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

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

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

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contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates.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.

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.

Trademarks

The Microchip name and logo, the Microchip logo, KEELOQ®, MPLAB, PIC, PICmicro, PICSTART and PRO MATE 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.

dsPIC, dsPICDEM.net, ECONOMONITOR, FanSense, FlexROM, fuzzyLAB, In-Circuit Serial Programming, ICSP, ICEPIC, microPort, Migratable Memory, MPASM, MPLIB, MPLINK, MPSIM, PICC, PICDEM, PICDEM.net, rPIC, Select Mode 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.

© 2002, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

Printed on recycled paper.
PIC16F630/676

14-Pin FLASH-Based 8-Bit CMOS Microcontroller

Devices included in this Data Sheet:
• PIC16F630
• PIC16F676

High Performance RISC CPU:
• Only 35 instructions to learn
• All single cycle instructions (200 ns), except for program branches which are two-cycle
• Operating speed:
  - DC - 20 MHz oscillator/clock input
  - DC - 200 ns instruction cycle
• Memory
  - 1024 x 14 words of FLASH Program Memory
  - 64 x 8 bytes of Data Memory (SRAM)
  - 128 x 8 bytes of EEPROM data memory
• Interrupt capability
• 8-level hardware stack
• Direct, Indirect and Relative Addressing modes

Peripheral Features:
• 12 I/O pins with individual direction control
• High current sink/source for direct LED drive
• Analog comparator module with:
  - One analog comparator
  - Programmable on-chip comparator voltage reference (CVREF) module
  - Programmable input multiplexing from device inputs
  - Comparator output is externally accessible
• Analog-to-Digital Converter module (PIC16F676):
  - 10-bit resolution
  - Programmable 8-channel input
  - Voltage reference input
• Timer0: 8-bit timer/counter with 8-bit programmable prescaler
• Enhanced Timer1:
  - 16-bit timer/counter with prescaler
  - External Gate Input mode
  - Option to use OSC1 and OSC2 in LP mode as Timer1 oscillator, if INTOSC Oscillator mode selected

Special Microcontroller Features:
• Low power Power-on Reset (POR)
• Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
• Low power Brown-out Detect (BOD)
• Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
• Multiplexed MCLR pin
• Interrupt-on-pin change
• Individual programmable weak pull-ups
• Programmable code protection
• Power saving SLEEP mode
• Selectable oscillator options
  - RC: External RC oscillator
  - INTOSC: Precision 4 MHz internal oscillator
  - EC: External Clock input
  - XT: Standard crystal/resonator
  - HS: High speed crystal/resonator
  - LP: Power saving, low frequency crystal
• Fast Wake-up when running from INTOSC (8 clocks/2 µs typical)
• In-Circuit Serial Programming™ (ICSP™) via two pins
• Four user programmable ID locations

CMOS Technology:
• Low power, high speed CMOS FLASH technology
• Fully static design
• Wide operating voltage range
  - PIC16F630/676 – 2.0V to 5.5V
• Industrial and Extended temperature range
• Low power consumption
  - < 1.0 mA @ 5.5V, 4.0 MHz
  - 400 µA @ 2.0V, 4.0 MHz
  - < 1.0 µA typical standby current @ 2.0V
TABLE 1: SUMMARY OF FEATURES

<table>
<thead>
<tr>
<th>Device</th>
<th>Program Memory</th>
<th>Data SRAM (Bytes)</th>
<th>EEPROM (Bytes)</th>
<th>I/O</th>
<th>10-bit ADC</th>
<th>Comparators</th>
<th>Timers 8-bit/16-bit</th>
<th>CCP (PWM)</th>
<th>BOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIC16F630</td>
<td>1792</td>
<td>1024</td>
<td>64</td>
<td>12</td>
<td>—</td>
<td>1</td>
<td>1 / 1</td>
<td>—</td>
<td>Yes</td>
</tr>
<tr>
<td>PIC16F676</td>
<td>1792</td>
<td>1024</td>
<td>64</td>
<td>12</td>
<td>8</td>
<td>1</td>
<td>1 / 1</td>
<td>—</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Pin Diagrams

14-pin PDIP, SOIC, TSSOP

- PIC16F630
  - VDD
  - RA5/T1CKI/OSC1/CLKIN
  - RA4/T1G/OSC2/CLKOUT
  - RA3/MCLR/VPP
  - RC5
  - RC4
  - RC3

- PIC16F676
  - VDD
  - RA5/T1CKI/OSC1/CLKIN
  - RA4/T1G/OSC2/AN3/CLKOUT
  - RA3/MCLR/VPP
  - RC5
  - RC4
  - RC3/AN7

- RA0/CIN+/ICSPDAT
- RA1/CIN-/ICSPCLK
- RA2/COUT/T0CKI/INT
- RC0
- RC1
- RC2
- RA0/AN0/CIN+/ICSPDAT
- RA1/AN1/CIN-/VREF/ICSPCLK
- RA2/AN2/COUT/T0CKI/INT
- RC0/AN4
- RC1/AN5
- RC2/AN6
Table of Contents

1.0 Device Overview ........................................................................................................................................... 5
2.0 Memory Organization ...................................................................................................................................... 7
3.0 Ports A and C .................................................................................................................................................. 19
4.0 Timer0 Module ................................................................................................................................................ 29
5.0 Timer1 Module with Gate Control ................................................................................................................ 32
6.0 Comparator Module ....................................................................................................................................... 37
7.0 Analog-to-Digital Converter (A/D) Module (PIC16F676 only) ........................................................................ 43
8.0 Data EEPROM Memory .................................................................................................................................. 49
9.0 Special Features of the CPU .......................................................................................................................... 53
10.0 Instruction Set Summary .............................................................................................................................. 71
11.0 Development Support .................................................................................................................................. 79
12.0 Electrical Specifications .................................................................................................................................. 85
13.0 Packaging Information .................................................................................................................................. 103
Appendix A: Data Sheet Revision History .............................................................................................................. 107
Appendix B: Device Differences .......................................................................................................................... 107
Appendix C: Device Migrations .......................................................................................................................... 108
Appendix D: Migrating from other PICmicro Devices .......................................................................................... 108
Appendix E: Development Tool Version Requirements .......................................................................................... 109
Index ........................................................................................................................................................................ 111
On-Line Support ................................................................................................................................................. 115
Reader Response .................................................................................................................................................. 116
Product Identification System ............................................................................................................................. 117

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 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 to receive the most current information on all of our products.
1.0 DEVICE OVERVIEW

This document contains device specific information for the PIC16F630/676. 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. 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.

The PIC16F630 and PIC16F676 devices are covered by this Data Sheet. They are identical, except the PIC16F676 has a 10-bit A/D converter. They come in 14-pin PDIP, SOIC and TSSOP packages. Figure 1-1 shows a block diagram of the PIC16F630/676 devices. Table 1-1 shows the pinout description.

FIGURE 1-1: PIC16F630/676 BLOCK DIAGRAM
<table>
<thead>
<tr>
<th>Name</th>
<th>Function</th>
<th>Input Type</th>
<th>Output Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RA0/AN0/CIN+/ICSPDAT</td>
<td>RA0</td>
<td>TTL</td>
<td>CMOS</td>
<td>Bi-directional I/O w/ programmable pull-up and Interrupt-on-change</td>
</tr>
<tr>
<td>AN0</td>
<td>AN</td>
<td>—</td>
<td>A/D Channel 0 input</td>
<td></td>
</tr>
<tr>
<td>CIN+</td>
<td>AN</td>
<td>—</td>
<td>Comparator input</td>
<td></td>
</tr>
<tr>
<td>ICSPDAT</td>
<td>TTL</td>
<td>CMOS</td>
<td>Serial Programming Data I/O</td>
<td></td>
</tr>
<tr>
<td>RA1/AN1/CIN-/VREF/ICSPCLK</td>
<td>RA1</td>
<td>TTL</td>
<td>CMOS</td>
<td>Bi-directional I/O w/ programmable pull-up and Interrupt-on-change</td>
</tr>
<tr>
<td>AN1</td>
<td>AN</td>
<td>—</td>
<td>A/D Channel 1 input</td>
<td></td>
</tr>
<tr>
<td>CIN-</td>
<td>AN</td>
<td>—</td>
<td>Comparator input</td>
<td></td>
</tr>
<tr>
<td>VREF</td>
<td>AN</td>
<td>—</td>
<td>External Voltage reference</td>
<td></td>
</tr>
<tr>
<td>ICSPCLK</td>
<td>ST</td>
<td>—</td>
<td>Serial Programming Clock</td>
<td></td>
</tr>
<tr>
<td>RA2/AN2/COUT/T0CKI/INT</td>
<td>RA2</td>
<td>ST</td>
<td>CMOS</td>
<td>Bi-directional I/O w/ programmable pull-up and Interrupt-on-change</td>
</tr>
<tr>
<td>AN2</td>
<td>AN</td>
<td>—</td>
<td>A/D Channel 2 input</td>
<td></td>
</tr>
<tr>
<td>COUT</td>
<td>—</td>
<td>CMOS</td>
<td>Comparator output</td>
<td></td>
</tr>
<tr>
<td>T0CKI</td>
<td>ST</td>
<td>—</td>
<td>Timer0 clock input</td>
<td></td>
</tr>
<tr>
<td>INT</td>
<td>ST</td>
<td>—</td>
<td>External Interrupt</td>
<td></td>
</tr>
<tr>
<td>RA3/MCLR/VPP</td>
<td>RA3</td>
<td>TTL</td>
<td>—</td>
<td>Input port with Interrupt-on-change</td>
</tr>
<tr>
<td>MCLR</td>
<td>ST</td>
<td>—</td>
<td>Master Clear</td>
<td></td>
</tr>
<tr>
<td>VPP</td>
<td>HV</td>
<td>—</td>
<td>Programming voltage</td>
<td></td>
</tr>
<tr>
<td>RA4/T1G/AN3/OSC2/CLKOUT</td>
<td>RA4</td>
<td>TTL</td>
<td>CMOS</td>
<td>Bi-directional I/O w/ programmable pull-up and Interrupt-on-change</td>
</tr>
<tr>
<td>T1G</td>
<td>ST</td>
<td>—</td>
<td>Timer1 gate</td>
<td></td>
</tr>
<tr>
<td>AN3</td>
<td>AN3</td>
<td>—</td>
<td>A/D Channel 3 input</td>
<td></td>
</tr>
<tr>
<td>OSC2</td>
<td>—</td>
<td>XTAL</td>
<td>Crystal/Resonator</td>
<td></td>
</tr>
<tr>
<td>CLKOUT</td>
<td>—</td>
<td>CMOS</td>
<td>Fosc/4 output</td>
<td></td>
</tr>
<tr>
<td>RA5/T1CKI/OSC1/CLKIN</td>
<td>RA5</td>
<td>TTL</td>
<td>CMOS</td>
<td>Bi-directional I/O w/ programmable pull-up and Interrupt-on-change</td>
</tr>
<tr>
<td>T1CKI</td>
<td>ST</td>
<td>—</td>
<td>Timer1 clock</td>
<td></td>
</tr>
<tr>
<td>OSC1</td>
<td>XTAL</td>
<td>—</td>
<td>Crystal/Resonator</td>
<td></td>
</tr>
<tr>
<td>CLKIN</td>
<td>ST</td>
<td>—</td>
<td>External clock input/RC oscillator connection</td>
<td></td>
</tr>
<tr>
<td>RC0/AN4</td>
<td>RC0</td>
<td>TTL</td>
<td>CMOS</td>
<td>Bi-directional I/O</td>
</tr>
<tr>
<td>AN4</td>
<td>AN4</td>
<td>—</td>
<td>A/D Channel 4 input</td>
<td></td>
</tr>
<tr>
<td>RC1/AN5</td>
<td>RC1</td>
<td>TTL</td>
<td>CMOS</td>
<td>Bi-directional I/O</td>
</tr>
<tr>
<td>AN5</td>
<td>AN5</td>
<td>—</td>
<td>A/D Channel 5 input</td>
<td></td>
</tr>
<tr>
<td>RC2/AN6</td>
<td>RC2</td>
<td>TTL</td>
<td>CMOS</td>
<td>Bi-directional I/O</td>
</tr>
<tr>
<td>AN6</td>
<td>AN6</td>
<td>—</td>
<td>A/D Channel 6 input</td>
<td></td>
</tr>
<tr>
<td>RC3/AN7</td>
<td>RC3</td>
<td>TTL</td>
<td>CMOS</td>
<td>Bi-directional I/O</td>
</tr>
<tr>
<td>AN7</td>
<td>AN7</td>
<td>—</td>
<td>A/D Channel 7 input</td>
<td></td>
</tr>
<tr>
<td>RC4</td>
<td>RC4</td>
<td>TTL</td>
<td>CMOS</td>
<td>Bi-directional I/O</td>
</tr>
<tr>
<td>RC5</td>
<td>RC5</td>
<td>TTL</td>
<td>CMOS</td>
<td>Bi-directional I/O</td>
</tr>
<tr>
<td>VSS</td>
<td>VSS</td>
<td>Power</td>
<td>—</td>
<td>Ground reference</td>
</tr>
<tr>
<td>VDD</td>
<td>Vdd</td>
<td>Power</td>
<td>—</td>
<td>Positive supply</td>
</tr>
</tbody>
</table>

Legend:  
Shade = PIC16F676 only  
TTL = TTL input buffer  
ST = Schmitt Trigger input buffer
2.0 MEMORY ORGANIZATION

2.1 Program Memory Organization

The PIC16F630/676 devices have a 13-bit program counter capable of addressing an 8K x 14 program memory space. Only the first 1K x 14 (0000h - 03FFh) for the PIC16F630/676 devices is physically implemented. Accessing a location above these boundaries will cause a wrap around within the first 1K x 14 space. The RESET vector is at 0000h and the interrupt vector is at 0004h (see Figure 2-1).

FIGURE 2-1: PROGRAM MEMORY MAP AND STACK FOR THE PIC16F630/676

2.2 Data Memory Organization

The data memory (see Figure 2-2) is partitioned into two banks, which contain the General Purpose registers and the Special Function registers. The Special Function registers are located in the first 32 locations of each bank. Register locations 20h-5Fh are General Purpose registers, implemented as static RAM and are mapped across both banks. All other RAM is unimplemented and returns ‘0’ when read. RP0 (STATUS<5>) is the bank select bit.

- RP0 = 0 Bank 0 is selected
- RP0 = 1 Bank 1 is selected

Note: The IRP and RP1 bits STATUS<7:6> are reserved and should always be maintained as ‘0’s.

2.2.1 GENERAL PURPOSE REGISTER FILE

The register file is organized as 64 x 8 in the PIC16F630/676 devices. Each register is accessed, either directly or indirectly, through the File Select Register FSR (see Section 2.4).
2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function registers are registers used by the CPU and peripheral functions for controlling the desired operation of the device (see Table 2-1). These registers are static RAM.

The special registers can be classified into two sets: core and peripheral. The Special Function registers associated with the “core” are described in this section. Those related to the operation of the peripheral features are described in the section of that peripheral feature.

FIGURE 2-2: DATA MEMORY MAP OF THE PIC16F630/676

<table>
<thead>
<tr>
<th>File Address</th>
<th>File Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Indirect addr.(1)</td>
<td>Indirect addr.(1)</td>
</tr>
<tr>
<td>TMR0 01h</td>
<td>OPTION_REG 81h</td>
</tr>
<tr>
<td>PCL 02h</td>
<td>PCL 82h</td>
</tr>
<tr>
<td>STATUS 03h</td>
<td>STATUS 83h</td>
</tr>
<tr>
<td>FSR 04h</td>
<td>FSR 84h</td>
</tr>
<tr>
<td>PORTA 05h</td>
<td>TRISA 85h</td>
</tr>
<tr>
<td>06h</td>
<td>86h</td>
</tr>
<tr>
<td>PORTC 07h</td>
<td>TRISC 87h</td>
</tr>
<tr>
<td>08h</td>
<td>88h</td>
</tr>
<tr>
<td>09h</td>
<td>89h</td>
</tr>
<tr>
<td>PCLATH 0Ah</td>
<td>PCLATH 8Ah</td>
</tr>
<tr>
<td>INTCN 0Bh</td>
<td>INTCN 8Bh</td>
</tr>
<tr>
<td>PIR1 0Ch</td>
<td>PIE1 8Ch</td>
</tr>
<tr>
<td>0Dh</td>
<td>8Dh</td>
</tr>
<tr>
<td>TMR1L 0Eh</td>
<td>PCON 8Eh</td>
</tr>
<tr>
<td>TMR1H 0Fh</td>
<td>8Fh</td>
</tr>
<tr>
<td>T1CON 10h</td>
<td>OSCCAL 90h</td>
</tr>
<tr>
<td>11h</td>
<td>91h</td>
</tr>
<tr>
<td>12h</td>
<td>92h</td>
</tr>
<tr>
<td>13h</td>
<td>93h</td>
</tr>
<tr>
<td>14h</td>
<td>94h</td>
</tr>
<tr>
<td>15h</td>
<td>95h</td>
</tr>
<tr>
<td>16h</td>
<td>96h</td>
</tr>
<tr>
<td>17h</td>
<td>97h</td>
</tr>
<tr>
<td>18h</td>
<td>98h</td>
</tr>
<tr>
<td>CMCON 19h</td>
<td>VRCON 99h</td>
</tr>
<tr>
<td>1Ah</td>
<td>9Ah</td>
</tr>
<tr>
<td>1Bh</td>
<td>EEDAT 9Ah</td>
</tr>
<tr>
<td>1Ch</td>
<td>EEADR 9Bh</td>
</tr>
<tr>
<td>1Dh</td>
<td>EECON1 9Ch</td>
</tr>
<tr>
<td>ADRESH(2) 1Eh</td>
<td>EECON2(1) 9Dh</td>
</tr>
<tr>
<td>ADCONH(3) 1Fh</td>
<td>ADRESH(2) 9Eh</td>
</tr>
<tr>
<td>20h</td>
<td>ADCONL(3) 9Fh</td>
</tr>
<tr>
<td>20h-5Fh</td>
<td>accesses 20h-5Fh</td>
</tr>
</tbody>
</table>

General Purpose Registers
64 Bytes

Bank 0
Bank 1

Unimplemented data memory locations, read as ‘0’.
1: Not a physical register.
2: PIC16F676 only.
### Table 2-1: PIC16F630/676 Special Registers Summary Bank 0

<table>
<thead>
<tr>
<th>Addr</th>
<th>Name</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>Value on POR Reset</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>00h</td>
<td>INDF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>xxxxx xxxxx</td>
<td>18,61</td>
</tr>
<tr>
<td>01h</td>
<td>TMR0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>xxxxx xxxxx</td>
<td>29</td>
</tr>
<tr>
<td>02h</td>
<td>PCL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>xxxxx xxxxx</td>
<td>17</td>
</tr>
<tr>
<td>03h</td>
<td>STATUS</td>
<td>IRP(2)</td>
<td>RP1(2)</td>
<td>RP0</td>
<td>TO</td>
<td>PD</td>
<td>Z</td>
<td>DC</td>
<td>C</td>
<td>0001 1xxx</td>
<td>11</td>
</tr>
<tr>
<td>04h</td>
<td>FSR</td>
<td>xxxxx</td>
<td>xxxxx</td>
<td>xxxxx</td>
<td>xxxxx</td>
<td>xxxxx</td>
<td>xxxxx</td>
<td>xxxxx</td>
<td>xxxxx</td>
<td>xxxxx xxxxx</td>
<td>18</td>
</tr>
<tr>
<td>05h</td>
<td>PORTA</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—xx xxxxx</td>
<td>19</td>
</tr>
<tr>
<td>06h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>07h</td>
<td>PORTC</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—xx xxxxx</td>
<td>26</td>
</tr>
<tr>
<td>08h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>09h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>0Ah</td>
<td>PCLATH</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>---0 0000</td>
<td>17</td>
</tr>
<tr>
<td>0Bh</td>
<td>INTCON</td>
<td>GIE</td>
<td>PEIE</td>
<td>T0IE</td>
<td>INTE</td>
<td>RAIE</td>
<td>T0IF</td>
<td>INTF</td>
<td>RAIF</td>
<td>0000 0000</td>
<td>13</td>
</tr>
<tr>
<td>0Ch</td>
<td>PIR1</td>
<td>EEIF</td>
<td>ADIF</td>
<td>—</td>
<td>—</td>
<td>CMIF</td>
<td>—</td>
<td>—</td>
<td>TMR1IF</td>
<td>00-- 0--0</td>
<td>15</td>
</tr>
<tr>
<td>0Dh</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>0Eh</td>
<td>TMR1L</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>0Fh</td>
<td>TMR1H</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>10h</td>
<td>TICON</td>
<td></td>
<td></td>
<td></td>
<td>T1GE</td>
<td>T1CKPS1</td>
<td>T1CKPS0</td>
<td>T1OSCEN</td>
<td>T1SYNC</td>
<td>TMR1CS</td>
<td>TMR1ON</td>
</tr>
<tr>
<td>11h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>12h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>13h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>14h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>15h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>16h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>17h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>18h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>19h</td>
<td>CMCON</td>
<td></td>
<td></td>
<td>—</td>
<td>—</td>
<td>COUT</td>
<td>—</td>
<td>CINV</td>
<td>CIS</td>
<td>CM2</td>
<td>CM1</td>
</tr>
<tr>
<td>1Ah</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>1Bh</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>1Ch</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>1Dh</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>1Eh</td>
<td>ADRESH(3)</td>
<td></td>
<td></td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>1Fh</td>
<td>ADCON0(3)</td>
<td>ADFM</td>
<td>VCFG</td>
<td>—</td>
<td>CHS2</td>
<td>CHS1</td>
<td>CHS0</td>
<td>GO/DONE</td>
<td>ADON</td>
<td>00-0 0000</td>
<td>45,61</td>
</tr>
</tbody>
</table>

**Legend:**
- — Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition shaded = unimplemented

**Note:**
1: Other (non Power-up) Resets include MCLR Reset, Brown-out Reset and Watchdog Timer Reset during normal operation.
2: IRP & RP1 bits are reserved, always maintain these bits clear.
3: PIC16F676 only
### TABLE 2-2: PIC16F630/676 SPECIAL FUNCTION REGISTERS SUMMARY BANK 1

<table>
<thead>
<tr>
<th>Addr</th>
<th>Name</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>Value on POR Reset</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bank 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>80h</td>
<td>INDF (5)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>xxxxx xxxxx</td>
<td>18,61</td>
</tr>
<tr>
<td>81h</td>
<td>OPTION_REG (6)</td>
<td>RAPI</td>
<td>INTEDG</td>
<td>T0CS</td>
<td>T0SE</td>
<td>PSA</td>
<td>PS2</td>
<td>PS1</td>
<td>PS0</td>
<td>1111 1111</td>
<td>12,30</td>
</tr>
<tr>
<td>82h</td>
<td>PCL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0000 0000</td>
<td>17</td>
</tr>
<tr>
<td>83h</td>
<td>STATUS</td>
<td>IR(2)</td>
<td>RP1(2)</td>
<td>RP0</td>
<td>TO</td>
<td>TOE</td>
<td>PSA</td>
<td>PS2</td>
<td>PS1</td>
<td>PS0</td>
<td>11</td>
</tr>
<tr>
<td>84h</td>
<td>FSR (7)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>xxxxx xxxxx</td>
<td>18</td>
</tr>
<tr>
<td>85h</td>
<td>TRISA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TRISA2</td>
<td>TRISA3</td>
<td>TRISA4</td>
<td>TRISA5</td>
<td>TRIS0 ---11 1111</td>
<td>19</td>
</tr>
<tr>
<td>86h</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>87h</td>
<td>TRISC</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>TRISC2</td>
<td>TRISC3</td>
<td>TRISC4</td>
<td>TRISC5</td>
<td>TRIS0 ---11 1111</td>
<td></td>
</tr>
<tr>
<td>88h</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>89h</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8Ah</td>
<td>PCLATH</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Write buffer for upper 5 bits of program counter</td>
<td>---0 0000</td>
</tr>
<tr>
<td>8Bh</td>
<td>INTCON</td>
<td>GIE</td>
<td>PEIE</td>
<td>T0IE</td>
<td>INTE</td>
<td>RAIE</td>
<td>T0IF</td>
<td>INTF</td>
<td>RAIF</td>
<td>0000 0000</td>
<td>13</td>
</tr>
<tr>
<td>8Ch</td>
<td>PIE1</td>
<td>EEIE</td>
<td>ADIE</td>
<td></td>
<td>CMIE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>00-- 0--0</td>
<td>14</td>
</tr>
<tr>
<td>8Dh</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8Eh</td>
<td>PCON</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>POR BOR</td>
<td></td>
</tr>
<tr>
<td>8Fh</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>90h</td>
<td>OSCCAL</td>
<td>CAL5</td>
<td>CAL4</td>
<td>CAL3</td>
<td>CAL2</td>
<td>CAL1</td>
<td>CAL0</td>
<td></td>
<td></td>
<td>1000 00--</td>
<td>16</td>
</tr>
<tr>
<td>91h</td>
<td>ANSEL(3)</td>
<td>ANS7</td>
<td>ANS6</td>
<td>ANS5</td>
<td>ANS4</td>
<td>ANS3</td>
<td>ANS2</td>
<td>ANS1</td>
<td>ANS0</td>
<td>1111 1111</td>
<td>46</td>
</tr>
<tr>
<td>92h</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>93h</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>94h</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>95h</td>
<td>WPUA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>WPUA2</td>
<td>WPUA1</td>
<td>WPUA4</td>
<td>WPUA5</td>
<td>WPUA0 ---11-111</td>
<td>20</td>
</tr>
<tr>
<td>96h</td>
<td>IOCA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>IOCA2</td>
<td>IOCA1</td>
<td>IOCA4</td>
<td>IOCA5</td>
<td>IOCA0 ---00 0000</td>
<td>21</td>
</tr>
<tr>
<td>97h</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>98h</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>99h</td>
<td>VRCON</td>
<td>VREN</td>
<td></td>
<td>VRR</td>
<td></td>
<td>VR3</td>
<td>VR2</td>
<td>VR1</td>
<td>VR0</td>
<td>0--0-- 0000</td>
<td>42</td>
</tr>
<tr>
<td>9Ah</td>
<td>EEDAT</td>
<td></td>
<td></td>
<td></td>
<td>WRERR</td>
<td>WREN</td>
<td>WR</td>
<td>RD</td>
<td></td>
<td>0000 0000</td>
<td>49</td>
</tr>
<tr>
<td>9Bh</td>
<td>EEADR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0000 0000</td>
<td>49</td>
</tr>
<tr>
<td>9Ch</td>
<td>EECON1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>WRERR</td>
<td>WREN</td>
<td>WR</td>
<td>RD</td>
<td>0000 0000</td>
<td>50</td>
</tr>
<tr>
<td>9Dh</td>
<td>EECON2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0000 0000</td>
<td>49</td>
</tr>
<tr>
<td>9Eh</td>
<td>ADRESL(3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Least Significant 2 bits of the left shifted result or 8 bits of the right shifted result</td>
<td>xxxxx xxxxx</td>
</tr>
<tr>
<td>9Fh</td>
<td>ADCON1(3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Legend:**
- = Unimplemented locations read as '0', u = unchanged, x = unknown, q = value depends on condition, shaded = unimplemented

**Note 1:** Other (non Power-up) Resets include MCLR Reset, Brown-out Reset and Watchdog Timer Reset during normal operation.

**Note 2:** IRP & RP1 bits are reserved, always maintain these bits clear.

**Note 3:** PIC16F676 only
2.2.2.1 STATUS Register

The STATUS register, shown in Register 2-1, contains:
- the arithmetic status of the ALU
- the RESET status
- the bank select bits for data memory (SRAM)

The STATUS register can be the destination for any instruction, like any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the TO and PD bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

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

It is recommended, therefore, that only BCF, BSF, SWAPF and MOVWF instructions are used to alter the STATUS register, because these instructions do not affect any STATUS bits. For other instructions not affecting any STATUS bits, see the “Instruction Set Summary”.

Note 1: Bits IRP and RP1 (STATUS<7:6>) are not used by the PIC16F630/676 and should be maintained as clear. Use of these bits is not recommended, since this may affect upward compatibility with future products.

2: The C and DC bits operate as a Borrow and Digit Borrow out bit, respectively, in subtraction. See the SUBLW and SUBWF instructions for examples.

### REGISTER 2-1: STATUS — STATUS REGISTER (ADDRESS: 03h OR 83h)

<table>
<thead>
<tr>
<th>Reserved</th>
<th>Reserved</th>
<th>R/W-0</th>
<th>R-1</th>
<th>R-1</th>
<th>R/W-x</th>
<th>R/W-x</th>
<th>R/W-x</th>
</tr>
</thead>
<tbody>
<tr>
<td>bit 7</td>
<td>IRP: This bit is reserved and should be maintained as ‘0’</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 6</td>
<td>RP1: This bit is reserved and should be maintained as ‘0’</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 5</td>
<td>RP0: Register Bank Select bit (used for direct addressing)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = Bank 1 (80h - FFh)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = Bank 0 (00h - 7Fh)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 4</td>
<td>TO: Time-out bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = After power-up, CLRWD T instruction, or SLEEP instruction</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = A WDT time-out occurred</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 3</td>
<td>PD: Power-down bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = After power-up or by the CLRWD T instruction</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = By execution of the SLEEP instruction</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 2</td>
<td>Z: Zero bit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = The result of an arithmetic or logic operation is zero</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = The result of an arithmetic or logic operation is not zero</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 1</td>
<td>DC: Digit carry/borrow bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>For borrow, the polarity is reversed.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = A carry-out from the 4th low order bit of the result occurred</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = No carry-out from the 4th low order bit of the result</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 0</td>
<td>C: Carry/borrow bit (ADDWF, ADDLW, SUBLW, SUBWF instructions)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 = A carry-out from the Most Significant bit of the result occurred</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 = No carry-out from the Most Significant bit of the result occurred</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: 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
2.2.2.2 OPTION Register

The OPTION register is a readable and writable register, which contains various control bits to configure:

- TMR0/WDT prescaler
- External RA2/INT interrupt
- TMR0
- Weak pull-ups on PORTA

REGISTER 2-2: OPTION_REG — OPTION REGISTER (ADDRESS: 81h)

<table>
<thead>
<tr>
<th>R/W-1</th>
<th>R/W-1</th>
<th>R/W-1</th>
<th>R/W-1</th>
<th>R/W-1</th>
<th>R/W-1</th>
<th>R/W-1</th>
<th>R/W-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAPU</td>
<td>INTEDG</td>
<td>T0CS</td>
<td>T0SE</td>
<td>PSA</td>
<td>PS2</td>
<td>PS1</td>
<td>PS0</td>
</tr>
</tbody>
</table>

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

bit 6  
**INTEDG**: Interrupt Edge Select bit
1 = Interrupt on rising edge of RA2/INT pin
0 = Interrupt on falling edge of RA2/INT pin

bit 5  
**T0CS**: TMR0 Clock Source Select bit
1 = Transition on RA2/T0CKI pin
0 = Internal instruction cycle clock (CLKOUT)

bit 4  
**T0SE**: TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on RA2/T0CKI pin
0 = Increment on low-to-high transition on RA2/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

<table>
<thead>
<tr>
<th>Bit Value</th>
<th>TMR0 Rate</th>
<th>WDT Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1:2</td>
<td>1:1</td>
</tr>
<tr>
<td>001</td>
<td>1:4</td>
<td>1:2</td>
</tr>
<tr>
<td>010</td>
<td>1:8</td>
<td>1:4</td>
</tr>
<tr>
<td>011</td>
<td>1:16</td>
<td>1:8</td>
</tr>
<tr>
<td>100</td>
<td>1:32</td>
<td>1:16</td>
</tr>
<tr>
<td>101</td>
<td>1:64</td>
<td>1:32</td>
</tr>
<tr>
<td>110</td>
<td>1:128</td>
<td>1:64</td>
</tr>
<tr>
<td>111</td>
<td>1:256</td>
<td>1:128</td>
</tr>
</tbody>
</table>

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 the various enable and flag bits for TMR0 register overflow, PORTA change and external RA2/INT pin interrupts.

### REGISTER 2-3: INTCON — INTERRUPT CONTROL REGISTER (ADDRESS: 0Bh OR 8Bh)

<table>
<thead>
<tr>
<th>Bit</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>GIE</td>
<td>Global Interrupt Enable bit</td>
</tr>
<tr>
<td>6</td>
<td>PEIE</td>
<td>Peripheral Interrupt Enable bit</td>
</tr>
<tr>
<td>5</td>
<td>T0IE</td>
<td>TMR0 Overflow Interrupt Enable bit</td>
</tr>
<tr>
<td>4</td>
<td>INTE</td>
<td>RA2/INT External Interrupt Enable bit</td>
</tr>
<tr>
<td>3</td>
<td>RAIE</td>
<td>Port Change Interrupt Enable bit</td>
</tr>
<tr>
<td>2</td>
<td>T0IF</td>
<td>TMR0 Overflow Interrupt Flag bit</td>
</tr>
<tr>
<td>1</td>
<td>INTF</td>
<td>RA2/INT External Interrupt Flag bit</td>
</tr>
<tr>
<td>0</td>
<td>RAIF</td>
<td>Port Change Interrupt Flag bit</td>
</tr>
</tbody>
</table>

**Note:** Interrupt flag bits are 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.

<table>
<thead>
<tr>
<th>Bit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>GIE: Global Interrupt Enable bit</td>
</tr>
<tr>
<td>6</td>
<td>PEIE: Peripheral Interrupt Enable bit</td>
</tr>
<tr>
<td>5</td>
<td>T0IE: TMR0 Overflow Interrupt Enable bit</td>
</tr>
<tr>
<td>4</td>
<td>INTE: RA2/INT External Interrupt Enable bit</td>
</tr>
<tr>
<td>3</td>
<td>RAIE: Port Change Interrupt Enable bit</td>
</tr>
<tr>
<td>2</td>
<td>T0IF: TMR0 Overflow Interrupt Flag bit</td>
</tr>
<tr>
<td>1</td>
<td>INTF: RA2/INT External Interrupt Flag bit</td>
</tr>
<tr>
<td>0</td>
<td>RAIF: Port Change Interrupt Flag bit</td>
</tr>
</tbody>
</table>

**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

Note 1: IOCA register must also be enabled.

Note 2: T0IF bit is set when Timer0 rolls over. Timer0 is unchanged on RESET and should be initialized before clearing T0IF bit.
2.2.2.4 PIE1 Register

The PIE1 register contains the interrupt enable bits, as shown in Register 2-4.

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

### REGISTER 2-4: PIE1 — PERIPHERAL INTERRUPT ENABLE REGISTER 1 (ADDRESS: 8Ch)

<table>
<thead>
<tr>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5-4</th>
<th>Bit 3</th>
<th>Bit 2-1</th>
<th>Bit 0</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>EEIE</strong></td>
<td><strong>ADIE</strong></td>
<td>—</td>
<td>—</td>
<td>—</td>
<td><strong>TMR1IE</strong></td>
</tr>
</tbody>
</table>

|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|

- **EEIE**: EE Write Complete Interrupt Enable bit
  - 1 = Enables the EE write complete interrupt
  - 0 = Disables the EE write complete interrupt

- **ADIE**: A/D Converter Interrupt Enable bit (PIC16F676 only)
  - 1 = Enables the A/D converter interrupt
  - 0 = Disables the A/D converter interrupt

- **CMIE**: Comparator Interrupt Enable bit
  - 1 = Enables the comparator interrupt
  - 0 = Disables the comparator interrupt

- **TMR1IE**: TMR1 Overflow Interrupt Enable bit
  - 1 = Enables the TMR1 overflow interrupt
  - 0 = Disables the TMR1 overflow interrupt
### 2.2.2.5 PIR1 Register

The PIR1 register contains the interrupt flag bits, as shown in Register 2-5.

#### Note:
Interrupt flag bits are 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 — PERIPHERAL INTERRUPT REGISTER 1 (ADDRESS: 0Ch)

<table>
<thead>
<tr>
<th></th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>U-0</th>
<th>U-0</th>
<th>R/W-0</th>
<th>U-0</th>
<th>U-0</th>
<th>R/W-0</th>
</tr>
</thead>
<tbody>
<tr>
<td>EEIF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADIF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CMIF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TMR1IF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **bit 7** EEIF: EEPROM Write Operation Interrupt Flag bit
  - 1 = The write operation completed (must be cleared in software)
  - 0 = The write operation has not completed or has not been started

- **bit 6** ADIF: A/D Converter Interrupt Flag bit (PIC16F676 only)
  - 1 = The A/D conversion is complete (must be cleared in software)
  - 0 = The A/D conversion is not complete

- **bit 5-4** Unimplemented: Read as ‘0’

- **bit 3** CMIF: Comparator Interrupt Flag bit
  - 1 = Comparator input has changed (must be cleared in software)
  - 0 = Comparator input has not changed

- **bit 2-1** Unimplemented: Read as ‘0’

- **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
2.2.2.6 PCON Register

The Power Control (PCON) register contains flag bits to differentiate between a:

- Power-on Reset (POR)
- Brown-out Reset (BOR)
- Watchdog Timer Reset (WDT)
- External MCLR Reset

The PCON Register bits are shown in Register 2-6.

**REGISTER 2-6: PCON — POWER CONTROL REGISTER (ADDRESS: 8Eh)**

<table>
<thead>
<tr>
<th>U-0</th>
<th>U-0</th>
<th>U-0</th>
<th>U-0</th>
<th>U-0</th>
<th>U-0</th>
<th>R/W-0</th>
<th>R/W-x</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>POR</td>
<td>BOD</td>
</tr>
</tbody>
</table>

- **bit 7:** 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:** BOD: Brown-out Detect 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)

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

2.2.2.7 OSCCAL Register

The Oscillator Calibration register (OSCCAL) is used to calibrate the internal 4 MHz oscillator. It contains 6 bits to adjust the frequency up or down to achieve 4 MHz.

The OSCCAL register bits are shown in Register 2-7.

**REGISTER 2-7: OSCCAL — INTERNAL OSCILLATOR CALIBRATION REGISTER (ADDRESS: 90h)**

<table>
<thead>
<tr>
<th>R/W-1</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>U-0</th>
<th>U-0</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAL5</td>
<td>CAL4</td>
<td>CAL3</td>
<td>CAL2</td>
<td>CAL1</td>
<td>CAL0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **bit 7-2:** CAL5:CAL0: 6-bit Signed Oscillator Calibration bits
  - 111111 = Maximum frequency
  - 100000 = Center frequency
  - 000000 = Minimum frequency
- **bit 1-0:** Unimplemented: Read as '0'

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

The program counter (PC) is 13-bits wide. The low byte comes from the PCL register, which is a readable and writable register. The high byte (PC<12:8>) is not directly readable or writable and comes from PCLATH. On any RESET, the PC is cleared. Figure 2-3 shows the two situations for the loading of the PC. The upper example in Figure 2-3 shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in Figure 2-3 shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

FIGURE 2-3: LOADING OF PC IN DIFFERENT SITUATIONS

2.3.1 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL). When performing a table read using a computed GOTO method, care should be exercised if the table location crosses a PCL memory boundary (each 256-byte block). Refer to the Application Note “Implementing a Table Read” (AN556).

2.3.2 STACK

The PIC16F630/676 family has an 8 level x 13-bit wide hardware stack (see Figure 2-1). 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 affected by a PUSH or POP operation.

The stack operates as a circular buffer. This means that after the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

Note 1: There are no STATUS bits to indicate stack overflow or stack underflow conditions.

2: There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, RETURN, RETLW and RETFIE instructions or the vectoring to an interrupt address.
2.4 Indirect Addressing, INDF and FSR Registers

The INDF register is not a physical register. Addressing the INDF register will cause indirect addressing.

Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no operation (although STATUS bits may be affected). 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.

EXAMPLE 2-1: INDIRECT ADDRESSING

A simple program to clear RAM location 20h-2Fh using indirect addressing is shown in Example 2-1.

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

FIGURE 2-4: DIRECT/INDIRECT ADDRESSING PIC16F630/676

For memory map detail see Figure 2-2.

**Note 1:** The RP1 and IRP bits are reserved; always maintain these bits clear.
3.0 PORTS A AND C

There are as many as twelve general purpose I/O pins available. Depending on which peripherals are enabled, some or all of the pins may not be available as general purpose I/O. In general, when a peripheral is enabled, the associated pin may not be used as a general purpose I/O pin.

3.1 PORTA and the TRISA Registers

PORTA is an 6-bit wide, bi-directional port. The corresponding data direction register is TRISA. Setting a TRISA bit (= 1) will make the corresponding PORTA pin an input (i.e., put the corresponding output driver in a Hi-impedance mode). Clearing a TRISA bit (= 0) will make the corresponding PORTA pin an output (i.e., put the contents of the output latch on the selected pin). The exception is RA3, which is input only and its TRIS bit will always read as ‘1’. Example 3-1 shows how to initialize PORTA.

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, this value is modified and then written to the port data latch. RA3 reads ‘0’ when MCLREN = 1.

The TRISA register controls the direction of the PORTA 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. I/O pins configured as analog input always read ‘0’.

Note: Ansel bits must be clear for digital input, otherwise I/O pins which can be analog inputs read ‘0’.

Example 3-1: Initializing PORTA

```
bcf STATUS,RP0 ;Bank 0
clrf PORTA ;Init PORTA
movlw 05h ;Set RA<2:0> to
cmov CMCON ;digital I/O
bsf STATUS,RP0 ;Bank 1
clr ANSEL ;digital I/O
movlw 0Ch ;Set RA<3:2> as inputs
cmov TRISA ;and set RA<5:4,1:0>
bcf STATUS,RP0 ;Bank 0
```

3.1.1 ADDITIONAL PIN FUNCTIONS

Every PORTA pin on the PIC16F630/676 has an interrupt-on-change option and every PORTA pin, except RA3, has a weak pull-up option. The next two sections describe these functions.

3.1.1.1 Weak Pull-up

Each of the PORTA pins, except RA3, has an individually configurable weak internal pull-up. Control bits WPUAx enable or disable each pull-up. Refer to Register 3-3. Each 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 by the RAPU bit (OPTION<7>).

---

**Register 3-1: PORTA — PORTA REGISTER (ADDRESS: 05h)**

<table>
<thead>
<tr>
<th>bit 7-6:</th>
<th>bit 5-0:</th>
<th>bit 4-0:</th>
</tr>
</thead>
<tbody>
<tr>
<td>__ __</td>
<td>RA5</td>
<td>RA4</td>
</tr>
<tr>
<td>RA3</td>
<td>RA2</td>
<td>RA1</td>
</tr>
<tr>
<td>RA0</td>
<td>bit 0</td>
<td>bit 0</td>
</tr>
</tbody>
</table>

**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

---

© 2002 Microchip Technology Inc. Preliminary DS40039A-page 19
### REGISTER 3-2: TRISA — PORTA TRISTATE REGISTER (ADDRESS: 85h)

<table>
<thead>
<tr>
<th></th>
<th>U-0</th>
<th>U-0</th>
<th>R/W-x</th>
<th>R/W-x</th>
<th>R-1</th>
<th>R/W-x</th>
<th>R/W-x</th>
<th>R/W-x</th>
</tr>
</thead>
<tbody>
<tr>
<td>bit 7-6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 5-0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Unimplemented: Read as '0'*

*Note: TRIS<3> always reads 1.*

### REGISTER 3-3: WPUA — WEAK PULL-UP REGISTER (ADDRESS: 95h)

<table>
<thead>
<tr>
<th></th>
<th>U-0</th>
<th>U-0</th>
<th>R/W-1</th>
<th>R/W-1</th>
<th>U-0</th>
<th>R/W-1</th>
<th>R/W-1</th>
<th>R/W-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>bit 7-6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 5-4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bit 2-0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Unimplemented: Read as '0'*

*Note 1: Global RAPU must be enabled for individual pull-ups to be enabled.*

*Note 2: The weak pull-up device is automatically disabled if the pin is in Output mode (TRISA = 0).*
3.1.1.2 Interrupt-on-change

Each of the PORTA pins is individually configurable as an interrupt-on-change pin. Control bits IOCAx enable or disable the interrupt function for each pin. Refer to Register 3-4. The interrupt-on-change is disabled on a Power-on Reset.

For enabled interrupt-on-change pins, the values are compared with the old value latched on the last read of PORTA. The ‘mismatch’ outputs of the last read are OR'd together to set, or clear, the PORTA Change Interrupt flag bit (RAIF) in the INTCON register.

For enabled interrupt-on-change pins, the values are compared with the old value latched on the last read of PORTA. The ‘mismatch’ outputs of the last read are OR'd together to set, or clear, the PORTA Change Interrupt flag bit (RAIF) in the INTCON register.

This interrupt can wake the device from SLEEP. The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

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

b) Clear the flag bit RAIF.

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

Note: If a change on the I/O pin should occur when the read operation is being executed (start of the Q2 cycle), then the RAIF interrupt flag may not get set.

REGISTER 3-4: IOCA — INTERRUPT-ON-CHANGE PORTA REGISTER (ADDRESS: 96h)

<table>
<thead>
<tr>
<th>U-0</th>
<th>U-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
</tr>
</thead>
<tbody>
<tr>
<td>—</td>
<td>—</td>
<td>IOCA5</td>
<td>IOCA4</td>
<td>IOCA3</td>
<td>IOCA2</td>
<td>IOCA1</td>
<td>IOCA0</td>
</tr>
</tbody>
</table>

bit 7-6  
Unimplemented: Read as ‘0’

bit 5-0  
IOCA<5:0>: Interrupt-on-Change PORTA Control bit

1 = Interrupt-on-change enabled
0 = Interrupt-on-change disabled

Note: Global interrupt enables (GIE and RAIE) must be enabled for individual interrupts to be recognized.

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
3.1.2 PIN DESCRIPTIONS AND DIAGRAMS

Each PORTA pin is multiplexed with other functions. The pins and their combined functions are briefly described here. For specific information about individual functions such as the comparator or the A/D, refer to the appropriate section in this Data Sheet.

3.1.2.1 RA0/AN0/CIN+

Figure 3-1 shows the diagram for this pin. The RA0 pin is configurable to function as one of the following:
- a general purpose I/O
- an analog input for the A/D (PIC16F676 only)
- an analog input to the comparator

3.1.2.2 RA1/AN1/CIN-/VREF

Figure 3-1 shows the diagram for this pin. The RA1 pin is configurable to function as one of the following:
- as a general purpose I/O
- an analog input for the A/D (PIC16F676 only)
- an analog input to the comparator
- a voltage reference input for the A/D (PIC16F676 only)
3.1.2.3 RA2/AN2/T0CKI/INT/COUT
Figure 3-2 shows the diagram for this pin. The RA2 pin is configurable to function as one of the following:
- a general purpose I/O
- an analog input for the A/D (PIC16F676 only)
- a digital output from the comparator
- the clock input for TMR0
- an external edge triggered interrupt

**FIGURE 3-2: BLOCK DIAGRAM OF RA2**

3.1.2.4 RA3/MCLR/VPP
Figure 3-3 shows the diagram for this pin. The RA3 pin is configurable to function as one of the following:
- a general purpose input
- as Master Clear Reset

**FIGURE 3-3: BLOCK DIAGRAM OF RA3**
3.1.2.5 RA4/AN3/T1G/OSC2/CLKOUT

Figure 3-4 shows the diagram for this pin. The RA4 pin is configurable to function as one of the following:

- a general purpose I/O
- an analog input for the A/D (PIC16F676 only)
- a TMR1 gate input
- a crystal/resonator connection
- a clock output

FIGURE 3-4: BLOCK DIAGRAM OF RA4

3.1.2.6 RA5/T1CKI/OSC1/CLKIN

Figure 3-5 shows the diagram for this pin. The RA5 pin is configurable to function as one of the following:

- a general purpose I/O
- a TMR1 clock input
- a crystal/resonator connection
- a clock input

FIGURE 3-5: BLOCK DIAGRAM OF RA5

Note 1: CLK modes are XT, HS, LP, LPTMR1 and CLKOUT Enable.
2: With CLKOUT option.

Note 1: When using Timer1 with LP oscillator, the Schmitt Trigger is by-passed.
<table>
<thead>
<tr>
<th>Address</th>
<th>Name</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>Value on: POR</th>
<th>Value on all other RESETS</th>
</tr>
</thead>
<tbody>
<tr>
<td>05h</td>
<td>PORTA</td>
<td>—</td>
<td>—</td>
<td>RA5</td>
<td>RA4</td>
<td>RA3</td>
<td>RA2</td>
<td>RA1</td>
<td>RA0</td>
<td>--.xx xxxx</td>
<td>--uu uuuu</td>
</tr>
<tr>
<td>0Bh/8Bh</td>
<td>INTCN</td>
<td>GIE</td>
<td>PEIE</td>
<td>T0IE</td>
<td>INTE</td>
<td>RAIE</td>
<td>T0IF</td>
<td>INTF</td>
<td>RAIF</td>
<td>0000 0000</td>
<td>0000 0000u</td>
</tr>
<tr>
<td>19h</td>
<td>CMCON</td>
<td>—</td>
<td>COUT</td>
<td>—</td>
<td>CINV</td>
<td>CIS</td>
<td>CM2</td>
<td>CM1</td>
<td>CM0</td>
<td>-0-0 0000</td>
<td>-0-0 0000</td>
</tr>
<tr>
<td>81h</td>
<td>OPTION_REG</td>
<td>RAPU</td>
<td>INTEDG</td>
<td>T0CS</td>
<td>T0SE</td>
<td>PSA</td>
<td>PS2</td>
<td>PS1</td>
<td>PS0</td>
<td>1111 1111</td>
<td>1111 1111</td>
</tr>
<tr>
<td>85h</td>
<td>TRISA</td>
<td>—</td>
<td>—</td>
<td>TRISA5</td>
<td>TRISA4</td>
<td>TRISA3</td>
<td>TRISA2</td>
<td>TRISA1</td>
<td>TRISA0</td>
<td>--11 1111</td>
<td>--11 1111</td>
</tr>
<tr>
<td>91h</td>
<td>ANSEL(1)</td>
<td>ANS7</td>
<td>ANS6</td>
<td>ANS5</td>
<td>ANS4</td>
<td>ANS3</td>
<td>ANS2</td>
<td>ANS1</td>
<td>ANS0</td>
<td>ANS7</td>
<td>ANS7</td>
</tr>
<tr>
<td>95h</td>
<td>WPUA</td>
<td>—</td>
<td>—</td>
<td>WPUAS</td>
<td>WPUA4</td>
<td>—</td>
<td>WPUA2</td>
<td>WPUA1</td>
<td>WPUA0</td>
<td>--11 -1111</td>
<td>--11 -1111</td>
</tr>
<tr>
<td>96h</td>
<td>IOCA</td>
<td>—</td>
<td>—</td>
<td>IOCA5</td>
<td>IOCA4</td>
<td>IOCA3</td>
<td>IOCA2</td>
<td>IOCA1</td>
<td>IOCA0</td>
<td>--00 0000</td>
<td>--00 0000</td>
</tr>
</tbody>
</table>

**Note 1:** PIC16F676 only.

Legend:  
\(x\) = unknown, \(u\) = unchanged, \(-\) = unimplemented locations read as '0'. Shaded cells are not used by PORTA.
3.2 PORTC

Port C is a general purpose I/O port consisting of 6 bidirectional pins. The pins can be configured for either digital I/O or analog input to A/D converter. For specific information about individual functions such as the comparator or the A/D, refer to the appropriate section in this Data Sheet.

**Note:** Ansel bits must be clear for digital input, otherwise I/O pins which can be analog inputs read ‘0’.

### EXAMPLE 3-2: INITIALIZING PORTC

```asm
bcf STATUS, RP0 ; Bank 0
clrf PORTC ; Init PORTC
bsf STATUS, RP0 ; Bank 1
clrf ANSEL ; digital I/O
movlw 0Ch ; Set RC<3:2> as inputs
movwf TRISC ; and RC<5:4,1:0> as outputs
bcf STATUS, RP0 ; Bank 0
```

3.2.1 RC0/AN4, RC1/AN5, RC2/AN6, RC3/AN7

The RC0/RC1/RC2/RC3 pins are configurable to function as one of the following:
- a general purpose I/O
- an analog input for the A/D Converter
(PIC16F676 only)

### FIGURE 3-6:  Block Diagram of RC0/RC1/RC2/RC3 Pins

3.2.2 RC4 AND RC5

The RC4 and RC5 pins are configurable to function as a general purpose I/Os.

### FIGURE 3-7:  Block Diagram of RC4 and RC5 Pins
### REGISTER 3-5: PORTC — PORTC REGISTER (ADDRESS: 07h)

<table>
<thead>
<tr>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>RC5</td>
<td>RC4</td>
<td>RC3</td>
<td>RC2</td>
<td>RC1</td>
<td>RC0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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

**bit 5-0:** PORTC<5:0>: General Purpose I/O pin

- 1 = Port pin is >V_{IH}
- 0 = Port pin is <V_{IL}

### REGISTER 3-6: TRISC — PORTC TRISTATE REGISTER (ADDRESS: 87h)

<table>
<thead>
<tr>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRISC5</td>
<td>TRISC4</td>
<td>TRISC3</td>
<td>TRISC2</td>
<td>TRISC1</td>
<td>TRISC0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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

**bit 5-0:** TRISC<5:0>: Port C Tri-State Control bit

- 1 = Port C pin configured as an input (tri-stated)
- 0 = Port C pin configured as an output

### TABLE 3-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTC

<table>
<thead>
<tr>
<th>Address</th>
<th>Name</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>Value on POR</th>
<th>Value on all other RESETS</th>
</tr>
</thead>
<tbody>
<tr>
<td>07h</td>
<td>PORTC</td>
<td></td>
<td></td>
<td>RC5</td>
<td>RC4</td>
<td>RC3</td>
<td>RC2</td>
<td>RC1</td>
<td>RC0</td>
<td>---xx xxxx</td>
<td>---uu uuuu</td>
</tr>
<tr>
<td>87h</td>
<td>TRISC</td>
<td></td>
<td></td>
<td>TRISC5</td>
<td>TRISC4</td>
<td>TRISC3</td>
<td>TRISC2</td>
<td>TRISC1</td>
<td>TRISC0</td>
<td>--11 1111</td>
<td>--11 1111</td>
</tr>
<tr>
<td>91h</td>
<td>ANSEL(1)</td>
<td>ANS7</td>
<td>ANS6</td>
<td>ANS5</td>
<td>ANS4</td>
<td>ANS3</td>
<td>ANS2</td>
<td>ANS1</td>
<td>ANS0</td>
<td>1111 1111</td>
<td>1111 1111</td>
</tr>
</tbody>
</table>

**Note:** PIC16F676 only.

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by PORTC.
4.0 TIMER0 MODULE

The Timer0 module timer/counter has the following features:
- 8-bit timer/counter
- Readable and writable
- 8-bit software programmable prescaler
- Internal or external clock select
- Interrupt on overflow from FFh to 00h
- Edge select for external clock

Figure 4-1 is a block diagram of the Timer0 module and the prescaler shared with the WDT.

Note: Additional information on the Timer0 module is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

4.1 Timer0 Operation

Timer mode is selected by clearing the T0CS bit (OPTION_REG<5>). In Timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If TMR0 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 the T0CS bit (OPTION_REG<5>). In this mode, the Timer0 module will increment either on every rising or falling edge of pin RA2/T0CKI. The incrementing edge is determined by the source edge (T0SE) control bit (OPTION_REG<4>). Clearing the T0SE bit selects the rising edge.

Note: Counter mode has specific external clock requirements. Additional information on these requirements is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

4.2 Timer0 Interrupt

A Timer0 interrupt is generated when the TMR0 register timer/counter overflows from FFh to 00h. This overflow sets the T0IF bit. The interrupt can be masked by clearing the T0IE bit (INTCON<5>). The T0IF bit (INTCON<2>) must be cleared in software by the Timer0 module Interrupt Service Routine before reenabling this interrupt. The Timer0 interrupt cannot wake the processor from SLEEP since the timer is shut-off during SLEEP.

Figure 4-1: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER

Note 1: T0SE, T0CS, PSA, PS0-PS2 are bits in the Option register.
4.3 Using Timer0 with an External Clock

When no prescaler is used, the external clock input is the same as the prescaler output. The synchronization of T0CKI, with the internal phase clocks, is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks. Therefore, it is necessary for T0CKI to be high for at least 2Tosc (and a small RC delay of 20 ns) and low for at least 2Tosc (and a small RC delay of 20 ns). Refer to the electrical specification of the desired device.

Note: The ANSEL register (91h) must be ‘0’ to operate an analog channel as a digital input. The ANSEL register is defined for the PIC16F676 only.

REGISTER 4-1: OPTION_REG — OPTION REGISTER (ADDRESS: 81h)

<table>
<thead>
<tr>
<th>R/W-1</th>
<th>R/W-1</th>
<th>R/W-1</th>
<th>R/W-1</th>
<th>R/W-1</th>
<th>R/W-1</th>
<th>R/W-1</th>
<th>R/W-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAPU</td>
<td>INTEDG</td>
<td>T0CS</td>
<td>T0SE</td>
<td>PSA</td>
<td>PS2</td>
<td>PS1</td>
<td>PS0</td>
</tr>
</tbody>
</table>

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

bit 6  INTEDG: Interrupt Edge Select bit
       1 = Interrupt on rising edge of RA2/INT pin
       0 = Interrupt on falling edge of RA2/INT pin

bit 5  T0CS: TMR0 Clock Source Select bit
       1 = Transition on RA2/T0CKI pin
       0 = Internal instruction cycle clock (CLKOUT)

bit 4  T0SE: TMR0 Source Edge Select bit
       1 = Increment on high-to-low transition on RA2/T0CKI pin
       0 = Increment on low-to-high transition on RA2/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

<table>
<thead>
<tr>
<th>Bit Value</th>
<th>TMR0 Rate</th>
<th>WDT Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>1:2</td>
<td>1:1</td>
</tr>
<tr>
<td>001</td>
<td>1:4</td>
<td>1:2</td>
</tr>
<tr>
<td>010</td>
<td>1:8</td>
<td>1:4</td>
</tr>
<tr>
<td>011</td>
<td>1:16</td>
<td>1:8</td>
</tr>
<tr>
<td>100</td>
<td>1:32</td>
<td>1:16</td>
</tr>
<tr>
<td>101</td>
<td>1:64</td>
<td>1:32</td>
</tr>
<tr>
<td>110</td>
<td>1:128</td>
<td>1:64</td>
</tr>
<tr>
<td>111</td>
<td>1:256</td>
<td>1:128</td>
</tr>
</tbody>
</table>

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
4.4 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module, or as a postscaler for the Watchdog Timer. For simplicity, this counter will be referred to as “prescaler” throughout this Data Sheet. The prescaler assignment is controlled in software by the control bit PSA (OPTION_REG<3>). Clearing the PSA bit will assign the prescaler to Timer0. Prescale values are selectable via the PS2:PS0 bits (OPTION_REG<2:0>.

The prescaler is not readable or writable. When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g., CLRF 1, MOVWF 1, BSF 1, x....etc.) will clear the prescaler. When assigned to WDT, a CLRWDT instruction will clear the prescaler along with the Watchdog Timer.

4.4.1 SWITCHING PRESCALER ASSIGNMENT

The prescaler assignment is fully under software control (i.e., it can be changed “on the fly” during program execution). To avoid an unintended device RESET, the following instruction sequence (Example 4-1) must be executed when changing the prescaler assignment from Timer0 to WDT.

EXAMPLE 4-1: CHANGING PRESCALER (TIMER0→WDT)

```
bcf STATUS,RP0 ;Bank 0
clrwdt ;Clear WDT
clr TMR0 ;Clear TMR0 and
bsf STATUS,RP0 ;Bank 1
preset
movlw b'00101111' ;Required if desired
movwf OPTION_REG ; PS2:PS0 is
clrwdt ; 000 or 001
movlw b'00101xxx' ;Set postscaler to
movwf OPTION_REG ; desired WDT rate
bcf STATUS,RP0 ;Bank 0
```

To change prescaler from the WDT to the TMR0 module, use the sequence shown in Example 4-2. This precaution must be taken even if the WDT is disabled.

EXAMPLE 4-2: CHANGING PRESCALER (WDT→TIMER0)

```
clrwdt ;Clear WDT and
bsf STATUS,RP0 ;Bank 1
movlw b'xxxx0xxx' ;Select TMR0,
movwf OPTION_REG ; prescale, and
bcf STATUS,RP0 ;Bank 0
```

<table>
<thead>
<tr>
<th>Address</th>
<th>Name</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>Value on POR</th>
<th>Value on all other RESETS</th>
</tr>
</thead>
<tbody>
<tr>
<td>01h</td>
<td>TMR0</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td>xxxx</td>
<td></td>
<td></td>
<td>xxxxxxxxxx</td>
<td>uuuu uuuu</td>
</tr>
<tr>
<td>0Bh/8Bh</td>
<td>INTCN</td>
<td>GIE</td>
<td>PEIE</td>
<td>TOIE</td>
<td>INTE</td>
<td>RAIE</td>
<td>T0IF</td>
<td>INTF</td>
<td>RAIF</td>
<td>0000 0000</td>
<td>0000 0000</td>
</tr>
<tr>
<td>81h</td>
<td>OPTION_REG</td>
<td>RAPU</td>
<td>INTEG</td>
<td>T0CS</td>
<td>T0SE</td>
<td>PSA</td>
<td>PS2</td>
<td>PS1</td>
<td>PS0</td>
<td>1111 1111</td>
<td>1111 1111</td>
</tr>
<tr>
<td>85h</td>
<td>TRISA</td>
<td>—</td>
<td>—</td>
<td>TRISA5</td>
<td>TRISA4</td>
<td>TRISA3</td>
<td>TRISA2</td>
<td>TRISA1</td>
<td>TRISA0</td>
<td>--11 1111</td>
<td>--11 1111</td>
</tr>
</tbody>
</table>

Legend: — = Unimplemented locations, read as ‘0’, u = unchanged, x = unknown. Shaded cells are not used by the Timer0 module.
5.0 TIMER1 MODULE WITH GATE CONTROL

The PIC16F630/676 devices have a 16-bit timer. Figure 5-1 shows the basic block diagram of the Timer1 module. Timer1 has the following features:

- 16-bit timer/counter (TMR1H:TMR1L)
- Readable and writable
- Internal or external clock selection
- Synchronous or asynchronous operation
- Interrupt on overflow from FFFFh to 0000h
- Wake-up upon overflow (Asynchronous mode)
- Optional external enable input (T1G)
- Optional LP oscillator

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

Figure 5-1: TIMER1 BLOCK DIAGRAM

The Timer1 Control register (T1CON), shown in Register 5-1, is used to enable/disable Timer1 and select the various features of the Timer1 module.
5.1 Timer1 Modes of Operation

Timer1 can operate in one of three modes:
• 16-bit timer with prescaler
• 16-bit synchronous counter
• 16-bit asynchronous counter

In Timer mode, Timer1 is incremented on every instruction cycle. In Counter mode, Timer1 is incremented on the rising edge of the external clock input T1CKI. In addition, the Counter mode clock can be synchronized to the microcontroller system clock or run asynchronously.

In Counter and Timer modules, the counter/timer clock can be gated by the T1G input.

If an external clock oscillator is needed (and the microcontroller is using the INTOSC w/o CLKOUT), Timer1 can use the LP oscillator as a clock source.

Note: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge.

5.2 Timer1 Interrupt

The Timer1 register pair (TMR1H:TMR1L) increments to FFFFh and rolls over to 0000h. When Timer1 rolls over, the Timer1 interrupt flag bit (PIR1<0>) is set. To enable the interrupt on rollover, you must set these bits:
• Timer1 interrupt Enable bit (PIE1<0>)
• PEIE bit (INTCON<6>)
• GIE bit (INTCON<7>).

The interrupt is cleared by clearing the TMR1IF in the Interrupt Service Routine.

Note: The TMR1H:TMR1L register pair and the TMR1IF bit should be cleared before enabling interrupts.

5.3 Timer1 Prescaler

Timer1 has four prescaler options allowing 1, 2, 4, or 8 divisions of the clock input. The T1CKPS bits (T1CON<5:4>) control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMR1H or TMR1L.

FIGURE 5-2: TIMER1 INCREMENTING EDGE

T1CKI = 1 when TMR1 Enabled

T1CKI = 0 when TMR1 Enabled

Note: Arrows indicate counter increments.
1: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge of the clock.
## REGISTER 5-1: T1CON — TIMER1 CONTROL REGISTER (ADDRESS: 10h)

<table>
<thead>
<tr>
<th>U-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>TMR1GE</td>
<td>T1CKPS1</td>
<td>T1CKPS0</td>
<td>T1OSCEN</td>
<td>T1SYNC</td>
<td>TMR1CS</td>
<td>TMR1ON</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>bit 7</th>
<th>bit 0</th>
</tr>
</thead>
</table>

- **bit 7 Unimplemented**: Read as ‘0’
- **bit 6 TMR1GE**: Timer1 Gate Enable bit
  - If TMR1ON = 0: This bit is ignored
  - If TMR1ON = 1: 1 = Timer1 is on if T1G pin is low
  - 0 = Timer1 is on

- **bit 5-4 T1CKPS1:T1CKPS0**: Timer1 Input Clock Prescale Select bits
  - 11 = 1:8 Prescale Value
  - 10 = 1:4 Prescale Value
  - 01 = 1:2 Prescale Value
  - 00 = 1:1 Prescale Value

- **bit 3 T1OSCEN**: LP Oscillator Enable Control bit
  - If INTOSC without CLKOUT oscillator is active:
    - 1 = LP oscillator is enabled for Timer1 clock
    - 0 = LP oscillator is off
  - Else: This bit is ignored

- **bit 2 T1SYNC**: Timer1 External Clock Input Synchronization Control bit
  - TMR1CS = 1:
    - 1 = Do not synchronize external clock input
    - 0 = Synchronize external clock input
  - TMR1CS = 0:
    - This bit is ignored. Timer1 uses the internal clock.

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

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

**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
5.4 **Timer1 Operation in Asynchronous Counter Mode**

If control bit T1SYNC (T1CON<2>) is set, the external clock input is not synchronized. The timer continues to increment asynchronous to the internal phase clocks. The timer will continue to run during SLEEP and can generate an interrupt on overflow, which will wake-up the processor. However, special precautions in software are needed to read/write the timer (Section 5.4.1).

5.4.1 **READING AND WRITING TIMER1 IN ASYNCHRONOUS COUNTER MODE**

Reading TMR1H or TMR1L, while the timer is running from an external asynchronous clock, will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself, poses certain problems, since the timer may overflow between the reads.

For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers, while the register is incrementing. This may produce an unpredictable value in the timer register.

Reading the 16-bit value requires some care. Examples 12-2 and 12-3 in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023) show how to read and write Timer1 when it is running in Asynchronous mode.

5.5 **Timer1 Oscillator**

A crystal oscillator circuit is built-in between pins OSC1 (input) and OSC2 (amplifier output). It is enabled by setting control bit T1OSCEN (T1CON<3>). The oscillator is a low power oscillator rated up to 32 kHz. It will continue to run during SLEEP. It is primarily intended for a 32 kHz crystal. Table 5-1 shows the capacitor selection for the Timer1 oscillator.

5.6 **Timer1 Operation During SLEEP**

Timer1 can only operate during SLEEP when setup in Asynchronous Counter mode. In this mode, an external crystal or clock source can be used to increment the counter. To setup the timer to wake the device:

- Timer1 must be on (T1CON<0>)
- TMR1IE bit (PIE1<0>) must be set
- PEIE bit (INTCON<6>) must be set

The device will wake-up on an overflow. If the GIE bit (INTCON<7>) is set, the device will wake-up and jump to the Interrupt Service Routine on an overflow.

### TABLE 5-1: CAPACITOR SELECTION FOR THE TIMER1 OSCILLATOR

<table>
<thead>
<tr>
<th>Osc Type</th>
<th>Freq</th>
<th>C1</th>
<th>C2</th>
</tr>
</thead>
<tbody>
<tr>
<td>LP</td>
<td>32 kHz</td>
<td>15 pF</td>
<td>15 pF</td>
</tr>
</tbody>
</table>

These values are for design guidance only.

**Note:**
1. Higher capacitance increases the stability of oscillator but also increases the start-up time.
2. Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components.

### TABLE 5-2: REGISTERS ASSOCIATED WITH TIMER1 AS A TIMER/COUNTER

<table>
<thead>
<tr>
<th>Address</th>
<th>Name</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>Value on POR</th>
<th>Value on all other Resets</th>
</tr>
</thead>
<tbody>
<tr>
<td>0Bh/8Bh</td>
<td>INTCON</td>
<td>GIE</td>
<td>PEIE</td>
<td>T0IE</td>
<td>INTE</td>
<td>RAIE</td>
<td>T0IF</td>
<td>INTF</td>
<td>RAIF</td>
<td>0000 0000</td>
<td>0000 0000</td>
</tr>
<tr>
<td>0Ch</td>
<td>PIE1</td>
<td>EEIE</td>
<td>ADIF</td>
<td>—</td>
<td>—</td>
<td>CMIF</td>
<td>—</td>
<td>—</td>
<td>TMR1F</td>
<td>00-- 0--0</td>
<td>00-- 0--0</td>
</tr>
<tr>
<td>0Eh</td>
<td>TMR1L</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>CMIF</td>
<td>—</td>
<td>—</td>
<td>TMR1F</td>
<td>xxxx xxxx</td>
<td>xxxxxx xxxxx</td>
</tr>
<tr>
<td>0Fh</td>
<td>TMR1H</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>CMIF</td>
<td>—</td>
<td>—</td>
<td>TMR1F</td>
<td>xxxx xxxx</td>
<td>xxxxxx xxxxx</td>
</tr>
<tr>
<td>10h</td>
<td>T1CON</td>
<td>—</td>
<td>TMR1GE</td>
<td>T1CKPS1</td>
<td>T1CKPS0</td>
<td>T1OSCEN</td>
<td>T1SYNC</td>
<td>TMR1CS</td>
<td>TMR1ON</td>
<td>-000 0000</td>
<td>-000 0000</td>
</tr>
<tr>
<td>8Ch</td>
<td>PIE1</td>
<td>EEIE</td>
<td>ADIE</td>
<td>—</td>
<td>—</td>
<td>CMIE</td>
<td>—</td>
<td>—</td>
<td>TMR1IE</td>
<td>00-- 0--0</td>
<td>00-- 0--0</td>
</tr>
</tbody>
</table>

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

The PIC16F630/676 devices have one analog comparator. The inputs to the comparator are multiplexed with the RA0 and RA1 pins. There is an on-chip Comparator Voltage Reference that can also be applied to an input of the comparator. In addition, RA2 can be configured as the comparator output. The Comparator Control Register (CMCON), shown in Register 6-1, contains the bits to control the comparator.

REGISTER 6-1: CMCON — COMPARATOR CONTROL REGISTER (ADDRESS: 19h)

<table>
<thead>
<tr>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2-0</th>
</tr>
</thead>
<tbody>
<tr>
<td>COUT</td>
<td>CINV</td>
<td>Unimplemented</td>
<td>CIS</td>
<td>CM2:CM0</td>
<td></td>
</tr>
</tbody>
</table>

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

bit 7
Unimplemented: Read as ‘0’

bit 6
COUT: Comparator Output bit
When CINV = 0:
1 = VIN+ > VIN-
0 = VIN+ < VIN-
When CINV = 1:
0 = VIN+ > VIN-
1 = VIN+ < VIN-

bit 5
Unimplemented: Read as ‘0’

bit 4
CINV: Comparator Output Inversion bit
1 = Output inverted
0 = Output not inverted

bit 3
CIS: Comparator Input Switch bit
When CM2:CM0 = 110 or 101:
1 = VIN- connects to CIN+
0 = VIN- connects to CIN-

bit 2-0
CM2:CM0: Comparator Mode bits
Figure 6-2 shows the Comparator modes and CM2:CM0 bit settings
6.1 Comparator Operation

A single comparator is shown in Figure 6-1, along with the relationship between the analog input levels and the digital output. When the analog input at VIN+ is less than the analog input VIN-, the output of the comparator is a digital low level. When the analog input at VIN+ is greater than the analog input VIN-, the output of the comparator is a digital high level. The shaded areas of the output of the comparator in Figure 6-1 represent the uncertainty due to input offsets and response time.

**Note:** To use AN<7:0> as analog inputs, the appropriate bits must be programmed in the ANSEL register. This register is available for PIC16F676 only.

The polarity of the comparator output can be inverted by setting the CINV bit (CMCON<4>). Clearing CINV results in a non-inverted output. A complete table showing the output state versus input conditions and the polarity bit is shown in Table 6-1.

<table>
<thead>
<tr>
<th>Input Conditions</th>
<th>CINV</th>
<th>COUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIN- &gt; VIN+</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VIN- &lt; VIN+</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>VIN- &gt; VIN+</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>VIN- &lt; VIN+</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

**Note:** CINV bit (CMCON<4>) is clear.
6.2 Comparator Configuration

There are eight modes of operation for the comparator. The CMCON register, shown in Register 6-1, is used to select the mode. Figure 6-2 shows the eight possible modes. The TRISA register controls the data direction of the comparator pins for each mode. If the Comparator mode is changed, the comparator output level may not be valid for a specified period of time. Refer to the specifications in Section 12.0.

**Note:** Comparator interrupts should be disabled during a Comparator mode change. Otherwise, a false interrupt may occur.

### FIGURE 6-2: COMPARATOR I/O OPERATING MODES

| Comparator Reset (POR Default Value - low power) | Comparator Off (Lowest power) |
| CM2:CM0 = 000 | CM2:CM0 = 111 |
| RA1/CIN- | D |
| RA0/CIN+ | D |
| RA2/COUT | D |

| Comparator without Output | Comparator w/o Output and with Internal Reference |
| CM2:CM0 = 010 | CM2:CM0 = 100 |
| RA1/CIN- | A |
| RA0/CIN+ | A |
| RA2/COUT | D |
| RA2/COUT | D |

| Comparator with Output and Internal Reference | Multiplexed Input with Internal Reference and Output |
| CM2:CM0 = 011 | CM2:CM0 = 101 |
| RA1/CIN- | A |
| RA0/CIN+ | D |
| RA2/COUT | D |
| RA2/COUT | D |

| Comparator with Output | Multiplexed Input with Internal Reference |
| CM2:CM0 = 001 | CM2:CM0 = 110 |
| RA1/CIN- | A |
| RA0/CIN+ | A |
| RA2/COUT | D |
| RA2/COUT | D |

| Comparator without Output | Comparator w/o Output and with Internal Reference |
| CM2:CM0 = 010 | CM2:CM0 = 100 |
| RA1/CIN- | A |
| RA0/CIN+ | A |
| RA2/COUT | D |
| RA2/COUT | D |

### Diagrams:

- Comparator Reset (POR Default Value - low power)
- Comparator Off (Lowest power)
- Comparator without Output
- Comparator w/o Output and with Internal Reference
- Comparator with Output and Internal Reference
- Multiplexed Input with Internal Reference and Output
- Comparator with Output
- Multiplexed Input with Internal Reference

**A = Analog Input, ports always reads ‘0’**

**D = Digital Input**

**CIS = Comparator Input Switch (CMCON<3>)**
6.3 Analog Input Connection Considerations

A simplified circuit for an analog input is shown in Figure 6-3. Since the analog pins are connected to a digital output, they have reverse biased diodes to VDD and VSS. The analog input, therefore, must be between VSS and VDD. If the input voltage deviates from this range by more than 0.6V in either direction, one of the diodes is forward biased and a latchup may occur. A maximum source impedance of $10 \, \text{k}\Omega$ is recommended for the analog sources. Any external component connected to an analog input pin, such as a capacitor or a Zener diode, should have very little leakage current.

**FIGURE 6-3: ANALOG INPUT MODE**

![Diagram showing the connection of an analog input with reverse biased diodes to VDD and VSS.]

Legend:  
- **CPIN** = Input Capacitance  
- **VT** = Threshold Voltage  
- **ILEAKAGE** = Leakage Current at the pin due to Various Junctions  
- **RIC** = Interconnect Resistance  
- **RS** = Source Impedance  
- **VA** = Analog Voltage

6.4 Comparator Output

The comparator output, COUT, is read through the CMCON register. This bit is read-only. The comparator output may also be directly output to the RA2 pin in three of the eight possible modes, as shown in Figure 6-2. When in one of these modes, the output on RA2 is asynchronous to the internal clock. Figure 6-4 shows the comparator output block diagram.

**FIGURE 6-4: MODIFIED COMPARATOR OUTPUT BLOCK DIAGRAM**

![Diagram showing the modified comparator output block diagram.]

The TRISA<2> bit functions as an output enable/disable for the RA2 pin while the comparator is in an Output mode.

**Note 1:** When reading the PORTA register, all pins configured as analog inputs will read as a '0'. Pins configured as digital inputs will convert an analog input according to the TTL input specification.

**Note 2:** Analog levels on any pin that is defined as a digital input, may cause the input buffer to consume more current than is specified.
6.5 Comparator Reference

The comparator module also allows the selection of an internally generated voltage reference for one of the comparator inputs. The internal reference signal is used for four of the eight Comparator modes. The VRCON register, Register 6-2, controls the voltage reference module shown in Figure 6-5.

6.5.1 CONFIGURING THE VOLTAGE REFERENCE

The voltage reference can output 32 distinct voltage levels, 16 in a high range and 16 in a low range.

The following equations determine the output voltages:

- \( V_{RR} = 1 \) (low range): \( CV_{REF} = \left( \frac{VR3:VR0}{24} \right) \times V_{DD} \)
- \( V_{RR} = 0 \) (high range): \( CV_{REF} = \left( \frac{V_{DD}}{4} \right) + \left( \frac{VR3:VR0 \times V_{DD}}{32} \right) \)

6.5.2 VOLTAGE REFERENCE ACCURACY/ERROR

The full range of VSS to VDD cannot be realized due to the construction of the module. The transistors on the top and bottom of the resistor ladder network (Figure 6-5) keep CVREF from approaching VSS or VDD. The Voltage Reference is VDD derived and therefore, the CVREF output changes with fluctuations in VDD. The tested absolute accuracy of the Comparator Voltage Reference can be found in Section 12.0.

FIGURE 6-5: COMPARATOR VOLTAGE REFERENCE BLOCK DIAGRAM

6.6 Comparator Response Time

Response time is the minimum time, after selecting a new reference voltage or input source, before the comparator output is ensured to have a valid level. If the internal reference is changed, the maximum delay of the internal voltage reference must be considered when using the comparator outputs. Otherwise, the maximum delay of the comparators should be used (Table 12-4).

6.7 Operation During SLEEP

Both the comparator and voltage reference, if enabled before entering SLEEP mode, remain active during SLEEP. This results in higher SLEEP currents than shown in the power-down specifications. The additional current consumed by the comparator and the voltage reference is shown separately in the specifications. To minimize power consumption while in SLEEP mode, turn off the comparator, CM2:CM0 = 111, and voltage reference, VRCON<7> = 0.

While the comparator is enabled during SLEEP, an interrupt will wake-up the device. If the device wakes up from SLEEP, the contents of the CMCON and VRCON registers are not affected.

6.8 Effects of a RESET

A device RESET forces the CMCON and VRCON registers to their RESET states. This forces the comparator module to be in the Comparator Reset mode, CM2:CM0 = 000 and the voltage reference to its off state. Thus, all potential inputs are analog inputs with the comparator and voltage reference disabled to consume the smallest current possible.
6.9 Comparator Interrupts

The comparator interrupt flag is set whenever there is a change in the output value of the comparator. Software will need to maintain information about the status of the output bits, as read from CMCON<6>, to determine the actual change that has occurred. The CMIF bit, PIR1<3>, is the comparator interrupt flag. This bit must be reset in software by clearing it to 0. Since it is also possible to write a '1' to this register, a simulated interrupt may be initiated.

The CMIE bit (PIE1<3>) and the PEIE bit (INTCON<6>) must be set to enable the interrupt. In addition, the GIE bit must also be set. If any of these bits are cleared, the interrupt is not enabled, though the CMIF bit will still be set if an interrupt condition occurs.

The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

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

b) Clear flag bit CMIF.

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

**Note:** If a change in the CMCON register (COUT) should occur when a read operation is being executed (start of the Q2 cycle), then the CMIF (PIR1<3>) interrupt flag may not get set.

### TABLE 6-2: REGISTERS ASSOCIATED WITH COMPARATOR MODULE

<table>
<thead>
<tr>
<th>Address</th>
<th>Name</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>Value on POR</th>
<th>Value on all other RESETS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0Bh</td>
<td>INTCON</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0000 0000</td>
<td>0000 000u</td>
</tr>
<tr>
<td>0Ch</td>
<td>PIR1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>00-0-0000</td>
<td>00-0-0-0000</td>
</tr>
<tr>
<td>19h</td>
<td>CMCON</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0-0-0000</td>
<td>-0-0-0-0000</td>
</tr>
<tr>
<td>8Ch</td>
<td>PIE1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>00-0-0-0000</td>
<td>00-0-0-0000</td>
</tr>
<tr>
<td>85h</td>
<td>TRISA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-11 1111</td>
<td>-11 1111</td>
</tr>
<tr>
<td>99h</td>
<td>VRCON</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0-0-0-0000</td>
<td>0-0-0-0000</td>
</tr>
</tbody>
</table>

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by the comparator module.
7.0 ANALOG-TO-DIGITAL CONVERTER (A/D) MODULE (PIC16F676 ONLY)

The analog-to-digital converter (A/D) allows conversion of an analog input signal to a 10-bit binary representation of that signal. The PIC16F676 has eight analog inputs, multiplexed into one sample and hold circuit.

The output of the sample and hold is connected to the input of the converter. The converter generates a binary result via successive approximation and stores the result in a 10-bit register. The voltage reference used in the conversion is software selectable to either VDD or a voltage applied by the VREF pin. Figure 7-1 shows the block diagram of the A/D on the PIC16F676.

7.1 A/D Configuration and Operation

There are two registers available to control the functionality of the A/D module:

1. ADCON0 (Register 7-1)
2. ADCON1 (Register 7-2)
3. ANSEL (Register 7-3)

7.1.1 ANALOG PORT PINS

The ANS7:ANS0 bits (ANSEL<7:0>) and the TRISA bits control the operation of the A/D port pins. Set the corresponding TRISA bits to set the pin output driver to its high impedance state. Likewise, set the corresponding ANS bit to disable the digital input buffer.

Note: Analog voltages on any pin that is defined as a digital input may cause the input buffer to conduct excess current.

7.1.2 CHANNEL SELECTION

There are eight analog channels on the PIC16F676, AN0 through AN7. The CHS2:CHS0 bits (ADCON0<4:2>) control which channel is connected to the sample and hold circuit.

7.1.3 VOLTAGE REFERENCE

There are two options for the voltage reference to the A/D converter: either VDD is used, or an analog voltage applied to VREF is used. The VCFG bit (ADCON0<6>) controls the voltage reference selection. If VCFG is set, then the voltage on the VREF pin is the reference; otherwise, VDD is the reference.

7.1.4 CONVERSION CLOCK

The A/D conversion cycle requires 11 TAD. The source of the conversion clock is software selectable via the ADCS bits (ADCON1<6:4>). There are seven possible clock options:

- \( \frac{F_{OSC}}{2} \)
- \( \frac{F_{OSC}}{4} \)
- \( \frac{F_{OSC}}{8} \)
- \( \frac{F_{OSC}}{16} \)
- \( \frac{F_{OSC}}{32} \)
- \( \frac{F_{OSC}}{64} \)
- FRC (dedicated internal oscillator)

For correct conversion, the A/D conversion clock (1/TAD) must be selected to ensure a minimum TAD of 1.6 \( \mu s \). Table 7-1 shows a few TAD calculations for selected frequencies.
### 7.1.5 STARTING A CONVERSION

The A/D conversion is initiated by setting the GO/DONE bit (ADCON0<1>). When the conversion is complete, the A/D module:

- Clears the GO/DONE bit
- Sets the ADIF flag (PIR1<6>)
- Generates an interrupt (if enabled).

If the conversion must be aborted, the GO/DONE bit can be cleared in software. The ADRESH:ADRESL registers will not be updated with the partially complete A/D conversion sample. Instead, the ADRESH:ADRESL registers will retain the value of the previous conversion. After an aborted conversion, a 2 TAD delay is required before another acquisition can be initiated. Following the delay, an input acquisition is automatically started on the selected channel.

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

### 7.1.6 CONVERSION OUTPUT

The A/D conversion can be supplied in two formats: left or right shifted. The ADFM bit (ADCON0<7>) controls the output format. Figure 7-2 shows the output formats.

---

### Figure 7-2: 10-BIT A/D RESULT FORMAT

<table>
<thead>
<tr>
<th>A/D Clock Source (TAD)</th>
<th>Device Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Operation</strong></td>
<td><strong>20 MHz</strong></td>
</tr>
<tr>
<td>2 Tosc</td>
<td>000</td>
</tr>
<tr>
<td>4 Tosc</td>
<td>010</td>
</tr>
<tr>
<td>8 Tosc</td>
<td>001</td>
</tr>
<tr>
<td>16 Tosc</td>
<td>011</td>
</tr>
<tr>
<td>32 Tosc</td>
<td>010</td>
</tr>
<tr>
<td>64 Tosc</td>
<td>110</td>
</tr>
<tr>
<td>A/D RC</td>
<td>x11</td>
</tr>
</tbody>
</table>

Legend: Shaded cells are outside of recommended range.

**Note 1:** The A/D RC source has a typical TAD time of 4 µs for VDD > 3.0V.

**Note 2:** These values violate the minimum required TAD time.

**Note 3:** For faster conversion times, the selection of another clock source is recommended.

**Note 4:** When the device frequency is greater than 1 MHz, the A/D RC clock source is only recommended if the conversion will be performed during SLEEP.

---

DS40039A-page 44  Preliminary  © 2002 Microchip Technology Inc.
### REGISTER 7-1: ADCON0 — A/D CONTROL REGISTER (ADDRESS: 1Fh)

<table>
<thead>
<tr>
<th>R/W-0</th>
<th>R/W-0</th>
<th>U-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADFM</td>
<td>VCFG</td>
<td>—</td>
<td>CHS2</td>
<td>CHS1</td>
<td>CHS0</td>
<td>GO/DONE</td>
<td>ADON</td>
</tr>
</tbody>
</table>

- **bit 7** ADFM: A/D Result Formed Select bit
  - 1 = Right justified
  - 0 = Left justified

- **bit 6** VCFG: Voltage Reference bit
  - 1 = VREF pin
  - 0 = VDD

- **bit 5** Unimplemented: Read as zero

- **bits 4-2** CHS2:CHS0: Analog Channel Select bits
  - 000 = Channel 00 (AN0)
  - 001 = Channel 01 (AN1)
  - 010 = Channel 02 (AN2)
  - 011 = Channel 03 (AN3)
  - 100 = Channel 04 (AN4)
  - 101 = Channel 05 (AN5)
  - 110 = Channel 06 (AN6)
  - 111 = Channel 07 (AN7)

- **bit 1** GO/DONE: A/D Conversion Status bit
  - 1 = A/D conversion cycle in progress. Setting this bit starts an A/D conversion cycle.
  - This bit is automatically cleared by hardware when the A/D conversion has completed.
  - 0 = A/D conversion completed/not in progress

- **bit 0** ADON: A/D Conversion Status bit
  - 1 = A/D converter module is operating
  - 0 = A/D converter is shut-off 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

### REGISTER 7-2: ADCON1 — A/D CONTROL REGISTER 1 (ADDRESS: 9Fh)

<table>
<thead>
<tr>
<th>U-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>U-0</th>
<th>U-0</th>
<th>U-0</th>
<th>U-0</th>
</tr>
</thead>
<tbody>
<tr>
<td>—</td>
<td>ADCS2</td>
<td>ADCS1</td>
<td>ADCS0</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

- **bit 7**: Unimplemented: Read as ‘0’.

- **bits 6-4**: ADCS<2:0>: A/D Conversion Clock Select bits
  - 000 = Fosc/2
  - 001 = Fosc/8
  - 010 = Fosc/32
  - x11 = Frc (clock derived from a dedicated internal oscillator = 500 kHz max)
  - 100 = Fosc/4
  - 101 = Fosc/16
  - 110 = Fosc/64

- **bit 3-0**: Unimplemented: Read as ‘0’.

#### Legend:
- R = Readable bit
- W = Writable bit
- U = Unimplemented bit, read as ‘0’
- n = Value at POR
- ‘1’ = Bit is set
- ‘0’ = Bit is cleared
- x = Bit is unknown
REGISTER 7-3:  ANSEL — ANALOG SELECT REGISTER (ADDRESS: 91h) (PIC16F676 ONLY)

R/W-1  R/W-1  R/W-1  R/W-1  R/W-1  R/W-1  R/W-1  R/W-1
ANS7  ANS6  ANS5  ANS4  ANS3  ANS2  ANS1  ANS0

bit 7-0:  ANS<7:0>: Analog Select between analog or digital function on pins AN<7:0>, respectively.
0 = Digital I/O. Pin is assigned to port or special function.
1 = Analog input. Pin is assigned as analog input.\(^{(1)}\)

Note 1: Setting a pin to an analog input automatically disables the digital input circuitry,
weak pull-ups, and interrupt-on-change if available. The corresponding TRIS bit
must be set to Input mode in order to allow external control of the voltage on the pin.

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
7.2 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 7-3. The source impedance (RS) and the internal sampling switch (RSS) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch (RSS) impedance varies over the device voltage (VDD), see Figure 7-3. **The maximum recommended impedance for analog sources is 10 kΩ.** As the impedance is decreased, the acquisition time may be decreased.

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, Equation 7-1 may be used. This equation assumes that 1/2 LSb error is used (1024 steps for the A/D). The 1/2 LSb error is the maximum error allowed for the A/D to meet its specified resolution.

To calculate the minimum acquisition time, TACQ, see the PICmicro™ Mid-Range Reference Manual (DS33023).

**EQUATION 7-1: ACQUISITION TIME**

\[
T_{ACQ} = T_{AMP} + T_{C} + T_{COFF} + 2 \mu s + T_C + \left(\frac{\text{Temperature} - 25^\circ C}{0.05 \mu s/\circ C}\right)
\]

\[
T_{CHOLD} = \left(\frac{1}{2047}\right) \left(\frac{R_I + R_S + R_S}{1 k \Omega + 7 k \Omega + 10 k \Omega}\right) \ln(0.0004885)
\]

\[
= 16.47 \mu s
\]

\[
T_{ACQ} = 2 \mu s + 16.47 \mu s + \left(\frac{50^\circ C - 25^\circ C}{0.05 \mu s/\circ C}\right)
\]

\[
= 19.72 \mu s
\]

**Note 1:** The reference voltage (VREF) has no effect on the equation, since it cancels itself out.

**2:** The charge holding capacitor (CHOLD) is not discharged after each conversion.

**3:** The maximum recommended impedance for analog sources is 10 kΩ. This is required to meet the pin leakage specification.

**FIGURE 7-3: ANALOG INPUT MODEL**
7.3 A/D Operation During SLEEP

The A/D converter module can operate during SLEEP. This requires the A/D clock source to be set to the internal oscillator. When the RC clock source is selected, the A/D waits one instruction before starting the conversion. This allows the SLEEP instruction to be executed, thus eliminating much of the switching noise from the conversion. When the conversion is complete, the GO/DONE bit is cleared, and the result is loaded into the ADRESH:ADRESL registers. If the A/D interrupt is enabled, the device awakens from SLEEP. If the A/D interrupt is not enabled, the A/D module is turned off, although the ADON bit remains set.

7.4 Effects of RESET

A device RESET forces all registers to their RESET state. Thus, the A/D module is turned off and any pending conversion is aborted. The ADRESH:ADRESL registers are unchanged.

<table>
<thead>
<tr>
<th>Address</th>
<th>Name</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>Value on: POR</th>
<th>Value on all other RESETS</th>
</tr>
</thead>
<tbody>
<tr>
<td>05h</td>
<td>PORTA</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>PORTA5</td>
<td>PORTA4</td>
<td>PORTA3</td>
<td>PORTA2</td>
<td>PORTA1</td>
<td>PORTA0</td>
<td>--xx xxxx --uu uuuuu</td>
</tr>
<tr>
<td>07h</td>
<td>PORTC</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>PORTC5</td>
<td>PORTC4</td>
<td>PORTC3</td>
<td>PORTC2</td>
<td>PORTC1</td>
<td>PORTC0</td>
<td>--xx xxxx --uu uuuuu</td>
</tr>
<tr>
<td>0Bh, 8Bh</td>
<td>INTCON</td>
<td>GIE</td>
<td>PEIE</td>
<td>T0IE</td>
<td>INTE</td>
<td>RAIE</td>
<td>T0IF</td>
<td>INTF</td>
<td>RAIF</td>
<td>0000 0000 0000 0000u</td>
<td></td>
</tr>
<tr>
<td>0Ch</td>
<td>PIR1</td>
<td>EEIF</td>
<td>ADIF</td>
<td>—</td>
<td>—</td>
<td>CMIF</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>TMR1IF</td>
<td>00-- 0--0 00-- 0--0</td>
</tr>
<tr>
<td>1Eh</td>
<td>ADRESH</td>
<td>Most Significant 8 bits of the Left Shifted A/D result or 2 bits of the Right Shifted Result</td>
<td>xxxx xxxx</td>
<td>yyyy yyyy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1Fh</td>
<td>ADCON0</td>
<td>ADFM</td>
<td>VCFG</td>
<td>—</td>
<td>—</td>
<td>CHS1</td>
<td>CHS0</td>
<td>GO</td>
<td>ADON</td>
<td>00-- 0000 00-- 0000</td>
<td></td>
</tr>
<tr>
<td>85h</td>
<td>TRISA</td>
<td>—</td>
<td>—</td>
<td>TRISA5</td>
<td>TRISA4</td>
<td>TRISA3</td>
<td>TRISA2</td>
<td>TRISA1</td>
<td>TRISA0</td>
<td>--11 1111 --11 1111</td>
<td></td>
</tr>
<tr>
<td>87h</td>
<td>TRISC</td>
<td>—</td>
<td>—</td>
<td>TRISC5</td>
<td>TRISC4</td>
<td>TRISC3</td>
<td>TRISC2</td>
<td>TRISC1</td>
<td>TRISC0</td>
<td>--11 1111 --11 1111</td>
<td></td>
</tr>
<tr>
<td>8Ch</td>
<td>PIE1</td>
<td>EEIE</td>
<td>ADIE</td>
<td>—</td>
<td>—</td>
<td>CMIE</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>TMR1IE</td>
<td>00-- 0--0 00-- 0--0</td>
</tr>
<tr>
<td>91h</td>
<td>ANSEL</td>
<td>ANS7</td>
<td>ANS6</td>
<td>ANS5</td>
<td>ANS4</td>
<td>ANS3</td>
<td>ANS2</td>
<td>ANS1</td>
<td>ANS0</td>
<td>1111 1111 1111 1111</td>
<td></td>
</tr>
<tr>
<td>9Eh</td>
<td>ADRESL</td>
<td>Least Significant 2 bits of the Left Shifted A/D Result or 8 bits of the Right Shifted Result</td>
<td>xxxx xxxx</td>
<td>yyyy yyyy</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9Fh</td>
<td>ADCON1</td>
<td>—</td>
<td>ADCS2</td>
<td>ADCS1</td>
<td>ADCS0</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>-000 ---- -000 ----</td>
<td></td>
</tr>
</tbody>
</table>

Legend:  x = unknown,  u = unchanged,  - = unimplemented read as '0'. Shaded cells are not used for A/D converter module.
8.0 DATA EEPROM MEMORY

The EEPROM data memory is readable and writable during normal operation (full VDD range). This memory is not directly mapped in the register file space. Instead, it is indirectly addressed through the Special Function Registers. There are four SFRs used to read and write this memory:

- EECON1
- EECON2 (not a physically implemented register)
- EEDATA
- EEDR

EEDATA holds the 8-bit data for read/write, and EEDR holds the address of the EEPROM location being accessed. PIC16F630/676 devices have 128 bytes of data EEPROM with an address range from 0h to 7Fh.

The EEPROM data memory allows byte read and write. A byte write automatically erases the location and writes the new data (erase before write). The EEPROM data memory is rated for high erase/write cycles. The write time is controlled by an on-chip timer. The write time will vary with voltage and temperature as well as from chip to chip. Please refer to AC Specifications for exact limits.

When the data memory is code protected, the CPU may continue to read and write the data EEPROM memory. The device programmer can no longer access this memory.

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

REGISTER 8-1: EEDAT — EEPROM DATA REGISTER (ADDRESS: 9Ah)

<table>
<thead>
<tr>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
</tr>
</thead>
<tbody>
<tr>
<td>EEDAT7</td>
<td>EEDAT6</td>
<td>EEDAT5</td>
<td>EEDAT4</td>
<td>EEDAT3</td>
<td>EEDAT2</td>
<td>EEDAT1</td>
<td>EEDAT0</td>
</tr>
</tbody>
</table>

bit 7-0 EEDATn: Byte value to write to or read from Data EEPROM

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 8-2: EEAR — EEPROM ADDRESS REGISTER (ADDRESS: 9Bh)

<table>
<thead>
<tr>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
<th>R/W-0</th>
</tr>
</thead>
<tbody>
<tr>
<td>EADR6</td>
<td>EADR5</td>
<td>EADR4</td>
<td>EADR3</td>
<td>EADR2</td>
<td>EADR1</td>
<td>EADR0</td>
<td></td>
</tr>
</tbody>
</table>

bit 7-0 EEAR: Specifies one of 128 locations for EEPROM Read/Write Operation

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
8.1 EEADR

The EEADR register can address up to a maximum of 128 bytes of data EEPROM. Only seven of the eight bits in the register (EEADR<6:0>) are required. The MSb (bit 7) is ignored.

The upper bit should always be ‘0’ to remain upward compatible with devices that have more data EEPROM memory.

8.2 EECON1 AND EECON2 REGISTERS

EECON1 is the control register with four low order bits physically implemented. The upper four bits are non-implemented and read as ‘0’s.

Control bits RD and WR initiate read and write, respectively. These bits cannot be cleared, only set, in software. They are cleared in hardware at completion of the read or write operation. The inability to clear the WR bit in software prevents the accidental, premature termination of a write operation.

The WREN bit, when set, will allow a write operation. On power-up, the WREN bit is clear. The WRERR bit is set when a write operation is interrupted by a MCLR Reset, or a WDT Time-out Reset during normal operation. In these situations, following RESET, the user can check the WRERR bit, clear it, and rewrite the location. The data and address will be cleared, therefore, the EEDATA and EEADR registers will need to be re-initialized.

Interrupt flag bit EEIF in the PIR1 register is set when write is complete. This bit must be cleared in software.

EECON2 is not a physical register. Reading EECON2 will read all ‘0’s. The EECON2 register is used exclusively in the Data EEPROM write sequence.

REGISTER 8-3: EECON1 — EEPROM CONTROL REGISTER (ADDRESS: 9Ch)

<table>
<thead>
<tr>
<th>bit 7-4</th>
<th>Unimplemented: Read as ‘0’</th>
</tr>
</thead>
<tbody>
<tr>
<td>bit 3</td>
<td>WRERR: EEPROM Error Flag bit</td>
</tr>
<tr>
<td>1</td>
<td>A write operation is prematurely terminated (any MCLR Reset, any WDT Reset during normal operation or BOD detect)</td>
</tr>
<tr>
<td>0</td>
<td>The write operation completed</td>
</tr>
<tr>
<td>bit 2</td>
<td>WREN: EEPROM Write Enable bit</td>
</tr>
<tr>
<td>1</td>
<td>Allows write cycles</td>
</tr>
<tr>
<td>0</td>
<td>Inhibits write to the data EEPROM</td>
</tr>
<tr>
<td>bit 1</td>
<td>WR: Write Control bit</td>
</tr>
<tr>
<td>1</td>
<td>Initiates a write cycle (The bit is cleared by hardware once write is complete. The WR bit can only be set, not cleared, in software.)</td>
</tr>
<tr>
<td>0</td>
<td>Write cycle to the data EEPROM is complete</td>
</tr>
<tr>
<td>bit 0</td>
<td>RD: Read Control bit</td>
</tr>
<tr>
<td>1</td>
<td>Initiates an EEPROM read (Read takes one cycle. RD is cleared in hardware. The RD bit can only be set, not cleared, in software.)</td>
</tr>
<tr>
<td>0</td>
<td>Does not initiate an EEPROM read</td>
</tr>
</tbody>
</table>

Legend:
S = Bit can only be set
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
8.3 READING THE EEPROM DATA MEMORY

To read a data memory location, the user must write the address to the EEADR register and then set control bit RD (EECON1<0>), as shown in Example 8-1. The data is available, in the very next cycle, in the EEDATA register. Therefore, it can be read in the next instruction. EEDATA holds this value until another read, or until it is written to by the user (during a write operation).

EXAMPLE 8-1: DATA EEPROM READ

```assembly
bsf STATUS,RP0 ;Bank 1
movlw CONFIG_ADDR ;
movwf EEADR ;Address to read
bsf EECON1,RD ;EE Read
movf EEDATA,W ;Move data to W
```

8.4 WRITING TO THE EEPROM DATA MEMORY

To write an EEPROM data location, the user must first write the address to the EEADR register and the data to the EEDATA register. Then the user must follow a specific sequence to initiate the write for each byte, as shown in Example 8-2.

EXAMPLE 8-2: DATA EEPROM WRITE

```assembly
bsf STATUS,RP0 ;Bank 1
bsf EECON1,WREN ;Enable write
bcf INTCON,GIE ;Disable INTs
movlw 55h ;Unlock write
movwf EECON2 ;
movlw AAh ;
movwf EECON2 ;
bsf EECON1,WR ;Start the write
bsf INTCON,GIE ;Enable INTs
```

The write will not initiate if the above sequence is not exactly followed (write 55h to EECON2, write AAh to EECON2, then set WR bit) for each byte. We strongly recommend that interrupts be disabled during this code segment. A cycle count is executed during the required sequence. Any number that is not equal to the required cycles to execute the required sequence will prevent the data from being written into the EEPROM.

Additionally, the WREN bit in EECON1 must be set to enable write. This mechanism prevents accidental writes to data EEPROM due to errant (unexpected) code execution (i.e., lost programs). The user should keep the WREN bit clear at all times, except when updating EEPROM. The WREN bit is not cleared by hardware.

After a write sequence has been initiated, clearing the WREN bit will not affect this write cycle. The WR bit will be inhibited from being set unless the WREN bit is set.

At the completion of the write cycle, the WR bit is cleared in hardware and the EE Write Complete Interrupt Flag bit (EEIF) is set. The user can either enable this interrupt or poll this bit. The EEIF bit (PIR<7>) register must be cleared by software.

8.5 WRITE VERIFY

Depending on the application, good programming practice may dictate that the value written to the Data EEPROM should be verified (see Example 8-3) to the desired value to be written.

EXAMPLE 8-3: WRITE VERIFY

```assembly
bsf STATUS,RP0 ;Bank 0
: ;Any code
bsf STATUS,RP0 ;Bank 1 READ
movf EEDATA,W ;EEDATA not changed
movlw 55h ;from previous write
movf EEDATA,W
xorwf EEDATA,W ;Is data written
btfss STATUS,Z ;Is data the same
goto WRITE_ERR ;No, handle error
: ;Yes, continue
```

8.6 PROTECTION AGAINST SPURIOUS WRITE

There are conditions when the user may not want to write to the data EEPROM memory. To protect against spurious EEPROM writes, various mechanisms have been built in. On power-up, WREN is cleared. Also, the Power-up Timer (72 ms duration) prevents EEPROM write.

The write initiate sequence and the WREN bit together help prevent an accidental write during:
- brown-out
- power glitch
- software malfunction
8.7 DATA EEPROM OPERATION DURING CODE PROTECT

Data memory can be code protected by programming the CPD bit to '0'.

When the data memory is code protected, the CPU is able to read and write data to the Data EEPROM. It is recommended to code protect the program memory when code protecting data memory. This prevents anyone from programming zeroes over the existing code (which will execute as NOOPS) to reach an added routine, programmed in unused program memory, which outputs the contents of data memory. Programming unused locations to '0' will also help prevent data memory code protection from becoming breached.

TABLE 8-1: REGISTERS/BITS ASSOCIATED WITH DATA EEPROM

<table>
<thead>
<tr>
<th>Address</th>
<th>Name</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>Value on Power-on Reset</th>
<th>Value on all other RESETS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0Ch</td>
<td>PIR1</td>
<td>EEIF</td>
<td>ADIF</td>
<td>—</td>
<td>—</td>
<td>CMIF</td>
<td>—</td>
<td>—</td>
<td>TMR1IF</td>
<td>00-- 0--0</td>
<td>00-- 0--0</td>
</tr>
<tr>
<td>9Ah</td>
<td>EEDATA</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0000 0000 0000 0000</td>
<td></td>
</tr>
<tr>
<td>9Bh</td>
<td>EEADR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>EEDATA</td>
<td>-000 0000 0000 0000</td>
<td></td>
</tr>
<tr>
<td>9Ch</td>
<td>EECON1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>WRERR</td>
<td>----- x000</td>
<td>----- q000</td>
</tr>
<tr>
<td>9Dh</td>
<td>EECON2(1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>RD</td>
<td>----- -----</td>
<td>----- -----</td>
</tr>
</tbody>
</table>

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

Shaded cells are not used by Data EEPROM module.

Note 1: EECON2 is not a physical register.
9.0 SPECIAL FEATURES OF THE CPU

Certain special circuits that deal with the needs of real time applications are what sets a microcontroller apart from other processors. The PIC16F630/676 family has a host of such features intended to:

- maximize system reliability
- minimize cost through elimination of external components
- provide power saving Operating modes and offer code protection.

These features are:

- Oscillator 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

The PIC16F630/676 has a Watchdog Timer that is controlled by 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 of 72 ms (nominal) on power-up only, designed to keep the part in RESET while the power supply stabilizes. There is also circuitry to reset the device if a brown-out occurs, which can provide at least a 72 ms RESET. With these three functions on-chip, most applications need no external RESET circuitry.

The 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
- 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 (see Register 9-1).
9.1 Configuration Bits

The configuration bits can be programmed (read as ‘0’), or left unprogrammed (read as ‘1’) to select various
device configurations, as shown in Register 9-1. These
bits are mapped in program memory location 2007h.

REGISTER 9-1: CONFIG — CONFIGURATION WORD (ADDRESS: 2007h)

<table>
<thead>
<tr>
<th>R/P-1</th>
<th>R/P-1</th>
<th>U-0</th>
<th>U-0</th>
<th>U-0</th>
<th>R/P-1</th>
<th>R/P-1</th>
<th>R/P-1</th>
<th>R/P-1</th>
<th>R/P-1</th>
<th>R/P-1</th>
<th>R/P-1</th>
<th>R/P-1</th>
<th>R/P-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>BG1</td>
<td>BG0</td>
<td></td>
<td></td>
<td></td>
<td>CPD</td>
<td>CP</td>
<td>BODEN</td>
<td>MCLRE</td>
<td>PWRT</td>
<td>WDTE</td>
<td>F0SC2</td>
<td>F0SC1</td>
<td>F0SC0</td>
</tr>
</tbody>
</table>

bit 13-12  **BG1:BG0**: Bandgap Calibration bits(1)
  00 = Lowest bandgap voltage
  11 = Highest bandgap voltage

bit 11-9  **Unimplemented**: Read as ‘0’

bit 8  **CPD**: Data Code Protection bit(2)
  1 = Data memory code protection is disabled
  0 = Data memory code protection is enabled

bit 7  **CP**: Code Protection bit(3)
  1 = Program Memory code protection is disabled
  0 = Program Memory code protection is enabled

bit 6  **BODEN**: Brown-out Detect Enable bit(4)
  1 = BOD enabled
  0 = BOD disabled

bit 5  **MCLRE**: RA3/MCLR pin function select(5)
  1 = RA3/MCLR pin function is MCLR
  0 = RA3/MCLR pin function is digital I/O, MCLR internally tied to Vdd

bit 4  **PWRT**: Power-up Timer Enable bit
  1 = PWRT disabled
  0 = PWRT enabled

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

bit 2-0  **Fosc2:Fosc0**: Oscillator Selection bits
  111 = RC oscillator: CLKOUT function on RA4/OSC2/CLKOUT pin, RC on RA5/OSC1/CLKIN
  110 = RC oscillator: I/O function on RA4/OSC2/CLKOUT pin, RC on RA5/OSC1/CLKIN
  101 = INTOSC oscillator: CLKOUT function on RA4/OSC2/CLKOUT pin, I/O function on RA5/OSC1/CLKIN
  100 = INTOSC oscillator: I/O function on RA4/OSC2/CLKOUT pin, I/O function on RA5/OSC1/CLKIN
  011 = EC: I/O function on RA4/OSC2/CLKOUT pin, CLkin on RA5/OSC1/CLKIN
  010 = HS oscillator: High speed crystal/resonator on RA4/OSC2/CLKOUT and RA5/OSC1/CLKIN
  001 = XT oscillator: Crystal/resonator on RA4/OSC2/CLKOUT and RA5/OSC1/CLKIN
  000 = LP oscillator: Low power crystal on RA4/OSC2/CLKOUT and RA5/OSC1/CLKIN

**Note 1**: The Bandgap Calibration bits are factory programmed and must be read and saved prior to erasing
the device as specified in the PIC16F630/676 Programming Specification. These bits are not
reflected in an export of the configuration word. Microchip Development Tools maintain all calibra-
tion bits to factory settings.

**Note 2**: The entire data EEPROM will be erased when the code protection is turned off.

**Note 3**: The entire program EEPROM will be erased, including OSCCAL value, when the code protection is
turned off.

**Note 4**: Enabling Brown-out Reset does not automatically enable Power-up Timer.

**Note 5**: When MCLRE is asserted in INTOSC or RC mode, the internal clock oscillator is disabled.

Legend:
P = Programmed using ICSP
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

Note: Address 2007h is beyond the user program memory space. It belongs to the special con-
figuration memory space (2000h - 3FFFh), which can be accessed only during program-
ning. See PIC16F630/676 Programming Specification for more information.
9.2 Oscillator Configurations

9.2.1 OSCILLATOR TYPES

The PIC16F630/676 can be operated in eight different Oscillator Option modes. The user can program three configuration bits (FOSC2 through FOSC0) to select one of these eight modes:

- **LP** Low Power Crystal
- **XT** Crystal/Resonator
- **HS** High Speed Crystal/Resonator
- **RC** External Resistor/Capacitor (2 modes)
- **INTOSC** Internal Oscillator (2 modes)
- **EC** External Clock In

**Note:** Additional information on oscillator configurations is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

9.2.2 CRYSTAL OSCILLATOR / CERAMIC RESONATORS

In XT, LP or HS modes a crystal or ceramic resonator is connected to the OSC1 and OSC2 pins to establish oscillation (see Figure 9-1). The PIC16F630/676 oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may yield a frequency outside of the crystal manufacturer’s specifications. When in XT, LP or HS modes, the device can have an external clock source to drive the OSC1 pin (see Figure 9-2).

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

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

**TABLE 9-1: CAPACITOR SELECTION FOR CERAMIC RESONATORS**

<table>
<thead>
<tr>
<th>Mode</th>
<th>Freq</th>
<th>OSC1(C1)</th>
<th>OSC2(C2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>XT</td>
<td>455 kHz</td>
<td>68 - 100 pF</td>
<td>68 - 100 pF</td>
</tr>
<tr>
<td></td>
<td>2.0 MHz</td>
<td>15 - 68 pF</td>
<td>15 - 68 pF</td>
</tr>
<tr>
<td></td>
<td>4.0 MHz</td>
<td>15 - 68 pF</td>
<td>15 - 68 pF</td>
</tr>
<tr>
<td>HS</td>
<td>8.0 MHz</td>
<td>10 - 68 pF</td>
<td>10 - 68 pF</td>
</tr>
<tr>
<td></td>
<td>16.0 MHz</td>
<td>10 - 22 pF</td>
<td>10 - 22 pF</td>
</tr>
</tbody>
</table>

**Note 1:** Higher capacitance increases the stability of the oscillator but also increases the start-up time. These values are for design guidance only. Since each resonator has its own characteristics, the user should consult the resonator manufacturer for appropriate values of external components.

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

<table>
<thead>
<tr>
<th>Mode</th>
<th>Freq</th>
<th>OSC1(C1)</th>
<th>OSC2(C2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LP</td>
<td>32 kHz</td>
<td>15 - 30 pF</td>
<td>15 - 30 pF</td>
</tr>
<tr>
<td>XT</td>
<td>100 kHz</td>
<td>68 - 150 pF</td>
<td>150 - 200 pF</td>
</tr>
<tr>
<td></td>
<td>2 MHz</td>
<td>15 - 30 pF</td>
<td>15 - 30 pF</td>
</tr>
<tr>
<td></td>
<td>4 MHz</td>
<td>15 - 30 pF</td>
<td>15 - 30 pF</td>
</tr>
<tr>
<td>HS</td>
<td>8 MHz</td>
<td>15 - 30 pF</td>
<td>15 - 30 pF</td>
</tr>
<tr>
<td></td>
<td>10 MHz</td>
<td>15 - 30 pF</td>
<td>15 - 30 pF</td>
</tr>
<tr>
<td></td>
<td>20 MHz</td>
<td>15 - 30 pF</td>
<td>15 - 30 pF</td>
</tr>
</tbody>
</table>

**Note 1:** Higher capacitance increases the stability of the oscillator but also increases the start-up time. These values are for design guidance only. Rs may be required in HS mode as well as XT mode to avoid over-driving crystals with low drive level specifications. Since each crystal has its own characteristics, the user should consult the crystal manufacturer for appropriate values of external components.
9.2.3 EXTERNAL CLOCK IN

For applications where a clock is already available elsewhere, users may directly drive the PIC16F630/676 provided that this external clock source meets the AC/DC timing requirements listed in Section 12.0. Figure 9-2 below shows how an external clock circuit should be configured. This Oscillator mode sets RA4 as an I/O.

9.2.4 RC OSCILLATOR

For applications where precise timing is not a requirement, the RC oscillator option is available. The operation and functionality of the RC oscillator is dependent upon a number of variables. The RC oscillator frequency is a function of:

- Supply voltage
- Resistor (REXT) and capacitor (CEXT) values
- Operating temperature.

The oscillator frequency will vary from unit to unit due to normal process parameter variation. The difference in lead frame capacitance between package types will also affect the oscillation frequency, especially for low CEXT values. The user also needs to account for the tolerance of the external R and C components. Figure 9-3 shows how the R/C combination is connected.

Two options are available for this Oscillator mode which allow RA4 to be used as a general purpose I/O or to output FOSC/4.

FIGURE 9-3: RC OSCILLATOR MODE

9.2.5 INTERNAL 4 MHz OSCILLATOR

When calibrated, the internal oscillator provides a fixed 4 MHz (nominal) system clock. The internal oscillator starts in 8 clock cycles. See Electrical Specifications, Section 12.0, for information on variation over voltage and temperature.

Two options are available for this Oscillator mode which allow RA4 to be used as a general purpose I/O or to output Fosc/4.

9.2.5.1 Calibrating the Internal Oscillator

A calibration instruction is programmed into the last location of program memory. This instruction is a RETLW XX, where the literal is the calibration value. The literal is placed in the OSCCAL register to set the calibration of the internal oscillator. Example 9-1 demonstrates how to calibrate the internal oscillator. For best operation, decouple (with capacitance) VDD and Vss as close to the device as possible.

EXAMPLE 9-1: CALIBRATING THE INTERNAL OSCILLATOR

bsf STATUS, RP0 ;Bank 1
call 3FFh ;Get the cal value
movwf OSCCAL ;Calibrate
bcf STATUS, RP0 ;Bank 0

9.2.6 CLKOUT

The PIC16F630/676 devices can be configured to provide a clock out signal in the INTOSC and RC Oscillator modes. When configured, the oscillator frequency divided by four (Fosc/4) is output on the RA4/OSC2/CLKOUT pin. Fosc/4 can be used for test purposes or to synchronize other logic.

Note: Erasing the device will also erase the pre-programmed internal calibration value for the internal oscillator. The calibration value must be saved prior to erasing part as specified in the PIC16F630/676 Programming Specification. Microchip Development Tools maintain all calibration bits to their factory settings.
9.3 RESET

The PIC16F630/676 differentiates between various kinds of RESET:

a) Power-on Reset (POR)
b) WDT Reset during normal operation
c) WDT Reset during SLEEP
d) MCLR Reset (normal operation)
e) Brown-out Detect (BOD)

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
- MCLR Reset
- WDT Reset
- WDT Reset during SLEEP
- Brown-out Detect (BOD) Reset

They are not affected by a WDT wake-up, since this is viewed as the resumption of normal operation. TO and 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 9-7 for a full description of RESET states of all registers.

A simplified block diagram of the On-Chip Reset Circuit is shown in Figure 9-4.

The MCLR Reset path has a noise filter to detect and ignore small pulses. See Table 12-4 in the Electrical Specifications section for pulse width specification.

FIGURE 9-4: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT

---

Note 1: This is a separate oscillator from the INTOSC/EC oscillator.
9.3.1 MCLR

PIC16F630/676 devices have a noise filter in the MCLR Reset path. The filter will detect and ignore small pulses.

It should be noted that a WDT Reset does not drive MCLR pin low.

The behavior of the ESD protection on the MCLR pin has been altered from previous devices of this family. Voltages applied to the pin that exceed its specification can result in both MCLR Resets and excessive current beyond the device specification during the ESD event. For this reason, Microchip recommends that the MCLR pin no longer be tied directly to VDD. The use of an RC network, as shown in Figure 9-5, is suggested.

An internal MCLR option is enabled by setting the MCLRE bit in the configuration word. When enabled, MCLR is internally tied to VDD.

FIGURE 9-5: RECOMMENDED MCLR CIRCUIT

9.3.2 POWER-ON RESET (POR)

The on-chip POR circuit holds the chip in RESET until VDD has reached a high enough level for proper operation. To take advantage of the POR, simply tie the MCLR pin through a resistor to VDD. This will eliminate external RC components usually needed to create Power-on Reset. A maximum rise time for VDD is specified. See Electrical Specifications for details.

Note: The POR circuit does not produce an internal RESET when VDD declines.

When the device starts normal operation (exits the RESET condition), device operating parameters (i.e., voltage, frequency, temperature, etc.) 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.

For additional information, refer to Application Note AN607 “Power-up Trouble Shooting”.

9.3.3 POWER-UP TIMER (PWRT)

The Power-up Timer provides a fixed 72 ms (nominal) time-out on power-up only, from POR or Brown-out Reset. The Power-up Timer operates on an internal oscillator. The chip is kept in RESET as long as PWRT is active. The PWRT delay allows the VDD to rise to an acceptable level. A configuration bit, PWREN, can disable (if set) or enable (if cleared or programmed) the Power-up Timer. A Brown-out Reset will use the Power-up Timer, only if the PWREN configuration bit is programmed.

The Power-up Time delay will vary from chip to chip and due to:

- VDD variation
- Temperature variation
- Process variation

See DC parameters for details.

9.3.4 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.

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

9.3.5 BROWN-OUT DETECT (BOD)

The PIC16F630/676 members have on-chip Brown-out Detect circuitry. A configuration bit, BODEN, can disable (if clear/programmed) or enable (if set) the Brown-out Detect circuitry. If VDD falls below VBOR for greater than parameter (TBOR) in Table 12-4 (see Section 12.0), the Brown-out situation will reset the device. A 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 BVDD (see Figure 9-6). This will occur regardless of VDD slew-rate. The Power-up Timer will now be invoked, if enabled, and will keep the chip in RESET an additional 72 ms.

Note: A Brown-out Reset does not enable the Power-up Timer if the PWREN bit in the configuration word is set.

If VDD drops below BVDD 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 BVDD, the Power-Up Timer will execute a 72 ms RESET.
9.3.6 TIME-OUT SEQUENCE

On power-up, the time-out sequence is as follows: first, PWRT time-out is invoked after POR has expired. Then, OST is activated. The total time-out will vary based on oscillator configuration and PWRT bit status. For example, in EC mode with PWRT bit erased (PWRT disabled), there will be no time-out at all. Figure 9-7, Figure 9-8 and Figure 9-9 depict time-out sequences.

Since the time-outs occur from the POR pulse, if MCLR is kept low long enough, the time-outs will expire. Then bringing MCLR high will begin execution immediately (see Figure 9-8). This is useful for testing purposes or to synchronize more than one PIC16F630/676 device operating in parallel.

Table 9-6 shows the RESET conditions for some special registers, while Table 9-7 shows the RESET conditions for all the registers.

9.3.7 POWER CONTROL (PCON) STATUS REGISTER

The Power Control/Status register, PCON (address 8Eh) has two bits.

Bit0 is BOD (Brown-out). BOD is unknown on Power-on Reset. It must then be set by the user and checked on subsequent RESETS to see if $BOD = 0$, indicating that a brown-out has occurred. The BOD status bit is a don't care and is not necessarily predictable if the brown-out circuit is disabled (by setting BODEN bit = 0 in the configuration word).

Bit1 is POR (Power-on Reset). It is a ‘0’ on Power-on Reset and unaffected otherwise. The user must write a ‘1’ to this bit following a Power-on Reset. On a subsequent RESET, if POR is ‘0’, it will indicate that a Power-on Reset must have occurred (i.e., VDD may have gone too low).

Note 1: 72 ms delay only if PWRT bit is programmed to ‘0’.

**FIGURE 9-6: BROWN-OUT SITUATIONS**

- [Diagram showing different brown-out situations]
- Note 1: 72 ms delay only if PWRT bit is programmed to ‘0’.
### TABLE 9-3: TIME-OUT IN VARIOUS SITUATIONS

<table>
<thead>
<tr>
<th>Oscillator Configuration</th>
<th>Power-up PWRTE = 0</th>
<th>Power-up PWRTE = 1</th>
<th>Brown-out PWRTE = 0</th>
<th>Brown-out PWRTE = 1</th>
<th>Wake-up from SLEEP</th>
</tr>
</thead>
<tbody>
<tr>
<td>XT, HS, LP</td>
<td>TPWRT + 1024•TOSC</td>
<td>1024•TOSC</td>
<td>TPWRT + 1024•TOSC</td>
<td>1024•TOSC</td>
<td>1024•TOSC</td>
</tr>
<tr>
<td>RC, EC, INTOSC</td>
<td>TPWRT</td>
<td>—</td>
<td>TPWRT</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

### TABLE 9-4: STATUS/PCON BITS AND THEIR SIGNIFICANCE

<table>
<thead>
<tr>
<th>POR</th>
<th>BOD</th>
<th>TO</th>
<th>PD</th>
<th>Value on POR Reset</th>
<th>Value on all other Resets(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>u</td>
<td>1</td>
<td>1</td>
<td>Power-on Reset</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>Brown-out Detect</td>
<td></td>
</tr>
<tr>
<td>u</td>
<td>u</td>
<td>0</td>
<td>u</td>
<td>WDT Reset</td>
<td></td>
</tr>
<tr>
<td>u</td>
<td>u</td>
<td>0</td>
<td>0</td>
<td>WDT Wake-up</td>
<td></td>
</tr>
<tr>
<td>u</td>
<td>u</td>
<td>u</td>
<td>u</td>
<td>MCLR Reset during normal operation</td>
<td></td>
</tr>
<tr>
<td>u</td>
<td>u</td>
<td>1</td>
<td>0</td>
<td>MCLR Reset during SLEEP</td>
<td></td>
</tr>
</tbody>
</table>

Legend: u = unchanged, x = unknown

### TABLE 9-5: SUMMARY OF REGISTERS ASSOCIATED WITH BROWN-OUT

<table>
<thead>
<tr>
<th>Address</th>
<th>Name</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>Value on POR Reset</th>
<th>Value on all other Resets(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>03h</td>
<td>STATUS</td>
<td>IRP</td>
<td>RP1</td>
<td>RPO</td>
<td>TO</td>
<td>PD</td>
<td>Z</td>
<td>DC</td>
<td>C</td>
<td>0001 1xxx</td>
<td>000q quuu</td>
</tr>
<tr>
<td>8Eh</td>
<td>PCON</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>POR BOD</td>
<td></td>
</tr>
</tbody>
</table>

Note 1: Other (non Power-up) Resets include MCLR Reset, Brown-out Detect and Watchdog Timer Reset during normal operation.

### TABLE 9-6: INITIALIZATION CONDITION FOR SPECIAL REGISTERS

<table>
<thead>
<tr>
<th>Condition</th>
<th>Program Counter</th>
<th>STATUS Register</th>
<th>PCON Register</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power-on Reset</td>
<td>000h</td>
<td>0001 1xxx</td>
<td>---- --0x</td>
</tr>
<tr>
<td>MCLR Reset during normal operation</td>
<td>000h</td>
<td>000u uuuu</td>
<td>---- --uu</td>
</tr>
<tr>
<td>MCLR Reset during SLEEP</td>
<td>000h</td>
<td>0001 0uuu</td>
<td>---- --uu</td>
</tr>
<tr>
<td>WDT Reset</td>
<td>000h</td>
<td>0000 uuuu</td>
<td>---- --uu</td>
</tr>
<tr>
<td>WDT Wake-up</td>
<td>PC + 1</td>
<td>uuuu 0uuu</td>
<td>---- --uu</td>
</tr>
<tr>
<td>Brown-out Detect</td>
<td>000h</td>
<td>0001 1uuu</td>
<td>---- --10</td>
</tr>
<tr>
<td>Interrupt Wake-up from SLEEP</td>
<td>PC + 1(1)</td>
<td>uuu1 0uuu</td>
<td>---- --uu</td>
</tr>
</tbody>
</table>

Legend: u = unchanged, x = unknown, - = unimplemented bit, reads as ‘0’.  
**Note 1:** When the wake-up is due to an interrupt and global enable bit GIE is set, the PC is loaded with the interrupt vector (0004h) after execution of PC+1.
<table>
<thead>
<tr>
<th>Register</th>
<th>Address</th>
<th>Power-on Reset</th>
<th>MCLR Reset</th>
<th>WDT Reset</th>
<th>Brown-out Detect(1)</th>
<th>Wake-up from SLEEP through interrupt</th>
<th>Wake-up from SLEEP through WDT time-out</th>
</tr>
</thead>
<tbody>
<tr>
<td>W</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>INDF</td>
<td>00h/80h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>TMR0</td>
<td>01h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>PCL</td>
<td>02h/82h</td>
<td>0000 0000</td>
<td>0000 0000</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>STATUS</td>
<td>03h/83h</td>
<td>0001 1xxx</td>
<td>000q quuu(4)</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>FSR</td>
<td>04h/84h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>PORTA</td>
<td>05h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>PORTC</td>
<td>07h</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>PCLATH</td>
<td>0Ah/8Ah</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>INTCON</td>
<td>0Bh/8Bh</td>
<td>0000 0000</td>
<td>0000 000u</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
</tr>
<tr>
<td>PIR1</td>
<td>0Ch</td>
<td>00-- 0--0</td>
<td>00-- 0--0</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>T1CON</td>
<td>10h</td>
<td>-000 0000</td>
<td>-uuu uuq</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>CMCON</td>
<td>19h</td>
<td>-0-0 0000</td>
<td>-uu uuq</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>ADRESH</td>
<td>1Eh</td>
<td>xxxxx xxxxx</td>
<td>uu uuq</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>ADCON0</td>
<td>1Fh</td>
<td>00-0 0000</td>
<td>00-0 000u</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
</tr>
<tr>
<td>OPTION_REG</td>
<td>81h</td>
<td>1111 1111</td>
<td>1111 1111</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
</tr>
<tr>
<td>TRISA</td>
<td>85h</td>
<td>--11 1111</td>
<td>--11 1111</td>
<td>--uu uuq</td>
<td>--uu uuq</td>
<td>--uu uuq</td>
<td>--uu uuq</td>
</tr>
<tr>
<td>TRISC</td>
<td>87h</td>
<td>--11 1111</td>
<td>--11 1111</td>
<td>--uu uuq</td>
<td>--uu uuq</td>
<td>--uu uuq</td>
<td>--uu uuq</td>
</tr>
<tr>
<td>PIE1</td>
<td>8Ch</td>
<td>00-- 0--0</td>
<td>00-- 0--0</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
</tr>
<tr>
<td>PCON</td>
<td>8Eh</td>
<td>---- ----</td>
<td>---- ----</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
</tr>
<tr>
<td>OSCCAL</td>
<td>90h</td>
<td>1000 00--</td>
<td>1000 00--</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
</tr>
<tr>
<td>ANSEL</td>
<td>91h</td>
<td>1111 1111</td>
<td>1111 1111</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
</tr>
<tr>
<td>WPUA</td>
<td>95h</td>
<td>--11 --111</td>
<td>--11 --111</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
</tr>
<tr>
<td>IOWA</td>
<td>96h</td>
<td>--00 0000</td>
<td>--00 0000</td>
<td>--uu uuq</td>
<td>--uu uuq</td>
<td>--uu uuq</td>
<td>--uu uuq</td>
</tr>
<tr>
<td>VRCON</td>
<td>99h</td>
<td>0--0 0000</td>
<td>0--0 0000</td>
<td>u-u uuq</td>
<td>u-u uuq</td>
<td>u-u uuq</td>
<td>u-u uuq</td>
</tr>
<tr>
<td>EEDATA</td>
<td>9Ah</td>
<td>0000 0000</td>
<td>0000 0000</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
</tr>
<tr>
<td>EEADR</td>
<td>9Bh</td>
<td>-000 0000</td>
<td>-000 0000</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
</tr>
<tr>
<td>EECON1</td>
<td>9Ch</td>
<td>---- x000</td>
<td>---- q000</td>
<td>---- uuq</td>
<td>---- uuq</td>
<td>---- uuq</td>
<td>---- uuq</td>
</tr>
<tr>
<td>EECON2</td>
<td>9Dh</td>
<td>---- ----</td>
<td>---- ----</td>
<td>---- ----</td>
<td>---- ----</td>
<td>---- ----</td>
<td>---- ----</td>
</tr>
<tr>
<td>ADRESL</td>
<td>9Eh</td>
<td>xxxxx xxxxx</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
<td>uu uuq</td>
</tr>
<tr>
<td>ADCON1</td>
<td>9Fh</td>
<td>-000 ----</td>
<td>-000 ----</td>
<td>-000 ----</td>
<td>-000 ----</td>
<td>-000 ----</td>
<td>-000 ----</td>
</tr>
</tbody>
</table>

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

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

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

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

**Note 4:** See Table 9-6 for RESET value for specific condition.

**Note 5:** If wake-up was due to data EEPROM write completing, bit 7 = 1; A/D conversion completing, bit 6 = 1; Comparator input changing, bit 3 = 1; or Timer1 rolling over, bit 0 = 1. All other interrupts generating a wake-up will cause these bits to = u.

**Note 6:** If RESET was due to brown-out, then bit 0 = 0. All other RESETS will cause bit 0 = u.
FIGURE 9-7: TIME-OUT SEQUENCE ON POWER-UP (MCLR NOT TIED TO VDD): CASE 1

- VDD
- MCLR
- Internal POR
- TPWRT
- PWRT Time-out
- OST Time-out
- Internal RESET

FIGURE 9-8: TIME-OUT SEQUENCE ON POWER-UP (MCLR NOT TIED TO VDD): CASE 2

- VDD
- MCLR
- Internal POR
- TPWRT
- PWRT Time-out
- OST Time-out
- Internal RESET

FIGURE 9-9: TIME-OUT SEQUENCE ON POWER-UP (MCLR TIED TO VDD)

- VDD
- MCLR
- Internal POR
- TPWRT
- PWRT Time-out
- OST Time-out
- Internal RESET
9.4 Interrupts

The PIC16F630/676 has 7 sources of interrupt:

- External Interrupt RA2/INT
- TMR0 Overflow Interrupt
- PORTA Change Interrupts
- Comparator Interrupt
- A/D Interrupt (PIC16F676 only)
- TMR1 Overflow Interrupt
- EEPROM Data Write Interrupt

The Interrupt Control register (INTCON) and Peripheral Interrupt register (PIR) record individual interrupt requests in flag bits. It also has individual and global interrupt enable bits.

A global interrupt enable bit, GIE (INTCON<7>) enables (if set) all unmasked interrupts, or disables (if cleared) all interrupts. Individual interrupts can be disabled through their corresponding enable bits in INTCON register and PIE register. GIE is cleared on RESET.

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

The following interrupt flags are contained in the INTCON register:

- INT pin interrupt
- PORTA change interrupt
- TMR0 overflow interrupt.

The peripheral interrupt flags are contained in the special register PIR1. The corresponding interrupt enable bit is contained in Special Register PIE1.

The following interrupt flags are contained in the PIR register:

- EEPROM data write interrupt
- A/D interrupt
- Comparator interrupt
- Timer1 overflow interrupt

When an interrupt is serviced:

- The GIE is cleared to disable any further interrupt
- The return address is pushed onto the stack
- 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 multiple interrupt requests.

**Note 1:** Individual interrupt flag bits are set, regardless of the status of their corresponding mask bit or the GIE bit.

**Note 2:** When an instruction that clears the GIE bit is executed, any interrupts that were pending for execution in the next cycle are ignored. The interrupts which were ignored are still pending to be serviced when the GIE bit is set again.
FIGURE 9-10: INTERRUPT LOGIC

Note 1: PIC16F676 only.
9.4.1 RA2/INT INTERRUPT

External interrupt on RA2/INT pin is edge-triggered; either rising if INTEDG bit (OPTION<6>) is set, or falling, if INTEDG bit is clear. When a valid edge appears on the RA2/INT pin, the INTF bit (INTCON<1>) is set. This interrupt can be disabled by clearing the INTE control bit (INTCON<4>). The INTF bit must be cleared in software in the Interrupt Service Routine before re-enabling this interrupt. The RA2/INT interrupt can wake-up the processor from SLEEP if the INTE bit was set prior to going into SLEEP. The status of the GIE bit decides whether or not the processor branches to the interrupt vector following wake-up. See Section 9.7 for details on SLEEP and Figure 9-17 for timing of wake-up from SLEEP through RA2/INT interrupt.

Note: The ANSEL register (91h) must be ‘0’ to operate an analog channel as a digital input. The ANSEL register is defined for the PIC16F676 only.

9.4.2 TMR0 INTERRUPT

An overflow (FFh → 00h) in the TMR0 register will set the TOIF (INTCON<2>) bit. The interrupt can be enabled/disabled by setting/clearing TOIE (INTCON<5>) bit. For operation of the Timer0 module, see Section 4.0.

9.4.3 PORTA INTERRUPT

An input change on PORTA change sets the RAIF (INTCON<0>) bit. The interrupt can be enabled/disabled by setting/clearing the RAIE (INTCON<3>) bit. Plus individual pins can be configured through the IOCA register.

Note: If a change on the I/O pin should occur when the read operation is being executed (start of the Q2 cycle), then the RAIF interrupt flag may not get set.

9.4.4 COMPARATOR INTERRUPT

See Section 6.9 for description of comparator interrupt.

9.4.5 A/D CONVERTER INTERRUPT

After a conversion is complete, the ADIF flag (PIR<6>) is set. The interrupt can be enabled/disabled by setting or clearing ADIE (PIE<6>).

See Section 7.0 for operation of the A/D converter interrupt.

FIGURE 9-11: INT PIN INTERRUPT TIMING

Note 1: INTF flag is sampled here (every Q1).
2: Asynchronous interrupt latency = 3-4 TCY. Synchronous latency = 3 TCY, where TCY = instruction cycle time. Latency is the same whether Inst (PC) is a single cycle or a 2-cycle instruction.
3: CLKOUT is available only in RC Oscillator mode.
4: For minimum width of INT pulse, refer to AC specs.
5: INTF is enabled to be set any time during the Q4-Q1 cycles.
TABLE 9-8: SUMMARY OF INTERRUPT REGISTERS

<table>
<thead>
<tr>
<th>Address</th>
<th>Name</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>Value on POR Reset</th>
<th>Value on all other RESETS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0Bh, 8Bh</td>
<td>INTCON</td>
<td>GIE</td>
<td>PEIE</td>
<td>TOIE</td>
<td>INTE</td>
<td>RAIE</td>
<td>TOIF</td>
<td>INTF</td>
<td>RAIF</td>
<td>0000 0000</td>
<td>0000 000u</td>
</tr>
<tr>
<td>0Ch</td>
<td>PIR1</td>
<td>EEIE</td>
<td>ADIF</td>
<td>—</td>
<td>—</td>
<td>CMIF</td>
<td>—</td>
<td>—</td>
<td>TMR1IF</td>
<td>00-- 0--0</td>
<td>00-- 0--0</td>
</tr>
<tr>
<td>8Ch</td>
<td>PIE1</td>
<td>EEIE</td>
<td>ADIE</td>
<td>—</td>
<td>—</td>
<td>CMIE</td>
<td>—</td>
<td>—</td>
<td>TMR1IE</td>
<td>00-- 0--0</td>
<td>00-- 0--0</td>
</tr>
</tbody>
</table>

Legend: x = unknown, u = unchanged, - = unimplemented read as '0', q = value depends upon condition.
Shaded cells are not used by the Interrupt module.

9.5 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 (e.g., W register and STATUS register). This must be implemented in software.

Example 9-2 stores and restores the STATUS and W registers. The user register, W_TEMP, must be defined in both banks and must be defined at the same offset from the bank base address (i.e., W_TEMP is defined at 0x20 in Bank 0 and it must also be defined at 0xA0 in Bank 1). The user register, STATUS_TEMP, must be defined in Bank 0. The Example 9-2:

- Stores the W register
- Stores the STATUS register in Bank 0
- Executes the ISR code
- Restores the STATUS (and bank select bit register)
- Restores the W register

EXAMPLE 9-2: SAVING THE STATUS AND W REGISTERS IN RAM

```
MOVF W_TEMP ;copy W to temp register, could be in either bank
SWAPF STATUS,W ;swap status to be saved into W
BCF STATUS,RP0 ;change to bank 0 regardless of current bank
MOVF STATUS_TEMP ;save status to bank 0 register
  ;(ISR)
  ;
  SWAPF STATUS_TEMP,W;swap STATUS_TEMP register into W, sets bank to original state
MOVF STATUS ;move W into STATUS register
SWAPF W_TEMP,F ;swap W_TEMP
SWAPF W_TEMP,W ;swap W_TEMP into W
```

9.6 Watchdog Timer (WDT)

The Watchdog Timer is a free running, on-chip RC oscillator, which requires no external components. This RC oscillator is separate from the external RC oscillator of the CLKIN pin. That means that the WDT will run, even if the clock on the OSC1 and OSC2 pins of the device has been stopped (for example, by execution of a SLEEP instruction). During normal operation, a WDT time-out generates a device RESET. If the device is in SLEEP mode, a WDT time-out causes the device to wake-up and continue with normal operation. The WDT can be permanently disabled by programming the configuration bit WDTE as clear (Section 9.1).

9.6.1 WDT PERIOD

The WDT has a nominal time-out period of 18 ms, (with no prescaler). The time-out periods vary with temperature, VDD and process variations from part to part (see DC specs). If longer time-out periods are desired, a prescaler with a division ratio of up to 1:128 can be assigned to the WDT under software control by writing to the OPTION register. Thus, time-out periods up to 2.3 seconds can be realized.

The CLRWD T and SLEEP instructions clear the WDT and the prescaler, if assigned to the WDT, and prevent it from timing out and generating a device RESET.

The TO bit in the STATUS register will be cleared upon a Watchdog Timer time-out.

9.6.2 WDT PROGRAMMING CONSIDERATIONS

It should also be taken in account that under worst case conditions (i.e., VDD = Min., Temperature = Max., Max. WDT prescaler) it may take several seconds before a WDT time-out occurs.
FIGURE 9-12: WATCHDOG TIMER BLOCK DIAGRAM

TABLE 9-9: SUMMARY OF WATCHDOG TIMER REGISTERS

<table>
<thead>
<tr>
<th>Address</th>
<th>Name</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
<th>Value on POR Reset</th>
<th>Value on all other RESETS</th>
</tr>
</thead>
<tbody>
<tr>
<td>81h</td>
<td>OPTION_REG</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1111 1111</td>
<td>1111 1111</td>
</tr>
<tr>
<td>2007h</td>
<td>Config. bits</td>
<td>CP</td>
<td>BODEN</td>
<td>MCLRE</td>
<td>PWRT</td>
<td>WDTE</td>
<td>FOSC</td>
<td>FOSC</td>
<td>FOSC</td>
<td>uuuuuuuu</td>
<td>uuuuuuuuuuuuuuuuuuu</td>
</tr>
</tbody>
</table>

Legend: u = Unchanged, shaded cells are not used by the Watchdog Timer.

Note 1: T0SE, T0CS, PSA, PS0-PS2 are bits in the Option register.
9.7 Power-Down Mode (SLEEP)

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

If the Watchdog Timer is enabled:
- WDT will be cleared but keeps running
- PD bit in the STATUS register is cleared
- TO bit is set
- Oscillator driver is turned off
- I/O ports maintain the status they had before SLEEP was executed (driving high, low, or hi-impedance).

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

The MCLR pin must be at a logic high level (VIHMC).

Note: It should be noted that a RESET generated by a WDT time-out does not drive MCLR pin low.

9.7.1 WAKE-UP FROM SLEEP

The device can wake-up from SLEEP through one of the following events:
1. External RESET input on MCLR pin
2. Watchdog Timer Wake-up (if WDT was enabled)
3. Interrupt from RA2/INT pin, PORTA change, or a peripheral interrupt.

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

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, then branches to the interrupt address (0004h). In cases where the execution of the instruction following SLEEP is not desirable, the user should have an NOP after the SLEEP instruction.

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

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

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

---

FIGURE 9-13: WAKE-UP FROM SLEEP THROUGH INTERRUPT

Note: 1: XT, HS or LP Oscillator mode assumed.
2: T0ST = 1024Tosc (drawing not to scale). Approximately 1 μs delay for RC Osc mode and approximately 2 μs delay for INTOSC OSC mode.
3: GIE = '1' assumed. In this case after wake-up, the processor jumps to the interrupt routine. If GIE = '0', execution will continue in-line.
4: CLKOUT is not available in XT, HS or EC Osc modes, but shown here for timing reference.
9.8 Code Protection

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

### Note:
The entire data EEPROM and FLASH program memory will be erased when the code protection is turned off. The INTOSC calibration data is also erased. See PIC16F630/676 Programming Specification for more information.

9.9 ID Locations

Four memory locations (2000h-2003h) are designated as ID locations where the user can store checksum or other code identification numbers. These locations are not accessible during normal execution but are readable and writable during Program/Verify. Only the Least Significant 4 bits of the ID locations are used.

9.10 In-Circuit Serial Programming

The PIC16F630/676 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
- 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.

The device is placed into a Program/Verify mode by holding the RA0 and RA1 pins low, while raising the MCLR (VPP) pin from VIL to VIHH (see Programming Specification). RA0 becomes the programming clock and RA1 becomes the programming data. Both RA0 and RA1 are Schmitt Trigger inputs in this mode.

After RESET, to place the device into Programming/Verify mode, the program counter (PC) is at location 00h. A 6-bit command is then supplied to the device. Depending on the command, 14 bits of program data are then supplied to or from the device, depending on whether the command was a load or a read. For complete details of serial programming, please refer to the PIC16F630/676 Programming Specification.

A typical In-Circuit Serial Programming connection is shown in Figure 9-18.

### FIGURE 9-14: TYPICAL IN-CIRCUIT SERIAL PROGRAMMING CONNECTION

- External Connector Signals
- To Normal Connections
- PIC16F630/676
- +5V
- 0V
- VPP
- CLK
- Data I/O
- Vdd
- Vss
- RA3/MCLR/VPP
- RA1
- RA0

Note: The entire data EEPROM and FLASH program memory will be erased when the code protection is turned off. The INTOSC calibration data is also erased. See PIC16F630/676 Programming Specification for more information.
10.0 INSTRUCTION SET SUMMARY

The PIC16F630/676 instruction set is highly orthogonal and is comprised of three basic categories:

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

Each PIC16 instruction is a 14-bit word divided into an **opcode**, which specifies the instruction type, and one or more **operands**, which further specify the operation of the instruction. The formats for each of the categories is presented in Figure 10-1, while the various opcode fields are summarized in Table 10-1.

Table 10-2 lists the instructions recognized by the MPASM™ assembler. A complete description of each instruction is also available in the PICmicro™ Mid-Range Reference Manual (DS33023).

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

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

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

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

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

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

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

10.1 READ-MODIFY-WRITE OPERATIONS

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

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

TABLE 10-1: OPCODE FIELD DESCRIPTIONS

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

FIGURE 10-1: GENERAL FORMAT FOR INSTRUCTIONS

**Byte-oriented file register operations**

```
<table>
<thead>
<tr>
<th>Opcode</th>
<th>d</th>
<th>f (FILE #)</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>8</td>
<td>7 6 0</td>
</tr>
</tbody>
</table>
```

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

**Bit-oriented file register operations**

```
<table>
<thead>
<tr>
<th>Opcode</th>
<th>b (BIT #)</th>
<th>f (FILE #)</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>10</td>
<td>9</td>
</tr>
</tbody>
</table>
```

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

**Literal and control operations**

**General**

```
<table>
<thead>
<tr>
<th>Opcode</th>
<th>8</th>
<th>7</th>
<th>0</th>
</tr>
</thead>
</table>
```

k = 8-bit immediate value

**CALL and GOTO instructions only**

```
<table>
<thead>
<tr>
<th>Opcode</th>
<th>11</th>
<th>10</th>
<th>0</th>
</tr>
</thead>
</table>
```

k = 11-bit immediate value
### TABLE 10-2: PIC16F630/676 INSTRUCTION SET

<table>
<thead>
<tr>
<th>Mnemonic, Operands</th>
<th>Description</th>
<th>Cycles</th>
<th>14-Bit Opcode</th>
<th>Status Affected</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>MSb</td>
<td>LSb</td>
<td></td>
</tr>
<tr>
<td><strong>BYTE-ORIENTED FILE REGISTER OPERATIONS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDWF f, d</td>
<td>Add W and f</td>
<td>1</td>
<td>00 0111 dfff ffff</td>
<td>C,DC,Z</td>
<td>1,2</td>
</tr>
<tr>
<td>ANDWF f, d</td>
<td>AND W with f</td>
<td>1</td>
<td>00 0101 dfff ffff</td>
<td>Z</td>
<td>1,2</td>
</tr>
<tr>
<td>CLR f</td>
<td>Clear f</td>
<td>1</td>
<td>00 0000 1fff ffff</td>
<td>Z</td>
<td>2</td>
</tr>
<tr>
<td>CLRW -</td>
<td>Clear W</td>
<td>1</td>
<td>00 0001 0xxx xxxx</td>
<td>Z</td>
<td></td>
</tr>
<tr>
<td>COMF f, d</td>
<td>Complement f</td>
<td>1</td>
<td>00 1001 dfff ffff</td>
<td>Z</td>
<td>1,2</td>
</tr>
<tr>
<td>DEC f, d</td>
<td>Decrement f</td>
<td>1</td>
<td>00 0011 dfff ffff</td>
<td>Z</td>
<td>1,2</td>
</tr>
<tr>
<td>DECSFSZ f, d</td>
<td>Decrement f, Skip if 0</td>
<td>1(2)</td>
<td>00 1011 dfff ffff</td>
<td>Z</td>
<td>1,2,3</td>
</tr>
<tr>
<td>INC f, d</td>
<td>Increment f</td>
<td>1</td>
<td>00 1010 dfff ffff</td>
<td>Z</td>
<td>1,2</td>
</tr>
<tr>
<td>INCFSZ f, d</td>
<td>Increment f, Skip if 0</td>
<td>1(2)</td>
<td>00 1111 dfff ffff</td>
<td>Z</td>
<td>1,2,3</td>
</tr>
<tr>
<td>IORWF f, d</td>
<td>Inclusive OR W with f</td>
<td>1</td>
<td>00 0100 dfff ffff</td>
<td>Z</td>
<td>1,2</td>
</tr>
<tr>
<td>MOVF f, d</td>
<td>Move f</td>
<td>1</td>
<td>00 1000 dfff ffff</td>
<td>Z</td>
<td>1,2</td>
</tr>
<tr>
<td>MOVWF f, d</td>
<td>Move W to f</td>
<td>1</td>
<td>00 0000 1fff ffff</td>
<td>Z</td>
<td>1,2</td>
</tr>
<tr>
<td>NOP</td>
<td>No Operation</td>
<td>1</td>
<td>00 0000 0xxx 0000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RLF f, d</td>
<td>Rotate Left f through Carry</td>
<td>1</td>
<td>00 1101 dfff ffff</td>
<td>C</td>
<td>1,2</td>
</tr>
<tr>
<td>RRF f, d</td>
<td>Rotate Right f through Carry</td>
<td>1</td>
<td>00 1100 dfff ffff</td>
<td>C</td>
<td>1,2</td>
</tr>
<tr>
<td>SUBWF f, d</td>
<td>Subtract W from f</td>
<td>1</td>
<td>00 0010 dfff ffff</td>
<td>C,DC,Z</td>
<td>1,2</td>
</tr>
<tr>
<td>SWAPF f, d</td>
<td>Swap nibbles in f</td>
<td>1</td>
<td>00 1110 dfff ffff</td>
<td>Z</td>
<td>1,2</td>
</tr>
<tr>
<td>XORWF f, d</td>
<td>Exclusive OR W with f</td>
<td>1</td>
<td>00 0110 dfff ffff</td>
<td>Z</td>
<td>1,2</td>
</tr>
<tr>
<td><strong>BIT-ORIENTED FILE REGISTER OPERATIONS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BCF f, b</td>
<td>Bit Clear f</td>
<td>1</td>
<td>01 00bb bfff ffff</td>
<td>1,2</td>
<td></td>
</tr>
<tr>
<td>BSF f, b</td>
<td>Bit Set f</td>
<td>1</td>
<td>01 01bb bfff ffff</td>
<td>1,2</td>
<td></td>
</tr>
<tr>
<td>BTFSC f, b</td>
<td>Bit Test f, Skip if Clear</td>
<td>1 (2)</td>
<td>01 10bb bfff ffff</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>BTFSS f, b</td>
<td>Bit Test f, Skip if Set</td>
<td>1 (2)</td>
<td>01 11bb bfff ffff</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td><strong>LITERAL AND CONTROL OPERATIONS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADDLW k</td>
<td>Add literal and W</td>
<td>1</td>
<td>11 111x kkkk kkkk</td>
<td>C,DC,Z</td>
<td></td>
</tr>
<tr>
<td>ANDLW k</td>
<td>AND literal with W</td>
<td>1</td>
<td>11 1001 kkkk kkkk</td>
<td>Z</td>
<td></td>
</tr>
<tr>
<td>CALL k</td>
<td>Call subroutine</td>
<td>2</td>
<td>10 0kx kkkk kkkk</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CLRWDT -</td>
<td>Clear Watchdog Timer</td>
<td>1</td>
<td>00 0000 0110 0100</td>
<td>TO,PD</td>
<td></td>
</tr>
<tr>
<td>GOTO k</td>
<td>Go to address</td>
<td>2</td>
<td>10 1kx kkkk kkkk</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IORLW k</td>
<td>Inclusive OR literal with W</td>
<td>1</td>
<td>11 1000 kkkk kkkk</td>
<td>Z</td>
<td></td>
</tr>
<tr>
<td>MOVLW k</td>
<td>Move literal to W</td>
<td>1</td>
<td>11 0xxk kkkk kkkk</td>
<td>C,DC,Z</td>
<td></td>
</tr>
<tr>
<td>RETFIE</td>
<td>Return from interrupt</td>
<td>2</td>
<td>00 0000 0000 1001</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RETLW k</td>
<td>Return with literal in W</td>
<td>2</td>
<td>01 0xxk kkkk kkkk</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RETURN</td>
<td>Return from Subroutine</td>
<td>2</td>
<td>00 0000 0000 1000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SLEEP</td>
<td>Go into Standby mode</td>
<td>1</td>
<td>00 0000 0110 0011</td>
<td>TO,PD</td>
<td></td>
</tr>
<tr>
<td>SUBLW k</td>
<td>Subtract W from literal</td>
<td>1</td>
<td>11 110x kkkk kkkk</td>
<td>C,DC,Z</td>
<td></td>
</tr>
<tr>
<td>XORLW k</td>
<td>Exclusive OR literal with W</td>
<td>1</td>
<td>11 1010 kkkk kkkk</td>
<td>Z</td>
<td></td>
</tr>
</tbody>
</table>

**Note 1:** When an I/O register is modified as a function of itself (e.g., MOVF PORTA, 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 TMR0 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.

---

**Note:** Additional information on the mid-range instruction set is available in the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023).
10.2 Instruction Descriptions

**ADDLW**  Add Literal and W

Syntax: \[ \text{label} ] ADDLW \; k
Operands: \( 0 \leq k \leq 255 \)
Operation: \( (W) + k \rightarrow (W) \)
Status Affected: C, DC, Z
Description: The contents of the W register are added to the eight-bit literal \( k \) and the result is placed in the W register.

**ADDWF**  Add W and f

Syntax: \[ \text{label} ] ADDWF \; f,d
Operands: \( 0 \leq f \leq 127 \)
\( d \in \{0,1\} \)
Operation: \( (W) + (f) \rightarrow \text{destination} \)
Status Affected: C, DC, Z
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 \( f \).

**ANDLW**  AND Literal with W

Syntax: \[ \text{label} ] ANDLW \; k
Operands: \( 0 \leq k \leq 255 \)
Operation: \( (W) \; .AND. \; (k) \rightarrow (W) \)
Status Affected: Z
Description: The contents of W register are AND'ed with the eight-bit literal \( k \). The result is placed in the W register.

**ANDWF**  AND W with f

Syntax: \[ \text{label} ] ANDWF \; f,d
Operands: \( 0 \leq f \leq 127 \)
\( d \in \{0,1\} \)
Operation: \( (W) \; .AND. \; (f) \rightarrow \text{destination} \)
Status Affected: Z
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 \( f \).

**BCF**  Bit Clear f

Syntax: \[ \text{label} ] BCF \; f,b
Operands: \( 0 \leq f \leq 127 \)
\( 0 \leq b \leq 7 \)
Operation: \( 0 \rightarrow (f<b>) \)
Status Affected: None
Description: Bit \( b \) in register \( f \) is cleared.

**BSF**  Bit Set f

Syntax: \[ \text{label} ] BSF \; f,b
Operands: \( 0 \leq f \leq 127 \)
\( 0 \leq b \leq 7 \)
Operation: \( 1 \rightarrow (f<b>) \)
Status Affected: None
Description: Bit \( b \) in register \( f \) is set.

**BTFSS**  Bit Test f, Skip if Set

Syntax: \[ \text{label} ] BTFSS \; f,b
Operands: \( 0 \leq f \leq 127 \)
\( 0 \leq b < 7 \)
Operation: skip if \( (f<b>) = 1 \)
Status Affected: None
Description: If bit \( b \) in register \( f \) is '0', the next instruction is executed. If bit \( b \) is '1', then the next instruction is discarded and a NOP is executed instead, making this a 2-cycle instruction.

**BTFSC**  Bit Test, Skip if Clear

Syntax: \[ \text{label} ] BTFSC \; f,b
Operands: \( 0 \leq f \leq 127 \)
\( 0 \leq b < 7 \)
Operation: skip if \( (f<b>) = 0 \)
Status Affected: None
Description: If bit \( b \) in register \( f \) is '0', the next instruction is executed. If bit \( b \), in register \( f \), is '0', the next instruction is discarded and a NOP is executed instead, making this a 2-cycle instruction.
### CALL Call Subroutine

**Syntax:**
```
[ label ]  CALL  k
```

**Operands:**

\[ 0 \leq k \leq 2047 \]

**Operation:**

\[
(\text{PC}+1) \rightarrow \text{TOS},
\]

\[
k \rightarrow \text{PC}<10:0>,
\]

\[
(\text{PCLATH}<4:3>) \rightarrow \text{PC}<12:11>
\]

**Status Affected:** None

**Description:** Call Subroutine. First, return address (PC+1) is pushed onto the stack. The eleven-bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a two-cycle instruction.

### CLRWDT Clear Watchdog Timer

**Syntax:**
```
[ label ]  CLRWDT
```

**Operands:** None

**Operation:**

\[
00h \rightarrow \text{WDT}
\]

\[
0 \rightarrow \text{WDT prescaler},
\]

\[
1 \rightarrow \text{TO}
\]

\[
1 \rightarrow \text{PD}
\]

**Status Affected:** TO, PD

**Description:** CLRWDT instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits TO and PD are set.

### CLRF Clear f

**Syntax:**
```
[ label ]  CLRF  f
```

**Operands:**

\[ 0 \leq f \leq 127 \]

**Operation:**

\[
00h \rightarrow (f)
\]

\[
1 \rightarrow Z
\]

**Status Affected:** Z

**Description:** The contents of register 'f' are cleared and the Z bit is set.

### COMF Complement f

**Syntax:**
```
[ label ]  COMF  f,d
```

**Operands:**

\[ 0 \leq f \leq 127 \]

\[ d \in [0,1] \]

**Operation:**

\[
(f) \rightarrow (\text{destination})
\]

**Status Affected:** Z

**Description:** The contents of register 'f' are complemented. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

### CLRW Clear W

**Syntax:**
```
[ label ]  CLRW
```

**Operands:** None

**Operation:**

\[
00h \rightarrow (W)
\]

\[
1 \rightarrow Z
\]

**Status Affected:** Z

**Description:** W register is cleared. Zero bit (Z) is set.

### DECF Decrement f

**Syntax:**
```
[ label ]  DECF  f,d
```

**Operands:**

\[ 0 \leq f \leq 127 \]

\[ d \in [0,1] \]

**Operation:**

\[
(\text{destination})
\]

\[
(1) \rightarrow (f)
\]

**Status Affected:** Z

**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'.
**DECFSZ**

**Decrement f, Skip if 0**

**Syntax:** \[ label \]  DECFSZ  f,d  
**Operands:** \[ 0 ≤ f ≤ 127 \]  
\[ d ∈ [0,1] \]  
**Operation:** \( (f) - 1 \rightarrow (destination); \)  
skip if result = 0  
**Status Affected:** None  
**Description:** 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'. If the result is 1, the next instruction is executed. If the result is 0, then a NOP is executed instead, making it a 2-cycle instruction.

**INCFSZ**

**Increment f, Skip if 0**

**Syntax:** \[ label \]  INCFSZ  f,d  
**Operands:** \[ 0 ≤ f ≤ 127 \]  
\[ d ∈ [0,1] \]  
**Operation:** \( (f) + 1 \rightarrow (destination); \)  
skip if result = 0  
**Status Affected:** None  
**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 1, the next instruction is executed. If the result is 0, a NOP is executed instead, making it a 2-cycle instruction.

**GOTO**

**Unconditional Branch**

**Syntax:** \[ label \]  GOTO  k  
**Operands:** \[ 0 ≤ k ≤ 2047 \]  
**Operation:** \( k \rightarrow PC<10:0>\)  
\( PCLATH<4:3> \rightarrow PC<12:11>\)  
**Status Affected:** None  
**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.

**IORLW**

**Inclusive OR Literal with W**

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

**INCF**

**Increment f**

**Syntax:** \[ label \]  INCF  f,d  
**Operands:** \[ 0 ≤ f ≤ 127 \]  
\[ d ∈ [0,1] \]  
**Operation:** \( (f) + 1 \rightarrow (destination) \)  
**Status Affected:** Z  
**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'.

**IORWF**

**Inclusive OR W with f**

**Syntax:** \[ label \]  IORWF  f,d  
**Operands:** \[ 0 ≤ f ≤ 127 \]  
\[ d ∈ [0,1] \]  
**Operation:** \( (W) .OR. (f) \rightarrow (destination) \)  
**Status Affected:** Z  
**Description:** Inclusive OR the W register with register 'f'. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'.

---

© 2002 Microchip Technology Inc.  
**Preliminary**  
DS40039A-page 75
<table>
<thead>
<tr>
<th>Instruction</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOVF</td>
<td>The contents of register f are moved to a destination dependant 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.</td>
</tr>
<tr>
<td>Syntax: [ label] MOVF f,d</td>
<td>Operands: 0 ≤ f ≤ 127 d ∈ [0,1] Operation: (f) → (destination) Status Affected: Z</td>
</tr>
<tr>
<td>MOVLW</td>
<td>The eight-bit literal 'k' is loaded into W register. The don't cares will assemble as 0's.</td>
</tr>
<tr>
<td>Syntax: [ label] MOVLW k</td>
<td>Operands: 0 ≤ k ≤ 255 Operation: k → (W) Status Affected: None</td>
</tr>
<tr>
<td>MOVWF</td>
<td>Move data from W register to register 'f'.</td>
</tr>
<tr>
<td>Syntax: [ label] MOVWF f</td>
<td>Operands: 0 ≤ f ≤ 127 Operation: (W) → (f) Status Affected: None</td>
</tr>
<tr>
<td>NOP</td>
<td>No operation.</td>
</tr>
<tr>
<td>Syntax: [ label] NOP</td>
<td>Operands: None Operation: No operation Status Affected: None</td>
</tr>
<tr>
<td>RETFIE</td>
<td>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.</td>
</tr>
<tr>
<td>Syntax: [ label] RETFIE</td>
<td>Operands: None Operation: TOS → PC, 1 → GIE Status Affected: None</td>
</tr>
<tr>
<td>RETLW</td>
<td>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.</td>
</tr>
<tr>
<td>Syntax: [ label] RETLW k</td>
<td>Operands: 0 ≤ k ≤ 255 Operation: k → (W): TOS → PC Status Affected: None</td>
</tr>
</tbody>
</table>
### RLF Rotate Left $f$ through Carry

<table>
<thead>
<tr>
<th>Syntax:</th>
<th>[ label ] RLF $f,d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operands:</td>
<td>$0 \leq f \leq 127$</td>
</tr>
<tr>
<td></td>
<td>$d \in [0,1]$</td>
</tr>
<tr>
<td>Operation:</td>
<td>See description below</td>
</tr>
<tr>
<td>Status Affected:</td>
<td>C</td>
</tr>
<tr>
<td>Description:</td>
<td>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'.</td>
</tr>
</tbody>
</table>

### SLEEP

<table>
<thead>
<tr>
<th>Syntax:</th>
<th>[ label ] SLEEP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operands:</td>
<td>None</td>
</tr>
<tr>
<td>Operation:</td>
<td>$00h \rightarrow \text{WDT},$</td>
</tr>
<tr>
<td></td>
<td>$0 \rightarrow \text{WDT prescaler},$</td>
</tr>
<tr>
<td></td>
<td>$1 \rightarrow \text{TO},$</td>
</tr>
<tr>
<td></td>
<td>$0 \rightarrow \text{PD}$</td>
</tr>
<tr>
<td>Status Affected:</td>
<td>TO, PD</td>
</tr>
<tr>
<td>Description:</td>
<td>The power-down status bit, PD is cleared. Time-out status bit, TO is set. Watchdog Timer and its prescaler are cleared. The processor is put into SLEEP mode with the oscillator stopped.</td>
</tr>
</tbody>
</table>

### RETURN Return from Subroutine

<table>
<thead>
<tr>
<th>Syntax:</th>
<th>[ label ] RETURN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operands:</td>
<td>None</td>
</tr>
<tr>
<td>Operation:</td>
<td>TOS $\rightarrow$ PC</td>
</tr>
<tr>
<td>Status Affected:</td>
<td>None</td>
</tr>
<tr>
<td>Description:</td>
<td>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.</td>
</tr>
</tbody>
</table>

### RRW Rotate Right $f$ through Carry

<table>
<thead>
<tr>
<th>Syntax:</th>
<th>[ label ] RRW $f,d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operands:</td>
<td>$0 \leq f \leq 127$</td>
</tr>
<tr>
<td></td>
<td>$d \in [0,1]$</td>
</tr>
<tr>
<td>Operation:</td>
<td>See description below</td>
</tr>
<tr>
<td>Status Affected:</td>
<td>C</td>
</tr>
<tr>
<td>Description:</td>
<td>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 stored back in register 'f'.</td>
</tr>
</tbody>
</table>

### SUBLW Subtract $W$ from Literal

<table>
<thead>
<tr>
<th>Syntax:</th>
<th>[ label ] SUBLW $k$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operands:</td>
<td>$0 \leq k \leq 255$</td>
</tr>
<tr>
<td>Operation:</td>
<td>$k - (W) \rightarrow (W)$</td>
</tr>
<tr>
<td>Status Affected:</td>
<td>C, DC, Z</td>
</tr>
<tr>
<td>Description:</td>
<td>The W register is subtracted (2's complement method) from the eight-bit literal 'k'. The result is placed in the W register.</td>
</tr>
</tbody>
</table>

### SUBWF Subtract $W$ from $f$

<table>
<thead>
<tr>
<th>Syntax:</th>
<th>[ label ] SUBWF $f,d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operands:</td>
<td>$0 \leq f \leq 127$</td>
</tr>
<tr>
<td></td>
<td>$d \in [0,1]$</td>
</tr>
<tr>
<td>Operation:</td>
<td>$(f) - (W) \rightarrow (destination)$</td>
</tr>
<tr>
<td>Status Affected:</td>
<td>C, DC, Z</td>
</tr>
<tr>
<td>Description:</td>
<td>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'.</td>
</tr>
</tbody>
</table>
### SWAPF  Swap Nibbles in \( f \)

<table>
<thead>
<tr>
<th>Syntax:</th>
<th>([ label ] ) SWAPF ( f,d )</th>
</tr>
</thead>
</table>
| Operands: | \( 0 \leq f \leq 127 \)  
\( d \in [0,1] \) |
| Operation: | \((f<3:0>) \rightarrow (destination<7:4>), \)  
\((f<7:4>) \rightarrow (destination<3:0>)\) |
| Status Affected: | None |
| Description: | The upper and lower nibbles of register \( f \) are exchanged. If \( d \) is 0, the result is placed in the \( W \) register. If \( d \) is 1, the result is placed in register \( f \). |

### XORWF  Exclusive OR \( W \) with \( f \)

<table>
<thead>
<tr>
<th>Syntax:</th>
<th>([ label ] ) XORWF ( f,d )</th>
</tr>
</thead>
</table>
| Operands: | \( 0 \leq f \leq 127 \)  
\( d \in [0,1] \) |
| Operation: | \((W) \ .XOR. (f) \rightarrow (destination)\) |
| Status Affected: | \( Z \) |
| Description: | Exclusive OR the contents of the \( W \) register with register \( f \). If \( d \) is 0, the result is stored in the \( W \) register. If \( d \) is 1, the result is stored back in register \( f \). |

### XORLW  Exclusive OR \( W \) with Literal \( k \)

<table>
<thead>
<tr>
<th>Syntax:</th>
<th>([ label ] ) XORLW ( k )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operands:</td>
<td>( 0 \leq k \leq 255 )</td>
</tr>
<tr>
<td>Operation:</td>
<td>((W) \ .XOR. k \rightarrow (W))</td>
</tr>
<tr>
<td>Status Affected:</td>
<td>( Z )</td>
</tr>
<tr>
<td>Description:</td>
<td>The contents of the ( W ) register are XOR'ed with the eight-bit literal ( k ). The result is placed in the ( W ) register.</td>
</tr>
</tbody>
</table>
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
- Simulators
  - MPLAB SIM Software Simulator
- Emulators
  - MPLAB ICE 2000 In-Circuit Emulator
  - ICEPIC™ In-Circuit Emulator
- In-Circuit Debugger
  - MPLAB ICD
- Device Programmers
  - PRO MATE® II Universal Device Programmer
  - PICSTART® Plus Entry-Level Development Programmer
- Low Cost Demonstration Boards
  - PICDEM™ 1 Demonstration Board
  - PICDEM 2 Demonstration Board
  - PICDEM 3 Demonstration Board
  - PICDEM 17 Demonstration Board
  - KEELOQ® Demonstration Board

11.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8-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
- A project manager
- Customizable toolbar and key mapping
- A status bar
- 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
  - absolute listing file
  - machine code

The ability to use MPLAB IDE with multiple debugging tools allows users to easily switch from the cost-effective simulator to a full-featured emulator with minimal retraining.

11.2 MPASM Assembler

The MPASM assembler is a full-featured universal macro assembler for all PICmicro MCU’s.

The MPASM assembler has a command line interface and a Windows shell. It can be used as a stand-alone application on a Windows 3.x or greater system, or it can be used through MPLAB IDE. 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, an absolute LST file that contains source lines and generated machine code, and a COD file 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, respectively. These compilers provide powerful integration capabilities and ease of use not found with other compilers.

For easier source level debugging, the compilers provide symbol information that is compatible with the MPLAB IDE memory display.
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 also link relocatable objects from pre-compiled libraries, using directives from a linker script.

The MPLIB object librarian is a librarian for pre-compiled code to be used with the MPLINK object linker. When a routine from a library is called from another 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 MPLIB object librarian manages the creation and modification of library files.

The MPLINK object linker features include:
• Integration with MPASM assembler and MPLAB C17 and MPLAB C18 C compilers.
• Allows all memory areas to be defined as sections to provide link-time flexibility.

The MPLIB object librarian features include:
• Easier linking because single libraries can be included instead of many smaller files.
• Helps keep code maintainable by grouping related modules together.
• Allows libraries to be created and modules to be added, listed, replaced, deleted or extracted.

11.5 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 of the pins. 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 the MPLAB C18 C compilers and the MPASM assembler. The software simulator offers the flexibility to develop and debug code outside of the laboratory environment, making it an excellent multi-project software development tool.

11.6 MPLAB ICE High Performance
Universal In-Circuit Emulator with
MPLAB IDE

The MPLAB ICE universal in-circuit emulator is intended to provide the product development engineer with a complete microcontroller design tool set for PICmicro microcontrollers (MCUs). Software control of the MPLAB ICE in-circuit emulator is provided by the MPLAB Integrated Development Environment (IDE), 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 in-circuit emulator system has been designed as a real-time emulation system, with advanced features that are generally found on more expensive development tools. The PC platform and Microsoft® Windows environment were chosen to best make these features available to you, the end user.

11.7 ICEPIC In-Circuit Emulator

The ICEPIC low cost, in-circuit emulator is a solution for the Microchip Technology PIC16C5X, PIC16C6X, PIC16C7X and PIC16CXXX families of 8-bit One-Time-Programmable (OTP) microcontrollers. The modular system can support different subsets of PIC16C5X or PIC16CXXX products through the use of interchangeable personality modules, or daughter boards. The emulator is capable of emulating without target application circuitry being present.
11.8 MPLAB ICD In-Circuit Debugger

Microchip's In-Circuit Debugger, MPLAB ICD, is a powerful, low cost, run-time development tool. This tool is based on the FLASH PICmicro MCUs and can be used to develop for this and other PICmicro microcontrollers. The MPLAB ICD utilizes the in-circuit debugging capability built into the FLASH devices. This feature, along with Microchip's In-Circuit Serial Programming™ 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 watching variables, single-stepping and setting break points. Running at full speed enables testing hardware in real-time.

11.9 PRO MATE II Universal Device Programmer

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

The PRO MATE II device programmer has programmable VDD and VPP supplies, which allow it to verify programmed memory at VDD min and VDD max for maximum reliability. It has an LCD display for instructions and error messages, keys to enter commands and a modular detachable socket assembly to support various package types. In stand-alone mode, the PRO MATE II device programmer can read, verify, or program PICmicro devices. It can also set code protection in this mode.

11.10 PICSTART Plus Entry Level 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 all PICmicro devices with 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.

11.11 PICDEM 1 Low Cost PICmicro Demonstration Board

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

11.12 PICDEM 2 Low Cost PIC16CXX Demonstration Board

The PICDEM 2 demonstration board is a simple demonstration board that supports the PIC16C62, PIC16C64, PIC16C65, PIC16C73 and PIC16C74 microcontrollers. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM 2 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The MPLAB ICE in-circuit emulator may also be used with the PICDEM 2 demonstration board to test firmware. A prototype area has been provided to the user for adding additional hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push button switches, a potentiometer for simulated analog input, a serial EEPROM to demonstrate usage of the I²C™ bus and separate headers for connection to an LCD module and a keypad.
11.13 PICDEM 3 Low Cost PIC16CXXX Demonstration Board

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

11.14 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. All necessary hardware is included to run basic demo programs, which are supplied on a 3.5-inch disk. A programmed sample is included and the user may erase it and program it with the other sample programs using the PRO MATE II device programmer, or the PICSTART Plus development programmer, and easily debug and test the sample code. In addition, the PICDEM 17 demonstration board supports downloading of programs to and executing out of external FLASH memory on board. The PICDEM 17 demonstration board is also usable with the MPLAB ICE in-circuit emulator, or the PICMASTER emulator and all of the sample programs can be run and modified using either emulator. Additionally, a generous prototype area is available for user hardware.

11.15 KEELOQ Evaluation and Programming Tools

KEELOQ evaluation and programming tools support Microchip’s HCS Secure Data Products. The HCS evaluation kit includes a LCD display to show changing codes, a decoder to decode transmissions and a programming interface to program test transmitters.
<table>
<thead>
<tr>
<th>PIC12CXXX</th>
<th>PIC16CXXX</th>
<th>PIC16F62X</th>
<th>PIC16C7X</th>
<th>PIC16C7XX</th>
<th>PIC16C8X</th>
<th>PIC16F8XX</th>
<th>PIC16C9XX</th>
<th>PIC17C4X</th>
<th>PIC17C7XX</th>
<th>PIC18CXX2</th>
<th>PIC18FXXX</th>
<th>24CXX/25CXX/93CXX</th>
<th>HCSXXX</th>
<th>MCRFXXX</th>
<th>MCP2510</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Software Tools</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MPLAB® Integrated Development Environment</td>
<td>MPLAB® C18 C Compiler</td>
<td>MPLAB® ICE In-Circuit Emulator</td>
<td>ICEPIC™ In-Circuit Emulator</td>
<td>MPLAB® ICD In-Circuit Debugger</td>
<td>PICSTART™ Plus Entry Level Development Programmer</td>
<td>PROMATE™II Universal Device Programmer</td>
<td>PICDEM™1 Demonstration Board</td>
<td>PICDEM™2 Demonstration Board</td>
<td>PICDEM™3 Demonstration Board</td>
<td>PICDEM™4A Demonstration Board</td>
<td>PICDEM™14A Demonstration Board</td>
<td>125 kHz Anticollision microID® Developer’s Kit</td>
<td>13.56 MHz Anticollision microID® Developer’s Kit</td>
<td>MCP2510 CAN Developer’s Kit</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Demos Boards and Eval Kits</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* Contact the Microchip Technology Inc. web site at www.microchip.com for information on how to use the MPLAB® ICD In-Circuit Debugger (DV164001) with PIC16C62, 63, 64, 65, 72, 73, 74, 76, 77.

** Contact Microchip Technology Inc. for availability date.

† Development tool is available on select devices.
12.0 ELECTRICAL SPECIFICATIONS

Absolute Maximum Ratings†

Ambient temperature under bias ........................................................................................................... -40 to +125°C
Storage temperature ........................................................................................................................ -65°C to +150°C
Voltage on VDD with respect to VSS ..................................................................................................... -0.3 to +6.5V
Voltage on MCLR with respect to VSS .................................................................................................. -0.3 to +13.5V
Voltage on all other pins with respect to VSS ........................................................................... -0.3V to (VDD + 0.3V)
Total power dissipation(1) ..................................................................................................................... 800 mW
Maximum current out of VSS pin .............................................................................................................. 300 mA
Maximum current into VDD pin .................................................................................................................... 250 mA
Input clamp current, IIK (Vi < 0 or Vi > VDD) .................................................................................... ±10 mA
Output clamp current, IOK (Vo < 0 or Vo >VDD) ..................................................................................... ±10 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 PORTC (combined) ................................................................. 200 mA
Maximum current sourced PORTA and PORTC (combined) .............................................................. 200 mA

Note 1: Power dissipation is calculated as follows: $\text{PDIS} = \text{VDD} \times (\text{IDD} - \sum \text{IOH}) + \sum (\text{VDD} - \text{VOH}) \times \text{IOH} + \sum (\text{VOL} \times \text{IOL})$.

† 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.

Note: Voltage spikes below VSS at the MCLR pin, inducing currents greater than 80 mA, may cause latchup. Thus, a series resistor of 50-100Ω should be used when applying a "low" level to the MCLR pin, rather than pulling this pin directly to VSS.
FIGURE 12-1: PIC16F630/676 WITH A/D DISABLED VOLTAGE-FREQUENCY GRAPH,
-40°C ≤ TA ≤ +85°C

Note 1: The shaded region indicates the permissible combinations of voltage and frequency.

FIGURE 12-2: PIC16F676 WITH A/D ENABLED VOLTAGE-FREQUENCY GRAPH,
-40°C ≤ TA ≤ +85°C

Note 1: The shaded region indicates the permissible combinations of voltage and frequency.
FIGURE 12-3: PIC16F676 WITH A/D ENABLED VOLTAGE-FREQUENCY GRAPH, 0°C ≤ TA ≤ +85°C

Note 1: The shaded region indicates the permissible combinations of voltage and frequency.

FIGURE 12-4: PIC16F630/676 VOLTAGE-FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C

Note 1: The shaded region indicates the permissible combinations of voltage and frequency.
12.1 DC Characteristics: PIC16F630/676-I (Industrial)

### DC CHARACTERISTICS

<table>
<thead>
<tr>
<th>Param No.</th>
<th>Sym</th>
<th>Characteristic</th>
<th>Min</th>
<th>Typ†</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
</table>
| D001      | VDD | Supply Voltage            | 2.0  | —    | 5.5  | V     | FOSC < = 4 MHz:  
            |     |                           | 2.2  | —    | 5.5  | V     | PIC16F630/676 with A/D off                                                                  |
| D001A     |     |                           | 2.5  | —    | 5.5  | V     | PIC16F676 with A/D on, 0°C to 85°C                                                            |
| D001B     |     |                           | 3.0  | —    | 5.5  | V     | PIC16F676 with A/D on, -40°C to 85°C                                                          |
| D001C     |     |                           | 4.5  | —    | 5.5  | V     | 4 MHz < FOSC < = 10 MHz                                                                    |

| D002      | VDR | RAM Data Retention Voltage(1) | 1.5* | —    | —    | V     | Device in SLEEP mode                                                                         |

| D003      | VPOR| VDD Start Voltage to ensure internal Power-on Reset signal | —    | VSS  | —    | V     | See section on Power-on Reset for details                                                     |

| D004      | SVDD| VDD Rise Rate to ensure internal Power-on Reset signal     | 0.05*| —    | —    | V/ms  | See section on Power-on Reset for details                                                     |

| D005      | VBDR| —                           | —    | 2.0  | —    | V     |                                                                                               |

| D010      | IDD | Supply Current(2,3)         | —    | 0.4  | TBD  | mA    | XT, RC osc configurations                                                                  |
|           |     |                            | —    | 0.9  | TBD  | mA    | XT, RC osc configurations                                                                  |
|           |     |                            | —    | 10   | TBD  | μA    | LP OSC configuration                                                                        |
|           |     |                            | —    | 5.2  | TBD  | mA    | HS osc configuration                                                                        |
|           |     |                            | —    | 4    | TBD  | mA    | EC OSC configuration                                                                        |
|           |     |                            | —    | 0.8  | TBD  | mA    | INTOSC OSC configuration                                                                    |

| D020      | IPD | Power Down Current(4)       | —    | 0.9  | TBD  | μA    | VDD = 2.0V, WDT disabled                                                                    |
| D021      |     |                            | —    | 130  | 250  | μA    | VDD = 5.0V, BOR enabled                                                                     |
| D022      |     |                            | —    | 8    | 20   | μA    | VDD = 2.0V, Comparator enabled                                                              |
| D023      |     |                            | —    | 1    | 3    | μA    | VDD = 2.0V, A/D on, not converting                                                          |
| D024      |     |                            | —    | 5    | 18   | μA    | VDD = 2.0V, Timer1 on, 32 kHz ext. drive                                                     |
| D025      |     |                            | —    | 80   | 200  | μA    | VDD = 2.0V, CVREF enabled                                                                  |
| D026      |     |                            | —    | 2    | 5    | μA    | VDD = 2.0V, WDT enabled                                                                     |

* These parameters are characterized but not tested.
† Data in "Typ" column is at 5.0V, 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 in SLEEP mode without losing RAM data.

**Note 2:** The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

**Note 3:** The test conditions for all IDD measurements in active Operation mode are: OSC1 = external square wave, from rail to rail; all I/O pins tri-stated, pulled to VDD; MCLR = VDD; WDT disabled.

**Note 4:** The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD.
### 12.2 DC Characteristics: PIC16F630/676-E (Extended)

<table>
<thead>
<tr>
<th>Param No.</th>
<th>Sym</th>
<th>Characteristic</th>
<th>Min</th>
<th>Typ†</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>D001A</td>
<td>VDD</td>
<td>Supply Voltage</td>
<td>4.5</td>
<td>5.5</td>
<td>—</td>
<td>V</td>
<td>-40°C to +125°C</td>
</tr>
<tr>
<td>D002</td>
<td>VDR</td>
<td>RAM Data Retention Voltage(1)</td>
<td>1.5</td>
<td>—</td>
<td>—</td>
<td>V</td>
<td>Device in SLEEP mode</td>
</tr>
<tr>
<td>D003</td>
<td>VPOR</td>
<td>VDD Start Voltage to ensure internal Power-on Reset signal</td>
<td>—</td>
<td>Vss</td>
<td>—</td>
<td>V</td>
<td>See section on Power-on Reset for details</td>
</tr>
<tr>
<td>D004</td>
<td>SVDD</td>
<td>VDD Rise Rate to ensure internal Power-on Reset signal</td>
<td>0.05</td>
<td>—</td>
<td>—</td>
<td>V/ms</td>
<td>See section on Power-on Reset for details</td>
</tr>
<tr>
<td>D005</td>
<td>VBOR</td>
<td>—</td>
<td>—</td>
<td>2.0</td>
<td>—</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>D012</td>
<td>IDD</td>
<td>Supply Current(2,3)</td>
<td>—</td>
<td>0.9</td>
<td>TBD</td>
<td>mA</td>
<td>XT, RC osc configurations FOSC = 4 MHz, VDD = 5.5V</td>
</tr>
<tr>
<td>D013</td>
<td>—</td>
<td>5.2</td>
<td>TBD</td>
<td>mA</td>
<td></td>
<td></td>
<td>FOSC = 20 MHz, VDD = 5.5V</td>
</tr>
<tr>
<td>D014</td>
<td>—</td>
<td>4</td>
<td>TBD</td>
<td>mA</td>
<td></td>
<td></td>
<td>FOSC = 20 MHz, VDD = 5.5V</td>
</tr>
<tr>
<td>D015</td>
<td>—</td>
<td>1.2</td>
<td>TBD</td>
<td>mA</td>
<td></td>
<td></td>
<td>FOSC = 20 MHz, VDD = 5.5V</td>
</tr>
<tr>
<td>D020</td>
<td>IPD</td>
<td>Power Down Current(4)</td>
<td>—</td>
<td>1</td>
<td>TBD</td>
<td>µA</td>
<td>VDD = 4.5V, WDT disabled</td>
</tr>
<tr>
<td>D021</td>
<td>—</td>
<td>130</td>
<td>250</td>
<td>µA</td>
<td></td>
<td></td>
<td>VDD = 5.0V, BOR enabled</td>
</tr>
<tr>
<td>D022</td>
<td>—</td>
<td>20</td>
<td>75</td>
<td>µA</td>
<td></td>
<td></td>
<td>VDD = 4.5V, Comparator enabled</td>
</tr>
<tr>
<td>D023</td>
<td>—</td>
<td>1</td>
<td>4</td>
<td>µA</td>
<td></td>
<td></td>
<td>VDD = 4.5V, A/D on, not converting</td>
</tr>
<tr>
<td>D024</td>
<td>—</td>
<td>15</td>
<td>60</td>
<td>µA</td>
<td></td>
<td></td>
<td>VDD = 4.5V, Timer1 on, 32 kHz ext. drive</td>
</tr>
<tr>
<td>D025</td>
<td>—</td>
<td>175</td>
<td>275</td>
<td>µA</td>
<td></td>
<td></td>
<td>VDD = 4.5V, CVREF enabled</td>
</tr>
<tr>
<td>D026</td>
<td>—</td>
<td>12</td>
<td>42</td>
<td>µA</td>
<td></td>
<td></td>
<td>VDD = 4.5V, WDT enabled</td>
</tr>
</tbody>
</table>

* These parameters are characterized but not tested.
† Data in "Typ" column is at 5.0V, 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 in SLEEP mode without losing RAM data.

**Note 2:** The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.

**Note 3:** The test conditions for all IDD measurements in active Operation mode are: OSC1 = external square wave, from rail to rail; all I/O pins tri-stated, pulled to VDD; MCLR = VDD; MCLR = VDD; WDT disabled.

**Note 4:** The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD.
12.3 DC Characteristics: PIC16F630/676-I (Industrial), PIC16F630/676-E (Extended)

**DC CHARACTERISTICS**

**Standard Operating Conditions (unless otherwise stated)**
Operating temperature: -40°C ≤ TA ≤ +85°C for industrial
-40°C ≤ TA ≤ +125°C for extended

<table>
<thead>
<tr>
<th>Param No.</th>
<th>Sym</th>
<th>Characteristic</th>
<th>Min</th>
<th>Typ†</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>D030</td>
<td>VIL</td>
<td>Input Low Voltage</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>I/O ports</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>with TTL buffer</td>
<td>Vss</td>
<td>—</td>
<td>0.8</td>
<td>V</td>
<td>4.5V ≤ VDD ≤ 5.5V</td>
</tr>
<tr>
<td>D030A</td>
<td></td>
<td>with Schmitt Trigger buffer</td>
<td>Vss</td>
<td>—</td>
<td>0.15 VDD</td>
<td>V</td>
<td>Otherwise</td>
</tr>
<tr>
<td>D031</td>
<td></td>
<td>MCLR, OSC1 (RC mode)</td>
<td>Vss</td>
<td>—</td>
<td>0.2 VDD</td>
<td>V</td>
<td>Entire range</td>
</tr>
<tr>
<td>D032</td>
<td></td>
<td>OSC1 (XT and LP modes)</td>
<td>Vss</td>
<td>—</td>
<td>0.3</td>
<td>V</td>
<td>(Note 1)</td>
</tr>
<tr>
<td>D033</td>
<td></td>
<td>OSC1 (HS mode)</td>
<td>Vss</td>
<td>—</td>
<td>0.3 VDD</td>
<td>V</td>
<td>(Note 1)</td>
</tr>
<tr>
<td>D033A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D040</td>
<td>VIH</td>
<td>Input High Voltage</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>I/O ports</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>with TTL buffer</td>
<td>2.0</td>
<td>—</td>
<td>VDD</td>
<td>V</td>
<td>4.5 V ≤ VDD ≤ 5.5 V</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.25 VDD+0.8)</td>
<td>—</td>
<td>—</td>
<td>VDD</td>
<td>V</td>
<td>otherwise</td>
</tr>
<tr>
<td></td>
<td></td>
<td>with Schmitt Trigger buffer</td>
<td>0.8VDD</td>
<td>—</td>
<td>VDD</td>
<td>V</td>
<td>entire range</td>
</tr>
<tr>
<td>D042</td>
<td></td>
<td>MCLR, RA2/AN2/T0CKI/INT/COUT</td>
<td>0.8VDD</td>
<td>—</td>
<td>VDD</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>D043</td>
<td></td>
<td>OSC1 (XT and LP modes)</td>
<td>1.6</td>
<td>—</td>
<td>VDD</td>
<td>V</td>
<td>(Note 1)</td>
</tr>
<tr>
<td>D043A</td>
<td></td>
<td>OSC1 (HS mode)</td>
<td>0.7VDD</td>
<td>—</td>
<td>VDD</td>
<td>V</td>
<td>(Note 1)</td>
</tr>
<tr>
<td>D043B</td>
<td></td>
<td>OSC1 (RC mode)</td>
<td>0.9VDD</td>
<td>—</td>
<td>VDD</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>D070</td>
<td>IPUR</td>
<td>PORTA Weak Pull-up Current</td>
<td>50*</td>
<td>250</td>
<td>400*</td>
<td>μA</td>
<td>VDD = 5.0 V, VPIN = VSS</td>
</tr>
<tr>
<td>D060</td>
<td>IIL</td>
<td>Input Leakage Current(3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>I/O ports</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Analog inputs</td>
<td></td>
<td></td>
<td>±1</td>
<td>μA</td>
<td>Vss ≤ VPIN ≤ VDD, Pin at hi-impedance</td>
</tr>
<tr>
<td>D060A</td>
<td></td>
<td>VREF</td>
<td></td>
<td></td>
<td>±TBD</td>
<td>μA</td>
<td>Vss ≤ VPIN ≤ VDD</td>
</tr>
<tr>
<td>D060B</td>
<td></td>
<td>MCLR(2)</td>
<td></td>
<td></td>
<td>±TBD</td>
<td>μA</td>
<td>Vss ≤ VPIN ≤ VDD</td>
</tr>
<tr>
<td>D061</td>
<td></td>
<td>OSC1</td>
<td></td>
<td></td>
<td>±5</td>
<td>μA</td>
<td>Vss ≤ VPIN ≤ VDD</td>
</tr>
<tr>
<td>D063</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>±5</td>
<td>μA</td>
<td>Vss ≤ VPIN ≤ VDD, XT, HS and LP osc configuration</td>
</tr>
<tr>
<td>D080</td>
<td>VOL</td>
<td>Output Low Voltage</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>I/O ports</td>
<td></td>
<td></td>
<td>0.6</td>
<td>V</td>
<td>IOL = 8.5 mA, VDD = 4.5V (Ind.)</td>
</tr>
<tr>
<td>D083</td>
<td></td>
<td>OSC2/CLKOUT (RC mode)</td>
<td></td>
<td>0.6</td>
<td>V</td>
<td>IOL = 1.6 mA, VDD = 4.5V (Ind.) IOL = 1.2 mA, VDD = 4.5V (Ext.)</td>
<td></td>
</tr>
<tr>
<td>D090</td>
<td>VOH</td>
<td>Output High Voltage</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>I/O ports</td>
<td></td>
<td></td>
<td>0.7VDD</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>D092</td>
<td></td>
<td>OSC2/CLKOUT (RC mode)</td>
<td>0.7VDD</td>
<td>—</td>
<td>—</td>
<td>V</td>
<td>IOH = -1.3 mA, VDD = 4.5V (Ind.) IOH = -1.0 mA, VDD = 4.5V (Ext.)</td>
</tr>
</tbody>
</table>

* These parameters are characterized but not tested.
† Data in 'Typ' column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**Note 1:** In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended to use an external clock in RC mode.

**Note 2:** The leakage current on the MCLR 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.

**Note 3:** Negative current is defined as current sourced by the pin.
### DC CHARACTERISTICS

#### Capacitive Loading Specs on Output Pins

<table>
<thead>
<tr>
<th>Param No.</th>
<th>Sym</th>
<th>Characteristic</th>
<th>Min</th>
<th>Typ†</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>D100</td>
<td>C0SC2</td>
<td>OSC2 pin</td>
<td>—</td>
<td>—</td>
<td>15*</td>
<td>pF</td>
<td>In XT, HS and LP modes when external clock is used to drive OSC1</td>
</tr>
<tr>
<td>D101</td>
<td>ClO</td>
<td>All I/O pins</td>
<td>—</td>
<td>—</td>
<td>50*</td>
<td>pF</td>
<td></td>
</tr>
</tbody>
</table>

#### Data EEPROM Memory

<table>
<thead>
<tr>
<th>Param No.</th>
<th>Sym</th>
<th>Characteristic</th>
<th>Min</th>
<th>Typ†</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>D120</td>
<td>ED</td>
<td>Cell Endurance</td>
<td>100K</td>
<td>1M</td>
<td>—</td>
<td>E/W</td>
<td>-40°C ≤ TA ≤ +85°C</td>
</tr>
<tr>
<td>D121</td>
<td>VDRW</td>
<td>VDD for read/write</td>
<td>—</td>
<td>5.5</td>
<td>V</td>
<td></td>
<td>VMIN = Minimum operating voltage</td>
</tr>
</tbody>
</table>

#### Program FLASH Memory

<table>
<thead>
<tr>
<th>Param No.</th>
<th>Sym</th>
<th>Characteristic</th>
<th>Min</th>
<th>Typ†</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>D130</td>
<td>EP</td>
<td>Endurance</td>
<td>10K</td>
<td>100K</td>
<td>—</td>
<td>E/W</td>
<td>-40°C ≤ TA ≤ +85°C</td>
</tr>
<tr>
<td>D131</td>
<td>VPR</td>
<td>VDD for read</td>
<td>VMIN</td>
<td>—</td>
<td>5.5</td>
<td>V</td>
<td>VMIN = Minimum operating voltage</td>
</tr>
</tbody>
</table>

* These parameters are characterized but not tested.
† Data in ‘Typ’ column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.
12.4 TIMING PARAMETER SYMBOLOGY

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

1. TppS2ppS
2. TppS

<table>
<thead>
<tr>
<th>T</th>
<th>Frequency</th>
<th>T</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Lowercase letters (pp) and their meanings:

<table>
<thead>
<tr>
<th>pp</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>cc</td>
<td>CCP1</td>
<td></td>
</tr>
<tr>
<td>ck</td>
<td>CLKOUT</td>
<td></td>
</tr>
<tr>
<td>cs</td>
<td>CS</td>
<td></td>
</tr>
<tr>
<td>di</td>
<td>SDI</td>
<td></td>
</tr>
<tr>
<td>do</td>
<td>SDO</td>
<td></td>
</tr>
<tr>
<td>dt</td>
<td>Data in</td>
<td></td>
</tr>
<tr>
<td>io</td>
<td>I/O port</td>
<td></td>
</tr>
<tr>
<td>mc</td>
<td>MCLR</td>
<td></td>
</tr>
</tbody>
</table>

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>osc</td>
<td>OSC1</td>
<td></td>
</tr>
<tr>
<td>rd</td>
<td>RD</td>
<td></td>
</tr>
<tr>
<td>rw</td>
<td>RD or WR</td>
<td></td>
</tr>
<tr>
<td>sc</td>
<td>SCK</td>
<td></td>
</tr>
<tr>
<td>ss</td>
<td>SS</td>
<td></td>
</tr>
<tr>
<td>t0</td>
<td>T0CKI</td>
<td></td>
</tr>
<tr>
<td>t1</td>
<td>T1CKI</td>
<td></td>
</tr>
<tr>
<td>wr</td>
<td>WR</td>
<td></td>
</tr>
</tbody>
</table>

Uppercase letters and their meanings:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td></td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>Fall</td>
<td></td>
</tr>
<tr>
<td>H</td>
<td>High</td>
<td></td>
</tr>
<tr>
<td>I</td>
<td>Invalid (Hi-impedance)</td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>Low</td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>Period</td>
<td></td>
</tr>
<tr>
<td>R</td>
<td>Rise</td>
<td></td>
</tr>
<tr>
<td>V</td>
<td>Valid</td>
<td></td>
</tr>
<tr>
<td>Z</td>
<td>Hi-impedance</td>
<td></td>
</tr>
</tbody>
</table>

FIGURE 12-5: LOAD CONDITIONS

Load Condition 1

![Load Condition 1 Diagram]

Load Condition 2

![Load Condition 2 Diagram]

\[ RL = 464\Omega \]

\[ CL = 50 \, \text{pF} \quad \text{for all pins} \]

\[ 15 \, \text{pF} \quad \text{for OSC2 output} \]
### 12.5 AC CHARACTERISTICS: PIC16F630/676 (INDUSTRIAL, EXTENDED)

#### FIGURE 12-6: EXTERNAL CLOCK TIMING

![External Clock Timing Diagram]

#### TABLE 12-1: EXTERNAL CLOCK TIMING REQUIREMENTS

<table>
<thead>
<tr>
<th>Param No.</th>
<th>Sym</th>
<th>Characteristic</th>
<th>Min</th>
<th>Typ†</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Fosc</td>
<td>External CLIN Frequency(1)</td>
<td>DC</td>
<td>—</td>
<td>32</td>
<td>kHz</td>
<td>LP Osc mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>DC</td>
<td>4</td>
<td>20</td>
<td>MHz</td>
<td>XT mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>DC</td>
<td>20</td>
<td>20</td>
<td>MHz</td>
<td>HS mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>DC</td>
<td>20</td>
<td>20</td>
<td>MHz</td>
<td>EC mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Oscillator Frequency(1)</td>
<td>5</td>
<td>—</td>
<td>32</td>
<td>kHz</td>
<td>LP Osc mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>—</td>
<td>4</td>
<td>MHz</td>
<td>INTOSC mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TBD</td>
<td>—</td>
<td>4</td>
<td>MHz</td>
<td>RC Osc mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0.1</td>
<td>—</td>
<td>4</td>
<td>MHz</td>
<td>XT Osc mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>—</td>
<td>20</td>
<td>MHz</td>
<td>HS Osc mode</td>
</tr>
<tr>
<td>1</td>
<td>Tosc</td>
<td>External CLIN Period(1)</td>
<td>32</td>
<td>—</td>
<td>∞</td>
<td>µs</td>
<td>LP Osc mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>50</td>
<td>—</td>
<td>∞</td>
<td>ns</td>
<td>HS Osc mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>50</td>
<td>—</td>
<td>∞</td>
<td>ns</td>
<td>EC Osc mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>250</td>
<td>—</td>
<td>∞</td>
<td>ns</td>
<td>XT Osc mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Oscillator Period(1)</td>
<td>30</td>
<td>—</td>
<td>200</td>
<td>µs</td>
<td>LP Osc mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>250</td>
<td>—</td>
<td>TBD</td>
<td>ns</td>
<td>INTOSC mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>250</td>
<td>—</td>
<td>10,000</td>
<td>ns</td>
<td>RC Osc mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>250</td>
<td>—</td>
<td>10,000</td>
<td>ns</td>
<td>XT Osc mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>50</td>
<td>—</td>
<td>1,000</td>
<td>ns</td>
<td>HS Osc mode</td>
</tr>
<tr>
<td>2</td>
<td>Tcy</td>
<td>Instruction Cycle Time(1)</td>
<td>200</td>
<td>Tcy</td>
<td>∞</td>
<td>ns</td>
<td>Tcy = 4/Fosc</td>
</tr>
<tr>
<td>3</td>
<td>TosL, TosH</td>
<td>External CLIN (OSC1) High</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>µs</td>
<td>LP oscillator, Tosc L/H duty cycle</td>
</tr>
<tr>
<td></td>
<td></td>
<td>External CLIN Low</td>
<td>2*</td>
<td>—</td>
<td>—</td>
<td>ns</td>
<td>HS oscillator, Tosc L/H duty cycle</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>20*</td>
<td>—</td>
<td>—</td>
<td>ns</td>
<td>XT oscillator, Tosc L/H duty cycle</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>100 *</td>
<td>—</td>
<td>—</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>TosR, TosF</td>
<td>External CLIN Rise</td>
<td>—</td>
<td>—</td>
<td>50*</td>
<td>ns</td>
<td>LP oscillator</td>
</tr>
<tr>
<td></td>
<td></td>
<td>External CLIN Fall</td>
<td>—</td>
<td>—</td>
<td>25*</td>
<td>ns</td>
<td>XT oscillator</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>—</td>
<td>—</td>
<td>15*</td>
<td>ns</td>
<td>HS oscillator</td>
</tr>
</tbody>
</table>

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

**Note 1:** Instruction cycle period (Tcy) equals four times the input oscillator time-base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at ‘min’ values with an external clock applied to OSC1 pin. When an external clock input is used, the ‘max’ cycle time limit is ‘DC’ (no clock) for all devices.
### TABLE 12-2: CALIBRATED INTERNAL OSCILLATOR

**AC Characteristics**

<table>
<thead>
<tr>
<th>Param No.</th>
<th>Sym</th>
<th>Characteristic</th>
<th>Min*</th>
<th>Typ(1)</th>
<th>Max*</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal Calibrated Oscillator Frequency</td>
<td>3.92</td>
<td>4.00</td>
<td>4.08</td>
<td>MHz</td>
<td>VDD = 3.75V, +85°C (Ind.)</td>
<td>VDD = 3.75V, +125°C (Ext.)</td>
<td></td>
</tr>
<tr>
<td>Internal Calibrated Oscillator Frequency</td>
<td>3.80</td>
<td>4.00</td>
<td>4.20</td>
<td>MHz</td>
<td>2.0V ≤ VDD ≤ 5.5V</td>
<td>-40°C ≤ TA ≤ +85°C (Ind.)</td>
<td>-40°C ≤ TA ≤ +125°C (Ext.)</td>
</tr>
<tr>
<td>Internal Oscillator Start-Up Time</td>
<td>—</td>
<td>8</td>
<td>—</td>
<td>TOSC</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* These parameters are characterized but not tested.

**Note 1:** Data in the Typical ("Typ") column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.
FIGURE 12-7: CLKOUT AND I/O TIMING

TABLE 12-3: CLKOUT AND I/O TIMING REQUIREMENTS

<table>
<thead>
<tr>
<th>Param No.</th>
<th>Sym</th>
<th>Characteristic</th>
<th>Min</th>
<th>Typ†</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>TosH2ckL</td>
<td>OSC1↑ to CLKOUT↓</td>
<td></td>
<td>75</td>
<td>200</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>TosH2ckH</td>
<td>OSC1↑ to CLKOUT↑</td>
<td></td>
<td>75</td>
<td>200</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>TckR</td>
<td>CLKOUT rise time</td>
<td></td>
<td>35</td>
<td>100</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>TckF</td>
<td>CLKOUT fall time</td>
<td></td>
<td>35</td>
<td>100</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>TckL2ioV</td>
<td>CLKOUT↓ to Port out valid</td>
<td></td>
<td>—</td>
<td>20</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>TioV2ckH</td>
<td>Port in valid before CLKOUT↑</td>
<td></td>
<td>—</td>
<td>—</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>TckH2ioI</td>
<td>Port in hold after CLKOUT↑</td>
<td></td>
<td>0</td>
<td>—</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>TosH2ioV</td>
<td>OSC1↑ (Q1 cycle) to Port out valid</td>
<td></td>
<td>—</td>
<td>50</td>
<td>150*</td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>—</td>
<td>300</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>TosH2iol</td>
<td>OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)</td>
<td>100</td>
<td>—</td>
<td>—</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>TioV2osH</td>
<td>Port input valid to OSC1↑ (I/O in setup time)</td>
<td>0</td>
<td>—</td>
<td>—</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>TioR</td>
<td>Port output rise time</td>
<td></td>
<td>10</td>
<td>40</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>TioF</td>
<td>Port output fall time</td>
<td></td>
<td>10</td>
<td>40</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>Tinp</td>
<td>INT pin high or low time</td>
<td></td>
<td>25</td>
<td>—</td>
<td>ns</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>Trbp</td>
<td>PORTA change INT high or low time</td>
<td></td>
<td>TCY</td>
<td>—</td>
<td>ns</td>
<td></td>
</tr>
</tbody>
</table>

* These parameters are characterized but not tested.
† Data in ‘Typ’ column is at 5.0V, 25°C unless otherwise stated.

Note 1: Measurements are taken in RC mode where CLKOUT output is 4xTosc.
FIGURE 12-8: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING

FIGURE 12-9: BROWN-OUT RESET TIMING AND CHARACTERISTICS

Note 1: 72 ms delay only if PWRT bit in configuration word is programmed to ‘0’.
<table>
<thead>
<tr>
<th>Param No.</th>
<th>Sym</th>
<th>Characteristic</th>
<th>Min</th>
<th>Typ†</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>TMCL</td>
<td>MCLR Pulse Width (low)</td>
<td>2 TBD</td>
<td>— TBD</td>
<td>— TBD</td>
<td>µs ms</td>
<td>VDD = 5V, -40°C to +85°C Extended temperature</td>
</tr>
<tr>
<td>31</td>
<td>TWDT</td>
<td>Watchdog Timer Time-out Period (No Prescaler)</td>
<td>7* TBD</td>
<td>18 TBD</td>
<td>33* TBD</td>
<td>ms ms</td>
<td>VDD = 5V, -40°C to +85°C Extended temperature</td>
</tr>
<tr>
<td>32</td>
<td>TOST</td>
<td>Oscillation Start-up Timer Period</td>
<td>—</td>
<td>1024Tosc</td>
<td>—</td>
<td>—</td>
<td>Tosc = OSC1 period</td>
</tr>
<tr>
<td>33*</td>
<td>TPWRT</td>
<td>Power-up Timer Period</td>
<td>28* TBD</td>
<td>72 TBD</td>
<td>132* TBD</td>
<td>ms ms</td>
<td>VDD = 5V, -40°C to +85°C Extended Temperature</td>
</tr>
<tr>
<td>34</td>
<td>TIOZ</td>
<td>I/O Hi-impedance from MCLR Low or Watchdog Timer Reset</td>
<td>—</td>
<td>—</td>
<td>2.0</td>
<td>µs</td>
<td></td>
</tr>
<tr>
<td>BVDD</td>
<td></td>
<td>Brown-out Reset Voltage</td>
<td>2.0</td>
<td>2.1</td>
<td>V</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Brown-out Hysteresis</td>
<td>TBD</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>TBOR</td>
<td>Brown-out Reset Pulse Width</td>
<td>100* —</td>
<td>—</td>
<td>µs</td>
<td>VDD ≤ BVDD (D005)</td>
<td></td>
</tr>
</tbody>
</table>

* 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 12-10: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS

TABLE 12-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

<table>
<thead>
<tr>
<th>Param No.</th>
<th>Sym</th>
<th>Characteristic</th>
<th>Min</th>
<th>Typ†</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>40*</td>
<td>T0H</td>
<td>T0CKI High Pulse Width</td>
<td>No Prescaler</td>
<td>0.5 TCY + 20</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>With Prescaler</td>
<td>10</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>41*</td>
<td>T0L</td>
<td>T0CKI Low Pulse Width</td>
<td>No Prescaler</td>
<td>0.5 TCY + 20</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>With Prescaler</td>
<td>10</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>42*</td>
<td>T0P</td>
<td>T0CKI Period</td>
<td>Greater of: 20 or TCY + 40 (N)</td>
<td>—</td>
<td>—</td>
<td>ns</td>
<td>(N = \text{prescale value (2, 4, ..., 256)})</td>
</tr>
<tr>
<td>45*</td>
<td>T1H</td>
<td>T1CKI High Time</td>
<td>Synchronous, No Prescaler</td>
<td>0.5 TCY + 20</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Synchronous, with Prescaler</td>
<td>15</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Asynchronous</td>
<td>30</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>46*</td>
<td>T1L</td>
<td>T1CKI Low Time</td>
<td>Synchronous, No Prescaler</td>
<td>0.5 TCY + 20</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Synchronous, with Prescaler</td>
<td>15</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Asynchronous</td>
<td>30</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td>47*</td>
<td>T1P</td>
<td>T1CKI Input Period</td>
<td>Synchronous</td>
<td>Greater of: 30 or TCY + 40 (N)</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Asynchronous</td>
<td>60</td>
<td>—</td>
<td>—</td>
<td>ns</td>
</tr>
<tr>
<td></td>
<td>F1</td>
<td>Timer1 oscillator input frequency range (oscillator enabled by setting bit T1OSCEN)</td>
<td>DC</td>
<td>—</td>
<td>200*</td>
<td>kHz</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>T0CKE2Tmr1</td>
<td>Delay from external clock edge to timer increment</td>
<td>2 Tosc*</td>
<td>—</td>
<td>7 Tosc*</td>
<td>—</td>
<td></td>
</tr>
</tbody>
</table>

* These parameters are characterized but not tested.
† Data in ‘Typ’ column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.
### TABLE 12-6: COMPARATOR SPECIFICATIONS

| Sym  | Characteristics                     | Standard Operating Conditions  
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>-40°C to +125°C (unless otherwise stated)</td>
</tr>
<tr>
<td>VOS</td>
<td>Input Offset Voltage</td>
<td>Min</td>
</tr>
<tr>
<td></td>
<td></td>
<td>—</td>
</tr>
<tr>
<td>VCM</td>
<td>Input Common Mode Voltage</td>
<td>0</td>
</tr>
<tr>
<td>CMRR</td>
<td>Common Mode Rejection Ratio</td>
<td>+55*</td>
</tr>
<tr>
<td>TRT</td>
<td>Response Time&lt;sup&gt;(1)&lt;/sup&gt;</td>
<td>—</td>
</tr>
</tbody>
</table>
| TMC2COV | Comparator Mode Change to   
| Output Valid | | | | | | |

* These parameters are characterized but not tested.

**Note 1:** Response time measured with one comparator input at (VDD - 1.5)/2 while the other input transitions from VSS to VDD - 1.5V.

### TABLE 12-7: COMPARATOR VOLTAGE REFERENCE SPECIFICATIONS

| Sym  | Characteristics                     | Standard Operating Conditions  
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>-40°C to +125°C (unless otherwise stated)</td>
</tr>
<tr>
<td></td>
<td>Resolution</td>
<td>Min</td>
</tr>
<tr>
<td></td>
<td></td>
<td>—</td>
</tr>
<tr>
<td></td>
<td></td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>Absolute Accuracy</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td></td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>Unit Resistor Value (R)</td>
<td>—</td>
</tr>
<tr>
<td></td>
<td>Settling Time&lt;sup&gt;(1)&lt;/sup&gt;</td>
<td>—</td>
</tr>
</tbody>
</table>

* These parameters are characterized but not tested.

**Note 1:** Settling time measured while VRR = 1 and VR<3:0> transitions from 0000 to 1111.
## TABLE 12-8: PIC16F676 A/D CONVERTER CHARACTERISTICS:

<table>
<thead>
<tr>
<th>Param No.</th>
<th>Sym</th>
<th>Characteristic</th>
<th>Min</th>
<th>Typ†</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>A01</td>
<td>NR</td>
<td>Resolution</td>
<td>—</td>
<td>—</td>
<td>10</td>
<td>bits</td>
<td></td>
</tr>
<tr>
<td>A02</td>
<td>EABS</td>
<td>Total Absolute Error*</td>
<td>—</td>
<td>—</td>
<td>TBD</td>
<td>LSb</td>
<td>VREF = 3.0V</td>
</tr>
<tr>
<td>A03</td>
<td>EIL</td>
<td>Integral Error</td>
<td>—</td>
<td>—</td>
<td>TBD</td>
<td>LSb</td>
<td>VREF = 3.0V</td>
</tr>
<tr>
<td>A04</td>
<td>EDL</td>
<td>Differential Error</td>
<td>—</td>
<td>—</td>
<td>TBD</td>
<td>LSb</td>
<td>No missing codes to 10 bits VREF = 3.0V</td>
</tr>
<tr>
<td>A05</td>
<td>EFS</td>
<td>Full Scale Range</td>
<td>2.2*</td>
<td>—</td>
<td>5.5*</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>A06</td>
<td>EOF</td>
<td>Offset Error</td>
<td>—</td>
<td>—</td>
<td>TBD</td>
<td>LSb</td>
<td>VREF = 3.0V</td>
</tr>
<tr>
<td>A07</td>
<td>EGN</td>
<td>Gain Error</td>
<td>—</td>
<td>—</td>
<td>TBD</td>
<td>LSb</td>
<td>VREF = 3.0V</td>
</tr>
<tr>
<td>A10</td>
<td>—</td>
<td>Monotonicity</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>—</td>
<td>VSS ≤ VAIN ≤ VREF+</td>
</tr>
<tr>
<td>A20</td>
<td>VREF</td>
<td>Reference voltage</td>
<td>0</td>
<td>—</td>
<td>—</td>
<td>V</td>
<td>2.0</td>
</tr>
<tr>
<td>A20A</td>
<td>—</td>
<td>Reference V High (VDD or VREF)</td>
<td>VSS</td>
<td>—</td>
<td>VDD</td>
<td>V</td>
<td>Absolute minimum to ensure 10-bit accuracy</td>
</tr>
<tr>
<td>A25</td>
<td>VAIN</td>
<td>Analog Input Voltage</td>
<td>VSS</td>
<td>—</td>
<td>VREF</td>
<td>V</td>
<td></td>
</tr>
<tr>
<td>A30</td>
<td>Z Ain</td>
<td>Recommended Impedance of Analog Voltage Source</td>
<td>—</td>
<td>—</td>
<td>2.5</td>
<td>kΩ</td>
<td></td>
</tr>
<tr>
<td>A50</td>
<td>IREF</td>
<td>VREF Input Current(2)</td>
<td>10</td>
<td>—</td>
<td>1000</td>
<td>µA</td>
<td>During VAIN acquisition. Based on differential of VHOLD to VAIN.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>—</td>
<td>10</td>
<td>µA</td>
<td>During A/D conversion cycle.</td>
</tr>
</tbody>
</table>

* These parameters are characterized but not tested.
† Data in ‘Typ’ column is at 5.0V, 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 leakage current. The power-down current spec includes any such leakage from the A/D module.

**Note 2:** VREF current is from External VREF or VDD pin, whichever is selected as reference input.

**Note 3:** The A/D conversion result never decreases with an increase in the input voltage and has no missing codes.
FIGURE 12-11: PIC16F676 A/D CONVERSION TIMING (NORMAL MODE)

TABLE 12-9: PIC16F676 A/D CONVERSION REQUIREMENTS

<table>
<thead>
<tr>
<th>Param No.</th>
<th>Sym</th>
<th>Characteristic</th>
<th>Min</th>
<th>Typ†</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>130 TAD</td>
<td>A/D Clock Period</td>
<td>1.6</td>
<td>—</td>
<td>—</td>
<td>μs</td>
<td>TOSC based, VREF ≥ 3.0V</td>
<td></td>
</tr>
<tr>
<td>130 TAD</td>
<td>A/D Internal Oscillator Period</td>
<td>3.0*</td>
<td>—</td>
<td>—</td>
<td>μs</td>
<td>TOSC based, VREF full range</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.0*</td>
<td>6.0</td>
<td>9.0*</td>
<td>μs</td>
<td>ADCS&lt;1:0&gt; = 11 (RC mode)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>μs</td>
<td>At VDD = 2.5V</td>
</tr>
<tr>
<td>131 TCNV</td>
<td>Conversion Time (not including</td>
<td>—</td>
<td>11</td>
<td>—</td>
<td>TAD</td>
<td>Set GO bit to new data in A/D result register</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Acquisition Time)(1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>132 TACQ</td>
<td>Acquisition Time</td>
<td>(Note 2)</td>
<td>11.5</td>
<td>—</td>
<td>—</td>
<td>μs</td>
<td>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., 4.1 mV @ 4.096 V) from the last sampled voltage (as stored on CHOLD).</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>5*</td>
<td></td>
<td>—</td>
<td>μs</td>
<td></td>
</tr>
<tr>
<td>134 TGO</td>
<td>Q4 to A/D Clock Start</td>
<td>—</td>
<td>TOSC/2</td>
<td>—</td>
<td>—</td>
<td>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.</td>
<td></td>
</tr>
</tbody>
</table>

* These parameters are characterized but not tested.
† Data in ‘Typ’ column is at 5.0V, 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 7.1 for minimum conditions.
FIGURE 12-12: PIC16F676 A/D CONVERSION TIMING (SLEEP MODE)

TABLE 12-10: PIC16F676 A/D CONVERSION REQUIREMENTS (SLEEP MODE)

<table>
<thead>
<tr>
<th>Param No.</th>
<th>Sym</th>
<th>Characteristic</th>
<th>Min</th>
<th>Typ†</th>
<th>Max</th>
<th>Units</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>130</td>
<td>TAD</td>
<td>A/D Clock Period</td>
<td>1.6</td>
<td>—</td>
<td>—</td>
<td>µs</td>
<td>VREF ≥ 3.0V</td>
</tr>
<tr>
<td>130</td>
<td>TAD</td>
<td>A/D Internal Oscillator Period</td>
<td>3.0*</td>
<td>6.0</td>
<td>9.0*</td>
<td>µs</td>
<td>VREF full range</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2.0*</td>
<td>4.0</td>
<td>6.0*</td>
<td>µs</td>
<td>ADCS&lt;1:0&gt; = 11 (RC mode)</td>
</tr>
<tr>
<td>131</td>
<td>TCNV</td>
<td>Conversion Time (not including Acquisition Time)</td>
<td>—</td>
<td>11</td>
<td>—</td>
<td>TADC</td>
<td>At VDD = 2.5V</td>
</tr>
<tr>
<td>132</td>
<td>TACQ</td>
<td>Acquisition Time</td>
<td>—</td>
<td>11.5</td>
<td>—</td>
<td>µs</td>
<td>At VDD = 5.0V</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Note 2)</td>
<td>5*</td>
<td>—</td>
<td>—</td>
<td>µs</td>
<td>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., 4.1 mV @ 4.096V) from the last sampled voltage (as stored on CHOLD).</td>
</tr>
<tr>
<td>134</td>
<td>TGO</td>
<td>Q4 to A/D Clock Start</td>
<td>—</td>
<td>TOSC/2 + TCY</td>
<td>—</td>
<td>—</td>
<td>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.</td>
</tr>
</tbody>
</table>

* These parameters are characterized but not tested.
† Data in ‘Typ’ column is at 5.0V, 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.
Note 2: See Section 7.1 for minimum conditions.
13.0 PACKAGING INFORMATION

13.1 Package Marking Information

14-Lead PDIP (Skinny DIP)

Example

16F630-I

0215/017

14-Lead SOIC

Example

16F630-E

0215/017

14-Lead TSSOP

Example

16F630

0215

017

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.
13.2 Package Details

The following sections give the technical details of the packages.

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

---

**Units** | INCHES | MILLIMETERS
--- | --- | ---
Dimension Limits | MIN | NOM | MAX | MIN | NOM | MAX
Number of Pins | \(n\) | 14 | 14
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 | 3.66
Base to Seating Plane | \(A1\) | .015 | .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 | \(D\) | .740 | .750 | .760 | 18.80 | 19.05 | 19.30
Tip to Seating Plane | \(L\) | .125 | .130 | .135 | 3.18 | 3.30 | 3.43
Lead Thickness | \(c\) | .008 | .012 | .015 | .20 | .29 | .38
Upper Lead Width | \(B1\) | .045 | .058 | .070 | 1.14 | 1.46 | 1.78
Lower Lead Width | \(B\) | .014 | .018 | .022 | .36 | .46 | .56
Overall Row Spacing | \(\varepsilon B\) | .310 | .370 | .430 | 7.87 | 9.40 | 10.92
Mold Draft Angle Top | \(\alpha\) | 5 | 10 | 15 | 5 | 10 | 15
Mold Draft Angle Bottom | \(\beta\) | 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 \(.010\) \((0.254\text{mm})\) per side.
JEDEC Equivalent: MS-001
Drawing No. C04-005
14-Lead Plastic Small Outline (SL) – Narrow, 150 mil (SOIC)

<table>
<thead>
<tr>
<th>Units</th>
<th>INCHES*</th>
<th>MILLIMETERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Pins</td>
<td>n</td>
<td>14</td>
</tr>
<tr>
<td>Pitch</td>
<td>P</td>
<td>.050</td>
</tr>
<tr>
<td>Overall Height</td>
<td>A</td>
<td>.053 .061 .069</td>
</tr>
<tr>
<td>Molded Package Thickness</td>
<td>A2</td>
<td>.052 .056 .061</td>
</tr>
<tr>
<td>Standoff §</td>
<td>A1</td>
<td>.004 .007 .010</td>
</tr>
<tr>
<td>Overall Width</td>
<td>E</td>
<td>.226 .236 .244</td>
</tr>
<tr>
<td>Molded Package Width</td>
<td>E1</td>
<td>.150 .154 .157</td>
</tr>
<tr>
<td>Overall Length</td>
<td>D</td>
<td>.337 .342 .347</td>
</tr>
<tr>
<td>Chamfer Distance</td>
<td>h</td>
<td>.010 .015 .020</td>
</tr>
<tr>
<td>Foot Length</td>
<td>L</td>
<td>.016 .033 .050</td>
</tr>
<tr>
<td>Foot Angle</td>
<td>φ</td>
<td>.0 .4 .8</td>
</tr>
<tr>
<td>Lead Thickness</td>
<td>c</td>
<td>.008 .009 .010</td>
</tr>
<tr>
<td>Lead Width</td>
<td>B</td>
<td>.014 .017 .020</td>
</tr>
<tr>
<td>Mold Draft Angle Top</td>
<td>α</td>
<td>0 12 15</td>
</tr>
<tr>
<td>Mold Draft Angle Bottom</td>
<td>β</td>
<td>0 12 15</td>
</tr>
</tbody>
</table>

* 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-012
Drawing No. C04-065
PIC16F630/676

14-Lead Plastic Thin Shrink Small Outline (ST) – 4.4 mm (TSSOP)

<table>
<thead>
<tr>
<th>Units</th>
<th>INCHES</th>
<th>MILLIMETERS*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimension Limits</td>
<td>MIN</td>
<td>NOM</td>
</tr>
<tr>
<td>Number of Pins</td>
<td>n</td>
<td>14</td>
</tr>
<tr>
<td>Pitch</td>
<td>P</td>
<td>0.026</td>
</tr>
<tr>
<td>Overall Height</td>
<td>A</td>
<td>.043</td>
</tr>
<tr>
<td>Molded Package Thickness</td>
<td>A2</td>
<td>.033</td>
</tr>
<tr>
<td>Standoff §</td>
<td>A1</td>
<td>.002</td>
</tr>
<tr>
<td>Overall Width</td>
<td>E</td>
<td>.246</td>
</tr>
<tr>
<td>Molded Package Width</td>
<td>E1</td>
<td>.169</td>
</tr>
<tr>
<td>Molded Package Length</td>
<td>D</td>
<td>.193</td>
</tr>
<tr>
<td>Foot Length</td>
<td>L</td>
<td>.020</td>
</tr>
<tr>
<td>Foot Angle</td>
<td>φ</td>
<td>0</td>
</tr>
<tr>
<td>Lead Thickness</td>
<td>c</td>
<td>.004</td>
</tr>
<tr>
<td>Lead Width</td>
<td>B</td>
<td>.007</td>
</tr>
<tr>
<td>Mold Draft Angle Top</td>
<td>α</td>
<td>0</td>
</tr>
<tr>
<td>Mold Draft Angle Bottom</td>
<td>β</td>
<td>0</td>
</tr>
</tbody>
</table>

* Controlling Parameter
§ Significant Characteristic

Notes:
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .005’ (0.127mm) per side.
JEDEC Equivalent: MO-153
Drawing No. C04-087
APPENDIX A: DATA SHEET

Revision A

This is a new data sheet.

APPENDIX B: DEVICE DIFFERENCES

The differences between the PIC16F630/676 devices listed in this data sheet are shown in Table B-1.

<table>
<thead>
<tr>
<th>Feature</th>
<th>PIC16F630</th>
<th>PIC16F676</th>
</tr>
</thead>
<tbody>
<tr>
<td>A/D</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
APPENDIX C: DEVICE MIGRATIONS

This section is intended to describe the functional and electrical specification differences when migrating between functionally similar devices (such as from a PIC16C74A to a PIC16C74B).

Not Applicable

APPENDIX D: MIGRATING FROM OTHER PICmicro® DEVICES

This discusses some of the issues in migrating from other PICmicro devices to the PIC12F6XX family of devices.

D.1 PIC16C505 to PIC16F630/676

See Microchip website for availability (www.microchip.com).

D.2 PIC12F6XX to PIC16F630/676

See Microchip website for availability (www.microchip.com).

Note: 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.
APPENDIX E: DEVELOPMENT TOOL VERSION REQUIREMENTS

This lists the minimum requirements (software/firmware) of the specified development tool to support the devices listed in this data sheet.

MPLAB® IDE: TBD
MPLAB® SIMULATOR: TBD

MPLAB® ICE 3000:
- PIC16F630/676 Processor Module:
  - Part Number: TBD
- PIC16F630/676 Device Adapter:
  - Socket Part Number
  - 14-pin SOIC: TBD
  - 14-pin PDIP: TBD
  - 14-pin TSSOP: TBD

MPLAB® ICD: TBD
PRO MATE® II: TBD
PICSTART® Plus: TBD
MPASM™ Assembler: TBD

Note: Please read all associated README.TXT files that are supplied with the development tools. These "read me" files will discuss product support and any known limitations.
INDEX

A
A/D ................................................................. 43
Acquisition Requirements .............................. 47
Block Diagram ................................................. 43
Calculating Acquisition Time ............................ 47
Configuration and Operation ......................... 43
Effects of a RESET ........................................... 48
Internal Sampling Switch (RSS) Impedance ........ 47
Operation During SLEEP ................................. 48
PIC16F676 Converter Characteristics ............... 100
Source Impedance .......................................... 47
Summary of Registers ...................................... 48
Absolute Maximum Ratings ......................... 85
AC Characteristics
Industrial and Extended .................................. 93
Analog Input Connection Considerations .......... 40
Analog-to-Digital Converter, See A/D
Assembler
MPASM Assembler ........................................ 79
B
Block Diagram
TM0/WDT Prescaler ........................................ 29
Block Diagrams
Analog Input Mode ......................................... 40
Analog Input Model ......................................... 47
Comparator Output ......................................... 40
Comparator Voltage Reference ....................... 41
On-Chip Reset Circuit ..................................... 57
RA0 and RA1 Pins .......................................... 22
RA2 ............................................................. 23
RA3 ............................................................. 23
RA4 ............................................................. 24
RA5 ............................................................. 24
RC Oscillator Mode ........................................ 56
RC0/RC1/RC2/RC3 Pins .................................. 26
RC4 AND RC5 Pins ........................................ 26
Timer1 ......................................................... 32
Watchdog Timer ............................................. 68
Brown-out
Associated Registers ...................................... 60
Brown-out Detect (BOD) ................................. 58
Brown-out Reset Timing and Characteristics .... 96
C
Calibrated Internal Oscillator Frequencies ........... 94
CLKOUT .................................................... 56
Code Examples
Changing Prescaler ........................................ 31
Data EEPROM Read ....................................... 51
Data EEPROM Write ..................................... 51
Initializing PORTA ........................................ 19
Initializing PORTC ........................................ 26
Saving STATUS and W Registers in RAM .......... 67
Write Verly .................................................. 51
Code Protection ............................................ 70
Comparator ................................................. 37
Associated Registers ...................................... 42
Configuration ............................................... 39
Effects of a RESET ........................................ 41
I/O Operating Modes ..................................... 39
Interrupts ..................................................... 42
Operation ..................................................... 38
Operation During SLEEP ............................... 41
Output ......................................................... 40
Reference ..................................................... 41
Response Time ............................................. 41
Comparator Specifications ............................. 99
Comparator Voltage Reference Specifications .... 99
Configuration Bits ......................................... 74
Configuring the Voltage Reference .................. 41
Crystal Operation ......................................... 55
D
Data EEPROM Memory
Associated Registers/Bits ............................... 52
Code Protection ............................................ 52
EEADR Register .......................................... 49
EECON1 Register ......................................... 49
EECON2 Register ......................................... 49
EEDATA Register ......................................... 49
Data Memory Organization ............................. 7
DC Characteristics
Extended .................................................... 89
Extended and Industrial .................................. 90
Industrial .................................................... 88
Development Support .................................... 79
Development Tool Version Requirements ........ 109
Device Differences ........................................ 107
Device Migrations ......................................... 108
Device Overview .......................................... 5
E
EEPROM Data Memory
Reading ....................................................... 51
Spurious Write ............................................. 51
Write Verly .................................................. 51
Writing ....................................................... 51
Electrical Specifications ................................. 85
F
Firmware Instructions ..................................... 71
G
General Purpose Register File ......................... 7
I
ICEPIC In-Circuit Emulator ............................ 80
ID Locations ................................................ 70
In-Circuit Serial Programming ....................... 70
Indirect Addressing, INDF and FSR Registers .... 18
Instruction Format ......................................... 71
Instruction Set ............................................. 71
ADDLW ..................................................... 73
ADDWF ..................................................... 73
ANDLW ..................................................... 73
ANDWF ..................................................... 73
BCF ......................................................... 73
BSF ......................................................... 73
BTFS ....................................................... 73
BTFF ....................................................... 73
CALL ....................................................... 74
CLRF ....................................................... 74
CLRW ....................................................... 74
CLRWDT .................................................... 74
COMF ....................................................... 74
DECF ....................................................... 74
DECFSZ .................................................... 75
GOTO ....................................................... 75
INCF ....................................................... 75
INCFSZ .................................................... 75
IORLW ..................................................... 75
IORWF ..................................................... 75
MOVF ....................................................... 76
MOVLW ..................................................... 76
© 2002 Microchip Technology Inc.
MOVWF ................................................................. 76
NOP ................................................................. 76
RETIE ............................................................ 76
RETURN .......................................................... 77
SLEEP .............................................................. 77
SWAPF ............................................................. 78
XORLW ............................................................ 78
XORWF ............................................................ 78
Summary Table ..................................................... 72

Internal 4 MHz Oscillator ........................................ 56
Internal Sampling Switch (Rss) Impedance ................. 47

Interrupts ............................................................... 64
A/D Converter ...................................................... 66
Comparator ........................................................ 66
Context Saving ...................................................... 67
PORTA .............................................................. 66
RA2/INT ............................................................. 66
Summary of Registers ........................................... 67
TMRO ............................................................. 66

K
KEELoQ Evaluation and Programming Tools .................. 82

M
MCLR ................................................................. 58
Memory Organization
Data EEPROM Memory ........................................... 49
Migrating from other PICmicro Devices ..................... 108
MPLAB C17 and MPLAB C18 C Compilers ..................... 79
MPLAB ICE High Performance Universal In-Circuit Emulator with MPLAB IIDE ........................................... 80
MPLAB Integrated Development Environment Software ....... 79
MPLINK Object Linker/MPLIB Object Librarian .............. 80

O
OPCODE Field Descriptions ..................................... 71
Oscillator Configurations ........................................ 55
Oscillator Start-up Timer (OST) .................................. 58

P
Packaging ................................................................ 103
Details .................................................................... 104
Marking .................................................................. 103
PCL and PCLATH ................................................... 17
Computed GOTO .................................................... 17
Stack ..................................................................... 17
PICDEM 1 Low Cost PICmicro Demonstration Board ...... 81
PICDEM 17 Demonstration Board ............................... 82
PICDEM 2 Low Cost PIC16CXX Demonstration Board ...... 81
PICDEM 3 Low Cost PIC16CXX Demonstration Board ... 82
PICSTART Plus Entry Level Development Programmer .... 81

Pin Descriptions
PIC16F630 .......................................................... 6
PIC16F676 .......................................................... 6
PORTA
Additional Pin Functions ........................................ 19
Interrupt-on-Change ............................................. 21
Weak Pull-up ....................................................... 19
Associated Registers ........................................... 25
Pin Descriptions and Diagrams ................................. 22
PORTA and TRISA Registers ................................... 19
PORTC .............................................................. 27

Associated Registers .................................................. 27
Power Control/Status Register (PCON) ......................... 59
Power-Down Mode (SLEEP) .................................... 69
Power-on Reset (POR) ............................................ 58
Power-up Timer (PWRT) .......................................... 58
Prescaler .............................................................. 31
Switching Prescaler Assignment ................................. 31
PRO MATE II Universal Device Programmer ................ 81
Program Memory Organization ..................................... 7
Programming, Device Instructions .................................. 71

R
RC Oscillator .......................................................... 56
READ-MODIFY-WRITE OPERATIONS ...................... 71

Registers
ADCON0 (A/D Control) ........................................... 45
ADCON1 ............................................................. 45
CMCON (Comparator Control) .................................... 37
CONFIG (Configuration Word) ................................ 54
EEDAT (EEPROM Address) ..................................... 49
EECON1 (EEPROM Control) ..................................... 50
EEDAT (EEPROM Data) ........................................... 49
INTCON (Interrupt Control) ..................................... 13
IOCA (Interrupt-on-Change PORTA) ......................... 21
Maps
PIC16F630 ....................................................... 8
PIC16F676 ....................................................... 8
OPTION_REG (Option) ........................................... 12, 30
OSCCL ............................................................. 16
OSCCAL (Oscillator Calibration) ............................... 16
PIE1 (Peripheral Interrupt Enable 1) ......................... 14
PIR1 (Peripheral Interrupt 1) .................................... 15
PORTC .............................................................. 27
STATUS .......................................................... 11
T1CON (Timer1 Control) ......................................... 34
TRISC ............................................................. 27
VRCON (Voltage Reference Control) ......................... 42
WPUA (Weak Pull-up PORTA) .................................. 20

RESET ............................................................. 57
Revision History ..................................................... 107

S
Software Simulator (MPLAB SIM) ............................ 80
Special Features of the CPU ..................................... 53
Special Function Registers ........................................ 8

T
Time-out Sequence ................................................ 59
Timer0
Associated Registers ............................................. 31
External Clock ................................................... 30
Interrupt .......................................................... 29
Operation ........................................................ 29
TOCKI .......................................................... 30
Timer1
Associated Registers ............................................. 35
Asynchronous Counter Mode ................................... 35
Reading and Writing ............................................. 35
Capacitor Selection ............................................... 35
Interrupt .......................................................... 33
Modes of Operations ............................................. 33
Operation During SLEEP ....................................... 35
Oscillator .......................................................... 35
Prescaler ........................................................... 33
Timer1 Module with Gate Control ....................... 32
Timing Diagrams
CLKOUT and I/O ............................................... 95
External Clock .................................................... 93
INT Pin Interrupt.......................................................... 66
PIC16F676 A/D Conversion (Normal Mode) .......... 101
PIC16F676 A/D Conversion Timing
(SLEEP Mode) .......................................................... 102
RESET, Watchdog Timer, Oscillator Start-up Timer and
Power-up Timer .......................................................... 96
Time-out Sequence on Power-up (MCLR not Tied to
VDD)/
   Case 1 ................................................................. 62
   Case 2 ................................................................. 62
Time-out Sequence on Power-up
(MCLR Tied to VDD) .................................................. 62
     Timer0 and Timer1 External Clock ..................... 98
     Timer1 Incrementing Edge ................................. 33
Timing Parameter Symbology ................................. 92
TRISA Registers ...................................................... 19

V
Voltage Reference Accuracy/Error ....................... 41

W
Watchdog Timer
   Summary of Registers ............................................. 68
   Watchdog Timer (WDT) ......................................... 67
   WWW, On-Line Support ........................................... 3
ON-LINE SUPPORT

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

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

Connecting to the Microchip Internet Web Site

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

www.microchip.com

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

ftp://ftp.microchip.com

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

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

Systems Information and Upgrade Hot Line

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

1-800-755-2345 for U.S. and most of Canada, and
1-480-792-7302 for the rest of the world.
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 Data Sheet.

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: PIC16F630/676
Literature Number: DS40039A

Questions:

1. What are the best features of this document?

________________________________________________________________________

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

________________________________________________________________________

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

________________________________________________________________________

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

________________________________________________________________________

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

________________________________________________________________________

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

________________________________________________________________________

7. How would you improve this document?

________________________________________________________________________

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

________________________________________________________________________
PRODUCT IDENTIFICATION SYSTEM

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

<table>
<thead>
<tr>
<th>PART NO.</th>
<th>X</th>
<th>XX</th>
<th>XXX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device</td>
<td>Temperature Range</td>
<td>Package</td>
<td>Pattern</td>
</tr>
</tbody>
</table>

- **Device**
  - PIC16F6XX: Standard V DD range 2.0V to 5.5V
  - PIC16F6XXT: V DD range 2.0V to 5.5V (Tape and Reel)

- **Temperature Range**
  - I = -40°C to +85°C
  - E = -40°C to +125°C

- **Package**
  - P = PDIP
  - SN = SOIC (Gull Wing, 150 mil body)
  - ST = TSSOP (4.4mm)

- **Pattern**
  - 3-Digit Pattern Code for QTP (blank otherwise).

Examples:
- a) PIC16F630 - E/P 301 = Extended Temp., PDIP package, 20 MHz, QTP pattern #301.
- b) PIC16F676 - I/SO = Industrial Temp., SOIC package, 20 MHz.

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)

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

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