_{SS}$ ......................................................................................... 0V to +8.5V

Total power dissipation (Note 1)(PDIP and SOIC) ........................................................................................ 1.0W

Total power dissipation (Note 1)(SSOP) ..................................................................................................... 0.65W

Maximum current out of $V_{SS}$ pin .......................................................................................................... 300 mA

Maximum current into $V_{DD}$ pin .............................................................................................................. 250 mA

Input clamp current, $I_{IK}$ ($V_I$ < 0 or $V_I$ > $V_{DD}$) .......................................................................... ±20 mA

Output clamp current, $I_{OK}$ ($V_O$ < 0 or $V_O$ > $V_{DD}$) ..................................................................... ±20 mA

Maximum output current sunk by any I/O pin ........................................................................................... 25 mA

Maximum output current sourced by any I/O pin ...................................................................................... 25 mA

Maximum current sunk by PORTA and PORTB (combined) ..................................................................... 200 mA

Maximum current sourced by PORTA and PORTB (combined) ............................................................... 200 mA

**Note 1:** Power dissipation is calculated as follows: $P_{dis} = V_{DD} \times \{I_{DD} - \Sigma I_{OH}\} + \Sigma \{(V_{DD}-V_{OH}) \times I_{OH}\} + \Sigma(V_{Ol} \times I_{OL})$

**Note 2:** Voltage spikes below $V_{SS}$ at the $\overline{MCLR}$/$V_{PP}$ pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100Ω should be used when applying a "low" level to the $\overline{MCLR}$/$V_{PP}$ pin rather than pulling this pin directly to $V_{SS}$.

† NOTICE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

**FIGURE 12-1: PIC16C712/716 VOLTAGE-FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C**



**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.
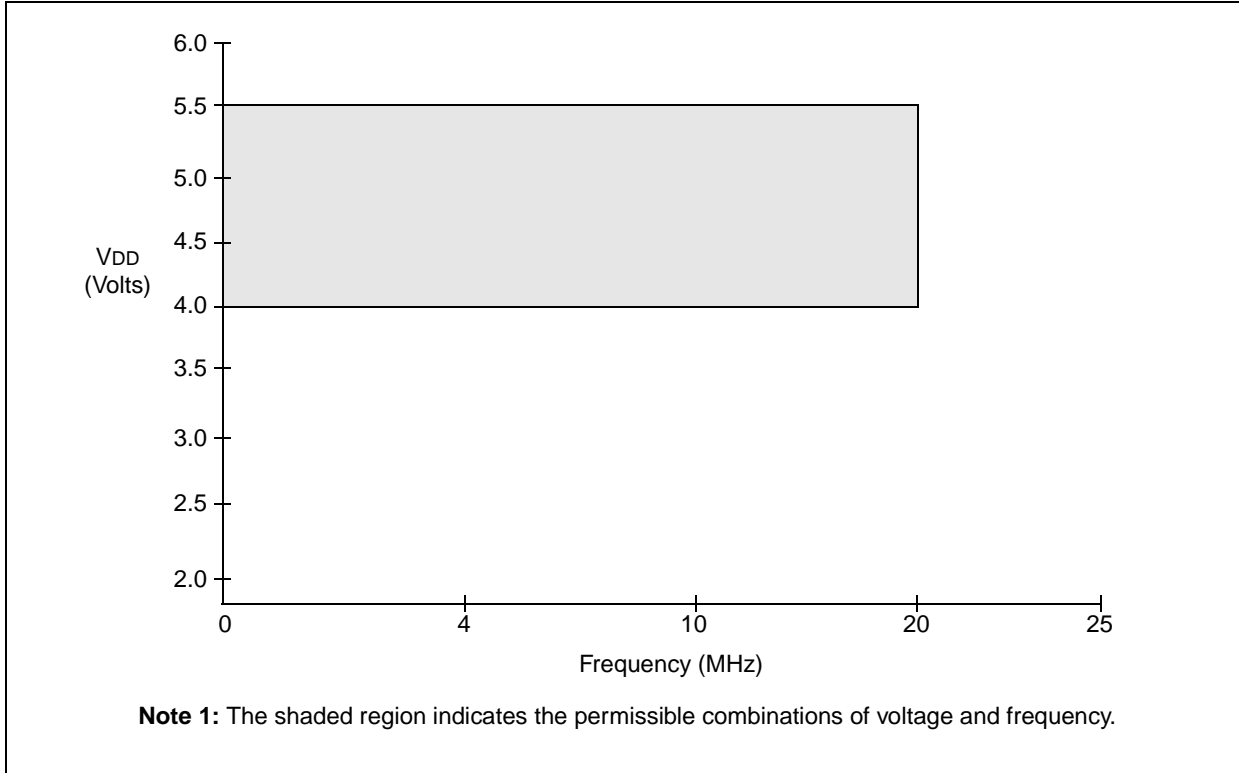
**FIGURE 12-2: PIC16LC712/716 VOLTAGE-FREQUENCY GRAPH, 0°C ≤ TA ≤ +70°C**



**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.
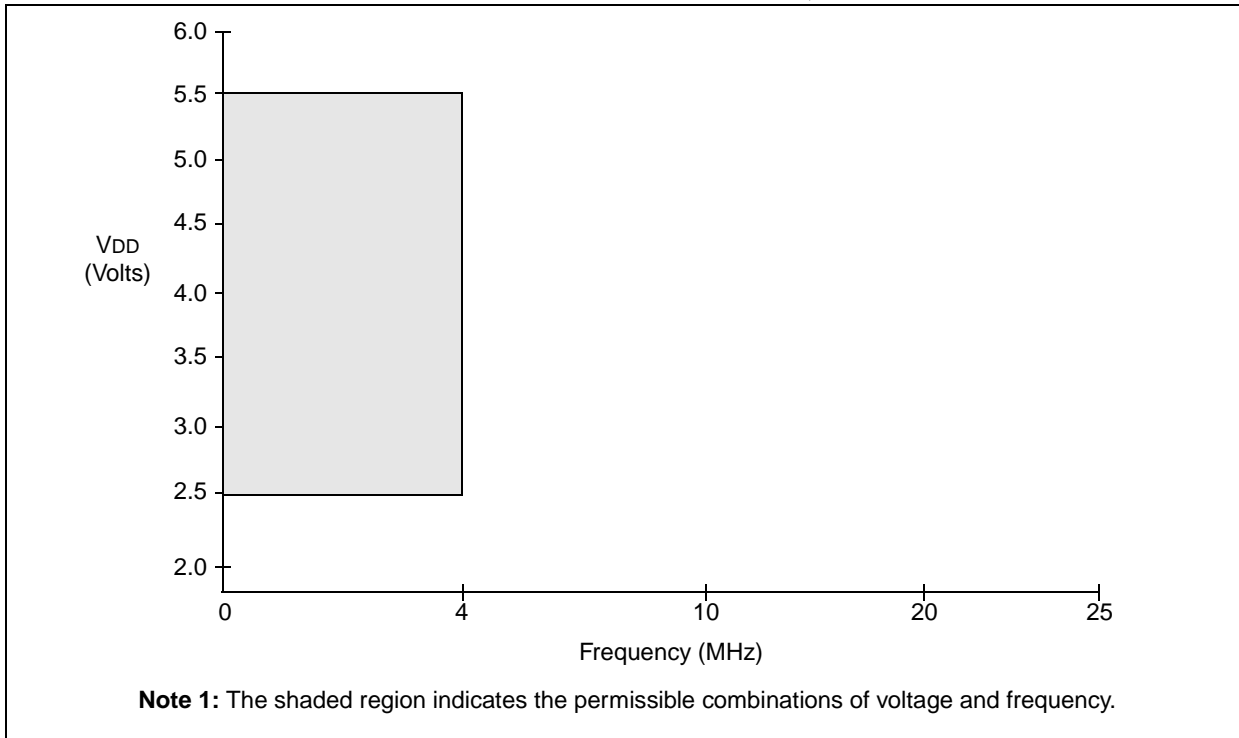
## 12.1 DC Characteristics: PIC16C712/716-04 (Commercial, Industrial, Extended)
PIC16C712/716-20 (Commercial, Industrial, Extended)

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated) | | | | |
|---|---|---|---|---|---|---|---|
| | | | Operating temperature | 0°C $\leq$ TA $\leq$ +70°C for commercial | | | |
| | | | | -40°C $\leq$ TA $\leq$ +85°C for industrial | | | |
| | | | | -40°C $\leq$ TA $\leq$ +125°C for extended | | | |

| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| D001 | VDD | **Supply Voltage** | 4.0 | - | 5.5 | V | |
| D001A | | | VBOR* | - | 5.5 | V | BOR enabled (Note 7) |
| D002* | VDR | **RAM Data Retention Voltage**[1] | - | 1.5 | - | V | |
| D003 | VPOR | **VDD Start Voltage** to ensure internal Power-on Reset signal | - | Vss | - | V | See section on Power-on Reset for details |
| D004*  D004A* | SVDD | **VDD Rise Rate** to ensure internal Power-on Reset signal | 0.05  TBD | -  - | -  - | V/ms | PWRT enabled (PWRTE bit clear)  PWRT disabled (PWRTE bit set)  See section on Power-on Reset for details |
| D005 | VBOR | **Brown-out Reset** voltage trip point | 3.65 | - | 4.35 | V | BODEN bit set |
| D010  D013 | IDD | **Supply Current**[2,5] | -  - | 0.8  4.0 | 2.5  8.0 | mA  mA | FOSC = 4 MHz, VDD = 4.0V  FOSC = 20 MHz, VDD = 4.0V |
| D020     D021  D021B | IPD | **Power-down Current**[3,5] | -  -  -  - | 10.5  1.5  1.5  2.5 | 42  16  19  19 | µA  µA  µA  µA | VDD = 4.0V, WDT enabled,-40°C to +85°C  VDD = 4.0V, WDT disabled, 0°C to +70°C  VDD = 4.0V, WDT disabled,-40°C to +85°C  VDD = 4.0V, WDT disabled,-40°C to +125°C |
| D022*  D022A* | ΔIWDT  ΔIBOR | **Module Differential Current**[6]  Watchdog Timer  Brown-out Reset | -  - | 6.0  TBD | 20  200 | µA  µA | WDTE bit set, VDD = 4.0V  BODEN bit set, VDD = 5.0V |
| 1A | FOSC | LP Oscillator Operating Frequency  RC Oscillator Operating Frequency  XT Oscillator Operating Frequency  HS Oscillator Operating Frequency | 0  0  0  0 | —  —  —  — | 200  4  4  20 | KHz  MHz  MHz  MHz | All temperatures  All temperatures  All temperatures  All temperatures |

\* 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:** This is the limit to which VDD can be lowered without losing RAM data.

**2:** The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.
The test conditions for all IDD measurements in active operation mode are:
OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD,
MCLR = VDD; WDT enabled/disabled as specified.

**3:** The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and VSS.

**4:** For RC osc mode, current through Rext is not included. The current through the resistor can be estimated by the formula Ir = VDD/2Rext (mA) with Rext in kOhm.

**5:** Timer1 oscillator (when enabled) adds approximately 20 µA to the specification. This value is from characterization and is for design guidance only. This is not tested.

**6:** The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.

**7:** This is the voltage where the device enters the Brown-out Reset. When BOR is enabled, the device will operate correctly to this trip point.

## 12.2 DC Characteristics: PIC16LC712/716-04 (Commercial, Industrial)

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated) Operating temperature    0°C ≤ TA ≤ +70°C for commercial    -40°C ≤ TA ≤ +85°C for industrial | | | | |
|---|---|---|---|---|---|---|---|
| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| D001 | VDD | **Supply Voltage** | 2.5 | - | 5.5 | V | |
| | | | VBOR* | - | 5.5 | V | BOR enabled (Note 7) |
| D002* | VDR | **RAM Data Retention Voltage**[1] | - | 1.5 | - | V | |
| D003 | VPOR | **VDD Start Voltage** to ensure internal Power-on Reset signal | - | VSS | - | V | See section on Power-on Reset for details |
| D004* D004A* | SVDD | **VDD Rise Rate** to ensure internal Power-on Reset signal | 0.05 TBD | - - | - - | V/ms | PWRT enabled ($\overline{\text{PWRTE}}$ bit clear) PWRT disabled ($\overline{\text{PWRTE}}$ bit set) See section on Power-on Reset for details |
| D005 | VBOR | **Brown-out Reset** voltage trip point | 3.65 | - | 4.35 | V | BODEN bit set |
| D010 D010A | IDD | **Supply Current**[2,5] | - - | 2.0 22.5 | 3.8 48 | mA µA | XT, RC osc modes FOSC = 4 MHz, VDD = 3.0V (Note 4) LP osc mode FOSC = 32 kHz, VDD = 3.0V, WDT disabled |
| D020 D021 D021A | IPD | **Power-down Current**[3,5] | - - - | 7.5 0.9 0.9 | 30 5 5 | µA µA µA | VDD = 3.0V, WDT enabled, -40°C to +85°C VDD = 3.0V, WDT disabled, 0°C to +70°C VDD = 3.0V, WDT disabled, -40°C to +85°C |
| D022* D022A* | ΔIWDT ΔIBOR | **Module Differential Current**[6] Watchdog Timer Brown-out Reset | - - | 6.0 TBD | 20 200 | µA µA | WDTE bit set, VDD = 4.0V BODEN bit set, VDD = 5.0V |
| 1A | FOSC | LP Oscillator Operating Frequency RC Oscillator Operating Frequency XT Oscillator Operating Frequency HS Oscillator Operating Frequency | 0 0 0 0 | — — — — | 200 4 4 20 | KHz MHz MHz MHz | All temperatures All temperatures All temperatures All temperatures |

\* 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:** This is the limit to which VDD can be lowered without losing RAM data.

**2:** The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.
The test conditions for all IDD measurements in active operation mode are:
OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD,
$\overline{\text{MCLR}}$ = VDD; WDT enabled/disabled as specified.

**3:** The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and VSS.

**4:** For RC osc mode, current through Rext is not included. The current through the resistor can be estimated by the formula Ir = VDD/2Rext (mA) with Rext in kOhm.

**5:** Timer1 oscillator (when enabled) adds approximately 20 µA to the specification. This value is from characterization and is for design guidance only. This is not tested.

**6:** The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.

**7:** This is the voltage where the device enters the Brown-out Reset. When BOR is enabled, the device will operate correctly to this trip point.

## 12.3 DC Characteristics: PIC16C712/716-04 (Commercial, Industrial, Extended)
PIC16C712716-20 (Commercial, Industrial, Extended)
PIC16LC712/716-04 (Commercial, Industrial)

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature $\quad 0°C \leq T_A \leq +70°C$ for commercial<br>$\quad -40°C \leq T_A \leq +85°C$ for industrial<br>$\quad -40°C \leq T_A \leq +125°C$ for extended<br>Operating voltage $V_{DD}$ range as described in DC spec Section 12.1 and Section 12.2 | | | | | |
|---|---|---|---|---|---|---|---|
| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| | $V_{IL}$ | **Input Low Voltage**<br>I/O ports | | | | | |
| D030 | | with TTL buffer | Vss | - | 0.8V | V | $4.5V \leq V_{DD} \leq 5.5V$ |
| D030A | | | Vss | - | 0.15VDD | V | otherwise |
| D031 | | with Schmitt Trigger buffer | Vss | - | 0.2VDD | V | |
| D032 | | $\overline{MCLR}$, OSC1 (in RC mode) | Vss | - | 0.2VDD | V | |
| D033 | | OSC1 (in XT, HS and LP modes) | Vss | - | 0.3VDD | V | Note1 |
| | $V_{IH}$ | **Input High Voltage**<br>I/O ports | | - | | | |
| D040 | | with TTL buffer | 2.0 | - | VDD | V | $4.5V \leq V_{DD} \leq 5.5V$ |
| D040A | | | 0.25VDD + 0.8V | - | VDD | V | otherwise |
| D041 | | with Schmitt Trigger buffer | 0.8VDD | - | VDD | V | For entire VDD range |
| D042 | | $\overline{MCLR}$ | 0.8VDD | - | VDD | V | |
| D042A | | OSC1 (XT, HS and LP modes) | 0.7VDD | - | VDD | V | Note1 |
| D043 | | OSC1 (in RC mode) | 0.9VDD | - | VDD | V | |
| | $I_{IL}$ | **Input Leakage Current** (Notes 2, 3) | | | | | |
| D060 | | I/O ports | - | - | ±1 | µA | $Vss \leq V_{PIN} \leq V_{DD}$, Pin at hi-impedance |
| D061 | | $\overline{MCLR}$, RA4/T0CKI | - | - | ±5 | µA | $Vss \leq V_{PIN} \leq V_{DD}$ |
| D063 | | OSC1 | - | - | ±5 | µA | $Vss \leq V_{PIN} \leq V_{DD}$, XT, HS and LP osc modes |
| D070 | $I_{PURB}$ | **PORTB weak pull-up current** | 50 | 250 | 400 | µA | $V_{DD} = 5V$, $V_{PIN} = Vss$ |
| | $V_{OL}$ | **Output Low Voltage**<br>I/O ports | | | | | |
| D080 | | | - | - | 0.6 | V | $I_{OL} = 8.5$ mA, $V_{DD} = 4.5V$, -40°C to +85°C |
| | | | - | - | 0.6 | V | $I_{OL} = 7.0$ mA, $V_{DD} = 4.5V$, -40°C to +125°C |
| D083 | | OSC2/CLKOUT (RC osc mode) | - | - | 0.6 | V | $I_{OL} = 1.6$ mA, $V_{DD} = 4.5V$, -40°C to +85°C |
| | | | - | - | 0.6 | V | $I_{OL} = 1.2$ mA, $V_{DD} = 4.5V$, -40°C to +125°C |

\* 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:** In RC oscillator mode, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PICmicro be driven with external clock in RC mode.

**2:** The leakage current on the $\overline{MCLR}$/VPP pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

**3:** Negative current is defined as current sourced by the pin.

# PIC16C712/716

| | | Standard Operating Conditions (unless otherwise stated) | | | | | |
|---|---|---|---|---|---|---|---|
| **DC CHARACTERISTICS** | | Operating temperature $\quad 0°C \leq T_A \leq +70°C$ for commercial<br>$\qquad\qquad\qquad\qquad -40°C \leq T_A \leq +85°C$ for industrial<br>$\qquad\qquad\qquad\qquad -40°C \leq T_A \leq +125°C$ for extended<br>Operating voltage $V_{DD}$ range as described in DC spec Section 12.1 and Section 12.2 | | | | | |

| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| D090 | $V_{OH}$ | **Output High Voltage**<br>I/O ports (Note 3) | $V_{DD}$-0.7 | - | - | V | $I_{OH}$ = -3.0 mA, $V_{DD}$ = 4.5V, -40°C to +85°C |
| | | | $V_{DD}$-0.7 | - | - | V | $I_{OH}$ = -2.5 mA, $V_{DD}$ = 4.5V, -40°C to +125°C |
| D092 | | OSC2/CLKOUT (RC osc mode) | $V_{DD}$-0.7 | - | - | V | $I_{OH}$ = -1.3 mA, $V_{DD}$ = 4.5V, -40°C to +85°C |
| | | | $V_{DD}$-0.7 | - | - | V | $I_{OH}$ = -1.0 mA, $V_{DD}$ = 4.5V, -40°C to +125°C |
| D150* | $V_{OD}$ | **Open-Drain High Voltage** | - | - | 8.5 | V | RA4 pin |
| D100 | $C_{OSC2}$ | **Capacitive Loading Specs on Output Pins**<br>OSC2 pin | - | - | 15 | pF | In XT, HS and LP modes when external clock is used to drive OSC1. |
| D101 | $C_{IO}$ | All I/O pins and OSC2 (in RC mode) | - | - | 50 | pF | |

\* 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:** In RC oscillator mode, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PICmicro be driven with external clock in RC mode.

**2:** The leakage current on the $\overline{MCLR}$/$V_{PP}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

**3:** Negative current is defined as current sourced by the pin.

**Preliminary**

## 12.4 AC (Timing) Characteristics

12.4.1 TIMING PARAMETER SYMBOLOGY

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

1. TppS2ppS
2. TppS

| T | | | |
|---|---|---|---|
| F | Frequency | T | Time |

Lowercase letters (pp) and their meanings:

| pp | | | |
|---|---|---|---|
| cc | CCP1 | osc | OSC1 |
| ck | CLKOUT | rd | $\overline{RD}$ |
| cs | $\overline{CS}$ | rw | $\overline{RD}$ or $\overline{WR}$ |
| di | SDI | sc | SCK |
| do | SDO | ss | $\overline{SS}$ |
| dt | Data in | t0 | T0CKI |
| io | I/O port | t1 | T1CKI |
| mc | $\overline{MCLR}$ | wr | $\overline{WR}$ |

Uppercase letters and their meanings:

| S | | | |
|---|---|---|---|
| F | Fall | P | Period |
| H | High | R | Rise |
| I | Invalid (Hi-impedance) | V | Valid |
| L | Low | Z | Hi-impedance |

## 12.4.2   TIMING CONDITIONS

The temperature and voltages specified in Table 12-1 apply to all timing specifications, unless otherwise noted. Figure 12-1 specifies the load conditions for the timing specifications.

**TABLE 12-1     TEMPERATURE AND VOLTAGE SPECIFICATIONS - AC**

| **AC CHARACTERISTICS** | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature     0°C   $\leq T_A \leq$   +70°C   for commercial<br>                            -40°C   $\leq T_A \leq$   +85°C   for industrial<br>                            -40°C   $\leq T_A \leq$   +125°C   for extended<br>Operating voltage $V_{DD}$ range as described in DC spec Section 12.1 and Section 12.2.<br>LC parts operate for commercial/industrial temp's only. |
| --- | --- |

**FIGURE 12-1:   LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS**
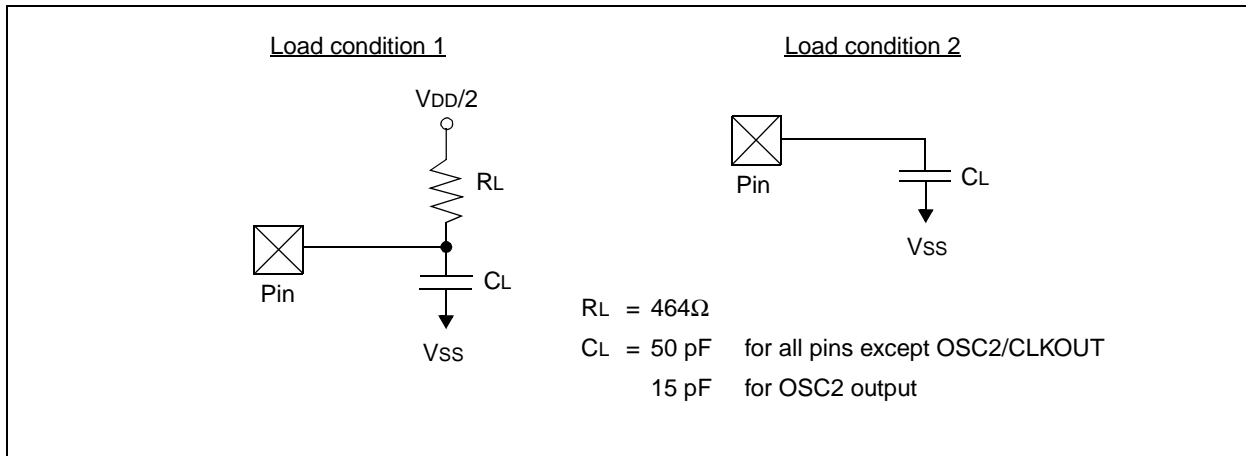
## 12.4.3    TIMING DIAGRAMS AND SPECIFICATIONS

**FIGURE 12-2:    EXTERNAL CLOCK TIMING**
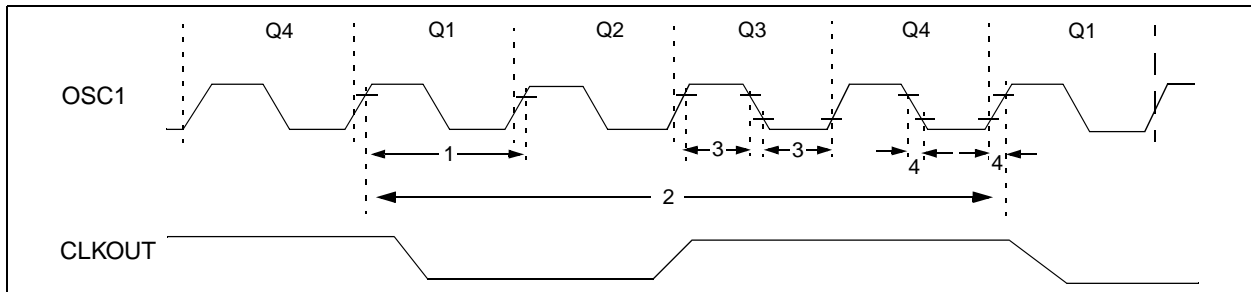


**TABLE 12-2    EXTERNAL CLOCK TIMING REQUIREMENTS**

| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| 1A | FOSC | **External CLKIN Frequency** (Note 1) | DC | — | 4 | MHz | RC and XT osc modes |
| | | | DC | — | 4 | MHz | HS osc mode (-04) |
| | | | DC | — | 20 | MHz | HS osc mode (-20) |
| | | | DC | — | 200 | kHz | LP osc mode |
| | | **Oscillator Frequency** (Note 1) | DC | — | 4 | MHz | RC osc mode |
| | | | 0.1 | — | 4 | MHz | XT osc mode |
| | | | 4 | — | 20 | MHz | HS osc mode |
| | | | 5 | — | 200 | kHz | LP osc mode |
| 1 | TOSC | **External CLKIN Period** (Note 1) | 250 | — | — | ns | RC and XT osc modes |
| | | | 250 | — | — | ns | HS osc mode (-04) |
| | | | 50 | — | — | ns | HS osc mode (-20) |
| | | | 5 | — | — | µs | LP osc mode |
| | | **Oscillator Period** (Note 1) | 250 | — | — | ns | RC osc mode |
| | | | 250 | — | 10,000 | ns | XT osc mode |
| | | | 250 | — | 250 | ns | HS osc mode (-04) |
| | | | 50 | — | 250 | ns | HS osc mode (-20) |
| | | | 5 | — | — | µs | LP osc mode |
| 2 | TCY | **Instruction Cycle Time** (Note 1) | 200 | — | DC | ns | TCY = 4/FOSC |
| 3* | TosL, TosH | **External Clock in (OSC1) High or Low Time** | 100 | — | — | ns | XT oscillator |
| | | | 2.5 | — | — | µs | LP oscillator |
| | | | 15 | — | — | ns | HS oscillator |
| 4* | TosR, TosF | **External Clock in (OSC1) Rise or Fall Time** | — | — | 25 | ns | XT oscillator |
| | | | — | — | 50 | ns | LP oscillator |
| | | | — | — | 15 | 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 the OSC1/CLKIN pin.

When an external clock input is used, the "Max." cycle time limit is "DC" (no clock) for all devices.

# PIC16C712/716

**FIGURE 12-3: CLKOUT AND I/O TIMING**



**Note:** Refer to Figure 12-1 for load conditions.
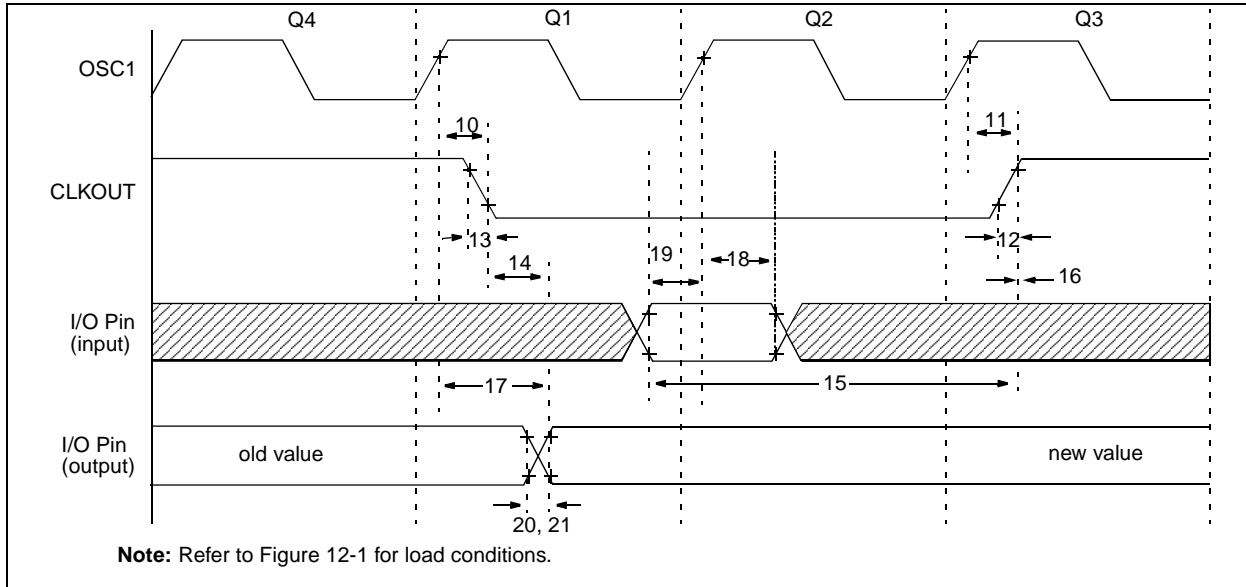
**TABLE 12-3    CLKOUT AND I/O TIMING REQUIREMENTS**

| Param No. | Sym | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 10* | TosH2ckL | OSC1↑ to CLKOUT↓ | | — | 75 | 200 | ns | Note 1 |
| 11* | TosH2ckH | OSC1↑ to CLKOUT↑ | | — | 75 | 200 | ns | Note 1 |
| 12* | TckR | CLKOUT rise time | | — | 35 | 100 | ns | Note 1 |
| 13* | TckF | CLKOUT fall time | | — | 35 | 100 | ns | Note 1 |
| 14* | TckL2ioV | CLKOUT ↓ to Port out valid | | — | — | 0.5TCY + 20 | ns | Note 1 |
| 15* | TioV2ckH | Port in valid before CLKOUT ↑ | | Tosc + 200 | — | — | ns | Note 1 |
| 16* | TckH2ioI | Port in hold after CLKOUT ↑ | | 0 | — | — | ns | Note 1 |
| 17* | TosH2ioV | OSC1↑ (Q1 cycle) to Port out valid | | — | 50 | 150 | ns | |
| 18* | TosH2ioI | OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time) | Standard | 100 | — | — | ns | |
| 18A* | | | Extended (LC) | 200 | — | — | ns | |
| 19* | TioV2osH | Port input valid to OSC1↑ (I/O in setup time) | | 0 | — | — | ns | |
| 20* | TioR | Port output rise time | Standard | — | 10 | 40 | ns | |
| 20A* | | | Extended (LC) | — | — | 80 | ns | |
| 21* | TioF | Port output fall time | Standard | — | 10 | 40 | ns | |
| 21A* | | | Extended (LC) | — | — | 80 | ns | |
| 22††* | TINP | INT pin high or low time | | TCY | — | — | ns | |
| 23††* | TRBP | RB7:RB4 change INT high or low time | | TCY | — | — | ns | |

\*    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.

††   These parameters are asynchronous events not related to any internal clock edge.

**Note1:**   Measurements are taken in RC Mode where CLKOUT output is 4 x TOSC.

**FIGURE 12-4:** RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING



**Note:** Refer to Figure 12-1 for load conditions.
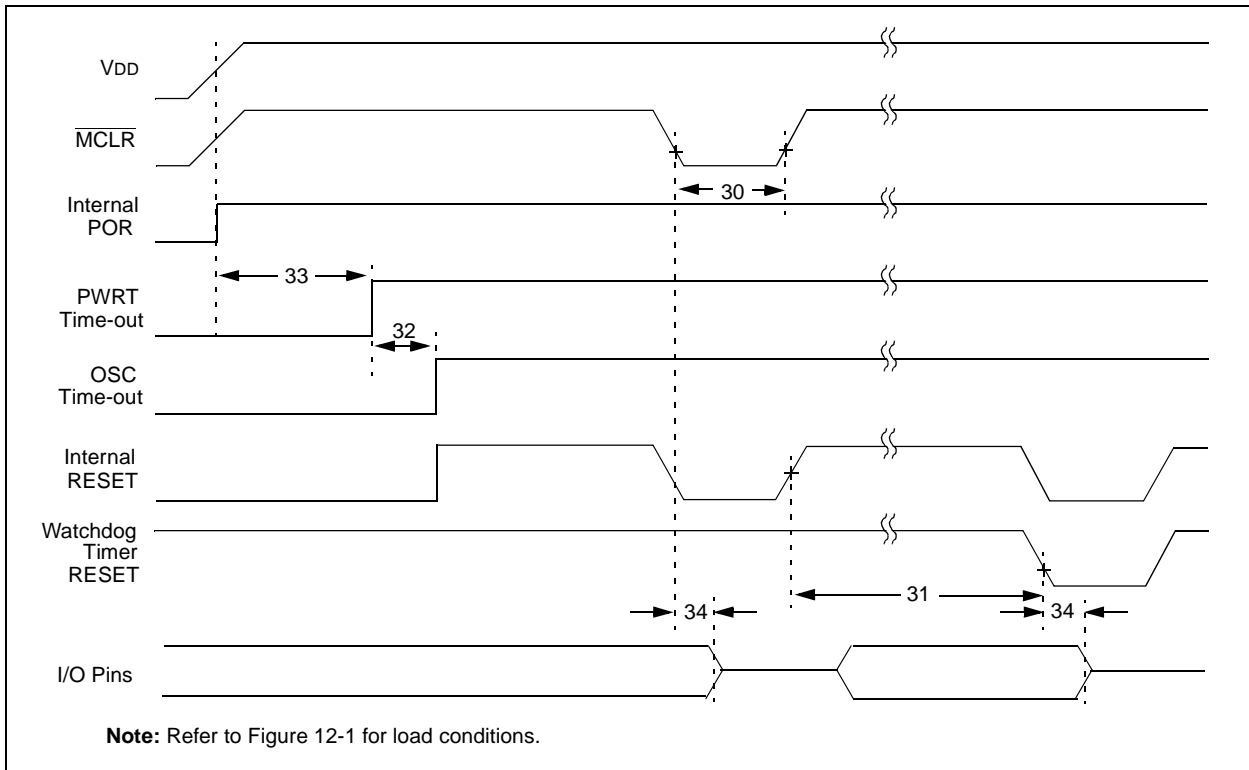
**FIGURE 12-5:** BROWN-OUT RESET TIMING
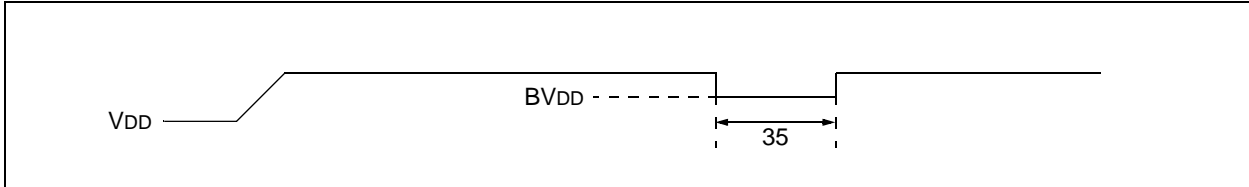


**TABLE 12-4** RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER, AND BROWN-OUT RESET REQUIREMENTS

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| 30 | TmcL | $\overline{\text{MCLR}}$ Pulse Width (low) | 2 | — | — | µs | VDD = 5V, -40°C to +125°C |
| 31* | TWDT | Watchdog Timer Time-out Period (No Prescaler) | 7 | 18 | 33 | ms | VDD = 5V, -40°C to +125°C |
| 32 | TOST | Oscillation Start-up Timer Period | — | 1024 TOSC | — | — | TOSC = OSC1 period |
| 33* | TPWRT | Power-up Timer Period | 28 | 72 | 132 | ms | VDD = 5V, -40°C to +125°C |
| 34 | TIOZ | I/O Hi-impedance from $\overline{\text{MCLR}}$ Low or WDT reset | — | — | 2.1 | µs | |
| 35 | TBOR | Brown-out Reset Pulse Width | 100 | — | — | µs | VDD ≤ BVDD (D005) |

\*   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.

**FIGURE 12-6: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS**



**Note:** Refer to Figure 12-1 for load conditions.
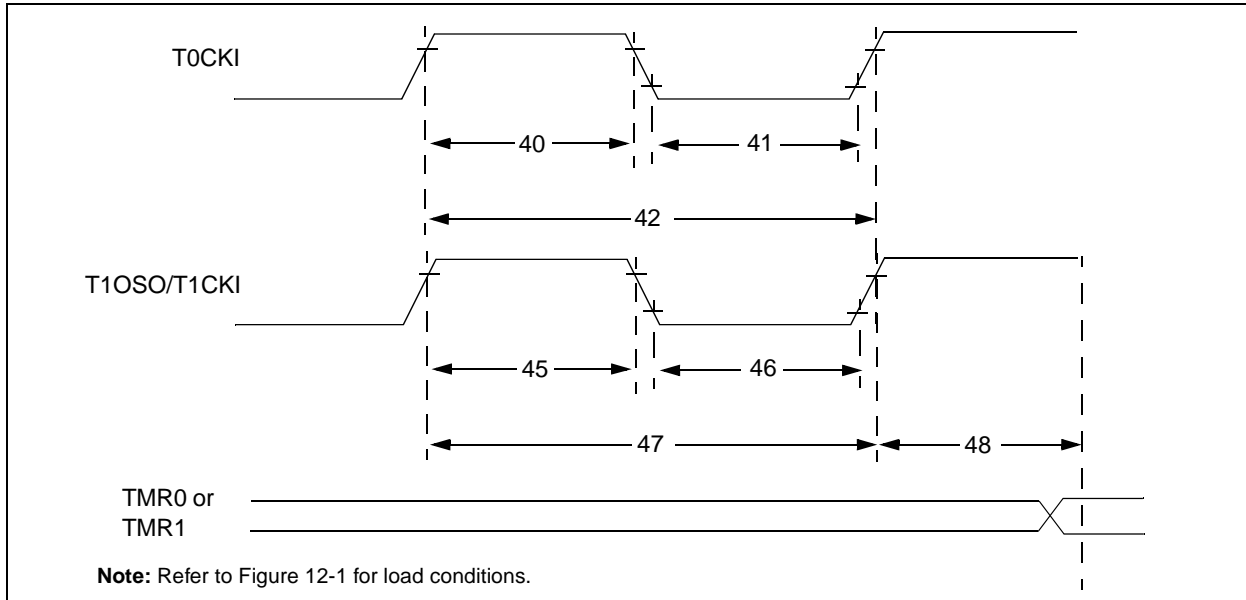
**TABLE 12-5    TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS**

| Param No. | Sym | Characteristic | | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|---|
| 40* | Tt0H | T0CKI High Pulse Width | No Prescaler | | 0.5TCY + 20 | — | — | ns | Must also meet parameter 42 |
| | | | With Prescaler | | 10 | — | — | ns | |
| 41* | Tt0L | T0CKI Low Pulse Width | No Prescaler | | 0.5TCY + 20 | — | — | ns | Must also meet parameter 42 |
| | | | With Prescaler | | 10 | — | — | ns | |
| 42* | Tt0P | T0CKI Period | No Prescaler | | TCY + 40 | — | — | ns | |
| | | | With Prescaler | | Greater of: 20 or $\frac{TCY + 40}{N}$ | — | — | ns | N = prescale value (2, 4,..., 256) |
| 45* | Tt1H | T1CKI High Time | Synchronous, Prescaler = 1 | | 0.5TCY + 20 | — | — | ns | Must also meet parameter 47 |
| | | | Synchronous, Prescaler = 2,4,8 | Standard | 15 | — | — | ns | |
| | | | | Extended (LC) | 25 | — | — | ns | |
| | | | Asynchronous | Standard | 30 | — | — | ns | |
| | | | | Extended (LC) | 50 | — | — | ns | |
| 46* | Tt1L | T1CKI Low Time | Synchronous, Prescaler = 1 | | 0.5TCY + 20 | — | — | ns | Must also meet parameter 47 |
| | | | Synchronous, Prescaler = 2,4,8 | Standard | 15 | — | — | ns | |
| | | | | Extended (LC) | 25 | — | — | ns | |
| | | | Asynchronous | Standard | 30 | — | — | ns | |
| | | | | Extended (LC) | 50 | — | — | ns | |
| 47* | Tt1P | T1CKI input period | Synchronous | Standard | Greater of: 30 OR $\frac{TCY + 40}{N}$ | — | — | ns | N = prescale value (1, 2, 4, 8) |
| | | | | Extended (LC) | Greater of: 50 OR $\frac{TCY + 40}{N}$ | | | | N = prescale value (1, 2, 4, 8) |
| | | | Asynchronous | Standard | 60 | — | — | ns | |
| | | | | Extended (LC) | 100 | — | — | ns | |
| | Ft1 | Timer1 oscillator input frequency range (oscillator enabled by setting bit T1OSCEN) | | | DC | — | 200 | kHz | |
| 48 | TCKEZtmr1 | Delay from external clock edge to timer increment | | | 2Tosc | — | 7Tosc | — | |

\*    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.

**FIGURE 12-7: CAPTURE/COMPARE/PWM TIMINGS**



**Note:** Refer to Figure 12-1 for load conditions.
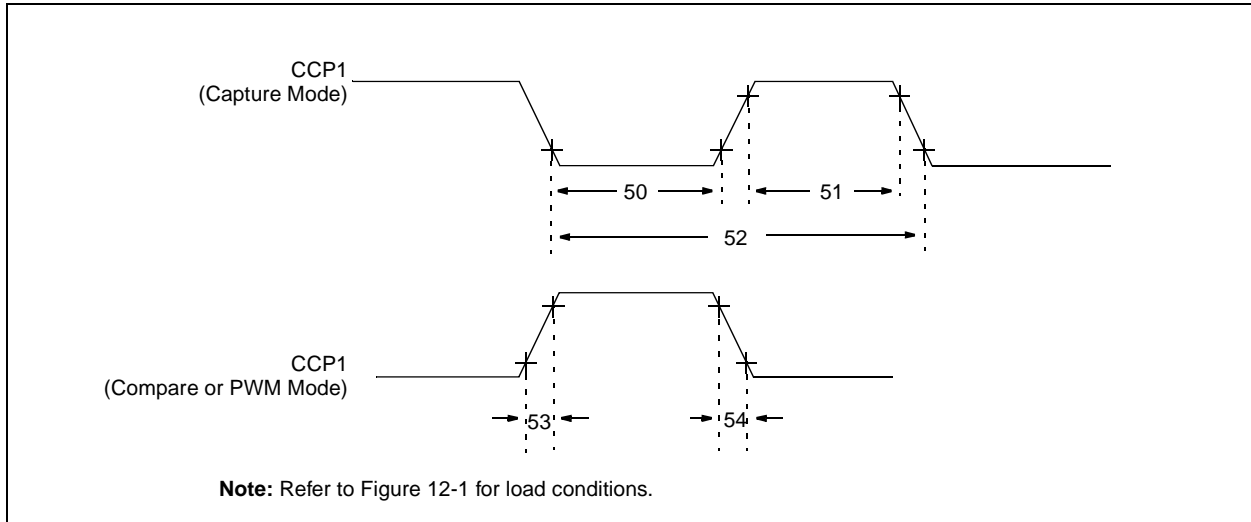
**TABLE 12-6    CAPTURE/COMPARE/PWM REQUIREMENTS**

| Param No. | Sym | Characteristic | | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|---|
| 50* | TccL | CCP1 input low time | No Prescaler | | $0.5T_{CY} + 20$ | — | — | ns | |
| | | | With Prescaler | Standard | 10 | — | — | ns | |
| | | | | Extended (LC) | 20 | — | — | ns | |
| 51* | TccH | CCP1 input high time | No Prescaler | | $0.5T_{CY} + 20$ | — | — | ns | |
| | | | With Prescaler | Standard | 10 | — | — | ns | |
| | | | | Extended (LC) | 20 | — | — | ns | |
| 52* | TccP | CCP1 input period | | | $\dfrac{3T_{CY} + 40}{N}$ | — | — | ns | N = prescale value (1,4, or 16) |
| 53* | TccR | CCP1 output rise time | Standard | | — | 10 | 25 | ns | |
| | | | Extended (LC) | | — | 25 | 45 | ns | |
| 54* | TccF | CCP1 output fall time | Standard | | — | 10 | 25 | ns | |
| | | | Extended (LC) | | — | 25 | 45 | ns | |

\*    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.

**TABLE 12-7    A/D CONVERTER CHARACTERISTICS:**
**PIC16C712/716-04 (COMMERCIAL, INDUSTRIAL, EXTENDED)**
**PIC16C712/716-20 (COMMERCIAL, INDUSTRIAL, EXTENDED)**
**PIC16LC712/716-04 (COMMERCIAL, INDUSTRIAL)**

| Param No. | Sym | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| A01 | N$_R$ | Resolution | | — | — | 8-bits | bit | V$_{REF}$ = V$_{DD}$ = 5.12V, V$_{SS}$ ≤ V$_{AIN}$ ≤ V$_{REF}$ |
| A02 | E$_{ABS}$ | Total Absolute error | | — | — | < ± 1 | LSb | V$_{REF}$ = V$_{DD}$ = 5.12V, V$_{SS}$ ≤ V$_{AIN}$ ≤ V$_{REF}$ |
| A03 | E$_{IL}$ | Integral linearity error | | — | — | < ± 1 | LSb | V$_{REF}$ = V$_{DD}$ = 5.12V, V$_{SS}$ ≤ V$_{AIN}$ ≤ V$_{REF}$ |
| A04 | E$_{DL}$ | Differential linearity error | | — | — | < ± 1 | LSb | V$_{REF}$ = V$_{DD}$ = 5.12V, V$_{SS}$ ≤ V$_{AIN}$ ≤ V$_{REF}$ |
| A05 | E$_{FS}$ | Full scale error | | — | — | < ± 1 | LSb | V$_{REF}$ = V$_{DD}$ = 5.12V, V$_{SS}$ ≤ V$_{AIN}$ ≤ V$_{REF}$ |
| A06 | E$_{OFF}$ | Offset error | | — | — | < ± 1 | LSb | V$_{REF}$ = V$_{DD}$ = 5.12V, V$_{SS}$ ≤ V$_{AIN}$ ≤ V$_{REF}$ |
| A10 | — | Monotonicity | | — | guaranteed (Note 3) | — | — | V$_{SS}$ ≤ V$_{AIN}$ ≤ V$_{REF}$ |
| A20 | V$_{REF}$ | Reference voltage | | 2.5V | — | V$_{DD}$ + 0.3 | V | |
| A25 | V$_{AIN}$ | Analog input voltage | | V$_{SS}$ - 0.3 | — | V$_{REF}$ + 0.3 | V | |
| A30 | Z$_{AIN}$ | Recommended impedance of analog voltage source | | — | — | 10.0 | kΩ | |
| A40 | I$_{AD}$ | A/D conversion current (V$_{DD}$) | Standard | — | 180 | — | μA | Average current consumption when A/D is on. (Note 1) |
| | | | Extended (LC) | — | 90 | — | μA | |
| A50 | I$_{REF}$ | V$_{REF}$ input current (Note 2) | | 10 | — | 1000 | μA | During V$_{AIN}$ acquisition. Based on differential of V$_{HOLD}$ to V$_{AIN}$ to charge C$_{HOLD}$, see Section 9.1. |
| | | | | — | — | 10 | μA | During A/D Conversion cycle |

**2:**    *    These parameters are characterized but not tested.
**3:**    †    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:**    When A/D is off, it will not consume any current other than minor leakage current.
The power-down current spec includes any such leakage from the A/D module.
**2:**    V$_{REF}$ current is from RA3 pin or V$_{DD}$ pin, whichever is selected as reference input.
**3:**    The A/D conversion result never decreases with an increase in the Input Voltage, and has no missing codes.

**FIGURE 12-8: A/D CONVERSION TIMING**



**Note1:** If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed.
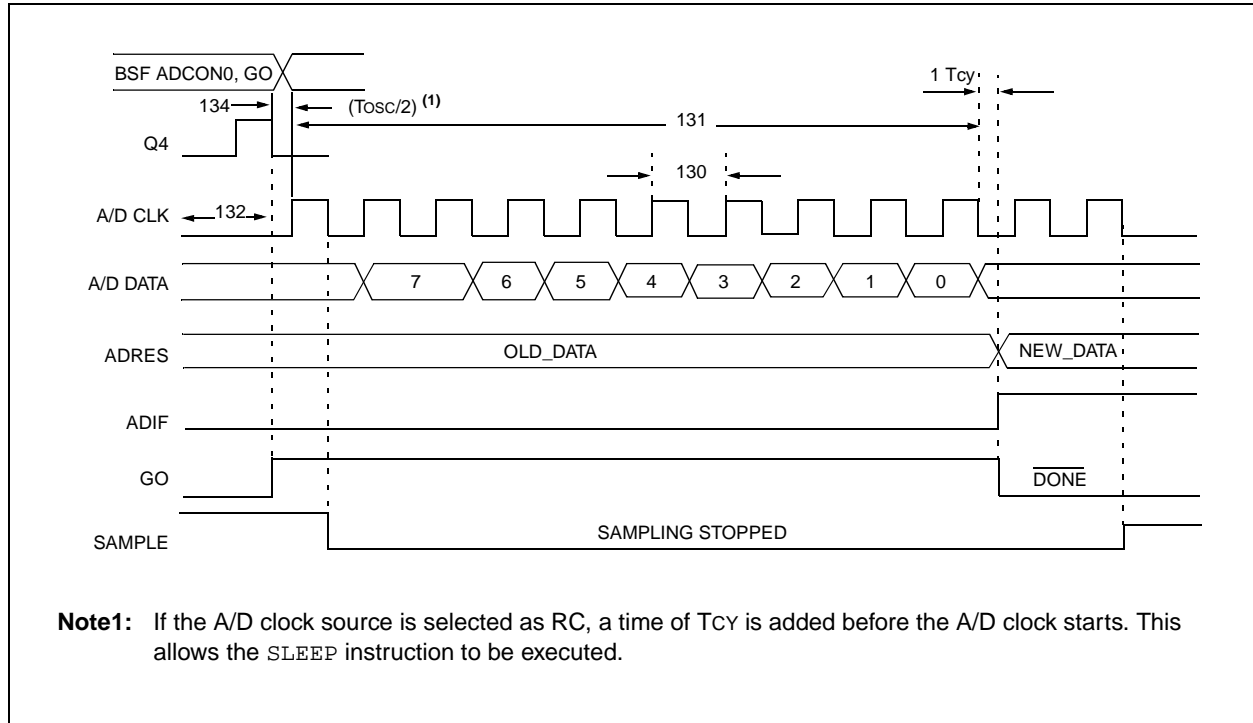
**TABLE 12-8    A/D CONVERSION REQUIREMENTS**

| Param No. | Sym | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 130 | TAD | A/D clock period | Standard | 1.6 | — | — | μs | TOSC based, VREF ≥ 3.0V |
| | | | Extended (LC) | 2.0 | — | — | μs | TOSC based, VREF full range |
| | | | Standard | 2.0 | 4.0 | 6.0 | μs | A/D RC Mode |
| | | | Extended (LC) | 3.0 | 6.0 | 9.0 | μs | A/D RC Mode |
| 131 | TCNV | Conversion time (not including S/H time) (Note 1) | | 11 | — | 11 | TAD | |
| 132 | TACQ | Acquisition time | | Note 2 | 20 | — | μs | |
| | | | | 5* | — | — | μs | The minimum time is the amplifier settling time. This may be used if the "new" input voltage has not changed by more than 1 LSb (i.e., 20.0 mV @ 5.12V) from the last sampled voltage (as stated on CHOLD). |
| 134 | TGO | Q4 to A/D clock start | | — | TOSC/2 § | — | — | If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed. |
| 135 | TSWC | Switching from convert → sample time | | 1.5 § | — | — | TAD | |

**:**  \* 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.

**:**  § This specification ensured by design.

**Note 1:** ADRES register may be read on the following TCY cycle.

**2:** See Section 9.1 for min conditions.

**NOTES:**

**Preliminary**                    © 1999 Microchip Technology Inc.

## 13.0   DC AND AC CHARACTERISTICS GRAPHS AND TABLES

The graphs and tables provided in this section are for **design guidance** and are **not tested**.

In some graphs or tables, the data presented are **outside specified operating range** (i.e., outside specified $V_{DD}$ range). This is for **information only** and devices are guaranteed to operate properly only within the specified range.

The data presented in this section is a **statistical summary** of data collected on units from different lots over a period of time and matrix samples. 'Typical' represents the mean of the distribution at 25°C. 'Max' or 'min' represents (mean + 3$\sigma$) or (mean - 3$\sigma$) respectively, where $\sigma$ is standard deviation, over the whole temperature range.

## Graphs and Tables not available at this time.

**NOTES:**

**Preliminary** © 1999 Microchip Technology Inc.

## 14.0  PACKAGING INFORMATION

### 14.1  Package Marking Information

18-Lead PDIP
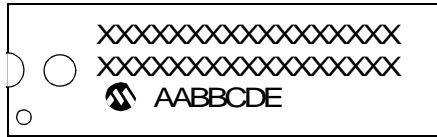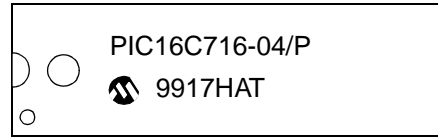
```
XXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXX
      AABBCDE
```

Example

```
PIC16C716-04/P
    9917HAT
```

18-Lead CERDIP Windowed

```
MICROCHIP      XXXXXXXX
               XXXXXXXX
               AABBCDE
```

Example

```
MICROCHIP      16C716
               /JW
               9917CAT
```

18-Lead SOIC

```
XXXXXXXXXXX
XXXXXXXXXXX
XXXXXXXXXXX
    AABBCDE
```

Example

```
PIC16C712
-20/SO
    9910/SAA
```

20-Lead SSOP

```
XXXXXXXXXX
XXXXXXXXXX

   AABBCDE
```

Example

```
PIC16C712
-20I/SS025

   9917SBP
```
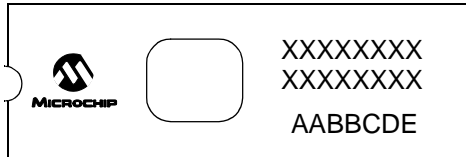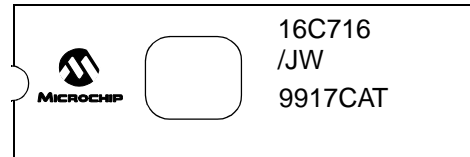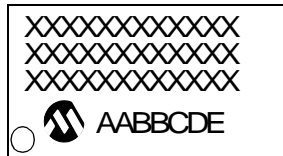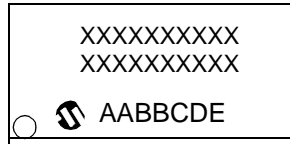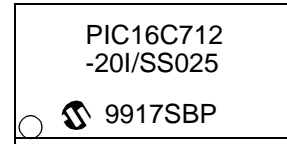
| **Legend:** MM...M | Microchip part number information |
| --- | --- |
| XX...X | Customer specific information* |
| AA | Year code (last 2 digits of calendar year) |
| BB | Week code (week of January 1 is week '01') |
| C | Facility code of the plant at which wafer is manufactured |
| | O = Outside Vendor |
| | C = 5" Line |
| | S = 6" Line |
| | H = 8" Line |
| D | Mask revision number |
| E | Assembly code of the plant or country of origin in which part was assembled |
| **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 OTP marking consists of Microchip part number, year code, week code, facility code, mask rev#, and assembly code.  For  OTP 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.

**Package Type:    K04-007  18-Lead Plastic Dual In-line (P) – 300 mil**



| Units | | INCHES* | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| PCB Row Spacing | | | 0.300 | | | 7.62 | |
| Number of Pins | n | | 18 | | | 18 | |
| Pitch | p | | 0.100 | | | 2.54 | |
| Lower Lead Width | B | 0.013 | 0.018 | 0.023 | 0.33 | 0.46 | 0.58 |
| Upper Lead Width | B1† | 0.055 | 0.060 | 0.065 | 1.40 | 1.52 | 1.65 |
| Shoulder Radius | R | 0.000 | 0.005 | 0.010 | 0.00 | 0.13 | 0.25 |
| Lead Thickness | c | 0.005 | 0.010 | 0.015 | 0.13 | 0.25 | 0.38 |
| Top to Seating Plane | A | 0.110 | 0.155 | 0.155 | 2.79 | 3.94 | 3.94 |
| Top of Lead to Seating Plane | A1 | 0.075 | 0.095 | 0.115 | 1.91 | 2.41 | 2.92 |
| Base to Seating Plane | A2 | 0.000 | 0.020 | 0.020 | 0.00 | 0.51 | 0.51 |
| Tip to Seating Plane | L | 0.125 | 0.130 | 0.135 | 3.18 | 3.30 | 3.43 |
| Package Length | D‡ | 0.890 | 0.895 | 0.900 | 22.61 | 22.73 | 22.86 |
| Molded Package Width | E‡ | 0.245 | 0.255 | 0.265 | 6.22 | 6.48 | 6.73 |
| Radius to Radius Width | E1 | 0.230 | 0.250 | 0.270 | 5.84 | 6.35 | 6.86 |
| Overall Row Spacing | eB | 0.310 | 0.349 | 0.387 | 7.87 | 8.85 | 9.83 |
| Mold Draft Angle Top | α | 5 | 10 | 15 | 5 | 10 | 15 |
| Mold Draft Angle Bottom | β | 5 | 10 | 15 | 5 | 10 | 15 |

\*  Controlling Parameter.

†  Dimension "B1" does not include dam-bar protrusions. Dam-bar protrusions shall not exceed 0.003"
   (0.076 mm) per side or 0.006" (0.152 mm) more than dimension "B1."

‡  Dimensions "D" and "E" do not include mold flash or protrusions. Mold flash or protrusions shall not
   exceed 0.010" (0.254 mm) per side or 0.020" (0.508 mm) more than dimensions "D" or "E."

JEDEC equivalent:  MS-001 AC

**Package Type:    K04-010  18-Lead Ceramic Dual In-line with Window (JW) – 300 mil**
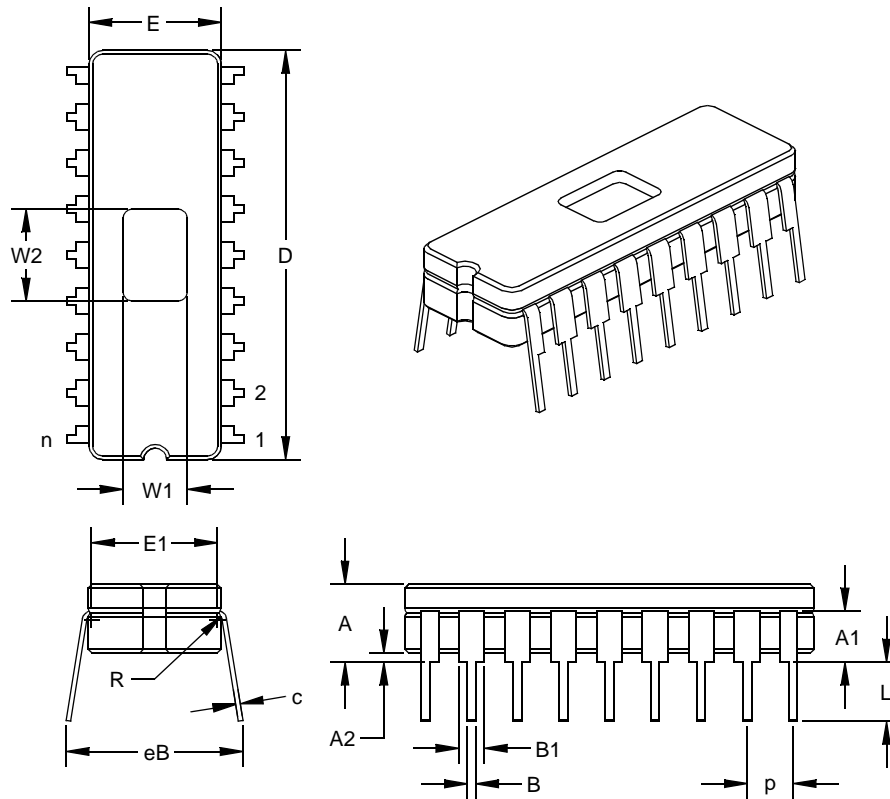


| Units | | INCHES* | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| PCB Row Spacing | | | 0.300 | | | 7.62 | |
| Number of Pins | n | | 18 | | | 18 | |
| Pitch | p | 0.098 | 0.100 | 0.102 | 2.49 | 2.54 | 2.59 |
| Lower Lead Width | B | 0.016 | 0.019 | 0.021 | 0.41 | 0.47 | 0.53 |
| Upper Lead Width | B1 | 0.050 | 0.055 | 0.060 | 1.27 | 1.40 | 1.52 |
| Shoulder Radius | R | 0.010 | 0.013 | 0.015 | 0.25 | 0.32 | 0.38 |
| Lead Thickness | c | 0.008 | 0.010 | 0.012 | 0.20 | 0.25 | 0.30 |
| Top to Seating Plane | A | 0.175 | 0.183 | 0.190 | 4.45 | 4.64 | 4.83 |
| Top of Lead to Seating Plane | A1 | 0.091 | 0.111 | 0.131 | 2.31 | 2.82 | 3.33 |
| Base to Seating Plane | A2 | 0.015 | 0.023 | 0.030 | 0.00 | 0.57 | 0.76 |
| Tip to Seating Plane | L | 0.125 | 0.138 | 0.150 | 3.18 | 3.49 | 3.81 |
| Package Length | D | 0.880 | 0.900 | 0.920 | 22.35 | 22.86 | 23.37 |
| Package Width | E | 0.285 | 0.298 | 0.310 | 7.24 | 7.56 | 7.87 |
| Radius to Radius Width | E1 | 0.255 | 0.270 | 0.285 | 6.48 | 6.86 | 7.24 |
| Overall Row Spacing | eB | 0.345 | 0.385 | 0.425 | 8.76 | 9.78 | 10.80 |
| Window Width | W1 | 0.130 | 0.140 | 0.150 | 0.13 | 0.14 | 0.15 |
| Window Length | W2 | 0.190 | 0.200 | 0.210 | 0.19 | 0.2 | 0.21 |

     *   Controlling Parameter.
JEDEC equivalent:        MO-036 AE

**Package Type:   K04-051  18-Lead Plastic Small Outline (SO) – Wide, 300 mil**
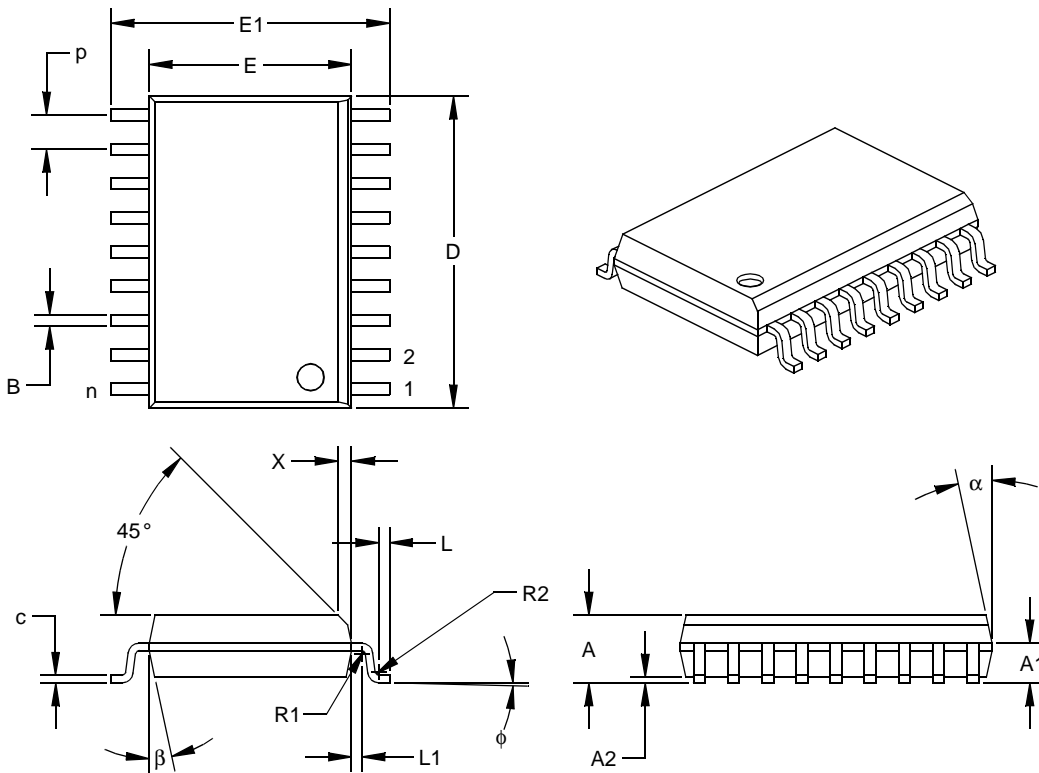


| Units | | INCHES* | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Pitch | p | | 0.050 | | | 1.27 | |
| Number of Pins | n | | 18 | | | 18 | |
| Overall Pack. Height | A | 0.093 | 0.099 | 0.104 | 2.36 | 2.50 | 2.64 |
| Shoulder Height | A1 | 0.048 | 0.058 | 0.068 | 1.22 | 1.47 | 1.73 |
| Standoff | A2 | 0.004 | 0.008 | 0.011 | 0.10 | 0.19 | 0.28 |
| Molded Package Length | D‡ | 0.450 | 0.456 | 0.462 | 11.43 | 11.58 | 11.73 |
| Molded Package Width | E‡ | 0.292 | 0.296 | 0.299 | 7.42 | 7.51 | 7.59 |
| Outside Dimension | E1 | 0.394 | 0.407 | 0.419 | 10.01 | 10.33 | 10.64 |
| Chamfer Distance | X | 0.010 | 0.020 | 0.029 | 0.25 | 0.50 | 0.74 |
| Shoulder Radius | R1 | 0.005 | 0.005 | 0.010 | 0.13 | 0.13 | 0.25 |
| Gull Wing Radius | R2 | 0.005 | 0.005 | 0.010 | 0.13 | 0.13 | 0.25 |
| Foot Length | L | 0.011 | 0.016 | 0.021 | 0.28 | 0.41 | 0.53 |
| Foot Angle | φ | 0 | 4 | 8 | 0 | 4 | 8 |
| Radius Centerline | L1 | 0.010 | 0.015 | 0.020 | 0.25 | 0.38 | 0.51 |
| Lead Thickness | c | 0.009 | 0.011 | 0.012 | 0.23 | 0.27 | 0.30 |
| Lower Lead Width | B† | 0.014 | 0.017 | 0.019 | 0.36 | 0.42 | 0.48 |
| Mold Draft Angle Top | α | 0 | 12 | 15 | 0 | 12 | 15 |
| Mold Draft Angle Bottom | β | 0 | 12 | 15 | 0 | 12 | 15 |

\*   Controlling Parameter.

†   Dimension "B" does not include dam-bar protrusions. Dam-bar protrusions shall not exceed 0.003"
    (0.076 mm) per side or 0.006" (0.152 mm) more than dimension "B."

‡   Dimensions "D" and "E" do not include mold flash or protrusions. Mold flash or protrusions shall not
    exceed 0.010" (0.254 mm) per side or 0.020" (0.508 mm) more than dimensions "D" or "E."

JEDEC equivalent:   MS-013 AB

**Package Type:    K04-072  20-Lead Plastic Shrink Small Outine (SS) – 5.30 mm**
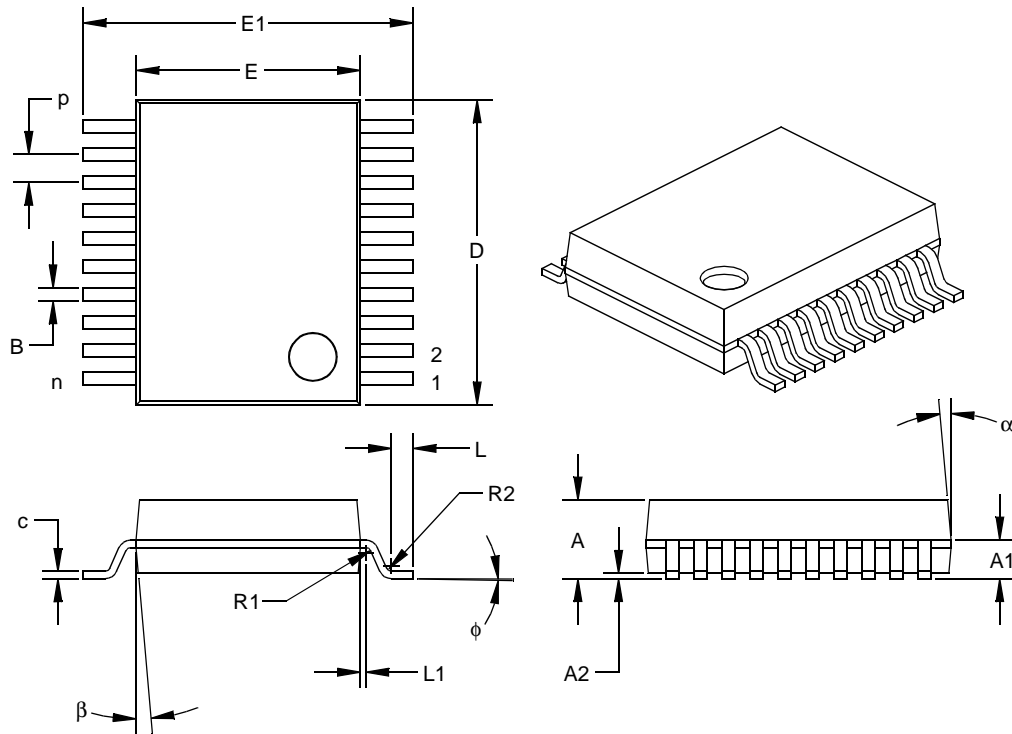


| Units | | INCHES | | | MILLIMETERS* | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Pitch | p | | 0.026 | | | 0.65 | |
| Number of Pins | n | | 20 | | | 20 | |
| Overall Pack. Height | A | 0.068 | 0.073 | 0.078 | 1.73 | 1.86 | 1.99 |
| Shoulder Height | A1 | 0.026 | 0.036 | 0.046 | 0.66 | 0.91 | 1.17 |
| Standoff | A2 | 0.002 | 0.005 | 0.008 | 0.05 | 0.13 | 0.21 |
| Molded Package Length | D‡ | 0.278 | 0.283 | 0.289 | 7.07 | 7.20 | 7.33 |
| Molded Package Width | E‡ | 0.205 | 0.208 | 0.212 | 5.20 | 5.29 | 5.38 |
| Outside Dimension | E1 | 0.301 | 0.306 | 0.311 | 7.65 | 7.78 | 7.90 |
| Shoulder Radius | R1 | 0.005 | 0.005 | 0.010 | 0.13 | 0.13 | 0.25 |
| Gull Wing Radius | R2 | 0.005 | 0.005 | 0.010 | 0.13 | 0.13 | 0.25 |
| Foot Length | L | 0.015 | 0.020 | 0.025 | 0.38 | 0.51 | 0.64 |
| Foot Angle | φ | 0 | 4 | 8 | 0 | 4 | 8 |
| Radius Centerline | L1 | 0.000 | 0.005 | 0.010 | 0.00 | 0.13 | 0.25 |
| Lead Thickness | c | 0.005 | 0.007 | 0.009 | 0.13 | 0.18 | 0.22 |
| Lower Lead Width | B† | 0.010 | 0.012 | 0.015 | 0.25 | 0.32 | 0.38 |
| Mold Draft Angle Top | α | 0 | 5 | 10 | 0 | 5 | 10 |
| Mold Draft Angle Bottom | β | 0 | 5 | 10 | 0 | 5 | 10 |

\*   Controlling Parameter.

†   Dimension "B" does not include dam-bar protrusions. Dam-bar protrusions shall not exceed 0.003"
    (0.076 mm) per side or 0.006" (0.152 mm) more than dimension "B."

‡   Dimensions "D" and "E" do not include mold flash or protrusions. Mold flash or protrusions shall not
    exceed 0.010" (0.254 mm) per side or 0.020" (0.508 mm) more than dimensions "D" or "E."

JEDEC equivalent:  MO-150 AE

---

**NOTES:**

## APPENDIX A: REVISION HISTORY

| Version | Date | Revision Description |
|---------|------|----------------------|
| A | 2/99 | This is a new data sheet. However, the devices described in this data sheet are the upgrades to the devices found in the *PIC16C6X Data Sheet*, DS30234, and the *PIC16C7X Data Sheet*, DS30390. |

## APPENDIX B: CONVERSION CONSIDERATIONS

There are no previous versions of this device.

## APPENDIX C: MIGRATION FROM BASE-LINE TO MID-RANGE DEVICES

This section discusses how to migrate from a baseline device (i.e., PIC16C5X) to a mid-range device (i.e., PIC16CXXX).

The following are the list of modifications over the PIC16C5X microcontroller family:

1. Instruction word length is increased to 14-bits. This allows larger page sizes both in program memory (2K now as opposed to 512 before) and register file (128 bytes now versus 32 bytes before).

2. A PC high latch register (PCLATH) is added to handle program memory paging. Bits PA2, PA1, PA0 are removed from STATUS register.

3. Data memory paging is redefined slightly. STATUS register is modified.

4. Four new instructions have been added: RETURN, RETFIE, ADDLW, and SUBLW. Two instructions TRIS and OPTION are being phased out although they are kept for compatibility with PIC16C5X.

5. OPTION_REG and TRIS registers are made addressable.

6. Interrupt capability is added. Interrupt vector is at 0004h.

7. Stack size is increased to 8 deep.

8. Reset vector is changed to 0000h.

9. Reset of all registers is revisited. Five different reset (and wake-up) types are recognized. Registers are reset differently.

10. Wake up from SLEEP through interrupt is added.

11. Two separate timers, Oscillator Start-up Timer (OST) and Power-up Timer (PWRT) are included for more reliable power-up. These timers are invoked selectively to avoid unnecessary delays on power-up and wake-up.

12. PORTB has weak pull-ups and interrupt on change feature.

13. T0CKI pin is also a port pin (RA4) now.

14. FSR is made a full eight bit register.

15. "In-circuit serial programming" is made possible. The user can program PIC16CXX devices using only five pins: VDD, VSS, $\overline{MCLR}$/VPP, RB6 (clock) and RB7 (data in/out).

16. PCON status register is added with a Power-on Reset status bit ($\overline{POR}$).

17. Code protection scheme is enhanced such that portions of the program memory can be protected, while the remainder is unprotected.

18. Brown-out protection circuitry has been added. Controlled by configuration word bit BODEN. Brown-out reset ensures the device is placed in a reset condition if VDD dips below a fixed set-point.

To convert code written for PIC16C5X to PIC16CXXX, the user should take the following steps:

1. Remove any program memory page select operations (PA2, PA1, PA0 bits) for CALL, GOTO.

2. Revisit any computed jump operations (write to PC or add to PC, etc.) to make sure page bits are set properly under the new scheme.

3. Eliminate any data memory page switching. Redefine data variables to reallocate them.

4. Verify all writes to STATUS, OPTION, and FSR registers since these have changed.

5. Change reset vector to 0000h.

**NOTES:**

**Preliminary**

## ON-LINE SUPPORT

Microchip provides on-line support on the Microchip World Wide Web (WWW) site.

The web site is used by Microchip as a means to make files and information easily available to customers. To view the site, the user must have access to the Internet and a web browser, such as Netscape or Microsoft Explorer. Files are also available for FTP download from our FTP site.

### Connecting to the Microchip Internet Web Site

The Microchip web site is available by using your favorite Internet browser to attach to:

**www.microchip.com**

The file transfer site is available by using an FTP service to connect to:

**ftp://ftp.microchip.com**

The web site and file transfer site provide a variety of services. Users may download files for the latest Development Tools, Data Sheets, Application Notes, User's Guides, Articles and Sample Programs. A variety of Microchip specific business information is also available, including listings of Microchip sales offices, distributors and factory representatives. Other data available for consideration is:

- Latest Microchip Press Releases
- Technical Support Section with Frequently Asked Questions
- Design Tips
- Device Errata
- Job Postings
- Microchip Consultant Program Member Listing
- Links to other useful web sites related to Microchip Products
- Conferences for products, Development Systems, technical information and more
- Listing of seminars and events

### Systems Information and Upgrade Hot Line

The Systems Information and Upgrade Line provides system users a listing of the latest versions of all of Microchip's development systems software products. Plus, this line provides information on how customers can receive any currently available upgrade kits. The Hot Line Numbers are:

1-800-755-2345 for U.S. and most of Canada, and

1-480-786-7302 for the rest of the world.

981103

# PIC16C712/716

## 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) 786-7578.

Please list the following information, and use this outline to provide us with your comments about this Data Sheet.

To: Technical Publications Manager      Total Pages Sent _____

RE: Reader Response

From: Name _____

Company _____

Address _____

City / State / ZIP / Country _____

Telephone: (_____) _____ - _____     FAX: (_____) _____ - _____

Application (optional): _____

Would you like a reply? ____Y ____N

Device: **PIC16C712/716**      Literature Number: **DS41106A**

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 data sheet easy to follow? If not, why?

   _____

   _____

4. What additions to the data sheet do you think would enhance the structure and subject?

   _____

   _____

5. What deletions from the data sheet 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?

   _____

   _____

8. How would you improve our software, systems, and silicon products?

   _____

   _____

## PIC16C712/716 PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

| PART NO. | -XX | X | /XX | XXX |
|---|---|---|---|---|
| Device | Frequency Range | Temperature Range | Package | Pattern |

| | |
|---|---|
| Device | PIC16C712[1], PIC16C712T[2];$V_{DD}$ range 4.0V to 5.5V<br>PIC16LC712[1], PIC16LC712T[2];$V_{DD}$ range 2.5V to 5.5V<br>PIC16C716[1], PIC16C716T[2];$V_{DD}$ range 4.0V to 5.5V<br>PIC16LC716[1], PIC16LC716T[2];$V_{DD}$ range 2.5V to 5.5V |
| Frequency Range | 04    = 4 MHz<br>20    = 20 MHz |
| Temperature Range | blank  =    0°C to    70°C   (Commercial)<br>I      =  -40°C to  +85°C  (Industrial)<br>E     =  -40°C to +125°C  (Extended) |
| Package | JW  =   Windowed CERDIP<br>SO  =   SOIC<br>P    =   PDIP<br>SS  =   SSOP |
| Pattern | QTP, SQTP, Code or Special Requirements<br>(blank otherwise) |

**Examples:**

a) PIC16C716 - 04/P 301 = Commercial temp., PDIP package, 4 MHz, normal $V_{DD}$ limits, QTP pattern #301.

b) PIC16LC712 - 04I/SO = Industrial temp., SOIC package, 200 kHz, Extended $V_{DD}$ limits.

c) PIC16C712 - 20I/P = Industrial temp., PDIP package, 20MHz, normal $V_{DD}$ limits.

**Note 1:** C    = CMOS<br>          LC   = Low Power CMOS
**2:** T    = in tape and reel - SOIC, SSOP packages only.
**3:** LC extended temperature device is not offered.
**4:** LC is not offered at 20 MHz

\* JW Devices are UV erasable and can be programmed to any device configuration. JW Devices meet the electrical requirement of each oscillator type (including LC devices).

## Sales and Support

**Data Sheets**
Products supported by a preliminary Data Sheet may have an errata sheet describing minor operational differences and recommended workarounds. To determine if an errata sheet exists for a particular device, please contact one of the following:

1. Your local Microchip sales office
2. The Microchip Corporate Literature Center U.S. FAX: (480) 786-7277
3. The Microchip Worldwide Site (www.microchip.com)

Please specify which device, revision of silicon and Data Sheet (include Literature #) you are using.

**New Customer Notification System**
Register on our web site (www.microchip.com/cn) to receive the most current information on our products.

**Note the following details of the code protection feature on PICmicro® MCUs.**

- The PICmicro family meets the specifications contained in the Microchip Data Sheet.
- Microchip believes that its family of PICmicro microcontrollers is one of the most secure products of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the PICmicro microcontroller in a manner outside the operating specifications contained in the data sheet. The person doing so may be engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable".
- Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our product.

If you have any further questions about this matter, please contact the local sales office nearest to you.

**Trademarks**

The Microchip name and logo, the Microchip logo, FilterLab, KEELOQ, microID, MPLAB, PIC, PICmicro, PICMASTER, PICSTART, PRO MATE, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

dsPIC, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, microPort, Migratable Memory, MPASM, MPLIB, MPLINK, MPSIM, MXDEV, PICC, PICDEM, PICDEM.net, rfPIC, Select Mode and Total Endurance are trademarks of Microchip Technology Incorporated in the U.S.A.

Serialized Quick Turn Programming (SQTP) is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.
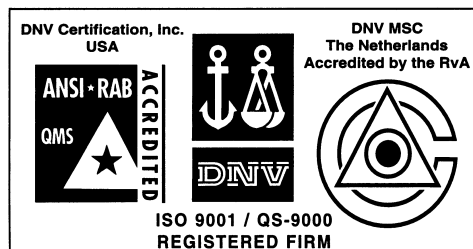
*Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999. The Company's quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs and microperipheral products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001 certified.*