



# **PIC16F716**

## **Data Sheet**

8-bit Flash-based Microcontroller  
with A/D Converter and  
Enhanced Capture/Compare/PWM

---

**Note the following details of the code protection feature on Microchip devices:**

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families 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 Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is 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 products. Attempts to break microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

---

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights.

#### Trademarks

The Microchip name and logo, the Microchip logo, dsPIC, KEELoQ, MPLAB, PIC, PICmicro, PICSTART, PRO MATE and PowerSmart are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

FilterLab, microID, MXDEV, MXLAB, PICMASTER, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

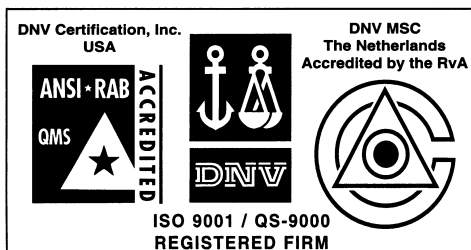
Accuron, Application Maestro, dsPICDEM, dsPICDEM.net, ECAN, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, microPort, Migratable Memory, MPASM, MPLIB, MPLINK, MPSIM, PICC, PICkit, PICDEM, PICDEM.net, PowerCal, PowerInfo, PowerMate, PowerTool, rLAB, rPIC, Select Mode, SmartSensor, SmartShunt, SmartTel and Total Endurance are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

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.

© 2003, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 Printed on recycled paper.



*Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999 and Mountain View, California in March 2002. The Company's quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELoQ® code hopping devices, Serial EEPROMs, microperipherals, non-volatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001 certified.*

## 8-bit Flash-based Microcontroller with A/D Controller and Enhanced Capture/Compare PWM

### 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
- Interrupt capability  
(up to 7 internal/external interrupt sources)
- 8-level deep hardware stack
- Direct, Indirect and Relative Addressing modes

### Special Microcontroller Features

- 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
- Dual level Brown-out Reset circuitry
  - 2.5 V<sub>BOR</sub> (Typical)
  - 4.0 V<sub>BOR</sub> (Typical)
- Programmable code protection
- Power saving Sleep mode
- Selectable oscillator options
- Fully static design
- In-Circuit Serial Programming™ (ICSP™)

### CMOS Technology

- Wide operating voltage range:
  - Industrial: 2.0V to 5.5V
  - Extended: 3.0V to 5.5V
- High Sink/Source Current 25/25 mA
- Wide temperature range:
  - Industrial: -40°C to 85°C
  - Extended: -40°C to 125°C

### Low-Power Features:

- Standby Current:
  - 100 nA @ 2.0V, typical
- Operating Current:
  - 14 µA @ 32 kHz, 2.0V, typical
  - 120 µA @ 1 MHz, 2.0V, typical
- Watchdog Timer Circuit:
  - 1 µA @ 2.0V, typical
- Timer1 Oscillator Current:
  - 3.0 µA @ 32 kHz, 2.0V, typical

### Peripheral Features:

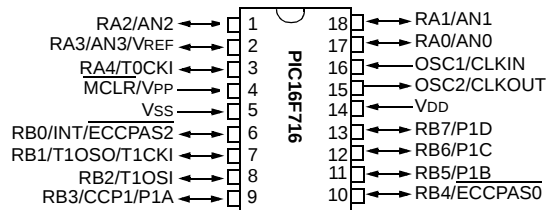
- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler can be incremented during Sleep via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Enhanced Capture, Compare, PWM module:
  - Capture is 16-bit, max. resolution is 12.5 ns
  - Compare is 16-bit, max. resolution is 200 ns
  - PWM maximum resolution is 10-bit
  - Enhanced PWM:
    - Single, Half-Bridge and Full-Bridge modes
    - Digitally programmable dead-band delay
    - Auto-shutdown/restart
- 8-bit multi-channel Analog-to-Digital converter
- 13 I/O pins with individual direction control
- Programmable weak pull-ups on PORTB

| Device    | Memory    |         | I/O | 8-bit A/D (ch) | Timers 8/16 | PWM (outputs) | V <sub>DD</sub> Range |
|-----------|-----------|---------|-----|----------------|-------------|---------------|-----------------------|
|           | Flash     | Data    |     |                |             |               |                       |
| PIC16F716 | 2048 x 14 | 128 x 8 | 13  | 4              | 2/1         | 1/2/4         | 2.0V - 5.5V           |

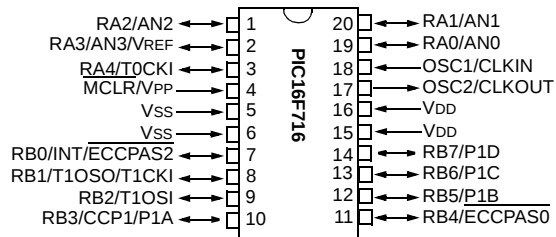
# PIC16F716

## Pin Diagrams

### 18-pin PDIP, SOIC



### 20-pin SSOP



## Table of Contents

|      |                                                                 |     |
|------|-----------------------------------------------------------------|-----|
| 1.0  | Device Overview .....                                           | 5   |
| 2.0  | Memory Organization .....                                       | 7   |
| 3.0  | I/O Ports .....                                                 | 19  |
| 4.0  | Timer0 Module .....                                             | 27  |
| 5.0  | Timer1 Module .....                                             | 29  |
| 6.0  | Timer2 Module .....                                             | 31  |
| 7.0  | Enhanced Capture/Compare/PWM (ECCP) Module.....                 | 33  |
| 8.0  | Analog-to-Digital Converter (A/D) Module.....                   | 49  |
| 9.0  | Special Features of the CPU .....                               | 55  |
| 10.0 | Instruction Set Summary .....                                   | 71  |
| 11.0 | Development Support.....                                        | 85  |
| 12.0 | Electrical Characteristics .....                                | 91  |
| 13.0 | DC and AC Characteristics Graphs and Tables.....                | 107 |
| 14.0 | Packaging Information.....                                      | 109 |
|      | Appendix A: Revision History.....                               | 113 |
|      | Appendix B: Conversion Considerations .....                     | 113 |
|      | Appendix C: Migration from Base-line to MID-RANGE Devices ..... | 114 |
|      | On-Line Support.....                                            | 115 |
|      | Systems Information and Upgrade Hot Line .....                  | 115 |
|      | Reader Response .....                                           | 116 |
|      | Index .....                                                     | 117 |
|      | Product Identification System .....                             | 123 |

## TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at [docerrors@mail.microchip.com](mailto:docerrors@mail.microchip.com) or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. We welcome your feedback.

### 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).

### Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. 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) 792-7277

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.

### Customer Notification System

Register on our web site at [www.microchip.com/cn](http://www.microchip.com/cn) to receive the most current information on all of our products.

# PIC16F716

---

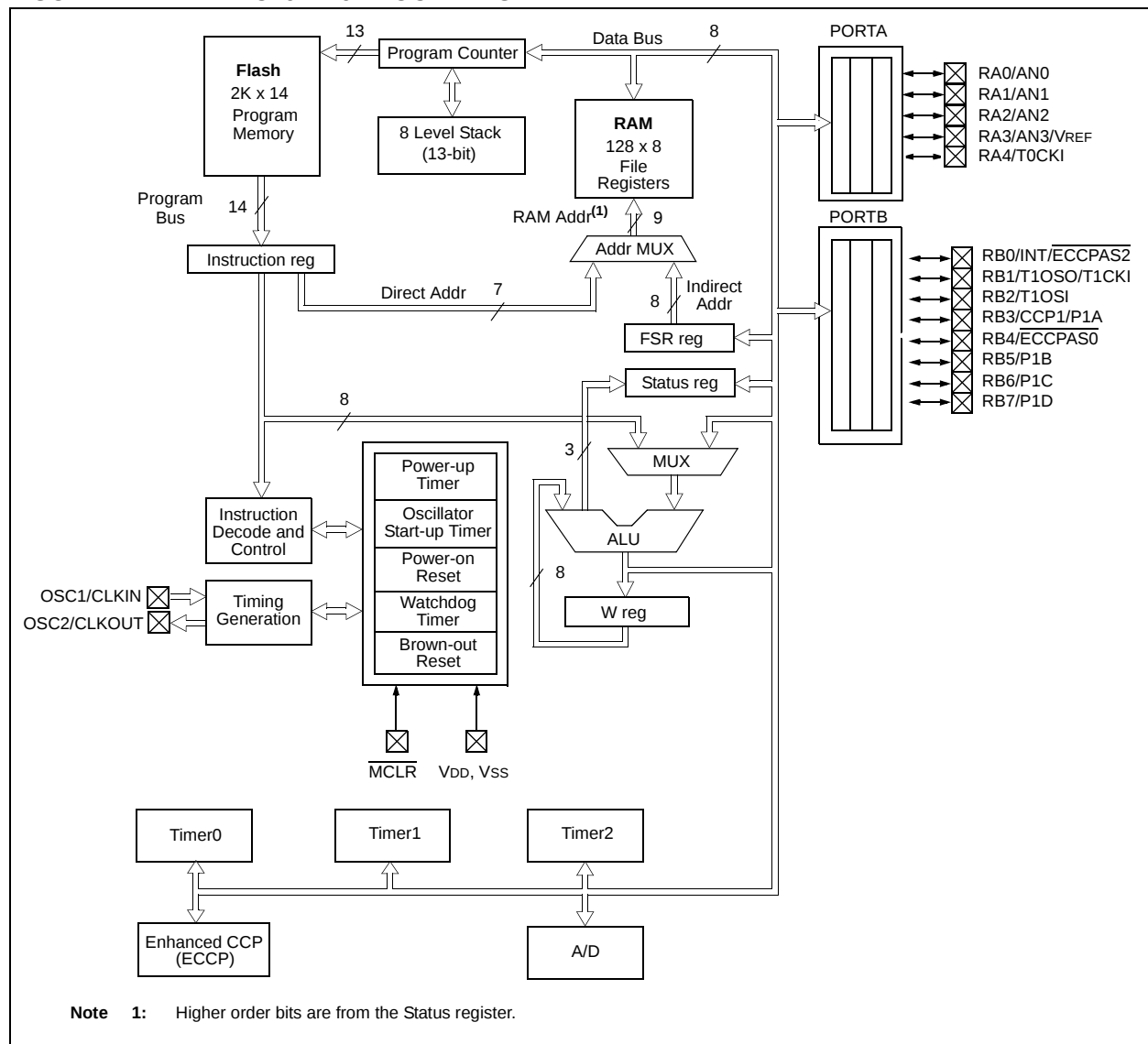
NOTES:

## 1.0 DEVICE OVERVIEW

This document contains device specific information for the PIC16F716. 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 web site ([www.microchip.com](http://www.microchip.com)). 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.

Figure 1-1 is the block diagram for the PIC16F716 device. The pinouts are listed in Table 1-1.

**FIGURE 1-1: PIC16F716 BLOCK DIAGRAM**



# PIC16F716

**TABLE 1-1: PIC16F716 PINOUT DESCRIPTION**

| Name            | Function | Input Type | Output Type | Description                                                                                                      |
|-----------------|----------|------------|-------------|------------------------------------------------------------------------------------------------------------------|
| MCLR/VPP        | MCLR     | ST         | —           | Master clear (Reset) input. This pin is an active low Reset to the device.                                       |
|                 | VPP      | P          | —           | Programming voltage input                                                                                        |
| OSC1/CLKIN      | OSC1     | XTAL       | —           | Oscillator crystal input                                                                                         |
|                 | CLKIN    | CMOS       | —           | External clock source input                                                                                      |
|                 | CLKIN    | ST         | —           | RC Oscillator mode                                                                                               |
| OSC2/CLKOUT     | OSC2     | XTAL       | —           | Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode.                          |
|                 | CLKOUT   | —          | CMOS        | In RC mode, OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate. |
| RA0/AN0         | RA0      | TTL        | CMOS        | Bidirectional I/O                                                                                                |
|                 | AN0      | AN         | —           | Analog Channel 0 input                                                                                           |
| RA1/AN1         | RA1      | TTL        | CMOS        | Bidirectional I/O                                                                                                |
|                 | AN1      | AN         | —           | Analog Channel 1 input                                                                                           |
| RA2/AN2         | RA2      | TTL        | CMOS        | Bidirectional I/O                                                                                                |
|                 | AN2      | AN         | —           | Analog Channel 2 input                                                                                           |
| RA3/AN3/VREF    | RA3      | TTL        | CMOS        | Bidirectional I/O                                                                                                |
|                 | AN3      | AN         | —           | Analog Channel 3 input                                                                                           |
|                 | VREF     | AN         | —           | A/D reference voltage input                                                                                      |
| RA4/T0CKI       | RA4      | ST         | OD          | Bidirectional I/O. Open drain when configured as output.                                                         |
|                 | T0CKI    | ST         | —           | Timer0 external clock input                                                                                      |
| RB0/INT/ECCPAS2 | RB0      | TTL        | CMOS        | Bidirectional I/O. Programmable weak pull-up.                                                                    |
|                 | INT      | ST         | —           | External Interrupt                                                                                               |
|                 | ECCPAS2  | ST         | —           | ECCP Auto-Shutdown pin                                                                                           |
| RB1/T1OSO/T1CKI | RB1      | TTL        | CMOS        | Bidirectional I/O. Programmable weak pull-up.                                                                    |
|                 | T1OSO    | —          | XTAL        | Timer1 oscillator output. Connects to crystal in Oscillator mode.                                                |
|                 | T1CKI    | ST         | —           | Timer1 external clock input                                                                                      |
| RB2/T1OSI       | RB2      | TTL        | CMOS        | Bidirectional I/O. Programmable weak pull-up.                                                                    |
|                 | T1OSI    | XTAL       | —           | Timer1 oscillator input. Connects to crystal in Oscillator mode.                                                 |
| RB3/CCP1/P1A    | RB3      | TTL        | CMOS        | Bidirectional I/O. Programmable weak pull-up.                                                                    |
|                 | CCP1     | ST         | CMOS        | Capture1 input, Compare1 output, PWM1 output.                                                                    |
|                 | P1A      | —          | CMOS        | PWM P1A output                                                                                                   |
| RB4/ECCPAS0     | RB4      | TTL        | CMOS        | Bidirectional I/O. Programmable weak pull-up. Interrupt-on-change.                                               |
|                 | ECCPAS0  | ST         | —           | ECCP Auto-Shutdown pin                                                                                           |
| RB5/P1B         | RB5      | TTL        | CMOS        | Bidirectional I/O. Programmable weak pull-up. Interrupt-on-change.                                               |
|                 | P1B      | —          | CMOS        | PWM P1B output                                                                                                   |
| RB6/P1C         | RB6      | TTL        | CMOS        | Bidirectional I/O. Programmable weak pull-up. Interrupt-on-change. ST input when used as ICSP programming clock. |
|                 | P1C      | —          | CMOS        | PWM P1C output                                                                                                   |
| RB7/P1D         | RB7      | TTL        | CMOS        | Bidirectional I/O. Programmable weak pull-up. Interrupt-on-change. ST input when used as ICSP programming data.  |
|                 | P1D      | —          | CMOS        | PWM P1D output                                                                                                   |
| VSS             | VSS      | P          | —           | Ground reference for logic and I/O pins.                                                                         |
| VDD             | VDD      | P          | —           | Positive supply for logic and I/O pins.                                                                          |

**Legend:** I = Input      AN = Analog input or output      OD = Open drain  
O = Output      TTL = TTL compatible input      ST = Schmitt Trigger input with CMOS levels  
P = Power      XTAL = Crystal      CMOS = CMOS compatible input or output

## 2.0 MEMORY ORGANIZATION

There are two memory blocks in the PIC16F716 PICmicro® microcontroller device. Each block (program memory and data memory) has its own bus so that concurrent access can occur.

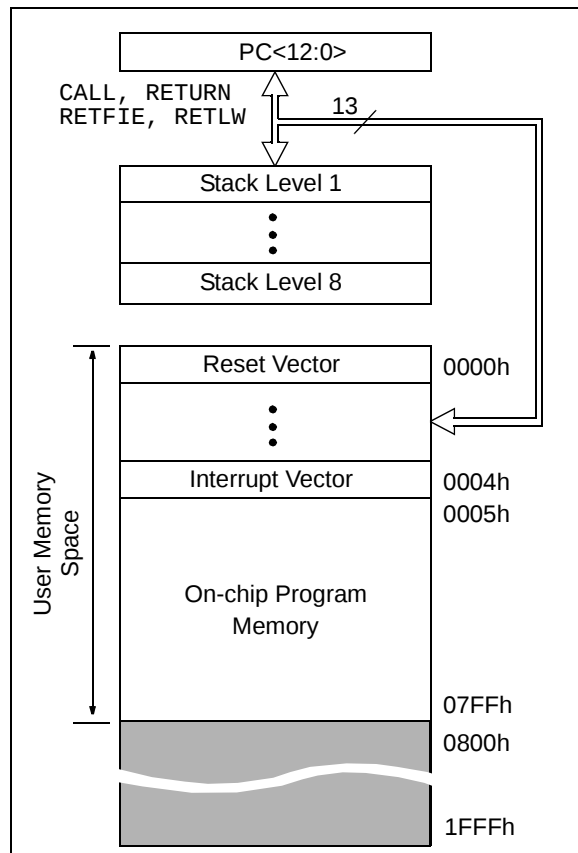
Additional information on device memory may be found in the PICmicro® Mid-Range Reference Manual, (DS33023).

### 2.1 Program Memory Organization

The PIC16F716 has a 13-bit program counter capable of addressing an 8K x 14 program memory space. The PIC16F716 has 2K x 14 words of program memory. Accessing a location above the physically implemented address will cause a wrap-around.

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

**FIGURE 2-1: PROGRAM MEMORY MAP AND STACK OF PIC16F716**



## 2.2 Data Memory Organization

The data memory is partitioned into multiple banks which contain the General Purpose Registers (GPR) and the Special Function Registers (SFR). Bits RP1 and RP0 of the Status register are the bank select bits.

| RP1:RP0 <sup>(1)</sup><br>(status<6:5>) | Bank             |
|-----------------------------------------|------------------|
| 00                                      | 0                |
| 01                                      | 1                |
| 10                                      | 2 <sup>(2)</sup> |
| 11                                      | 3 <sup>(2)</sup> |

**Note 1:** Maintain Status bit 6 clear to ensure upward compatibility with future products.

**2:** Not implemented

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. The upper 16 bytes of GPR space and some "high use" Special Function Registers in Bank 0 are mirrored in Bank 1 for code reduction and quicker access.

# PIC16F716

## 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 “Indirect Addressing, INDF and FSR Registers”).

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

| File Address |                           |                           | File Address |
|--------------|---------------------------|---------------------------|--------------|
| 00h          | INDF <sup>(1)</sup>       | INDF <sup>(1)</sup>       | 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          |                           |                           | 87h          |
| 08h          |                           |                           | 88h          |
| 09h          |                           |                           | 89h          |
| 0Ah          | PCLATH                    | PCLATH                    | 8Ah          |
| 0Bh          | INTCON                    | INTCON                    | 8Bh          |
| 0Ch          | PIR1                      | PIE1                      | 8Ch          |
| 0Dh          |                           |                           | 8Dh          |
| 0Eh          | TMR1L                     | PCON                      | 8Eh          |
| 0Fh          | TMR1H                     |                           | 8Fh          |
| 10h          | T1CON                     |                           | 90h          |
| 11h          | TMR2                      |                           | 91h          |
| 12h          | T2CON                     | PR2                       | 92h          |
| 13h          |                           |                           | 93h          |
| 14h          |                           |                           | 94h          |
| 15h          | CCPR1L                    |                           | 95h          |
| 16h          | CCPR1H                    |                           | 96h          |
| 17h          | CCP1CON                   |                           | 97h          |
| 18h          | PWM1CON                   |                           | 98h          |
| 19h          | ECCPAS                    |                           | 99h          |
| 1Ah          |                           |                           | 9Ah          |
| 1Bh          |                           |                           | 9Bh          |
| 1Ch          |                           |                           | 9Ch          |
| 1Dh          |                           |                           | 9Dh          |
| 1Eh          | ADRES                     |                           | 9Eh          |
| 1Fh          | ADCON0                    | ADCON1                    | 9Fh          |
| 20h          | General Purpose Registers | General Purpose Registers | A0h          |
|              | 80 Bytes                  | 32 Bytes                  | BFh          |
| 6Fh          |                           |                           | C0h          |
| 70h          | 16 Bytes                  |                           | EFh          |
| 7Fh          |                           | Accesses 70-7Fh           | F0h          |
|              |                           |                           | FFh          |
|              | Bank 0                    | Bank 1                    |              |

 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 BANK 0**

| Address | Name                    | Bit 7                                                                                          | Bit 6              | Bit 5            | Bit 4                                                    | Bit 3           | Bit 2                 | Bit 1            | Bit 0   | Value on:<br>POR, BOR | Page |
|---------|-------------------------|------------------------------------------------------------------------------------------------|--------------------|------------------|----------------------------------------------------------|-----------------|-----------------------|------------------|---------|-----------------------|------|
| 00h     | INDF <sup>(1)</sup>     | Addressing this location uses contents of FSR to address data memory (not a physical register) |                    |                  |                                                          |                 |                       |                  |         | 0000 0000             | 18   |
| 01h     | TMR0                    | Timer0 module's register                                                                       |                    |                  |                                                          |                 |                       |                  |         | xxxx xxxx             | 27   |
| 02h     | PCL <sup>(1)</sup>      | Program Counter's (PC) Least Significant Byte                                                  |                    |                  |                                                          |                 |                       |                  |         | 0000 0000             | 17   |
| 03h     | STATUS <sup>(1)</sup>   | IRP <sup>(4)</sup>                                                                             | RP1 <sup>(4)</sup> | RP0              | $\overline{TO}$                                          | $\overline{PD}$ | Z                     | DC               | C       | 0001 1xxx             | 11   |
| 04h     | FSR <sup>(1)</sup>      | Indirect data memory address pointer                                                           |                    |                  |                                                          |                 |                       |                  |         | xxxx xxxx             | 18   |
| 05h     | PORTA <sup>(5,6)</sup>  | —                                                                                              | —                  | — <sup>(7)</sup> | PORTA Data Latch when written: PORTA pins when read      |                 |                       |                  |         | --xx 0000             | 19   |
| 06h     | PORTB <sup>(5,6)</sup>  | PORTB Data Latch when written: PORTB pins when read                                            |                    |                  |                                                          |                 |                       |                  |         | xxxx xxxx             | 21   |
| 07h-09h | —                       | Unimplemented                                                                                  |                    |                  |                                                          |                 |                       |                  |         | —                     |      |
| 0Ah     | PCLATH <sup>(1,2)</sup> | —                                                                                              | —                  | —                | Write Buffer for the upper 5 bits of the Program Counter |                 |                       |                  |         | ---0 0000             | 17   |
| 0Bh     | INTCON <sup>(1)</sup>   | GIE                                                                                            | PEIE               | T0IE             | INTE                                                     | RBIE            | T0IF                  | INTF             | RBIF    | 0000 000x             | 13   |
| 0Ch     | PIR1                    | —                                                                                              | ADIF               | —                | —                                                        | —               | CCP1IF                | TMR2IF           | TMR1IF  | -0-- 0000             | 15   |
| 0Dh     | —                       | Unimplemented                                                                                  |                    |                  |                                                          |                 |                       |                  |         | —                     |      |
| 0Eh     | TMR1L                   | Holding register for the Least Significant Byte of the 16-bit TMR1 register                    |                    |                  |                                                          |                 |                       |                  |         | xxxx xxxx             | 29   |
| 0Fh     | TMR1H                   | Holding register for the Most Significant Byte of the 16-bit TMR1 register                     |                    |                  |                                                          |                 |                       |                  |         | xxxx xxxx             | 29   |
| 10h     | T1CON                   | —                                                                                              | —                  | T1CKPS1          | T1CKPS0                                                  | T1OSCEN         | $\overline{T1SYNC}$   | TMR1CS           | TMR1ON  | --00 0000             | 29   |
| 11h     | TMR2                    | Timer2 module's register                                                                       |                    |                  |                                                          |                 |                       |                  |         | 0000 0000             | 31   |
| 12h     | T2CON                   | —                                                                                              | TOUTPS3            | TOUTPS2          | TOUTPS1                                                  | TOUTPS0         | TMR2ON                | T2CKPS1          | T2CKPS0 | -000 0000             | 31   |
| 13h-14h | —                       | Unimplemented                                                                                  |                    |                  |                                                          |                 |                       |                  |         | —                     |      |
| 15h     | CCPR1L                  | Capture/Compare/PWM Register 1 (LSB)                                                           |                    |                  |                                                          |                 |                       |                  |         | xxxx xxxx             | 34   |
| 16h     | CCPR1H                  | Capture/Compare/PWM Register 1 (MSB)                                                           |                    |                  |                                                          |                 |                       |                  |         | xxxx xxxx             | 34   |
| 17h     | CCP1CON                 | P1M1                                                                                           | P1M0               | DC1B1            | DC1B0                                                    | CCP1M3          | CCP1M2                | CCP1M1           | CCP1M0  | 0000 0000             | 33   |
| 18h     | PWM1CON                 | PRSEN                                                                                          | PDC6               | PDC5             | PDC4                                                     | PDC3            | PDC2                  | PDC1             | PDC0    | 0000 0000             | 46   |
| 19h     | ECCPAS                  | ECCPASE                                                                                        | ECCPAS2            | — <sup>(8)</sup> | ECCPAS0                                                  | PSSAC1          | PSSAC0                | PSSBD1           | PSSBD0  | 00-0 0000             | 46   |
| 1Ah-1Dh | —                       | Unimplemented                                                                                  |                    |                  |                                                          |                 |                       |                  |         | —                     |      |
| 1Eh     | ADRES                   | A/D Result Register                                                                            |                    |                  |                                                          |                 |                       |                  |         | xxxx xxxx             | 49   |
| 1Fh     | ADCON0                  | ADCS1                                                                                          | ADCS0              | CHS2             | CHS1                                                     | CHS0            | GO/ $\overline{DONE}$ | — <sup>(7)</sup> | ADON    | 0000 0000             | 49   |

**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.
  - 8: ECCPAS1 bit is not used on PIC16F716.

# PIC16F716

**TABLE 2-2: SPECIAL FUNCTION REGISTER SUMMARY BANK 1**

| Address | Name                    | Bit 7                                                                                          | Bit 6              | Bit 5            | Bit 4                                                    | Bit 3                  | Bit 2  | Bit 1                   | Bit 0                   | Value on:<br>POR, BOR | Page      |    |
|---------|-------------------------|------------------------------------------------------------------------------------------------|--------------------|------------------|----------------------------------------------------------|------------------------|--------|-------------------------|-------------------------|-----------------------|-----------|----|
| 80h     | INDF <sup>(1)</sup>     | Addressing this location uses contents of FSR to address data memory (not a physical register) |                    |                  |                                                          |                        |        |                         |                         | 0000 0000             | 18        |    |
| 81h     | OPTION_REG              | $\overline{\text{RBP}}$                                                                        | INTEDG             | T0CS             | T0SE                                                     | PSA                    | PS2    | PS1                     | PS0                     | 1111 1111             | 12        |    |
| 82h     | PCL <sup>(1)</sup>      | Program Counter's (PC) Least Significant Byte                                                  |                    |                  |                                                          |                        |        |                         |                         | 0000 0000             | 17        |    |
| 83h     | STATUS <sup>(1)</sup>   | IRP <sup>(4)</sup>                                                                             | RP1 <sup>(4)</sup> | RP0              | $\overline{\text{TO}}$                                   | $\overline{\text{PD}}$ | Z      | DC                      | C                       | 0001 1xxx             | 11        |    |
| 84h     | FSR <sup>(1)</sup>      | Indirect data memory address pointer                                                           |                    |                  |                                                          |                        |        |                         |                         | xxxx xxxx             | 18        |    |
| 85h     | TRISA                   | —                                                                                              | —                  | — <sup>(7)</sup> | PORTA Data Direction Register                            |                        |        |                         |                         |                       | --11 1111 | 19 |
| 86h     | TRISB                   | PORTB Data Direction Register                                                                  |                    |                  |                                                          |                        |        |                         |                         | 1111 1111             | 21        |    |
| 87h-89h | —                       | Unimplemented                                                                                  |                    |                  |                                                          |                        |        |                         |                         | —                     |           |    |
| 8Ah     | PCLATH <sup>(1,2)</sup> | —                                                                                              | —                  | —                | Write Buffer for the upper 5 bits of the Program Counter |                        |        |                         |                         | --0 0000              | 17        |    |
| 8Bh     | INTCON <sup>(1)</sup>   | GIE                                                                                            | PEIE               | T0IE             | INTE                                                     | RBIE                   | T0IF   | INTF                    | RBIF                    | 0000 000x             | 13        |    |
| 8Ch     | PIE1                    | —                                                                                              | ADIE               | —                | —                                                        | —                      | CCP1IE | TMR2IE                  | TMR1IE                  | -0-- -000             | 14        |    |
| 8Dh     | —                       | Unimplemented                                                                                  |                    |                  |                                                          |                        |        |                         |                         | —                     |           |    |
| 8Eh     | PCON                    | —                                                                                              | —                  | —                | —                                                        | —                      | —      | $\overline{\text{POR}}$ | $\overline{\text{BOR}}$ | ---- --qq             | 16        |    |
| 8Fh-91h | —                       | Unimplemented                                                                                  |                    |                  |                                                          |                        |        |                         |                         | —                     |           |    |
| 92h     | PR2                     | Timer2 Period Register                                                                         |                    |                  |                                                          |                        |        |                         |                         | 1111 1111             | 32, 36    |    |
| 93h-9Eh | —                       | Unimplemented                                                                                  |                    |                  |                                                          |                        |        |                         |                         | —                     |           |    |
| 9Fh     | ADCON1                  | —                                                                                              | —                  | —                | —                                                        | —                      | PCFG2  | PCFG1                   | PCFG0                   | ---- -000             | 50        |    |

**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.

## 2.2.2.1 Status Register

The Status register, shown in Register 2-1, 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  $\overline{\text{TO}}$  and  $\overline{\text{PD}}$  bits are not writable. Therefore, the result of an instruction with the Status register as destination may be different than intended.

For example, `CLRF STATUS` will clear the upper-three bits and set the Z bit. This leaves the Status register as 000u u1uu (where u = unchanged).

It is recommended, therefore, that only BCF, BSF, SWAPF and MOVWF instructions are used to alter the Status register because these instructions do not affect 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:** The PIC16F716 does not use bits IRP and RP1 (STATUS<7:6>). Maintain these bits clear to ensure upward compatibility with future products.

**2:** The C and DC bits operate as a borrow and digit borrow bit, respectively, in subtraction. See the SUBLW and SUBWF instructions for examples.

### REGISTER 2-1: 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 <sup>(1)</sup> | RP1 <sup>(1)</sup> | RP0   | $\overline{\text{TO}}$ | $\overline{\text{PD}}$ | Z     | DC    | C     |
| bit 7              |                    |       |                        |                        |       |       | bit 0 |

bit 7 **IRP:** Register Bank Select bit (used for indirect addressing)<sup>(1)</sup>

1 = Bank 2, 3 (100h – 1FFh)

0 = Bank 0, 1 (00h – FFh)

bit 6-5 **RP1<sup>(1)</sup>:RP0:** Register Bank Select bits (used for direct addressing)

01 = Bank 1 (80h – FFh)

00 = Bank 0 (00h – 7Fh)

Each bank is 128 bytes

bit 4  **$\overline{\text{TO}}$ :** Time-out bit

1 = After power-up, CLRWDI 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 CLRWDI 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/borrow bit (ADDWF, ADDLW, SUBLW, SUBWF instructions) (for 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/borrow bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)<sup>(2)</sup>

1 = A carry-out from the Most Significant bit of the result occurred

0 = No carry-out from the Most Significant bit of the result occurred

**Note 1:** Reserved, maintain clear

**2:** For 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.

#### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

# PIC16F716

## 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.

### REGISTER 2-2: 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 |
|--------------------------|--------|-------|-------|-------|-------|-------|-------|
| $\overline{\text{RBPU}}$ | INTEDG | T0CS  | T0SE  | PSA   | PS2   | PS1   | PS0   |
| bit 7                    |        |       |       |       |       |       | bit 0 |

- bit 7  **$\overline{\text{RBPU}}$** : PORTB Weak Pull-up Enable bit  
1 = PORTB weak pull-ups are disabled  
0 = PORTB weak pull-ups are determined by alternate function or TRISBn bit value
- 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  |

#### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

## 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.

### REGISTER 2-3: 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  | TOIE  | INTE  | RBIE  | TOIF  | INTF  | RBIF  |
| bit 7 |       |       |       | bit 0 |       |       |       |

- 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 **TOIE:** TMR0 Overflow Interrupt Enable bit  
1 = Enables the TMR0 interrupt  
0 = Disables the TMR0 interrupt
- bit 4 **INTE:** 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 **TOIF:** 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

#### Legend:

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

# PIC16F716

## 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.

### REGISTER 2-4: PIE1 REGISTER (ADDRESS: 8Ch)

| U-0   | R/W-0 | U-0 | U-0 | U-0 | R/W-0  | R/W-0  | R/W-0  |
|-------|-------|-----|-----|-----|--------|--------|--------|
| —     | ADIE  | —   | —   | —   | CCP1IE | TMR2IE | TMR1IE |
| bit 7 |       |     |     |     |        |        | bit 0  |

- 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

#### Legend:

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

## 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.

### REGISTER 2-5: PIR1 REGISTER (ADDRESS: 0Ch)

| U-0   | R/W-0 | U-0 | U-0 | U-0   | R/W-0  | R/W-0  | R/W-0  |
|-------|-------|-----|-----|-------|--------|--------|--------|
| —     | ADIF  | —   | —   | —     | CCP1IF | TMR2IF | TMR1IF |
| bit 7 |       |     |     | bit 0 |        |        |        |

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

#### Legend:

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

# PIC16F716

## 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 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 BOREN configuration bit is set, BOR is '1' on Power-on Reset and reset to '0' when a Brown-out condition occurs. BOR must then be set by the user and checked on subsequent resets to see if it is clear, indicating that another Brown-out has occurred.

If the BOREN configuration bit is clear, BOR is unknown on Power-on Reset.

REGISTER 2-6: PCON REGISTER (ADDRESS: 8Eh)

|  | U-0   | U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0                   | R/W-q                   |
|--|-------|-----|-----|-----|-----|-----|-------------------------|-------------------------|
|  | —     | —   | —   | —   | —   | —   | $\overline{\text{POR}}$ | $\overline{\text{BOR}}$ |
|  | bit 7 |     |     |     |     |     | bit 0                   |                         |

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

bit 1      **POR:** Power-on Reset Status bit  
1 = No Power-on Reset occurred  
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0      **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)

|                   |                  |                                    |                    |                          |
|-------------------|------------------|------------------------------------|--------------------|--------------------------|
| <b>Legend:</b>    |                  |                                    |                    | q = Depends on condition |
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0' |                    |                          |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared               | x = Bit is unknown |                          |

## 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 MODIFYING PCL

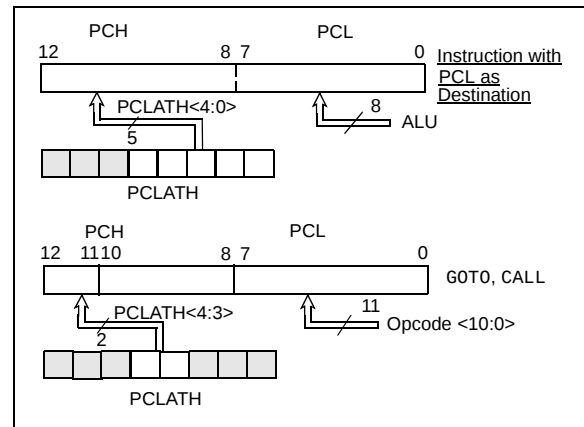
Executing any instruction with the PCL register as the destination simultaneously causes the Program Counter PC<12:8> bits (PCH) to be replaced by the contents of PCLATH register. This allows the entire contents of the program counter to be changed by first writing the desired upper 5 bits to the PCLATH register. When the lower 8 bits are then written to the PCL register, all 13 bits of the program counter will change to the values contained in the PCLATH register and those being written to the PCL register.

Care should be exercised when modifying the PCL register to jump into a look-up table or program branch table (computed GOTO). With PCLATH set to the table start address, if the table is greater than 255 instructions or if the lower 8 bits of the memory address rolls over from 0xFF to 0x00 in the middle of the table, then PCLATH must be incremented for each address rollover that occurs between the table beginning and the target address.

### 2.3.2 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).

**FIGURE 2-3: LOADING OF PC IN DIFFERENT SITUATIONS**



## 2.4 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.

Mid-range 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 8 times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

# PIC16F716

## 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 INDF 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.

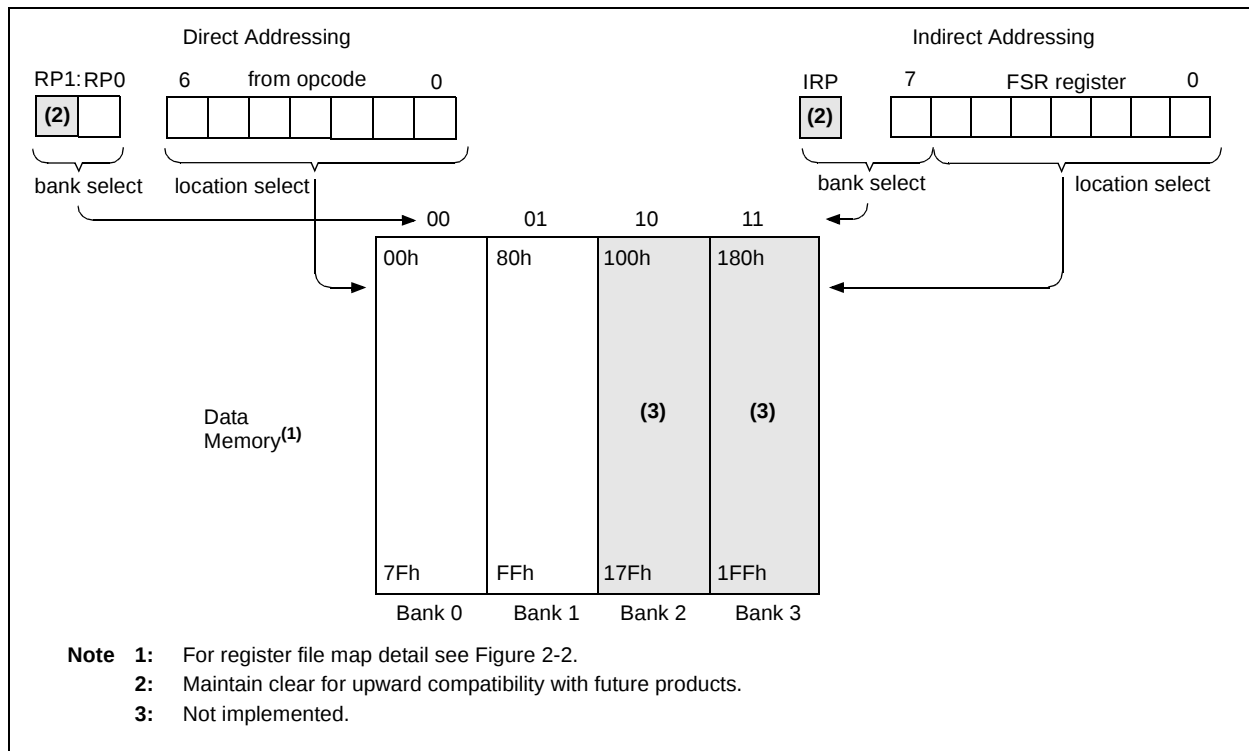
### EXAMPLE 2-2: HOW TO CLEAR RAM USING INDIRECT ADDRESSING

```

MOV LW 0x20 ;initialize pointer
MOV WF FSR ;to RAM
NEXT    CLR F INDF ;clear RAM & FSR
        INC F 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 2-4. However, IRP is not used in the PIC16F716.

**FIGURE 2-4: DIRECT/INDIRECT ADDRESSING**



## 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 bidirectional 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 High-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 Register 1).

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

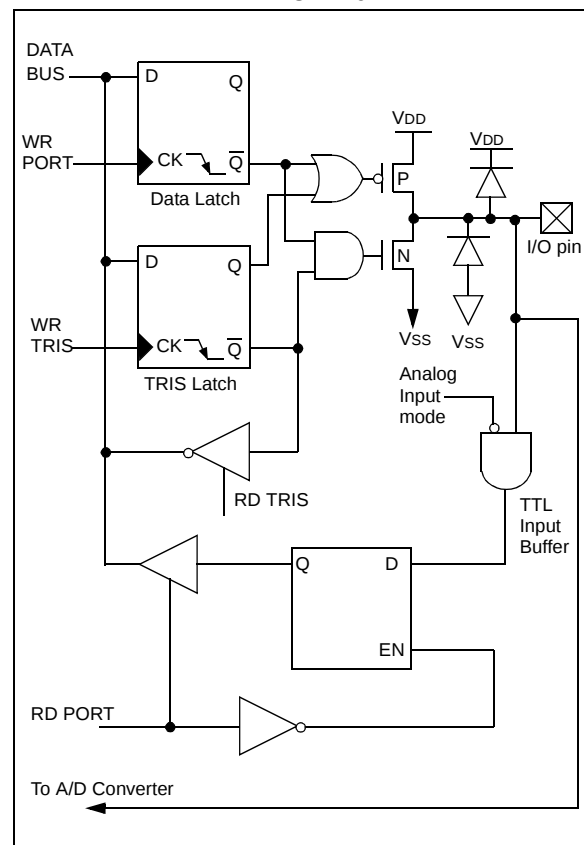
The TRISA register controls the direction of the RA pins, even when they are being used as analog inputs. The user must ensure the bits in the TRISA register are maintained set when using them as analog inputs.

**Note:** Setting RA3:0 to output while in Analog mode will force pins to output contents of data latch.

### 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**



---

[illegible]

| Name         | Bit#  | Buffer | Function                                                                     |
|--------------|-------|--------|------------------------------------------------------------------------------|
| RA0/AN0      | bit 0 | TTL    | Input/output or analog input                                                 |
| RA1/AN1      | bit 1 | TTL    | Input/output or analog input                                                 |
| RA2/AN2      | bit 2 | TTL    | Input/output or analog input                                                 |
| RA3/AN3/VREF | bit 3 | TTL    | Input/output or analog input or VREF                                         |
| RA4/T0CKI    | bit 4 | ST     | Input/output or external clock input for Timer0<br>Output is open drain type |

| 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  | —     | —     | — <sup>(1)</sup> | RA4                           | RA3   | RA2   | RA1   | RA0   | --xx 0000         | --uu uuuu                 |
| 85h     | TRISA  | —     | —     | — <sup>(1)</sup> | PORTA Data Direction Register |       |       |       |       | --11 1111         | --11 1111                 |
| 9Fh     | ADCON1 | —     | —     | —                | —                             | —     | PCFG2 | PCFG1 | PCFG0 | ---- -000         | ---- -000                 |

## 3.2 PORTB and the TRISB Register

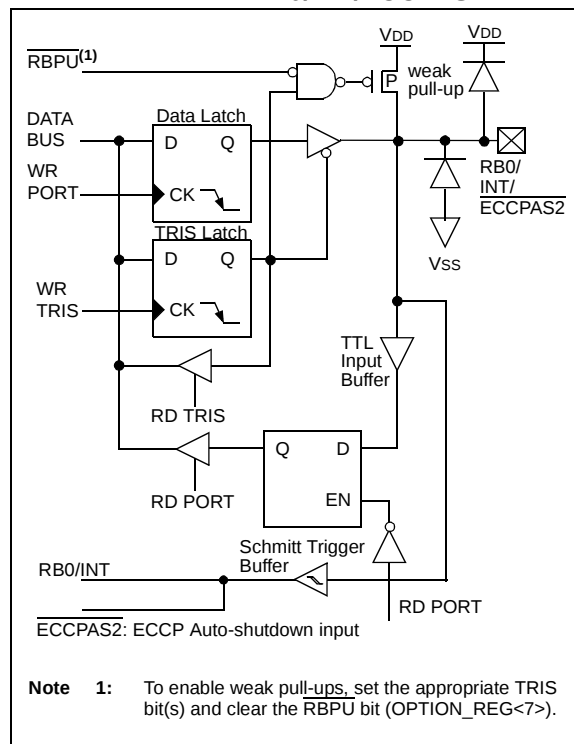
PORTB is an 8-bit wide bidirectional port. The corresponding data direction register is TRISB. Setting a TRISB bit (= 1) will make the corresponding PORTB pin an input (i.e., put the corresponding output driver in a High-impedance mode). Clearing a TRISB bit (= 0) will make the corresponding PORTB pin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

### EXAMPLE 3-2: INITIALIZING PORTB

```
BCF    STATUS, RP0    ;select Bank 0
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
```

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.

**FIGURE 3-3: BLOCK DIAGRAM OF RB0/INT/ECCPAS2 PIN**



PORTB pins RB7:RB0 are multiplexed with several peripheral functions (Table 3-3).

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 (such as BSF, BCF, XORWF) with TRISB as the 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:

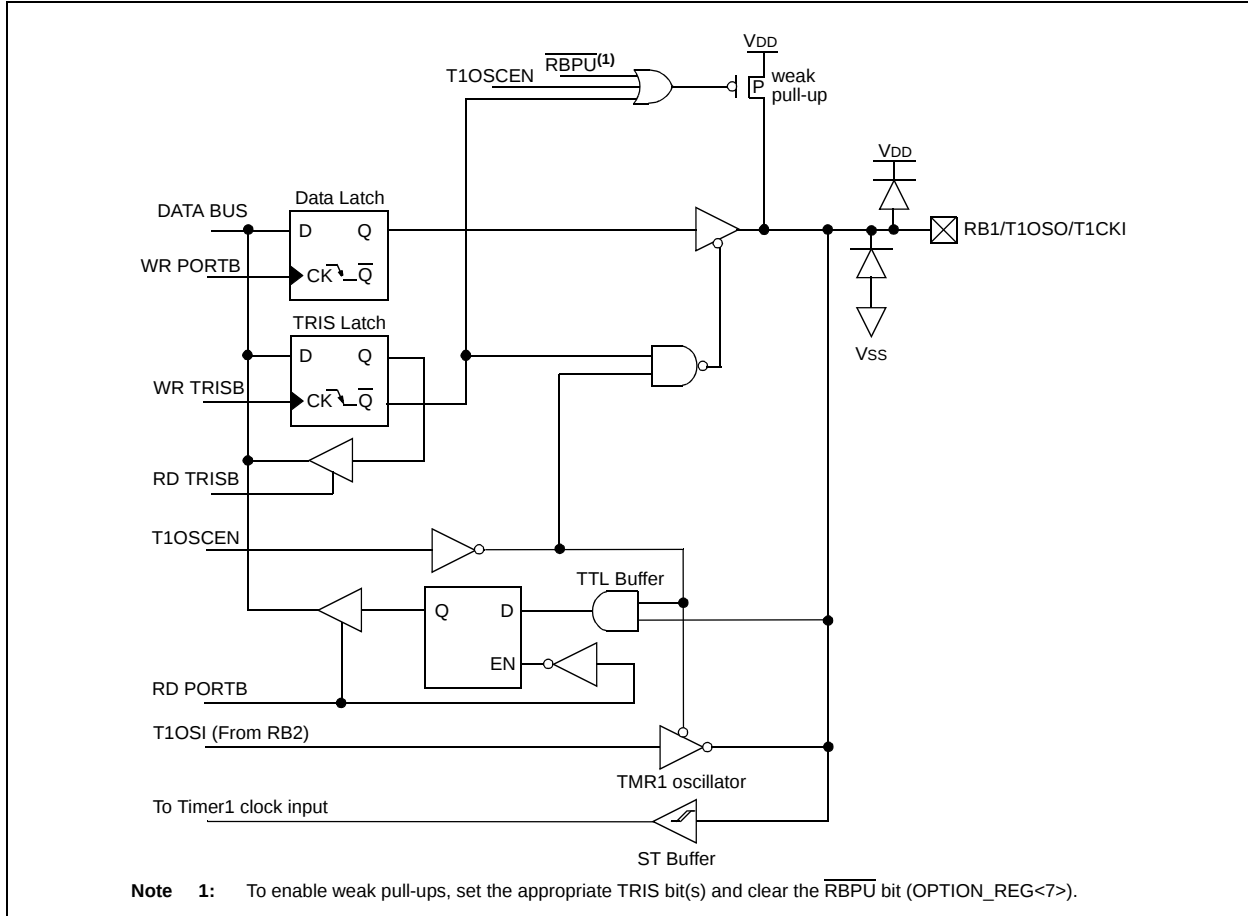
1. Perform a read of PORTB to end the mismatch condition.
2. Clear flag bit RBIF.

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.

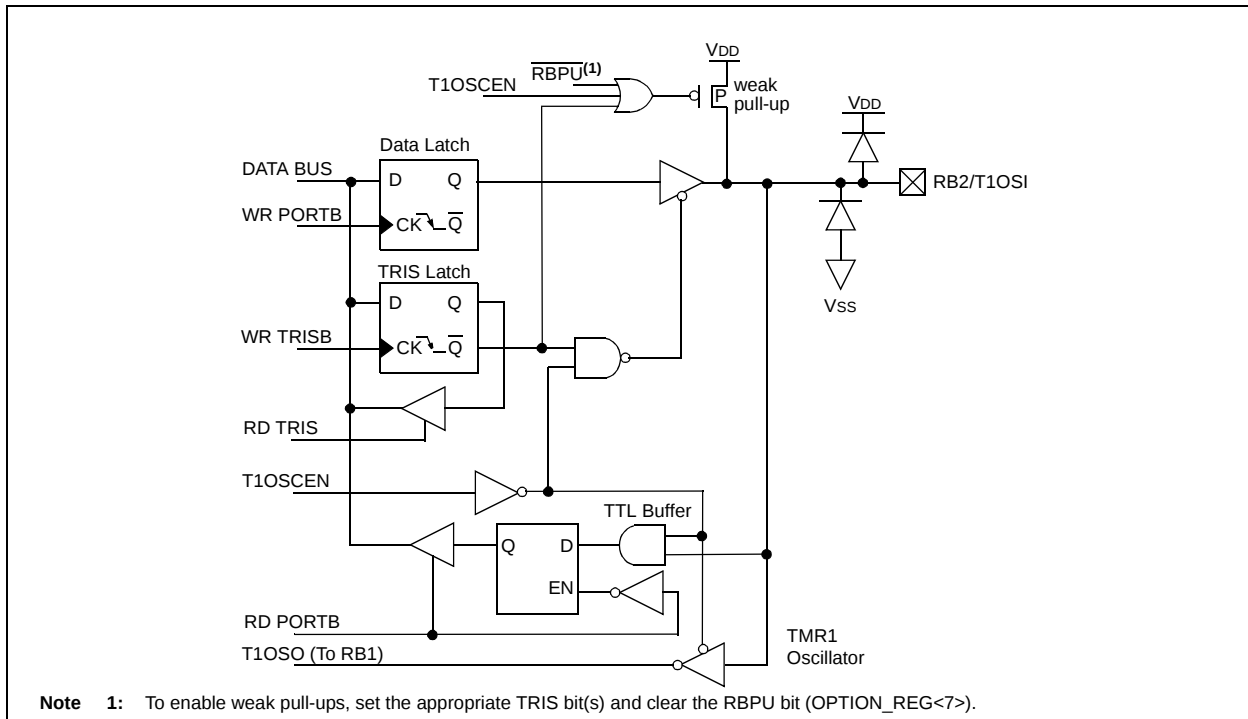
The interrupt-on-change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt-on-change feature. Polling of PORTB is not recommended while using the interrupt-on-change feature.

# PIC16F716

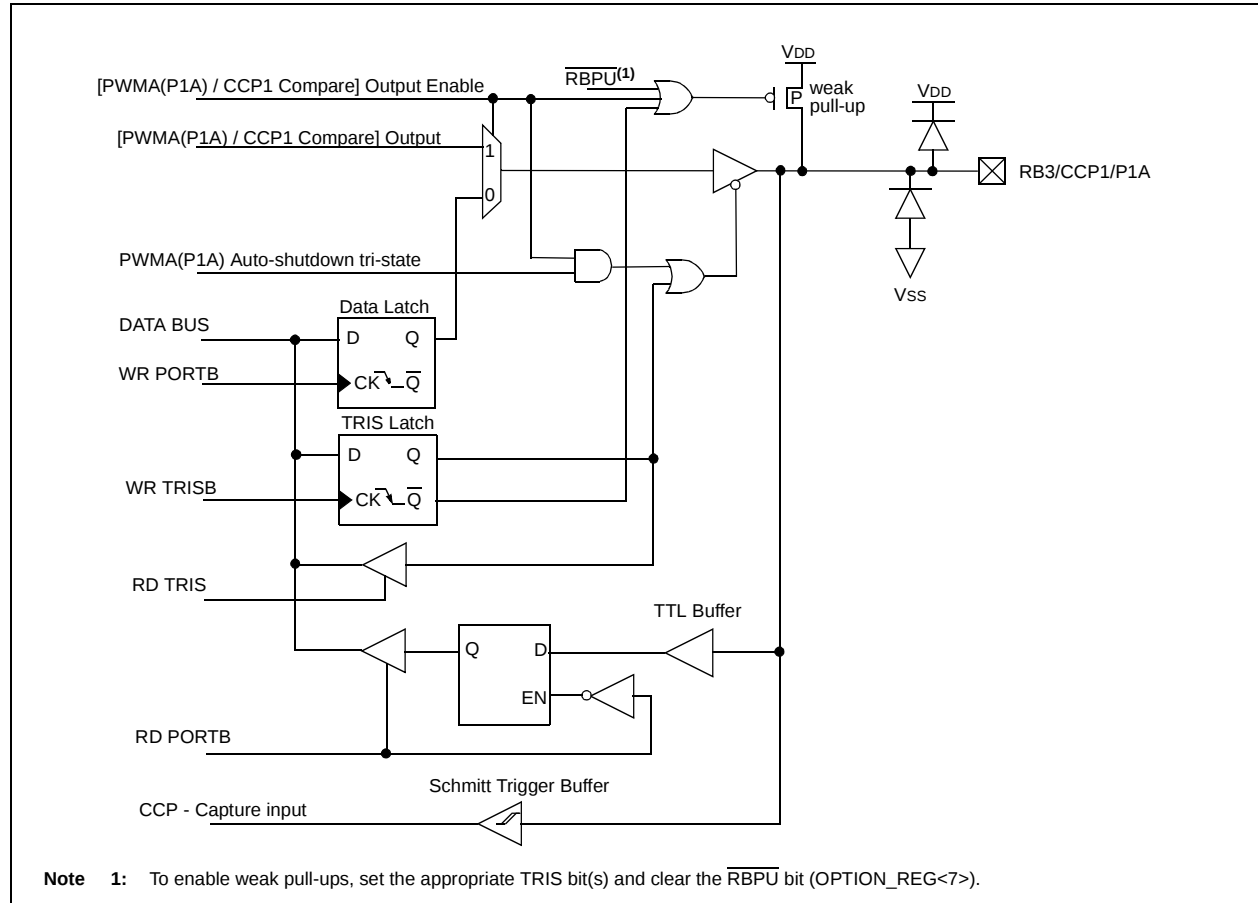
**FIGURE 3-4: BLOCK DIAGRAM OF RB1/T1OSO/T1CKI PIN**



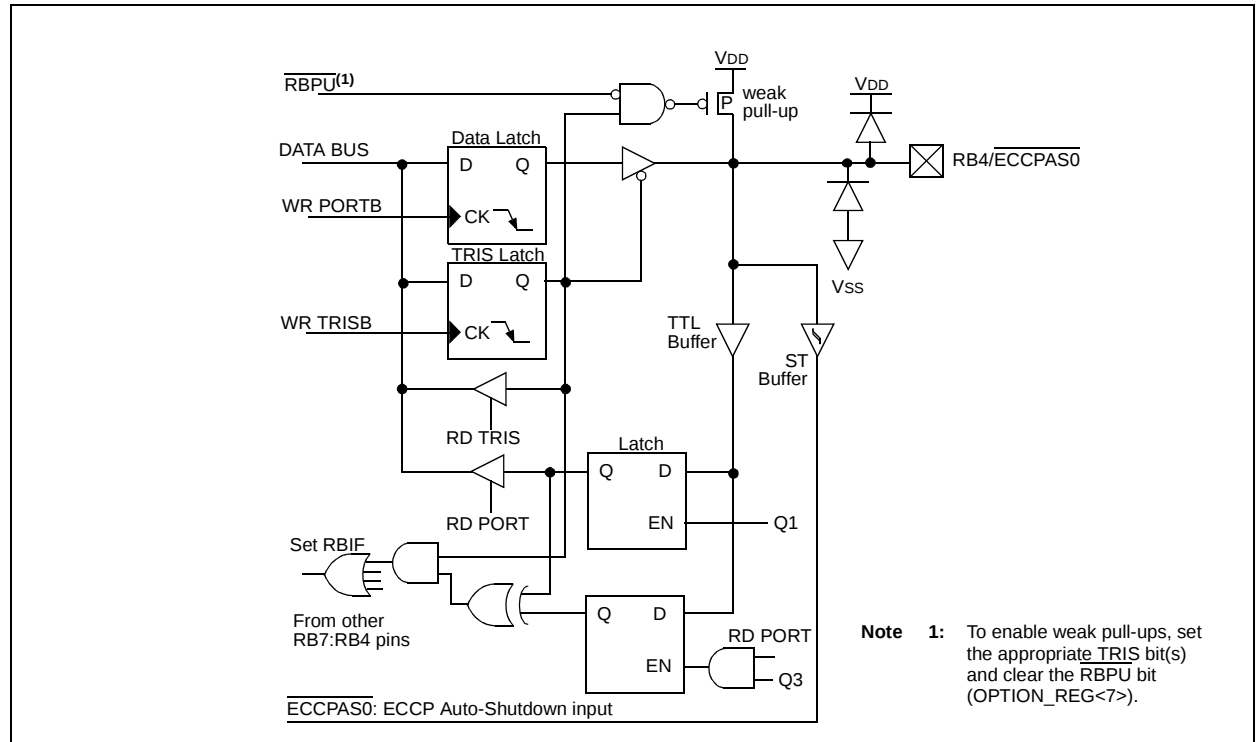
**FIGURE 3-5: BLOCK DIAGRAM OF RB2/T1OSI PIN**



**FIGURE 3-6: BLOCK DIAGRAM OF RB3/CCP1/P1A PIN**



**FIGURE 3-7: BLOCK DIAGRAM OF RB4/ECCPAS0 PIN**

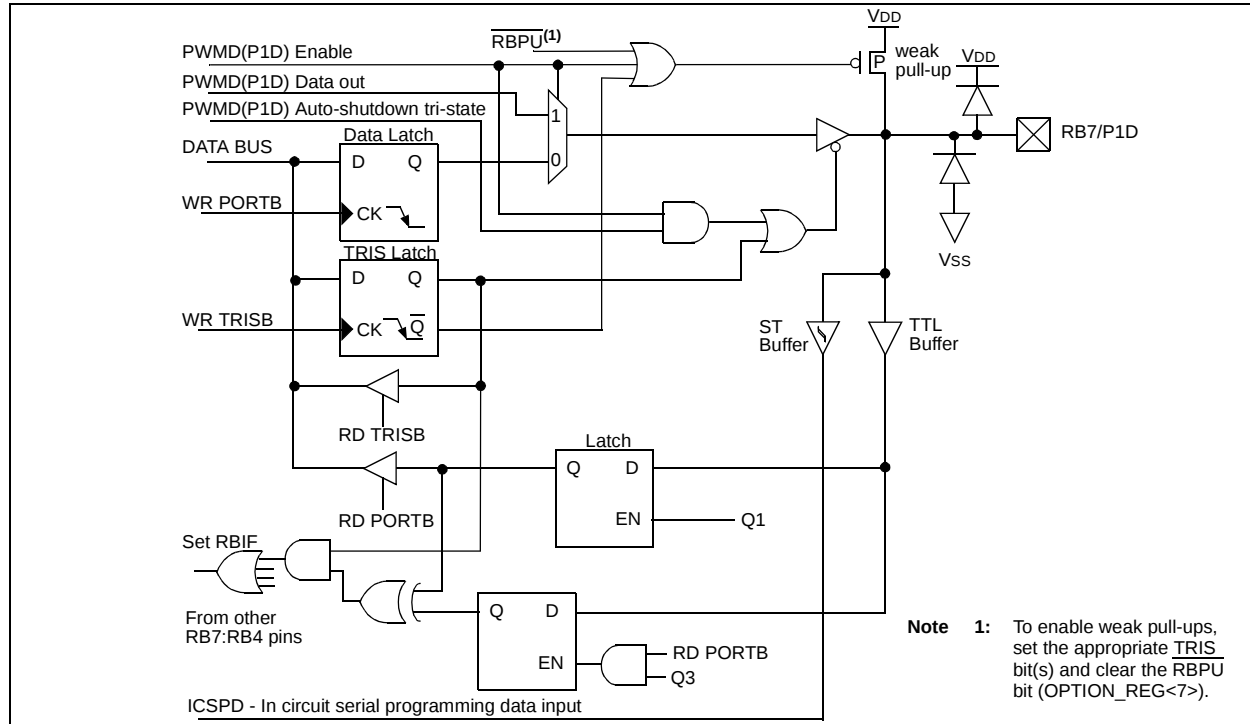


\_\_\_\_\_



\_\_\_\_\_

**FIGURE 3-10: BLOCK DIAGRAM OF RB7/P1D PIN**



**TABLE 3-3: PORTB FUNCTIONS**

| Name                | Bit#  | Buffer                | Function                                                                                                                                                                         |
|---------------------|-------|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| RB0/INT/<br>ECCPAS2 | bit 0 | TTL/ST <sup>(1)</sup> | Input/output pin or external interrupt input. Internal software programmable weak pull-up. ECCP auto-shutdown input.                                                             |
| RB1/T1OS0/<br>T1CKI | bit 1 | TTL/ST <sup>(1)</sup> | Input/output pin or Timer1 oscillator output, or Timer1 clock input. Internal software programmable weak pull-up. See <b>Section 5.0 “Timer1 Module”</b> for detailed operation. |
| RB2/T1OSI           | bit 2 | TTL/XTAL              | Input/output pin or Timer1 oscillator input. Internal software programmable weak pull-up. See <b>Section 5.0 “Timer1 Module”</b> for detailed operation.                         |
| RB3/CCP1/<br>P1A    | bit 3 | TTL/ST <sup>(1)</sup> | Input/output pin or Capture1 input, or Compare1 output, or PWM A output. Internal software programmable weak pull-up. See CCP1 section for detailed operation.                   |
| RB4/<br>ECCPAS0     | bit 4 | TTL                   | Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up. ECCP auto-shutdown input.                                                              |
| RB5/P1B             | bit 5 | TTL                   | Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up. PWM B output.                                                                          |
| RB6/P1C             | bit 6 | TTL/ST <sup>(2)</sup> | Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up. PWM C output. Serial programming clock.                                                |
| RB7/P1D             | bit 7 | TTL/ST <sup>(2)</sup> | Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up. PWM D output. Serial programming data.                                                 |

**Legend:** TTL = TTL input, ST = Schmitt Trigger input, XTAL = Crystal Oscillator 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.

# PIC16F716

**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:<br>POR,<br>BOR | Value on all<br>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.

## 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 postscale for the Watchdog Timer, respectively (Figure 4-2). For simplicity, this counter is being referred to as “prescaler” throughout this data sheet.

**Note:** There is only one prescaler available, which is mutually exclusively shared between the Timer0 module and 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.

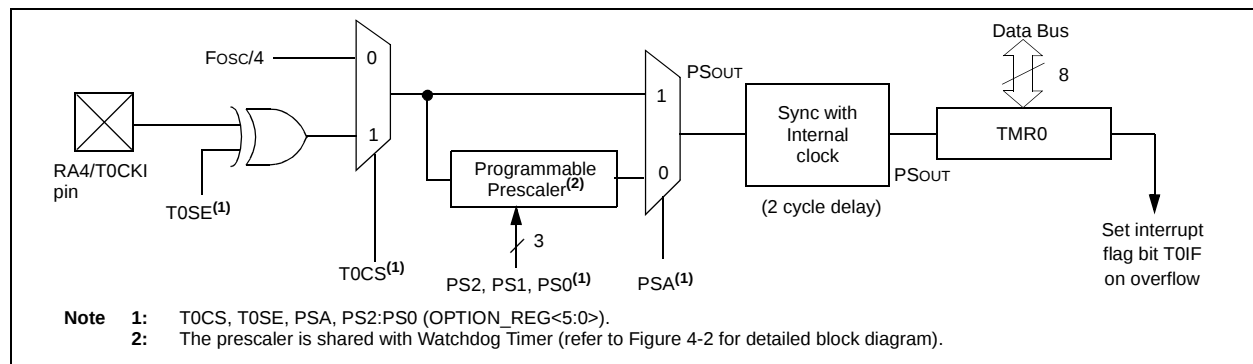
Setting bit PSA will assign the prescaler to the Watchdog Timer (WDT). When the prescaler is assigned to the WDT, prescale values of 1:1, 1:2, ..., 1:128 are selectable.

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 CLRWD 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.

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

**FIGURE 4-1: TIMER0 BLOCK DIAGRAM**



# PIC16F716

## 4.2.1 SWITCHING PRESCALER ASSIGNMENT

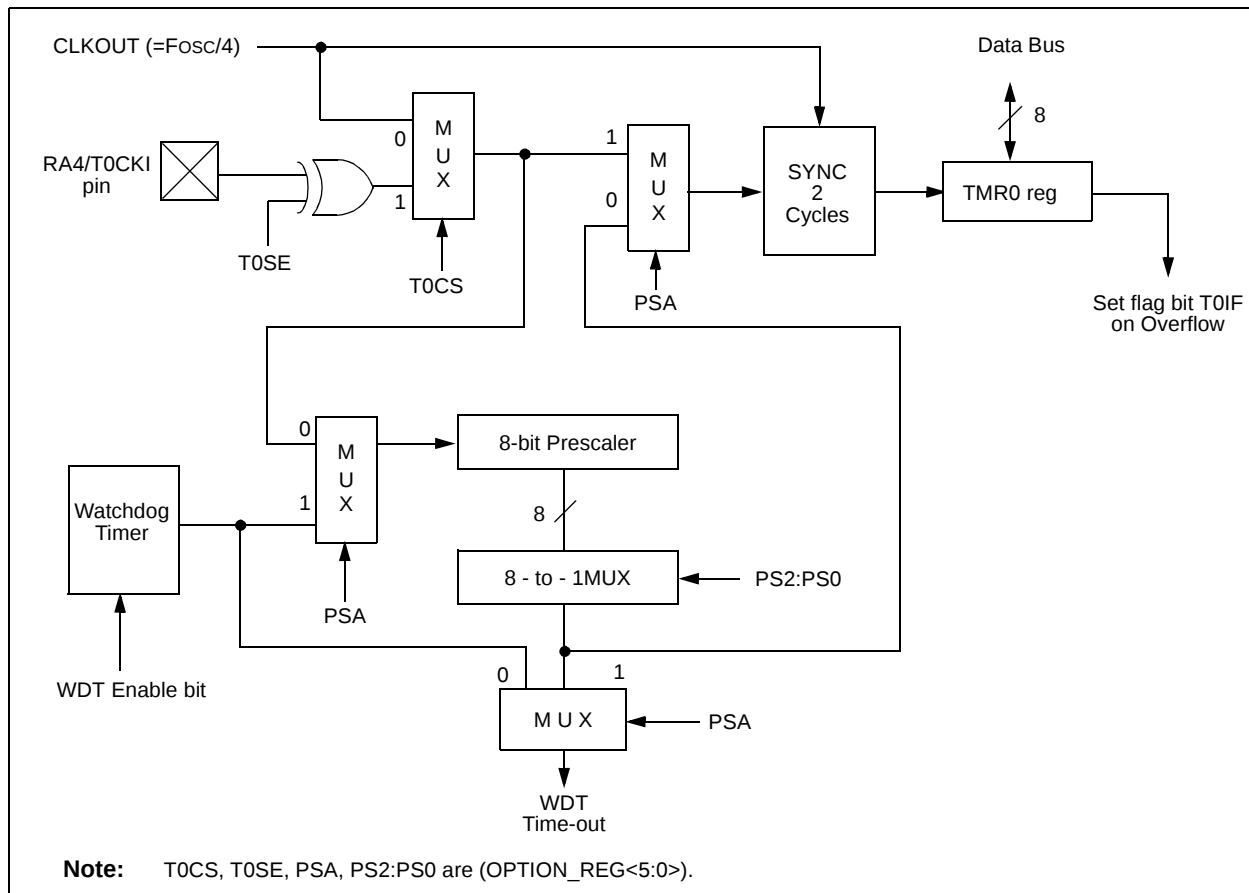
The prescaler assignment is fully under software control (i.e., it can be changed “on-the-fly” during program execution).

**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**



**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 | RBPV                     | INTEDG | T0CS             | T0SE  | PSA                           | PS2   | PS1   | PS0   | 1111 1111          | 1111 1111                 |
| 85h     | TRISA      | —                        | —      | — <sup>(1)</sup> | 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 bits, 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 ECCP module trigger

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

Figure 5-1 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 ECCP module (Section 7.0 "Enhanced Capture/Compare/PWM (ECCP) Module").

### REGISTER 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 |
| bit 7   |                                                                                                                                                                                                                                                                                 |     |         |         |         |        |        | bit 0  |
| bit 7-6 | <b>Unimplemented:</b> Read as '0'                                                                                                                                                                                                                                               |     |         |         |         |        |        |        |
| bit 5-4 | <b>T1CKPS1:T1CKPS0:</b> Timer1 Input Clock Prescale Select bits<br>11 = 1:8 Prescale value<br>10 = 1:4 Prescale value<br>01 = 1:2 Prescale value<br>00 = 1:1 Prescale value                                                                                                     |     |         |         |         |        |        |        |
| bit 3   | <b>T1OSCEN:</b> Timer1 Oscillator Enable Control bit<br>1 = Oscillator is enabled<br>0 = Oscillator is shut off <sup>(1)</sup>                                                                                                                                                  |     |         |         |         |        |        |        |
| bit 2   | <b>T1SYNC:</b> Timer1 External Clock Input Synchronization Control bit<br><u>TMR1CS = 1</u><br>1 = Do not synchronize external clock input<br>0 = Synchronize external clock input<br><u>TMR1CS = 0</u><br>This bit is ignored. Timer1 uses the internal clock when TMR1CS = 0. |     |         |         |         |        |        |        |
| bit 1   | <b>TMR1CS:</b> Timer1 Clock Source Select bit<br>1 = External clock from pin RB1/T1OSO/T1CKI (on the rising edge after the first falling edge)<br>0 = Internal clock (Fosc/4)                                                                                                   |     |         |         |         |        |        |        |
| bit 0   | <b>TMR1ON:</b> Timer1 On bit<br>1 = Enables Timer1<br>0 = Stops Timer1                                                                                                                                                                                                          |     |         |         |         |        |        |        |

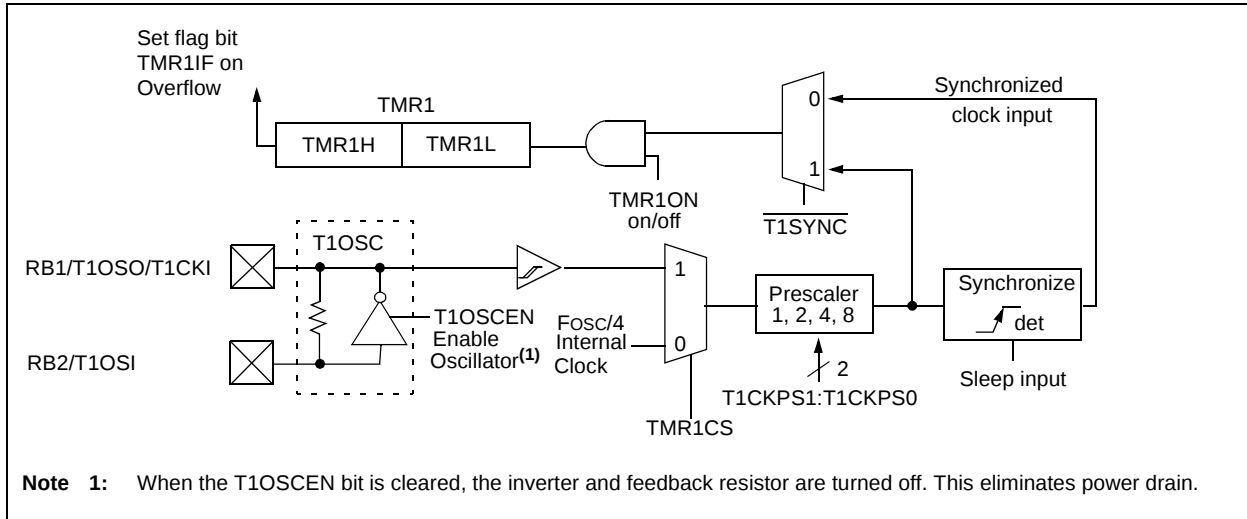
**Note 1:** The oscillator inverter and feedback resistor are turned off to eliminate power drain.

#### Legend:

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

# PIC16F716

**FIGURE 5-1: TIMER1 BLOCK DIAGRAM**



## 5.2 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 designed to operate with a 32.768 kHz tuning fork crystal. It will continue to run during Sleep.

The user must provide a software time delay to ensure proper oscillator start-up.

**Note 1:** Circuit guidelines for the LP oscillator (32 kHz), as shown in **Section 9.2 “Oscillator Configurations”**, also apply to the Timer1 Oscillator.

**2:** The Timer1 register pair, TMR1H and TMR1L, in combination with the Timer1 overflow flag (TMR1IF) can be used as the oscillator start-up stabilization timer.

## 5.3 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>).

## 5.4 Resetting Timer1 using an ECCP Trigger Output

If the ECCP 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 ECCP 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 the ECCP, the write will take precedence.

In this mode of operation, the CCP1H:CCP1L register pair effectively becomes the period register for Timer1.

**TABLE 5-1: 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                 |

**Legend:** x = unknown, u = unchanged, - = unimplemented read as ‘0’. Shaded cells are not used by the Timer1 module.

## 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 Register 6-1.  
Timer2 can be shut off by clearing control bit TMR2ON (T2CON<2>) to minimize power consumption.

Figure 6-1 is a simplified block diagram of the Timer2 module.

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

### REGISTER 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 |
| bit 7   |                                                             |         |         |         |         |        |         | bit 0   |
| bit 7   | <b>Unimplemented:</b> Read as '0'                           |         |         |         |         |        |         |         |
| bit 6-3 | <b>TOUTPS3:TOUTPS0:</b> 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   | <b>TMR2ON:</b> Timer2 On bit                                |         |         |         |         |        |         |         |
|         | 1 = Timer2 is on                                            |         |         |         |         |        |         |         |
|         | 0 = Timer2 is off                                           |         |         |         |         |        |         |         |
| bit 1-0 | <b>T2CKPS1:T2CKPS0:</b> Timer2 Clock Prescale Select bits   |         |         |         |         |        |         |         |
|         | 00 = Prescaler is 1                                         |         |         |         |         |        |         |         |
|         | 01 = Prescaler is 4                                         |         |         |         |         |        |         |         |
|         | 1x = Prescaler is 16                                        |         |         |         |         |        |         |         |

#### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

# PIC16F716

## 6.1 Timer2 Operation

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

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

The input clock ( $F_{osc}/4$ ) 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:

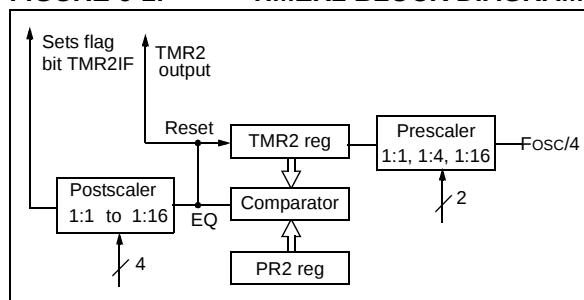
- A write to the TMR2 register
- A write to the T2CON register
- Any device Reset (Power-on Reset, MCLR Reset, Watchdog Timer Reset, or Brown-out Reset)

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.

**FIGURE 6-1: TIMER2 BLOCK DIAGRAM**



**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    | TOIE    | INTE    | RBIE    | TOIF   | 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                 |
| 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.

## 7.0 ENHANCED CAPTURE/ COMPARE/PWM (ECCP) MODULE

The ECCP (Enhanced Capture/Compare/PWM) module contains a 16-bit register, which can operate as:

- 16-bit Capture register
- 16-bit Compare register
- PWM Master/Slave Duty Cycle register

Table 7-1 shows the timer resources of the ECCP 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 ECCP operation. All the CCP1CON bits are readable and writable.

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

**TABLE 7-1: ECCP MODE - TIMER RESOURCE**

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

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

|       |       |       |       |        |        |        |        |
|-------|-------|-------|-------|--------|--------|--------|--------|
| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0  | R/W-0  | R/W-0  | R/W-0  |
| P1M1  | P1M0  | DC1B1 | DC1B0 | CCP1M3 | CCP1M2 | CCP1M1 | CCP1M0 |

bit 7

bit 0

bit 7-6

**P1M1:P1M0:** PWM Output Configuration bits

CCP1M<3:2> = 00, 01, 10

xx = P1A assigned as Capture/Compare I/O. P1B, P1C, P1D assigned as Port pins.

CCP1M<3:2> = 11

00 = Single output, P1A modulated. P1B, P1C, P1D assigned as Port pins.

01 = Quad output forward. P1D modulated, P1A active. P1B and P1C inactive.

10 = Dual output. P1A, P1B modulated with dead-time control. P1C, P1D assigned as port pins.

11 = Quad output reverse. P1B modulated, P1C active. P1A and P1D inactive.

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:** ECCP Mode Select bits

0000 = Capture/Compare/PWM off (resets ECCP module)

0001 = Unused (Reserved)

0010 = Compare mode, toggle output on match (CCP1IF bit is set)

0011 = Unused (Reserved)

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 CCP1 output on match (CCP1IF bit is set)

1001 = Compare mode, clear CCP1 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, TMR1 is reset, and an A/D conversion is started if the A/D module is enabled. CCP1 pin is unaffected).

1100 = PWM mode. P1A, P1C active-high; P1B, P1D active-high.

1101 = PWM mode. P1A, P1C active-high; P1B, P1D active-low.

1110 = PWM mode. P1A, P1C active-low; P1B, P1D active-high.

1111 = PWM mode. P1A, P1C active-low; P1B, P1D active-low.

#### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

# PIC16F716

## 7.1 Capture Mode

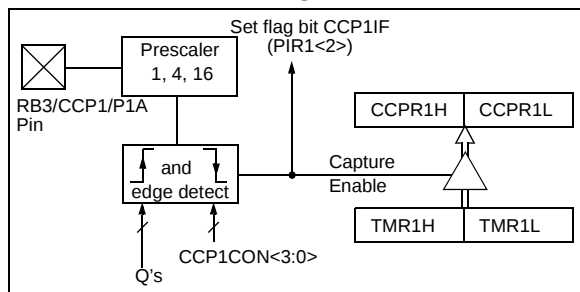
In Capture mode, CCP1H:CCP1L captures the 16-bit value of the TMR1 register when an event occurs on pin RB3/CCP1/P1A. An event is defined as:

- 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 CCP1 is read, the old captured value will be lost.

**Note:** Always reset the ECCP module (CCP1M3:CCP1M0 = '0000') between changing from one capture mode to another. This is necessary to reset the internal capture counter.

**FIGURE 7-1: CAPTURE MODE OPERATION BLOCK DIAGRAM**



### 7.1.1 CCP1 PIN CONFIGURATION

In Capture mode, the RB3/CCP1/P1A pin should be configured as an input by setting the TRISB<3> bit.

**Note:** If the RB3/CCP1/P1A is configured as an output, a write to PORTB can cause a capture condition.

### 7.1.2 TIMER1 MODE SELECTION

Timer1 must be running in Timer mode or Synchronized Counter mode for the ECCP 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 ECCP PRESCALER

There are four prescaler settings, specified by bits CCP1M3:CCP1M0. Whenever the ECCP module is turned off, or the ECCP 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 ECCP module off
MOVLW   NEW_CAPT_PS ;Load the W reg with
                        ;the new prescaler
                        ;mode value and ECCP ON
MOVWF   CCP1CON    ;Load CCP1CON with this
                        ;value
```

## 7.2 Compare Mode

In Compare mode, the 16-bit CCP1 register value is constantly compared against the TMR1 register pair value. When a match occurs, the RB3/CCP1/P1A pin is either:

- Driven High
- Driven Low
- Toggle output (high-to-low or low-to-high)
- 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.

Changing the ECCP mode to clear output on match (CCP1M<3:0> = 1000) presets the CCP1 output latch to the logic 1 level. Changing the ECCP mode to set output on match (CCP1M<3:0> = 1001) presets the CCP1 output latch to the logic 0 level.

## 7.2.1 CCP1 PIN CONFIGURATION

The user must configure the RB3/CCP1/P1A pin as the CCP1 output by clearing the TRISB<3> bit.

**Note:** Clearing the CCP1CON register will force the RB3/CCP1/P1A compare output latch to the default low level. This is not the PORTB I/O data latch.

## 7.2.2 TIMER1 MODE SELECTION

Timer1 must be running in Timer mode or Synchronized Counter mode if the ECCP 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 mode is chosen, the CCP1 pin is not affected. Only a CCP interrupt is generated (if enabled).

## 7.2.4 SPECIAL EVENT TRIGGER

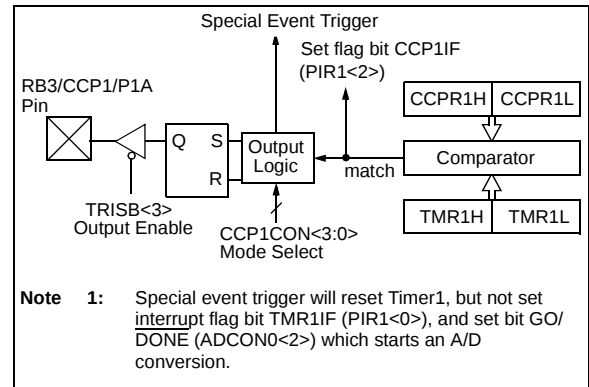
In this mode, an internal hardware trigger is generated which may be used to initiate an action.

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

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

**FIGURE 7-2: COMPARE MODE OPERATION BLOCK DIAGRAM**



**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 |
|---------|---------|-----------------------------------------------------------------------------|-------|---------|---------|---------|--------|--------|--------|-------------------|---------------------------|
| 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 TMR1 register  |       |         |         |         |        |        |        | xxxx xxxx         | uuuu uuuu                 |
| 10h     | T1CON   | —                                                                           | —     | T1CKPS1 | T1CKPS0 | T1OSCEN | T1SYN  | 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 | P1M1                                                                        | P1M0  | DC1B1   | DC1B0   | CCP1M3  | CCP1M2 | CCP1M1 | CCP1M0 | 0000 0000         | 0000 0000                 |
| 86h     | TRISB   | PORTB Data direction register                                               |       |         |         |         |        |        |        | 1111 1111         | 1111 1111                 |
| 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.

# PIC16F716

## 7.3 PWM Mode

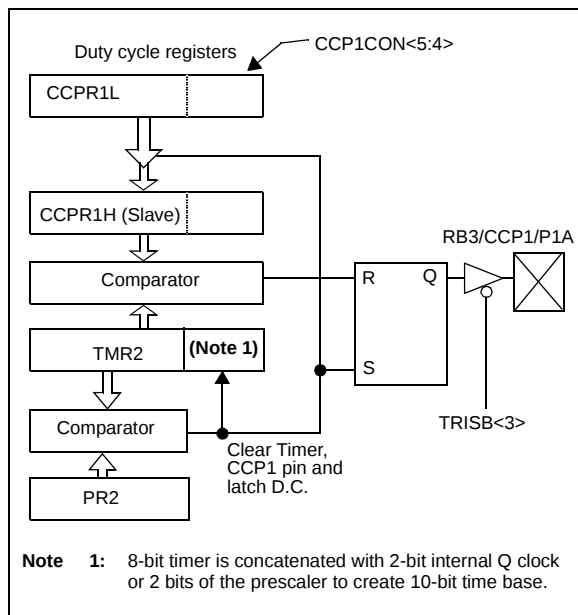
In Pulse Width Modulation (PWM) mode, the CCP1 pin produces up to a 10-bit resolution PWM output. Since the CCP1 pin is multiplexed with the PORTB data latch, the TRISB<3> bit must be cleared to make the CCP1 pin an output.

**Note:** Clearing the CCP1CON register will force the CCP1 PWM output latch to the default low level. This is not the PORTB I/O data latch.

Figure 7-3 shows a simplified block diagram of the CCP module in PWM mode.

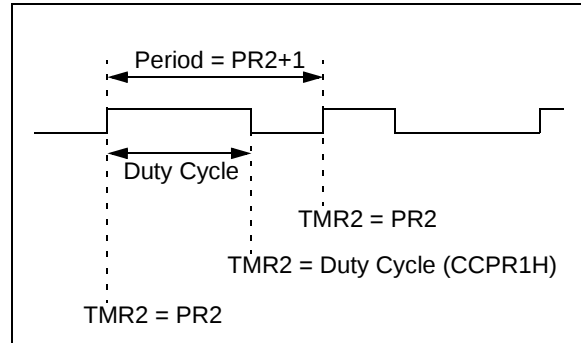
For a step-by-step procedure on how to setup the ECCP module for PWM operation, see **Section 7.3.3 “Set-Up for PWM Operation”**.

**FIGURE 7-3: SIMPLIFIED PWM BLOCK DIAGRAM**



A PWM output (Figure 7-4) 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 7-4: 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:

**EQUATION 7-1:**

$$\text{PWM Period} = [(PR2) + 1] \cdot 4 \cdot T_{osc} \cdot (\text{TMR2 prescale value})$$

PWM frequency is defined as  $1/[\text{PWM period}]$ .

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 “Timer2 Module”**) 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:

### EQUATION 7-2:

$$\text{PWM Duty Cycle} = (\text{CCPR1L:CCP1CON<5:4>} \cdot \text{TOSC} \cdot (\text{TMR2 prescale value}))$$

CCPR1L and CCP1CON<5:4> can be written to at any time, but the duty cycle value is not latched into CCPR1H until 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 is given by the following equation:

### EQUATION 7-3:

$$\text{Max resolution} = \frac{\log\left(\frac{F_{\text{OSC}}}{F_{\text{PWM}}}\right)}{\log(2)} \text{ bits}$$

**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 ECCP 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 TRISB<3> 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         | 6.6       |

**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 |
|---------|---------|-------------------------------------|---------|---------|---------|---------|--------|---------|---------|-------------------|---------------------------|
| 0Bh,8Bh | INTCON  | GIE                                 | PEIE    | TOIE    | INTE    | RBIE    | TOIF   | 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 | P1M1                                | P1M0    | DC1B1   | DC1B0   | CCP1M3  | CCP1M2 | CCP1M1  | CCP1M0  | 0000 0000         | 0000 0000                 |
| 86h     | TRISB   | PORB Data direction register        |         |         |         |         |        |         |         | 1111 1111         | 1111 1111                 |
| 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.

---

### 7.4.1 PWM OUTPUT CONFIGURATIONS

The P1M1:P1M0 bits in the CCP1CON register allows one of four configurations:

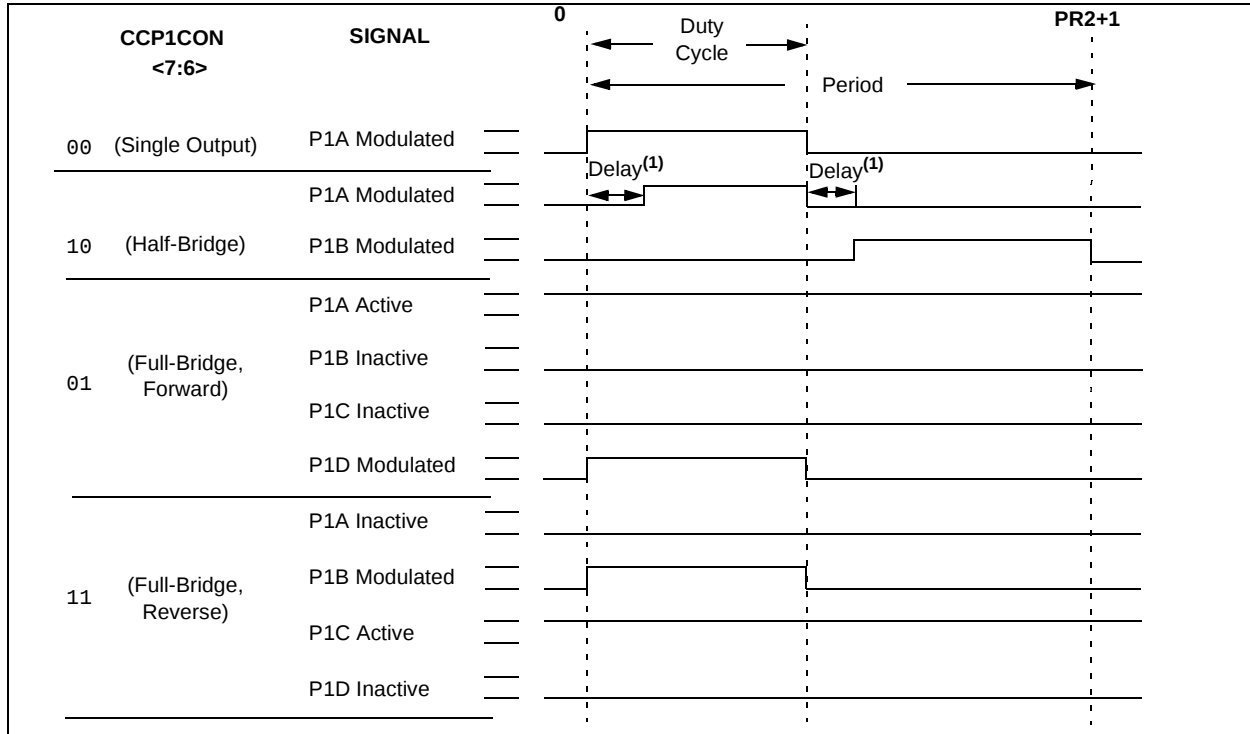
- Single Output
- Half-Bridge Output
- Full-Bridge Output Forward mode
- Full-Bridge Output Reverse mode

The general relationship of the outputs in all configurations is summarized in Figure 7-6.

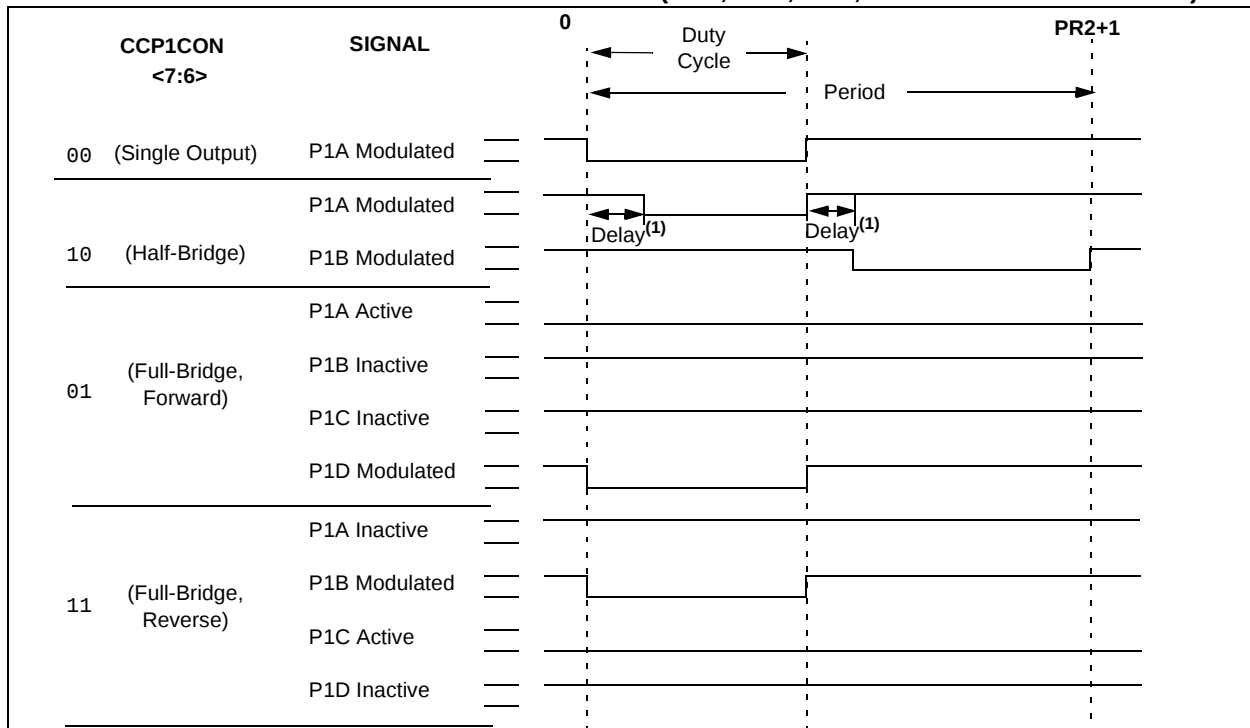
As before, the user must manually configure the appropriate TRISB bits for output.

The diagram illustrates the internal structure of the CCP1 module. On the left, the 'Duty Cycle Registers' block contains 'CCPR1L' and 'CCPR1H (Slave)'. 'CCPR1L' is connected to a 'Comparator' and the 'S' (Set) input of a flip-flop. 'CCPR1H (Slave)' is connected to the 'Q' output of the flip-flop. The flip-flop's 'R' (Reset) input is connected to the 'Clear Timer, set CCP1 pin and latch D.C.' signal. The 'TMR2' register is connected to the 'Comparator' and the 'Q' output of the flip-flop. The 'PR2' register is connected to 'TMR2'. The 'Output Controller' block receives inputs from 'P1M<1:0>' (2 bits), 'CCP1M<3:0>' (4 bits), and 'PWM1CON'. It has four outputs: 'CCP1/P1A', 'P1B', 'P1C', and 'P1D'. Each output is connected to a pin through an inverter and a pull-up resistor. The pins are labeled 'RB3/CCP1/P1A', 'RB5/P1B', 'RB6/P1C', and 'RB7/P1D'. The TRISB register is connected to the pins: 'TRISB<3>' to RB3, 'TRISB<5>' to RB5, 'TRISB<6>' to RB6, and 'TRISB<7>' to RB7. A 'Note 1' points to the 'TMR2' register, stating: 'The 8-bit timer TMR2 register is concatenated with the 2-bit internal Q clock, or 2 bits of the prescaler to create the 10-bit time base.'

**FIGURE 7-6: PWM OUTPUT RELATIONSHIPS (P1A, P1B, P1C, P1D ACTIVE-HIGH STATE)**



**FIGURE 7-7: PWM OUTPUT RELATIONSHIPS (P1A, P1B, P1C, P1D ACTIVE-LOW STATE)**



Relationships:

- Period = 4 \* T<sub>osc</sub> \* (PR2 + 1) \* (TMR2 prescale value)
- Duty Cycle = T<sub>osc</sub> \* (CCPR1L<7:0> : CCP1CON<5:4>) \* (TMR2 prescale value)
- Delay = 4 \* T<sub>osc</sub> \* (PWM1CON<6:0>)

**Note 1:** Dead-band delay is programmed using the PWM1CON register (Section 7.4.4 “Programmable Dead-Band Delay”).

# PIC16F716

**FIGURE 7-8: PWM OUTPUT RELATIONSHIPS (P1A, P1C ACTIVE-HIGH. P1B, P1D ACTIVE-LOW)**

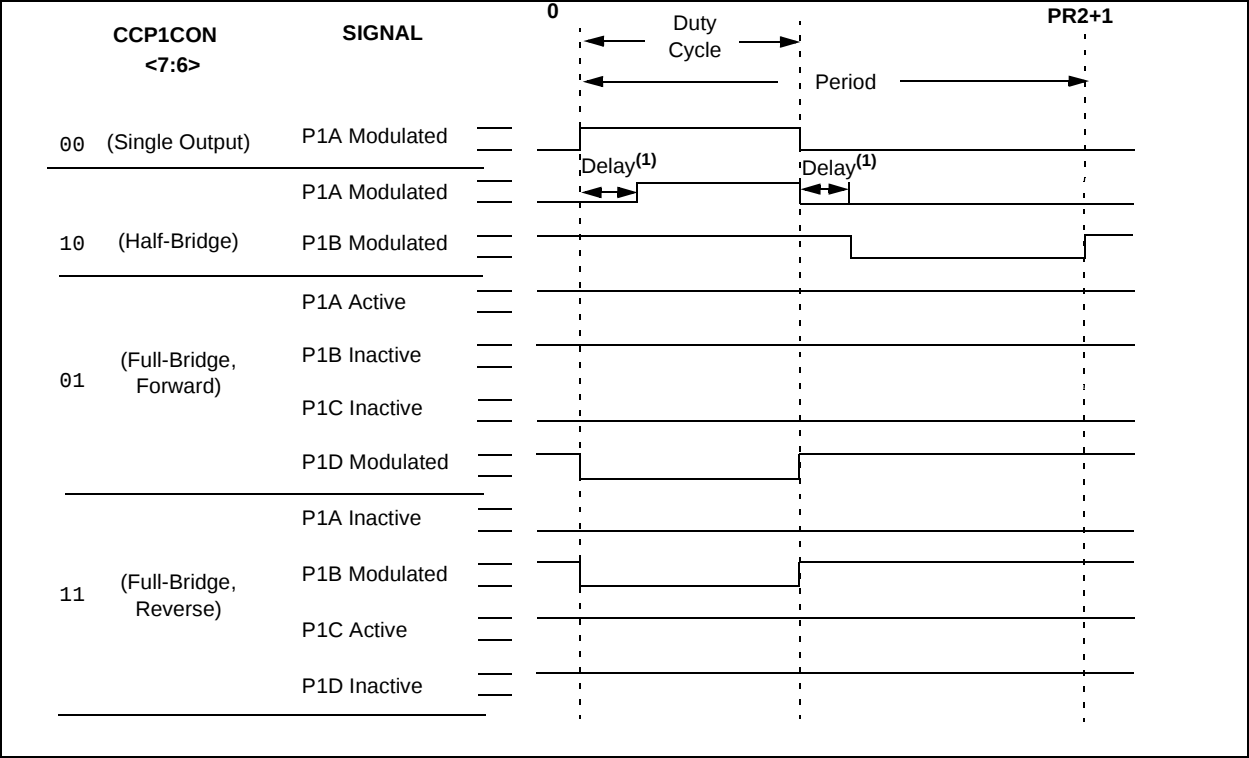
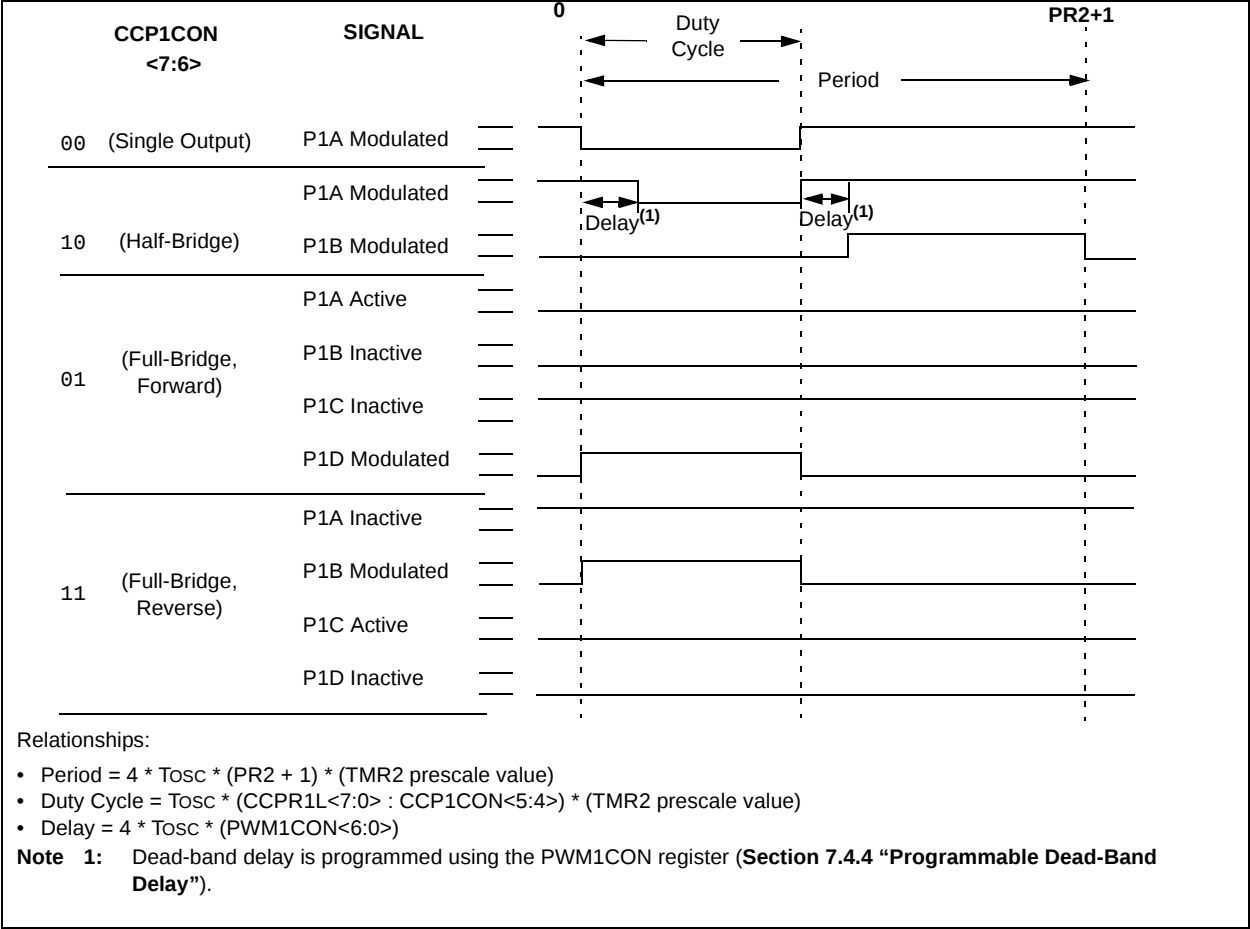


FIGURE 7-9: PWM OUTPUT RELATIONSHIPS (P1A, P1C ACTIVE-LOW. P1B, P1D ACTIVE-HIGH)



# PIC16F716

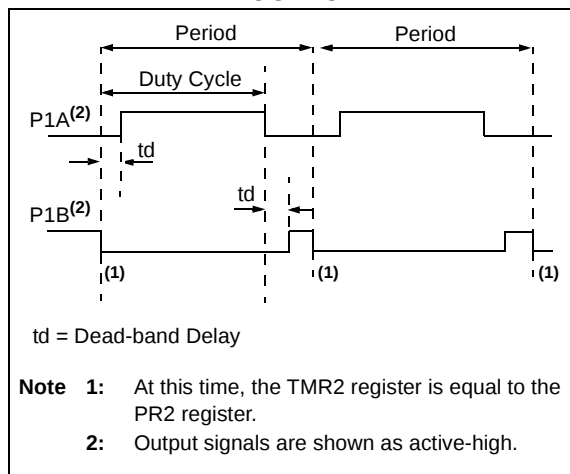
## 7.4.2 HALF-BRIDGE MODE

In the Half-Bridge Output mode, two pins are used as outputs to drive push-pull loads. The PWM output signal is output on the RB3/CCP1/P1A pin, while the complementary PWM output signal is output on the RB5/P1B pin (Figure 7-12). This mode can be used for half-bridge applications, as shown in Figure 7-11 or for full-bridge applications, where four power switches are being modulated with two PWM signals.

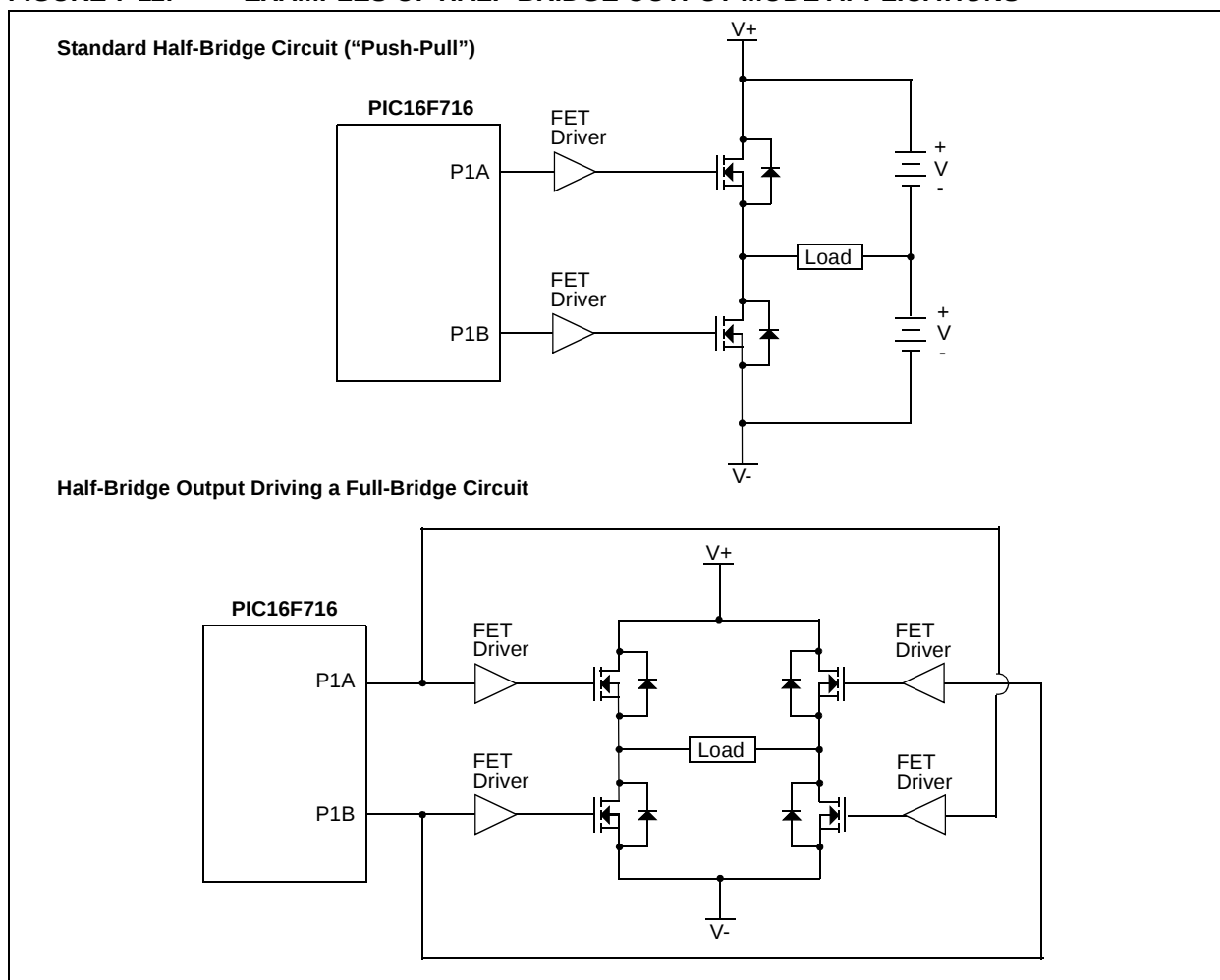
In Half-Bridge Output mode, the programmable dead-band delay can be used to prevent shoot-through current in half-bridge power devices. The value of PWM1CON bits PDC6:PDC0 sets the number of instruction cycles before the output is driven active. If the value is greater than the duty cycle, the corresponding output remains inactive during the entire cycle. See **Section 7.4.4 “Programmable Dead-Band Delay”** for more details of the dead-band delay operations.

Since the P1A and P1B outputs are multiplexed with the PORTB<3> and PORTB<5> data latches, the TRISB<3> and TRISB<5> bits must be cleared to configure P1A and P1B as outputs.

**FIGURE 7-10: HALF-BRIDGE PWM OUTPUT**



**FIGURE 7-11: EXAMPLES OF HALF-BRIDGE OUTPUT MODE APPLICATIONS**

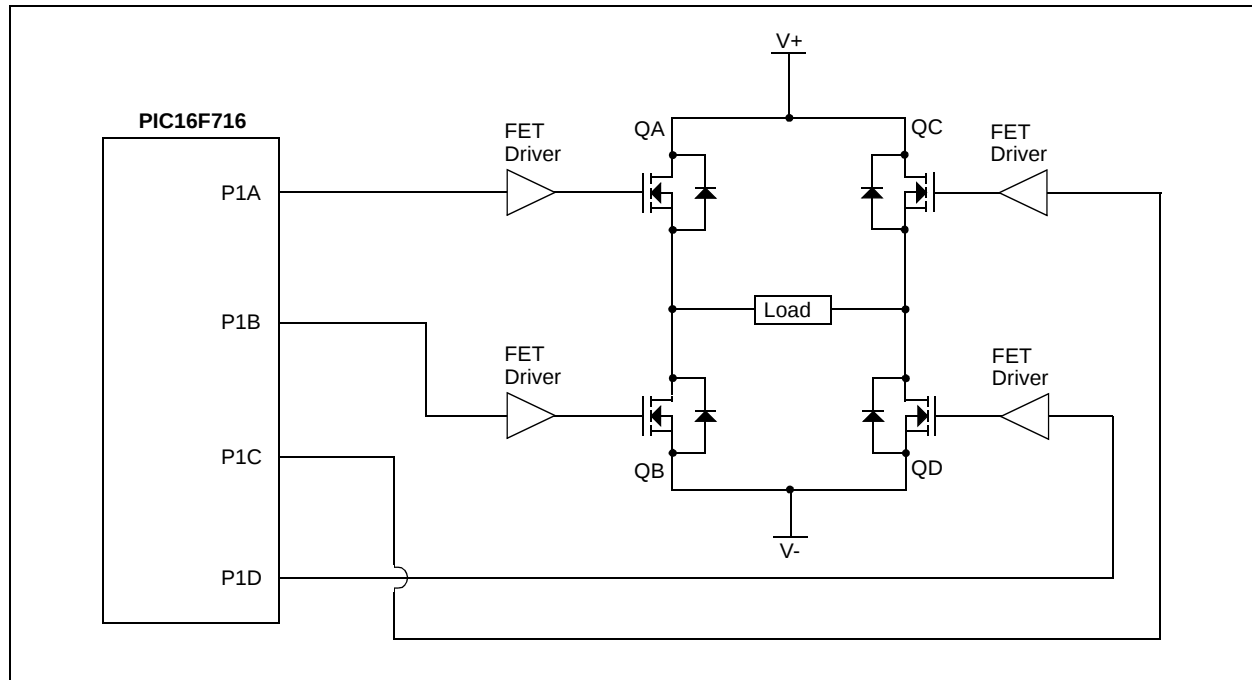


## 7.4.3 FULL-BRIDGE MODE

In Full-Bridge Output mode, four pins are used as outputs; however, only two outputs are active at a time. In the Forward mode, pin RB3/CCP1/P1A is continuously active, and pin RB7/P1D is modulated. In the Reverse mode, RB6/P1C pin is continuously active, and RB5/P1B pin is modulated. These are illustrated in Figure 7-6 through Figure 7-9.

P1A, P1B, P1C and P1D outputs are multiplexed with the PORTB<3> and PORTB<5:7> data latches. The TRISB<3> and TRISB<5:7> bits must be cleared to make the P1A, P1B, P1C, and P1D pins output.

**FIGURE 7-12: EXAMPLE OF FULL-BRIDGE APPLICATION**



### 7.4.3.1 Direction Change in Full-Bridge Mode

In the Full-Bridge Output mode, the P1M1 bit in the CCP1CON register allows the user to control the Forward/Reverse direction. When the application firmware changes this direction control bit, the module will assume the new direction on the next PWM cycle.

Just before the end of the current PWM period, the modulated outputs (P1B and P1D) are placed in their inactive state, while the unmodulated outputs (P1A and P1C) are switched to drive in the opposite direction. This occurs in a time interval of  $(4 \cdot T_{osc} \cdot (\text{Timer2 Prescale value}))$  before the next PWM period begins. The Timer2 prescaler will be either 1, 4 or 16, depending on the value of the T2CKPSx bits (T2CON<1:0>). During the interval from the switch of the unmodulated outputs to the beginning of the next period, the modulated outputs (P1B and P1D) remain inactive. This relationship is shown in Figure 7-13.

**Note:** In the Full-Bridge Output mode, the ECCP module does not provide any dead-band delay. In general, since only one output is modulated at all times, dead-band delay is not required. However, there is a situation where a dead-band delay might be required. This situation occurs when both of the following conditions are true:

1. The direction of the PWM output changes when the duty cycle of the output is at or near 100%.
2. The turn off time of the power switch, including the power device and driver circuit, is greater than the turn on time.

# PIC16F716

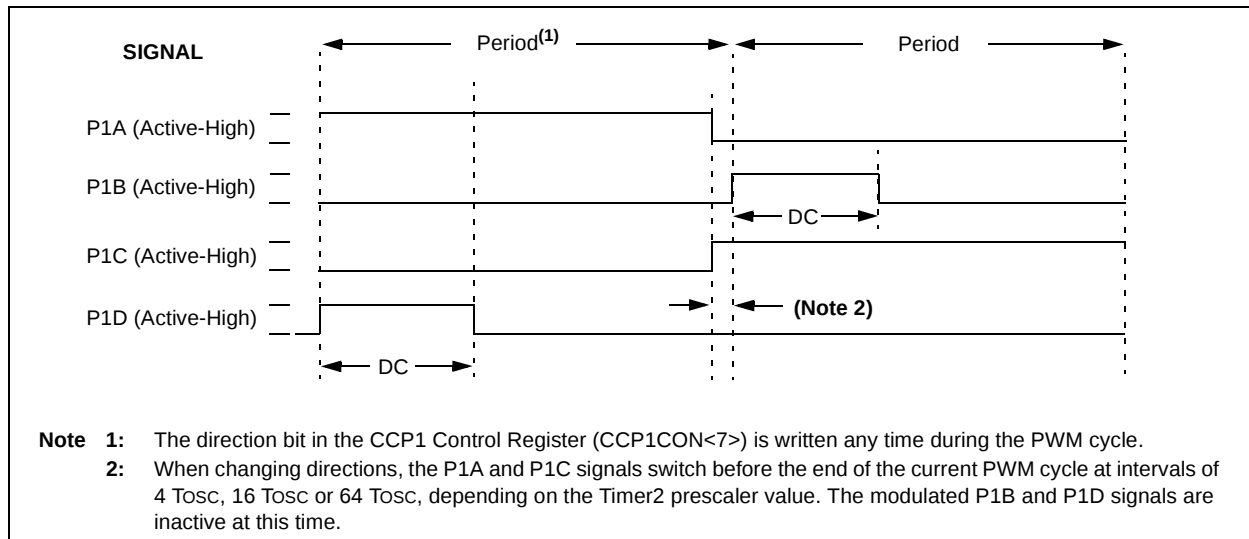
Figure 7-14 shows an example where the PWM direction changes from forward to reverse, at a near 100% duty cycle. At time  $t_1$ , the output P1A and P1D become inactive, while output P1C becomes active. In this example, since the turn-off time of the power devices is longer than the turn-on time, a shoot-through current may flow through power devices QC and QD (see Figure 7-12) for the duration of 't'. The same phenomenon will occur to power devices QA and QB for PWM direction change from reverse to forward.

If changing PWM direction at high duty cycle is required for an application, one of the following requirements must be met:

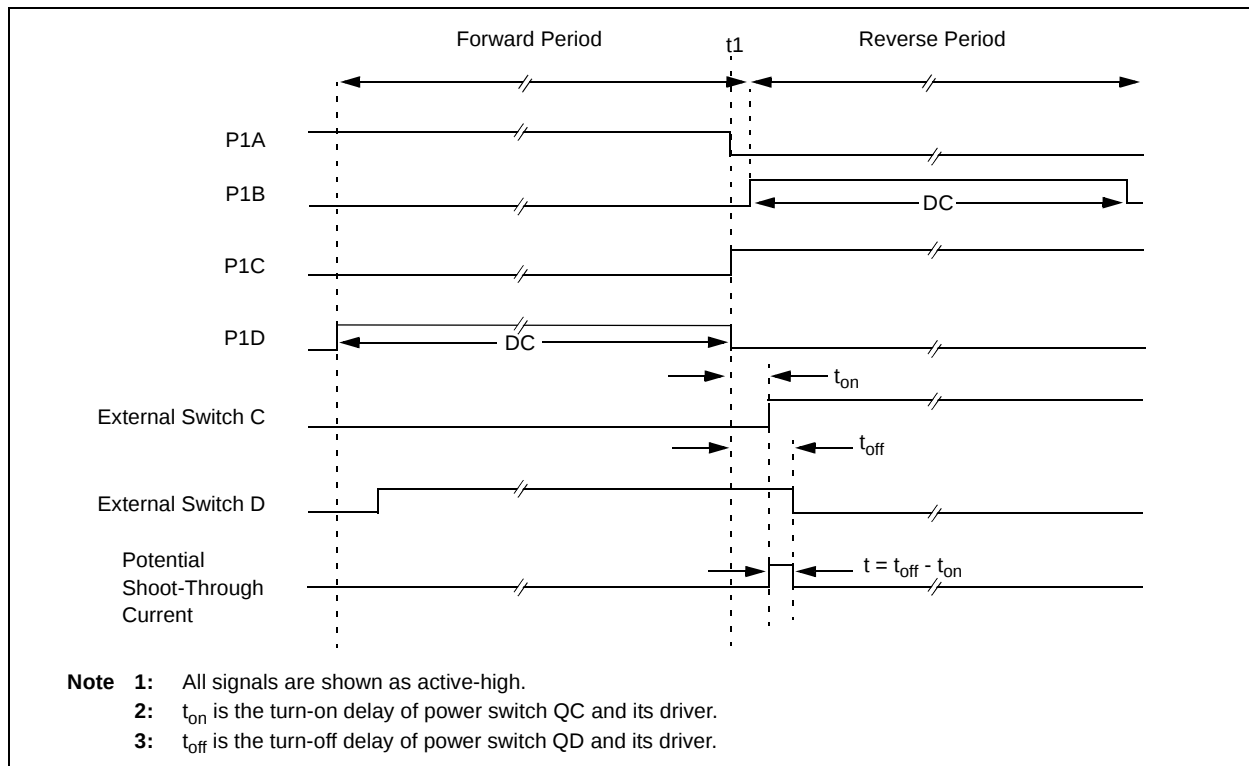
1. Reduce PWM for a PWM period before changing directions.
2. Use switch drivers that can drive the switches off faster than they can drive them on.

Other options to prevent shoot-through current may exist.

**FIGURE 7-13: PWM DIRECTION CHANGE**



**FIGURE 7-14: PWM DIRECTION CHANGE AT NEAR 100% DUTY CYCLE**



## 7.4.4 PROGRAMMABLE DEAD-BAND DELAY

In half-bridge applications where all power switches are modulated at the PWM frequency at all times, the power switches normally require more time to turn off than to turn on. If both the upper and lower power switches are switched at the same time (one turned on, and the other turned off), both switches may be on for a short period of time until one switch completely turns off. During this brief interval, a very high current (*shoot-through current*) may flow through both power switches, shorting the bridge supply. To avoid this potentially destructive shoot-through current from flowing during switching, turning on either of the power switches is normally delayed to allow the other switch to completely turn off.

In the Half-Bridge Output mode, a digitally programmable dead-band delay is available to avoid shoot-through current from destroying the bridge power switches. The delay occurs at the signal transition from the non-active state to the active state. See Figure 7-10 for illustration. The lower seven bits of the PWM1CON register (Register 7-2) sets the delay period in terms of microcontroller instruction cycles (Tcy or 4 Tosc).

## 7.4.5 ENHANCED PWM AUTO-SHUTDOWN

When the ECCP is programmed for any of the enhanced PWM modes, the active output pins may be configured for auto-shutdown. Auto-shutdown immediately places the enhanced PWM output pins into a defined shutdown state when a shutdown event occurs.

A shutdown event can be caused by a logic low level on either or both of the RB0/INT/ECCPAS2 or RB4/ECCPAS0 pins. The auto-shutdown feature can be disabled by not selecting any auto-shutdown sources. The auto-shutdown sources to be used are selected using the ECCPAS2 and ECCPAS0 bits (ECCPAS<6> and ECCPAS<4>).

When a shutdown occurs, the output pins are asynchronously placed in their shutdown states, specified by the PSSAC1:PSSAC0 and PSSBD1:PSSBD0 bits (ECCPAS<3:0>). Each pin pair (P1A/P1C and P1B/P1D) may be set to drive high, drive low, or be tri-stated (not driving). The ECCPASE bit (ECCPAS<7>) is also set to hold the enhanced PWM outputs in their shutdown states.

The ECCPASE bit is set by hardware when a shutdown event occurs. If automatic restarts are not enabled, the ECCPASE bit must be cleared by firmware when the cause of the shutdown clears. If automatic restarts are enabled, the ECCPASE bit is automatically cleared when the cause of the auto-shutdown has cleared.

If the ECCPASE bit is set when a PWM period begins, the PWM outputs remain in their shutdown state for that entire PWM period. When the ECCPASE bit is cleared, the PWM outputs will return to normal operation at the beginning of the next PWM period.

|                                                                                           |
|-------------------------------------------------------------------------------------------|
| <b>Note:</b> Writing to the ECCPASE bit is disabled while a shutdown condition is active. |
|-------------------------------------------------------------------------------------------|

# PIC16F716

## REGISTER 7-2: PWM1CON: PWM CONFIGURATION REGISTER (ADDRESS: 18h)

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|-------|-------|-------|-------|-------|-------|-------|-------|
| PRSEN | PDC6  | PDC5  | PDC4  | PDC3  | PDC2  | PDC1  | PDC0  |
| bit 7 |       |       |       |       |       |       | bit 0 |

- bit 7 **PRSEN:** PWM Restart Enable bit  
 1 = Upon auto-shutdown, the ECCPASE bit clears automatically once the shutdown event goes away; the PWM restarts automatically.  
 0 = Upon auto-shutdown, ECCPASE must be cleared in firmware to restart the PWM.
- bit 6-0 **PDC<6:0>:** PWM Delay Count bits  
 Number of Fosc/4 (4\*Tosc) cycles between the scheduled time when a PWM signal **should** transition active, and the **actual** time it transitions active.

### Legend:

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

## REGISTER 7-3: ECCPAS – ENHANCED CCP AUTO SHUT DOWN REGISTER (ADDRESS: 19h)

| R/W-0   | R/W-0   | U-0 | R/W-0   | R/W-0  | R/W-0  | R/W-0  | R/W-0  |
|---------|---------|-----|---------|--------|--------|--------|--------|
| ECCPASE | ECCPAS2 | —   | ECCPAS0 | PSSAC1 | PSSAC0 | PSSBD1 | PSSBD0 |
| bit 7   |         |     |         |        |        |        | bit 0  |

- bit 7 **ECCPASE:** ECCP Auto-Shutdown Event Status bit  
 1 = A shutdown event has occurred. Must be reset in firmware to re-enable ECCP if PRSEN = 0  
 0 = ECCP outputs enabled, no shutdown event
- bit 6 **ECCPAS2:** ECCP Auto-Shutdown bit 2  
 1 = RB0 (INT) pin low level ('0') causes shutdown  
 0 = RB0 (INT) pin has no effect on ECCP
- bit 5 **Unimplemented:** Read as '0'
- bit 4 **ECCPAS0:** ECCP Auto-Shutdown bit '0'  
 1 = RB4 pin low level ('0') causes shutdown  
 0 = RB4 pin has no effect on ECCP
- bit 3-2 **PSSAC<1:0>:** Pin P1A and P1C Shutdown State Control  
 00 = Drive Pins P1A and P1C to '0'  
 01 = Drive Pins P1A and P1C to '1'  
 1x = Pins P1A and P1C tri-state
- bit 1-0 **PSSBD<1:0>:** Pin P1B and P1D Shutdown State Control  
 00 = Drive Pins P1B and P1D to '0'  
 01 = Drive Pins P1B and P1D to '1'  
 1x = Pins P1B and P1D tri-state

### Legend:

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

## 7.4.5.1 Auto-Shutdown and Automatic Restart

The auto-shutdown feature can be configured to allow automatic restarts of the module following a shutdown event. This is enabled by setting the PRSEN bit of the PWM1CON register (PWM1CON<7>).

In Shutdown mode with PRSEN = 1 (PWM1CON <7>), the ECCPASE bit will remain set for as long as the cause of the shutdown continues. When the shutdown condition clears, the ECCPASE bit is cleared. If PRSEN = 0 (Figure 7-16), once a shutdown condition occurs, the ECCPASE bit will remain set until it is cleared by firmware. Once ECCPASE is cleared, the enhanced PWM will resume at the beginning of the next PWM period.

**Note:** Writing to the ECCPASE bit is disabled while a shutdown condition is active.

The ECCPASE bit cannot be cleared as long as the cause of the shutdown persists.

The Auto-shutdown mode can be forced by writing a '1' to the ECCPASE bit.

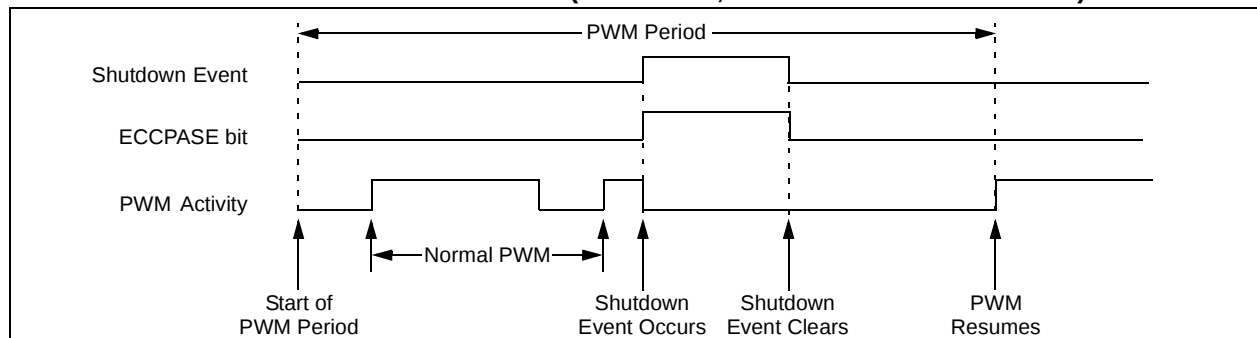
## 7.4.6 START-UP CONSIDERATIONS

When the ECCP module is used in the PWM mode, the application hardware must use the proper external pull-up and/or pull-down resistors on the PWM output pins. When the microcontroller is released from Reset, all of the I/O pins are in the high-impedance state. The external circuits must keep the power switch devices in the off state, until the microcontroller drives the I/O pins with the proper signal levels, or activates the PWM output(s).

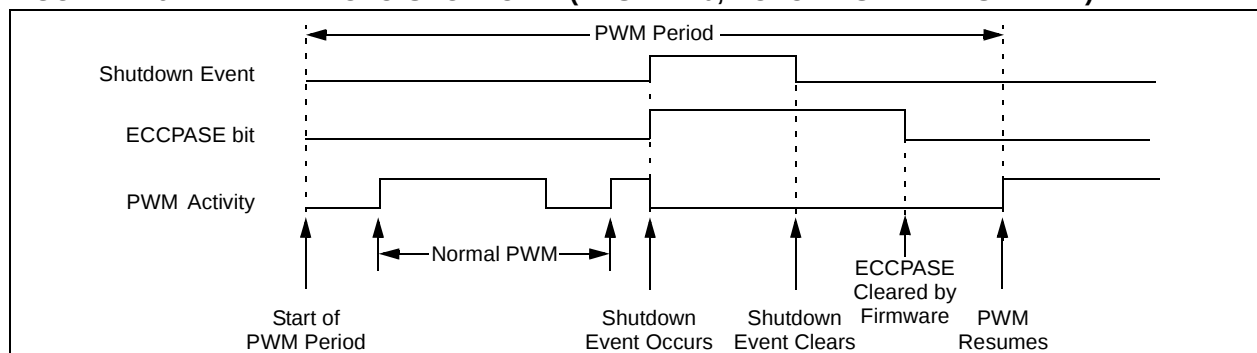
The CCP1M1:CCP1M0 bits (CCP1CON<1:0>) allow the user to choose whether the PWM output signals are active-high or active-low for each pair of PWM output pins (P1A/P1C and P1B/P1D). The PWM output polarities must be selected before the PWM pins are configured as outputs. Changing the polarity configuration while the PWM pins are configured as outputs is not recommended since it may result in damage to the application circuits.

The P1A, P1B, P1C and P1D output latches may not be in the proper states when the PWM module is initialized. Enabling the PWM pins for output at the same time as the ECCP module may cause damage to the application circuit. The ECCP module must be enabled in the proper Output mode and complete a full PWM cycle before configuring the PWM pins as outputs. The completion of a full PWM cycle is indicated by the TMR2IF bit being set as the second PWM period begins.

**FIGURE 7-15: PWM AUTO-SHUTDOWN (PRSEN = 1, AUTO-RESTART ENABLED)**



**FIGURE 7-16: PWM AUTO-SHUTDOWN (PRSEN = 0, AUTO-RESTART DISABLED)**



# PIC16F716

## 7.4.7 SETUP FOR PWM OPERATION

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

1. Configure the PWM pins P1A and P1B (and P1C and P1D, if used) as inputs by setting the corresponding TRISB bits.
2. Set the PWM period by loading the PR2 register.
3. Configure the ECCP module for the desired PWM mode and configuration by loading the CCP1CON register with the appropriate values:
  - Select one of the available output configurations and direction with the P1M1:P1M0 bits.
  - Select the polarities of the PWM output signals with the CCP1M3:CCP1M0 bits.
4. Set the PWM duty cycle by loading the CCPR1L register and CCP1CON<5:4> bits.
5. For Half-Bridge Output mode, set the dead-band delay by loading PWM1CON<6:0> with the appropriate value.
6. If auto-shutdown operation is required, load the ECCPAS register.
  - Select the auto-shutdown sources using the ECCPAS<2> AND ECCPAS<0> bits.
  - Select the shutdown states of the PWM output pins using PSSAC1:PSSAC0 and PSSBD1:PSSBD0 bits.
  - Set the ECCPASE bit (ECCPAS<7>).
7. If auto-restart operation is required, set the PRSEN bit (PWM1CON<7>).

## 8. Configure and start TMR2:

- Clear the TMR2 interrupt flag bit by clearing the TMR2IF bit (PIR1<1>).
  - Set the TMR2 prescale value by loading the T2CKPSx bits (T2CON<1:0>).
  - Enable Timer2 by setting the TMR2ON bit (T2CON<2>).
9. Enable PWM outputs after a new PWM cycle has started:
- Wait until TMR2 overflows (TMR2IF bit is set).
  - Enable the CCP1/P1A, P1B, P1C and/or P1D pin outputs by clearing the respective TRISB bits.
  - Clear the ECCPASE bit (ECCPAS<7>).

See the previous section for additional details.

## 7.4.8 EFFECTS OF A RESET

Both Power-on and subsequent Resets will force all ports to Input mode and the ECCP registers to their Reset states.

This forces the Enhanced CCP module to reset to a state compatible with the standard ECCP module.

**TABLE 7-5: REGISTERS ASSOCIATED WITH ENHANCED 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 |
|---------|---------|--------------------------------------------------|---------|---------|---------|---------|--------|---------|---------|-------------------|---------------------------|
| 0Bh     | INTCON  | GIE                                              | PEIE    | TMR0IE  | INT0IE  | RBIE    | TMR0IF | INT0IF  | RBIF    | 0000 000x         | 0000 000u                 |
| 0Ch     | PIR1    | —                                                | ADIF    | —       | —       | —       | CCP1IF | TMR2IF  | TMR1IF  | -0-- -000         | -0-- -000                 |
| 8Ch     | PIE1    | —                                                | ADIE    | —       | —       | —       | CCP1IE | TMR2IE  | TMR1IE  | -0-- --00         | -0-- --00                 |
| 11h     | TMR2    | Timer2 Module Register                           |         |         |         |         |        |         |         | 0000 0000         | 0000 0000                 |
| 92h     | PR2     | Timer2 Module Period Register                    |         |         |         |         |        |         |         | 1111 1111         | 1111 1111                 |
| 12h     | T2CON   | —                                                | TOUTPS3 | TOUTPS2 | TOUTPS1 | TOUTPS0 | TMR2ON | T2CKPS1 | T2CKPS0 | -000 0000         | -000 0000                 |
| 86h     | TRISB   | PORTB Data Direction Register                    |         |         |         |         |        |         |         | 1111 1111         | 1111 1111                 |
| 16h     | CCPR1H  | Enhanced Capture/Compare/PWM Register1 High Byte |         |         |         |         |        |         |         | xxxx xxxx         | uuuu uuuu                 |
| 15h     | CCPR1L  | Enhanced Capture/Compare/PWM Register1 Low Byte  |         |         |         |         |        |         |         | xxxx xxxx         | uuuu uuuu                 |
| 17h     | CCP1CON | P1M1                                             | P1M0    | DC1B1   | DC1B0   | CCP1M3  | CCP1M2 | CCP1M1  | CCP1M0  | 0000 0000         | 0000 0000                 |
| 19h     | ECCPAS  | ECCPASE                                          | ECCPAS2 | —       | ECCPAS0 | PSSAC1  | PSSAC0 | PSSBD1  | PSSBD0  | 00-0 0000         | 00-0 0000                 |
| 18h     | PWM1CON | PRSEN                                            | PDC6    | PDC5    | PDC4    | PDC3    | PDC2   | PDC1    | PDC0    | 0000 0000         | 0000 0000                 |

**Legend:** x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by the ECCP module in enhanced PWM mode.

## 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/VREF 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 Register 8-1, controls the operation of the A/D module. The ADCON1 register, shown in Register 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.

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

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0   | R/W-0 | R/W-0 |
|-------|-------|-------|-------|-------|---------|-------|-------|
| ADCS1 | ADCS0 | CHS2  | CHS1  | CHS0  | GO/DONE | —     | ADON  |
| bit 7 |       |       |       |       |         |       | bit 0 |

|         |                                                                                                                                                                                                                                                                   |
|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| bit 7-6 | <b>ADCS1:ADCS0:</b> A/D Conversion Clock Select bits<br>00 = $F_{osc}/2$<br>01 = $F_{osc}/8$<br>10 = $F_{osc}/32$<br>11 = FRC (Clock derived from the internal ADC RC oscillator)                                                                                 |
| bit 5-3 | <b>CHS2:CHS0:</b> Analog Channel Select bits<br>000 = channel 0, (RA0/AN0)<br>001 = channel 1, (RA1/AN1)<br>010 = channel 2, (RA2/AN2)<br>011 = channel 3, (RA3/AN3)<br>1xx = reserved, do not use                                                                |
| bit 2   | <b>GO/DONE:</b> A/D Conversion Status bit<br>If ADON = 1<br>1 = A/D conversion in progress (Setting this bit starts the A/D conversion)<br>0 = A/D conversion not in progress (This bit is automatically cleared by hardware when the A/D conversion is complete) |
| bit 1   | <b>Reserved:</b> Maintain this bit as '0'                                                                                                                                                                                                                         |
| bit 0   | <b>ADON:</b> A/D On bit<br>1 = A/D converter module is operating<br>0 = A/D converter module is shutoff and consumes no operating current                                                                                                                         |

#### Legend:

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

# PIC16F716

## REGISTER 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 |
| bit 7 |     |     |     |     | bit 0 |       |       |

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

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

| PCFG2:PCFG0 | AN3<br>RA3 | AN2<br>RA2 | AN1<br>RA1 | AN0<br>RA0 | VREF |
|-------------|------------|------------|------------|------------|------|
| 0x0         | A          | A          | A          | A          | VDD  |
| 0x1         | VREF       | A          | A          | A          | RA3  |
| 100         | A          | D          | A          | A          | VDD  |
| 101         | VREF       | D          | A          | A          | RA3  |
| 11x         | D          | D          | D          | D          | VDD  |

**Legend:** A = Analog input, D = Digital I/O

### Legend:

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

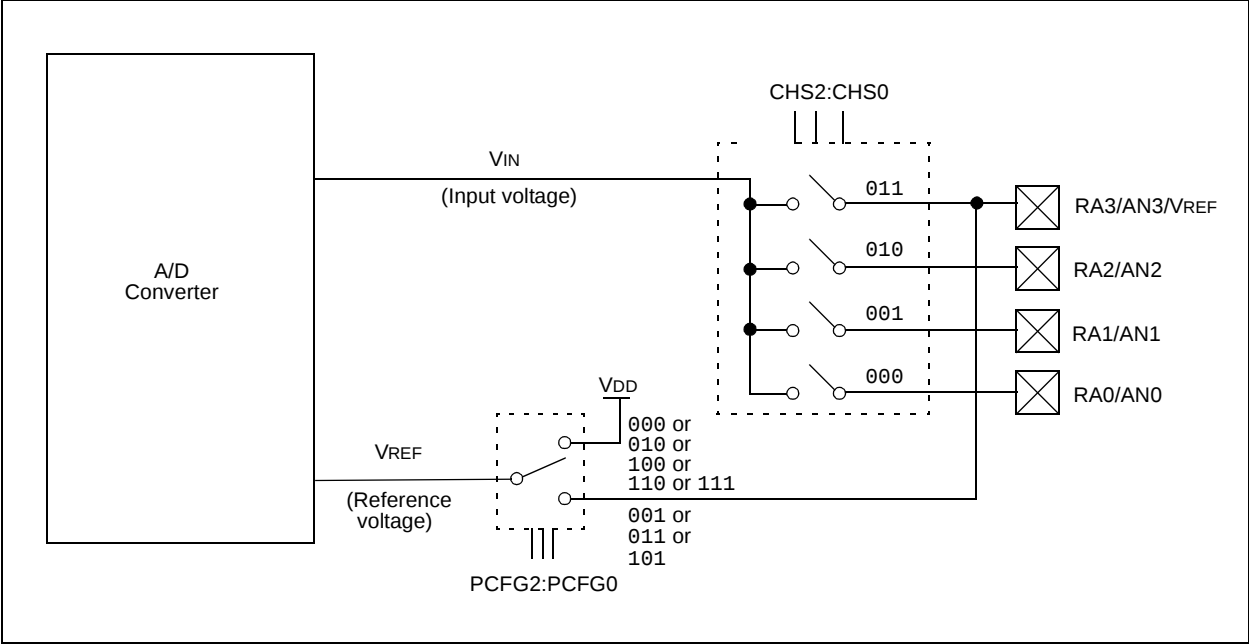
The ADRES register contains the result of the A/D conversion. When the A/D conversion is complete, the result is loaded into the ADRES register, the GO/DONE bit (ADCON0<2>) is cleared and the A/D interrupt flag bit ADIF is set. The block diagram of the A/D module is shown in Figure 8-1.

The value that is in the ADRES register is not modified for any 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 "A/D Acquisition Requirements"**. 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:

- 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)
- Configure A/D interrupt (if desired):
  - Clear ADIF bit
  - Set ADIE bit
  - Set GIE bit
- Wait the required acquisition time.
- Start conversion:
  - Set GO/DONE bit (ADCON0)
- Wait for A/D conversion to complete, by either:
  - Polling for the GO/DONE bit to be cleared
 OR
  - Waiting for the A/D interrupt
- Read A/D Result register (ADRES), clear bit ADIF if required.
- For the next conversion, go to step 1 or step 2 as required. The A/D conversion time per bit is defined as TAD. A minimum wait of 2TAD is required before next acquisition starts.

FIGURE 8-1: A/D BLOCK DIAGRAM



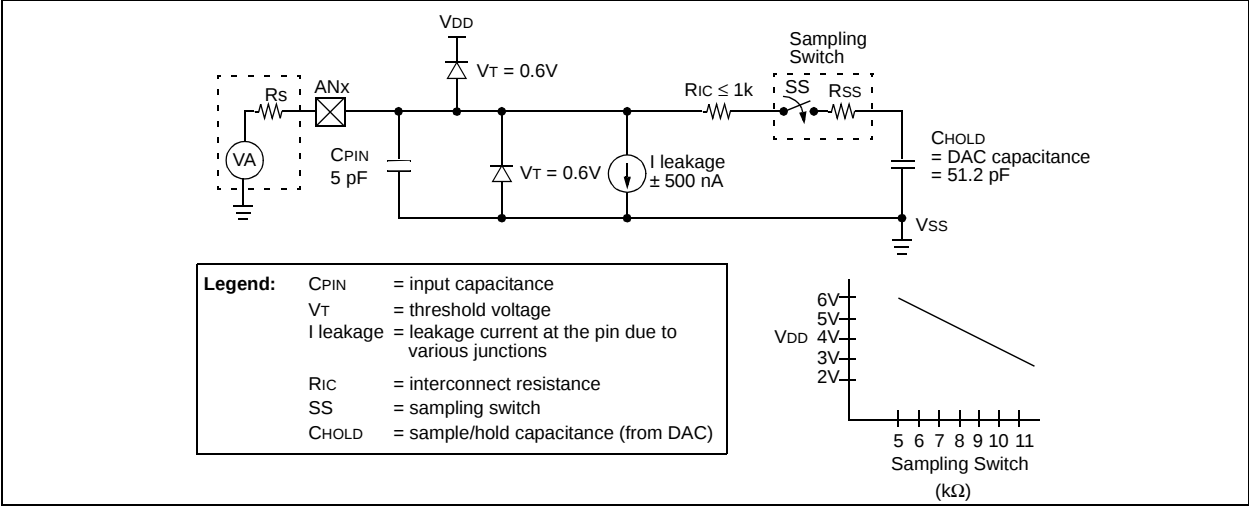
8.1 A/D Acquisition Requirements

For the A/D converter to meet its specified accuracy, the charge holding capacitor (CHOLD) must be allowed to fully charge to the input channel voltage level. The analog input model is shown in Figure 8-2. The source impedance ( $R_s$ ) and the internal sampling switch ( $R_{ss}$ ) impedance directly affect the time required to charge the capacitor CHOLD. 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. 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 8-2: ANALOG INPUT MODEL



# PIC16F716

## 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:

- 2 TOSC
- 8 TOSC
- 32 TOSC
- 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  $\mu$ s.

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.

**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                 |
| 2 TOSC                       | 00          | 100 ns <sup>(2)</sup>           | 400 ns <sup>(2)</sup>           | 1.6 $\mu$ s                     | 6 $\mu$ s                  |
| 8 TOSC                       | 01          | 400 ns <sup>(2)</sup>           | 1.6 $\mu$ s                     | 6.4 $\mu$ s                     | 24 $\mu$ s <sup>(3)</sup>  |
| 32 TOSC                      | 10          | 1.6 $\mu$ s                     | 6.4 $\mu$ s                     | 25.6 $\mu$ s <sup>(3)</sup>     | 96 $\mu$ s <sup>(3)</sup>  |
| RC                           | 11          | 2-6 $\mu$ s <sup>(1), (4)</sup> | 2-6 $\mu$ s <sup>(1), (4)</sup> | 2-6 $\mu$ s <sup>(1), (4)</sup> | 2-6 $\mu$ s <sup>(1)</sup> |

Legend: Shaded cells are outside of recommended range.

**Note 1:** The RC source has a typical  $T_{AD}$  time of 4  $\mu$ s.

**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.

## 8.4 A/D Conversions

**Note:** The GO/DONE bit should **NOT** be set in the same instruction that turns on the A/D.

## 8.5 Use of the ECCP Trigger

An A/D conversion can be started by the “special event trigger” of the ECCP 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/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/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  | —                   | —     | — <sup>(1)</sup> | RA4                           | RA3   | RA2     | RA1              | RA0    | --xx 0000         | --uu uuuu                 |
| 0Bh,8Bh | INTCON | GIE                 | PEIE  | TOIE             | INTE                          | RBIE  | TOIF    | 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/DONE | — <sup>(1)</sup> | ADON   | 0000 0000         | 0000 0000                 |
| 85h     | TRISA  | —                   | —     | — <sup>(1)</sup> | PORTA Data Direction Register |       |         |                  |        |                   | --11 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 bit, do not use.

# PIC16F716

---

NOTES:

## 9.0 SPECIAL FEATURES OF THE CPU

The PIC16F716 device has 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™)

The PIC16F716 device has 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 configuration memory space (2000h – 3FFFh), which can be accessed only during programming.

# PIC16F716

## REGISTER 9-1: CONFIGURATION WORD

|                        |   |   |   |   |   |      |       |   |   |                                  |      |       |       |
|------------------------|---|---|---|---|---|------|-------|---|---|----------------------------------|------|-------|-------|
| $\overline{\text{CP}}$ | — | — | — | — | — | BORV | BOREN | — | — | $\overline{\text{PWRT}}\text{E}$ | WDTE | FOSC1 | FOSC0 |
| bit 13                 |   |   |   |   |   |      |       |   |   |                                  |      |       | bit 0 |

bit 13  **$\overline{\text{CP}}$** : Flash Program Memory Code Protection bit  
1 = Code protection off  
0 = All program memory code protected

bit 12-8 **Unimplemented**: Read as '1'

bit 7 **BORV**: Brown-out Reset Voltage bit  
1 = VBOR set to 4.0V  
0 = VBOR set to 2.5V

bit 6 **BOREN**: Brown-out Reset Enable bit<sup>(1)</sup>  
1 = BOR enabled  
0 = BOR disabled

bit 5-4 **Unimplemented**: Read as '1'

bit 3  **$\overline{\text{PWRT}}\text{E}$** : Power-up Timer Enable bit <sup>(1)</sup>  
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 does not automatically enable Power-up Timer ( $\overline{\text{PWRT}}\text{E}$ ).

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

## 9.2 Oscillator Configurations

### 9.2.1 OSCILLATOR TYPES

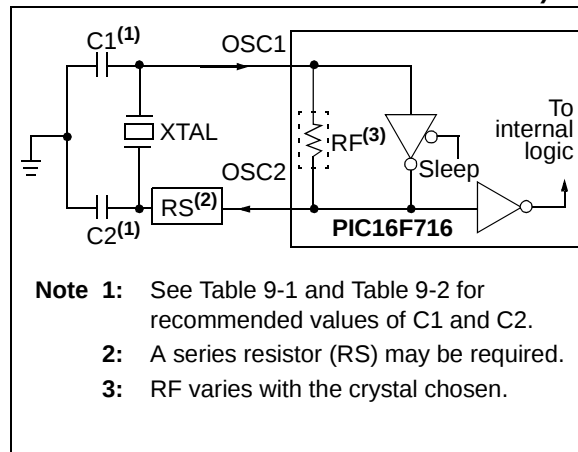
The PIC16F716 can be operated in four different oscillator modes. The user can program two configuration bits (FOSC1 and FOSC0) to select one of these four modes:

- LP - Low-power Crystal
- XT - Crystal/Resonator
- HS - High-speed Crystal/Resonator
- RC - Resistor/Capacitor

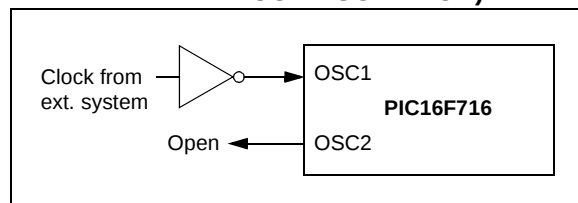
### 9.2.2 CRYSTAL OSCILLATOR/CERAMIC RESONATORS

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 9-1). The PIC16F716 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 9-2).

**FIGURE 9-1: CRYSTAL/CERAMIC RESONATOR OPERATION (HS, XT OR LP OSC CONFIGURATION)**



**FIGURE 9-2: EXTERNAL CLOCK INPUT OPERATION (HS, XT OR LP OSC CONFIGURATION)**



**TABLE 9-1: CERAMIC RESONATORS**

| Ranges Tested: |          |           |           |
|----------------|----------|-----------|-----------|
| Mode           | Freq     | OSC1 (C1) | OSC2 (C2) |
| XT             | 455 kHz  | 68-100 pF | 68-100 pF |
|                | 2.0 MHz  | 15-68 pF  | 15-68 pF  |
| HS             | 4.0 MHz  | 10-68 pF  | 10-68 pF  |
|                | 8.0 MHz  | 15-68 pF  | 15-68 pF  |
|                | 16.0 MHz | 10-22 pF  | 10-22 pF  |

**Note 1:** 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       | 15-33 pF      | 15-33 pF      |
|          | 200 kHz      | 5-10 pF       | 5-10 pF       |
| XT       | 200 kHz      | 47-68 pF      | 47-68 pF      |
|          | 1 MHz        | 15-33 pF      | 15-33 pF      |
|          | 4 MHz        | 15-33 pF      | 15-33 pF      |
| HS       | 4 MHz        | 15-33 pF      | 15-33 pF      |
|          | 8 MHz        | 15-33 pF      | 15-33 pF      |
|          | 20 MHz       | 15-33 pF      | 15-33 pF      |

**Note 1:** These values are for design guidance only. See notes at bottom of page.

**Note 1:** Higher capacitance increases the stability of the 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.

**3:** RS may be required to avoid overdriving crystals with low drive level specification.

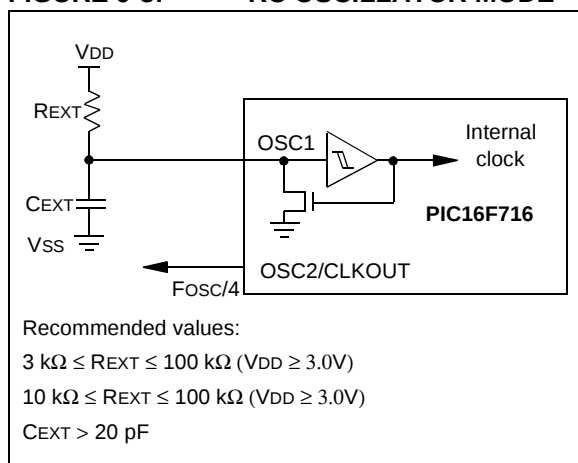
**4:** When using an external clock for the OSC1 input, loading of the OSC2 pin must be kept to a minimum by leaving the OSC2 pin unconnected.

# PIC16F716

## 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<sub>EXT</sub>) and capacitor (C<sub>EXT</sub>) 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<sub>EXT</sub> values. The user also needs to take into account variation due to tolerance of external R and C components used. Figure 9-3 shows how the R/C combination is connected to the PIC16F716.

**FIGURE 9-3: RC OSCILLATOR MODE**



## 9.3 Reset

The PIC16F716 differentiates between various kinds of Reset:

- Power-on Reset (POR)
- $\overline{\text{MCLR}}$  Reset during normal operation
- $\overline{\text{MCLR}}$  Reset during Sleep
- WDT Reset (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  $\overline{\text{MCLR}}$  and WDT Reset, on  $\overline{\text{MCLR}}$  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  $\overline{\text{TO}}$  and  $\overline{\text{PD}}$  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 9-5.

The PICmicro<sup>®</sup> microcontrollers have an  $\overline{\text{MCLR}}$  noise filter in the  $\overline{\text{MCLR}}$  Reset path. The filter will detect and ignore small pulses.

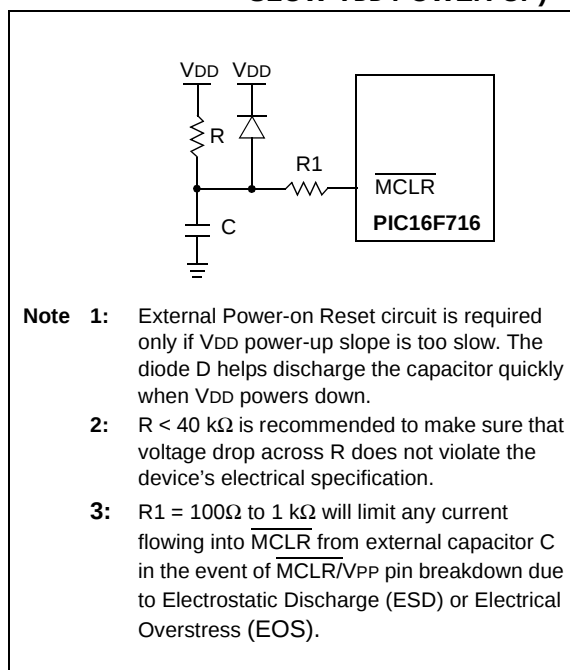
It should be noted that a WDT Reset does not drive the  $\overline{\text{MCLR}}$  pin low.

## 9.4 Power-On Reset (POR)

A Power-on Reset pulse is generated on-chip when V<sub>DD</sub> rise is detected. To take advantage of the POR, just tie the  $\overline{\text{MCLR}}$  pin directly (or through a resistor) to V<sub>DD</sub>. This will eliminate external RC components usually needed to create a Power-on Reset. A maximum rise time for V<sub>DD</sub> is specified (parameter D004). For a slow rise time, see Figure 9-4.

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 9-4: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW V<sub>DD</sub> POWER-UP)**



## 9.5 Power-up Timer (PWRT)

The Power-up Timer provides a fixed nominal time-out, 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. The power-up timer enable configuration bit, `PWRTE`, 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 AC 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. This ensures that the crystal oscillator or resonator has started and stabilized. See AC parameters for details.

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 Programmable Brown-Out Reset (PBOR)

The PIC16F716 has on-chip Brown-out Reset circuitry. A configuration bit, `BOREN`, can disable (if clear/programmed) or enable (if set) the Brown-out Reset circuitry.

The BORV configuration bit selects the programmable Brown-out Reset threshold voltage (VBOR). When BORV is 1, VBOR is 4.0V. When BORV is 0, VBOR is 2.5V.

A Brown-out Reset occurs when VDD falls below VBOR for a time greater than parameter TBOR (see Table 12-4). A Brown-out Reset is not guaranteed to occur if VDD falls below VBOR for less than parameter TBOR.

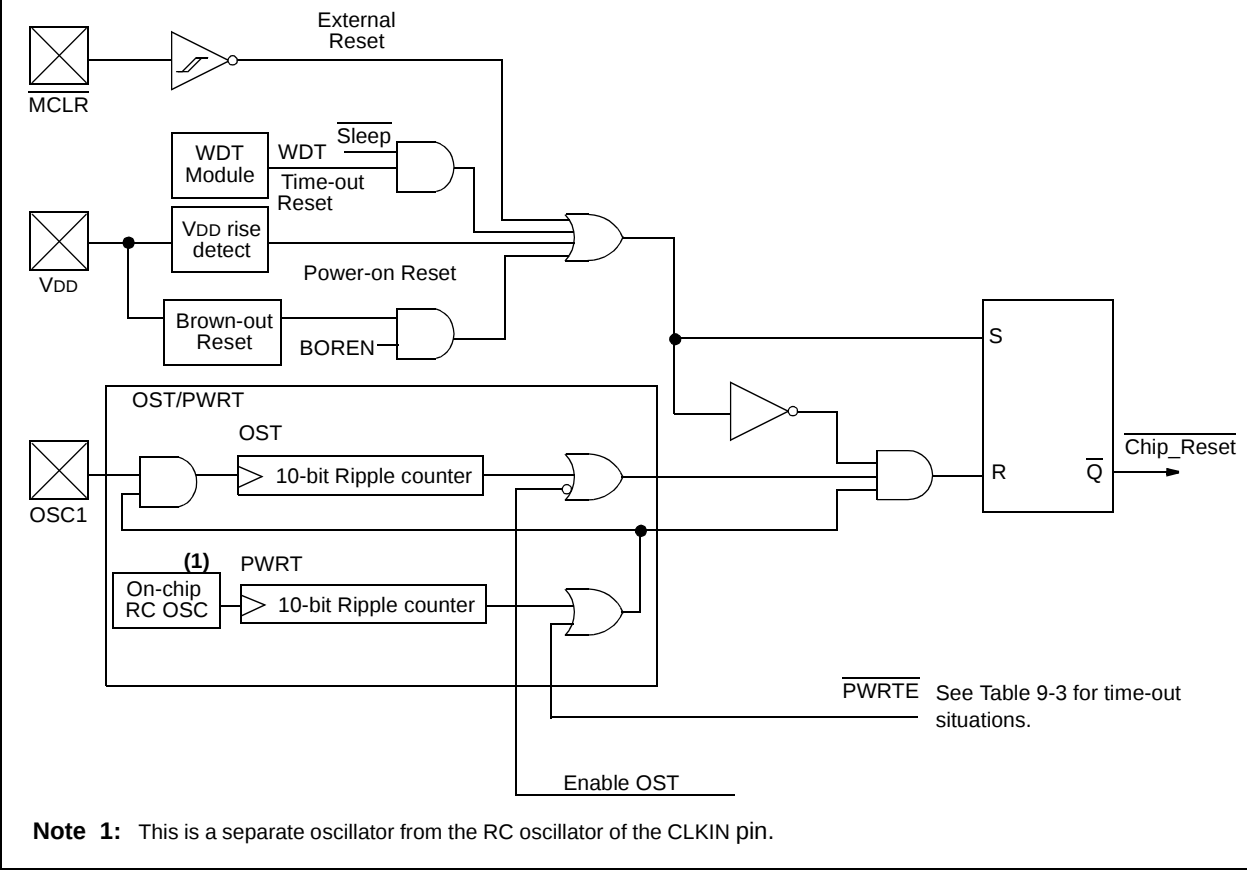
On any Reset (Power-on, Brown-out, Watchdog, etc.) the chip will remain in Reset until VDD rises above VBOR. The Power-up Timer will be invoked and will keep the chip in Reset an additional 72 ms only if the Power-up Timer enable bit in the configuration register is set to 0 (`PWRTE` = 0).

If the Power-up Timer is enabled and 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. See Figure 9-6.

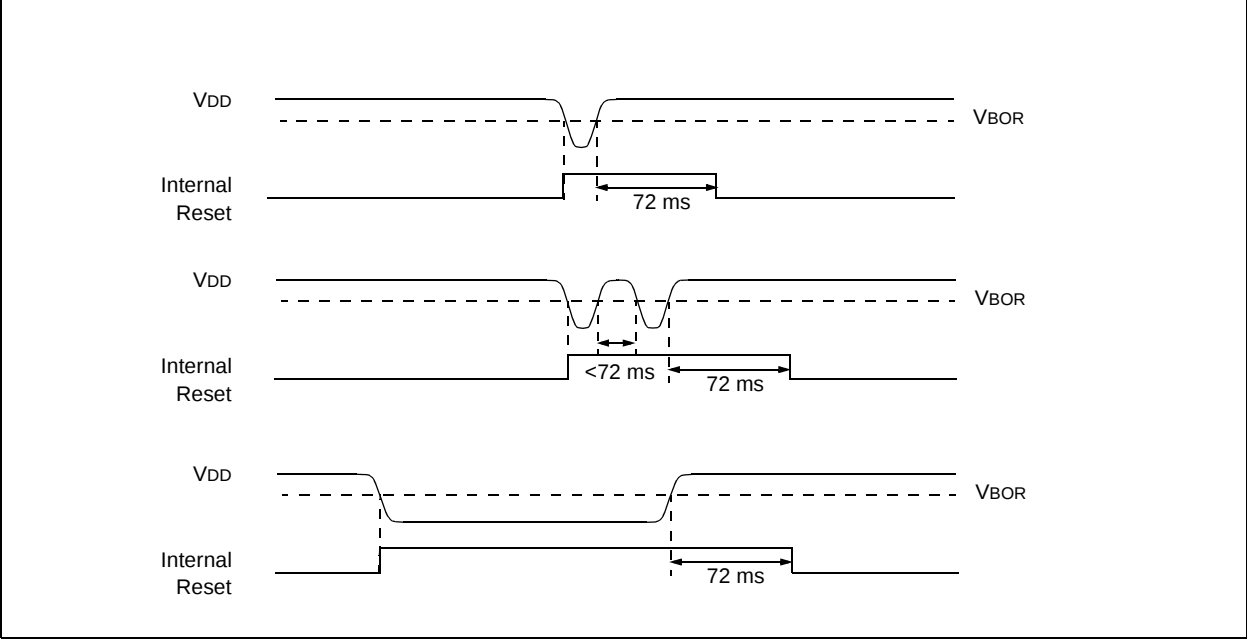
For operations where the desired brown-out voltage is other than 4.0V or 2.5V, an external brown-out circuit must be used. Figure 9-8, Figure 9-9 and Figure 9-10 show examples of external Brown-out Protection circuits.

# PIC16F716

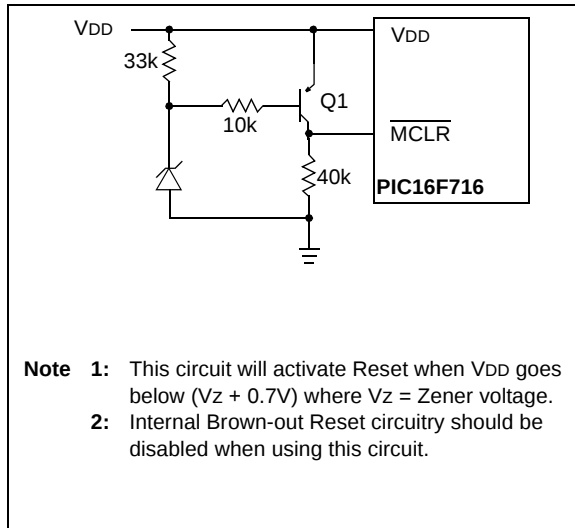
**FIGURE 9-5: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT**



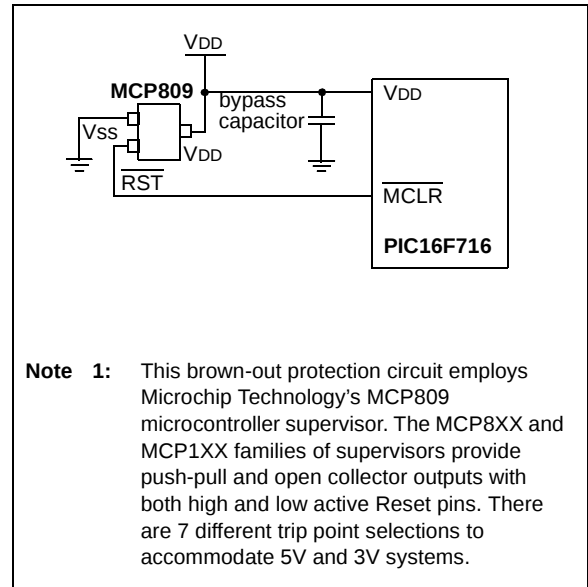
**FIGURE 9-6: BROWN-OUT SITUATIONS ( $\overline{\text{PWRT}} = 0$ )**



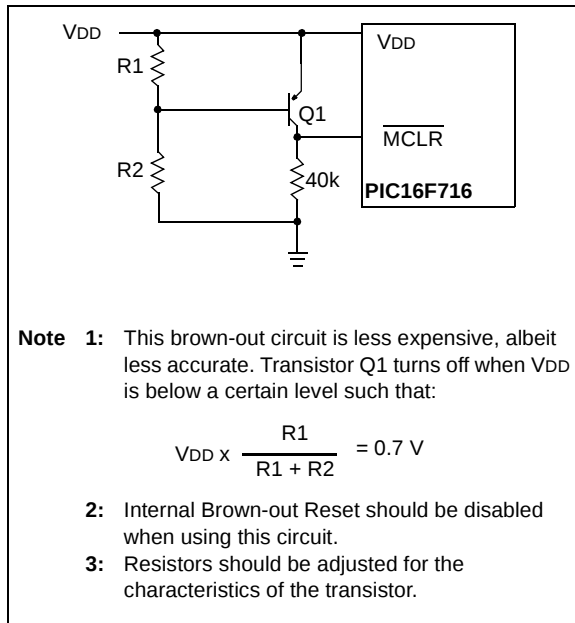
**FIGURE 9-7: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 1**



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



**FIGURE 9-8: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 2**



# PIC16F716

## 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 9-10, Figure 9-11, and Figure 9-12 depict time-out sequences on power-up.

Since the time-outs occur from the POR pulse, if  $\overline{\text{MCLR}}$  is kept low long enough, the time-outs will expire. Then bringing  $\overline{\text{MCLR}}$  high will begin execution immediately (Figure 9-12). This is useful for testing purposes or to synchronize more than one PIC16F716 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.

Bit 0 is the Brown-out Reset Status bit,  $\overline{\text{BOR}}$ . If the BOREN configuration bit is set,  $\overline{\text{BOR}}$  is '1' on Power-on Reset and reset to '0' when a Brown-out condition occurs.  $\overline{\text{BOR}}$  must then be set by the user and checked on subsequent resets to see if it is clear, indicating that another Brown-out has occurred.

If the BOREN configuration bit is clear,  $\overline{\text{BOR}}$  is unknown on Power-on Reset.

Bit 1 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 or Brown-out         |                               | Wake-up from Sleep |
|--------------------------|-------------------------------|-------------------------------|--------------------|
|                          | $\overline{\text{PWRTE}} = 0$ | $\overline{\text{PWRTE}} = 1$ |                    |
| XT, HS, LP               | 72 ms + 1024 Tosc             | 1024 Tosc                     | 1024 Tosc          |
| RC                       | 72 ms                         | —                             | —                  |

TABLE 9-4: STATUS BITS AND THEIR SIGNIFICANCE

| POR | BOR | TO | PD |                                                                             |
|-----|-----|----|----|-----------------------------------------------------------------------------|
| 0   | x   | 1  | 1  | Power-on Reset (BOREN = 0)                                                  |
| 0   | 1   | 1  | 1  | Power-on Reset (BOREN = 1)                                                  |
| 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 (BOREN = 0)         | 000h                  | 0001 1xxx       | ---- --0x     |
| Power-on Reset (BOREN = 1)         | 000h                  | 0001 1xxx       | ---- --01     |
| MCLR Reset during normal operation | 000h                  | 000u uuuu       | ---- --uu     |
| 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 <sup>(1)</sup> | 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).

# PIC16F716

**TABLE 9-6: INITIALIZATION CONDITIONS FOR ALL REGISTERS OF THE PIC16F716**

| Register                       | Power-on Reset,<br>Brown-out Reset | MCLR Resets<br>WDT Reset | Wake-up via WDT or<br>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 <sup>(2)</sup>           |
| STATUS                         | 0001 1xxx                          | 000q quuu <sup>(3)</sup> | uuuq quuu <sup>(3)</sup>        |
| FSR                            | xxxx xxxx                          | uuuu uuuu                | uuuu uuuu                       |
| PORTA <sup>(4), (5), (6)</sup> | --xx 0000                          | --xx 0000                | --uu uuuu                       |
| PORTB <sup>(4), (5)</sup>      | xxxx xxxx                          | uuuu uuuu                | uuuu uuuu                       |
| PCLATH                         | ---0 0000                          | ---0 0000                | ---u uuuu                       |
| INTCON                         | 0000 -00x                          | 0000 -00u                | uuuu -uuu <sup>(1)</sup>        |
| PIR1                           | -0-- -000                          | -0-- -000                | -u-- -uuu <sup>(1)</sup>        |
| 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                        | 0000 0000                          | 0000 0000                | uuuu uuuu                       |
| PWM1CON                        | 0000 0000                          | 0000 0000                | uuuu uuuu                       |
| ECCPAS                         | 00-0 0000                          | 00-0 0000                | u-uu uuuu                       |
| ADRES                          | xxxx xxxx                          | uuuu uuuu                | uuuu uuuu                       |
| ADCON0                         | 0000 0000                          | 0000 0000                | uuuu uuuu                       |
| OPTION_REG                     | 1111 1111                          | 1111 1111                | uuuu uuuu                       |
| TRISA                          | --11 1111                          | --11 1111                | --uu uuuu                       |
| TRISB                          | 1111 1111                          | 1111 1111                | uuuu uuuu                       |
| PIE1                           | -0-- -000                          | -0-- -000                | -u-- -uuu                       |
| PCON                           | ---- -q                            | ---- -uu                 | ---- -uu                        |
| PR2                            | 1111 1111                          | 1111 1111                | uuuu uuuu                       |
| 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.

**6:** Output latches are unknown or unchanged. Analog inputs default to analog and read '0'.

FIGURE 9-10: TIME-OUT SEQUENCE ON POWER-UP ( $\overline{\text{MCLR}}$  TIED TO  $V_{DD}$ )

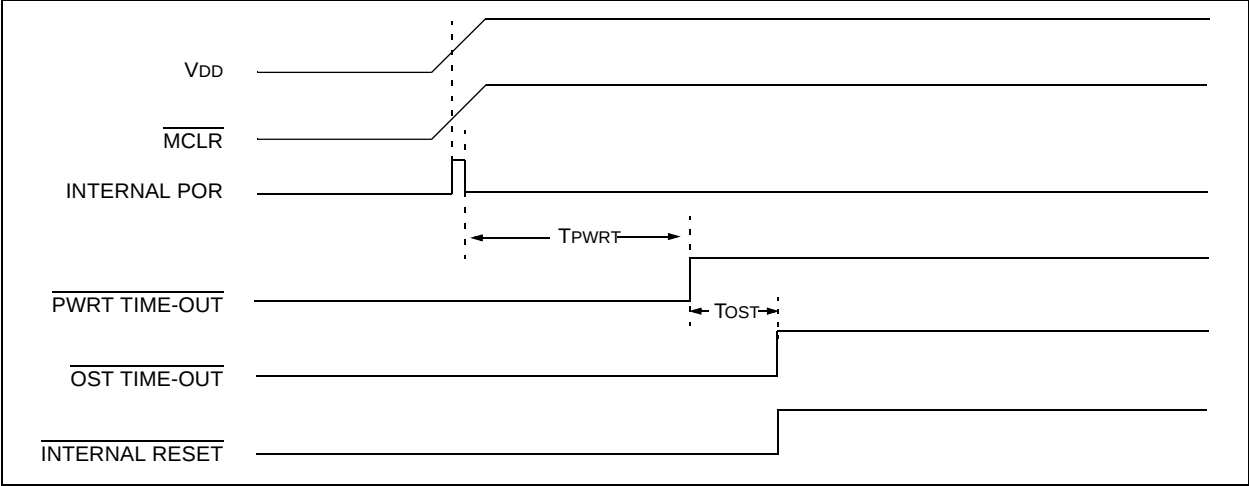


FIGURE 9-11: TIME-OUT SEQUENCE ON POWER-UP ( $\overline{\text{MCLR}}$  NOT TIED TO  $V_{DD}$ ): CASE 1

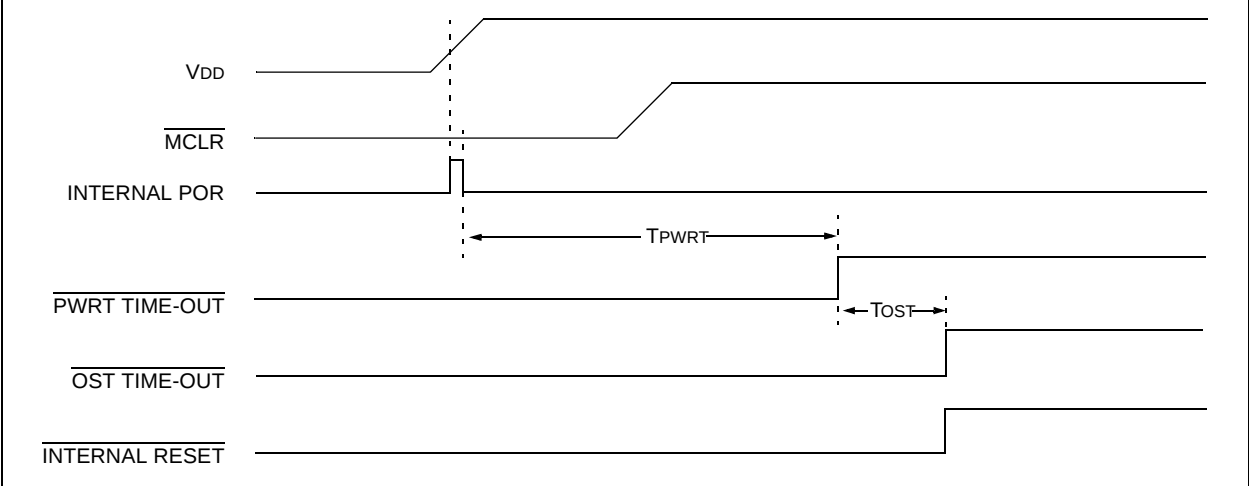
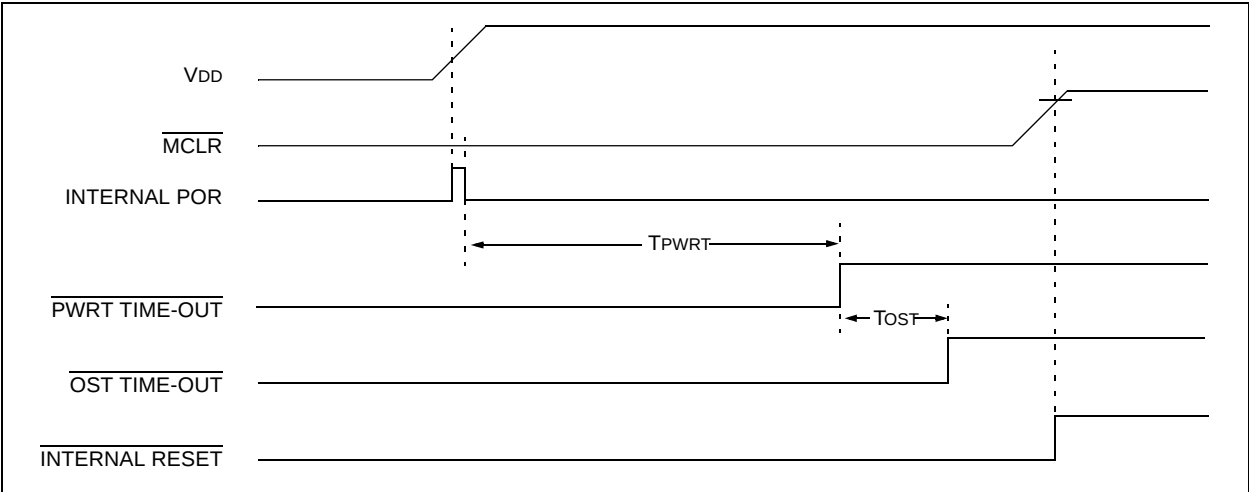


FIGURE 9-12: TIME-OUT SEQUENCE ON POWER-UP ( $\overline{\text{MCLR}}$  NOT TIED TO  $V_{DD}$ ): CASE 2



# PIC16F716

## 9.10 Interrupts

The PIC16F716 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 all un-masked interrupts when set, or disables all interrupts when cleared. 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 and when an interrupt vector occurs.

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

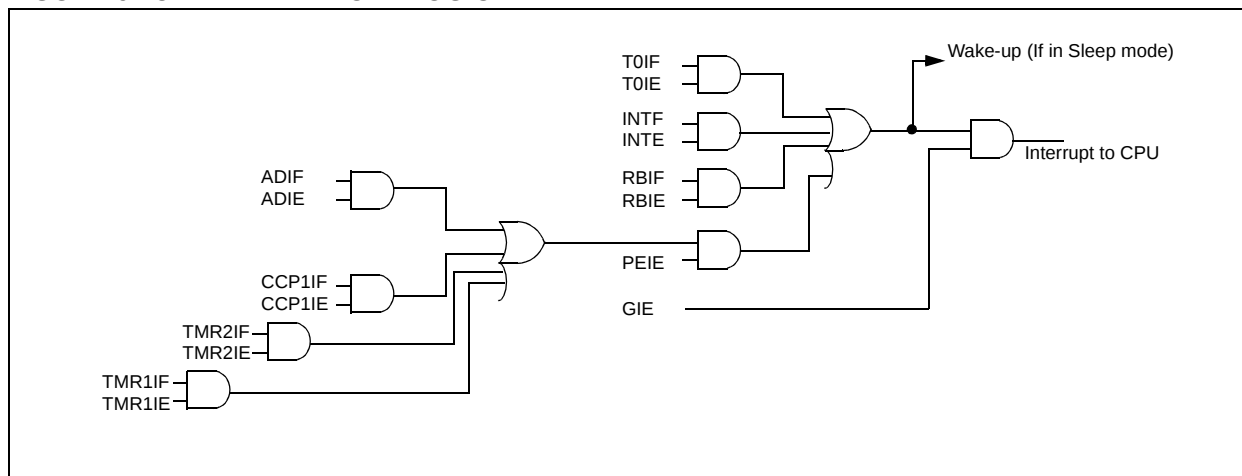
The RB0/INT pin interrupt, the RB port change interrupt and the TMR0 overflow interrupt flags are contained in the INTCON register.

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 9-13: INTERRUPT LOGIC**



## 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 “Power-down Mode (Sleep)”** for details on Sleep mode.

## 9.10.2 TMR0 INTERRUPT

An overflow (FFh → 00h) in the TMR0 register will set flag bit TOIF (INTCON<2>). The interrupt can be enabled/disabled by setting/clearing enable bit TOIE (INTCON<5>). (**Section 4.0 “Timer0 Module”**).

## 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 “PORTB and the TRISB Register”**).

## 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 firmware.

Example 9-1 stores and restores the W, Status, PCLATH and FSR registers. Context storage registers, W\_TEMP, STATUS\_TEMP, PCLATH\_TEMP and FSR\_TEMP, must be defined in Common RAM which are those addresses between 70h-7Fh in Bank 0 and between F0h-FFh in Bank 1.

The example:

- Stores the W register.
- Stores the Status register in Bank 0.
- Stores the PCLATH register
- Stores the FSR register.
- Executes the interrupt service routine code (User-generated).
- Restores all saved registers in reverse order from which they were stored

### 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
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     FSR_TEMP,W      ;Restore FSR
MOVWF    FSR             ;Move W into FSR
MOVF     PCLATH_TEMP, W  ;Restore PCLATH
MOVWF    PCLATH          ;Move W into PCLATH
SWAPF    STATUS_TEMP,W   ;Swap STATUS_TEMP register into W
MOVWF    STATUS          ;Move W into STATUS register
SWAPF    W_TEMP,F        ;Swap W_TEMP
SWAPF    W_TEMP,W        ;Swap W_TEMP into W
RETFIE                   ;Return from interrupt and enable GIE

```

# PIC16F716

## 9.12 Watchdog Timer (WDT)

The Watchdog Timer is 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.

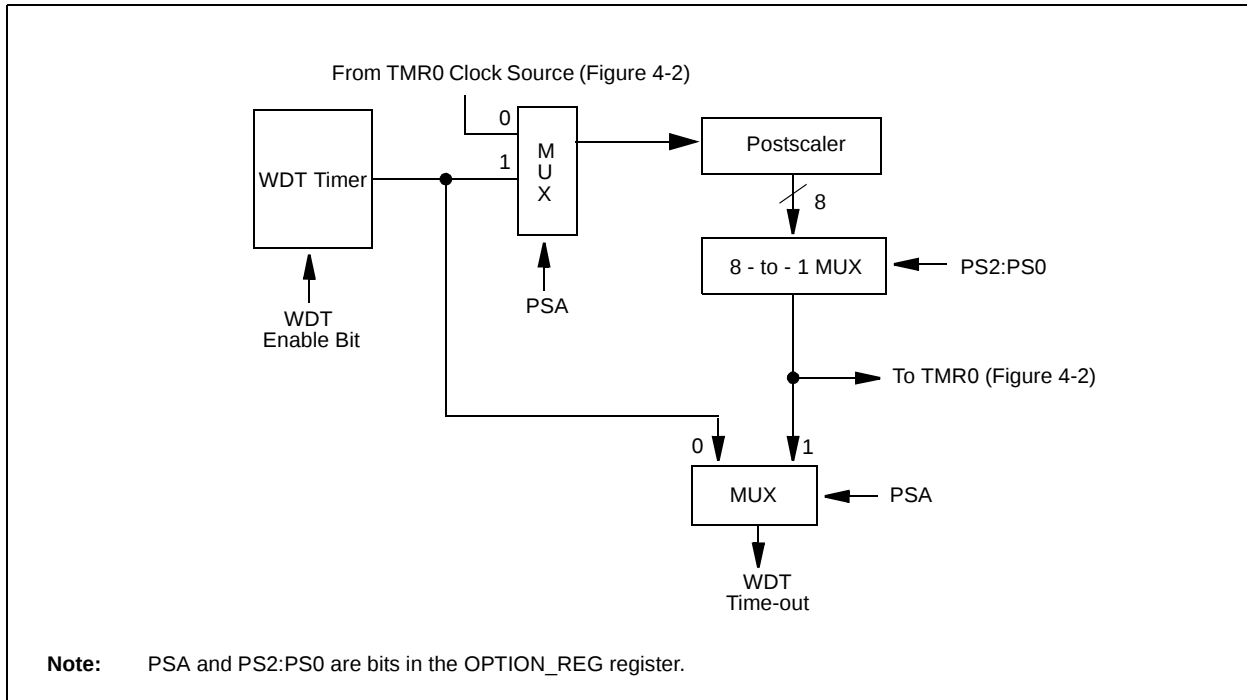
The WDT can be permanently disabled by clearing configuration bit WDTE (Section 9.1 “Configuration Bits”).

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 9-14: WATCHDOG TIMER BLOCK DIAGRAM**



**FIGURE 9-15: 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)       | BORV <sup>(1)</sup> | BOREN <sup>(1)</sup> | —     | —     | $\overline{PWRTE}$ <sup>(1)</sup> | WDTE  | FOSC1 | FOSC0 |
| 81h     | OPTION_REG   | N/A       | RBP $\overline{U}$  | INTEDG               | T0CS  | T0SE  | PSA                               | PS2   | PS1   | PS0   |

**Legend:** Shaded cells are not used by the Watchdog Timer.

**Note 1:** See Register 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 high-impedance).

For lowest current consumption in this mode, place all I/O pins at either  $V_{DD}$  or  $V_{SS}$ , 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  $TOCKI$  input should also be at  $V_{DD}$  or  $V_{SS}$  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 (parameter D042).

### 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. ECCP capture mode interrupt.
3. ADC running in ADRC mode.

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:

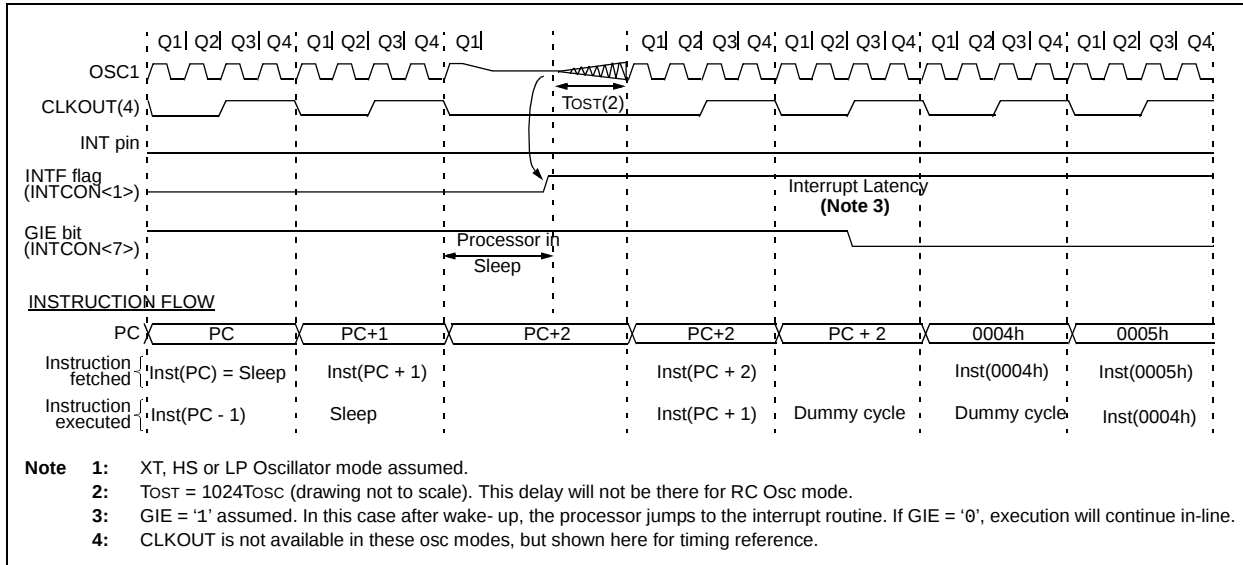
- If the interrupt occurs **before** the execution of a SLEEP instruction, the SLEEP instruction will complete as a NOP. Therefore, the WDT and WDT postscaler will not be cleared, the  $\overline{TO}$  bit will not be set and  $\overline{PD}$  bits will not be cleared.
- If the interrupt occurs **during 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.

# PIC16F716

**FIGURE 9-16: WAKE-UP FROM SLEEP THROUGH INTERRUPT**



## 9.14 Program Verification/Code Protection

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

## 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.

## 9.16 In-Circuit Serial Programming™

PIC16F716 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™) Specification, (DS40245).

## 10.0 INSTRUCTION SET SUMMARY

Each PIC16F716 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 PIC16F716 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)<br>The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools. |
| d       | Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1                                                                |
| label   | Label name                                                                                                                                                            |
| TOS     | Top of Stack                                                                                                                                                          |
| PC      | Program Counter                                                                                                                                                       |
| PCLATH  | Program Counter High Latch                                                                                                                                            |
| GIE     | Global Interrupt Enable bit                                                                                                                                           |
| WDT     | Watchdog Timer/Counter                                                                                                                                                |
| TO      | Time-out bit                                                                                                                                                          |
| PD      | Power-down bit                                                                                                                                                        |
| dest    | Destination either the W register or the specified register file location                                                                                             |
| [ ]     | Options                                                                                                                                                               |
| ( )     | Contents                                                                                                                                                              |
| →       | Assigned to                                                                                                                                                           |
| < >     | Register bit field                                                                                                                                                    |
| ∈       | In the set of                                                                                                                                                         |
| italics | User defined term (font is courier)                                                                                                                                   |

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

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

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

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

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

**Note 1:** Any unused opcode is reserved. Use of any reserved opcode may cause unexpected operation.

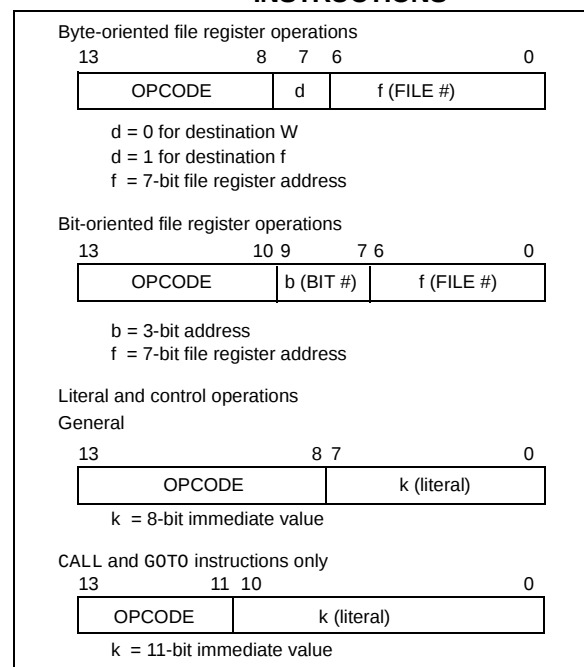
**2:** To maintain upward compatibility with future PICmicro products, do not use the OPTION and TRIS instructions.

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

0xhh

where h signifies a hexadecimal digit.

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



# PIC16F716

**TABLE 10-2: PIC16F716 INSTRUCTION SET**

| Mnemonic,<br>Operands                  |      | Description                  | Cycles           | 14-Bit Opcode |      |      |      | Status<br>Affected             | Notes |
|----------------------------------------|------|------------------------------|------------------|---------------|------|------|------|--------------------------------|-------|
|                                        |      |                              |                  | MSb           |      | LSb  |      |                                |       |
| 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 | 0xxx | xxxx | 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 <sup>(2)</sup> | 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 <sup>(2)</sup> | 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 <sup>(2)</sup> | 01            | 10bb | bfff | ffff |                                | 3     |
| BTFSS                                  | f, b | Bit Test f, Skip if Set      | 1 <sup>(2)</sup> | 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'.
- Note 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.
- Note 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.

## 10.1 Instruction Descriptions

### ADDLW Add Literal and W

|                  |                                                                                                                   |      |      |      |      |
|------------------|-------------------------------------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ <i>label</i> ] ADDLW    k                                                                                       |      |      |      |      |
| Operands:        | $0 \leq k \leq 255$                                                                                               |      |      |      |      |
| Operation:       | $(W) + k \rightarrow (W)$                                                                                         |      |      |      |      |
| Status Affected: | C, DC, Z                                                                                                          |      |      |      |      |
| Encoding:        | <table border="1"><tr><td>11</td><td>111x</td><td>kkkk</td><td>kkkk</td></tr></table>                             | 11   | 111x | kkkk | kkkk |
| 11               | 111x                                                                                                              | kkkk | kkkk |      |      |
| Description:     | The contents of the W register are added to the eight bit literal 'k' and the result is placed in the W register. |      |      |      |      |
| Words:           | 1                                                                                                                 |      |      |      |      |
| Cycles:          | 1                                                                                                                 |      |      |      |      |
| Example          | ADDLW    0x15<br><br>Before Instruction<br>W = 0x10<br>After Instruction<br>W = 0x25                              |      |      |      |      |

### ADDWF Add W and f

|                  |                                                                                                                                                                  |      |      |      |      |
|------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ <i>label</i> ] ADDWF f,d                                                                                                                                       |      |      |      |      |
| Operands:        | $0 \leq f \leq 127$<br>$d \in [0,1]$                                                                                                                             |      |      |      |      |
| Operation:       | $(W) + (f) \rightarrow (\text{dest})$                                                                                                                            |      |      |      |      |
| Status Affected: | C, DC, Z                                                                                                                                                         |      |      |      |      |
| Encoding:        | <table border="1"><tr><td>00</td><td>0111</td><td>dfff</td><td>ffff</td></tr></table>                                                                            | 00   | 0111 | dfff | ffff |
| 00               | 0111                                                                                                                                                             | dfff | ffff |      |      |
| Description:     | Add the contents of the W register with register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is 1 the result is stored back in register 'f'. |      |      |      |      |
| Words:           | 1                                                                                                                                                                |      |      |      |      |
| Cycles:          | 1                                                                                                                                                                |      |      |      |      |
| Example          | ADDWF REG1, 0<br><br>Before Instruction<br>W = 0x17<br>REG1 = 0xC2<br><br>After Instruction<br>W = 0xD9<br>REG1 = 0xC2<br>Z = 0<br>C = 0<br>DC = 0               |      |      |      |      |

### ANDLW AND Literal with W

|                  |                                                                                                               |      |      |      |      |
|------------------|---------------------------------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ <i>label</i> ] ANDLW k                                                                                      |      |      |      |      |
| Operands:        | $0 \leq k \leq 255$                                                                                           |      |      |      |      |
| Operation:       | (W) .AND. (k) $\rightarrow$ (W)                                                                               |      |      |      |      |
| Status Affected: | Z                                                                                                             |      |      |      |      |
| Encoding:        | <table border="1"><tr><td>11</td><td>1001</td><td>kkkk</td><td>kkkk</td></tr></table>                         | 11   | 1001 | kkkk | kkkk |
| 11               | 1001                                                                                                          | kkkk | kkkk |      |      |
| Description:     | The contents of W register are AND'ed with the eight bit literal 'k'. The result is placed in the W register. |      |      |      |      |
| Words:           | 1                                                                                                             |      |      |      |      |
| Cycles:          | 1                                                                                                             |      |      |      |      |
| Example          | ANDLW 0x5F<br><br>Before Instruction<br>W = 0xA3<br>After Instruction<br>W = 0x03                             |      |      |      |      |

### ANDWF AND W with f

|                  |                                                                                                                                                  |      |      |      |      |
|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ <i>label</i> ] ANDWF f,d                                                                                                                       |      |      |      |      |
| Operands:        | $0 \leq f \leq 127$<br>$d \in [0,1]$                                                                                                             |      |      |      |      |
| Operation:       | (W) .AND. (f) $\rightarrow$ (dest)                                                                                                               |      |      |      |      |
| Status Affected: | Z                                                                                                                                                |      |      |      |      |
| Encoding:        | <table border="1"><tr><td>00</td><td>0101</td><td>dfff</td><td>ffff</td></tr></table>                                                            | 00   | 0101 | dfff | ffff |
| 00               | 0101                                                                                                                                             | dfff | ffff |      |      |
| Description:     | AND the W register with register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is 1 the result is stored back in register 'f'. |      |      |      |      |
| Words:           | 1                                                                                                                                                |      |      |      |      |
| Cycles:          | 1                                                                                                                                                |      |      |      |      |
| Example          | ANDWF REG1, 1<br><br>Before Instruction<br>W = 0x17<br>REG1 = 0xC2<br><br>After Instruction<br>W = 0x17<br>REG1 = 0x02                           |      |      |      |      |

# PIC16F716

## BCF Bit Clear f

Syntax: [ *label* ] BCF f,b

Operands:  $0 \leq f \leq 127$   
 $0 \leq b \leq 7$

Operation:  $0 \rightarrow (f<b>)$

Status Affected: None

Encoding: 

|    |      |      |      |
|----|------|------|------|
| 01 | 00bb | bfff | ffff |
|----|------|------|------|

Description: Bit 'b' in register 'f' is cleared.

Words: 1

Cycles: 1

Example BCF REG1, 7  
Before Instruction  
REG1 = 0xC7  
After Instruction  
REG1 = 0x47

## BSF Bit Set f

Syntax: [ *label* ] BSF f,b

Operands:  $0 \leq f \leq 127$   
 $0 \leq b \leq 7$

Operation:  $1 \rightarrow (f<b>)$

Status Affected: None

Encoding: 

|    |      |      |      |
|----|------|------|------|
| 01 | 01bb | bfff | ffff |
|----|------|------|------|

Description: Bit 'b' in register 'f' is set.

Words: 1

Cycles: 1

Example BSF REG1, 7  
Before Instruction  
REG1 = 0x0A  
After Instruction  
REG1 = 0x8A

## BTFSC Bit Test f, Skip if Clear

Syntax: [ *label* ] BTFSC f,b

Operands:  $0 \leq f \leq 127$   
 $0 \leq b \leq 7$

Operation: skip if  $(f<b>) = 0$

Status Affected: None

Encoding: 

|    |      |      |      |
|----|------|------|------|
| 01 | 10bb | bfff | ffff |
|----|------|------|------|

Description: If bit 'b' in register 'f' is '0' then the next instruction is skipped.  
If bit 'b' is '0' then the next instruction fetched during the current instruction execution is discarded, and a NOP is executed instead, making this a two-cycle instruction.

Words: 1

Cycles: 1(2)

Example  
HERE BTFSC REG1  
FALSE GOTO PROCESS\_CODE  
TRUE .  
.  
.

Before Instruction

PC = address HERE

After Instruction

if  $REG<1> = 0$ ,

PC = address TRUE

if  $REG<1> \neq 1$ ,

PC = address FALSE

## BTFSS Bit Test f, Skip if Set

|                  |                                                                                                                                                                                                                                                                       |      |           |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|-----------|
| Syntax:          | [ <i>label</i> ] BTFSS f,b                                                                                                                                                                                                                                            |      |           |
| Operands:        | $0 \leq f \leq 127$<br>$0 \leq b < 7$                                                                                                                                                                                                                                 |      |           |
| Operation:       | skip if (f<b>) = 1                                                                                                                                                                                                                                                    |      |           |
| Status Affected: | None                                                                                                                                                                                                                                                                  |      |           |
| Encoding:        | 01                                                                                                                                                                                                                                                                    | 11bb | bfff ffff |
| Description:     | <p>If bit 'b' in register 'f' is '1' then the next instruction is skipped.</p> <p>If bit 'b' is '1', then the next instruction fetched during the current instruction execution, is discarded and a NOP is executed instead, making this a two-cycle instruction.</p> |      |           |
| Words:           | 1                                                                                                                                                                                                                                                                     |      |           |
| Cycles:          | 1(2)                                                                                                                                                                                                                                                                  |      |           |
| Example          | <pre> HERE    BTFSS    REG1 FALSE   GOTO     PROCESS_CODE TRUE    •         •         •  Before Instruction     PC = address HERE After Instruction     if FLAG&lt;1&gt; = 0,         PC = address FALSE if FLAG&lt;1&gt; = 1,     PC = address TRUE </pre>           |      |           |

## CALL Call Subroutine

|                  |                                                                                                                                                                                                                                        |      |           |
|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|-----------|
| Syntax:          | [ <i>label</i> ] CALL k                                                                                                                                                                                                                |      |           |
| Operands:        | $0 \leq k \leq 2047$                                                                                                                                                                                                                   |      |           |
| Operation:       | (PC)+ 1 → TOS,<br>k → PC<10:0>,<br>(PCLATH<4:3>) → PC<12:11>                                                                                                                                                                           |      |           |
| Status Affected: | None                                                                                                                                                                                                                                   |      |           |
| Encoding:        | 10                                                                                                                                                                                                                                     | 0kkk | kkkk kkkk |
| Description:     | <p>Call Subroutine. First, return address (PC+1) is pushed onto the stack. The eleven bit immediate address is loaded into PC bits &lt;10:0&gt;. The upper bits of the PC are loaded from PCLATH. CALL is a two-cycle instruction.</p> |      |           |
| Words:           | 1                                                                                                                                                                                                                                      |      |           |
| Cycles:          | 2                                                                                                                                                                                                                                      |      |           |
| Example          | <pre> HERE    CALL     THERE  Before Instruction     PC = Address HERE After Instruction     PC = Address THERE     TOS = Address HERE+1 </pre>                                                                                        |      |           |

## CLRF Clear f

|                  |                                                                                                              |      |           |
|------------------|--------------------------------------------------------------------------------------------------------------|------|-----------|
| Syntax:          | [ <i>label</i> ] CLRF f                                                                                      |      |           |
| Operands:        | $0 \leq f \leq 127$                                                                                          |      |           |
| Operation:       | 00h → (f)<br>1 → Z                                                                                           |      |           |
| Status Affected: | Z                                                                                                            |      |           |
| Encoding:        | 00                                                                                                           | 0001 | 1fff ffff |
| Description:     | <p>The contents of register 'f' are cleared and the Z bit is set.</p>                                        |      |           |
| Words:           | 1                                                                                                            |      |           |
| Cycles:          | 1                                                                                                            |      |           |
| Example          | <pre> CLRF    REG1  Before Instruction     REG1 = 0x5A After Instruction     REG1 = 0x00     Z    = 1 </pre> |      |           |

# PIC16F716

## CLRW Clear W

|                  |                                                                                       |      |      |      |      |
|------------------|---------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ <i>label</i> ] CLRW                                                                 |      |      |      |      |
| Operands:        | None                                                                                  |      |      |      |      |
| Operation:       | 00h → (W)<br>1 → Z                                                                    |      |      |      |      |
| Status Affected: | Z                                                                                     |      |      |      |      |
| Encoding:        | <table border="1"><tr><td>00</td><td>0001</td><td>0000</td><td>0011</td></tr></table> | 00   | 0001 | 0000 | 0011 |
| 00               | 0001                                                                                  | 0000 | 0011 |      |      |
| Description:     | W register is cleared. Zero bit (Z) is set.                                           |      |      |      |      |
| Words:           | 1                                                                                     |      |      |      |      |
| Cycles:          | 1                                                                                     |      |      |      |      |
| Example          | CLRW<br><br>Before Instruction<br>W = 0x5A<br>After Instruction<br>W = 0x00<br>Z = 1  |      |      |      |      |

## COMF Complement f

|                  |                                                                                                                                              |      |      |      |      |
|------------------|----------------------------------------------------------------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ <i>label</i> ] COMF f,d                                                                                                                    |      |      |      |      |
| Operands:        | 0 ≤ f ≤ 127<br>d ∈ [0,1]                                                                                                                     |      |      |      |      |
| Operation:       | ( $\bar{f}$ ) → (dest)                                                                                                                       |      |      |      |      |
| Status Affected: | Z                                                                                                                                            |      |      |      |      |
| Encoding:        | <table><tr><td>00</td><td>1001</td><td>dfff</td><td>ffff</td></tr></table>                                                                   | 00   | 1001 | dfff | ffff |
| 00               | 1001                                                                                                                                         | dfff | ffff |      |      |
| Description:     | The contents of register 'f' are complemented. If 'd' is 0 the result is stored in W. If 'd' is 1 the result is stored back in register 'f'. |      |      |      |      |
| Words:           | 1                                                                                                                                            |      |      |      |      |
| Cycles:          | 1                                                                                                                                            |      |      |      |      |
| Example          | COMF REG1, 0<br>Before Instruction<br>REG1 = 0x13<br>After Instruction<br>REG1 = 0x13<br>W = 0xEC                                            |      |      |      |      |

## CLRWDTClear Watchdog Timer

|                  |                                                                                                                                                                       |      |      |      |      |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ <i>label</i> ] CLRWDT                                                                                                                                               |      |      |      |      |
| Operands:        | None                                                                                                                                                                  |      |      |      |      |
| Operation:       | 00h → WDT<br>0 → WDT prescaler,<br>1 → $\overline{TO}$<br>1 → $\overline{PD}$                                                                                         |      |      |      |      |
| Status Affected: | $\overline{TO}$ , $\overline{PD}$                                                                                                                                     |      |      |      |      |
| Encoding:        | <table border="1"><tr><td>00</td><td>0000</td><td>0110</td><td>0100</td></tr></table>                                                                                 | 00   | 0000 | 0110 | 0100 |
| 00               | 0000                                                                                                                                                                  | 0110 | 0100 |      |      |
| Description:     | CLRWDT instruction resets the Watchdog Timer. It also resets the <u>prescaler</u> of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set.                |      |      |      |      |
| Words:           | 1                                                                                                                                                                     |      |      |      |      |
| Cycles:          | 1                                                                                                                                                                     |      |      |      |      |
| Example          | CLRWDT<br><br>Before Instruction<br>WDT counter = ?<br><br>After Instruction<br>WDT counter = 0x00<br>WDT prescaler = 0<br>$\overline{TO}$ = 1<br>$\overline{PD}$ = 1 |      |      |      |      |

## DECFDecrement f

|                  |                                                                                                                                    |      |      |      |      |
|------------------|------------------------------------------------------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ <i>label</i> ] DECF f,d                                                                                                          |      |      |      |      |
| Operands:        | $0 \leq f \leq 127$<br>$d \in [0,1]$                                                                                               |      |      |      |      |
| Operation:       | $(f) - 1 \rightarrow (\text{dest})$                                                                                                |      |      |      |      |
| Status Affected: | Z                                                                                                                                  |      |      |      |      |
| Encoding:        | <table border="1"><tr><td>00</td><td>0011</td><td>dfff</td><td>ffff</td></tr></table>                                              | 00   | 0011 | dfff | ffff |
| 00               | 0011                                                                                                                               | dfff | ffff |      |      |
| Description:     | Decrement register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is 1 the result is stored back in register 'f'. |      |      |      |      |
| Words:           | 1                                                                                                                                  |      |      |      |      |
| Cycles:          | 1                                                                                                                                  |      |      |      |      |
| Example          | DECF CNT, 1<br><br>Before Instruction<br>CNT = 0x01<br>Z = 0<br><br>After Instruction<br>CNT = 0x00<br>Z = 1                       |      |      |      |      |

| DECFSZ           |                                                                                                                                                                                                                                                                                                                                                                                                                          | Decrement f, Skip if 0 |      |  |  |      |        |         |      |      |      |          |   |  |  |   |  |  |   |  |
|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|------|--|--|------|--------|---------|------|------|------|----------|---|--|--|---|--|--|---|--|
| Syntax:          | [ <i>label</i> ] DECFSZ f,d                                                                                                                                                                                                                                                                                                                                                                                              |                        |      |  |  |      |        |         |      |      |      |          |   |  |  |   |  |  |   |  |
| Operands:        | $0 \leq f \leq 127$<br>$d \in [0,1]$                                                                                                                                                                                                                                                                                                                                                                                     |                        |      |  |  |      |        |         |      |      |      |          |   |  |  |   |  |  |   |  |
| Operation:       | $(f) - 1 \rightarrow (\text{dest});$ skip if result = 0                                                                                                                                                                                                                                                                                                                                                                  |                        |      |  |  |      |        |         |      |      |      |          |   |  |  |   |  |  |   |  |
| Status Affected: | None                                                                                                                                                                                                                                                                                                                                                                                                                     |                        |      |  |  |      |        |         |      |      |      |          |   |  |  |   |  |  |   |  |
| Encoding:        | <table border="1"><tr><td>00</td><td>1011</td><td>dfff</td><td>ffff</td></tr></table>                                                                                                                                                                                                                                                                                                                                    |                        |      |  |  | 00   | 1011   | dfff    | ffff |      |      |          |   |  |  |   |  |  |   |  |
| 00               | 1011                                                                                                                                                                                                                                                                                                                                                                                                                     | dfff                   | ffff |  |  |      |        |         |      |      |      |          |   |  |  |   |  |  |   |  |
| Description:     | <p>The contents of register 'f' are decremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.</p> <p>If the result is 0, the next instruction, which is already fetched, is discarded. A NOP is executed instead making it a two-cycle instruction.</p>                                                                                                    |                        |      |  |  |      |        |         |      |      |      |          |   |  |  |   |  |  |   |  |
| Words:           | 1                                                                                                                                                                                                                                                                                                                                                                                                                        |                        |      |  |  |      |        |         |      |      |      |          |   |  |  |   |  |  |   |  |
| Cycles:          | 1(2)                                                                                                                                                                                                                                                                                                                                                                                                                     |                        |      |  |  |      |        |         |      |      |      |          |   |  |  |   |  |  |   |  |
| Example          | <table><tr><td>HERE</td><td>DECFSZ</td><td>REG1, 1</td></tr><tr><td></td><td>GOTO</td><td>LOOP</td></tr><tr><td>CONTINUE</td><td>•</td><td></td></tr><tr><td></td><td>•</td><td></td></tr><tr><td></td><td>•</td><td></td></tr></table> <p>Before Instruction<br/>PC = address HERE</p> <p>After Instruction<br/>REG1 = REG1 - 1<br/>if REG1 = 0,<br/>PC = address CONTINUE<br/>if REG1 ≠ 0,<br/>PC = address HERE+1</p> |                        |      |  |  | HERE | DECFSZ | REG1, 1 |      | GOTO | LOOP | CONTINUE | • |  |  | • |  |  | • |  |
| HERE             | DECFSZ                                                                                                                                                                                                                                                                                                                                                                                                                   | REG1, 1                |      |  |  |      |        |         |      |      |      |          |   |  |  |   |  |  |   |  |
|                  | GOTO                                                                                                                                                                                                                                                                                                                                                                                                                     | LOOP                   |      |  |  |      |        |         |      |      |      |          |   |  |  |   |  |  |   |  |
| CONTINUE         | •                                                                                                                                                                                                                                                                                                                                                                                                                        |                        |      |  |  |      |        |         |      |      |      |          |   |  |  |   |  |  |   |  |
|                  | •                                                                                                                                                                                                                                                                                                                                                                                                                        |                        |      |  |  |      |        |         |      |      |      |          |   |  |  |   |  |  |   |  |
|                  | •                                                                                                                                                                                                                                                                                                                                                                                                                        |                        |      |  |  |      |        |         |      |      |      |          |   |  |  |   |  |  |   |  |

| GOTO             |                                                                                                                                                                                   | Unconditional Branch |      |  |  |    |      |      |      |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------|------|--|--|----|------|------|------|
| Syntax:          | [ <i>label</i> ] GOTO k                                                                                                                                                           |                      |      |  |  |    |      |      |      |
| Operands:        | 0 ≤ k ≤ 2047                                                                                                                                                                      |                      |      |  |  |    |      |      |      |
| Operation:       | k → PC<10:0><br>PCLATH<4:3> → PC<12:11>                                                                                                                                           |                      |      |  |  |    |      |      |      |
| Status Affected: | None                                                                                                                                                                              |                      |      |  |  |    |      |      |      |
| Encoding:        | <table border="1"><tr><td>10</td><td>1kkk</td><td>kkkk</td><td>kkkk</td></tr></table>                                                                                             |                      |      |  |  | 10 | 1kkk | kkkk | kkkk |
| 10               | 1kkk                                                                                                                                                                              | kkkk                 | kkkk |  |  |    |      |      |      |
| Description:     | GOTO is an unconditional branch. The eleven-bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. GOTO is a two-cycle instruction. |                      |      |  |  |    |      |      |      |
| Words:           | 1                                                                                                                                                                                 |                      |      |  |  |    |      |      |      |
| Cycles:          | 2                                                                                                                                                                                 |                      |      |  |  |    |      |      |      |
| Example          | GOTO THERE                                                                                                                                                                        |                      |      |  |  |    |      |      |      |
|                  | After Instruction                                                                                                                                                                 |                      |      |  |  |    |      |      |      |
|                  | PC = Address THERE                                                                                                                                                                |                      |      |  |  |    |      |      |      |

# PIC16F716

## INCF Increment f

Syntax: [ *label* ] INCF f,d

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

Operation:  $(f) + 1 \rightarrow (\text{dest})$

Status Affected: Z

Encoding: 

|    |      |      |      |
|----|------|------|------|
| 00 | 1010 | dfff | ffff |
|----|------|------|------|

Description: The contents of register 'f' are incremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.

Words: 1

Cycles: 1

Example INCF REG1, 1

Before Instruction

REG1 = 0xFF

Z = 0

After Instruction

REG1 = 0x00

Z = 1

## INCFSZ Increment f, Skip if 0

Syntax: [ *label* ] INCFSZ f,d

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

Operation:  $(f) + 1 \rightarrow (\text{dest})$ , skip if result = 0

Status Affected: None

Encoding: 

|    |      |      |      |
|----|------|------|------|
| 00 | 1111 | dfff | ffff |
|----|------|------|------|

Description: The contents of register 'f' are incremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.

If the result is 0, the next instruction, which is already fetched, is discarded. A NOP is executed instead making it a two-cycle instruction.

Words: 1

Cycles: 1(2)

Example HERE INCFSZ REG1, 1

GOTO LOOP

CONTINUE •  
•  
•

Before Instruction

PC = address HERE

After Instruction

REG1 = REG1 + 1

if CNT = 0,

PC = address CONTINUE

if REG1 ≠ 0,

PC = address HERE +1

## IORLW Inclusive OR Literal with W

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

## MOVLW Move Literal to W

|                  |                                                                                             |      |      |      |      |
|------------------|---------------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ <i>label</i> ]    MOVLW <i>k</i>                                                          |      |      |      |      |
| Operands:        | $0 \leq k \leq 255$                                                                         |      |      |      |      |
| Operation:       | $k \rightarrow (W)$                                                                         |      |      |      |      |
| Status Affected: | None                                                                                        |      |      |      |      |
| Encoding:        | <table border="1"><tr><td>11</td><td>00xx</td><td>kkkk</td><td>kkkk</td></tr></table>       | 11   | 00xx | kkkk | kkkk |
| 11               | 00xx                                                                                        | kkkk | kkkk |      |      |
| Description:     | The eight bit literal 'k' is loaded into W register. The don't cares will assemble as '0's. |      |      |      |      |
| Words:           | 1                                                                                           |      |      |      |      |
| Cycles:          | 1                                                                                           |      |      |      |      |
| Example          | MOVLW    0x5A<br><br>After Instruction<br>W = 0x5A                                          |      |      |      |      |

## IORWF Inclusive OR W with f

|                  |                                                                                                                                                           |      |      |      |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|------|------|------|
| Syntax:          | [ <i>label</i> ] IORWF f,d                                                                                                                                |      |      |      |
| Operands:        | $0 \leq f \leq 127$<br>$d \in [0,1]$                                                                                                                      |      |      |      |
| Operation:       | (W) .OR. (f) $\rightarrow$ (dest)                                                                                                                         |      |      |      |
| Status Affected: | Z                                                                                                                                                         |      |      |      |
| Encoding:        | 00                                                                                                                                                        | 0100 | dfff | ffff |
| Description:     | Inclusive OR the W register with register 'f'. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'. |      |      |      |
| Words:           | 1                                                                                                                                                         |      |      |      |
| Cycles:          | 1                                                                                                                                                         |      |      |      |
| Example          | IORWF REG1, 0                                                                                                                                             |      |      |      |
|                  | Before Instruction                                                                                                                                        |      |      |      |
|                  | REG1 = 0x13                                                                                                                                               |      |      |      |
|                  | W = 0x91                                                                                                                                                  |      |      |      |
|                  | After Instruction                                                                                                                                         |      |      |      |
|                  | REG1 = 0x13                                                                                                                                               |      |      |      |
|                  | W = 0x93                                                                                                                                                  |      |      |      |
|                  | Z = 1                                                                                                                                                     |      |      |      |

## MOVF Move f

|                  |                                                                                                                                                                                                                                                                     |      |      |      |      |
|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ <i>label</i> ]    MOVF    f,d                                                                                                                                                                                                                                     |      |      |      |      |
| Operands:        | $0 \leq f \leq 127$<br>$d \in [0,1]$                                                                                                                                                                                                                                |      |      |      |      |
| Operation:       | (f) $\rightarrow$ (dest)                                                                                                                                                                                                                                            |      |      |      |      |
| Status Affected: | Z                                                                                                                                                                                                                                                                   |      |      |      |      |
| Encoding:        | <table border="1"><tr><td>00</td><td>1000</td><td>dfff</td><td>ffff</td></tr></table>                                                                                                                                                                               | 00   | 1000 | dfff | ffff |
| 00               | 1000                                                                                                                                                                                                                                                                | dfff | ffff |      |      |
| Description:     | The contents of register 'f' is moved to a destination dependent upon the status of 'd'. If 'd' = 0, destination is W register. If 'd' = 1, the destination is file register 'f' itself. 'd' = 1 is useful to test a file register since status flag Z is affected. |      |      |      |      |
| Words:           | 1                                                                                                                                                                                                                                                                   |      |      |      |      |
| Cycles:          | 1                                                                                                                                                                                                                                                                   |      |      |      |      |
| Example          | MOVF        REG1, 0<br><br>After Instruction<br>W = value in REG1 register<br>Z = 1                                                                                                                                                                                 |      |      |      |      |

# PIC16F716

## MOVWF Move W to f

|                  |                                                                                                                 |      |      |      |      |
|------------------|-----------------------------------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ <i>label</i> ] MOVWF f                                                                                        |      |      |      |      |
| Operands:        | $0 \leq f \leq 127$                                                                                             |      |      |      |      |
| Operation:       | (W) $\rightarrow$ (f)                                                                                           |      |      |      |      |
| Status Affected: | None                                                                                                            |      |      |      |      |
| Encoding:        | <table border="1"><tr><td>00</td><td>0000</td><td>1fff</td><td>ffff</td></tr></table>                           | 00   | 0000 | 1fff | ffff |
| 00               | 0000                                                                                                            | 1fff | ffff |      |      |
| Description:     | Move data from W register to register 'f'.                                                                      |      |      |      |      |
| Words:           | 1                                                                                                               |      |      |      |      |
| Cycles:          | 1                                                                                                               |      |      |      |      |
| Example          | MOVWF REG1<br><br>Before Instruction<br>REG1 = 0xFF<br>W = 0x4F<br>After Instruction<br>REG1 = 0x4F<br>W = 0x4F |      |      |      |      |

## OPTION Load Option Register

|                  |                                                                                                                                                                                                                                                                                       |      |      |      |
|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|------|------|
| Syntax:          | [ <i>label</i> ]    OPTION                                                                                                                                                                                                                                                            |      |      |      |
| Operands:        | None                                                                                                                                                                                                                                                                                  |      |      |      |
| Operation:       | (W) → OPTION                                                                                                                                                                                                                                                                          |      |      |      |
| Status Affected: | None                                                                                                                                                                                                                                                                                  |      |      |      |
| Encoding:        | 00                                                                                                                                                                                                                                                                                    | 0000 | 0110 | 0010 |
| Description:     | <p>The contents of the W register are loaded in the OPTION register. This instruction is supported for code compatibility with PIC16C5X products. Since OPTION is a readable/writable register, the user can directly address it. Using only register instruction such as MOVWF .</p> |      |      |      |
| Words:           | 1                                                                                                                                                                                                                                                                                     |      |      |      |
| Cycles:          | 1                                                                                                                                                                                                                                                                                     |      |      |      |
| Example          |                                                                                                                                                                                                                                                                                       |      |      |      |

**To maintain upward compatibility with future PICmicro<sup>®</sup> products, do not use this instruction.**

## NOP No Operation

|                  |                      |      |      |      |
|------------------|----------------------|------|------|------|
| Syntax:          | [ <i>label</i> ] NOP |      |      |      |
| Operands:        | None                 |      |      |      |
| Operation:       | No operation         |      |      |      |
| Status Affected: | None                 |      |      |      |
| Encoding:        | 00                   | 0000 | 0xx0 | 0000 |
| Description:     | No operation.        |      |      |      |
| Words:           | 1                    |      |      |      |
| Cycles:          | 1                    |      |      |      |
| Example          | NOP                  |      |      |      |

## RETFIE Return from Interrupt

|                  |                                                                                                                                                                                                    |      |      |      |      |
|------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ <i>label</i> ] RETFIE                                                                                                                                                                            |      |      |      |      |
| Operands:        | None                                                                                                                                                                                               |      |      |      |      |
| Operation:       | TOS → PC,<br>1 → GIE                                                                                                                                                                               |      |      |      |      |
| Status Affected: | None                                                                                                                                                                                               |      |      |      |      |
| Encoding:        | <table border="1"><tr><td>00</td><td>0000</td><td>0000</td><td>1001</td></tr></table>                                                                                                              | 00   | 0000 | 0000 | 1001 |
| 00               | 0000                                                                                                                                                                                               | 0000 | 1001 |      |      |
| Description:     | Return from Interrupt. Stack is POPed and Top of Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a two-cycle instruction. |      |      |      |      |
| Words:           | 1                                                                                                                                                                                                  |      |      |      |      |
| Cycles:          | 2                                                                                                                                                                                                  |      |      |      |      |
| Example          | RETFIE<br><br>After Interrupt<br>PC = TOS<br>GIE = 1                                                                                                                                               |      |      |      |      |

## RETLW Return with Literal in W

|                  |                                                                                                                                                                                                                                                                         |      |      |      |
|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|------|------|
| Syntax:          | [ <i>label</i> ] RETLW k                                                                                                                                                                                                                                                |      |      |      |
| Operands:        | $0 \leq k \leq 255$                                                                                                                                                                                                                                                     |      |      |      |
| Operation:       | $k \rightarrow (W)$ ;<br>$TOS \rightarrow PC$                                                                                                                                                                                                                           |      |      |      |
| Status Affected: | None                                                                                                                                                                                                                                                                    |      |      |      |
| Encoding:        | 11                                                                                                                                                                                                                                                                      | 01xx | kkkk | kkkk |
| Description:     | The W register is loaded with the eight bit literal 'k'. The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.                                                                                                 |      |      |      |
| Words:           | 1                                                                                                                                                                                                                                                                       |      |      |      |
| Cycles:          | 2                                                                                                                                                                                                                                                                       |      |      |      |
| Example          | CALL TABLE;W contains table<br>;offset value<br>• ;W now has table value<br>•<br>•<br>TABLE ADDWF PC;W = offset<br>RETLW k1;Begin table<br>RETLW k2;<br>•<br>•<br>•<br>RETLW kn; End of table<br>Before Instruction<br>W = 0x07<br>After Instruction<br>W = value of k8 |      |      |      |

## RLF Rotate Left f through Carry

|                  |                                                                                                                                                                                                                                                                                            |      |      |      |      |
|------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ <i>label</i> ] RLF f,d                                                                                                                                                                                                                                                                   |      |      |      |      |
| Operands:        | $0 \leq f \leq 127$<br>$d \in [0,1]$                                                                                                                                                                                                                                                       |      |      |      |      |
| Operation:       | See description below                                                                                                                                                                                                                                                                      |      |      |      |      |
| Status Affected: | C                                                                                                                                                                                                                                                                                          |      |      |      |      |
| Encoding:        | <table border="1"><tr><td>00</td><td>1101</td><td>dfff</td><td>ffff</td></tr></table>                                                                                                                                                                                                      | 00   | 1101 | dfff | ffff |
| 00               | 1101                                                                                                                                                                                                                                                                                       | dfff | ffff |      |      |
| Description:     | <p>The contents of register 'f' are rotated one bit to the left through the Carry Flag. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is stored back in register 'f'.</p>  |      |      |      |      |
| Words:           | 1                                                                                                                                                                                                                                                                                          |      |      |      |      |
| Cycles:          | 1                                                                                                                                                                                                                                                                                          |      |      |      |      |
| Example          | <pre>RLF REG1, 0</pre> <p>Before Instruction</p> <pre>REG1=1110 0110 C = 0</pre> <p>After Instruction</p> <pre>REG1=1110 0110 W = 1100 1100 C = 1</pre>                                                                                                                                    |      |      |      |      |

## RETURN Return from Subroutine

|                  |                                                                                                                                                |      |      |      |      |
|------------------|------------------------------------------------------------------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ <i>label</i> ]    RETURN                                                                                                                     |      |      |      |      |
| Operands:        | None                                                                                                                                           |      |      |      |      |
| Operation:       | TOS → PC                                                                                                                                       |      |      |      |      |
| Status Affected: | None                                                                                                                                           |      |      |      |      |
| Encoding:        | <table border="1"><tr><td>00</td><td>0000</td><td>0000</td><td>1000</td></tr></table>                                                          | 00   | 0000 | 0000 | 1000 |
| 00               | 0000                                                                                                                                           | 0000 | 1000 |      |      |
| Description:     | Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two-cycle instruction. |      |      |      |      |
| Words:           | 1                                                                                                                                              |      |      |      |      |
| Cycles:          | 2                                                                                                                                              |      |      |      |      |
| Example          | RETURN<br><br>After Interrupt<br>PC = TOS                                                                                                      |      |      |      |      |

# PIC16F716

## RRF Rotate Right f through Carry

Syntax: `[label] RRF f,d`

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

Operation: See description below

Status Affected: C

Encoding: 

|    |      |      |      |
|----|------|------|------|
| 00 | 1100 | dfff | ffff |
|----|------|------|------|

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



Words: 1

Cycles: 1

Example `RRF REG1, 0`

Before Instruction

`REG1 = 1110 0110`

`C = 0`

After Instruction

`REG1 = 1110 0110`

`W = 0111 0011`

`C = 0`

## SLEEP

Syntax: `[label] SLEEP`

Operands: None

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

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

Encoding: 

|    |      |      |      |
|----|------|------|------|
| 00 | 0000 | 0110 | 0011 |
|----|------|------|------|

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

Words: 1

Cycles: 1

Example: `SLEEP`

## SUBLW Subtract W from Literal

Syntax: `[label] SUBLW k`

Operands:  $0 \leq k \leq 255$

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

Status Affected: C, DC, Z

Encoding: 

|    |      |      |      |
|----|------|------|------|
| 11 | 110x | kkkk | kkkk |
|----|------|------|------|

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

Words: 1

Cycles: 1

Example 1: `SUBLW 0x02`

Before Instruction

`W = 1`

`C = ?`

After Instruction

`W = 1`

`C = 1`; result is positive

Example 2: Before Instruction

`W = 2`

`C = ?`

After Instruction

`W = 0`

`C = 1`; result is zero

Example 3: Before Instruction

`W = 3`

`C = ?`

After Instruction

`W = 0xFF`

`C = 0`; result is negative

## SUBWF Subtract W from f

Syntax: `[label] SUBWF f,d`

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

Operation:  $(f) - (W) \rightarrow (\text{dest})$

Status Affected: C, DC, Z

Encoding:

|    |      |      |      |
|----|------|------|------|
| 00 | 0010 | dfff | ffff |
|----|------|------|------|

Description: Subtract (2's complement method) W register from register 'f'. If 'd' is 0 the result is stored in the W register. If 'd' is 1 the result is stored back in register 'f'.

Words: 1

Cycles: 1

Example 1: `SUBWF REG1, 1`

Before Instruction

REG1 = 3  
W = 2  
C = ?

After Instruction

REG1 = 1  
W = 2  
C = 1; result is positive  
Z = 0  
DC = 1

Example 2: Before Instruction

REG1 = 2  
W = 2  
C = ?

After Instruction

REG1 = 0  
W = 2  
C = 1; result is zero  
Z = DC = 1

Example 3: Before Instruction

REG1 = 1  
W = 2  
C = ?

After Instruction

REG1 = 0xFF  
W = 2  
C = 0; result is negative  
Z = DC = 0

## SWAPF Swap Nibbles in f

Syntax: `[label] SWAPF f,d`

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

Operation:  $(f<3:0>) \rightarrow (\text{dest}<7:4>),$   
 $(f<7:4>) \rightarrow (\text{dest}<3:0>)$

Status Affected: None

Encoding:

|    |      |      |      |
|----|------|------|------|
| 00 | 1110 | dfff | ffff |
|----|------|------|------|

Description: The upper and lower nibbles of register 'f' are exchanged. If 'd' is 0 the result is placed in W register. If 'd' is 1 the result is placed in register 'f'.

Words: 1

Cycles: 1

Example `SWAPF REG1, 0`

Before Instruction

REG1 = 0xA5

After Instruction

REG1 = 0xA5  
W = 0x5A

| TRIS             | Load TRIS Register                                                                                                                                                  |      |      |      |      |
|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|------|------|------|
| Syntax:          | [ label ] TRIS f                                                                                                                                                    |      |      |      |      |
| Operands:        | $5 \leq f \leq 6$                                                                                                                                                   |      |      |      |      |
| Operation:       | (W) → TRIS register f;                                                                                                                                              |      |      |      |      |
| Status Affected: | None                                                                                                                                                                |      |      |      |      |
| Encoding:        | <table><tr><td>00</td><td>0000</td><td>0110</td><td>0fff</td></tr></table>                                                                                          | 00   | 0000 | 0110 | 0fff |
| 00               | 0000                                                                                                                                                                | 0110 | 0fff |      |      |
| Description:     | The instruction is supported for code compatibility with the PIC16C5X products. Since TRIS registers are readable and writable, the user can directly address them. |      |      |      |      |
| Words:           | 1                                                                                                                                                                   |      |      |      |      |
| Cycles:          | 1                                                                                                                                                                   |      |      |      |      |
| Example          | <div><b>To maintain upward compatibility with future PICmicro<sup>®</sup> products, do not use this instruction.</b></div>                                          |      |      |      |      |

# PIC16F716

---

## **XORLW                      Exclusive OR Literal with W**

Syntax:                    [*label*] XORLW *k*

Operands:                 $0 \leq k \leq 255$

Operation:                (W) .XOR. *k*  $\rightarrow$  (W)

Status Affected:        Z

Encoding:                

|    |      |      |      |
|----|------|------|------|
| 11 | 1010 | kkkk | kkkk |
|----|------|------|------|

Description:              The contents of the W register are XOR'ed with the eight bit literal '*k*'. The result is placed in the W register.

Words:                    1

Cycles:                   1

Example:                  XORLW    0xAF

Before Instruction

W = 0xB5

After Instruction

W = 0x1A

## **XORWF                    Exclusive OR W with f**

Syntax:                    [*label*] XORWF *f*,*d*

Operands:                 $0 \leq f \leq 127$

$d \in [0,1]$

Operation:                (W) .XOR. (*f*)  $\rightarrow$  (*dest*)

Status Affected:        Z

Encoding:                

|    |      |      |      |
|----|------|------|------|
| 00 | 0110 | dfff | ffff |
|----|------|------|------|

Description:              Exclusive OR the contents of the W register with register '*f*'. If '*d*' is 0 the result is stored in the W register. If '*d*' is 1 the result is stored back in register '*f*'.

Words:                    1

Cycles:                   1

Example                  XORWF    REG1, 1

Before Instruction

REG1 = 0xAF

W     = 0xB5

After Instruction

REG1 = 0x1A

W     = 0xB5

## 11.0 DEVELOPMENT SUPPORT

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

- Integrated Development Environment
  - MPLAB® IDE Software
- Assemblers/Compilers/Linkers
  - MPASM™ Assembler
  - MPLAB C17 and MPLAB C18 C Compilers
  - MPLINK™ Object Linker/  
MPLIB™ Object Librarian
  - MPLAB C30 C Compiler
  - MPLAB ASM30 Assembler/Linker/Library
- Simulators
  - MPLAB SIM Software Simulator
  - MPLAB dsPIC30 Software Simulator
- Emulators
  - MPLAB ICE 2000 In-Circuit Emulator
  - MPLAB ICE 4000 In-Circuit Emulator
- In-Circuit Debugger
  - MPLAB ICD 2
- Device Programmers
  - PRO MATE® II Universal Device Programmer
  - PICSTART® Plus Development Programmer
- Low Cost Demonstration Boards
  - PICDEM™ 1 Demonstration Board
  - PICDEM.net™ Demonstration Board
  - PICDEM 2 Plus Demonstration Board
  - PICDEM 3 Demonstration Board
  - PICDEM 4 Demonstration Board
  - PICDEM 17 Demonstration Board
  - PICDEM 18R Demonstration Board
  - PICDEM LIN Demonstration Board
  - PICDEM USB Demonstration Board
- Evaluation Kits
  - KEELOQ®
  - PICDEM MSC
  - microID®
  - CAN
  - PowerSmart®
  - Analog

## 11.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8/16-bit microcontroller market. The MPLAB IDE is a Windows® based application that contains:

- An interface to debugging tools
  - simulator
  - programmer (sold separately)
  - emulator (sold separately)
  - in-circuit debugger (sold separately)
- A full-featured editor with color coded context
- A multiple project manager
- Customizable data windows with direct edit of contents
- High level source code debugging
- Mouse over variable inspection
- Extensive on-line help

The MPLAB IDE allows you to:

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

MPLAB IDE supports multiple debugging tools in a single development paradigm, from the cost effective simulators, through low cost in-circuit debuggers, to full-featured emulators. This eliminates the learning curve when upgrading to tools with increasing flexibility and power.

## 11.2 MPASM Assembler

The MPASM assembler is a full-featured, universal macro assembler for all PICmicro MCUs.

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

The MPASM assembler features include:

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

## 11.3 MPLAB C17 and MPLAB C18 C Compilers

The MPLAB C17 and MPLAB C18 Code Development Systems are complete ANSI C compilers for Microchip's PIC17CXXX and PIC18CXXX family of microcontrollers. These compilers provide powerful integration capabilities, superior code optimization and ease of use not found with other compilers.

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

## 11.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK object linker combines relocatable objects created by the MPASM assembler and the MPLAB C17 and MPLAB C18 C compilers. It can link relocatable objects from pre-compiled libraries, using directives from a linker script.

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

The object linker/library features include:

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

## 11.5 MPLAB C30 C Compiler

The MPLAB C30 C compiler is a full-featured, ANSI compliant, optimizing compiler that translates standard ANSI C programs into dsPIC30F assembly language source. The compiler also supports many command-line options and language extensions to take full advantage of the dsPIC30F device hardware capabilities, and afford fine control of the compiler code generator.

MPLAB C30 is distributed with a complete ANSI C standard library. All library functions have been validated and conform to the ANSI C library standard. The library includes functions for string manipulation, dynamic memory allocation, data conversion, time-keeping, and math functions (trigonometric, exponential and hyperbolic). The compiler provides symbolic information for high level source debugging with the MPLAB IDE.

## 11.6 MPLAB ASM30 Assembler, Linker, and Librarian

MPLAB ASM30 assembler produces relocatable machine code from symbolic assembly language for dsPIC30F devices. MPLAB C30 compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire dsPIC30F instruction set
- Support for fixed-point and floating-point data
- Command line interface
- Rich directive set
- Flexible macro language
- MPLAB IDE compatibility

## 11.7 MPLAB SIM Software Simulator

The MPLAB SIM software simulator allows code development in a PC hosted environment by simulating the PICmicro series microcontrollers on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a file, or user defined key press, to any pin. The execution can be performed in Single-Step, Execute Until Break, or Trace mode.

The MPLAB SIM simulator fully supports symbolic debugging using the MPLAB C17 and MPLAB C18 C Compilers, as well as the MPASM assembler. The software simulator offers the flexibility to develop and debug code outside of the laboratory environment, making it an excellent, economical software development tool.

## 11.8 MPLAB SIM30 Software Simulator

The MPLAB SIM30 software simulator allows code development in a PC hosted environment by simulating the dsPIC30F series microcontrollers on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a file, or user defined key press, to any of the pins.

The MPLAB SIM30 simulator fully supports symbolic debugging using the MPLAB C30 C Compiler and MPLAB ASM30 assembler. The simulator runs in either a Command Line mode for automated tasks, or from MPLAB IDE. This high speed simulator is designed to debug, analyze and optimize time intensive DSP routines.

## **11.9 MPLAB ICE 2000 High Performance Universal In-Circuit Emulator**

The MPLAB ICE 2000 universal in-circuit emulator is intended to provide the product development engineer with a complete microcontroller design tool set for PICmicro microcontrollers. Software control of the MPLAB ICE 2000 in-circuit emulator is advanced by the MPLAB Integrated Development Environment, which allows editing, building, downloading and source debugging from a single environment.

The MPLAB ICE 2000 is a full-featured emulator system with enhanced trace, trigger and data monitoring features. Interchangeable processor modules allow the system to be easily reconfigured for emulation of different processors. The universal architecture of the MPLAB ICE in-circuit emulator allows expansion to support new PICmicro microcontrollers.

The MPLAB ICE 2000 in-circuit emulator system has been designed as a real-time emulation system with advanced features that are typically found on more expensive development tools. The PC platform and Microsoft® Windows 32-bit operating system were chosen to best make these features available in a simple, unified application.

## **11.10 MPLAB ICE 4000 High Performance Universal In-Circuit Emulator**

The MPLAB ICE 4000 universal in-circuit emulator is intended to provide the product development engineer with a complete microcontroller design tool set for high-end PICmicro microcontrollers. Software control of the MPLAB ICE in-circuit emulator is provided by the MPLAB Integrated Development Environment, which allows editing, building, downloading and source debugging from a single environment.

The MPLAB ICE 4000 is a premium emulator system, providing the features of MPLAB ICE 2000, but with increased emulation memory and high speed performance for dsPIC30F and PIC18XXX devices. Its advanced emulator features include complex triggering and timing, up to 2 Mb of emulation memory, and the ability to view variables in real-time.

The MPLAB ICE 4000 in-circuit emulator system has been designed as a real-time emulation system with advanced features that are typically found on more expensive development tools. The PC platform and Microsoft Windows 32-bit operating system were chosen to best make these features available in a simple, unified application.

## **11.11 MPLAB ICD 2 In-Circuit Debugger**

Microchip's In-Circuit Debugger, MPLAB ICD 2, is a powerful, low cost, run-time development tool, connecting to the host PC via an RS-232 or high speed USB interface. This tool is based on the Flash PICmicro MCUs and can be used to develop for these and other PICmicro microcontrollers. The MPLAB ICD 2 utilizes the in-circuit debugging capability built into the Flash devices. This feature, along with Microchip's In-Circuit Serial Programming™ (ICSP™) protocol, offers cost effective in-circuit Flash debugging from the graphical user interface of the MPLAB Integrated Development Environment. This enables a designer to develop and debug source code by setting breakpoints, single-stepping and watching variables, CPU status and peripheral registers. Running at full speed enables testing hardware and applications in real-time. MPLAB ICD 2 also serves as a development programmer for selected PICmicro devices.

## **11.12 PRO MATE II Universal Device Programmer**

The PRO MATE II is a universal, CE compliant device programmer with programmable voltage verification at VDDMIN and VDDMAX for maximum reliability. It features an LCD display for instructions and error messages and a modular detachable socket assembly to support various package types. In Stand-alone mode, the PRO MATE II device programmer can read, verify, and program PICmicro devices without a PC connection. It can also set code protection in this mode.

## **11.13 PICSTART Plus Development Programmer**

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

# PIC16F716

---

## 11.14 PICDEM 1 PICmicro Demonstration Board

The PICDEM 1 demonstration board demonstrates the capabilities of the 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 sample microcontrollers provided with the PICDEM 1 demonstration board can be programmed with a PRO MATE II device programmer, or a PICSTART Plus development programmer. The PICDEM 1 demonstration board can be connected to the MPLAB ICE in-circuit emulator for testing. A prototype area extends the circuitry for additional application components. Features include an RS-232 interface, a potentiometer for simulated analog input, push button switches and eight LEDs.

## 11.15 PICDEM.net Internet/Ethernet Demonstration Board

The PICDEM.net demonstration board is an Internet/Ethernet demonstration board using the PIC18F452 microcontroller and TCP/IP firmware. The board supports any 40-pin DIP device that conforms to the standard pinout used by the PIC16F877 or PIC18C452. This kit features a user friendly TCP/IP stack, web server with HTML, a 24L256 Serial EEPROM for Xmodem download to web pages into Serial EEPROM, ICSP/MPLAB ICD 2 interface connector, an Ethernet interface, RS-232 interface, and a 16 x 2 LCD display. Also included is the book and CD-ROM *"TCP/IP Lean, Web Servers for Embedded Systems,"* by Jeremy Bentham

## 11.16 PICDEM 2 Plus Demonstration Board

The PICDEM 2 Plus demonstration board supports many 18-, 28-, and 40-pin microcontrollers, including PIC16F87X and PIC18FXX2 devices. All the necessary hardware and software is included to run the demonstration programs. The sample microcontrollers provided with the PICDEM 2 demonstration board can be programmed with a PRO MATE II device programmer, PICSTART Plus development programmer, or MPLAB ICD 2 with a Universal Programmer Adapter. The MPLAB ICD 2 and MPLAB ICE in-circuit emulators may also be used with the PICDEM 2 demonstration board to test firmware. A prototype area extends the circuitry for additional application components. Some of the features include an RS-232 interface, a 2 x 16 LCD display, a piezo speaker, an on-board temperature sensor, four LEDs, and sample PIC18F452 and PIC16F877 Flash microcontrollers.

## 11.17 PICDEM 3 PIC16C92X Demonstration Board

The PICDEM 3 demonstration board supports the PIC16C923 and PIC16C924 in the PLCC package. All the necessary hardware and software is included to run the demonstration programs.

## 11.18 PICDEM 4 8/14/18-Pin Demonstration Board

The PICDEM 4 can be used to demonstrate the capabilities of the 8-, 14-, and 18-pin PIC16XXXX and PIC18XXXX MCUs, including the PIC16F818/819, PIC16F87/88, PIC16F62XA and the PIC18F1320 family of microcontrollers. PICDEM 4 is intended to showcase the many features of these low pin count parts, including LIN and Motor Control using ECCP. Special provisions are made for low power operation with the super capacitor circuit, and jumpers allow on-board hardware to be disabled to eliminate current draw in this mode. Included on the demo board are provisions for Crystal, RC or Canned Oscillator modes, a five volt regulator for use with a nine volt wall adapter or battery, DB-9 RS-232 interface, ICD connector for programming via ICSP and development with MPLAB ICD 2, 2x16 liquid crystal display, PCB footprints for H-Bridge motor driver, LIN transceiver and EEPROM. Also included are: header for expansion, eight LEDs, four potentiometers, three push buttons and a prototyping area. Included with the kit is a PIC16F627A and a PIC18F1320. Tutorial firmware is included along with the User's Guide.

## 11.19 PICDEM 17 Demonstration Board

The PICDEM 17 demonstration board is an evaluation board that demonstrates the capabilities of several Microchip microcontrollers, including PIC17C752, PIC17C756A, PIC17C762 and PIC17C766. A programmed sample is included. The PRO MATE II device programmer, or the PICSTART Plus development programmer, can be used to reprogram the device for user tailored application development. The PICDEM 17 demonstration board supports program download and execution from external on-board Flash memory. A generous prototype area is available for user hardware expansion.

## **11.20 PICDEM 18R PIC18C601/801 Demonstration Board**

The PICDEM 18R demonstration board serves to assist development of the PIC18C601/801 family of Microchip microcontrollers. It provides hardware implementation of both 8-bit Multiplexed/De-multiplexed and 16-bit memory modes. The board includes 2 Mb external Flash memory and 128 Kb SRAM memory, as well as serial EEPROM, allowing access to the wide range of memory types supported by the PIC18C601/801.

## **11.21 PICDEM LIN PIC16C43X Demonstration Board**

The powerful LIN hardware and software kit includes a series of boards and three PICmicro microcontrollers. The small footprint PIC16C432 and PIC16C433 are used as slaves in the LIN communication and feature on-board LIN transceivers. A PIC16F874 Flash microcontroller serves as the master. All three microcontrollers are programmed with firmware to provide LIN bus communication.

## **11.22 PICKIT™ 1 Flash Starter Kit**

A complete "development system in a box", the PICKIT Flash Starter Kit includes a convenient multi-section board for programming, evaluation, and development of 8/14-pin Flash PIC® microcontrollers. Powered via USB, the board operates under a simple Windows GUI. The PICKIT 1 Starter Kit includes the user's guide (on CD ROM), PICKIT 1 tutorial software and code for various applications. Also included are MPLAB® IDE (Integrated Development Environment) software, software and hardware "Tips 'n Tricks for 8-pin Flash PIC® Microcontrollers" Handbook and a USB Interface Cable. Supports all current 8/14-pin Flash PIC microcontrollers, as well as many future planned devices.

## **11.23 PICDEM USB PIC16C7X5 Demonstration Board**

The PICDEM USB Demonstration Board shows off the capabilities of the PIC16C745 and PIC16C765 USB microcontrollers. This board provides the basis for future USB products.

## **11.24 Evaluation and Programming Tools**

In addition to the PICDEM series of circuits, Microchip has a line of evaluation kits and demonstration software for these products.

- KEELOQ evaluation and programming tools for Microchip's HCS Secure Data Products
- CAN developers kit for automotive network applications
- Analog design boards and filter design software
- PowerSmart battery charging evaluation/calibration kits
- IrDA® development kit
- microID development and rfLab™ development software
- SEEVAL® designer kit for memory evaluation and endurance calculations
- PICDEM MSC demo boards for Switching mode power supply, high power IR driver, delta sigma ADC, and flow rate sensor

Check the Microchip web page and the latest Product Line Card for the complete list of demonstration and evaluation kits.

# PIC16F716

---

NOTES:

## 12.0 ELECTRICAL CHARACTERISTICS

### Absolute Maximum Ratings<sup>(†)</sup>

|                                                                                                                        |                                   |
|------------------------------------------------------------------------------------------------------------------------|-----------------------------------|
| Ambient temperature under bias .....                                                                                   | -55°C to +125°C                   |
| Storage temperature .....                                                                                              | -65°C to +150°C                   |
| Voltage on any pin with respect to V <sub>SS</sub> (except V <sub>DD</sub> , $\overline{\text{MCLR}}$ , and RA4) ..... | -0.3V to (V <sub>DD</sub> + 0.3V) |
| Voltage on V <sub>DD</sub> with respect to V <sub>SS</sub> .....                                                       | -0.3V to +7.5V                    |
| Voltage on $\overline{\text{MCLR}}$ with respect to V <sub>SS</sub> ( <b>Note 2</b> ) .....                            | 0V to +13.25V                     |
| Voltage on RA4 with respect to V <sub>SS</sub> .....                                                                   | 0V to +8.5V                       |
| Total power dissipation ( <b>Note 1</b> ) (PDIP and SOIC) .....                                                        | 1.0W                              |
| Total power dissipation ( <b>Note 1</b> ) (SSOP) .....                                                                 | 0.65W                             |
| Maximum current out of V <sub>SS</sub> pin .....                                                                       | 300 mA                            |
| Maximum current into V <sub>DD</sub> pin .....                                                                         | 250 mA                            |
| Input clamp current, I <sub>IK</sub> (V <sub>I</sub> < 0 or V <sub>I</sub> > V <sub>DD</sub> ) .....                   | ±20 mA                            |
| Output clamp current, I <sub>OK</sub> (V <sub>O</sub> < 0 or V <sub>O</sub> > V <sub>DD</sub> ) .....                  | ±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:  $P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$

**2:** Voltage spikes below V<sub>SS</sub> at the  $\overline{\text{MCLR}}$ /V<sub>PP</sub> 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  $\overline{\text{MCLR}}$ /V<sub>PP</sub> pin rather than pulling this pin directly to V<sub>SS</sub>.

† 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.

# PIC16F716

FIGURE 12-1: PIC16F716 VOLTAGE-FREQUENCY GRAPH,  $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}^{(1)}$

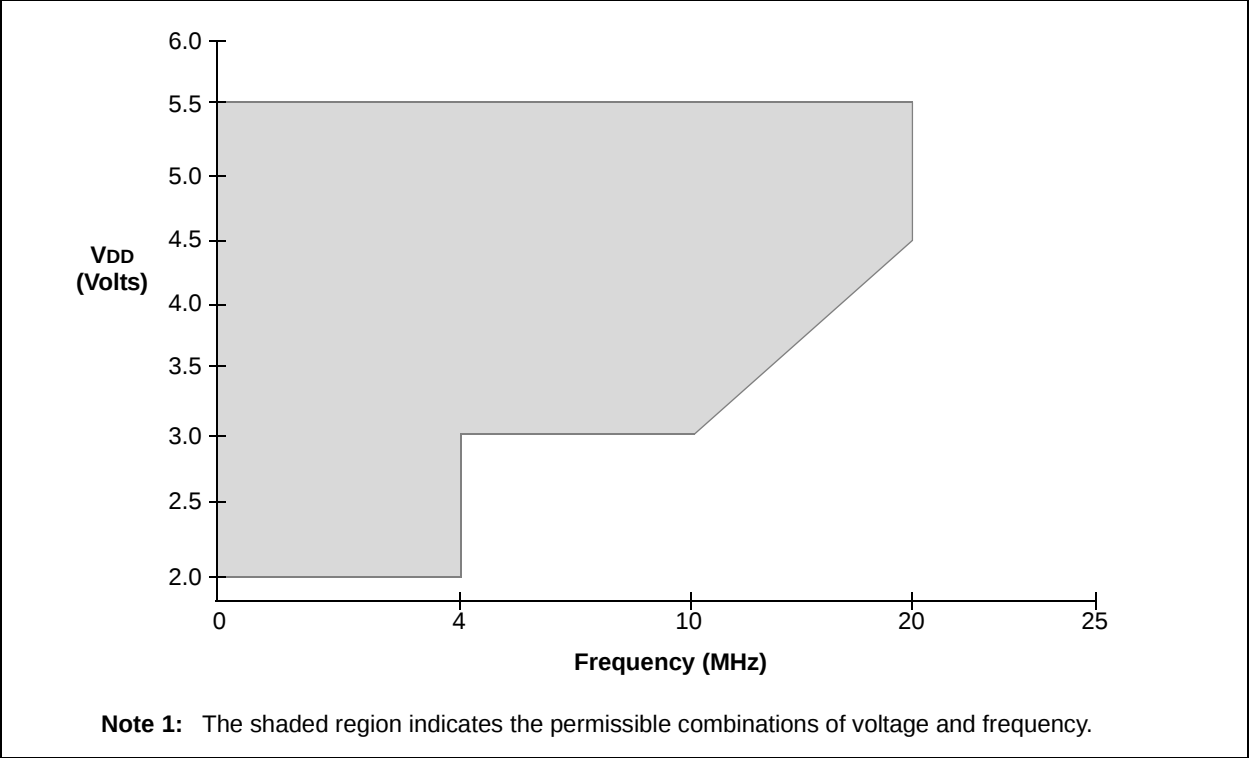
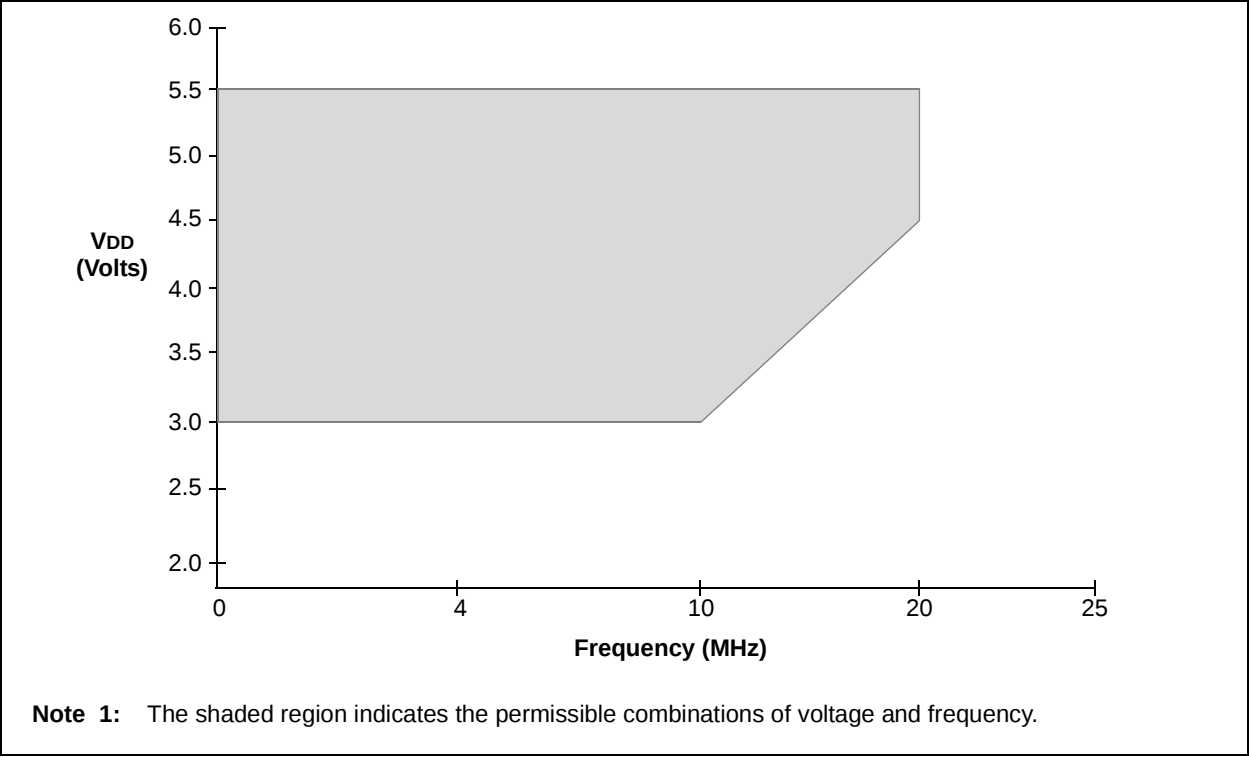


FIGURE 12-2: PIC16F716 VOLTAGE-FREQUENCY GRAPH,  $85^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}^{(1)}$



## 12.1 DC Characteristics: PIC16F716 (Industrial, Extended)

| DC CHARACTERISTICS |      |                                                            | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended |        |            |        |                                                                                                                              |
|--------------------|------|------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|------------|--------|------------------------------------------------------------------------------------------------------------------------------|
| Param No.          | Sym  | Characteristic                                             | Min                                                                                                                                                                                                                              | Typ†   | Max        | Units  | Conditions                                                                                                                   |
| D001<br>D001A      | VDD  | Supply Voltage                                             |                                                                                                                                                                                                                                  |        |            |        |                                                                                                                              |
|                    |      |                                                            | 2.0<br>3.0                                                                                                                                                                                                                       | —<br>— | 5.5<br>5.5 | V<br>V | Industrial<br>Extended                                                                                                       |
| D002*              | VDR  | RAM Data Retention Voltage <sup>(1)</sup>                  | —                                                                                                                                                                                                                                | 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*<br>D004A*    | SVDD | VDD Rise Rate to ensure internal Power-on Reset signal     | 0.05<br>TBD                                                                                                                                                                                                                      | —<br>— | —<br>—     | V/ms   | PWRT enabled ( <u>PWRTE</u> bit clear)<br>PWRT disabled ( <u>PWRTE</u> bit set)<br>See section on Power-on Reset for details |
| D005               | VBOR | Brown-out Reset voltage trip point                         |                                                                                                                                                                                                                                  |        |            |        |                                                                                                                              |
|                    |      |                                                            | 3.65                                                                                                                                                                                                                             | 4.0    | 4.35       | V      | BOREN bit set, BOR bit = '1'                                                                                                 |
|                    |      |                                                            | TBD                                                                                                                                                                                                                              | 2.5    | TBD        | V      | BOREN bit set, BOR bit = '0'                                                                                                 |

\* 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.

# PIC16F716

## 12.2 DC Characteristics: PIC16F716 (Industrial)

| DC CHARACTERISTICS |       |                                          | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ |      |      |       |     |                                     |
|--------------------|-------|------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|------|------|-------|-----|-------------------------------------|
| Para<br>m No.      | Sym   | Characteristic                           | Min                                                                                                                                      | Typ† | Max  | Units | VDD | Conditions                          |
| D001               | VDD   | Supply Voltage                           |                                                                                                                                          |      |      |       |     |                                     |
|                    |       |                                          | 2.0                                                                                                                                      | —    | 5.5  | V     | —   |                                     |
| D020               | IPD   | Power-down Base Current                  |                                                                                                                                          |      |      |       |     |                                     |
|                    |       |                                          | —                                                                                                                                        | 0.1  | 0.8  | μA    | 2.0 | WDT, BOR and T1OSC:<br>disabled     |
|                    |       |                                          | —                                                                                                                                        | 0.1  | 0.85 | μA    | 3.0 |                                     |
|                    |       |                                          | —                                                                                                                                        | 0.2  | 2.7  | μA    | 5.0 |                                     |
| D021               | ΔIMOD | Peripheral Module Current <sup>(1)</sup> |                                                                                                                                          |      |      |       |     |                                     |
|                    |       |                                          | —                                                                                                                                        | 1    | 2.0  | μA    | 2.0 | WDT Current                         |
|                    |       |                                          | —                                                                                                                                        | 2    | 3.5  | μA    | 3.0 |                                     |
|                    |       |                                          | —                                                                                                                                        | 9    | 13.5 | μA    | 5.0 |                                     |
| D022               |       |                                          | —                                                                                                                                        | TBD  | TBD  | μA    | 3.0 | BOR Current                         |
|                    |       |                                          | —                                                                                                                                        | 40   | TBD  | μA    | 4.5 |                                     |
|                    |       |                                          | —                                                                                                                                        | 45   | TBD  | μA    | 5.0 |                                     |
| D025               |       |                                          | —                                                                                                                                        | 1.8  | TBD  | μA    | 2.0 | T1osc Current                       |
|                    |       |                                          | —                                                                                                                                        | 2.6  | TBD  | μA    | 3.0 |                                     |
|                    |       |                                          | —                                                                                                                                        | 3.0  | TBD  | μA    | 5.0 |                                     |
| D010               | IDD   | Supply Current                           |                                                                                                                                          |      |      |       |     |                                     |
|                    |       |                                          | —                                                                                                                                        | 14   | 17   | μA    | 2.0 | Fosc = 32 kHz<br>LP Oscillator mode |
|                    |       |                                          | —                                                                                                                                        | 23   | 28   | μA    | 3.0 |                                     |
| D011               |       |                                          | —                                                                                                                                        | 120  | 160  | μA    | 2.0 | Fosc = 1 MHz<br>XT Oscillator mode  |
|                    |       |                                          | —                                                                                                                                        | 180  | 250  | μA    | 3.0 |                                     |
|                    |       |                                          | —                                                                                                                                        | 290  | 370  | μA    | 5.0 |                                     |
| D012               |       |                                          | —                                                                                                                                        | 220  | 300  | μA    | 2.0 | Fosc = 4 MHz<br>XT Oscillator mode  |
|                    |       |                                          | —                                                                                                                                        | 350  | 470  | μA    | 3.0 |                                     |
|                    |       |                                          | —                                                                                                                                        | 600  | 780  | μA    | 5.0 |                                     |
| D013               |       |                                          | —                                                                                                                                        | 2.1  | 2.9  | mA    | 4.5 | Fosc = 20 MHz<br>HS Oscillator mode |
|                    |       |                                          | —                                                                                                                                        | 2.5  | 3.3  | mA    | 5.0 |                                     |

† 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:** 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. Max values should be used when calculating total current consumption.

**2:** ADC on, not converting.

## 12.3 DC Characteristics: PIC16F716 (Extended)

| DC CHARACTERISTICS |       |                                          | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ |      |     |       |     |                                     |
|--------------------|-------|------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|------|-----|-------|-----|-------------------------------------|
| Param No.          | Sym   | Characteristic                           | Min                                                                                                                                       | Typ† | Max | Units | VDD | Conditions                          |
| D001               | VDD   | Supply Voltage                           |                                                                                                                                           |      |     |       |     |                                     |
|                    |       |                                          | 3.0                                                                                                                                       | —    | 5.5 | V     | —   |                                     |
| D020E              | IPD   | Power-down Base Current                  |                                                                                                                                           |      |     |       |     |                                     |
|                    |       |                                          | —                                                                                                                                         | 0.1  | TBD | μA    | 3.0 | WDT, BOR and T1OSC: disabled        |
|                    |       |                                          | —                                                                                                                                         | 0.2  | TBD | μA    | 5.0 |                                     |
| D021E              | ΔIMOD | Peripheral Module Current <sup>(1)</sup> |                                                                                                                                           |      |     |       |     |                                     |
|                    |       |                                          | —                                                                                                                                         | 2    | TBD | μA    | 3.0 | WDT Current                         |
|                    |       |                                          | —                                                                                                                                         | 9    | TBD | μA    | 5.0 |                                     |
| D022E              |       |                                          | —                                                                                                                                         | TBD  | TBD | μA    | 3.0 | BOR Current                         |
|                    |       |                                          | —                                                                                                                                         | 40   | TBD | μA    | 4.5 |                                     |
|                    |       |                                          | —                                                                                                                                         | 45   | TBD | μA    | 5.0 |                                     |
| D025E              |       |                                          | —                                                                                                                                         | 2.6  | TBD | μA    | 3.0 | T1osc Current                       |
|                    |       |                                          | —                                                                                                                                         | 3.0  | TBD | μA    | 5.0 |                                     |
| D010E              | IDD   | Supply Current                           |                                                                                                                                           |      |     |       |     |                                     |
|                    |       |                                          | —                                                                                                                                         | 21   | TBD | μA    | 3.0 | Fosc = 32 kHz<br>LP Oscillator mode |
|                    |       |                                          | —                                                                                                                                         | 38   | TBD | μA    | 5.0 |                                     |
| D011E              |       |                                          | —                                                                                                                                         | 182  | TBD | μA    | 3.0 | Fosc = 1 MHz<br>XT Oscillator mode  |
|                    |       |                                          | —                                                                                                                                         | 293  | TBD | μA    | 5.0 |                                     |
| D012E              |       |                                          | —                                                                                                                                         | 371  | TBD | μA    | 3.0 | Fosc = 4 MHz<br>XT Oscillator mode  |
|                    |       |                                          | —                                                                                                                                         | 668  | TBD | μA    | 5.0 |                                     |
| D013E              |       |                                          | —                                                                                                                                         | 2.6  | TBD | mA    | 4.5 | Fosc = 20 MHz<br>HS Oscillator mode |
|                    |       |                                          | —                                                                                                                                         | 3    | TBD | mA    | 5.0 |                                     |

† 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:** 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. Max values should be used when calculating total current consumption.

**2:** ADC on, not converting.

# PIC16F716

## 12.4 DC Characteristics: PIC16F716 (Industrial, Extended)

| DC CHARACTERISTICS                                 |       |                                                | Standard Operating Conditions (unless otherwise stated)                                                                                                                                                         |                                                 |                |        |                                           |                                                  |
|----------------------------------------------------|-------|------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------|----------------|--------|-------------------------------------------|--------------------------------------------------|
|                                                    |       |                                                | Operating temperature                    -40°C ≤ TA ≤ +85°C for industrial<br>-40°C ≤ TA ≤ +125°C for extended                                                                                                  |                                                 |                |        |                                           |                                                  |
|                                                    |       |                                                | Operating voltage VDD range as described in DC spec <b>Section 12.1 “DC Characteristics: PIC16F716 (Industrial, Extended)”</b> and <b>Section 12.4 “DC Characteristics: PIC16F716 (Industrial, Extended)”</b> . |                                                 |                |        |                                           |                                                  |
| Param No.                                          | Sym   | Characteristic                                 | Min                                                                                                                                                                                                             | Typ†                                            | Max            | Units  | Conditions                                |                                                  |
| D030<br>D030A<br>D031<br>D032<br>D033              | VIL   | <b>Input Low Voltage</b>                       |                                                                                                                                                                                                                 |                                                 |                |        |                                           |                                                  |
|                                                    |       | I/O ports                                      | VSS                                                                                                                                                                                                             | —                                               | 0.8            | V      | 4.5V ≤ VDD ≤ 5.5V                         |                                                  |
|                                                    |       | with TTL buffer                                | VSS                                                                                                                                                                                                             | —                                               | 0.15 VDD       | V      |                                           | otherwise                                        |
|                                                    |       | with Schmitt Trigger buffer                    | VSS                                                                                                                                                                                                             | —                                               | 0.2 VDD        | V      | <b>(Note1)</b>                            |                                                  |
|                                                    |       | MCLR, OSC1 (in RC mode)                        | VSS                                                                                                                                                                                                             | —                                               | 0.2 VDD        | V      |                                           |                                                  |
|                                                    |       | OSC1 (in HS mode)<br>OSC1 (in XT and LP modes) | VSS                                                                                                                                                                                                             | —                                               | 0.3 VDD<br>0.6 | V<br>V |                                           |                                                  |
| D040<br>D040A<br><br>D041<br>D042<br>D042A<br>D043 | VIH   | <b>Input High Voltage</b>                      |                                                                                                                                                                                                                 |                                                 |                |        |                                           |                                                  |
|                                                    |       | I/O ports                                      | 2.0                                                                                                                                                                                                             | —                                               | VDD            | V      | 4.5V ≤ VDD ≤ 5.5V                         |                                                  |
|                                                    |       | with TTL buffer                                | 0.25 VDD + 0.8V                                                                                                                                                                                                 | —                                               | VDD            | V      |                                           | otherwise                                        |
|                                                    |       | with Schmitt Trigger buffer                    | 0.8 VDD                                                                                                                                                                                                         | —                                               | VDD            | V      | For entire VDD range                      |                                                  |
|                                                    |       | MCLR                                           | 0.8 VDD                                                                                                                                                                                                         | —                                               | VDD            | V      |                                           |                                                  |
|                                                    |       | OSC1 (XT, HS and LP modes)                     | 0.7 VDD                                                                                                                                                                                                         | —                                               | VDD            | V      |                                           | <b>(Note1)</b>                                   |
|                                                    |       | OSC1 (in RC mode)                              | 0.9 VDD                                                                                                                                                                                                         | —                                               | VDD            | V      |                                           |                                                  |
|                                                    |       | D060<br><br><br>D061<br>D063                   | IIL                                                                                                                                                                                                             | <b>Input Leakage Current<sup>(2), (3)</sup></b> |                |        |                                           |                                                  |
| I/O ports                                          | —     |                                                |                                                                                                                                                                                                                 | —                                               | ±1             | μA     | VSS ≤ VPIN ≤ VDD, Pin at high-impedance   |                                                  |
|                                                    | —     |                                                |                                                                                                                                                                                                                 | —                                               | ±500           | nA     |                                           | VSS ≤ VPIN ≤ VDD, Pin configured as analog input |
| MCLR, RA4/T0CKI                                    | —     |                                                |                                                                                                                                                                                                                 | —                                               | ±5             | μA     | VSS ≤ VPIN ≤ VDD                          |                                                  |
| OSC1/CLKIN                                         | —     |                                                |                                                                                                                                                                                                                 | —                                               | ±5             | μA     | VSS ≤ VPIN ≤ VDD, XT, HS and LP osc modes |                                                  |
| D070                                               | IPURB | <b>PORTB weak pull-up current</b>              | 50                                                                                                                                                                                                              | 250                                             | 400            | μA     | VDD = 5V, VPIN = VSS                      |                                                  |
| D080<br><br><br>D083                               | VOL   | <b>Output Low Voltage</b>                      |                                                                                                                                                                                                                 |                                                 |                |        |                                           |                                                  |
|                                                    |       | I/O ports                                      | —                                                                                                                                                                                                               | —                                               | 0.6            | V      | IOL = 8.5 mA, VDD = 4.5V, -40°C to +85°C  |                                                  |
|                                                    |       |                                                | —                                                                                                                                                                                                               | —                                               | 0.6            | V      |                                           | IOL = 7.0 mA, VDD = 4.5V, -40°C to +125°C        |
|                                                    |       | OSC2/CLKOUT (RC Osc mode)                      | —                                                                                                                                                                                                               | —                                               | 0.6            | V      | IOL = 1.6 mA, VDD = 4.5V, -40°C to +85°C  |                                                  |
|                                                    |       |                                                | —                                                                                                                                                                                                               | —                                               | 0.6            | V      |                                           | IOL = 1.2 mA, VDD = 4.5V, -40°C to +125°C        |
| D090<br><br><br>D092                               | VOH   | <b>Output High Voltage</b>                     |                                                                                                                                                                                                                 |                                                 |                |        |                                           |                                                  |
|                                                    |       | I/O ports <sup>(3)</sup>                       | VDD-0.7                                                                                                                                                                                                         | —                                               | —              | V      | IOH = -3.0 mA, VDD = 4.5V, -40°C to +85°C |                                                  |
|                                                    |       |                                                | VDD-0.7                                                                                                                                                                                                         | —                                               | —              | V      |                                           | IOH = -2.5 mA, VDD = 4.5V, -40°C to +125°C       |
|                                                    |       | OSC2/CLKOUT (RC Osc mode)                      | VDD-0.7                                                                                                                                                                                                         | —                                               | —              | V      | IOH = -1.3 mA, VDD = 4.5V, -40°C to +85°C |                                                  |
|                                                    |       |                                                | VDD-0.7                                                                                                                                                                                                         | —                                               | —              | V      |                                           | IOH = -1.0 mA, VDD = 4.5V, -40°C to +125°C       |
| D150*                                              | VOD   | <b>Open-Drain High Voltage</b>                 | —                                                                                                                                                                                                               | —                                               | 8.5            | V      | RA4 pin                                   |                                                  |

\* These parameters are characterized but not tested.

† Data in “Type” 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 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.

| DC CHARACTERISTICS                             |       |                                    |     | Standard Operating Conditions (unless otherwise stated)                                                                                                                                                              |     |       |                                                                   |
|------------------------------------------------|-------|------------------------------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----|-------|-------------------------------------------------------------------|
|                                                |       |                                    |     | Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended                                                |     |       |                                                                   |
|                                                |       |                                    |     | Operating voltage $V_{DD}$ range as described in DC spec <b>Section 12.1 “DC Characteristics: PIC16F716 (Industrial, Extended)”</b> and <b>Section 12.4 “DC Characteristics: PIC16F716 (Industrial, Extended)”</b> . |     |       |                                                                   |
| Param No.                                      | Sym   | Characteristic                     | Min | Typ†                                                                                                                                                                                                                 | Max | Units | Conditions                                                        |
| <b>Capacitive Loading Specs on Output Pins</b> |       |                                    |     |                                                                                                                                                                                                                      |     |       |                                                                   |
| D100                                           | COSC2 | OSC2/CLKOUT pin                    | —   | —                                                                                                                                                                                                                    | 15  | pF    | In XT, HS and LP modes when external clock is used to drive OSC1. |
| D101                                           | Cio   | All I/O pins and OSC2 (in RC mode) | —   | —                                                                                                                                                                                                                    | 50  | pF    |                                                                   |

\* These parameters are characterized but not tested.

† Data in “Type” 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  $\overline{\text{MCLR}}/V_{PP}$  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.

## 12.5 AC (Timing) Characteristics

### 12.5.1 TIMING PARAMETER SYMBOLOGY

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

1. TppS2ppS
2. TppS

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

Lowercase letters (pp) and their meanings:

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

Uppercase letters and their meanings:

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

# PIC16F716

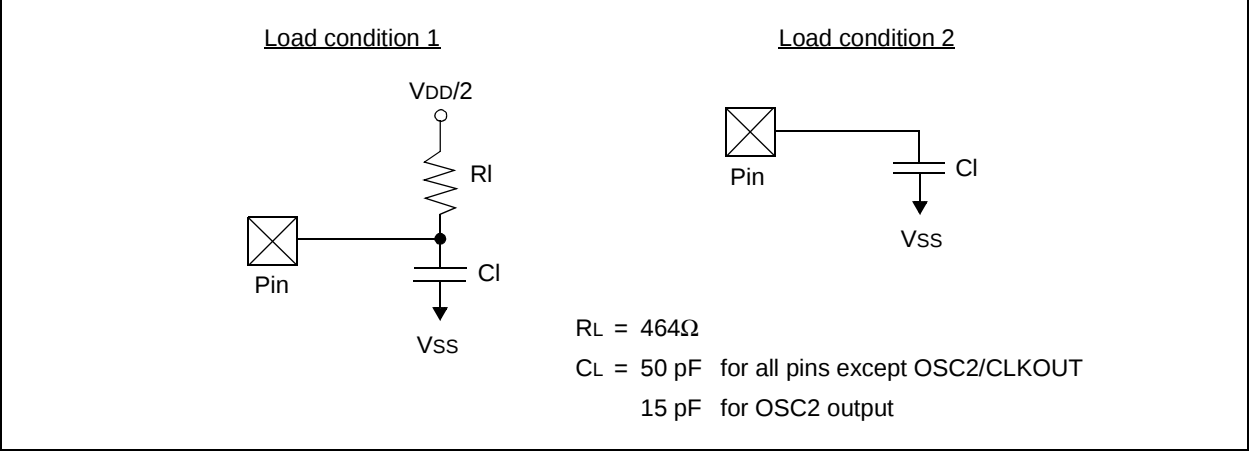
## 12.5.2 TIMING CONDITIONS

The temperature and voltages specified in Table 12-1 apply to all timing specifications, unless otherwise noted. Figure 12-3 specifies the load conditions for the timing specifications.

TABLE 12-1: TEMPERATURE AND VOLTAGE SPECIFICATIONS - AC

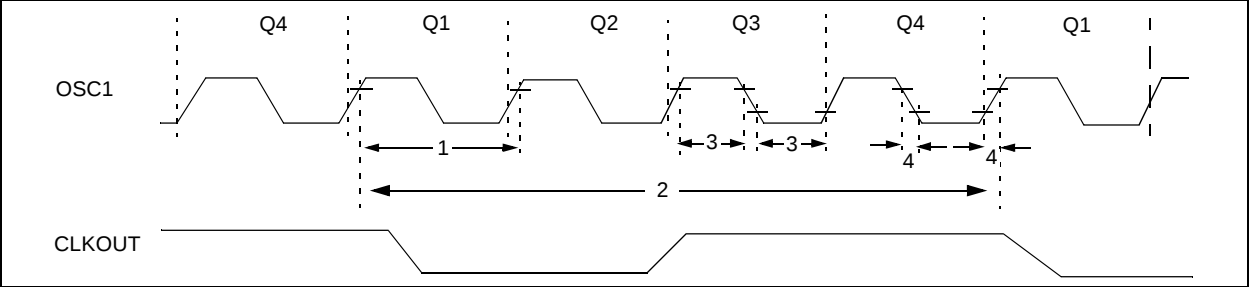
|                    |                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|--------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| AC CHARACTERISTICS | <b>Standard Operating Conditions (unless otherwise stated)</b>                                                                                                                                                                                                                                                                                                                                                                                        |
|                    | Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial<br>$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended<br>Operating voltage $V_{DD}$ range as described in DC spec <b>Section 12.1 “DC Characteristics: PIC16F716 (Industrial, Extended)”</b> and <b>Section 12.4 “DC Characteristics: PIC16F716 (Industrial, Extended)”</b> . LC parts operate for commercial/industrial temp's only. |

FIGURE 12-3: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS



## 12.5.3 TIMING DIAGRAMS AND SPECIFICATIONS

FIGURE 12-4: EXTERNAL CLOCK TIMING



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

| Param No. | Sym        | Characteristic                             | Min | Typ† | Max    | Units | Conditions                           |
|-----------|------------|--------------------------------------------|-----|------|--------|-------|--------------------------------------|
| 1A        | Fosc       | Ext. Clock Input Frequency <sup>(1)</sup>  | DC  | —    | 4      | MHz   | RC and XT Osc modes                  |
|           |            |                                            | DC  | —    | 20     | MHz   | HS Osc mode                          |
|           |            |                                            | DC  | —    | 200    | kHz   | LP Osc mode                          |
|           |            | Oscillator Frequency <sup>(1)</sup>        | 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 <sup>(1)</sup>       | 250 | —    | —      | ns    | RC and XT Osc modes                  |
|           |            |                                            | 50  | —    | —      | ns    | HS Osc mode                          |
|           |            |                                            | 5   | —    | —      | μs    | LP Osc mode                          |
|           |            | Oscillator Period <sup>(1)</sup>           | 250 | —    | —      | ns    | RC Osc mode                          |
|           |            |                                            | 250 | —    | 10,000 | ns    | XT Osc mode                          |
|           |            |                                            | 50  | —    | 250    | ns    | HS Osc mode                          |
|           |            |                                            | 5   | —    | —      | μs    | LP Osc mode                          |
|           |            |                                            |     |      |        |       |                                      |
| 2         | Tcy        | Instruction Cycle Time <sup>(1)</sup>      | 200 | —    | DC     | ns    | T <sub>cy</sub> = 4/F <sub>osc</sub> |
| 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                        |

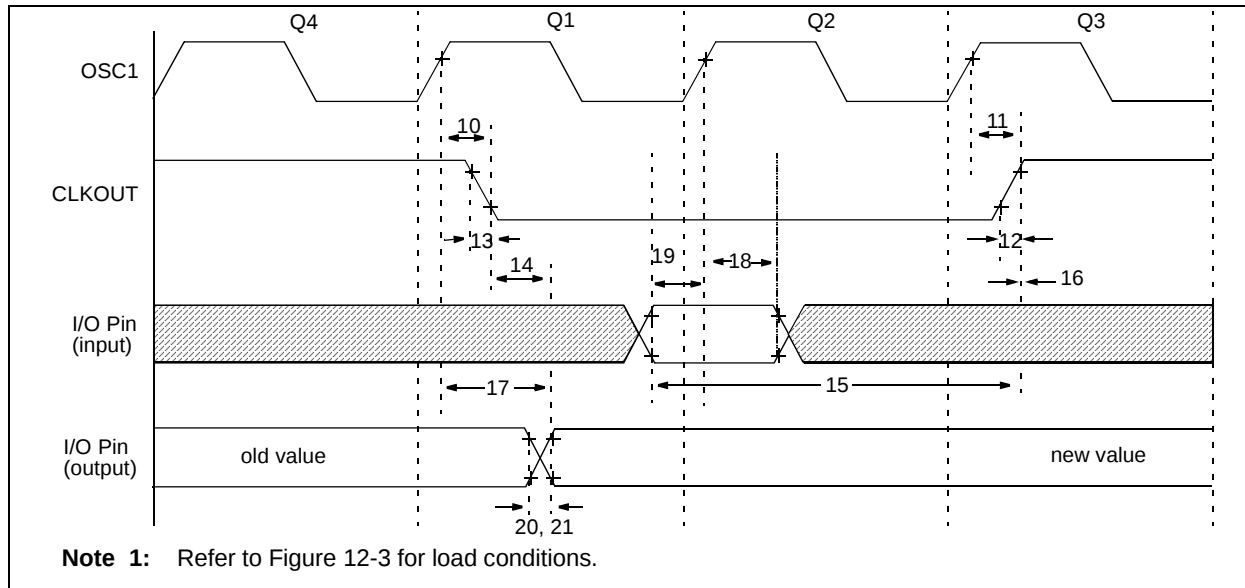
\* These parameters are characterized but not tested.

† 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 (T<sub>cy</sub>) 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.

# PIC16F716

**FIGURE 12-5: CLKOUT AND I/O TIMING**



**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                                | —             | —     | 20  | ns    | (Note 1)   |
| 15*       | TioV2ckH | Port input valid before CLKOUT ↑                          | Tosc + 200    | —     | —   | ns    | (Note 1)   |
| 16*       | TckH2ioI | Port input 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    |            |

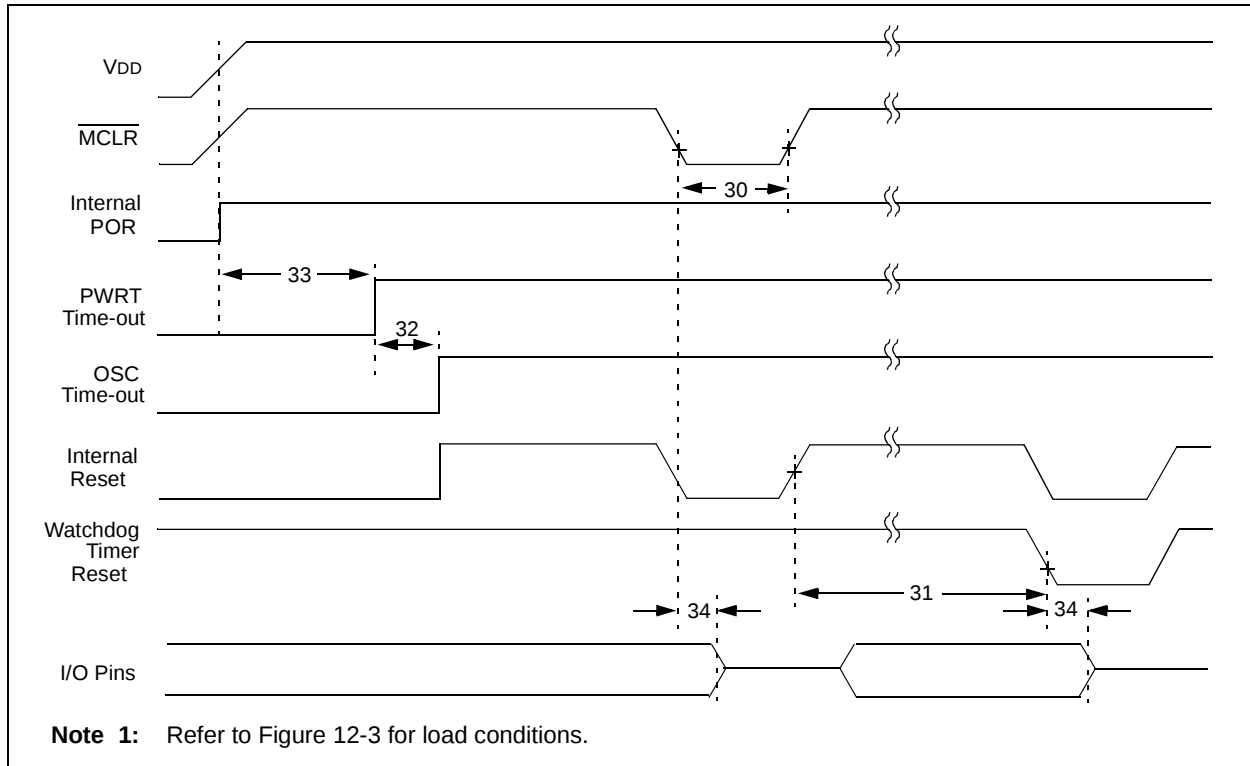
\* These parameters are characterized but not tested.

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

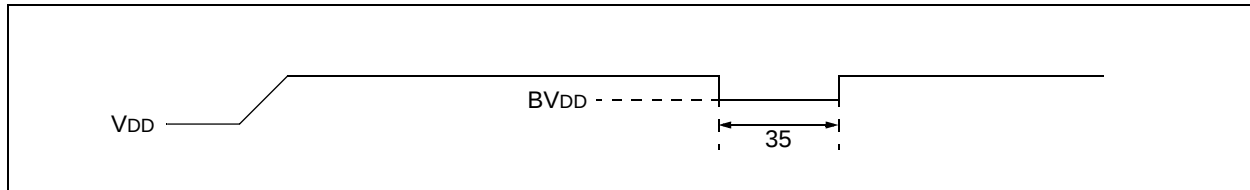
†† These parameters are asynchronous events not related to any internal clock edge.

**Note 1:** Measurements are taken in RC mode where CLKOUT output is 4 x Tosc.

**FIGURE 12-6: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING<sup>(1)</sup>**



**FIGURE 12-7: BROWN-OUT RESET TIMING**



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

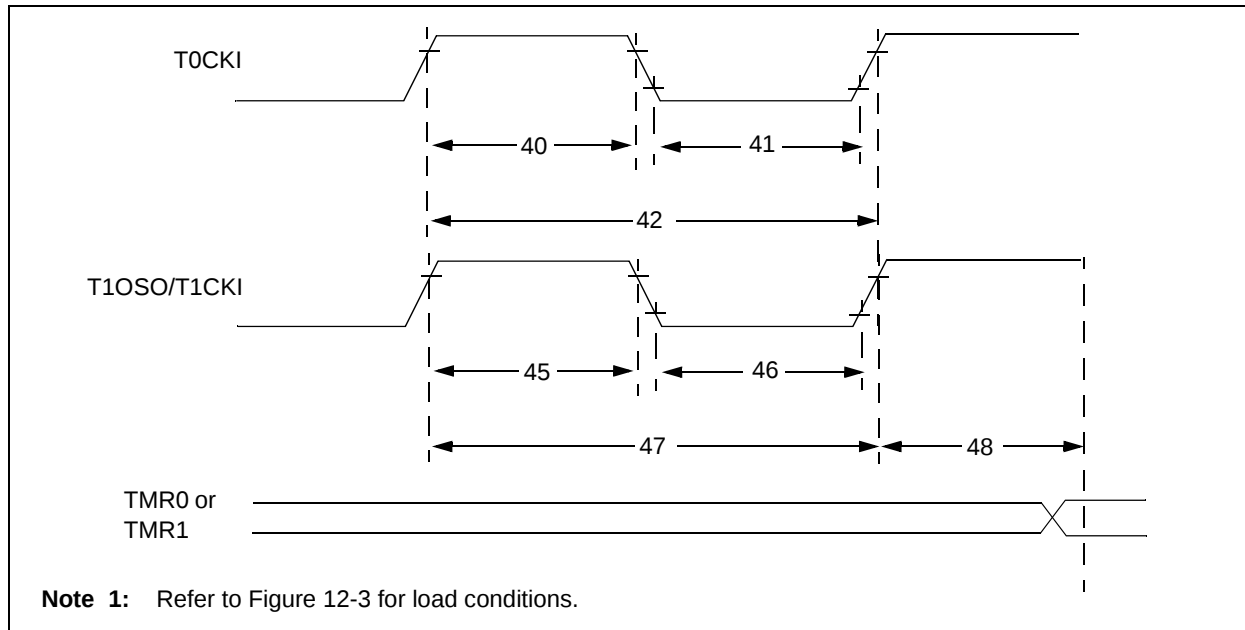
| Param No. | Sym   | Characteristic                                | Min | Typ†      | Max | Units | Conditions                |
|-----------|-------|-----------------------------------------------|-----|-----------|-----|-------|---------------------------|
| 30        | TMCLR | MCLR Pulse Width (low)                        | 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 +85°C  |
|           |       |                                               | TBD | TBD       | TBD | ms    | VDD = 5V, +85°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 +85°C  |
|           |       |                                               | TBD | TBD       | TBD | ms    | VDD = 5V, +85°C to +125°C |
| 34        | TIOZ  | I/O high-impedance from 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.

# PIC16F716

**FIGURE 12-8: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS<sup>(1)</sup>**



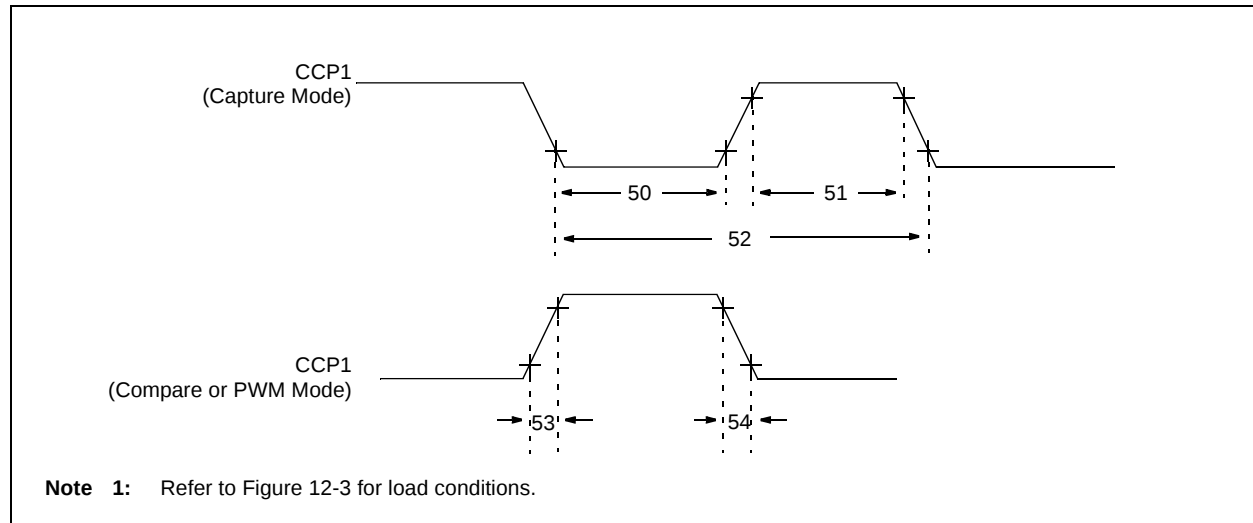
**TABLE 12-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS**

| Param No. | Sym       | Characteristic                                                                      |                                | Min                                            | Typ† | Max        | Units | Conditions                          |
|-----------|-----------|-------------------------------------------------------------------------------------|--------------------------------|------------------------------------------------|------|------------|-------|-------------------------------------|
| 40*       | Tt0H      | T0CKI High Pulse Width                                                              | No Prescaler                   | $0.5T_{CY} + 20$                               | —    | —          | ns    | Must also meet parameter 42         |
|           |           |                                                                                     | With Prescaler                 | 10                                             | —    | —          | ns    |                                     |
| 41*       | Tt0L      | T0CKI Low Pulse Width                                                               | No Prescaler                   | $0.5T_{CY} + 20$                               | —    | —          | ns    | Must also meet parameter 42         |
|           |           |                                                                                     | With Prescaler                 | 10                                             | —    | —          | ns    |                                     |
| 42*       | Tt0P      | T0CKI Period                                                                        | No Prescaler                   | $T_{CY} + 40$                                  | —    | —          | ns    | N = prescale value (2, 4, ..., 256) |
|           |           |                                                                                     | With Prescaler                 | Greater of:<br>$20$ or $\frac{T_{CY} + 40}{N}$ | —    | —          | ns    |                                     |
| 45*       | Tt1H      | T1CKI High Time                                                                     | Synchronous, Prescaler = 1     | $0.5T_{CY} + 20$                               | —    | —          | ns    | Must also meet parameter 47         |
|           |           |                                                                                     | Synchronous, Prescaler = 2,4,8 | 15                                             | —    | —          | ns    |                                     |
|           |           |                                                                                     | Asynchronous                   | 30                                             | —    | —          | ns    |                                     |
| 46*       | Tt1L      | T1CKI Low Time                                                                      | Synchronous, Prescaler = 1     | $0.5T_{CY} + 20$                               | —    | —          | ns    | Must also meet parameter 47         |
|           |           |                                                                                     | Synchronous, Prescaler = 2,4,8 | 15                                             | —    | —          | ns    |                                     |
|           |           |                                                                                     | Asynchronous                   | 30                                             | —    | —          | ns    |                                     |
| 47*       | Tt1P      | T1CKI input period                                                                  | Synchronous                    | Greater of:<br>$30$ OR $\frac{T_{CY} + 40}{N}$ | —    | —          | ns    | N = prescale value (1, 2, 4, 8)     |
|           |           |                                                                                     | Asynchronous                   | 60                                             | —    | —          | ns    |                                     |
|           | Ft1       | Timer1 oscillator input frequency range (oscillator enabled by setting bit T1OSCEN) |                                | 32.768                                         | —    | 32.768     | kHz   |                                     |
| 48*       | TCKEZtmr1 | Delay from external clock edge to timer increment                                   |                                | $2T_{osc}$                                     | —    | $7T_{osc}$ | —     |                                     |

\* 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 12-9: CAPTURE/COMPARE/PWM TIMINGS<sup>(1)</sup>**



**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    |                                  |
| 51*       | TccH | CCP1 input high time  | No Prescaler              | $0.5T_{CY} + 20$         | —     | —   | ns    |                                  |
|           |      |                       | With Prescaler   Standard | 10                       | —     | —   | ns    |                                  |
| 52*       | TccP | CCP1 input period     |                           | $\frac{3T_{CY} + 40}{N}$ | —     | —   | ns    | N = prescale value (1, 4, or 16) |
| 53*       | TccR | CCP1 output rise time | Standard                  | —                        | 10    | 40  | ns    |                                  |
| 53A*      |      |                       | Extended                  | —                        | —     | 80  | ns    |                                  |
| 54*       | TccF | CCP1 output fall time | Standard                  | —                        | 10    | 40  | ns    |                                  |
| 54A*      |      |                       | Extended                  | —                        | —     | 80  | 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.

# PIC16F716

**TABLE 12-7: A/D CONVERTER CHARACTERISTICS: PIC16F716 (INDUSTRIAL, EXTENDED)**

| Para<br>m<br>No. | Sym  | Characteristic                                 |          | Min       | Typ†                      | Max        | Units | Conditions                                                                                                                                                                                    |
|------------------|------|------------------------------------------------|----------|-----------|---------------------------|------------|-------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| A00              | VDD  | VDD Operation                                  |          | 2.5       | —                         | 5.5        | V     |                                                                                                                                                                                               |
| A01              | NR   | Resolution                                     |          | —         | —                         | 8-bits     | bit   | VREF = VDD = 5.12V,<br>VSS ≤ VAIN ≤ VREF                                                                                                                                                      |
| A02              | EABS | Total Absolute error                           |          | —         | —                         | < ± 1      | LSb   | VREF = VDD = 5.12V,<br>VSS ≤ VAIN ≤ VREF                                                                                                                                                      |
| A03              | EIL  | Integral linearity error                       |          | —         | —                         | < ± 1      | LSb   | VREF = VDD = 5.12V,<br>VSS ≤ VAIN ≤ VREF                                                                                                                                                      |
| A04              | EDL  | Differential linearity error                   |          | —         | —                         | < ± 1      | LSb   | VREF = VDD = 5.12V,<br>VSS ≤ VAIN ≤ VREF                                                                                                                                                      |
| A05              | EFS  | Full scale error                               |          | —         | —                         | < ± 1      | LSb   | VREF = VDD = 5.12V,<br>VSS ≤ VAIN ≤ VREF                                                                                                                                                      |
| A06              | EOFF | Offset error                                   |          | —         | —                         | < ± 1      | LSb   | VREF = VDD = 5.12V,<br>VSS ≤ VAIN ≤ VREF                                                                                                                                                      |
| A10              | —    | Monotonicity                                   |          | —         | guaranteed <sup>(3)</sup> | —          | —     | VSS ≤ VAIN ≤ VREF                                                                                                                                                                             |
| A20              | VREF | Reference voltage                              |          | 2.5V      | —                         | VDD + 0.3  | V     |                                                                                                                                                                                               |
| A25              | VAIN | Analog input voltage                           |          | VSS - 0.3 | —                         | VREF + 0.3 | V     |                                                                                                                                                                                               |
| A30              | ZAIN | Recommended impedance of analog voltage source |          | —         | —                         | 10.0       | kΩ    |                                                                                                                                                                                               |
| A40              | IAD  | A/D conversion current (VDD)                   | Standard | —         | 180                       | —          | μA    | Average current consumption when A/D is on. <sup>(1)</sup>                                                                                                                                    |
| A50              | IREF | VREF input current <sup>(2)</sup>              |          | 10        | —                         | 1000       | μA    | During VAIN acquisition. Based on differential of VHOLD to VAIN to charge CHOLD, see <b>Section 12.1 “DC Characteristics: PIC16F716 (Industrial, Extended)”</b> . During A/D Conversion cycle |
|                  |      |                                                |          | —         | —                         | 10         | μA    |                                                                                                                                                                                               |

\* 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.

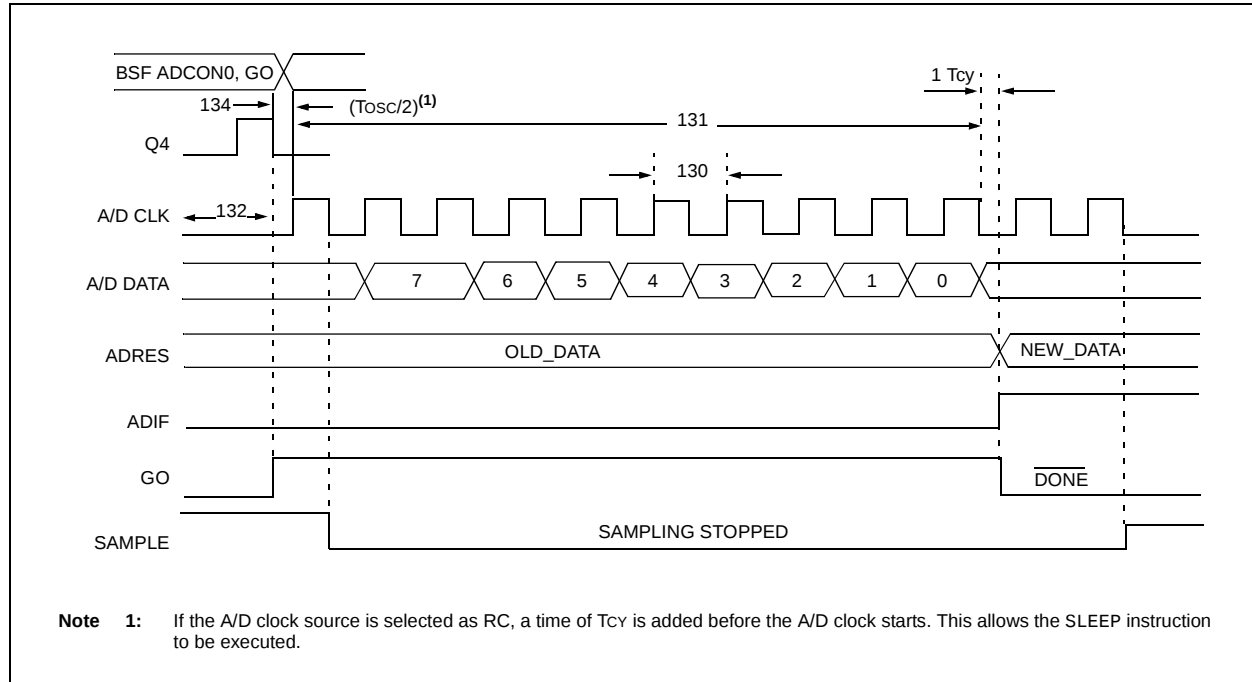
**Note 1:** When A/D is off, it will not consume any current other than minor leakage current.

The power-down current spec includes any such leakage from the A/D module.

**2:** VREF current is from RA3 pin or VDD 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 12-10: A/D CONVERSION TIMING**



**TABLE 12-8: A/D CONVERSION REQUIREMENTS**

| Param No. | Sym  | Characteristic                                          |            | Min      | Typ†      | Max | Units | Conditions                                                                                                                                                                                                  |
|-----------|------|---------------------------------------------------------|------------|----------|-----------|-----|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 130       | TAD  | A/D clock period                                        | Industrial | 1.6      | —         | —   | μs    | TOSC based, VREF ≥ 3.0V                                                                                                                                                                                     |
|           |      |                                                         | Industrial | 1.6      | 4.0       | 6.0 | μs    | A/D RC mode                                                                                                                                                                                                 |
|           |      |                                                         | Extended   | 1.6      | —         | —   | μs    | TOSC based, VREF ≥ 3.0V                                                                                                                                                                                     |
|           |      |                                                         | Extended   | 1.6      | 6.0       | 9.0 | μs    | A/D RC mode                                                                                                                                                                                                 |
| 131       | Tcnv | Conversion time (not including S/H time) <sup>(1)</sup> |            | 9.5      | —         | 9.5 | TAD   |                                                                                                                                                                                                             |
| 132       | TACQ | Acquisition time                                        |            | (Note 2) | 20        | —   | 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). |
|           |      |                                                         |            | 5*       | —         | —   | μs    |                                                                                                                                                                                                             |
| 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.

\*\* This specification ensured by design.

† 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:** ADRES register may be read on the following Tcy cycle.

**2:** See Section 12.1 "DC Characteristics: PIC16F716 (Industrial, Extended)" for min. conditions.

# PIC16F716

---

NOTES:

## 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 will 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  $(\text{mean} + 3\sigma)$  or  $(\text{mean} - 3\sigma)$  respectively, where  $\sigma$  is standard deviation, over the whole temperature range.

**Graphs and Tables not available at this time.**

# PIC16F716

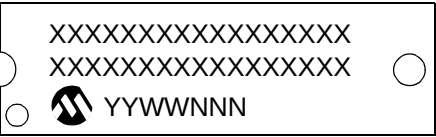
---

NOTES:

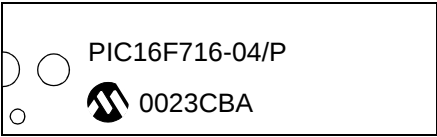
14.0 PACKAGING INFORMATION

14.1 Package Marking Information

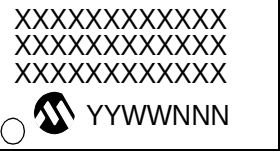
18-Lead PDIP



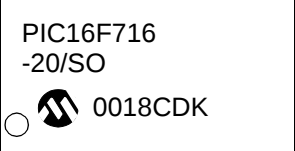
Example



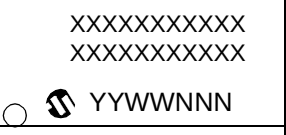
18-Lead SOIC



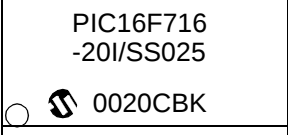
Example



20-Lead SSOP



Example



**Legend:**

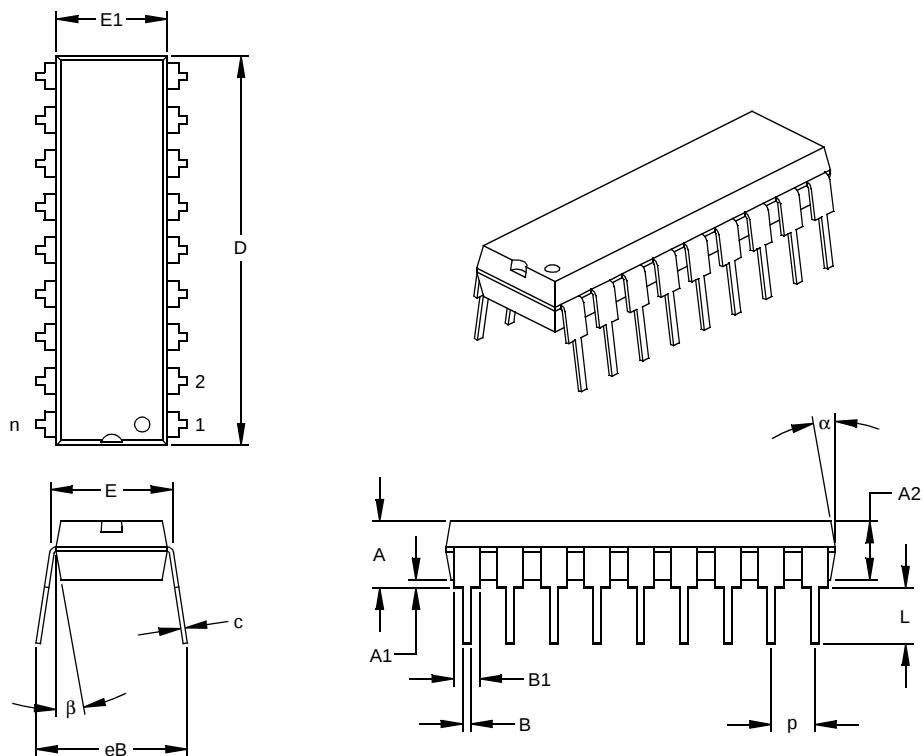
|        |                                            |
|--------|--------------------------------------------|
| XX...X | Customer specific information*             |
| Y      | Year code (last digit of calendar year)    |
| YY     | Year code (last 2 digits of calendar year) |
| WW     | Week code (week of January 1 is week '01') |
| NNN    | Alphanumeric traceability code             |

**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 PICmicro device marking consists of Microchip part number, year code, week code, and traceability code. For PICmicro device marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

# PIC16F716

## 18-Lead Plastic Dual In-line (P) – 300 mil (PDIP)



| Units                      |      | INCHES* |      |      | MILLIMETERS |       |       |
|----------------------------|------|---------|------|------|-------------|-------|-------|
| Dimension Limits           |      | MIN     | NOM  | MAX  | MIN         | NOM   | MAX   |
| Number of Pins             | n    |         | 18   |      |             | 18    |       |
| Pitch                      | p    |         | .100 |      |             | 2.54  |       |
| Top to Seating Plane       | A    | .140    | .155 | .170 | 3.56        | 3.94  | 4.32  |
| Molded Package Thickness   | A2   | .115    | .130 | .145 | 2.92        | 3.30  | 3.68  |
| Base to Seating Plane      | A1   | .015    |      |      | 0.38        |       |       |
| Shoulder to Shoulder Width | E    | .300    | .313 | .325 | 7.62        | 7.94  | 8.26  |
| Molded Package Width       | E1   | .240    | .250 | .260 | 6.10        | 6.35  | 6.60  |
| Overall Length             | D    | .890    | .898 | .905 | 22.61       | 22.80 | 22.99 |
| Tip to Seating Plane       | L    | .125    | .130 | .135 | 3.18        | 3.30  | 3.43  |
| Lead Thickness             | c    | .008    | .012 | .015 | 0.20        | 0.29  | 0.38  |
| Upper Lead Width           | B1   | .045    | .058 | .070 | 1.14        | 1.46  | 1.78  |
| Lower Lead Width           | B    | .014    | .018 | .022 | 0.36        | 0.46  | 0.56  |
| Overall Row Spacing        | § eB | .310    | .370 | .430 | 7.87        | 9.40  | 10.92 |
| Mold Draft Angle Top       | α    | 5       | 10   | 15   | 5           | 10    | 15    |
| Mold Draft Angle Bottom    | β    | 5       | 10   | 15   | 5           | 10    | 15    |

\* Controlling Parameter

§ Significant Characteristic

Notes:

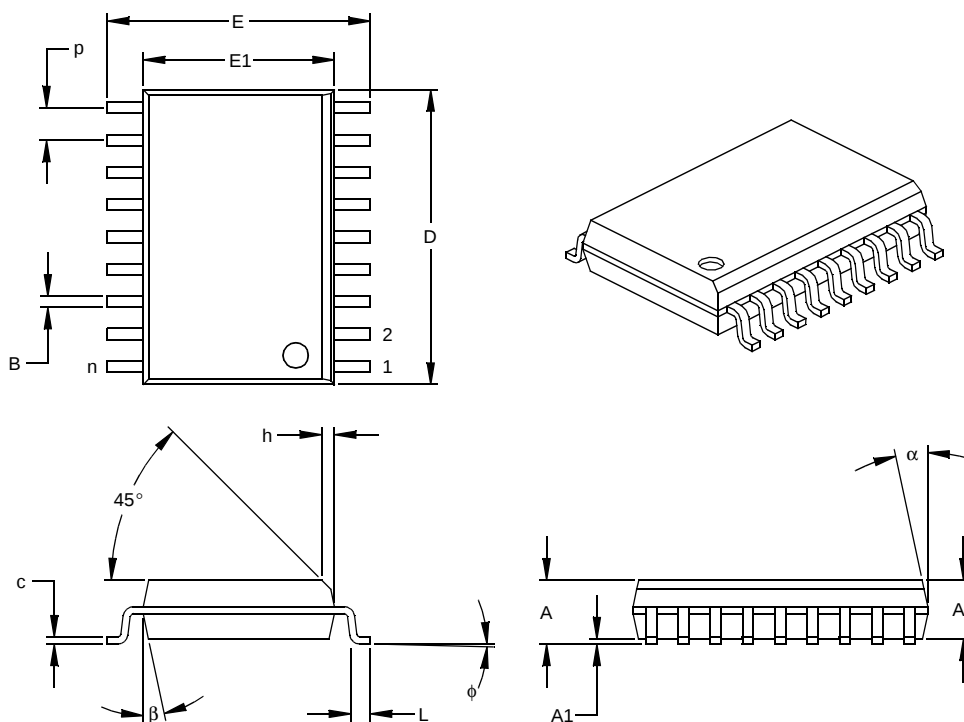
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed

.010" (0.254mm) per side.

JEDEC Equivalent: MS-001

Drawing No. C04-007

## 18-Lead Plastic Small Outline (SO) – Wide, 300 mil (SOIC)



| Units                    |    | INCHES* |      |      | MILLIMETERS |       |       |
|--------------------------|----|---------|------|------|-------------|-------|-------|
| Dimension Limits         |    | MIN     | NOM  | MAX  | MIN         | NOM   | MAX   |
| Number of Pins           | n  |         | 18   |      |             | 18    |       |
| Pitch                    | p  |         | .050 |      |             | 1.27  |       |
| Overall Height           | A  | .093    | .099 | .104 | 2.36        | 2.50  | 2.64  |
| Molded Package Thickness | A2 | .088    | .091 | .094 | 2.24        | 2.31  | 2.39  |
| Standoff §               | A1 | .004    | .008 | .012 | 0.10        | 0.20  | 0.30  |
| Overall Width            | E  | .394    | .407 | .420 | 10.01       | 10.34 | 10.67 |
| Molded Package Width     | E1 | .291    | .295 | .299 | 7.39        | 7.49  | 7.59  |
| Overall Length           | D  | .446    | .454 | .462 | 11.33       | 11.53 | 11.73 |
| Chamfer Distance         | h  | .010    | .020 | .029 | 0.25        | 0.50  | 0.74  |
| Foot Length              | L  | .016    | .033 | .050 | 0.41        | 0.84  | 1.27  |
| Foot Angle               | φ  | 0       | 4    | 8    | 0           | 4     | 8     |
| Lead Thickness           | c  | .009    | .011 | .012 | 0.23        | 0.27  | 0.30  |
| Lead Width               | B  | .014    | .017 | .020 | 0.36        | 0.42  | 0.51  |
| Mold Draft Angle Top     | α  | 0       | 12   | 15   | 0           | 12    | 15    |
| Mold Draft Angle Bottom  | β  | 0       | 12   | 15   | 0           | 12    | 15    |

\* Controlling Parameter

§ Significant Characteristic

### Notes:

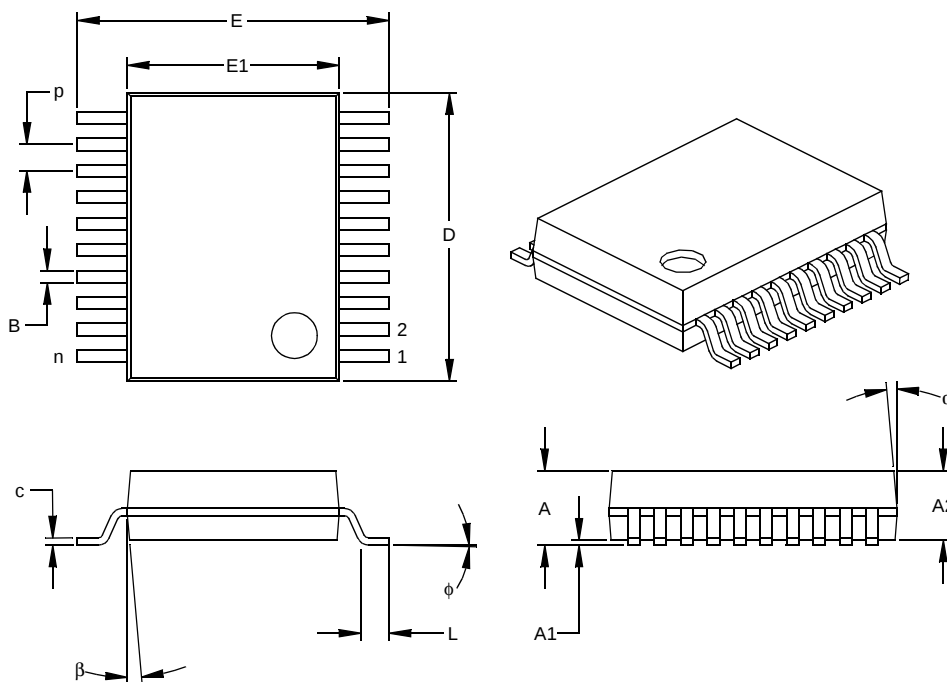
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.

JEDEC Equivalent: MS-013

Drawing No. C04-051

# PIC16F716

## 20-Lead Plastic Shrink Small Outline (SS) – 209 mil, 5.30 mm (SSOP)



| Units                    |    | INCHES* |      |      | MILLIMETERS |        |        |
|--------------------------|----|---------|------|------|-------------|--------|--------|
| Dimension Limits         |    | MIN     | NOM  | MAX  | MIN         | NOM    | MAX    |
| Number of Pins           | n  |         | 20   |      |             | 20     |        |
| Pitch                    | p  |         | .026 |      |             | 0.65   |        |
| Overall Height           | A  | .068    | .073 | .078 | 1.73        | 1.85   | 1.98   |
| Molded Package Thickness | A2 | .064    | .068 | .072 | 1.63        | 1.73   | 1.83   |
| Standoff §               | A1 | .002    | .006 | .010 | 0.05        | 0.15   | 0.25   |
| Overall Width            | E  | .299    | .309 | .322 | 7.59        | 7.85   | 8.18   |
| Molded Package Width     | E1 | .201    | .207 | .212 | 5.11        | 5.25   | 5.38   |
| Overall Length           | D  | .278    | .284 | .289 | 7.06        | 7.20   | 7.34   |
| Foot Length              | L  | .022    | .030 | .037 | 0.56        | 0.75   | 0.94   |
| Lead Thickness           | c  | .004    | .007 | .010 | 0.10        | 0.18   | 0.25   |
| Foot Angle               | φ  | 0       | 4    | 8    | 0.00        | 101.60 | 203.20 |
| Lead Width               | B  | .010    | .013 | .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

§ Significant Characteristic

Notes:

Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.

JEDEC Equivalent: MO-150

Drawing No. C04-072

## APPENDIX A: REVISION HISTORY

### Revision A (June 2003)

Original data sheet. However, the device described in this data sheet are upgrades to PIC16C716.

## APPENDIX B: CONVERSION CONSIDERATIONS

This is a Flash program memory version of the PIC16C716 device. Refer to the migration document, DS40059, for more information about differences between the PIC16F716 and PIC16C716.

## 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., PIC16F716).

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 PIC16F716 devices using only five pins: VDD, VSS, MCLR/VPP, RB6 (clock) and RB7 (data in/out).
16. PCON status register is added with a Power-on Reset Status bit (POR).
17. Brown-out protection circuitry has been added. Controlled by configuration word bits BOREN and BORV. Brown-out Reset ensures the device is placed in a Reset condition if VDD dips below a fixed setpoint.

To convert code written for PIC16C5X to PIC16F716, 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

**Note 1:** This device has been designed to perform to the parameters of its data sheet. It has been tested to an electrical specification designed to determine its conformance with these parameters. Due to process differences in the manufacture of this device, this device may have different performance characteristics than its earlier version. These differences may cause this device to perform differently in your application than the earlier version of this device.

- 2: The user should verify that the device oscillator starts and performs as expected. Adjusting the loading capacitor values and/or the Oscillator mode may be required.

## ON-LINE SUPPORT

Microchip provides on-line support on the Microchip World Wide Web 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® Internet 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 at the following URL:

**[www.microchip.com](http://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 the most current upgrade kits. The Hot Line Numbers are:

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

1-480-792-7302 for the rest of the world.

042003

# PIC16F716

---

## READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

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

To: Technical Publications Manager  
RE: Reader Response  
From: Name \_\_\_\_\_  
Company \_\_\_\_\_  
Address \_\_\_\_\_  
City / State / ZIP / Country \_\_\_\_\_  
Telephone: (\_\_\_\_) \_\_\_\_\_ - \_\_\_\_\_ FAX: (\_\_\_\_) \_\_\_\_\_ - \_\_\_\_\_

Application (optional):

Would you like a reply? \_\_\_Y \_\_\_N

Device: PIC16F716

Literature Number: DS41206A

Questions:

1. What are the best features of this document?

---

---

2. How does this document meet your hardware and software development needs?

---

---

3. Do you find the organization of this document easy to follow? If not, why?

---

---

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

---

---

5. What deletions from the document could be made without affecting the overall usefulness?

---

---

6. Is there any incorrect or misleading information (what and where)?

---

---

7. How would you improve this document?

---

---

## INDEX

### A

|                                                     |        |
|-----------------------------------------------------|--------|
| A/D .....                                           | 49     |
| A/D Converter Enable (ADIE Bit) .....               | 14     |
| A/D Converter Flag (ADIF Bit) .....                 | 15, 50 |
| A/D Converter Interrupt, Configuring .....          | 50     |
| ADCON0 Register .....                               | 9, 49  |
| ADCON1 Register .....                               | 10, 50 |
| ADRES Register .....                                | 9      |
| Analog Port Pins, Configuring .....                 | 52     |
| Channel Select (CHS2:CHS0 Bits) .....               | 49     |
| Clock Select (ADCS1:ADCS0 Bits) .....               | 49     |
| Configuring the Module .....                        | 50     |
| Conversion Clock (Tad) .....                        | 52     |
| Conversion Status (GO/DONE Bit) .....               | 49, 50 |
| Conversions .....                                   | 53     |
| Converter Characteristics .....                     | 104    |
| Module On/Off (ADON Bit) .....                      | 49     |
| Port Configuration Control (PCFG2:PCFG0 Bits) ..... | 50     |
| Sampling Requirements .....                         | 51     |
| Special Event Trigger (CCP) .....                   | 35, 53 |
| Timing Diagram .....                                | 105    |
| Absolute Maximum Ratings .....                      | 91     |
| ADCON0 Register .....                               | 9      |
| ADCON1 Register .....                               | 10     |
| ADDLW Instruction .....                             | 73     |
| ADDWF Instruction .....                             | 73     |
| ADRES Register .....                                | 9      |
| Analog Input Model .....                            | 51     |
| Analog-to-Digital Converter. See A/D .....          |        |
| ANDLW Instruction .....                             | 73     |
| ANDWF Instruction .....                             | 73     |
| Assembler .....                                     |        |
| MPASM Assembler .....                               | 85     |

### B

|                                |                |
|--------------------------------|----------------|
| Banking, Data Memory .....     | 7, 11          |
| BCF Instruction .....          | 74             |
| Block Diagrams .....           |                |
| A/D .....                      | 51             |
| Capture (CCP Module) .....     | 34             |
| Compare (CCP Module) .....     | 35             |
| Interrupt Sources .....        | 65             |
| On-Chip Reset Circuit .....    | 60             |
| PIC16F716 .....                | 5              |
| PORTA .....                    | 19, 20         |
| PORTB .....                    | 21             |
| PWM (CCP Module) .....         | 36             |
| PWM (Enhanced) .....           | 38             |
| RB1/T1OSO/T1CKI .....          | 22             |
| RB2/T1OSI .....                | 22             |
| RB3/CCP1/P1A .....             | 23             |
| RB4 .....                      | 23             |
| RB5 .....                      | 24             |
| RB6/P1C .....                  | 24             |
| RB7/P1D .....                  | 25             |
| Timer0 .....                   | 27             |
| Timer0/WDT Prescaler .....     | 28             |
| Timer1 .....                   | 30             |
| Timer2 .....                   | 32             |
| Watchdog Timer (WDT) .....     | 67             |
| BOR. See Brown-out Reset ..... |                |
| Brown-out Detect (BOD) .....   | 59             |
| Brown-out Reset (BOR) .....    | 55, 58, 62, 63 |

|                              |     |
|------------------------------|-----|
| BOR Enable (BODEN Bit) ..... | 56  |
| BOR Status (BOR Bit) .....   | 16  |
| Timing Diagram .....         | 101 |
| BSF Instruction .....        | 74  |
| BTFSC Instruction .....      | 74  |
| BTFSS Instruction .....      | 75  |

### C

|                                                  |            |
|--------------------------------------------------|------------|
| C Compilers .....                                |            |
| MPLAB C17 .....                                  | 86         |
| MPLAB C18 .....                                  | 86         |
| MPLAB C30 .....                                  | 86         |
| CALL Instruction .....                           | 75         |
| Capture (CCP Module) .....                       | 34         |
| CCP Pin Configuration .....                      | 34         |
| CCPR1H:CCPR1L Registers .....                    | 34         |
| Software Interrupt .....                         | 34         |
| Timer1 Mode Selection .....                      | 34         |
| Capture/Compare/PWM (CCP) .....                  | 33         |
| Capture Mode. See Capture .....                  |            |
| CCP1CON Register .....                           | 9          |
| CCPR1H Register .....                            | 9, 33      |
| CCPR1L Register .....                            | 9, 33      |
| Compare Mode. See Compare .....                  |            |
| Enable (CCP1IE Bit) .....                        | 14         |
| Flag (CCP1IF Bit) .....                          | 15         |
| PWM Mode. See PWM .....                          |            |
| Timer Resources .....                            | 33         |
| Timing Diagram .....                             | 103        |
| CLRF Instruction .....                           | 75         |
| CLRWF Instruction .....                          | 76         |
| CLRWDI Instruction .....                         | 76         |
| Code Examples .....                              |            |
| Capture (CCP Module) .....                       |            |
| Changing Between Capture Prescalers .....        | 34         |
| How to Clear RAM Using Indirect Addressing ..... | 18         |
| Initializing PORTA .....                         | 19         |
| Initializing PORTB .....                         | 21         |
| Code Protection .....                            | 55, 69     |
| CP1:CP0 Bits .....                               | 56         |
| COMF Instruction .....                           | 76         |
| Compare (CCP Module) .....                       | 34         |
| CCP Pin Configuration .....                      | 35         |
| Software Interrupt .....                         | 35         |
| Special Event Trigger .....                      | 30, 35, 53 |
| Timer1 Mode Selection .....                      | 35         |
| Configuration Bits .....                         | 55         |
| Conversion Considerations .....                  | 113        |

### D

|                                  |                |
|----------------------------------|----------------|
| Data Memory .....                | 7              |
| Bank Select (RP1:RP0 Bits) ..... | 7, 11          |
| General Purpose Registers .....  | 8              |
| Register File Map .....          | 8              |
| Special Function Registers ..... | 9              |
| DC Characteristics .....         | 93, 94, 95, 96 |
| DECF Instruction .....           | 76             |
| DECFSZ Instruction .....         | 77             |
| Demonstration Boards .....       |                |
| PICDEM 1 .....                   | 88             |
| PICDEM 17 .....                  | 88             |
| PICDEM 18R PIC18C601/801 .....   | 89             |
| PICDEM 2 Plus .....              | 88             |
| PICDEM 3 PIC16C92X .....         | 88             |

# PIC16F716

|                                    |    |
|------------------------------------|----|
| PICDEM 4 .....                     | 88 |
| PICDEM LIN PIC16C43X .....         | 89 |
| PICDEM USB PIC16C7X5 .....         | 89 |
| PICDEM.net Internet/Ethernet ..... | 88 |
| Development Support .....          | 85 |
| Direct Addressing .....            | 18 |

## E

|                                                |    |
|------------------------------------------------|----|
| ECCP                                           |    |
| Auto-Shutdown .....                            | 45 |
| and Automatic Restart .....                    | 47 |
| Start-up Considerations .....                  | 47 |
| Electrical Characteristics .....               | 91 |
| Enhanced Capture/Compare/PWM (ECCP)            |    |
| PWM Mode. See PWM (ECCP Module)                |    |
| Enhanced CCP Auto-Shutdown .....               | 45 |
| Enhanced PWM Mode. See PWM (ECCP Module) ..... | 38 |
| Errata .....                                   | 3  |
| Evaluation and Programming Tools .....         | 89 |
| External Power-on Reset Circuit .....          | 58 |

## G

|                        |    |
|------------------------|----|
| GOTO Instruction ..... | 77 |
|------------------------|----|

## I

|                                            |          |
|--------------------------------------------|----------|
| I/O Ports .....                            | 19       |
| ID Locations .....                         | 55, 69   |
| INCF Instruction .....                     | 78       |
| INCFSZ Instruction .....                   | 78       |
| In-Circuit Serial Programming (ICSP) ..... | 55, 69   |
| Indirect Addressing .....                  | 18       |
| FSR Register .....                         | 8, 9, 18 |
| INDF Register .....                        | 9        |
| Instruction Set                            |          |
| ADDLW .....                                | 73       |
| ADDWF .....                                | 73       |
| ANDLW .....                                | 73       |
| ANDWF .....                                | 73       |
| BCF .....                                  | 74       |
| BSF .....                                  | 74       |
| BTFSC .....                                | 74       |
| BTFSS .....                                | 75       |
| CALL .....                                 | 75       |
| CLRF .....                                 | 75       |
| CLRWF .....                                | 76       |
| CLRWDW .....                               | 76       |
| COMF .....                                 | 76       |
| DECF .....                                 | 76       |
| DECFSZ .....                               | 77       |
| GOTO .....                                 | 77       |
| INCF .....                                 | 78       |
| INCFSZ .....                               | 78       |
| IORLW .....                                | 79       |
| IORWF .....                                | 79       |
| MOVF .....                                 | 79       |
| MOVLW .....                                | 79       |
| MOVWF .....                                | 80       |
| NOP .....                                  | 80       |
| OPTION .....                               | 80       |
| RETFIE .....                               | 80       |
| RETLW .....                                | 81       |
| RETURN .....                               | 81       |
| RLF .....                                  | 81       |
| RRF .....                                  | 82       |
| SLEEP .....                                | 82       |
| SUBLW .....                                | 82       |

|                                                     |        |
|-----------------------------------------------------|--------|
| SUBWF .....                                         | 83     |
| SWAPF .....                                         | 83     |
| TRIS .....                                          | 83     |
| XORLW .....                                         | 84     |
| XORWF .....                                         | 84     |
| Instruction Set Summary .....                       | 71     |
| INT Interrupt (RB0/INT). See Interrupt Sources      |        |
| INTCON Register .....                               | 9, 13  |
| GIE Bit .....                                       | 13     |
| INTE Bit .....                                      | 13     |
| INTF Bit .....                                      | 13     |
| PEIE Bit .....                                      | 13     |
| RBIE Bit .....                                      | 13     |
| RBIF Bit .....                                      | 13     |
| TOIE Bit .....                                      | 13     |
| T0IF Bit .....                                      | 13     |
| Interrupt Sources .....                             | 55, 65 |
| A/D Conversion Complete .....                       | 50     |
| Capture Complete (CCP) .....                        | 34     |
| Compare Complete (CCP) .....                        | 35     |
| Interrupt-on-Change (RB7:RB4 ) .....                | 21     |
| RB0/INT Pin, External .....                         | 66     |
| TMR0 Overflow .....                                 | 28, 66 |
| TMR1 Overflow .....                                 | 29, 30 |
| TMR2 to PR2 Match .....                             | 32     |
| TMR2 to PR2 Match (PWM) .....                       | 31, 36 |
| Interrupts, Context Saving During .....             | 66     |
| Interrupts, Enable Bits                             |        |
| A/D Converter Enable (ADIE Bit) .....               | 14     |
| CCP1 Enable (CCP1IE Bit) .....                      | 14, 34 |
| Global Interrupt Enable (GIE Bit) .....             | 13, 65 |
| Interrupt-on-Change (RB7:RB4) Enable (RBIE Bit) ... | 13, 66 |
| Peripheral Interrupt Enable (PEIE Bit) .....        | 13     |
| RB0/INT Enable (INTE Bit) .....                     | 13     |
| TMR0 Overflow Enable (TOIE Bit) .....               | 13     |
| TMR1 Overflow Enable (TMR1IE Bit) .....             | 14     |
| TMR2 to PR2 Match Enable (TMR2IE Bit) .....         | 14     |
| Interrupts, Flag Bits                               |        |
| A/D Converter Flag (ADIF Bit) .....                 | 15, 50 |
| CCP1 Flag (CCP1IF Bit) .....                        | 15, 34 |
| Interrupt-on-Change (RB7:RB4) Flag (RBIF Bit) ..    | 13, 66 |
| RB0/INT Flag (INTF Bit) .....                       | 13     |
| TMR0 Overflow Flag (T0IF Bit) .....                 | 13, 66 |
| TMR1 Overflow Flag (TMR1IF Bit) .....               | 15     |
| TMR2 to PR2 Match Flag (TMR2IF Bit) .....           | 15     |
| IORLW Instruction .....                             | 79     |
| IORWF Instruction .....                             | 79     |

## M

|                                                        |            |
|--------------------------------------------------------|------------|
| Master Clear ( $\overline{\text{MCLR}}$ )              |            |
| $\overline{\text{MCLR}}$ Reset, Normal Operation ..... | 58, 62, 63 |
| $\overline{\text{MCLR}}$ Reset, Sleep .....            | 58, 62, 63 |
| Memory Organization                                    |            |
| Data Memory .....                                      | 7          |
| Program Memory .....                                   | 7          |
| Migration from Base-Line to Mid-Range Devices .....    | 114        |
| MOVF Instruction .....                                 | 79         |
| MOVLW Instruction .....                                | 79         |
| MOVWF Instruction .....                                | 80         |
| MPLAB ASM30 Assembler, Linker, Librarian .....         | 86         |
| MPLAB ICD 2 In-Circuit Debugger .....                  | 87         |
| MPLAB ICE 2000 High Performance Universal              |            |
| In-Circuit Emulator .....                              | 87         |
| MPLAB ICE 4000 High Performance Universal              |            |
| In-Circuit Emulator .....                              | 87         |

|                                                         |    |
|---------------------------------------------------------|----|
| MPLAB Integrated Development Environment Software ..... | 85 |
| MPLINK Object Linker/MPLIB Object Librarian .....       | 86 |

## N

|                      |    |
|----------------------|----|
| NOP Instruction..... | 80 |
|----------------------|----|

## O

|                                    |            |
|------------------------------------|------------|
| OPTION Instruction.....            | 80         |
| OPTION_REG Register .....          | 10, 12     |
| INTEDG Bit .....                   | 12         |
| PS2:PS0 Bits .....                 | 12, 27     |
| PSA Bit.....                       | 12, 27     |
| RBPB Bit.....                      | 12         |
| T0CS Bit.....                      | 12, 27     |
| T0SE Bit.....                      | 12, 27     |
| Oscillator Configuration.....      | 55, 57     |
| HS.....                            | 57, 62     |
| LP.....                            | 57, 62     |
| RC.....                            | 57, 58, 62 |
| Selection (FOSC1:FOSC0 Bits) ..... | 56         |
| XT .....                           | 57, 62     |
| Oscillator, Timer1.....            | 29, 30     |
| Oscillator, WDT .....              | 67         |

## P

|                                                  |                |
|--------------------------------------------------|----------------|
| Packaging .....                                  | 109            |
| Paging, Program Memory .....                     | 7, 17          |
| PCON Register .....                              | 16, 62         |
| BOR Bit.....                                     | 16             |
| POR Bit.....                                     | 16             |
| PICKIT 1 FLASH Starter Kit.....                  | 89             |
| PICSTART Plus Development Programmer .....       | 87             |
| PIE1 Register .....                              | 10, 14         |
| ADIE Bit .....                                   | 14             |
| CCP1IE Bit.....                                  | 14             |
| TMR1IE Bit.....                                  | 14             |
| TMR2IE Bit.....                                  | 14             |
| PIR1 Register .....                              | 9, 15          |
| ADIF Bit.....                                    | 15             |
| CCP1IF Bit.....                                  | 15             |
| TMR1IF Bit.....                                  | 15             |
| TMR2IF Bit.....                                  | 15             |
| Pointer, FSR .....                               | 18             |
| POR. See Power-on Reset                          |                |
| PORTA                                            |                |
| PORTA Register .....                             | 9, 19          |
| TRISA Register .....                             | 10, 19         |
| PORTB                                            |                |
| PORTB Register .....                             | 9, 21          |
| Pull-up Enable (RBPB Bit) .....                  | 12             |
| RB0/INT Edge Select (INTEDG Bit).....            | 12             |
| RB0/INT Pin, External.....                       | 66             |
| RB7:RB4 Interrupt-on-Change.....                 | 66             |
| RB7:RB4 Interrupt-on-Change Enable               |                |
| (RBIE Bit).....                                  | 13, 66         |
| RB7:RB4 Interrupt-on-Change Flag (RBIF Bit)..... | 66             |
| TRISB Register .....                             | 10, 21         |
| Postscaler, Timer2                               |                |
| Select (TOUTPS3:TOUTPS0 Bits) .....              | 31             |
| Postscaler, WDT .....                            | 27             |
| Assignment (PSA Bit) .....                       | 12, 27         |
| Rate Select (PS2:PS0 Bits) .....                 | 12, 27         |
| Switching Between Timer0 and WDT .....           | 28             |
| Power-down Mode. See Sleep                       |                |
| Power-on Reset (POR) .....                       | 55, 58, 62, 63 |
| Oscillator Start-up Timer (OST) .....            | 55, 59         |

|                                                    |           |
|----------------------------------------------------|-----------|
| POR Status ( $\overline{\text{POR}}$ Bit) .....    | 16        |
| Power Control (PCON) Register.....                 | 62        |
| Power-down (PD Bit) .....                          | 11, 58    |
| Power-on Reset Circuit, External .....             | 58        |
| Power-up Timer (PWRT) .....                        | 55, 59    |
| PWRT Enable (PWRTIE Bit) .....                     | 56        |
| Time-out ( $\overline{\text{TO}}$ Bit).....        | 11, 58    |
| Time-out Sequence .....                            | 62        |
| Time-out Sequence on Power-up.....                 | 64        |
| Timing Diagram .....                               | 101       |
| Prescaler, Capture.....                            | 34        |
| Prescaler, Timer0 .....                            | 27        |
| Assignment (PSA Bit) .....                         | 12, 27    |
| Rate Select (PS2:PS0 Bits) .....                   | 12, 27    |
| Switching Between Timer0 and WDT .....             | 28        |
| Prescaler, Timer1 .....                            | 30        |
| Select (T1CKPS1:T1CKPS0 Bits) .....                | 29        |
| Prescaler, Timer2 .....                            | 37        |
| Select (T2CKPS1:T2CKPS0 Bits) .....                | 31        |
| PRO MATE II Universal Device Programmer .....      | 87        |
| Program Counter                                    |           |
| PCL Register .....                                 | 9, 17     |
| PCLATH Register .....                              | 9, 17, 66 |
| Reset Conditions .....                             | 62        |
| Program Memory .....                               | 7         |
| Interrupt Vector .....                             | 7         |
| Paging .....                                       | 7, 17     |
| Program Memory Map .....                           | 7         |
| Reset Vector .....                                 | 7         |
| Program Verification .....                         | 69        |
| PWM (CCP Module) .....                             | 36        |
| CCPR1H:CCPR1L Registers .....                      | 36        |
| Duty Cycle .....                                   | 37        |
| Example Frequencies/Resolutions .....              | 37        |
| Output Diagram .....                               | 36        |
| Period .....                                       | 36        |
| Set-Up for PWM Operation.....                      | 37        |
| TMR2 to PR2 Match .....                            | 31, 36    |
| TMR2 to PR2 Match Enable (TMR2IE Bit) .....        | 14        |
| TMR2 to PR2 Match Flag (TMR2IF Bit) .....          | 15        |
| PWM (ECCP Module).....                             | 38        |
| Associated Registers.....                          | 48        |
| Direction Change in Full-Bridge Output Mode.....   | 43        |
| Effects of Reset .....                             | 48        |
| Full-Bridge Application Example.....               | 43        |
| Full-Bridge Mode .....                             | 43        |
| Half-Bridge Mode.....                              | 42        |
| Half-Bridge Output Mode Applications Example ..... | 42        |
| Output Configurations.....                         | 38        |
| Output Relationships (Active High and Low) .....   | 40        |
| Output Relationships (Active-High and Low).....    | 39        |
| Output Relationships Diagram.....                  | 39, 41    |
| Programmable Dead-band Delay .....                 | 45        |
| Setup for Operation .....                          | 48        |
| Shoot-through Current .....                        | 45        |
| Start-up Considerations.....                       | 47        |

## Q

|              |         |
|--------------|---------|
| Q-Clock..... | 37, 100 |
|--------------|---------|

# PIC16F716

## R

|                                                  |        |
|--------------------------------------------------|--------|
| RA3:RA0 .....                                    | 19     |
| RA4/T0CKI Pin .....                              | 20     |
| RAM. See Data Memory.                            |        |
| RB0 Pin .....                                    | 21     |
| Register File .....                              | 8      |
| Register File Map .....                          | 8      |
| Registers                                        |        |
| A/D ADCON0 .....                                 | 49     |
| A/D ADCON1 .....                                 | 49     |
| A/D ADRES .....                                  | 49, 50 |
| ADCON0 ADSC1:ADSC0 Bits .....                    | 49     |
| ADCON0 ADON Bit .....                            | 49     |
| ADCON0 CHS2:CHS0 Bits .....                      | 49     |
| ADCON0 GO/DONE Bit .....                         | 49, 50 |
| ADCON1 PCFG2:PCFG0 Bits .....                    | 50     |
| CCP1CON CCP1M3:CCP1M0 Bits .....                 | 33     |
| CCP1CON CCP1X:CCP1Y Bits .....                   | 33     |
| Compare (CCP Module)                             |        |
| CCPR1H:CCPR1L .....                              | 34     |
| INTCON Register                                  |        |
| RBIF .....                                       | 21     |
| PWM1CON (Enhanced PWM Configuration) .....       | 46     |
| T1CON Register                                   |        |
| T1CKPS1:T1CKPS0 Bits .....                       | 29     |
| T1OSCEN Bit .....                                | 29     |
| T1SYNC Bit .....                                 | 29     |
| TMR1CS Bit .....                                 | 29     |
| TMR1ON Bit .....                                 | 29     |
| T2CON Register T2CKPS1:T2CKPS0 Bits .....        | 31     |
| T2CON Register TMR2ON Bit .....                  | 31     |
| T2CON Register TOUTPS3:TOUTPS0 Bits .....        | 31     |
| Timer2                                           |        |
| PR2 .....                                        | 36     |
| Timer2 PR2 Register .....                        | 31     |
| Timer2 TMR2 Register .....                       | 31     |
| TMR1H Timer1 Register .....                      | 29     |
| TMR1L Timer1 Register .....                      | 29     |
| Reset .....                                      | 55, 58 |
| Brown-out Reset (BOR). See Brown-out Reset (BOR) |        |
| MCLR Reset. See MCLR                             |        |
| Power-on Reset (POR). See Power-on Reset (POR)   |        |
| Reset Conditions for PCON Register .....         | 62     |
| Reset Conditions for Program Counter .....       | 62     |
| Reset Conditions for Status Register .....       | 62     |
| Timing Diagram .....                             | 101    |
| WDT Reset. See Watchdog Timer (WDT)              |        |
| RETfie Instruction .....                         | 80     |
| RETLW Instruction .....                          | 81     |
| RETURN Instruction .....                         | 81     |
| Revision History .....                           | 113    |
| RLF Instruction .....                            | 81     |
| RRF Instruction .....                            | 82     |

## S

|                                        |            |
|----------------------------------------|------------|
| Shoot-through Current .....            | 45         |
| Sleep .....                            | 55, 58, 68 |
| Sleep Instruction .....                | 82         |
| Software Simulator (MPLAB SIM) .....   | 86         |
| Software Simulator (MPLAB SIM30) ..... | 86         |
| Special Event Trigger. See Compare     |            |
| Special Features of the CPU .....      | 55         |
| Special Function Registers .....       | 9          |
| Speed, Operating .....                 | 1          |
| Stack .....                            | 17         |

|                         |        |
|-------------------------|--------|
| Status Register .....   | 9, 66  |
| C Bit .....             | 11     |
| DC Bit .....            | 11     |
| IRP Bit .....           | 11     |
| PD Bit .....            | 11, 58 |
| RP1:RP0 Bits .....      | 11     |
| TO Bit .....            | 11, 58 |
| Z Bit .....             | 11     |
| SUBLW Instruction ..... | 82     |
| SUBWF Instruction ..... | 83     |
| SWAPF Instruction ..... | 83     |

## T

|                                                            |            |
|------------------------------------------------------------|------------|
| T1CON Register .....                                       | 9          |
| T2CON Register .....                                       | 9          |
| Timer0 .....                                               | 27         |
| Clock Source Edge Select (T0SE Bit) .....                  | 12, 27     |
| Clock Source Select (T0CS Bit) .....                       | 12, 27     |
| Overflow Enable (T0IE Bit) .....                           | 13         |
| Overflow Flag (T0IF Bit) .....                             | 13, 66     |
| Overflow Interrupt .....                                   | 28, 66     |
| Prescaler. See Prescaler, Timer0                           |            |
| Timing Diagram .....                                       | 102        |
| TMR0 Register .....                                        | 9          |
| Timer1 .....                                               | 29         |
| Clock Source Select (TMR1CS Bit) .....                     | 29         |
| External Clock Input Sync (T1SYNC Bit) .....               | 29         |
| Module On/Off (TMR1ON Bit) .....                           | 29         |
| Oscillator .....                                           | 29, 30     |
| Oscillator Enable (T1OSCEN Bit) .....                      | 29         |
| Overflow Enable (TMR1IE Bit) .....                         | 14         |
| Overflow Flag (TMR1IF Bit) .....                           | 15         |
| Overflow Interrupt .....                                   | 29, 30     |
| Prescaler. See Prescaler, Timer1                           |            |
| Special Event Trigger (CCP) .....                          | 30, 35     |
| T1CON Register .....                                       | 9          |
| Timing Diagram .....                                       | 102        |
| TMR1H Register .....                                       | 9          |
| TMR1L Register .....                                       | 9          |
| Timer2 .....                                               |            |
| Postscaler. See Postscaler, Timer2                         |            |
| PR2 Register .....                                         | 10         |
| Prescaler. See Prescaler, Timer2                           |            |
| T2CON Register .....                                       | 9          |
| TMR2 Register .....                                        | 9          |
| TMR2 to PR2 Match Enable (TMR2IE Bit) .....                | 14         |
| TMR2 to PR2 Match Flag (TMR2IF Bit) .....                  | 15         |
| TMR2 to PR2 Match Interrupt .....                          | 31, 32, 36 |
| Timing Diagrams                                            |            |
| Half-Bridge PWM Output .....                               | 42         |
| PWM Auto-Shutdown (PRSEN = 0, Auto-Restart Disabled) ..... | 47         |
| PWM Auto-Shutdown (PRSEN = 1, Auto-Restart Enabled) .....  | 47         |
| PWM Direction Change .....                                 | 44         |
| PWM Direction Change at Near 100% Duty Cycle .....         | 44         |
| Time-out Sequence on Power-up .....                        | 64         |
| Wake-up from Sleep via Interrupt .....                     | 69         |
| Timing Diagrams and Specifications .....                   | 98         |
| A/D Conversion .....                                       | 105        |
| Brown-out Reset (BOR) .....                                | 101        |
| Capture/Compare/PWM (CCP) .....                            | 103        |
| CLKOUT and I/O .....                                       | 100        |
| External Clock .....                                       | 98         |
| Oscillator Start-up Timer (OST) .....                      | 101        |
| Power-up Timer (PWRT) .....                                | 101        |

|                           |     |
|---------------------------|-----|
| Reset.....                | 101 |
| Timer0 and Timer1.....    | 102 |
| Watchdog Timer (WDT)..... | 101 |
| TRIS Instruction .....    | 83  |

## W

|                                   |            |
|-----------------------------------|------------|
| W Register .....                  | 66         |
| Wake-up from Sleep .....          | 55, 68     |
| Interrupts.....                   | 62, 63     |
| MCLR Reset .....                  | 63         |
| Timing Diagram.....               | 69         |
| WDT Reset .....                   | 63         |
| Watchdog Timer (WDT) .....        | 55, 67     |
| Enable (WDTE Bit).....            | 56, 67     |
| Postscaler. See Postscaler, WDT   |            |
| Programming Considerations .....  | 67         |
| RC Oscillator .....               | 67         |
| Time-out Period .....             | 67         |
| Timing Diagram.....               | 101        |
| WDT Reset, Normal Operation ..... | 58, 62, 63 |
| WDT Reset, Sleep .....            | 58, 62, 63 |
| WWW, On-Line Support .....        | 3          |

## X

|                         |    |
|-------------------------|----|
| XORLW Instruction ..... | 84 |
| XORWF Instruction ..... | 84 |

# PIC16F716

---

NOTES:

## PRODUCT IDENTIFICATION SYSTEM

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

| <u>PART NO.</u>   | <u>X</u>                                                          | <u>/XX</u> | <u>XXX</u> |
|-------------------|-------------------------------------------------------------------|------------|------------|
| Device            | Temperature Range                                                 | Package    | Pattern    |
| Device            | PIC16F716, PIC16F716T, VDD range 2.0V to 5.5V                     |            |            |
| Temperature Range | I = -40°C to +85°C (Industrial)<br>E = -40°C to +125°C (Extended) |            |            |
| Package           | SO = SOIC<br>P = PDIP<br>SS = SSOP                                |            |            |
| Pattern           | QTP, SQTP, Code or Special Requirements (blank otherwise)         |            |            |

**Examples:**

a) PIC16F716 -I/P 301= Industrial temp., PDIP package, QTP pattern #301.

b) PIC16F716 - E/SO = Extended temp, SOIC package

**Note 1:** T = in tape and reel SOIC and SSOP packages only.

## 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) 792-7277
3. The Microchip Worldwide Site ([www.microchip.com](http://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](http://www.microchip.com/cn)) to receive the most current information on our products.



## WORLDWIDE SALES AND SERVICE

### AMERICAS

#### Corporate Office

2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 480-792-7200  
Fax: 480-792-7277  
Technical Support: 480-792-7627  
Web Address: <http://www.microchip.com>

#### Atlanta

3780 Mansell Road, Suite 130  
Alpharetta, GA 30022  
Tel: 770-640-0034  
Fax: 770-640-0307

#### Boston

2 Lan Drive, Suite 120  
Westford, MA 01886  
Tel: 978-692-3848  
Fax: 978-692-3821

#### Chicago

333 Pierce Road, Suite 180  
Itasca, IL 60143  
Tel: 630-285-0071  
Fax: 630-285-0075

#### Dallas

4570 Westgrove Drive, Suite 160  
Addison, TX 75001  
Tel: 972-818-7423  
Fax: 972-818-2924

#### Detroit

Tri-Atria Office Building  
32255 Northwestern Highway, Suite 190  
Farmington Hills, MI 48334  
Tel: 248-538-2250  
Fax: 248-538-2260

#### Kokomo

2767 S. Albright Road  
Kokomo, IN 46902  
Tel: 765-864-8360  
Fax: 765-864-8387

#### Los Angeles

18201 Von Karman, Suite 1090  
Irvine, CA 92612  
Tel: 949-263-1888  
Fax: 949-263-1338

#### Phoenix

2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 480-792-7966  
Fax: 480-792-4338

#### San Jose

2107 North First Street, Suite 590  
San Jose, CA 95131  
Tel: 408-436-7950  
Fax: 408-436-7955

#### Toronto

6285 Northam Drive, Suite 108  
Mississauga, Ontario L4V 1X5, Canada  
Tel: 905-673-0699  
Fax: 905-673-6509

### ASIA/PACIFIC

#### Australia

Suite 22, 41 Rawson Street  
Epping 2121, NSW  
Australia  
Tel: 61-2-9868-6733  
Fax: 61-2-9868-6755

#### China - Beijing

Unit 915  
Bei Hai Wan Tai Bldg.  
No. 6 Chaoyangmen Beidajie  
Beijing, 100027, No. China  
Tel: 86-10-85282100  
Fax: 86-10-85282104

#### China - Chengdu

Rm. 2401-2402, 24th Floor,  
Ming Xing Financial Tower  
No. 88 TIDU Street  
Chengdu 610016, China  
Tel: 86-28-86766200  
Fax: 86-28-86766599

#### China - Fuzhou

Unit 28F, World Trade Plaza  
No. 71 Wusi Road  
Fuzhou 350001, China  
Tel: 86-591-7503506  
Fax: 86-591-7503521

#### China - Hong Kong SAR

Unit 901-6, Tower 2, Metroplaza  
223 Hing Fong Road  
Kwai Fong, N.T., Hong Kong  
Tel: 852-2401-1200  
Fax: 852-2401-3431

#### China - Shanghai

Room 701, Bldg. B  
Far East International Plaza  
No. 317 Xian Xia Road  
Shanghai, 200051  
Tel: 86-21-6275-5700  
Fax: 86-21-6275-5060

#### China - Shenzhen

Rm. 1812, 18/F, Building A, United Plaza  
No. 5022 Binhe Road, Futian District  
Shenzhen 518033, China  
Tel: 86-755-82901380  
Fax: 86-755-8295-1393

#### China - Shunde

Room 401, Hongjian Building  
No. 2 Fengxiangnan Road, Ronggui Town  
Shunde City, Guangdong 528303, China  
Tel: 86-765-8395507 Fax: 86-765-8395571

#### China - Qingdao

Rm. B505A, Fullhope Plaza,  
No. 12 Hong Kong Central Rd.  
Qingdao 266071, China  
Tel: 86-532-5027355 Fax: 86-532-5027205

#### India

Divyasree Chambers  
1 Floor, Wing A (A3/A4)  
No. 11, O'Shaugnessey Road  
Bangalore, 560 025, India  
Tel: 91-80-2290061 Fax: 91-80-2290062

#### Japan

Benex S-1 6F  
3-18-20, Shinyokohama  
Kohoku-Ku, Yokohama-shi  
Kanagawa, 222-0033, Japan  
Tel: 81-45-471- 6166 Fax: 81-45-471-6122

### Korea

168-1, Youngbo Bldg. 3 Floor  
Samsung-Dong, Kangnam-Ku  
Seoul, Korea 135-882  
Tel: 82-2-554-7200 Fax: 82-2-558-5932 or  
82-2-558-5934

### Singapore

200 Middle Road  
#07-02 Prime Centre  
Singapore, 188980  
Tel: 65-6334-8870 Fax: 65-6334-8850

### Taiwan

Kaohsiung Branch  
30F - 1 No. 8  
Min Chuan 2nd Road  
Kaohsiung 806, Taiwan  
Tel: 886-7-536-4818  
Fax: 886-7-536-4803

### Taiwan

Taiwan Branch  
11F-3, No. 207  
Tung Hua North Road  
Taipei, 105, Taiwan  
Tel: 886-2-2717-7175 Fax: 886-2-2545-0139

### EUROPE

#### Austria

Durisolstrasse 2  
A-4600 Wels  
Austria  
Tel: 43-7242-2244-399  
Fax: 43-7242-2244-393

#### Denmark

Regus Business Centre  
Lautrup høj 1-3  
Ballerup DK-2750 Denmark  
Tel: 45-4420-9895 Fax: 45-4420-9910

#### France

Parc d'Activite du Moulin de Massy  
43 Rue du Saule Trapu  
Batiment A - 1er Etage  
91300 Massy, France  
Tel: 33-1-69-53-63-20  
Fax: 33-1-69-30-90-79

#### Germany

Steinheilstrasse 10  
D-85737 Ismaning, Germany  
Tel: 49-89-627-144-0  
Fax: 49-89-627-144-44

#### Italy

Via Quasimodo, 12  
20025 Legnano (MI)  
Milan, Italy  
Tel: 39-0331-742611  
Fax: 39-0331-466781

#### Netherlands

P. A. De Biesbosch 14  
NL-5152 SC Drunen, Netherlands  
Tel: 31-416-690399  
Fax: 31-416-690340

#### United Kingdom

505 Eskdale Road  
Winnersh Triangle  
Wokingham  
Berkshire, England RG41 5TU  
Tel: 44-118-921-5869  
Fax: 44-118-921-5820

07/28/03