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

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

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contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates.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

**Trademarks**

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

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

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

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

All other trademarks mentioned herein are property of their respective companies.

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

Printed on recycled paper.

DNV Certification, Inc. USA — ANSI-RAB QMS ACCREDITED — DNV — DNV MSC The Netherlands Accredited by the RvA — ISO 9001 / QS-9000 REGISTERED FIRM

*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.*



© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

# PIC16C62X

## EPROM-Based 8-Bit CMOS Microcontrollers

### Devices included in this data sheet:

Referred to collectively as PIC16C62X.

- PIC16C620
- PIC16C620A
- PIC16C621
- PIC16C621A
- PIC16C622
- PIC16C622A
- PIC16CR620A

### 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 - 40 MHz clock input
  - DC - 100 ns instruction cycle

| Device | Program Memory | Data Memory |
|---|---|---|
| PIC16C620 | 512 | 80 |
| PIC16C620A | 512 | 96 |
| PIC16CR620A | 512 | 96 |
| PIC16C621 | 1K | 80 |
| PIC16C621A | 1K | 96 |
| PIC16C622 | 2K | 128 |
| PIC16C622A | 2K | 128 |

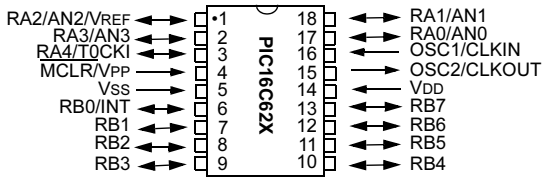
- Interrupt capability
- 16 special function hardware registers
- 8-level deep hardware stack
- Direct, Indirect and Relative addressing modes

### Peripheral Features:

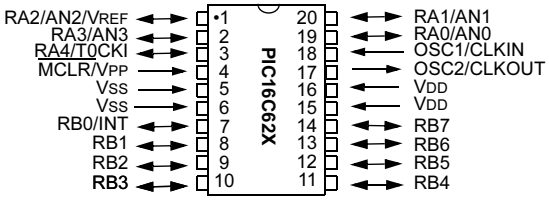
- 13 I/O pins with individual direction control
- High current sink/source for direct LED drive
- Analog comparator module with:
  - Two analog comparators
  - Programmable on-chip voltage reference ($V_{REF}$) module
  - Programmable input multiplexing from device inputs and internal voltage reference
  - Comparator outputs can be output signals
- Timer0: 8-bit timer/counter with 8-bit programmable prescaler

### Pin Diagrams

**PDIP, SOIC, Windowed CERDIP**




**SSOP**




### Special Microcontroller Features:

- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Brown-out Reset
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Programmable code protection
- Power saving SLEEP mode
- Selectable oscillator options
- Serial in-circuit programming (via two pins)
- Four user programmable ID locations

### CMOS Technology:

- Low power, high speed CMOS EPROM technology
- Fully static design
- Wide operating range
  - 2.5V to 5.5V
- Commercial, industrial and extended temperature range
- Low power consumption
  - < 2.0 mA @ 5.0V, 4.0 MHz
  - 15 μA typical @ 3.0V, 32 kHz
  - < 1.0 μA typical standby current @ 3.0V

© 2003 Microchip Technology Inc.


Downloaded from Arrow.com.

## Device Differences

| Device | Voltage Range | Oscillator | Process Technology (Microns) |
|---|---|---|---|
| PIC16C620(3) | 2.5 - 6.0 | See **Note 1** | 0.9 |
| PIC16C621(3) | 2.5 - 6.0 | See **Note 1** | 0.9 |
| PIC16C622(3) | 2.5 - 6.0 | See **Note 1** | 0.9 |
| PIC16C620A(4) | 2.7 - 5.5 | See **Note 1** | 0.7 |
| PIC16CR620A(2) | 2.5 - 5.5 | See **Note 1** | 0.7 |
| PIC16C621A(4) | 2.7 - 5.5 | See **Note 1** | 0.7 |
| PIC16C622A(4) | 2.7 - 5.5 | See **Note 1** | 0.7 |

**Note 1:** If you change from this device to another device, please verify oscillator characteristics in your application.

**2:** For ROM parts, operation from 2.5V - 3.0V will require the PIC16LCR62X parts.

**3:** For OTP parts, operation from 2.5V - 3.0V will require the PIC16LC62X parts.

**4:** For OTP parts, operations from 2.7V - 3.0V will require the PIC16LC62XA parts.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

**Table of Contents**

1.0 General Description ..... 5
2.0 PIC16C62X Device Varieties ..... 7
3.0 Architectural Overview ..... 9
4.0 Memory Organization ..... 13
5.0 I/O Ports ..... 25
6.0 Timer0 Module ..... 31
7.0 Comparator Module ..... 37
8.0 Voltage Reference Module ..... 43
9.0 Special Features of the CPU ..... 45
10.0 Instruction Set Summary ..... 61
11.0 Development Support ..... 75
12.0 Electrical Specifications ..... 81
13.0 Device Characterization Information ..... 109
14.0 Packaging Information ..... 113
Appendix A: Enhancements ..... 119
Appendix B: Compatibility ..... 119
Index ..... 121
On-Line Support ..... 123
Systems Information and Upgrade Hot Line ..... 123
Reader Response ..... 124
Product Identification System ..... 125

## 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.



Downloaded from Arrow.com.

**NOTES:**


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 1.0 GENERAL DESCRIPTION

The PIC16C62X devices are 18 and 20-Pin ROM/EPROM-based members of the versatile PICmicro® family of low cost, high performance, CMOS, fully-static, 8-bit microcontrollers.

All PICmicro microcontrollers employ an advanced RISC architecture. The PIC16C62X devices have enhanced core features, eight-level deep stack, and multiple internal and external interrupt sources. The separate instruction and data buses of the Harvard architecture allow a 14-bit wide instruction word with the separate 8-bit wide data. The two-stage instruction pipeline allows all instructions to execute in a single cycle, except for program branches (which require two cycles). A total of 35 instructions (reduced instruction set) are available. Additionally, a large register set gives some of the architectural innovations used to achieve a very high performance.

PIC16C62X microcontrollers typically achieve a 2:1 code compression and a 4:1 speed improvement over other 8-bit microcontrollers in their class.

The PIC16C620A, PIC16C621A and PIC16CR620A have 96 bytes of RAM. The PIC16C622(A) has 128 bytes of RAM. Each device has 13 I/O pins and an 8-bit timer/counter with an 8-bit programmable prescaler. In addition, the PIC16C62X adds two analog comparators with a programmable on-chip voltage reference module. The comparator module is ideally suited for applications requiring a low cost analog interface (e.g., battery chargers, threshold detectors, white goods controllers, etc).

PIC16C62X devices have special features to reduce external components, thus reducing system cost, enhancing system reliability and reducing power consumption. There are four oscillator options, of which the single pin RC oscillator provides a low cost solution, the LP oscillator minimizes power consumption, XT is a standard crystal, and the HS is for High Speed crystals. The SLEEP (Power-down) mode offers power savings. The user can wake-up the chip from SLEEP through several external and internal interrupts and RESET.

A highly reliable Watchdog Timer with its own on-chip RC oscillator provides protection against software lock- up.

A UV-erasable CERDIP-packaged version is ideal for code development while the cost effective One-Time-Programmable (OTP) version is suitable for production in any volume.

Table 1-1 shows the features of the PIC16C62X mid-range microcontroller families.

A simplified block diagram of the PIC16C62X is shown in Figure 3-1.

The PIC16C62X series fits perfectly in applications ranging from battery chargers to low power remote sensors. The EPROM technology makes customization of application programs (detection levels, pulse generation, timers, etc.) extremely fast and convenient. The small footprint packages make this microcontroller series perfect for all applications with space limitations. Low cost, low power, high performance, ease of use and I/O flexibility make the PIC16C62X very versatile.

### 1.1 Family and Upward Compatibility

Those users familiar with the PIC16C5X family of microcontrollers will realize that this is an enhanced version of the PIC16C5X architecture. Please refer to Appendix A for a detailed list of enhancements. Code written for the PIC16C5X can be easily ported to PIC16C62X family of devices (Appendix B). The PIC16C62X family fills the niche for users wanting to migrate up from the PIC16C5X family and not needing various peripheral features of other members of the PIC16XX mid-range microcontroller family.

### 1.2 Development Support

The PIC16C62X family is supported by a full-featured macro assembler, a software simulator, an in-circuit emulator, a low cost development programmer and a full-featured programmer. Third Party “C” compilers are also available.



Downloaded from Arrow.com.

**TABLE 1-1: PIC16C62X FAMILY OF DEVICES**

| | | PIC16C620[3] | PIC16C620A[1][4] | PIC16CR620A[2] | PIC16C621[3] | PIC16C621A[1][4] | PIC16C622[3] | PIC16C622A[1][4] |
|---|---|---|---|---|---|---|---|---|
| Clock | Maximum Frequency of Operation (MHz) | 20 | 40 | 20 | 20 | 40 | 20 | 40 |
| Memory | EPROM Program Memory (x14 words) | 512 | 512 | 512 | 1K | 1K | 2K | 2K |
| | Data Memory (bytes) | 80 | 96 | 96 | 80 | 96 | 128 | 128 |
| Peripherals | Timer Module(s) | TMR0 | TMR0 | TMRO | TMR0 | TMR0 | TMR0 | TMR0 |
| | Comparators(s) | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| | Internal Reference Voltage | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Features | Interrupt Sources | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| | I/O Pins | 13 | 13 | 13 | 13 | 13 | 13 | 13 |
| | Voltage Range (Volts) | 2.5-6.0 | 2.7-5.5 | 2.5-5.5 | 2.5-6.0 | 2.7-5.5 | 2.5-6.0 | 2.7-5.5 |
| | Brown-out Reset | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| | Packages | 18-pin DIP, SOIC; 20-pin SSOP | 18-pin DIP, SOIC; 20-pin SSOP | 18-pin DIP, SOIC; 20-pin SSOP | 18-pin DIP, SOIC; 20-pin SSOP | 18-pin DIP, SOIC; 20-pin SSOP | 18-pin DIP, SOIC; 20-pin SSOP | 18-pin DIP, SOIC; 20-pin SSOP |

All PICmicro® Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability. All PIC16C62X Family devices use serial programming with clock pin RB6 and data pin RB7.

**Note 1:** If you change from this device to another device, please verify oscillator characteristics in your application.

**2:** For ROM parts, operation from 2.0V - 2.5V will require the PIC16LCR62XA parts.

**3:** For OTP parts, operation from 2.5V - 3.0V will require the PIC16LC62X part.

**4:** For OTP parts, operation from 2.7V - 3.0V will require the PIC16LC62XA part.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 2.0 PIC16C62X DEVICE VARIETIES

A variety of frequency ranges and packaging options are available. Depending on application and production requirements, the proper device option can be selected using the information in the PIC16C62X Product Identification System section at the end of this data sheet. When placing orders, please use this page of the data sheet to specify the correct part number.

### 2.1 UV Erasable Devices

The UV erasable version, offered in CERDIP package, is optimal for prototype development and pilot programs. This version can be erased and reprogrammed to any of the Oscillator modes.

Microchip's PICSTART® and PRO MATE® programmers both support programming of the PIC16C62X.

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

### 2.2 One-Time-Programmable (OTP) Devices

The availability of OTP devices is especially useful for customers who need the flexibility for frequent code updates and small volume applications. In addition to the program memory, the configuration bits must also be programmed.

### 2.3 Quick-Turnaround-Production (QTP) Devices

Microchip offers a QTP programming service for factory production orders. This service is made available for users who chose not to program a medium to high quantity of units and whose code patterns have stabilized. The devices are identical to the OTP devices, but with all EPROM locations and configuration options already programmed by the factory. Certain code and prototype verification procedures apply before production shipments are available. Please contact your Microchip Technology sales office for more details.

### 2.4 Serialized Quick-Turnaround-Production℠ (SQTP℠) Devices

Microchip offers a unique programming service where a few user-defined locations in each device are programmed with different serial numbers. The serial numbers may be random, pseudo-random or sequential.

Serial programming allows each device to have a unique number, which can serve as an entry-code, password or ID number.



Downloaded from Arrow.com.

**NOTES:**


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 3.0 ARCHITECTURAL OVERVIEW

The high performance of the PIC16C62X family can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC16C62X uses a Harvard architecture, in which, program and data are accessed from separate memories using separate busses. This improves bandwidth over traditional von Neumann architecture, where program and data are fetched from the same memory. Separating program and data memory further allows instructions to be sized differently than 8-bit wide data word. Instruction opcodes are 14-bits wide making it possible to have all single word instructions. A 14-bit wide program memory access bus fetches a 14-bit instruction in a single cycle. A two-stage pipeline overlaps fetch and execution of instructions. Consequently, all instructions (35) execute in a single cycle (200 ns @ 20 MHz) except for program branches.

The PIC16C620(A) and PIC16CR620A address 512 x 14 on-chip program memory. The PIC16C621(A) addresses 1K x 14 program memory. The PIC16C622(A) addresses 2K x 14 program memory. All program memory is internal.

The PIC16C62X can directly or indirectly address its register files or data memory. All special function registers including the program counter are mapped in the data memory. The PIC16C62X has an orthogonal (symmetrical) instruction set that makes it possible to carry out any operation on any register using any Addressing mode. This symmetrical nature and lack of ‘special optimal situations’ make programming with the PIC16C62X simple yet efficient. In addition, the learning curve is reduced significantly.

The PIC16C62X devices contain an 8-bit ALU and working register. The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between data in the working register and any register file.

The ALU is 8-bits wide and capable of addition, subtraction, shift and logical operations. Unless otherwise mentioned, arithmetic operations are two's complement in nature. In two-operand instructions, typically one operand is the working register (W register). The other operand is a file register or an immediate constant. In single operand instructions, the operand is either the W register or a file register.

The W register is an 8-bit working register used for ALU operations. It is not an addressable register.

Depending on the instruction executed, the ALU may affect the values of the Carry (C), Digit Carry (DC), and Zero (Z) bits in the STATUS register. The C and DC bits operate as a Borrow and Digit Borrow out bit, respectively, bit in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

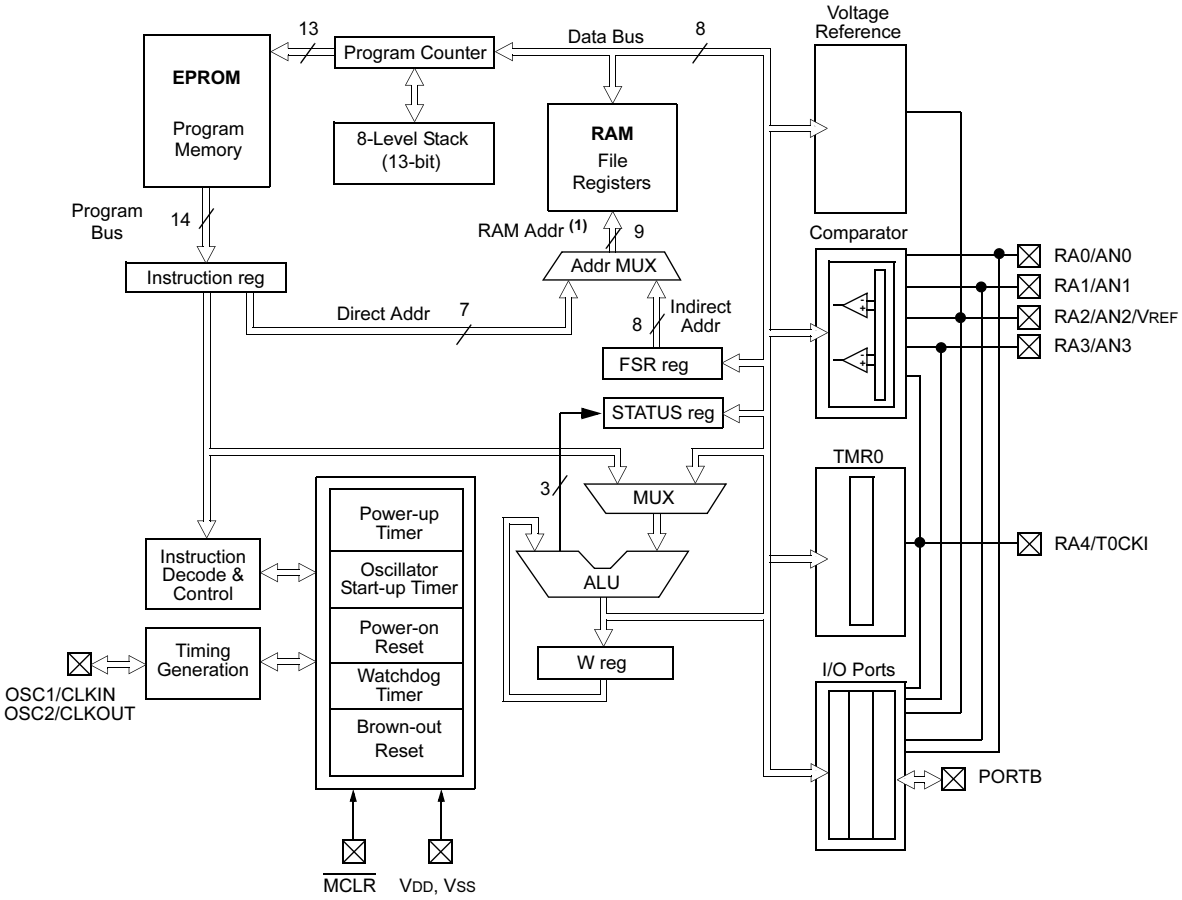
A simplified block diagram is shown in Figure 3-1, with a description of the device pins in Table 3-1.



Downloaded from Arrow.com.

**FIGURE 3-1: BLOCK DIAGRAM**

| Device | Program Memory | Data Memory (RAM) |
|---|---|---|
| PIC16C620 | 512 x 14 | 80 x 8 |
| PIC16C620A | 512 x 14 | 96 x 8 |
| PIC16CR620A | 512 x 14 | 96 x 8 |
| PIC16C621 | 1K x 14 | 80 x 8 |
| PIC16C621A | 1K x 14 | 96 x 8 |
| PIC16C622 | 2K x 14 | 128 x 8 |
| PIC16C622A | 2K x 14 | 128 x 8 |




**Note 1:** Higher order bits are from the STATUS register.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

**TABLE 3-1: PIC16C62X PINOUT DESCRIPTION**

| Name | DIP/SOIC Pin # | SSOP Pin # | I/O/P Type | Buffer Type | Description |
|---|---|---|---|---|---|
| OSC1/CLKIN | 16 | 18 | I | ST/CMOS | Oscillator crystal input/external clock source input. |
| OSC2/CLKOUT | 15 | 17 | O | — | Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. In RC mode, OSC2 pin outputs CLKOUT, which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate. |
| $\overline{MCLR}$/VPP | 4 | 4 | I/P | ST | Master Clear (Reset) input/programming voltage input. This pin is an Active Low Reset to the device. |
| | | | | | PORTA is a bi-directional I/O port. |
| RA0/AN0 | 17 | 19 | I/O | ST | Analog comparator input |
| RA1/AN1 | 18 | 20 | I/O | ST | Analog comparator input |
| RA2/AN2/VREF | 1 | 1 | I/O | ST | Analog comparator input or VREF output |
| RA3/AN3 | 2 | 2 | I/O | ST | Analog comparator input /output |
| RA4/T0CKI | 3 | 3 | I/O | ST | Can be selected to be the clock input to the Timer0 timer/counter or a comparator output. Output is open drain type. |
| | | | | | PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs. |
| RB0/INT | 6 | 7 | I/O | TTL/ST(1) | RB0/INT can also be selected as an external interrupt pin. |
| RB1 | 7 | 8 | I/O | TTL | |
| RB2 | 8 | 9 | I/O | TTL | |
| RB3 | 9 | 10 | I/O | TTL | |
| RB4 | 10 | 11 | I/O | TTL | Interrupt-on-change pin. |
| RB5 | 11 | 12 | I/O | TTL | Interrupt-on-change pin. |
| RB6 | 12 | 13 | I/O | TTL/ST(2) | Interrupt-on-change pin. Serial programming clock. |
| RB7 | 13 | 14 | I/O | TTL/ST(2) | Interrupt-on-change pin. Serial programming data. |
| VSS | 5 | 5,6 | P | — | Ground reference for logic and I/O pins. |
| VDD | 14 | 15,16 | P | — | Positive supply for logic and I/O pins. |

Legend: O = output I/O = input/output P = power
— = Not used I = Input ST = Schmitt Trigger input
TTL = TTL input

**Note 1:** This buffer is a Schmitt Trigger input when configured as the external interrupt.
**2:** This buffer is a Schmitt Trigger input when used in Serial Programming mode.



Downloaded from Arrow.com.

## 3.1 Clocking Scheme/Instruction Cycle

The clock input (OSC1/CLKIN pin) is internally divided by four to generate four non-overlapping quadrature clocks namely Q1, Q2, Q3 and Q4. Internally, the program counter (PC) is incremented every Q1, the instruction is fetched from the program memory and latched into the instruction register in Q4. The instruction is decoded and executed during the following Q1 through Q4. The clocks and instruction execution flow is shown in Figure 3-2.

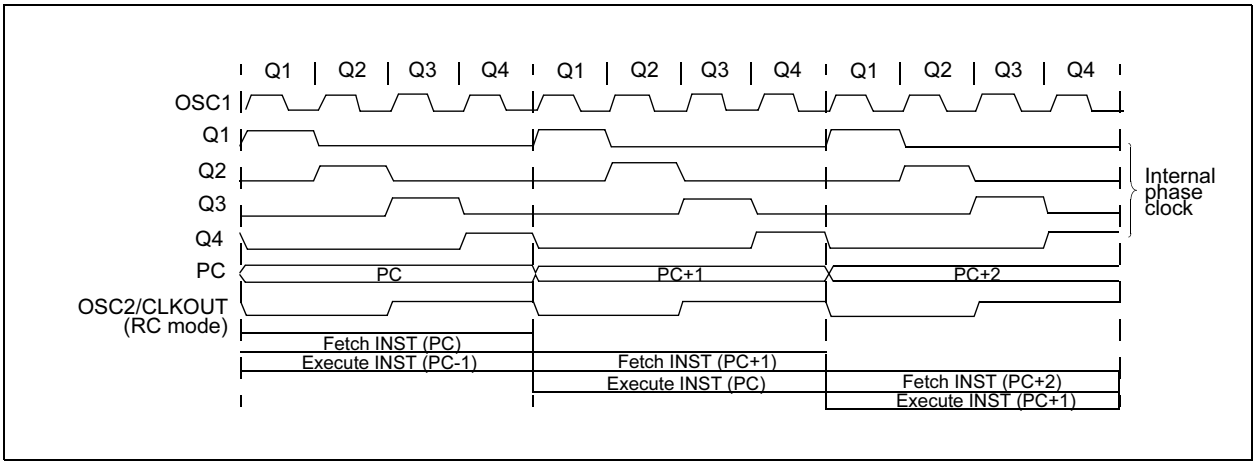
## 3.2 Instruction Flow/Pipelining

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

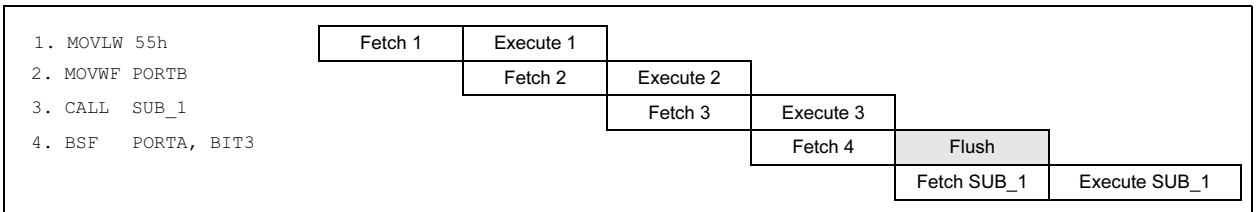
A fetch cycle begins with the program counter (PC) incrementing in Q1.

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

**FIGURE 3-2: CLOCK/INSTRUCTION CYCLE**




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




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


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

# 4.0 MEMORY ORGANIZATION

## 4.1 Program Memory Organization

The PIC16C62X has a 13-bit program counter capable of addressing an 8K x 14 program memory space. Only the first 512 x 14 (0000h - 01FFh) for the PIC16C620(A) and PIC16CR620, 1K x 14 (0000h - 03FFh) for the PIC16C621(A) and 2K x 14 (0000h - 07FFh) for the PIC16C622(A) are physically implemented. Accessing a location above these boundaries will cause a wrap-around within the first 512 x 14 space (PIC16C(R)620(A)) or 1K x 14 space (PIC16C621(A)) or 2K x 14 space (PIC16C622(A)). The RESET vector is at 0000h and the interrupt vector is at 0004h (Figure 4-1, Figure 4-2, Figure 4-3).

**FIGURE 4-1: PROGRAM MEMORY MAP AND STACK FOR THE PIC16C620/PIC16C620A/ PIC16CR620A**

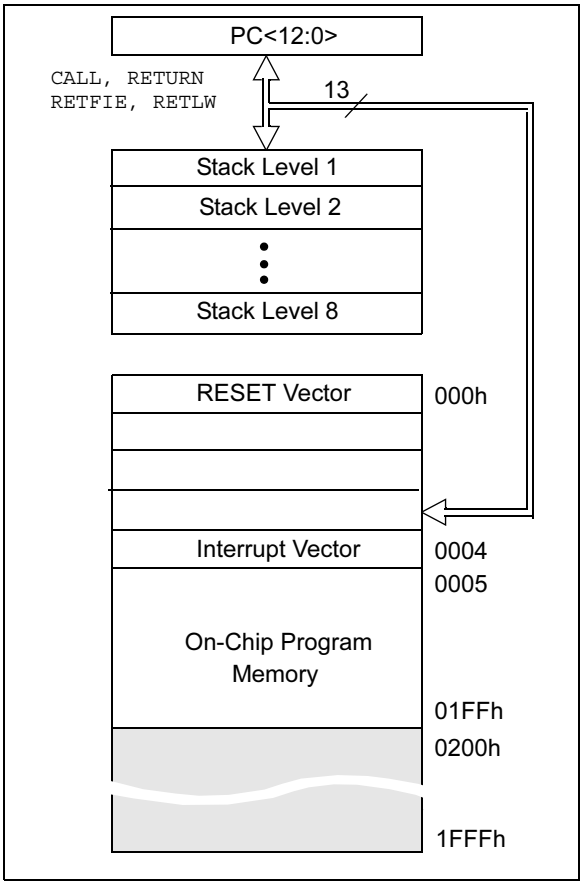



**FIGURE 4-2: PROGRAM MEMORY MAP AND STACK FOR THE PIC16C621/PIC16C621A**

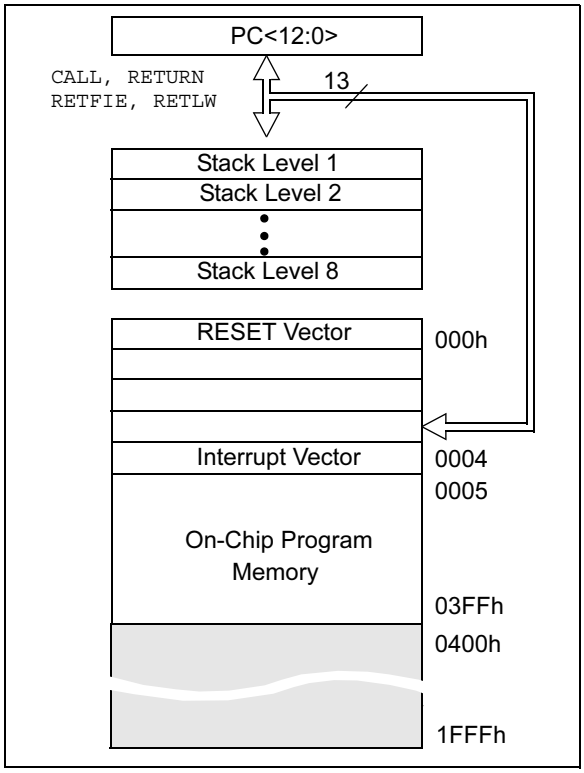
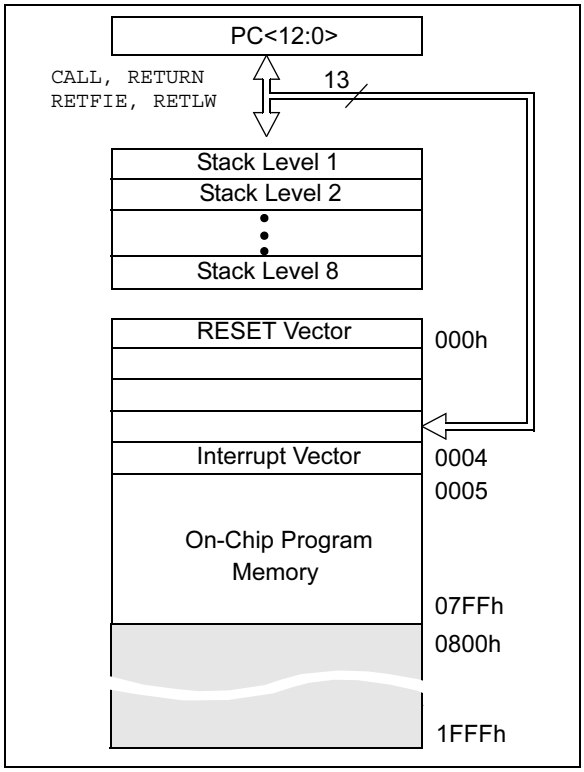



**FIGURE 4-3: PROGRAM MEMORY MAP AND STACK FOR THE PIC16C622/PIC16C622A**






Downloaded from Arrow.com.

## 4.2 Data Memory Organization

The data memory (Figure 4-4, Figure 4-5, Figure 4-6 and Figure 4-7) is partitioned into two banks, which contain the General Purpose Registers and the Special Function Registers. Bank 0 is selected when the RP0 bit is cleared. Bank 1 is selected when the RP0 bit (STATUS <5>) is set. The Special Function Registers are located in the first 32 locations of each bank. Register locations 20-7Fh (Bank0) on the PIC16C620A/CR620A/621A and 20-7Fh (Bank0) and A0-BFh (Bank1) on the PIC16C622 and PIC16C622A are General Purpose Registers implemented as static RAM. Some Special Purpose Registers are mapped in Bank 1.

Addresses F0h-FFh of bank1 are implemented as common ram and mapped back to addresses 70h-7Fh in bank0 on the PIC16C620A/621A/622A/CR620A.

### 4.2.1 GENERAL PURPOSE REGISTER FILE

The register file is organized as 80 x 8 in the PIC16C620/621, 96 x 8 in the PIC16C620A/621A/CR620A and 128 x 8 in the PIC16C622(A). Each is accessed either directly or indirectly through the File Select Register FSR (Section 4.4).


© 2003 Microchip Technology Inc.

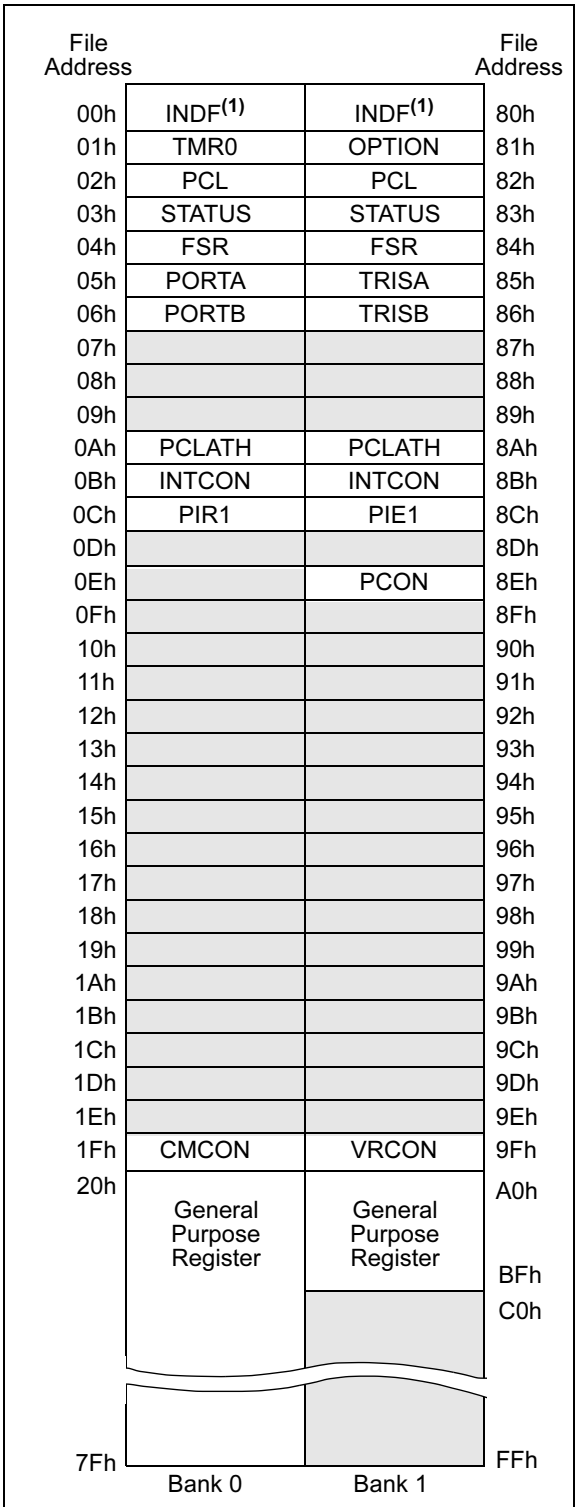
Downloaded from Arrow.com.

**FIGURE 4-4: DATA MEMORY MAP FOR THE PIC16C620/621**

Unimplemented data memory locations, read as '0'.

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

**FIGURE 4-5: DATA MEMORY MAP FOR THE PIC16C622**



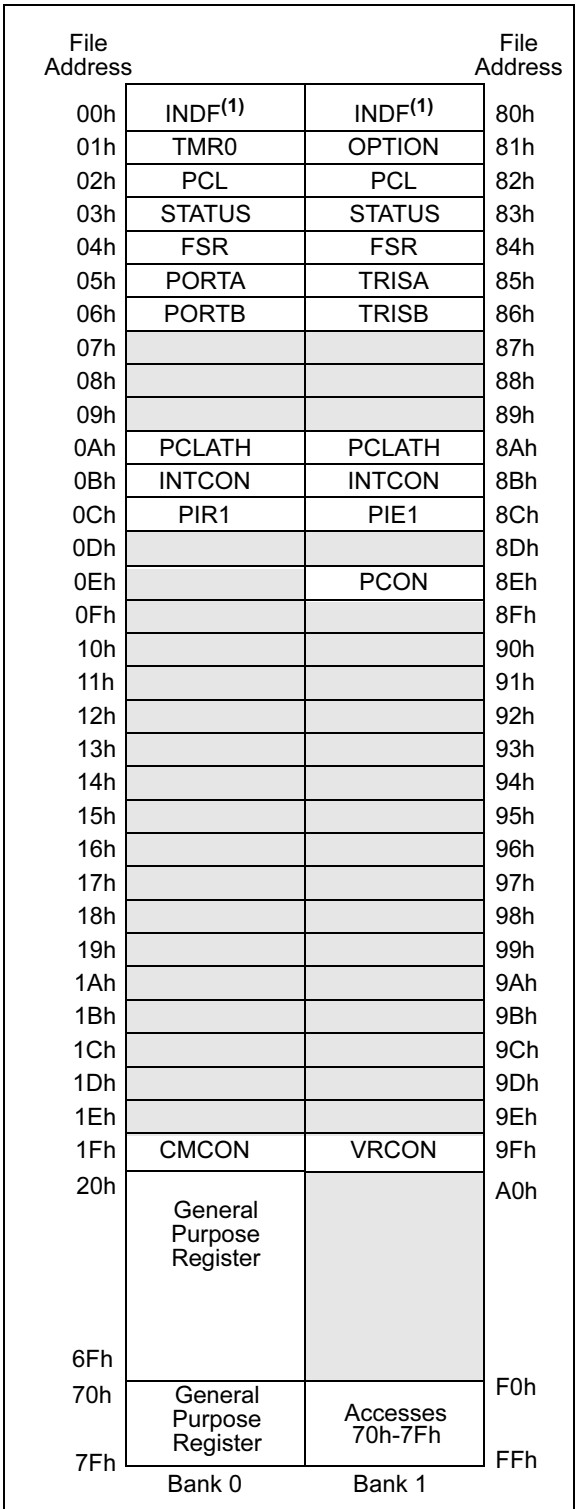

Unimplemented data memory locations, read as '0'.

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



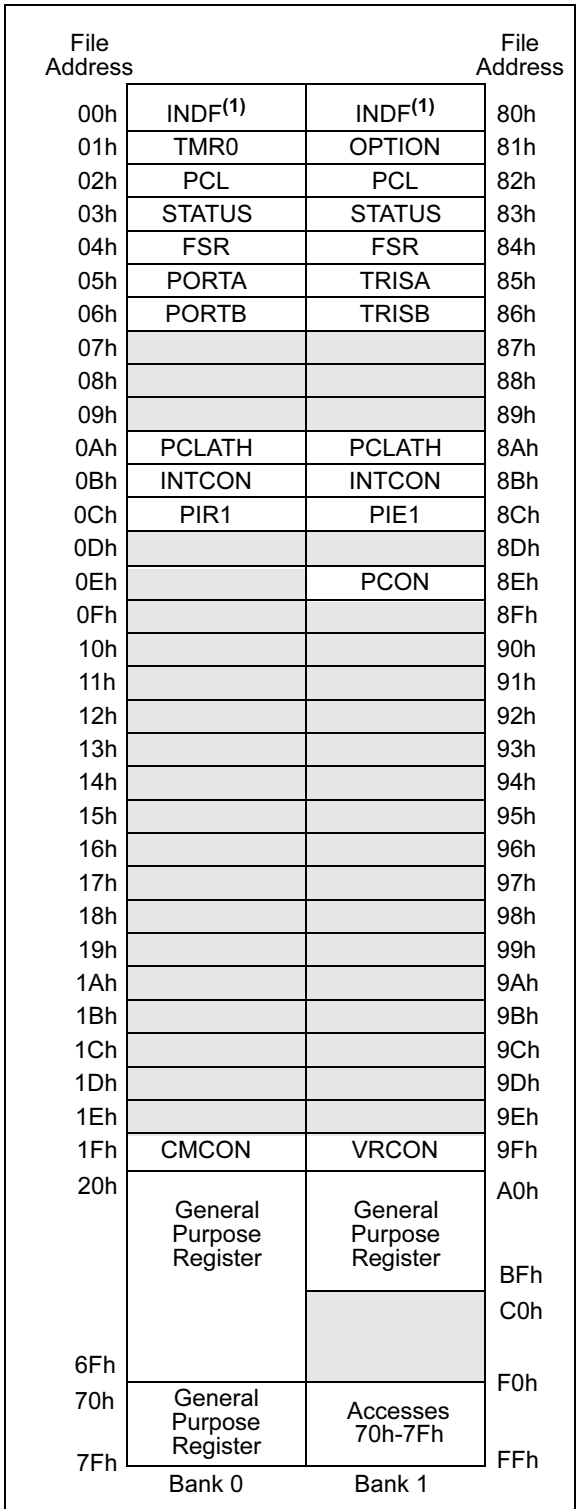
Downloaded from Arrow.com.

**FIGURE 4-6: DATA MEMORY MAP FOR THE PIC16C620A/CR620A/621A**




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

**FIGURE 4-7: DATA MEMORY MAP FOR THE PIC16C622A**




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


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

### 4.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and Peripheral functions for controlling the desired operation of the device (Table 4-1). These registers are static RAM.

The Special Function Registers can be classified into two sets (core and peripheral). The Special Function Registers associated with the "core" functions are described in this section. Those related to the operation of the peripheral features are described in the section of that peripheral feature.

**TABLE 4-1: SPECIAL REGISTERS FOR THE PIC16C62X**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR Reset | Value on all other RESETS(1) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bank 0** | | | | | | | | | | | |
| 00h | INDF | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | xxxx xxxx | xxxx xxxx |
| 01h | TMR0 | Timer0 Module's Register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 02h | PCL | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 0000 0000 |
| 03h | STATUS | IRP(2) | RP1(2) | RP0 | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C | 0001 1xxx | 000q quuu |
| 04h | FSR | Indirect data memory address pointer | | | | | | | | xxxx xxxx | uuuu uuuu |
| 05h | PORTA | — | — | — | $\overline{RA4}$ | $\overline{RA3}$ | $\overline{RA2}$ | $\overline{RA1}$ | RA0 | ---x 0000 | ---u 0000 |
| 06h | PORTB | RB7 | RB6 | RB5 | $\overline{RB4}$ | $\overline{RB3}$ | $\overline{RB2}$ | $\overline{RB1}$ | RB0 | xxxx xxxx | uuuu uuuu |
| 07h-09h | Unimplemented | | | | | | | | | — | — |
| 0Ah | PCLATH | — | — | — | Write buffer for upper 5 bits of program counter | | | | | ---0 0000 | ---0 0000 |
| 0Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | CMIF | — | — | — | — | — | — | -0-- ---- | -0-- ---- |
| 0Dh-1Eh | Unimplemented | | | | | | | | | — | — |
| 1Fh | CMCON | C2OUT | C1OUT | — | — | CIS | CM2 | CM1 | CM0 | 00-- 0000 | 00-- 0000 |
| Bank 1 | | | | | | | | | | | |
| 80h | INDF | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | xxxx xxxx | xxxx xxxx |
| 81h | OPTION | $\overline{RBPU}$ | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |
| 82h | PCL | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 0000 0000 |
| 83h | STATUS | IRP(2) | RP1(2) | RP0 | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C | 0001 1xxx | 000q quuu |
| 84h | FSR | Indirect data memory address pointer | | | | | | | | xxxx xxxx | uuuu uuuu |
| 85h | TRISA | — | — | — | TRISA4 | TRISA3 | TRISA2 | TRISA1 | TRISA0 | ---1 1111 | ---1 1111 |
| 86h | TRISB | TRISB7 | TRISB6 | TRISB5 | TRISB4 | TRISB3 | TRISB2 | TRISB1 | TRISB0 | 1111 1111 | 1111 1111 |
| 87h-89h | Unimplemented | | | | | | | | | — | — |
| 8Ah | PCLATH | — | — | — | Write buffer for upper 5 bits of program counter | | | | | ---0 0000 | ---0 0000 |
| 8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 8Ch | PIE1 | — | CMIE | — | — | — | — | — | — | -0-- ---- | -0-- ---- |
| 8Dh | Unimplemented | | | | | | | | | — | — |
| 8Eh | PCON | — | — | — | — | — | — | $\overline{POR}$ | $\overline{BOR}$ | ---- --0x | ---- --uq |
| 8Fh-9Eh | Unimplemented | | | | | | | | | — | — |
| 9Fh | VRCON | VREN | VROE | VRR | — | VR3 | VR2 | VR1 | VR0 | 000- 0000 | 000- 0000 |

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 $\overline{MCLR}$ Reset, Brown-out Reset and Watchdog Timer Reset during normal operation.

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



Downloaded from Arrow.com.

#### 4.2.2.1 STATUS Register

The STATUS register, shown in Register 4-1, contains the arithmetic status of the ALU, the RESET status and the bank select bits for data memory.

The STATUS register can be the destination for any instruction, 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 $\overline{TO}$ and $\overline{PD}$ bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, `CLRF STATUS` will clear the upper-three bits and set the Z bit. This leaves the STATUS register as `000uu1uu` (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 bit. For other instructions not affecting any STATUS bits, see the "Instruction Set Summary".

**Note 1:** The IRP and RP1 bits (STATUS<7:6>) are not used by the PIC16C62X and should be programmed as '0'. Use of these bits as general purpose R/W bits is NOT recommended, since this may affect upward compatibility with future products.

**2:** The C and DC bits operate as a $\overline{Borrow}$ and Digit $\overline{Borrow}$ out bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

**REGISTER 4-1: STATUS REGISTER (ADDRESS 03H OR 83H)**

| Reserved | Reserved | R/W-0 | R-1 | R-1 | R/W-x | R/W-x | R/W-x |
|---|---|---|---|---|---|---|---|
| IRP | RP1 | RP0 | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C |
| bit 7 | | | | | | | bit 0 |

bit 7 **IRP:** Register Bank Select bit (used for indirect addressing)
`1` = Bank 2, 3 (100h - 1FFh)
`0` = Bank 0, 1 (00h - FFh)
The IRP bit is reserved on the PIC16C62X; always maintain this bit clear.

bit 6-5 **RP<1:0>**: Register Bank Select bits (used for direct addressing)
`01` = Bank 1 (80h - FFh)
`00` = Bank 0 (00h - 7Fh)
Each bank is 128 bytes. The RP1 bit is reserved on the PIC16C62X; always maintain this bit clear.

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

bit 3 **$\overline{PD}$**: Power-down bit
`1` = After power-up or by the `CLRWDT` instruction
`0` = By execution of the `SLEEP` instruction

bit 2 **Z**: Zero bit
`1` = The result of an arithmetic or logic operation is zero
`0` = The result of an arithmetic or logic operation is not zero

bit 1 **DC**: Digit carry/$\overline{borrow}$ bit (`ADDWF`, `ADDLW`,`SUBLW`,`SUBWF` instructions)(for $\overline{borrow}$ the polarity is reversed)
`1` = A carry-out from the 4th low order bit of the result occurred
`0` = No carry-out from the 4th low order bit of the result

bit 0 **C**: Carry/$\overline{borrow}$ bit (`ADDWF`, `ADDLW`,`SUBLW`,`SUBWF` instructions)
`1` = A carry-out from the Most Significant bit of the result occurred
`0` = No carry-out from the Most Significant bit of the result occurred

**Note:** For $\overline{borrow}$ 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 |


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

#### 4.2.2.2 OPTION Register

The OPTION register is a readable and writable register, which contains various control bits to configure the TMR0/WDT prescaler, the external RB0/INT interrupt, TMR0 and the weak pull-ups on PORTB.

| **Note:** | To achieve a 1:1 prescaler assignment for TMR0, assign the prescaler to the WDT (PSA = 1). |
|---|---|

**REGISTER 4-2: OPTION REGISTER (ADDRESS 81H)**

| R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| RBPU | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 |
| bit 7 | | | | | | | bit 0 |

bit 7 **$\overline{\text{RBPU}}$: PORTB Pull-up Enable bit**
1 = PORTB pull-ups are disabled
0 = PORTB pull-ups are enabled by individual port latch values

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

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

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

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

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

| Bit Value | TMR0 Rate | WDT Rate |
|---|---|---|
| 000 | 1 : 2 | 1 : 1 |
| 001 | 1 : 4 | 1 : 2 |
| 010 | 1 : 8 | 1 : 4 |
| 011 | 1 : 16 | 1 : 8 |
| 100 | 1 : 32 | 1 : 16 |
| 101 | 1 : 64 | 1 : 32 |
| 110 | 1 : 128 | 1 : 64 |
| 111 | 1 : 256 | 1 : 128 |

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



Downloaded from Arrow.com.

### 4.2.2.3 INTCON Register

The INTCON register is a readable and writable register, which contains the various enable and flag bits for all interrupt sources except the comparator module. See Section 4.2.2.4 and Section 4.2.2.5 for a description of the comparator enable and flag bits.

**Note:** Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>).

**REGISTER 4-3: INTCON REGISTER (ADDRESS 0BH OR 8BH)**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-x |
|---|---|---|---|---|---|---|---|
| GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF |
| bit 7 | | | | | | | bit 0 |

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

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

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

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

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

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

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

bit 0 **RBIF**: RB Port Change Interrupt Flag bit
1 = When at least one of the RB<7:4> pins changed state (must be cleared in software)
0 = None of the RB<7:4> pins have changed state

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


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

#### 4.2.2.4 PIE1 Register

This register contains the individual enable bit for the comparator interrupt.

**REGISTER 4-4: PIE1 REGISTER (ADDRESS 8CH)**

| U-0 | R/W-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| — | CMIE | — | — | — | — | — | — |
| bit 7 | | | | | | | bit 0 |

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

bit 6 **CMIE**: Comparator Interrupt Enable bit
1 = Enables the Comparator interrupt
0 = Disables the Comparator interrupt

bit 5-0 **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 |

#### 4.2.2.5 PIR1 Register

This register contains the individual flag bit for the comparator interrupt.

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

**REGISTER 4-5: PIR1 REGISTER (ADDRESS 0CH)**

| U-0 | R/W-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| — | CMIF | — | — | — | — | — | — |
| bit 7 | | | | | | | bit 0 |

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

bit 6 **CMIF**: Comparator Interrupt Flag bit
1 = Comparator input has changed
0 = Comparator input has not changed

bit 5-0 **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 |



Downloaded from Arrow.com.

#### 4.2.2.6 PCON Register

The PCON register contains flag bits to differentiate between a Power-on Reset, an external MCLR Reset, WDT Reset or a Brown-out Reset.

| | |
|---|---|
| **Note:** | BOR is unknown on Power-on Reset. It must then be set by the user and checked on subsequent RESETS to see if BOR is cleared, indicating a brown-out has occurred. The BOR STATUS bit is a "don't care" and is not necessarily predictable if the brown-out circuit is disabled (by programming BODEN bit in the Configuration word). |

**REGISTER 4-6: PCON REGISTER (ADDRESS 8Eh)**

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | POR | BOR |
| bit 7 | | | | | | | bit 0 |

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

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

bit 0 **BOR**: Brown-out Reset STATUS bit
1 = No Brown-out Reset occurred
0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

| 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 |

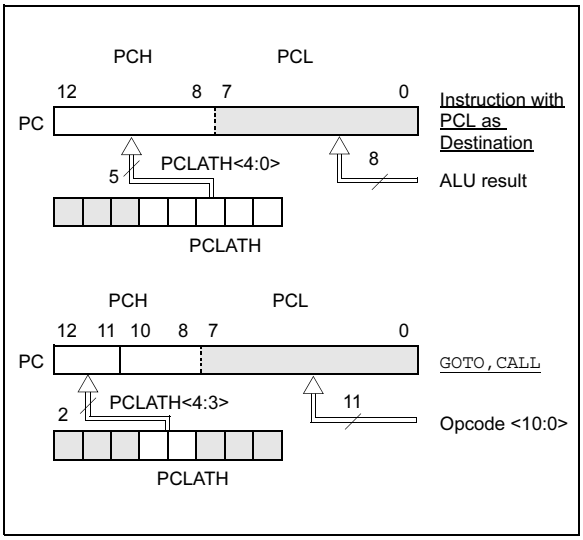

© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 4.3 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 4-8 shows the two situations for the loading of the PC. The upper example in the figure shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in the figure shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

**FIGURE 4-8: LOADING OF PC IN DIFFERENT SITUATIONS**




### 4.3.1 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (ADDWF PCL). When doing 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).

### 4.3.2 STACK

The PIC16C62X family has an 8-level deep x 13-bit wide hardware stack (Figure 4-2 and Figure 4-3). 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.



Downloaded from Arrow.com.

## 4.4 Indirect Addressing, INDF and FSR Registers

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

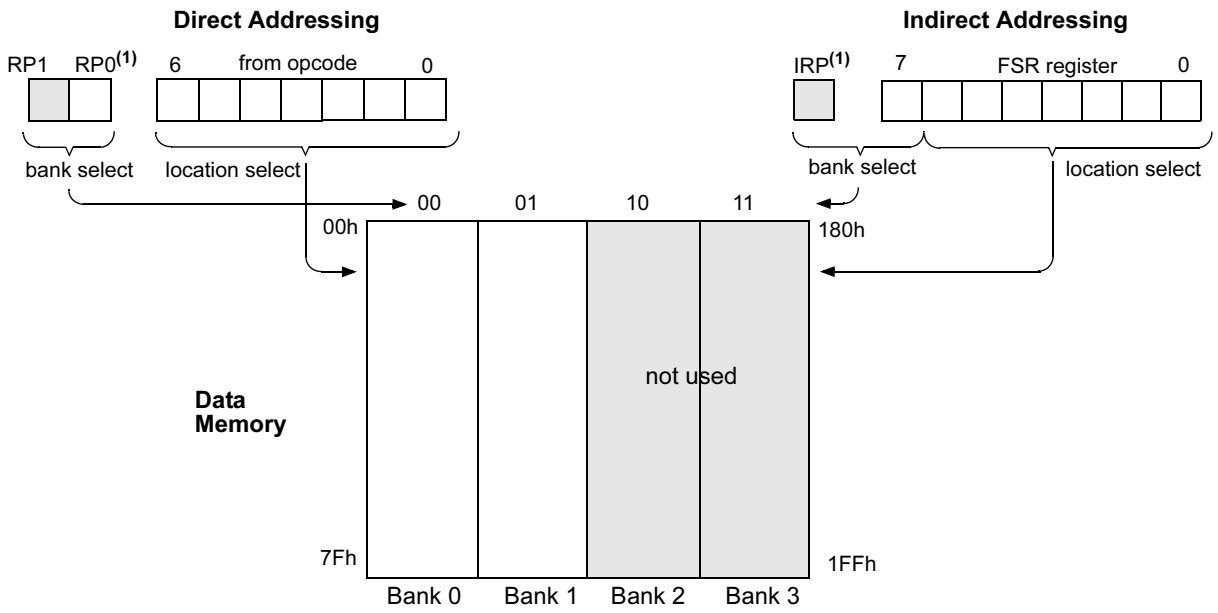
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select Register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no-operation (although STATUS 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 4-9. However, IRP is not used in the PIC16C62X.

A simple program to clear RAM location 20h-7Fh using indirect addressing is shown in Example 4-1.

**EXAMPLE 4-1: INDIRECT ADDRESSING**

```
      movlw   0x20       ;initialize pointer
      movwf   FSR        ;to RAM
NEXT  clrf    INDF       ;clear INDF register
      incf    FSR        ;inc pointer
      btfss   FSR,7      ;all done?
      goto    NEXT       ;no clear next
                         ;yes continue
CONTINUE:
```

**FIGURE 4-9: DIRECT/INDIRECT ADDRESSING PIC16C62X**




For memory map detail see (Figure 4-4, Figure 4-5, Figure 4-6 and Figure 4-7).

**Note 1:** The RP1 and IRP bits are reserved; always maintain these bits clear.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 5.0 I/O PORTS

The PIC16C62X have two ports, PORTA and PORTB. Some pins for these I/O ports are multiplexed with an alternate function for the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

### 5.1 PORTA and TRISA Registers

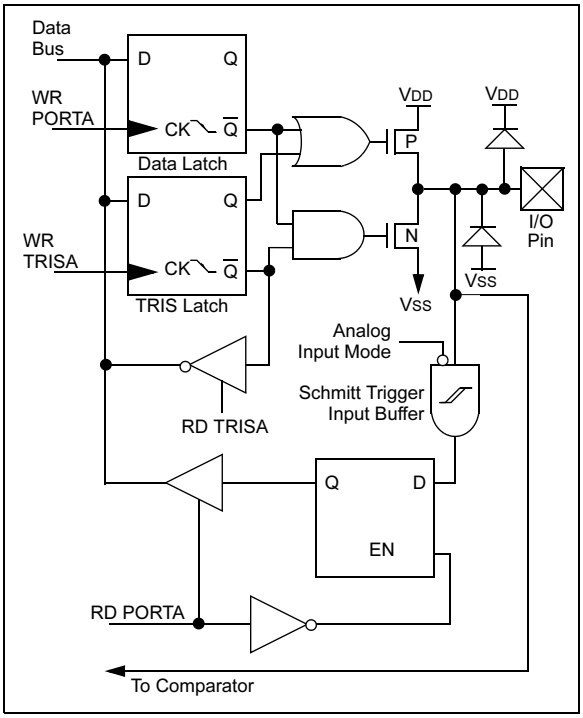
PORTA is a 5-bit wide latch. RA4 is a Schmitt Trigger input and an open drain output. Port RA4 is multiplexed with the T0CKI clock input. All other RA port pins have Schmitt Trigger input levels and full CMOS output drivers. All pins have data direction bits (TRIS registers), which can configure these pins as input or output.

A '1' in the TRISA register puts the corresponding output driver in a Hi-impedance mode. A '0' in the TRISA register puts the contents of the output latch on the selected pin(s).

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. So a write to a port implies that the port pins are first read, then this value is modified and written to the port data latch.

The PORTA pins are multiplexed with comparator and voltage reference functions. The operation of these pins are selected by control bits in the CMCON (comparator control register) register and the VRCON (voltage reference control register) register. When selected as a comparator input, these pins will read as '0's.

**FIGURE 5-1: BLOCK DIAGRAM OF RA1:RA0 PINS**




**Note:** On RESET, the TRISA register is set to all inputs. The digital inputs are disabled and the comparator inputs are forced to ground to reduce excess current consumption.

TRISA controls the direction of the RA pins, even when they are being used as comparator inputs. The user must make sure to keep the pins configured as inputs when using them as comparator inputs.

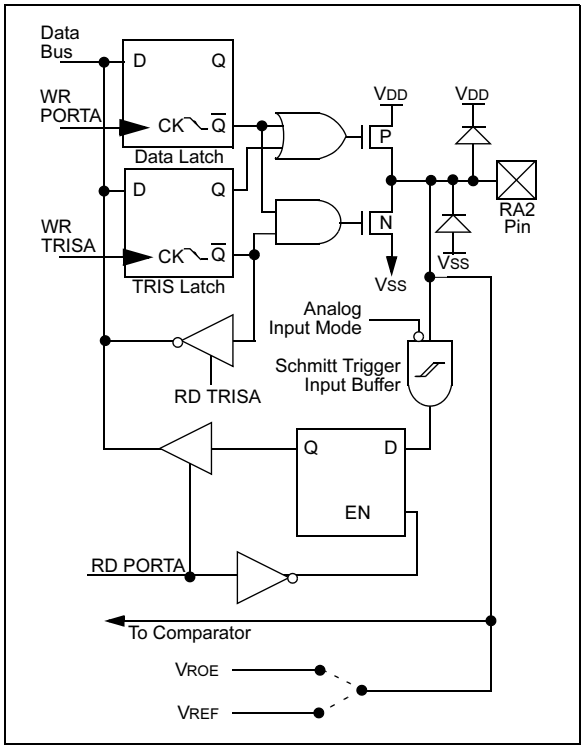
The RA2 pin will also function as the output for the voltage reference. When in this mode, the VREF pin is a very high impedance output and must be buffered prior to any external load. The user must configure TRISA<2> bit as an input and use high impedance loads.

In one of the Comparator modes defined by the CMCON register, pins RA3 and RA4 become outputs of the comparators. The TRISA<4:3> bits must be cleared to enable outputs to use this function.

**EXAMPLE 5-1: INITIALIZING PORTA**

```
CLRF    PORTA           ;Initialize PORTA by setting
                        ;output data latches
MOVLW   0X07            ;Turn comparators off and
MOVWF   CMCON           ;enable pins for I/O
                        ;functions
BSF     STATUS, RP0     ;Select Bank1
MOVLW   0x1F            ;Value used to initialize
                        ;data direction
MOVWF   TRISA           ;Set RA<4:0> as inputs
                        ;TRISA<7:5> are always
                        ;read as '0'.
```

**FIGURE 5-2: BLOCK DIAGRAM OF RA2 PIN**






Downloaded from Arrow.com.

**FIGURE 5-3: BLOCK DIAGRAM OF RA3 PIN**

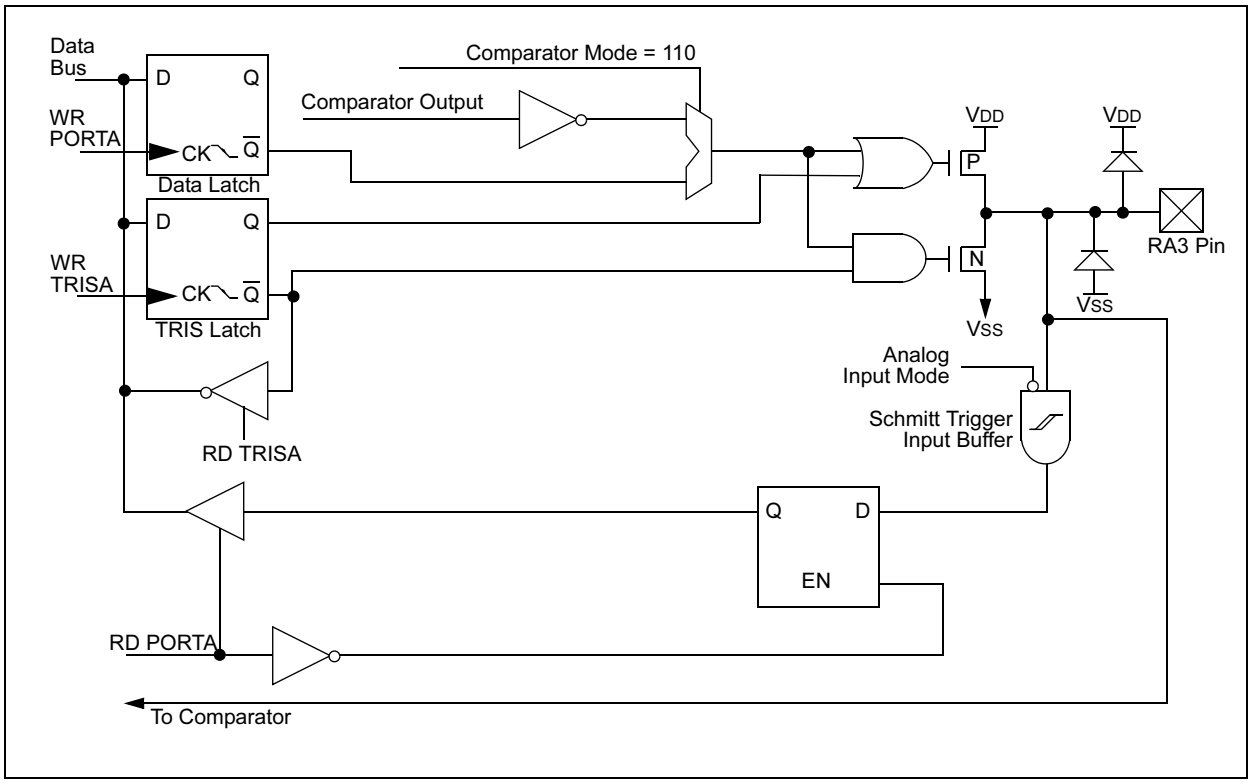
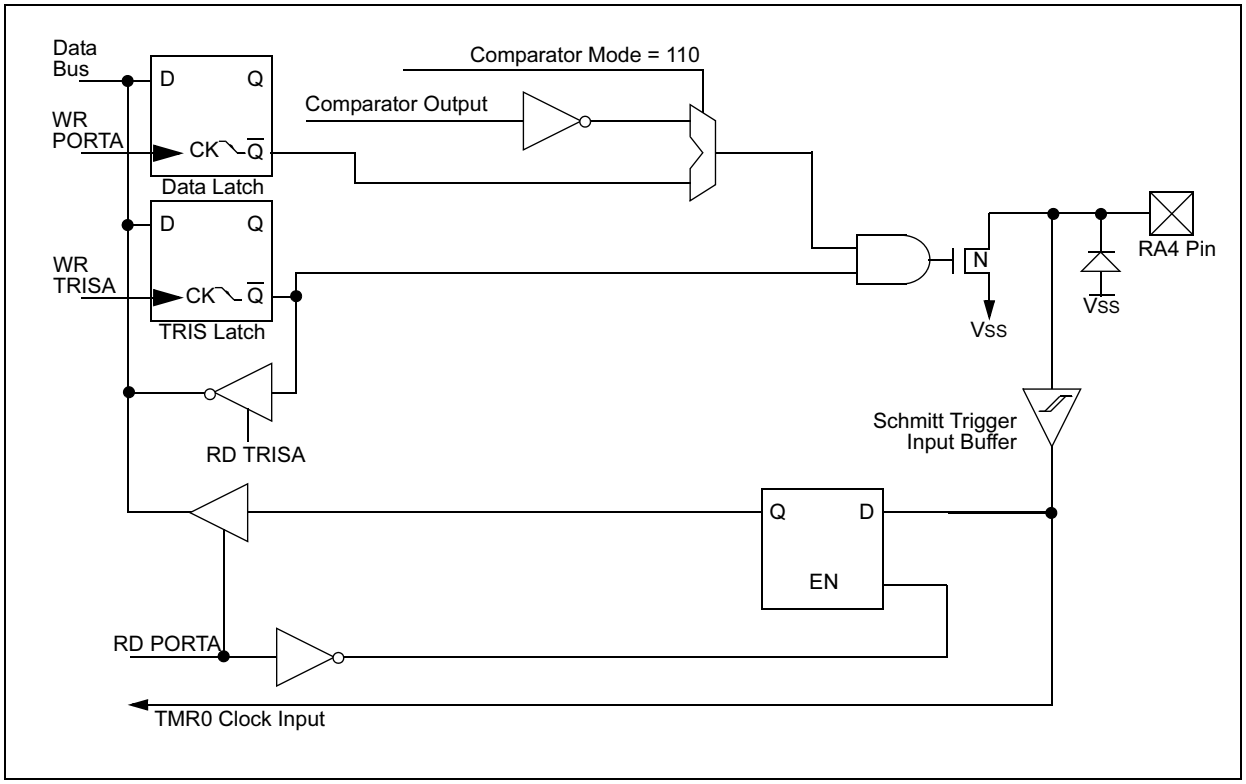



**FIGURE 5-4: BLOCK DIAGRAM OF RA4 PIN**





© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

**TABLE 5-1: PORTA FUNCTIONS**

| Name | Bit # | Buffer Type | Function |
|---|---|---|---|
| RA0/AN0 | bit0 | ST | Input/output or comparator input |
| RA1/AN1 | bit1 | ST | Input/output or comparator input |
| RA2/AN2/VREF | bit2 | ST | Input/output or comparator input or VREF output |
| RA3/AN3 | bit3 | ST | Input/output or comparator input/output |
| RA4/T0CKI | bit4 | ST | Input/output or external clock input for TMR0 or comparator output. Output is open drain type. |

Legend: ST = Schmitt Trigger input

**TABLE 5-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTA**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR | Value on All Other RESETS |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 05h | PORTA | — | — | — | RA4 | RA3 | RA2 | RA1 | RA0 | ---x 0000 | ---u 0000 |
| 85h | TRISA | — | — | — | TRISA 4 | TRISA 3 | TRISA 2 | TRISA 1 | TRISA 0 | ---1 1111 | ---1 1111 |
| 1Fh | CMCON | C2OUT | C1OUT | — | — | CIS | CM2 | CM1 | CM0 | 00-- 0000 | 00-- 0000 |
| 9Fh | VRCON | VREN | VROE | VRR | — | VR3 | VR2 | VR1 | VR0 | 000- 0000 | 000- 0000 |

Legend: — = Unimplemented locations, read as '0', u = unchanged, x = unknown

**Note:** Shaded bits are not used by PORTA.



Downloaded from Arrow.com.

## 5.2 PORTB and TRISB Registers

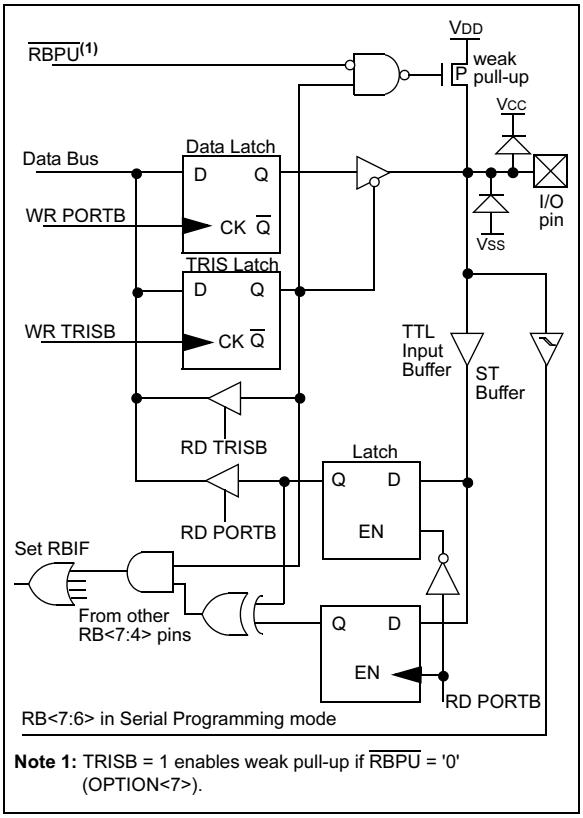
PORTB is an 8-bit wide, bi-directional port. The corresponding data direction register is TRISB. A '1' in the TRISB register puts the corresponding output driver in a High Impedance mode. A '0' in the TRISB register puts the contents of the output latch on the selected pin(s).

Reading PORTB 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. So a write to a port implies that the port pins are first read, then this value is modified and written to the port data latch.

Each of the PORTB pins has a weak internal pull-up (≈200 μA typical). A single control bit can turn on all the pull-ups. This is done by clearing the $\overline{\text{RBPU}}$ (OPTION<7>) bit. The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on Power-on Reset.

Four of PORTB's pins, RB<7:4>, have an interrupt on change feature. Only pins configured as inputs can cause this interrupt to occur (e.g., any RB<7:4> pin configured as an output is excluded from the interrupt on change comparison). The input pins (of RB<7:4>) are compared with the old value latched on the last read of PORTB. The "mismatch" outputs of RB<7:4> are OR'ed together to generate the RBIF interrupt (flag latched in INTCON<0>).

**FIGURE 5-5: BLOCK DIAGRAM OF RB<7:4> PINS**




**Note 1:** TRISB = 1 enables weak pull-up if $\overline{\text{RBPU}}$ = '0' (OPTION<7>).

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

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

b) Clear flag bit RBIF.

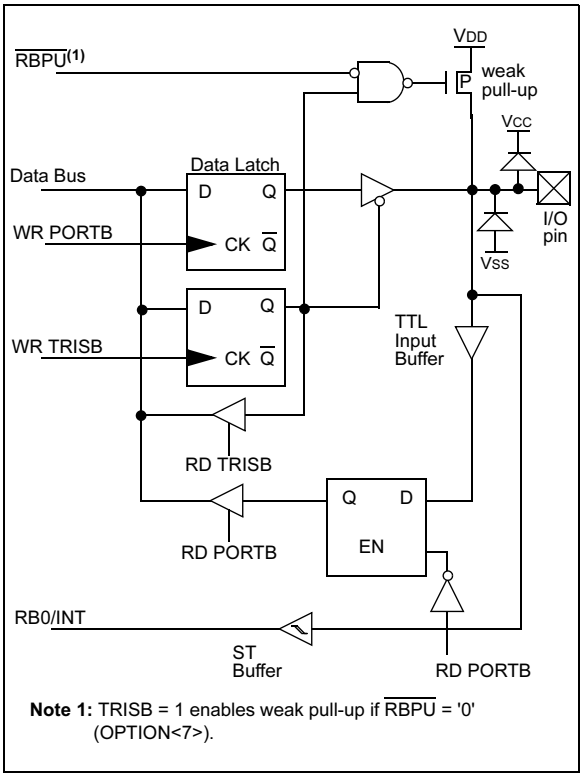
A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition and allow flag bit RBIF to be cleared.

This interrupt on mismatch feature, together with software configurable pull-ups on these four pins allow easy interface to a key pad and make it possible for wake-up on key-depression. (See AN552, "Implementing Wake-Up on Key Strokes.)

| **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 RBIF interrupt flag may not get set. |
|---|---|

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

**FIGURE 5-6: BLOCK DIAGRAM OF RB<3:0> PINS**




**Note 1:** TRISB = 1 enables weak pull-up if $\overline{\text{RBPU}}$ = '0' (OPTION<7>).


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

**TABLE 5-3: PORTB FUNCTIONS**

| Name | Bit # | Buffer Type | Function |
|---|---|---|---|
| RB0/INT | bit0 | TTL/ST(1) | Input/output or external interrupt input. Internal software programmable weak pull-up. |
| RB1 | bit1 | TTL | Input/output pin. Internal software programmable weak pull-up. |
| RB2 | bit2 | TTL | Input/output pin. Internal software programmable weak pull-up. |
| RB3 | bit3 | TTL | Input/output pin. Internal software programmable weak pull-up. |
| RB4 | bit4 | TTL | Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up. |
| RB5 | bit5 | TTL | Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up. |
| RB6 | bit6 | TTL/ST(2) | Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up. Serial programming clock pin. |
| RB7 | bit7 | TTL/ST(2) | Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up. Serial programming data pin. |

Legend: ST = Schmitt Trigger, TTL = TTL input

**Note 1:** This buffer is a Schmitt Trigger input when configured as the external interrupt.
**2:** This buffer is a Schmitt Trigger input when used in Serial Programming mode.

**TABLE 5-4: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR | Value on All Other RESETS |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 06h | PORTB | RB7 | RB6 | RB5 | RB4 | RB3 | RB2 | RB1 | RB0 | xxxx xxxx | uuuu uuuu |
| 86h | TRISB | TRISB7 | TRISB6 | TRISB5 | TRISB4 | TRISB3 | TRISB2 | TRISB1 | TRISB0 | 1111 1111 | 1111 1111 |
| 81h | OPTION | $\overline{\text{RBPU}}$ | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |

Legend: u = unchanged, x = unknown

**Note 1:** Shaded bits are not used by PORTB.



Downloaded from Arrow.com.

## 5.3 I/O Programming Considerations

### 5.3.1 BI-DIRECTIONAL I/O PORTS

Any instruction which writes, operates internally as a read followed by a write operation. The BCF and BSF instructions, for example, read the register into the CPU, execute the bit operation and write the result back to the register. Caution must be used when these instructions are applied to a port with both inputs and outputs defined. For example, a BSF operation on bit5 of PORTB will cause all eight bits of PORTB to be read into the CPU. Then the BSF operation takes place on bit5 and PORTB is written to the output latches. If another bit of PORTB is used as a bi-directional I/O pin (e.g., bit0) and it is defined as an input at this time, the input signal present on the pin itself would be read into the CPU and re-written to the data latch of this particular pin, overwriting the previous content. As long as the pin stays in the Input mode, no problem occurs. However, if bit0 is switched into Output mode later on, the content of the data latch may now be unknown.

Reading the port register reads the values of the port pins. Writing to the port register writes the value to the port latch. When using read-modify-write instructions (ex. BCF, BSF, etc.) on a port, the value of the port pins is read, the desired operation is done to this value, and this value is then written to the port latch.

Example 5-2 shows the effect of two sequential read-modify-write instructions (ex., BCF, BSF, etc.) on an I/O port.

A pin actively outputting a Low or High should not be driven from external devices at the same time in order to change the level on this pin ("wired-or", "wired-and"). The resulting high output currents may damage the chip.

**EXAMPLE 5-2: READ-MODIFY-WRITE INSTRUCTIONS ON AN I/O PORT**

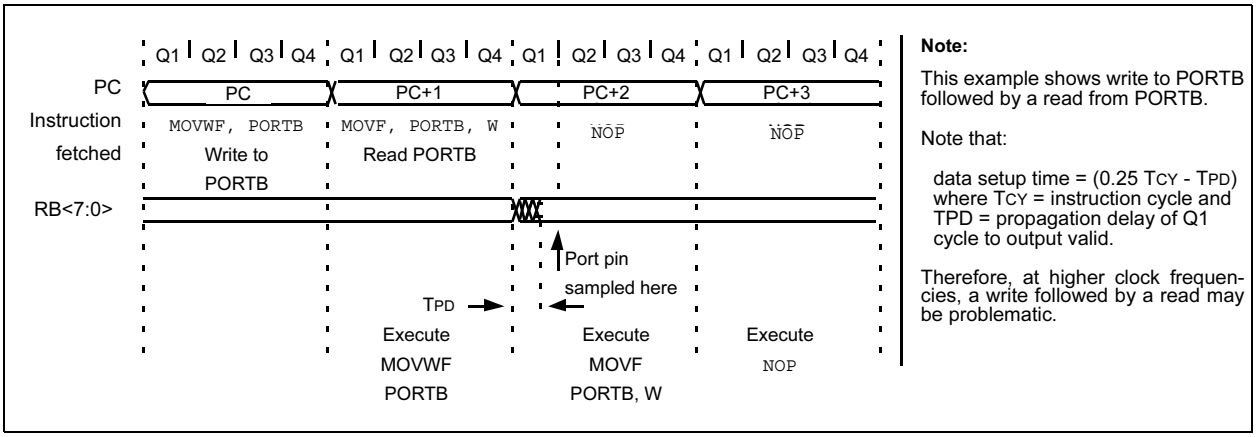
```
; Initial PORT settings:    PORTB<7:4> Inputs
;
;                           PORTB<3:0> Outputs
; PORTB<7:6> have external pull-up and are not
; connected to other circuitry
;
;                           PORT  latch   PORT  pins
;                           ----------   ---------
                                         -


    BCF  PORTB, 7         ; 01pp  pppp    11pp  pppp
    BCF  PORTB, 6         ; 10pp  pppp    11pp  pppp
    BSF  STATUS,RP0       ;
    BCF  TRISB, 7         ; 10pp  pppp    11pp  pppp
    BCF  TRISB, 6         ; 10pp  pppp    10pp  pppp
;
; Note that the user may have expected the pin
; values to be 00pp pppp. The 2nd BCF caused
; RB7 to be latched as the pin value (High).
```

### 5.3.2 SUCCESSIVE OPERATIONS ON I/O PORTS

The actual write to an I/O port happens at the end of an instruction cycle, whereas for reading, the data must be valid at the beginning of the instruction cycle (Figure 5-7). Therefore, care must be exercised if a write followed by a read operation is carried out on the same I/O port. The sequence of instructions should be such to allow the pin voltage to stabilize (load dependent) before the next instruction which causes that file to be read into the CPU is executed. Otherwise, the previous state of that pin may be read into the CPU rather than the new state. When in doubt, it is better to separate these instructions with a NOP or another instruction not accessing this I/O port.

**FIGURE 5-7: SUCCESSIVE I/O OPERATION**




**Note:**

This example shows write to PORTB followed by a read from PORTB.

Note that:

data setup time = (0.25 TCY - TPD) where TCY = instruction cycle and TPD = propagation delay of Q1 cycle to output valid.

Therefore, at higher clock frequencies, a write followed by a read may be problematic.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 6.0 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 6-1 is a simplified block diagram of the Timer0 module.

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

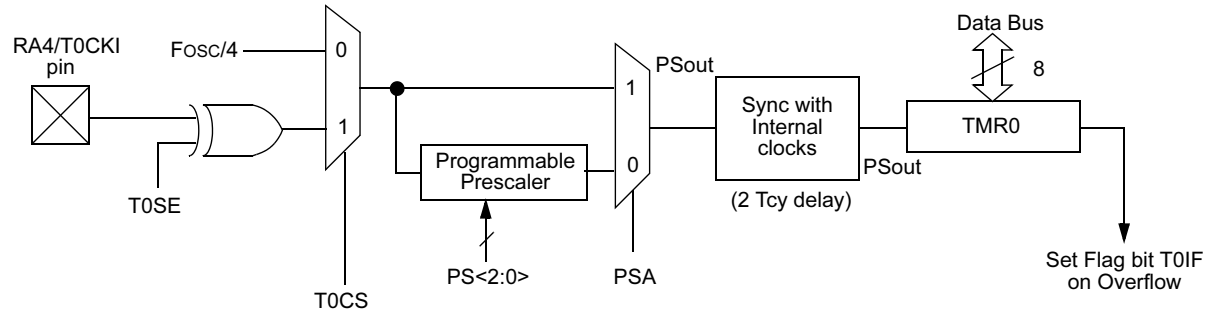
Counter mode is selected by setting the T0CS bit. In this mode, Timer0 will increment either on every rising or falling edge of pin RA4/T0CKI. The incrementing edge is determined by the source edge (T0SE) control bit (OPTION<4>). Clearing the T0SE bit selects the rising edge. Restrictions on the external clock input are discussed in detail in Section 6.2.

The prescaler is shared between the Timer0 module and the Watchdog Timer. The prescaler assignment is controlled in software by the control bit PSA (OPTION<3>). Clearing the PSA bit will assign the prescaler to Timer0. The prescaler is not readable or writable. When the prescaler is assigned to the Timer0 module, prescale value of 1:2, 1:4, ..., 1:256 are selectable. Section 6.3 details the operation of the prescaler.

### 6.1 TIMER0 Interrupt

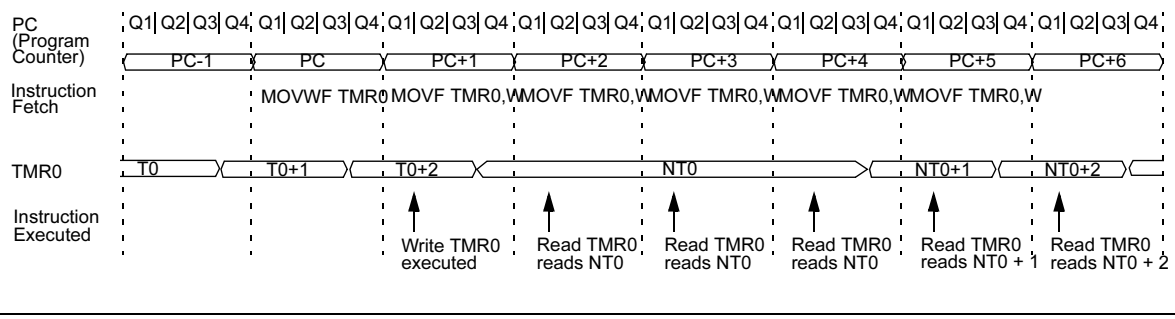
Timer0 interrupt is generated when the TMR0 register timer/counter overflows from FFh to 00h. This overflow sets the T0IF bit. The interrupt can be masked by clearing the T0IE bit (INTCON<5>). The T0IF bit (INTCON<2>) must be cleared in software by the Timer0 module interrupt service routine before re-enabling this interrupt. The Timer0 interrupt cannot wake the processor from SLEEP, since the timer is shut off during SLEEP. See Figure 6-4 for Timer0 interrupt timing.

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




**Note 1:** Bits T0SE, T0CS, PS2, PS1, PS0 and PSA are located in the OPTION register.
**2:** The prescaler is shared with Watchdog Timer (Figure 6-6).

**FIGURE 6-2: TIMER0 (TMR0) TIMING: INTERNAL CLOCK/NO PRESCALER**






Downloaded from Arrow.com.

**FIGURE 6-3: TIMER0 TIMING: INTERNAL CLOCK/PRESCALE 1:2**

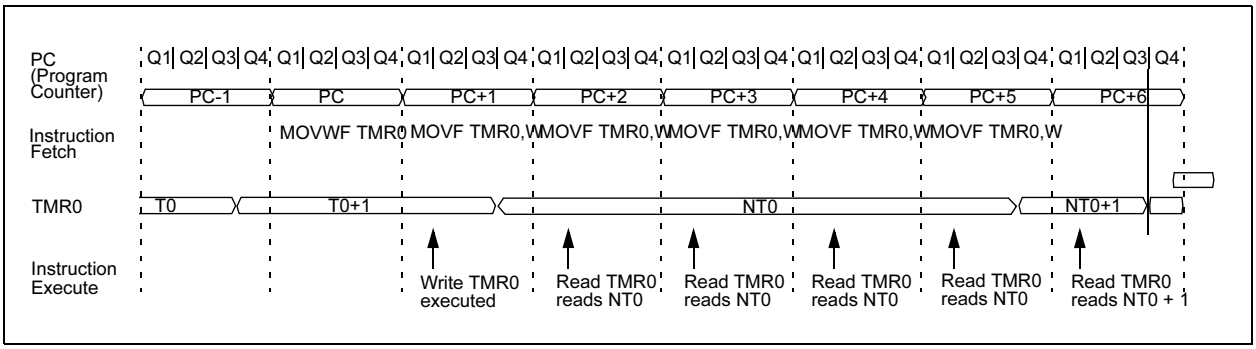
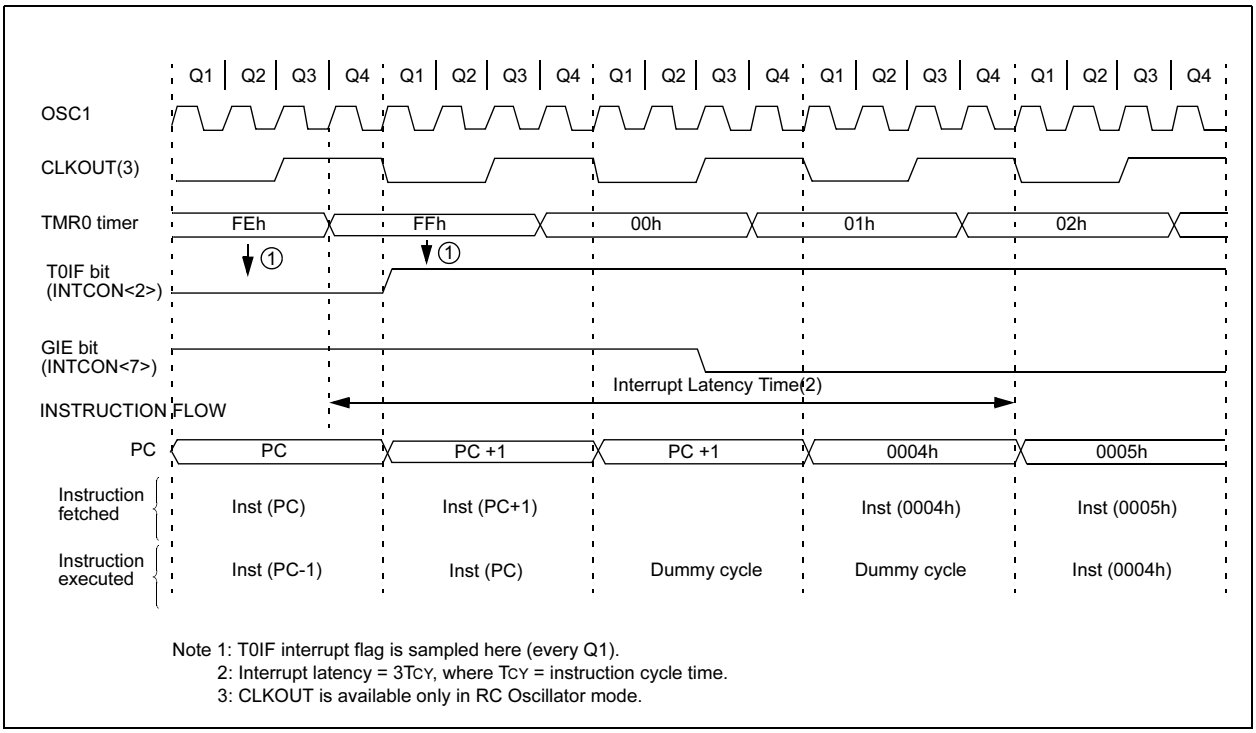



**FIGURE 6-4: TIMER0 INTERRUPT TIMING**




Note 1: T0IF interrupt flag is sampled here (every Q1).
2: Interrupt latency = 3TCY, where TCY = instruction cycle time.
3: CLKOUT is available only in RC Oscillator mode.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 6.2 Using Timer0 with External Clock

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

### 6.2.1 EXTERNAL CLOCK SYNCHRONIZATION

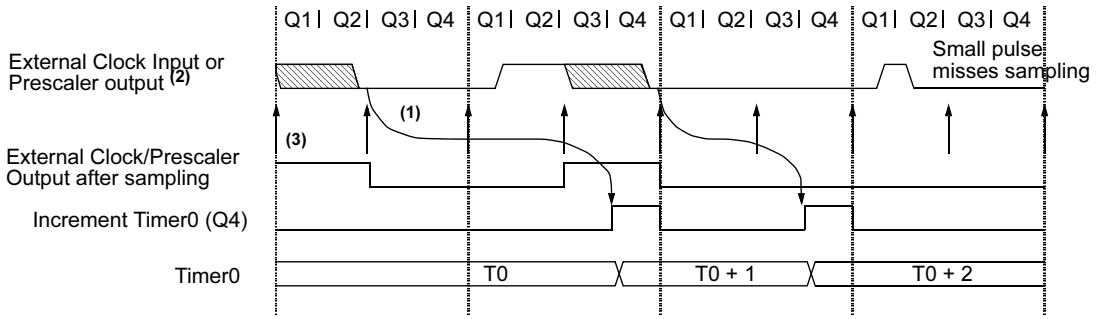
When no prescaler is used, the external clock input is the same as the prescaler output. The synchronization of T0CKI with the internal phase clocks is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks (Figure 6-5). Therefore, it is necessary for T0CKI 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.

When a prescaler is used, the external clock input is divided by the asynchronous ripple-counter type prescaler, so that the prescaler output is symmetrical. For the external clock to meet the sampling requirement, the ripple-counter must be taken into account. Therefore, it is necessary for T0CKI to have a period of at least 4TOSC (and a small RC delay of 40 ns) divided by the prescaler value. The only requirement on T0CKI high and low time is that they do not violate the minimum pulse width requirement of 10 ns. Refer to parameters 40, 41 and 42 in the electrical specification of the desired device.

### 6.2.2 TIMER0 INCREMENT DELAY

Since the prescaler output is synchronized with the internal clocks, there is a small delay from the time the external clock edge occurs to the time the TMR0 is actually incremented. Figure 6-5 shows the delay from the external clock edge to the timer incrementing.

**FIGURE 6-5: TIMER0 TIMING WITH EXTERNAL CLOCK**




**Note 1:** Delay from clock input change to Timer0 increment is 3Tosc to 7Tosc. (Duration of Q = Tosc). Therefore, the error in measuring the interval between two edges on Timer0 input = ±4Tosc max.
**2:** External clock if no prescaler selected, Prescaler output otherwise.
**3:** The arrows indicate the points in time where sampling occurs.



Downloaded from Arrow.com.

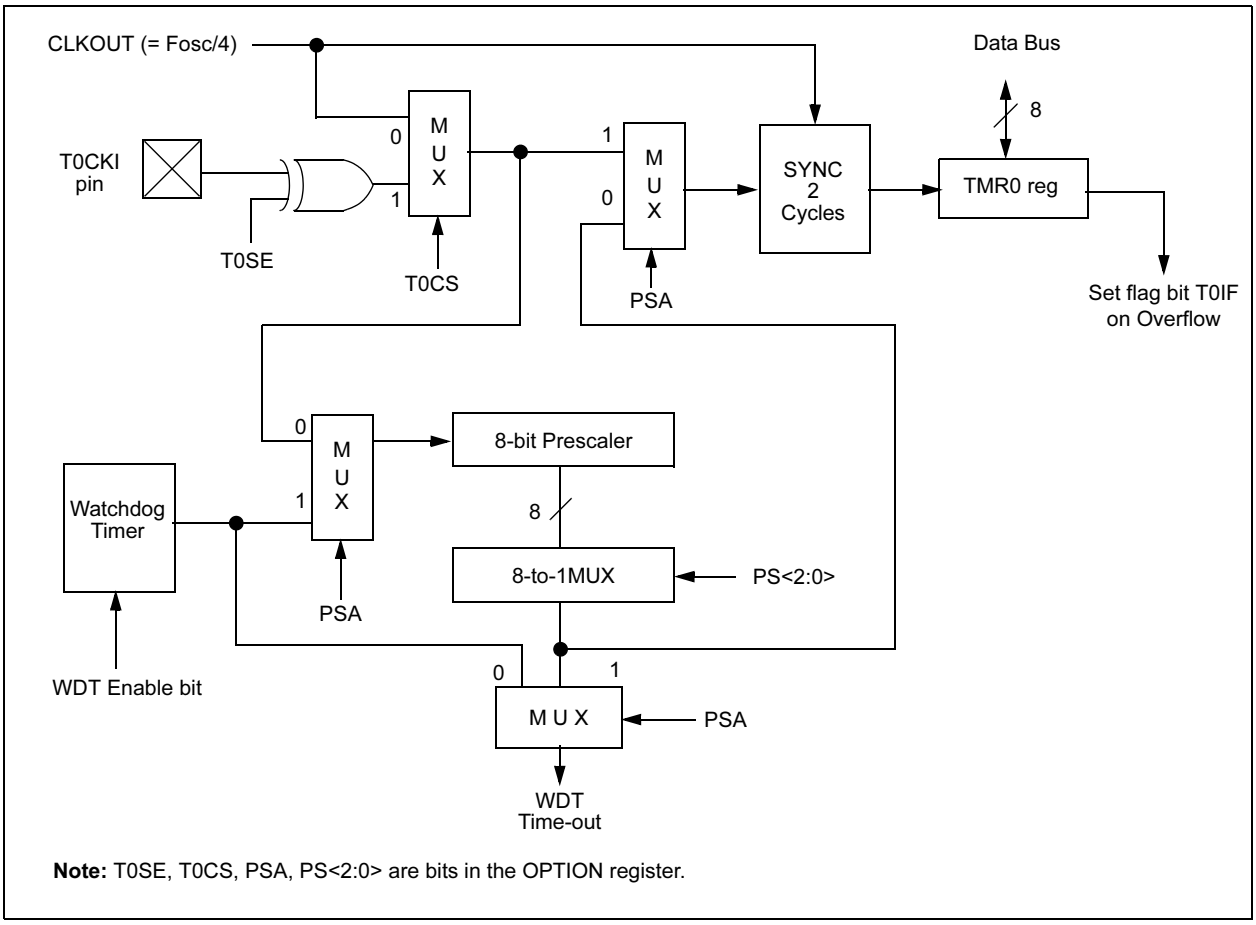
## 6.3 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module, or as a postscaler for the Watchdog Timer, respectively (Figure 6-6). For simplicity, this counter is being referred to as “prescaler” throughout this data sheet. Note that there is only one prescaler available which is mutually exclusive between the Timer0 module and the Watchdog Timer. Thus, a prescaler assignment for the Timer0 module means that there is no prescaler for the Watchdog Timer and vice-versa.

The PSA and PS<2:0> bits (OPTION<3:0>) determine the prescaler assignment and prescale ratio.

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. The prescaler is not readable or writable.

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




**Note:** T0SE, T0CS, PSA, PS<2:0> are bits in the OPTION register.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

### 6.3.1 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 6-1) must be executed when changing the prescaler assignment from Timer0 to WDT.)

**EXAMPLE 6-1: CHANGING PRESCALER (TIMER0→WDT)**

```
 1.BCF      STATUS, RP0    ;Skip if already in
                           ;Bank 0
 2.CLRWDT                  ;Clear WDT
 3.CLRF     TMR0           ;Clear TMR0 & Prescaler
 4.BSF      STATUS, RP0    ;Bank 1
 5.MOVLW    '00101111'b;   ;These 3 lines (5, 6, 7)
 6.MOVWF    OPTION         ;are required only if
                           ;desired PS<2:0> are
 7.CLRWDT                  ;000 or 001
 8.MOVLW    '00101xxx'b    ;Set Postscaler to
 9.MOVWF    OPTION         ;desired WDT rate
10.BCF      STATUS, RP0    ;Return to Bank 0
```

To change prescaler from the WDT to the TMR0 module, use the sequence shown in Example 6-2. This precaution must be taken even if the WDT is disabled.

**EXAMPLE 6-2: CHANGING PRESCALER (WDT→TIMER0)**

```
 CLRWDT                    ;Clear WDT and
                           ;prescaler
BSF          STATUS, RP0
MOVLW        b'xxxx0xxx'   ;Select TMR0, new
                           ;prescale value and
                           ;clock source
MOVWF        OPTION_REG
BCF         STATUS, RP0
```

**TABLE 6-1: REGISTERS ASSOCIATED WITH TIMER0**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR | Value on All Other RESETS |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 01h | TMR0 | Timer0 module register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 0Bh/8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 81h | OPTION | $\overline{\text{RBPU}}$ | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |
| 85h | TRISA | — | — | — | TRISA4 | TRISA3 | TRISA2 | TRISA1 | TRISA0 | ---1 1111 | ---1 1111 |

Legend: — = Unimplemented locations, read as '0', u = unchanged, x = unknown

**Note:** Shaded bits are not used by TMR0 module.



Downloaded from Arrow.com.

**NOTES:**


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 7.0 COMPARATOR MODULE

The comparator module contains two analog comparators. The inputs to the comparators are multiplexed with the RA0 through RA3 pins. The On-Chip Voltage Reference (Section 8.0) can also be an input to the comparators.

The CMCON register, shown in Register 7-1, controls the comparator input and output multiplexers. A block diagram of the comparator is shown in Figure 7-1.

**REGISTER 7-1: CMCON REGISTER (ADDRESS 1Fh)**

| R-0 | R-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| C2OUT | C1OUT | — | — | CIS | CM2 | CM1 | CM0 |
| bit 7 | | | | | | | bit 0 |

bit 7 **C2OUT**: Comparator 2 output
1 = C2 VIN+ > C2 VIN-
0 = C2 VIN+ < C2 VIN-

bit 6 **C1OUT**: Comparator 1 output
1 = C1 VIN+ > C1 VIN-
0 = C1 VIN+ < C1 VIN-

bit 5-4 **Unimplemented:** Read as '0'

bit 3 **CIS**: Comparator Input Switch
When CM<2:0>: = 001:
1 = C1 VIN- connects to RA3
0 = C1 VIN- connects to RA0
When CM<2:0> = 010:
1 = C1 VIN- connects to RA3
C2 VIN- connects to RA2
0 = C1 VIN- connects to RA0
C2 VIN- connects to RA1

bit 2-0 **CM<2:0>**: Comparator mode.

| 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 |



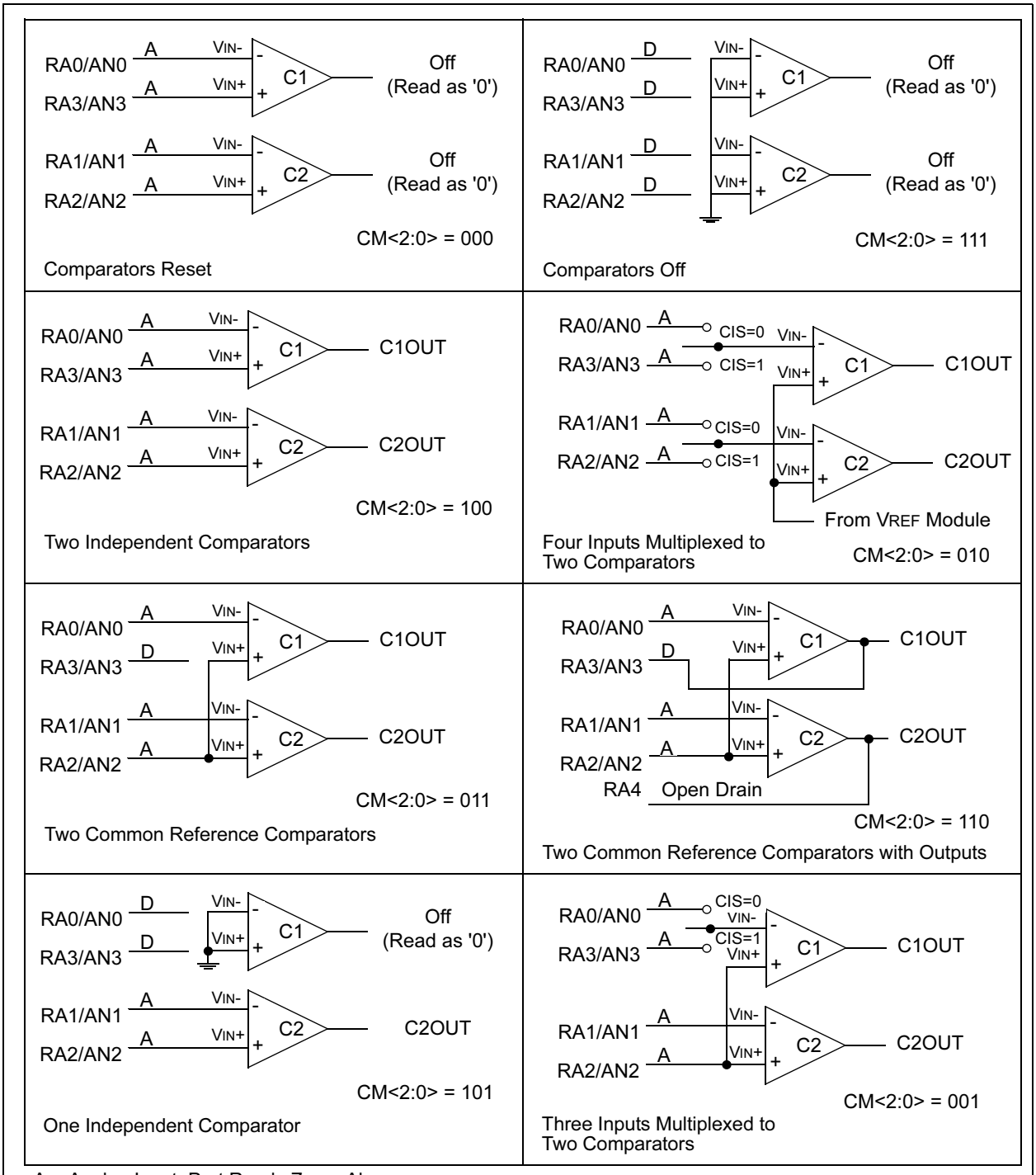
Downloaded from Arrow.com.

## 7.1 Comparator Configuration

There are eight modes of operation for the comparators. The CMCON register is used to select the mode. Figure 7-1 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 the specified mode change delay shown in Table 12-2.

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

**FIGURE 7-1: COMPARATOR I/O OPERATING MODES**




A = Analog Input, Port Reads Zeros Always
D = Digital Input
CIS = CMCON<3>, Comparator Input Switch


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

The code example in Example 7-1 depicts the steps required to configure the comparator module. RA3 and RA4 are configured as digital output. RA0 and RA1 are configured as the V- inputs and RA2 as the V+ input to both comparators.

**EXAMPLE 7-1: INITIALIZING COMPARATOR MODULE**

```
MOVLW   0x03          ;Init comparator mode
MOVWF   CMCON         ;CM<2:0> = 011
CLRF    PORTA         ;Init PORTA
BSF     STATUS,RP0    ;Select Bank1
MOVLW   0x07          ;Initialize data direction
MOVWF   TRISA         ;Set RA<2:0> as inputs
                      ;RA<4:3> as outputs
                      ;TRISA<7:5> always read '0'
BCF     STATUS,RP0    ;Select Bank 0
CALL    DELAY 10      ;10µs delay
MOVF    CMCON,F       ;Read CMCON to end change condition
BCF     PIR1,CMIF     ;Clear pending interrupts
BSF     STATUS,RP0    ;Select Bank 1
BSF     PIE1,CMIE     ;Enable comparator interrupts
BCF     STATUS,RP0    ;Select Bank 0
BSF     INTCON,PEIE   ;Enable peripheral interrupts
BSF     INTCON,GIE    ;Global interrupt enable
```

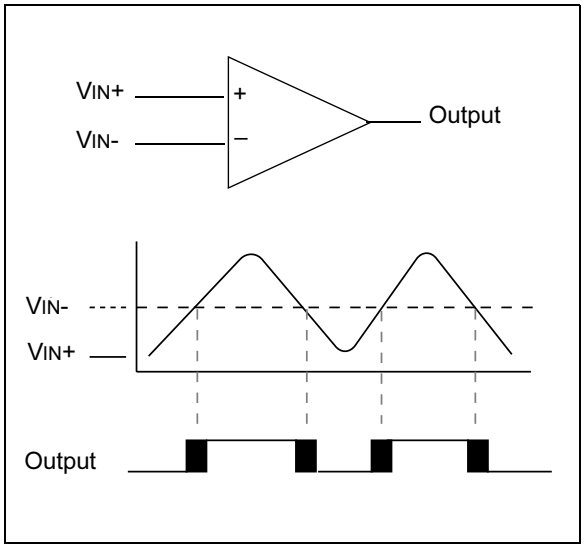
## 7.2 Comparator Operation

A single comparator is shown in Figure 7-2 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 7-2 represent the uncertainty due to input offsets and response time.

## 7.3 Comparator Reference

An external or internal reference signal may be used depending on the comparator Operating mode. The analog signal that is present at VIN- is compared to the signal at VIN+, and the digital output of the comparator is adjusted accordingly (Figure 7-2).

**FIGURE 7-2: SINGLE COMPARATOR**




### 7.3.1 EXTERNAL REFERENCE SIGNAL

When external voltage references are used, the comparator module can be configured to have the comparators operate from the same or different reference sources. However, threshold detector applications may require the same reference. The reference signal must be between VSS and VDD, and can be applied to either pin of the comparator(s).

### 7.3.2 INTERNAL REFERENCE SIGNAL

The comparator module also allows the selection of an internally generated voltage reference for the comparators. Section 10, Instruction Sets, contains a detailed description of the Voltage Reference Module that provides this signal. The internal reference signal is used when the comparators are in mode CM<2:0>=010 (Figure 7-1). In this mode, the internal voltage reference is applied to the VIN+ pin of both comparators.



Downloaded from Arrow.com.

## 7.4 Comparator Response Time

Response time is the minimum time, after selecting a new reference voltage or input source, before the comparator output has 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 12-2).

## 7.5 Comparator Outputs

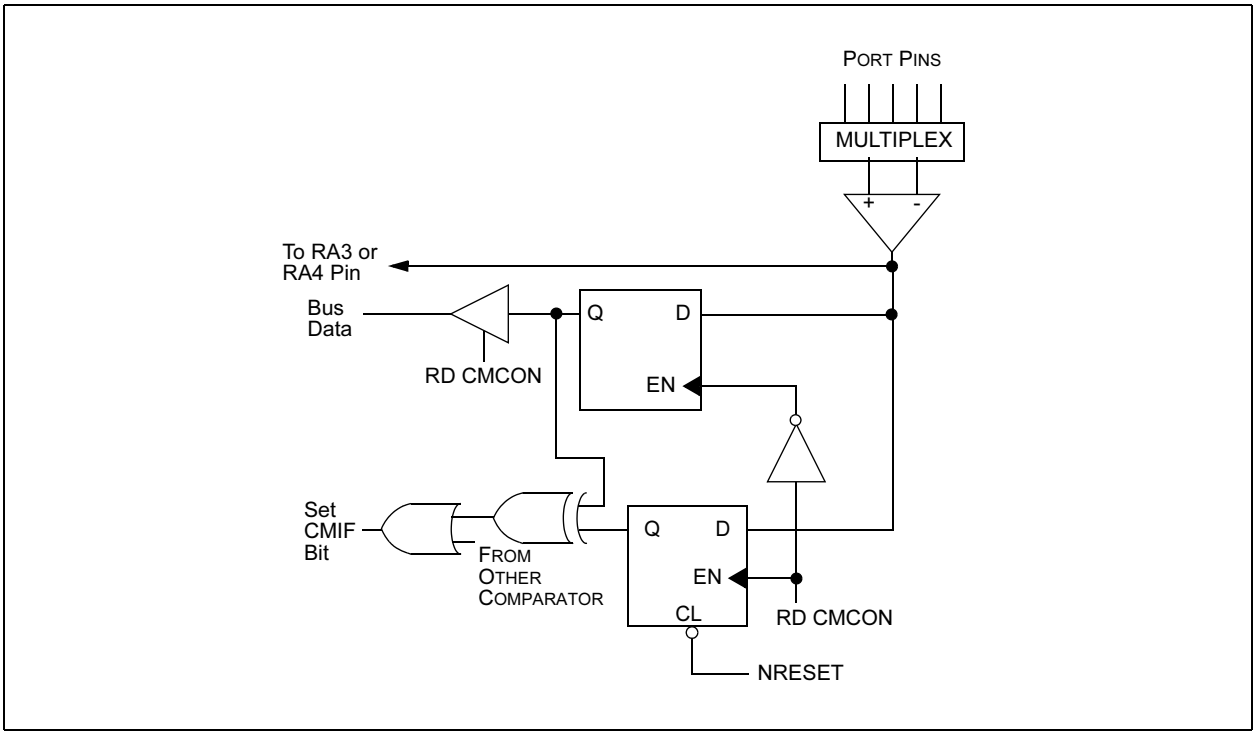
The comparator outputs are read through the CMCON register. These bits are read only. The comparator outputs may also be directly output to the RA3 and RA4 I/O pins. When the CM<2:0> = 110, multiplexors in the output path of the RA3 and RA4 pins will switch and the output of each pin will be the unsynchronized output of the comparator. The uncertainty of each of the comparators is related to the input offset voltage and the response time given in the specifications. Figure 7-3 shows the comparator output block diagram.

The TRISA bits will still function as an output enable/ disable for the RA3 and RA4 pins while in this mode.

**Note 1:** When reading the PORT 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 Schmitt Trigger input specification.

**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.

**FIGURE 7-3: COMPARATOR OUTPUT BLOCK DIAGRAM**





© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 7.6 Comparator Interrupts

The comparator interrupt flag is set whenever there is a change in the output value of either comparator. Software will need to maintain information about the status of the output bits, as read from CMCON<7:6>, to determine the actual change that has occurred. The CMIF bit, PIR1<6>, is the comparator interrupt flag. The CMIF bit must be RESET by clearing '0'. Since it is also possible to write a '1' to this register, a simulated interrupt may be initiated.

The CMIE bit (PIE1<6>) 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 clear, the interrupt is not enabled, though the CMIF bit will still be set if an interrupt condition occurs.

| **Note:** | If a change in the CMCON register (C1OUT or C2OUT) should occur when a read operation is being executed (start of the Q2 cycle), then the CMIF (PIR1<6>) interrupt flag may not get set. |
|---|---|

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.

## 7.7 Comparator Operation During SLEEP

When a comparator is active and the device is placed in SLEEP mode, the comparator remains active and the interrupt is functional if enabled. This interrupt will wake up the device from SLEEP mode when enabled. While the comparator is powered-up, higher SLEEP currents than shown in the power-down current specification will occur. Each comparator that is operational will consume additional current as shown in the comparator specifications. To minimize power consumption while in SLEEP mode, turn off the comparators, CM<2:0> = 111, before entering SLEEP. If the device wakes up from SLEEP, the contents of the CMCON register are not affected.

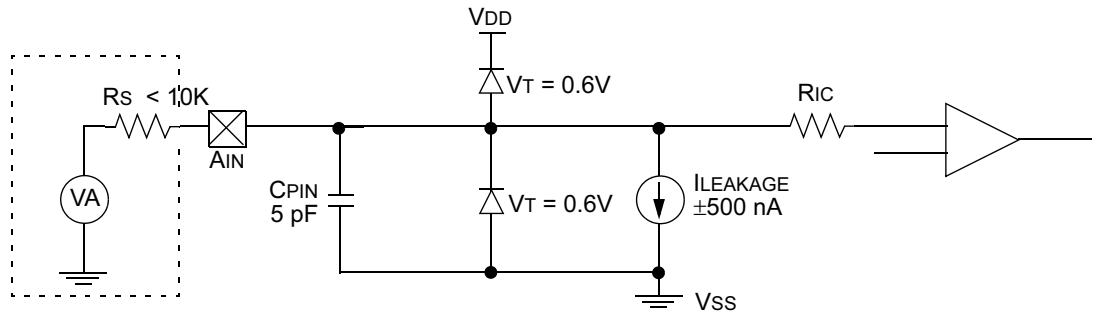
## 7.8 Effects of a RESET

A device RESET forces the CMCON register to its RESET state. This forces the comparator module to be in the comparator RESET mode, CM<2:0> = 000. This ensures that all potential inputs are analog inputs. Device current is minimized when analog inputs are present at RESET time. The comparators will be powered-down during the RESET interval.

## 7.9 Analog Input Connection Considerations

A simplified circuit for an analog input is shown in Figure 7-4. 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 kΩ 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 7-4: ANALOG INPUT MODEL**




| 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 |



Downloaded from Arrow.com.

**TABLE 7-1: REGISTERS ASSOCIATED WITH COMPARATOR MODULE**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR | Value on All Other RESETS |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1Fh | CMCON | C2OUT | C1OUT | — | — | CIS | CM2 | CM1 | CM0 | 00-- 0000 | 00-- 0000 |
| 9Fh | VRCON | VREN | VROE | VRR | — | VR3 | VR2 | VR1 | VR0 | 000- 0000 | 000- 0000 |
| 0Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | CMIF | — | — | — | — | — | — | -0-- ---- | -0-- ---- |
| 8Ch | PIE1 | — | CMIE | — | — | — | — | — | — | -0-- ---- | -0-- ---- |
| 85h | TRISA | — | — | — | TRISA4 | TRISA3 | TRISA2 | TRISA1 | TRISA0 | ---1 1111 | ---1 1111 |

Legend: x = unknown, u = unchanged, - = unimplemented, read as "0"


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 8.0 VOLTAGE REFERENCE MODULE

The Voltage Reference is a 16-tap resistor ladder network that provides a selectable voltage reference. The resistor ladder is segmented to provide two ranges of VREF values and has a power-down function to conserve power when the reference is not being used. The VRCON register controls the operation of the reference as shown in Register 8-1. The block diagram is given in Figure 8-1.

### 8.1 Configuring the Voltage Reference

The Voltage Reference can output 16 distinct voltage levels for each range. The equations used to calculate the output of the Voltage Reference are as follows:

if VRR = 1: VREF = (VR<3:0>/24) x VDD

if VRR = 0: VREF = (VDD x 1/4) + (VR<3:0>/32) x VDD

The setting time of the Voltage Reference must be considered when changing the VREF output (Table 12-1). Example 8-1 shows an example of how to configure the Voltage Reference for an output voltage of 1.25V with VDD = 5.0V.

**REGISTER 8-1: VRCON REGISTER(ADDRESS 9Fh)**

| R/W-0 | R/W-0 | R/W-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| VREN | VROE | VRR | — | VR3 | VR2 | VR1 | VR0 |
| bit 7 | | | | | | | bit 0 |

bit 7 **VREN:** VREF Enable
1 = VREF circuit powered on
0 = VREF circuit powered down, no IDD drain

bit 6 **VROE:** VREF Output Enable
1 = VREF is output on RA2 pin
0 = VREF is disconnected from RA2 pin

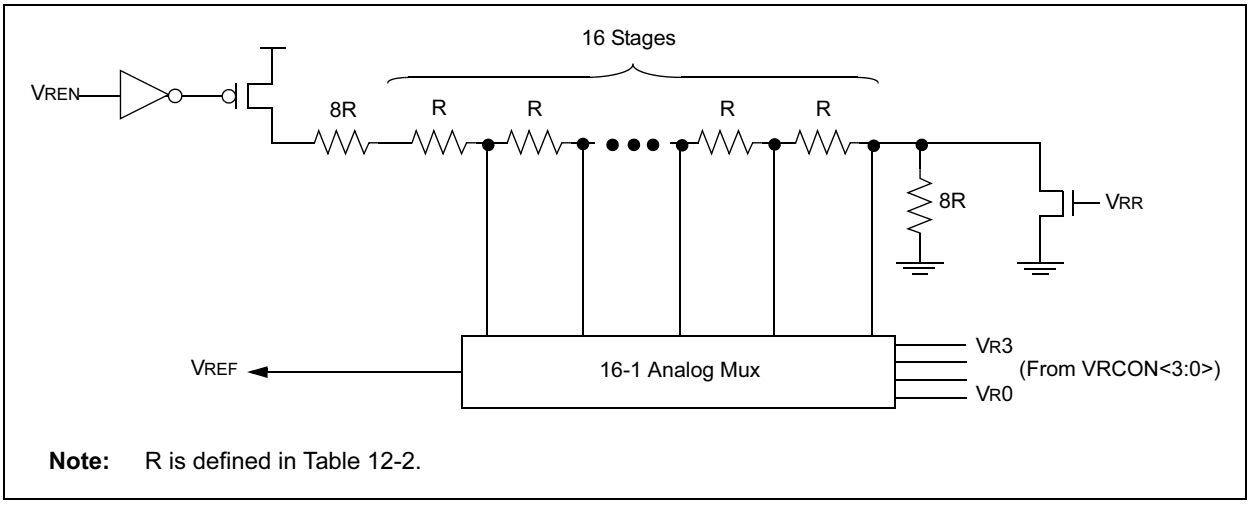
bit 5 **VRR:** VREF Range selection
1 = Low Range
0 = High Range

bit 4 **Unimplemented:** Read as '0'

bit 3-0 **VR<3:0>**: VREF value selection 0 ≤ VR [3:0] ≤ 15
when VRR = 1: VREF = (VR<3:0>/ 24) * VDD
when VRR = 0: VREF = 1/4 * VDD + (VR<3:0>/ 32) * VDD

| 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 |

**FIGURE 8-1: VOLTAGE REFERENCE BLOCK DIAGRAM**




**Note:** R is defined in Table 12-2.



Downloaded from Arrow.com.

**EXAMPLE 8-1: VOLTAGE REFERENCE CONFIGURATION**

```
MOVLW     0x02          ; 4 Inputs Muxed
MOVWF     CMCON         ; to 2 comps.
BSF       STATUS,RP0    ; go to Bank 1
MOVLW     0x0F          ; RA3-RA0 are
MOVWF     TRISA         ; inputs
MOVLW     0xA6          ; enable VREF
MOVWF     VRCON         ; low range
                        ; set VR<3:0>=6
BCF       STATUS,RP0    ; go to Bank 0
CALL      DELAY10       ; 10µs delay
```

## 8.2 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 8-1) keep VREF from approaching VSS or VDD. The voltage reference is VDD derived and therefore, the VREF output changes with fluctuations in VDD. The tested absolute accuracy of the voltage reference can be found in Table 12-2.

## 8.3 Operation During SLEEP

When the device wakes up from SLEEP through an interrupt or a Watchdog Timer time-out, the contents of the VRCON register are not affected. To minimize current consumption in SLEEP mode, the voltage reference should be disabled.

## 8.4 Effects of a RESET

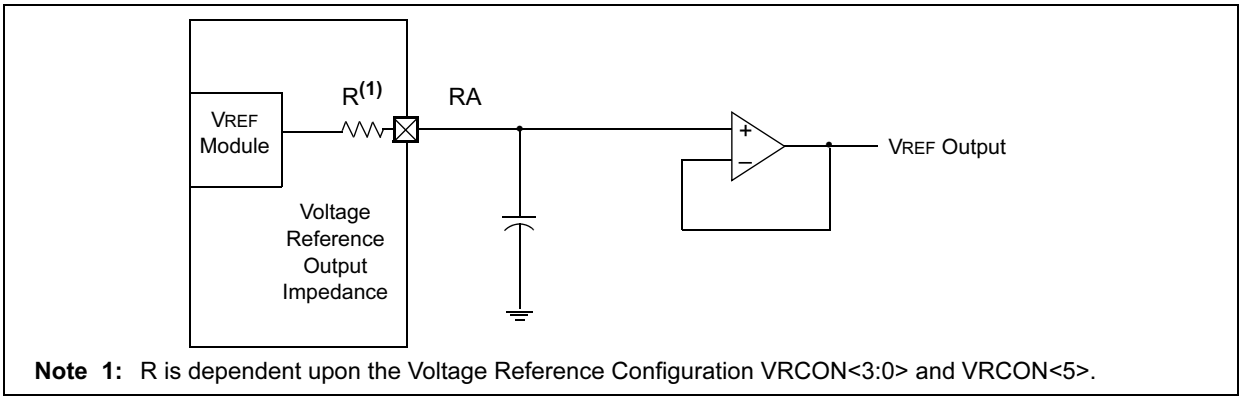
A device RESET disables the voltage reference by clearing bit VREN (VRCON<7>). This reset also disconnects the reference from the RA2 pin by clearing bit VROE (VRCON<6>) and selects the high voltage range by clearing bit VRR (VRCON<5>). The VREF value select bits, VRCON<3:0>, are also cleared.

## 8.5 Connection Considerations

The voltage reference module operates independently of the comparator module. The output of the reference generator may be connected to the RA2 pin if the TRISA<2> bit is set and the VROE bit, VRCON<6>, is set. Enabling the voltage reference output onto the RA2 pin with an input signal present will increase current consumption. Connecting RA2 as a digital output with VREF enabled will also increase current consumption.

The RA2 pin can be used as a simple D/A output with limited drive capability. Due to the limited drive capability, a buffer must be used in conjunction with the voltage reference output for external connections to VREF. Figure 8-2 shows an example buffering technique.

**FIGURE 8-2: VOLTAGE REFERENCE OUTPUT BUFFER EXAMPLE**




**Note 1:** R is dependent upon the Voltage Reference Configuration VRCON<3:0> and VRCON<5>.

**TABLE 8-1: REGISTERS ASSOCIATED WITH VOLTAGE REFERENCE**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value On POR | Value On All Other RESETS |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 9Fh | VRCON | VREN | VROE | VRR | — | VR3 | VR2 | VR1 | VR0 | 000- 0000 | 000- 0000 |
| 1Fh | CMCON | C2OUT | C1OUT | — | — | CIS | CM2 | CM1 | CM0 | 00-- 0000 | 00-- 0000 |
| 85h | TRISA | — | — | — | TRISA4 | TRISA3 | TRISA2 | TRISA1 | TRISA0 | ---1 1111 | ---1 1111 |

**Note:** - = Unimplemented, read as "0"


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 9.0 SPECIAL FEATURES OF THE CPU

Special circuits to deal with the needs of real-time applications are what sets a microcontroller apart from other processors. The PIC16C62X 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 are:

1. OSC selection
2. RESET
   Power-on Reset (POR)
   Power-up Timer (PWRT)
   Oscillator Start-up Timer (OST)
   Brown-out Reset (BOR)
3. Interrupts
4. Watchdog Timer (WDT)
5. SLEEP
6. Code protection
7. ID Locations
8. In-Circuit Serial Programming™

The PIC16C62X devices have a Watchdog Timer which 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 provides 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 or through an interrupt. Several oscillator options are also made available to allow the part to fit the application. The RC oscillator option saves system cost, while the LP crystal option saves power. A set of configuration bits are used to select various options.



Downloaded from Arrow.com.

## 9.1 Configuration Bits

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

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

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

| CP1 | CP0 (2) | CP1 | $\overline{CP0}$ (2) | $\overline{CP1}$ | CP0 (2) | | BODEN | CP1 | CP0 (2) | $\overline{PWRTE}$ | WDTE | F0SC1 | F0SC0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| bit 13 | | | | | | | | | | | | | bit 0 |

bit 13-8, 5-4: **CP<1:0>:** Code protection bit pairs (2)

Code protection for 2K program memory
11 = Program memory code protection off
10 = 0400h-07FFh code protected
01 = 0200h-07FFh code protected
00 = 0000h-07FFh code protected

Code protection for 1K program memory
11 = Program memory code protection off
10 = Program memory code protection off
01 = 0200h-03FFh code protected
00 = 0000h-03FFh code protected

Code protection for 0.5K program memory
11 = Program memory code protection off
10 = Program memory code protection off
01 = Program memory code protection off
00 = 0000h-01FFh code protected

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

bit 6 **BODEN**: Brown-out Reset Enable bit (1)
1 = BOR enabled
0 = BOR disabled

bit 3 **$\overline{PWRTE}$**: Power-up Timer Enable bit (1, 3)
1 = PWRT disabled
0 = PWRT enabled

bit 2 **WDTE**: Watchdog Timer Enable bit
1 = WDT enabled
0 = WDT disabled

bit 1-0 **FOSC1:FOSC0**: Oscillator Selection bits
11 = RC oscillator
10 = HS oscillator
01 = XT oscillator
00 = LP oscillator

**Note 1:** Enabling Brown-out Reset automatically enables Power-up Timer (PWRT) regardless of the value of bit $\overline{PWRTE}$. Ensure the Power-up Timer is enabled anytime Brown-out Detect Reset is enabled.
**2:** All of the CP<1:0> pairs have to be given the same value to enable the code protection scheme listed.
**3:** Unprogrammed parts default the Power-up Timer disabled.

| 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 |


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 9.2 Oscillator Configurations

### 9.2.1 OSCILLATOR TYPES

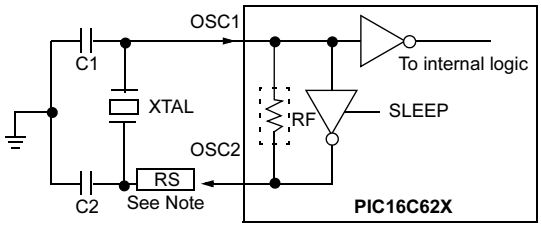
The PIC16C62X devices can be operated in four different oscillator options. The user can program two configuration bits (FOSC1 and FOSC0) to select one of these four modes:

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

### 9.2.2 CRYSTAL OSCILLATOR / CERAMIC RESONATORS

In XT, LP or HS modes, a crystal or ceramic resonator is connected to the OSC1 and OSC2 pins to establish oscillation (Figure 9-1). The PIC16C62X oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may give a frequency out of the crystal manufacturers specifications. When in XT, LP or HS modes, the device can have an external clock source to drive the OSC1 pin (Figure 9-2).

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




See Table 9-1 and Table 9-2 for recommended values of C1 and C2.

**Note:** A series resistor may be required for AT strip cut crystals.

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

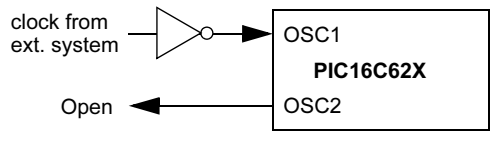



**TABLE 9-1: CAPACITOR SELECTION FOR CERAMIC RESONATORS**

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

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**

| Mode | Freq | OSC1(C1) | OSC2(C2) |
|---|---|---|---|
| LP | 32 kHz<br>200 kHz | 68 - 100 pF<br>15 - 30 pF | 68 - 100 pF<br>15 - 30 pF |
| XT | 100 kHz<br>2 MHz<br>4 MHz | 68 - 150 pF<br>15 - 30 pF<br>15 - 30 pF | 150 - 200 pF<br>15 - 30 pF<br>15 - 30 pF |
| HS | 8 MHz<br>10 MHz<br>20 MHz | 15 - 30 pF<br>15 - 30 pF<br>15 - 30 pF | 15 - 30 pF<br>15 - 30 pF<br>15 - 30 pF |

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 overdriving crystals with low drive level specification. Since each crystal has its own characteristics, the user should consult the crystal manufacturer for appropriate values of external components.

PRELIMINARY



Downloaded from Arrow.com.

### 9.2.3 EXTERNAL CRYSTAL OSCILLATOR CIRCUIT

Either a prepackaged oscillator can be used or a simple oscillator circuit with TTL gates can be built. Prepackaged oscillators provide a wide operating range and better stability. A well-designed crystal oscillator will provide good performance with TTL gates. Two types of crystal oscillator circuits can be used; one with series resonance or one with parallel resonance.

Figure 9-3 shows implementation of a parallel resonant oscillator circuit. The circuit is designed to use the fundamental frequency of the crystal. The 74AS04 inverter performs the 180° phase shift that a parallel oscillator requires. The 4.7 kΩ resistor provides the negative feedback for stability. The 10 kΩ potentiometers bias the 74AS04 in the linear region. This could be used for external oscillator designs.

**FIGURE 9-3: EXTERNAL PARALLEL RESONANT CRYSTAL OSCILLATOR CIRCUIT**

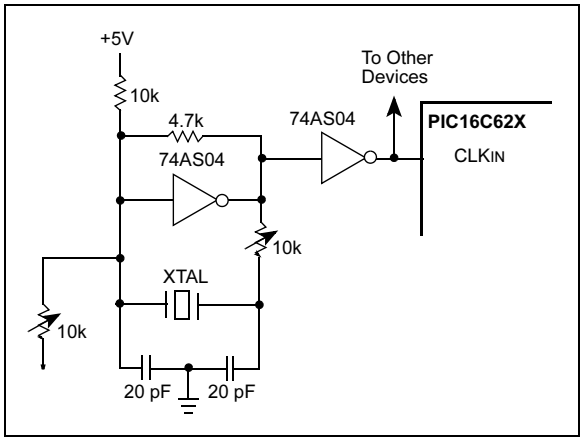
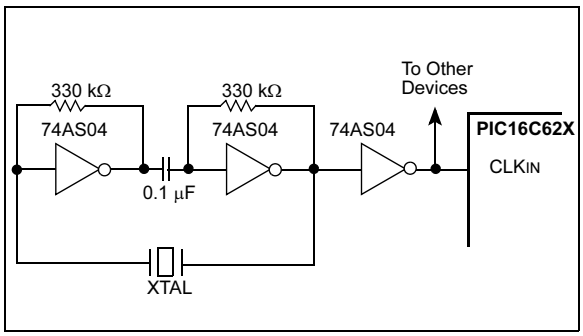



Figure 9-4 shows a series resonant oscillator circuit. This circuit is also designed to use the fundamental frequency of the crystal. The inverter performs a 180° phase shift in a series resonant oscillator circuit. The 330 kΩ resistors provide the negative feedback to bias the inverters in their linear region.

**FIGURE 9-4: EXTERNAL SERIES RESONANT CRYSTAL OSCILLATOR CIRCUIT**




### 9.2.4 RC OSCILLATOR

For timing insensitive applications the "RC" device option offers additional cost savings. The RC oscillator frequency is a function of the supply voltage, the resistor (REXT) and capacitor (CEXT) values, and the operating temperature. In addition to this, the oscillator frequency will vary from unit to unit due to normal process parameter variation. Furthermore, the difference in lead frame capacitance between package types will also affect the oscillation frequency, especially for low CEXT values. The user also needs to take into account variation due to tolerance of external R and C components used. Figure 9-5 shows how the R/C combination is connected to the PIC16C62X. For REXT values below 2.2 kΩ, the oscillator operation may become unstable or stop completely. For very high REXT values (e.g., 1 MΩ), the oscillator becomes sensitive to noise, humidity and leakage. Thus, we recommend to keep REXT between 3 kΩ and 100 kΩ.

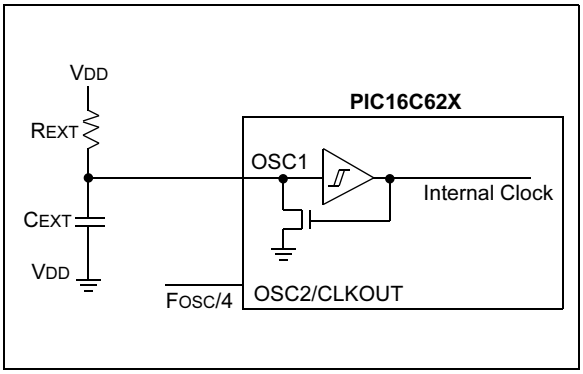
Although the oscillator will operate with no external capacitor (CEXT = 0 pF), we recommend using values above 20 pF for noise and stability reasons. With no or small external capacitance, the oscillation frequency can vary dramatically due to changes in external capacitances, such as PCB trace capacitance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

See Section 13.0 for RC frequency variation from part to part due to normal process variation. The variation is larger for larger R (since leakage current variation will affect RC frequency more for large R) and for smaller C (since variation of input capacitance will affect RC frequency more).

See Section 13.0 for variation of oscillator frequency due to VDD for given REXT/CEXT values, as well as frequency variation due to operating temperature for given R, C and VDD values.

The oscillator frequency, divided by 4, is available on the OSC2/CLKOUT pin, and can be used for test purposes or to synchronize other logic (Figure 3-2 for waveform).

**FIGURE 9-5: RC OSCILLATOR MODE**





© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 9.3 RESET

The PIC16C62X differentiates between various kinds of RESET:

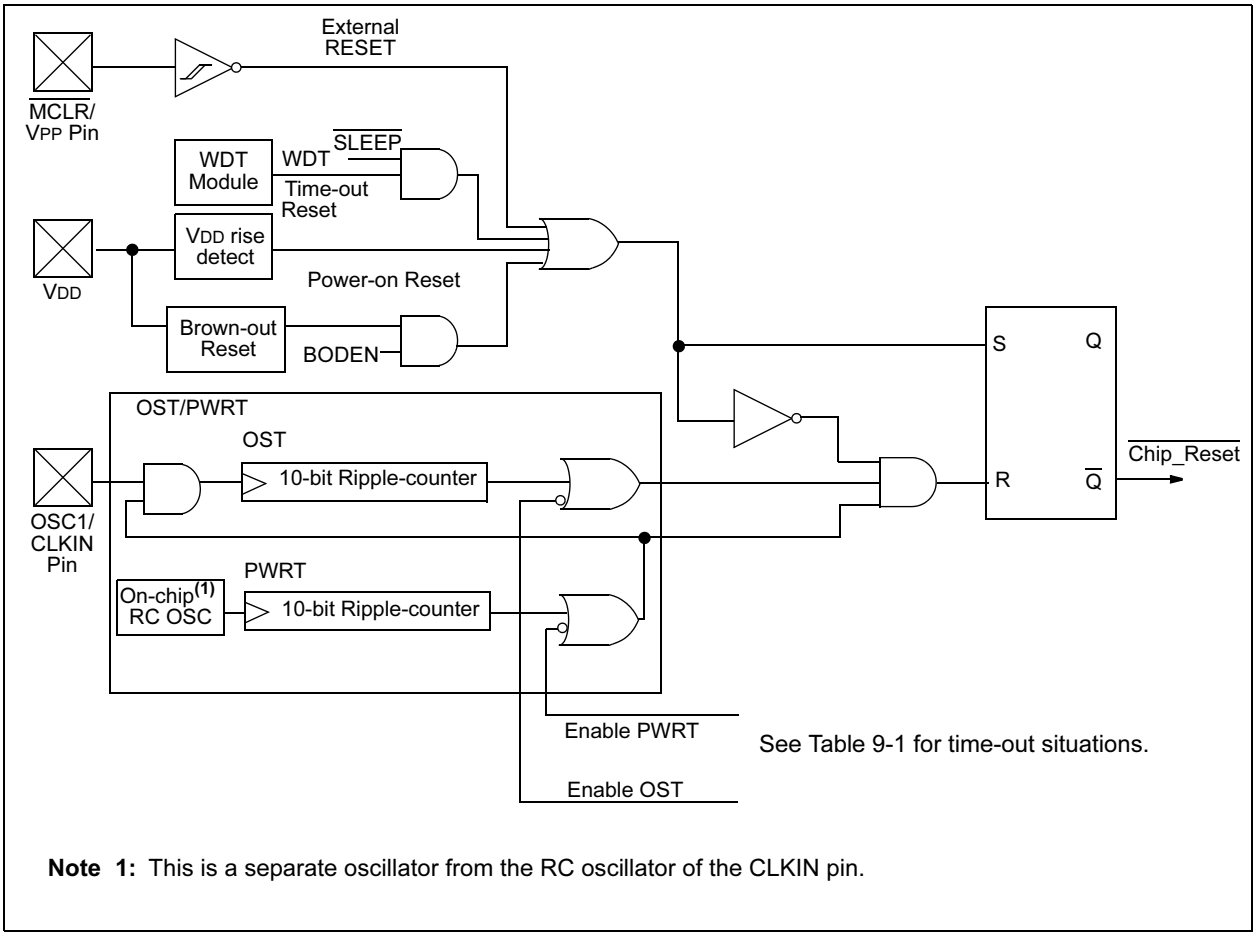
a) Power-on Reset (POR)
b) $\overline{MCLR}$ Reset during normal operation
c) $\overline{MCLR}$ Reset during SLEEP
d) WDT Reset (normal operation)
e) WDT wake-up (SLEEP)
f) Brown-out Reset (BOR)

Some registers are not affected in any RESET condition Their status is unknown on POR and unchanged in any other RESET. Most other registers are reset to a "RESET state" on Power-on Reset, $\overline{MCLR}$ Reset, WDT Reset and $\overline{MCLR}$ Reset during SLEEP. They are not affected by a WDT wake-up, since this is viewed as the resumption of normal operation. $\overline{TO}$ and $\overline{PD}$ bits are set or cleared differently in different RESET situations as indicated in Table 9-2. These bits are used in software to determine the nature of the RESET. See Table 9-5 for a full description of RESET states of all registers.

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

The $\overline{MCLR}$ Reset path has a noise filter to detect and ignore small pulses. See Table 12-5 for pulse width specification.

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




**Note 1:** This is a separate oscillator from the RC oscillator of the CLKIN pin.



Downloaded from Arrow.com.

## 9.4 Power-on Reset (POR), Power-up Timer (PWRT), Oscillator Start-up Timer (OST) and Brown-out Reset (BOR)

### 9.4.1 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, just 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 required. See Electrical Specifications for details.

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 (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”.

### 9.4.2 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 RC 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, PWRTE can disable (if set) or enable (if cleared or programmed) the Power-up Timer. The Power-up Timer should always be enabled when Brown-out Reset is enabled.

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

### 9.4.3 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.

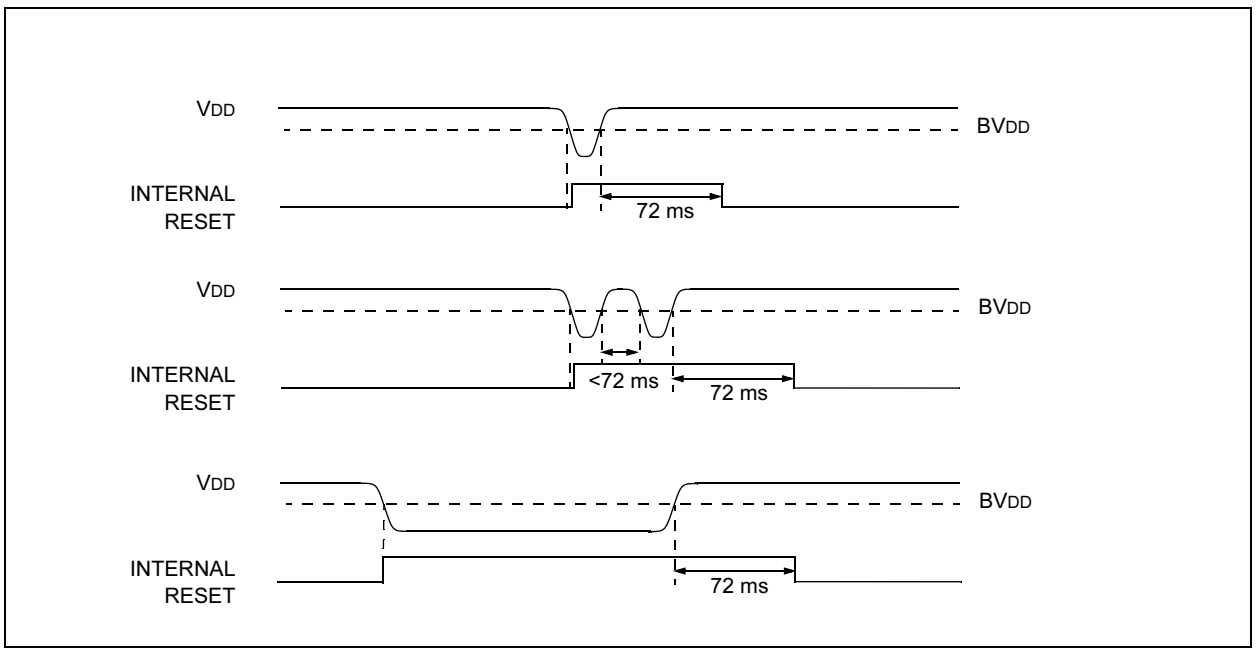
### 9.4.4 BROWN-OUT RESET (BOR)

The PIC16C62X members have on-chip Brown-out Reset circuitry. A configuration bit, BODEN, can disable (if clear/programmed) or enable (if set) the Brown-out Reset circuitry. If VDD falls below 4.0V refer to VBOR parameter D005 (VBOR) for greater than parameter (TBOR) in Table 12-5. The brown-out situation will RESET the chip. A RESET won’t occur if VDD falls below 4.0V for less than parameter (TBOR).

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

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. The Power-up Timer should always be enabled when Brown-out Reset is enabled. Figure 9-7 shows typical Brown-out situations.

**FIGURE 9-7: BROWN-OUT SITUATIONS**





© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

### 9.4.5 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 PWRTE bit status. For example, in RC mode with PWRTE bit erased (PWRT disabled), there will be no time-out at all. Figure 9-8, Figure 9-9 and Figure 9-10 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 9-9). This is useful for testing purposes or to synchronize more than one PIC16C62X device operating in parallel.

Table 9-4 shows the RESET conditions for some special registers, while Table 9-5 shows the RESET conditions for all the registers.

### 9.4.6 POWER CONTROL (PCON)/ STATUS REGISTER

The power control/STATUS register, PCON (address 8Eh), has two bits.

Bit0 is BOR (Brown-out). BOR is unknown on Power-on Reset. It must then be set by the user and checked on subsequent RESETS to see if BOR = 0, indicating that a brown-out has occurred. The BOR 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 ($V_{DD}$ may have gone too low).

**TABLE 9-1: TIME-OUT IN VARIOUS SITUATIONS**

| Oscillator Configuration | Power-up | | Brown-out Reset | Wake-up from SLEEP |
|---|---|---|---|---|
| | PWRTE = 0 | PWRTE = 1 | | |
| XT, HS, LP | 72 ms + 1024 TOSC | 1024 TOSC | 72 ms + 1024 TOSC | 1024 TOSC |
| RC | 72 ms | — | 72 ms | — |

**TABLE 9-2: STATUS/PCON BITS AND THEIR SIGNIFICANCE**

| POR | BOR | TO | PD | |
|---|---|---|---|---|
| 0 | x | 1 | 1 | Power-on Reset |
| 0 | x | 0 | x | Illegal, TO is set on POR |
| 0 | x | x | 0 | Illegal, PD is set on POR |
| 1 | 0 | x | x | Brown-out Reset |
| 1 | 1 | 0 | u | WDT Reset |
| 1 | 1 | 0 | 0 | WDT Wake-up |
| 1 | 1 | u | u | MCLR Reset during normal operation |
| 1 | 1 | 1 | 0 | MCLR Reset during SLEEP |

Legend: u = unchanged, x = unknown

**TABLE 9-3: SUMMARY OF REGISTERS ASSOCIATED WITH BROWN-OUT**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR Reset | Value on all other RESETS(1) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 83h | STATUS | | | | TO | PD | | | | 0001 1xxx | 000q quuu |
| 8Eh | PCON | — | — | — | — | — | — | POR | BOR | ---- --0x | ---- --uq |

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

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



Downloaded from Arrow.com.

**TABLE 9-4: INITIALIZATION CONDITION FOR SPECIAL REGISTERS**

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

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 9-5: INITIALIZATION CONDITION FOR REGISTERS**

| Register | Address | Power-on Reset | • MCLR Reset during normal operation<br>• MCLR Reset during SLEEP<br>• WDT Reset<br>• Brown-out Reset (1) | • Wake-up from SLEEP through interrupt<br>• Wake-up from SLEEP through WDT time-out |
|---|---|---|---|---|
| W | — | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| INDF | 00h | — | — | — |
| TMR0 | 01h | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PCL | 02h | 0000 0000 | 0000 0000 | PC + 1(3) |
| STATUS | 03h | 0001 1xxx | 000q quuu(4) | uuuq quuu(4) |
| FSR | 04h | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PORTA | 05h | ---x xxxx | ---u uuuu | ---u uuuu |
| PORTB | 06h | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| CMCON | 1Fh | 00-- 0000 | 00-- 0000 | uu-- uuuu |
| PCLATH | 0Ah | ---0 0000 | ---0 0000 | ---u uuuu |
| INTCON | 0Bh | 0000 000x | 0000 000u | uuuu uqqq(2) |
| PIR1 | 0Ch | -0-- ---- | -0-- ---- | -q-- ----(2,5) |
| OPTION | 81h | 1111 1111 | 1111 1111 | uuuu uuuu |
| TRISA | 85h | ---1 1111 | ---1 1111 | ---u uuuu |
| TRISB | 86h | 1111 1111 | 1111 1111 | uuuu uuuu |
| PIE1 | 8Ch | -0-- ---- | -0-- ---- | -u-- ---- |
| PCON | 8Eh | ---- --0x | ---- --uq(1,6) | ---- --uu |
| VRCON | 9Fh | 000- 0000 | 000- 0000 | uuu- uuuu |

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

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

**2:** One or more bits in INTCON, PIR1 and/or PIR2 will be affected (to cause wake-up).

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

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

**5:** If wake-up was due to comparator input changing, then bit 6 = 1. All other interrupts generating a wake-up will cause bit 6 = u.

**6:** If RESET was due to brown-out, then bit 0 = 0. All other RESETS will cause bit 0 = u.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

**FIGURE 9-8: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ NOT TIED TO $V_{DD}$): CASE 1**

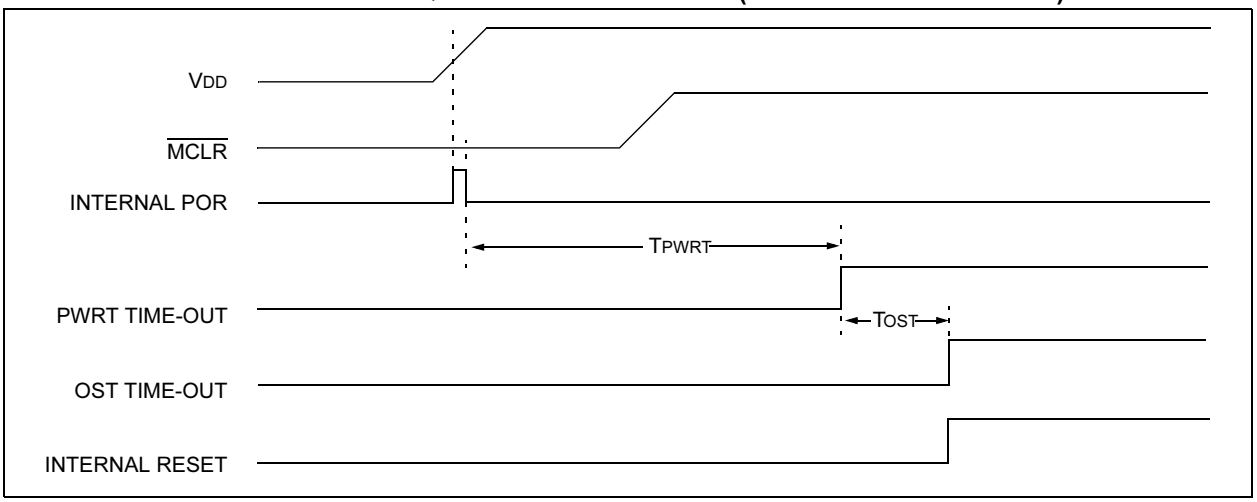



**FIGURE 9-9: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ NOT TIED TO $V_{DD}$): CASE 2**

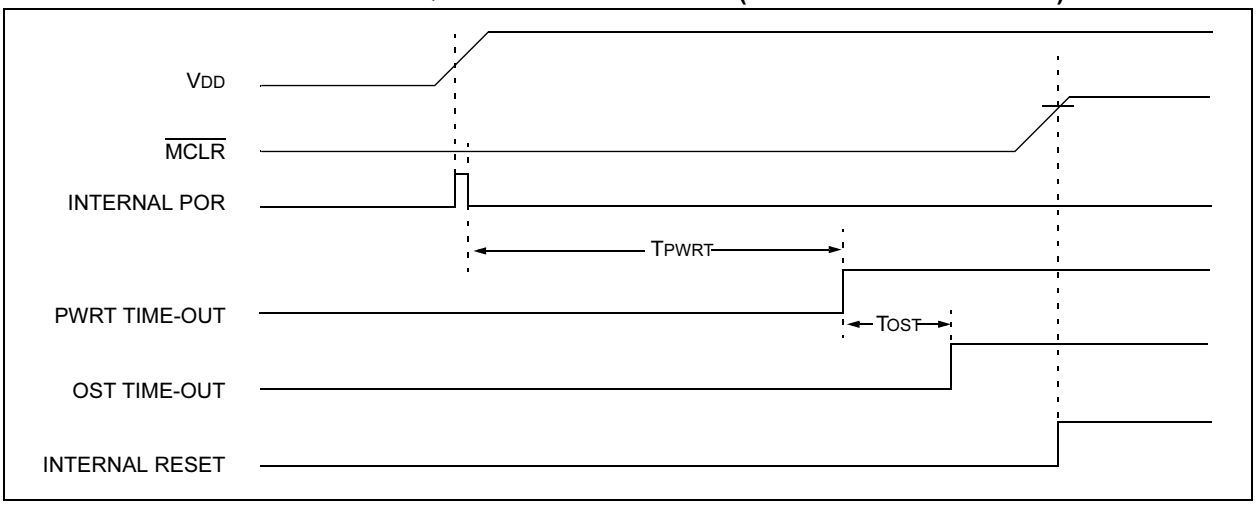
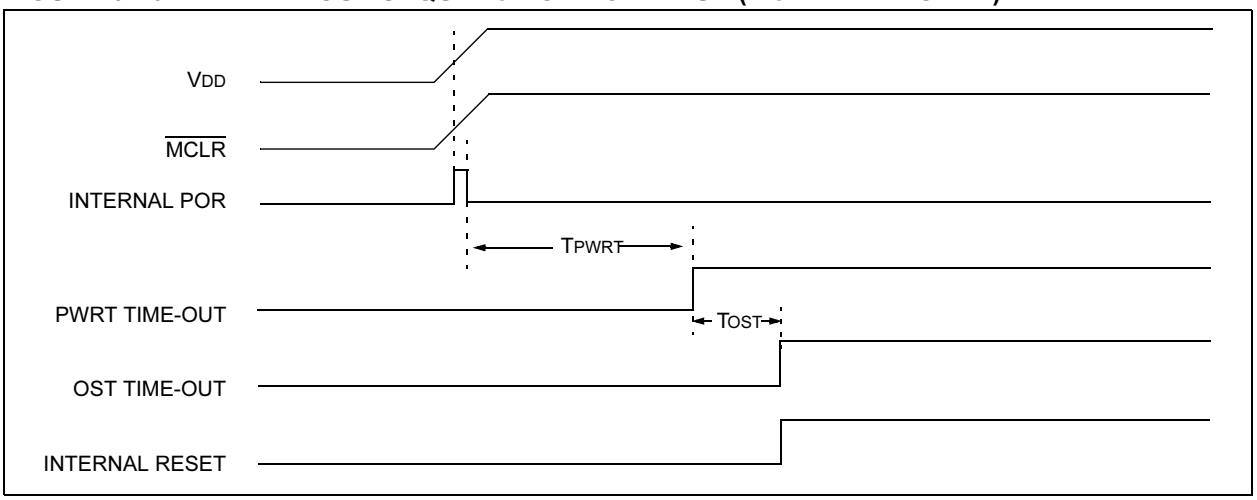


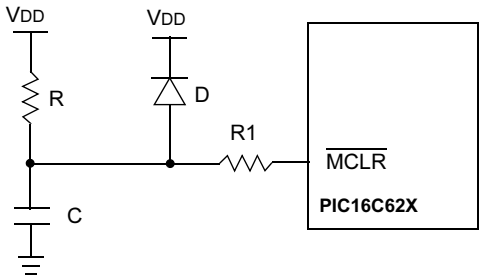

**FIGURE 9-10: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ TIED TO $V_{DD}$)**






Downloaded from Arrow.com.

**FIGURE 9-11: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW VDD POWER-UP)**

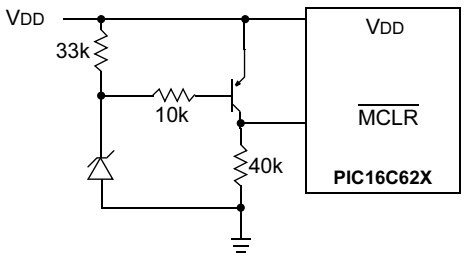



**Note 1:** External Power-on Reset circuit is required only if VDD power-up slope is too slow. The diode D helps discharge the capacitor quickly when VDD powers down.

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

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

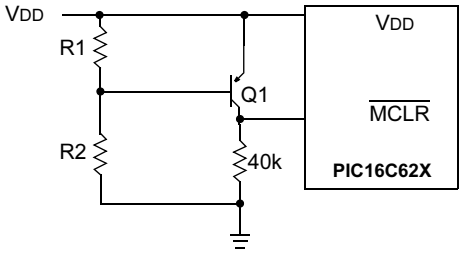
**FIGURE 9-12: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 1**




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

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

**FIGURE 9-13: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 2**



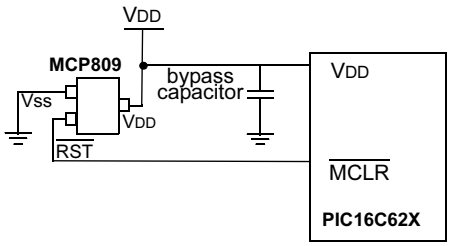

**Note 1:** This brown-out circuit is less expensive, albeit less accurate. Transistor Q1 turns off when VDD is below a certain level such that:

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

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

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

**FIGURE 9-14: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 3**




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


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 9.5 Interrupts

The PIC16C62X has 4 sources of interrupt:

- External interrupt RB0/INT
- TMR0 overflow interrupt
- PORTB change interrupts (pins RB<7:4>)
- Comparator interrupt

The interrupt control register (INTCON) records 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 un-masked interrupts or disables (if cleared) all interrupts. Individual interrupts can be disabled through their corresponding enable bits in INTCON register. GIE is cleared on RESET.

The "return from interrupt" instruction, `RETFIE`, exits interrupt routine, as well as sets the GIE bit, which re-enable RB0/INT interrupts.

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

The peripheral interrupt flag is contained in the special register PIR1. The corresponding interrupt enable bit is contained in special registers PIE1.

When an interrupt is responded to, the GIE is cleared to disable any further interrupt, the return address is pushed into the stack and the PC is loaded with 0004h.

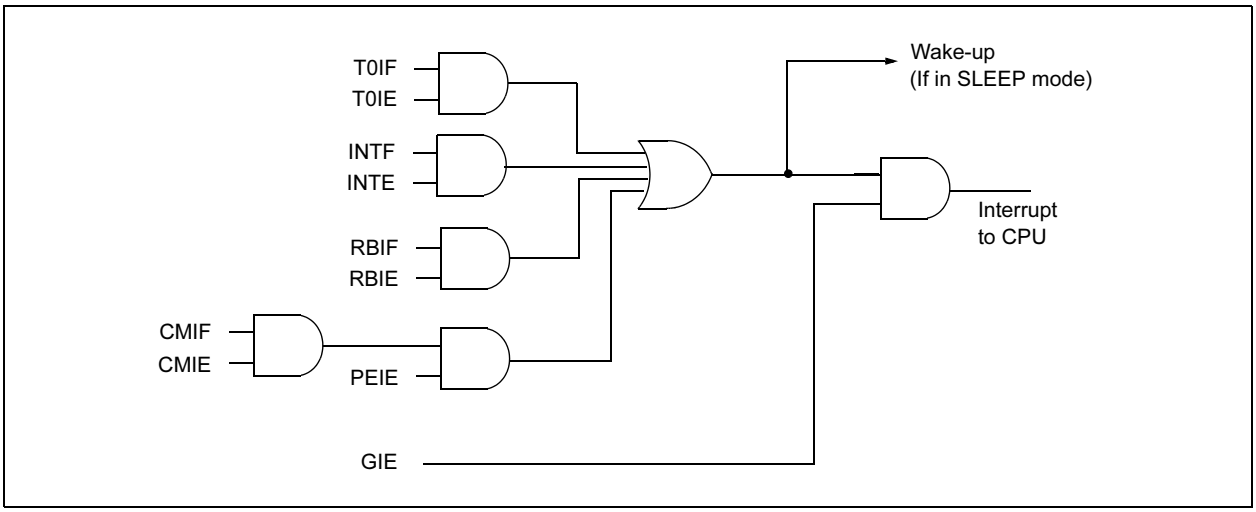
Once in the interrupt service routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid RB0/INT recursive interrupts.

For external interrupt events, such as the INT pin or PORTB change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends when the interrupt event occurs (Figure 9-16). The latency is the same for one or two cycle instructions. 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.

**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 CPU will execute a `NOP` in the cycle immediately following the instruction which clears the GIE bit. The interrupts which were ignored are still pending to be serviced when the GIE bit is set again.

**FIGURE 9-15: INTERRUPT LOGIC**






Downloaded from Arrow.com.

### 9.5.1 RB0/INT INTERRUPT

External interrupt on RB0/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 RB0/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 RB0/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 9.8 for details on SLEEP and Figure 9-18 for timing of wake-up from SLEEP through RB0/INT interrupt.

### 9.5.2 TMR0 INTERRUPT

An overflow (FFh → 00h) in the TMR0 register will set the T0IF (INTCON<2>) bit. The interrupt can be enabled/disabled by setting/clearing T0IE (INTCON<5>) bit. For operation of the Timer0 module, see Section 6.0.

### 9.5.3 PORTB INTERRUPT

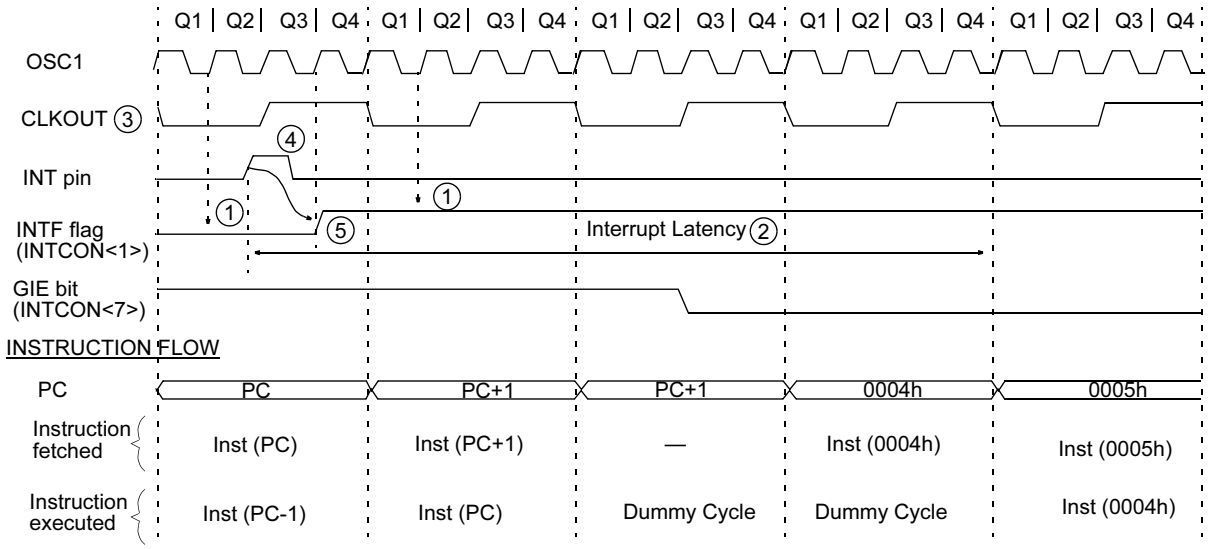
An input change on PORTB <7:4> sets the RBIF (INTCON<0>) bit. The interrupt can be enabled/disabled by setting/clearing the RBIE (INTCON<4>) bit. For operation of PORTB (Section 5.2).

| | |
|---|---|
| **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 RBIF interrupt flag may not get set. |

### 9.5.4 COMPARATOR INTERRUPT

See Section 7.6 for complete description of comparator interrupts.

**FIGURE 9-16: 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 two-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 anytime during the Q4-Q1 cycles.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

**TABLE 9-6: SUMMARY OF INTERRUPT REGISTERS**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR Reset | Value on all other RESETS(1) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | CMIF | — | — | — | — | — | — | -0-- ---- | -0-- ---- |
| 8Ch | PIE1 | — | CMIE | — | — | — | — | — | — | -0-- ---- | -0-- ---- |

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

## 9.6 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 will have to be implemented in software.

Example 9-3 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 9-3:

- 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 9-3: SAVING THE STATUS AND W REGISTERS IN RAM**

```
MOVWF   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
MOVWF   STATUS_TEMP     ;save status to bank 0
                        ;register
:
:       (ISR)
:
SWAPF   STATUS_TEMP,    ;swap STATUS_TEMP register
        W               ;into W, sets bank to origi-
                        nal
                        ;state
MOVWF   STATUS          ;move W into STATUS register
SWAPF   W_TEMP,F        ;swap W_TEMP
SWAPF   W_TEMP,W        ;swap W_TEMP into W
```



Downloaded from Arrow.com.

## 9.7 Watchdog Timer (WDT)

The Watchdog Timer is a free running on-chip RC oscillator which does not require any external components. This RC oscillator is separate from the RC oscillator of the 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).

### 9.7.1 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.

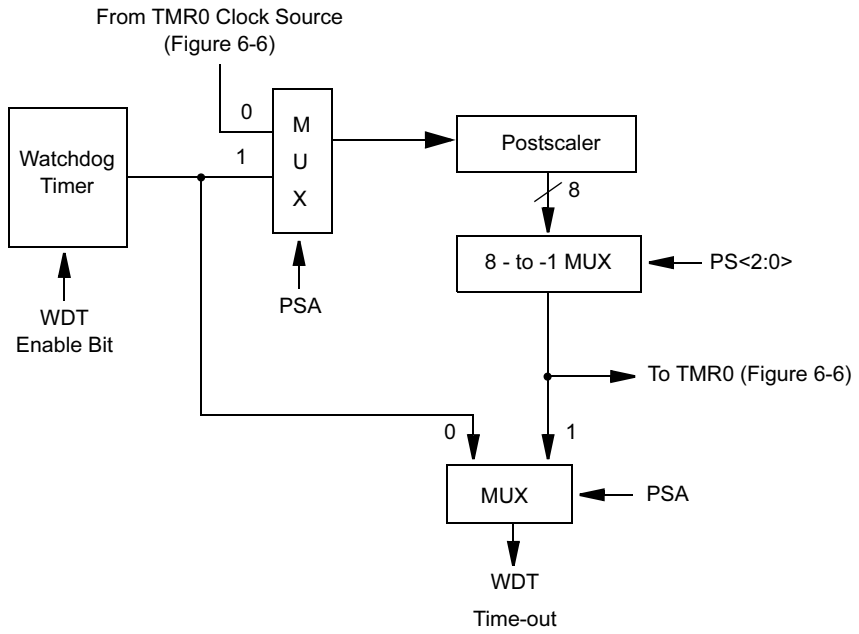
The CLRWDT and SLEEP instructions clear the WDT and the postscaler, if assigned to the WDT, and prevent it from timing out and generating a device RESET.

The $\overline{TO}$ bit in the STATUS register will be cleared upon a Watchdog Timer time-out.

### 9.7.2 WDT PROGRAMMING CONSIDERATIONS

It should also be taken in account that under worst case conditions (VDD = Min., Temperature = Max., max. WDT prescaler) it may take several seconds before a WDT time-out occurs.

**FIGURE 9-17: WATCHDOG TIMER BLOCK DIAGRAM**




**Note:** T0SE, T0CS, PSA, PS<2:0> are bits in the OPTION register.

**TABLE 9-7: SUMMARY OF WATCHDOG TIMER REGISTERS**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR Reset | Value on all other RESETS |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 2007h | Config. bits | — | BODEN | CP1 | CP0 | $\overline{PWRTE}$ | WDTE | FOSC1 | FOSC0 | — | — |
| 81h | OPTION | RBPU | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |

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

**Note:** – = Unimplemented location, read as "0"
+ = Reserved for future use


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 9.8 Power-Down Mode (SLEEP)

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

If enabled, the Watchdog Timer will be cleared but keeps running, the $\overline{PD}$ bit in the STATUS register is cleared, the $\overline{TO}$ bit is set, and the oscillator driver is turned off. The 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 VREF 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 PORTB should be considered.

The $\overline{MCLR}$ 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 $\overline{MCLR}$ pin low. |
|---|---|

### 9.8.1 WAKE-UP FROM SLEEP

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

1. External RESET input on $\overline{MCLR}$ pin
2. Watchdog Timer Wake-up (if WDT was enabled)
3. Interrupt from RB0/INT pin, RB Port change, or the Peripheral Interrupt (Comparator).

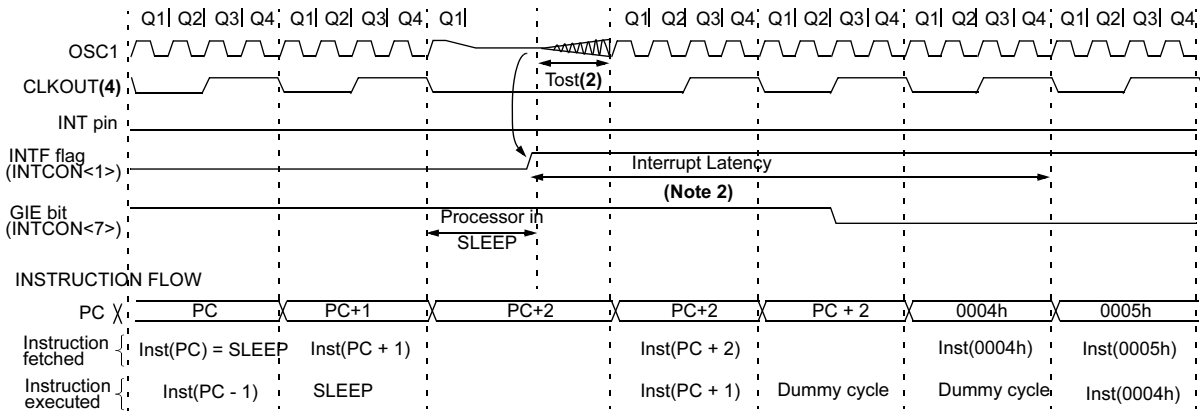
The first event will cause a device RESET. The two latter events are considered a continuation of program execution. The $\overline{TO}$ and $\overline{PD}$ bits in the STATUS register can be used to determine the cause of device RESET. $\overline{PD}$ bit, which is set on power-up, is cleared when SLEEP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

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

| **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 9-18: WAKE-UP FROM SLEEP THROUGH INTERRUPT**




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



Downloaded from Arrow.com.

## 9.9 Code Protection

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

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

## 9.10 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.

## 9.11 In-Circuit Serial Programming™

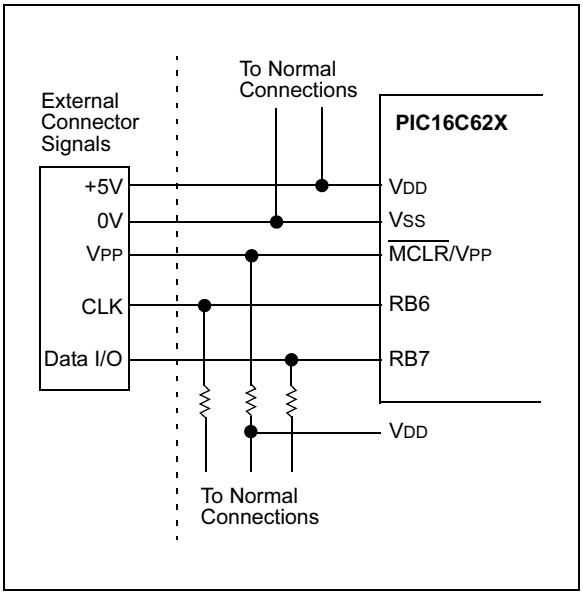
The PIC16C62X microcontrollers can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data and three other lines for power, ground and the programming voltage. This allows customers to manufacture boards with unprogrammed devices and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

The device is placed into a Program/Verify mode by holding the RB6 and RB7 pins low, while raising the $\overline{MCLR}$ ($V_{PP}$) pin from $V_{IL}$ to $V_{IHH}$ (see programming specification). RB6 becomes the programming clock and RB7 becomes the programming data. Both RB6 and RB7 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 if the command was a load or a read. For complete details of serial programming, please refer to the PIC16C6X/7X/9XX Programming Specification (DS30228).

A typical In-Circuit Serial Programming connection is shown in Figure 9-19.

**FIGURE 9-19: TYPICAL IN-CIRCUIT SERIAL PROGRAMMING CONNECTION**





© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 10.0 INSTRUCTION SET SUMMARY

Each PIC16C62X 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 PIC16C62X instruction set summary in Table 10-2 lists **byte-oriented**, **bit-oriented,** and **literal and control** operations. Table 10-1 shows the opcode field descriptions.

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

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

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

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

**TABLE 10-1: OPCODE FIELD DESCRIPTIONS**

| Field | Description |
|---|---|
| f | Register file address (0x00 to 0x7F) |
| W | Working register (accumulator) |
| b | Bit address within an 8-bit file register |
| k | Literal field, constant data or label |
| x | Don't care location (= 0 or 1)<br>The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools. |
| d | Destination select; d = 0: store result in W,<br>d = 1: store result in file register f.<br>Default is d = 1 |
| label | Label name |
| TOS | Top of Stack |
| PC | Program Counter |
| PCLATH | Program Counter High Latch |
| GIE | Global Interrupt Enable bit |
| WDT | Watchdog Timer/Counter |
| TO | Time-out bit |
| PD | Power-down bit |
| dest | Destination either the W register or the specified register file location |
| [ ] | Options |
| ( ) | Contents |
| → | Assigned to |
| < > | Register bit field |
| ∈ | In the set of |
| italics | User defined term (font is courier) |

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

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

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

Table 10-1 lists the instructions recognized by the MPASM™ assembler.

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

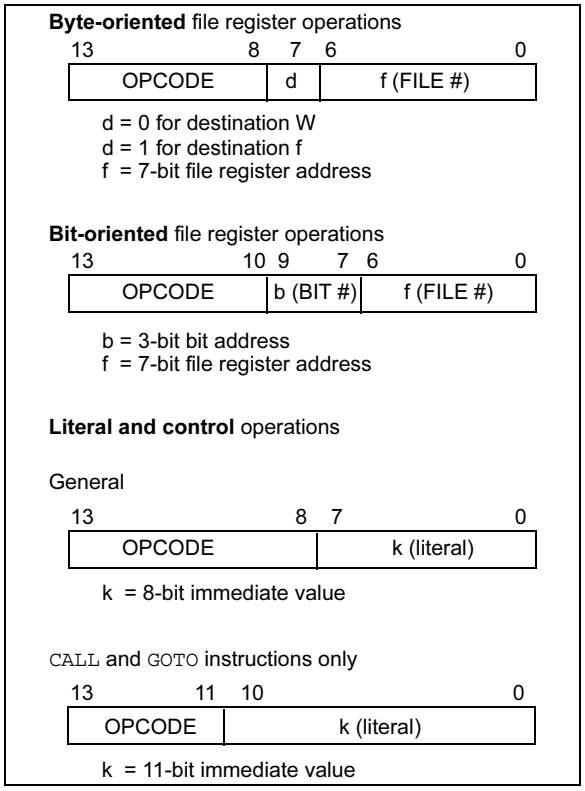
**Note:** To maintain upward compatibility with future PICmicro® products, do not use the OPTION and TRIS instructions.

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

0xhh

where h signifies a hexadecimal digit.

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






Downloaded from Arrow.com.

**TABLE 10-2: PIC16C62X INSTRUCTION SET**

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

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

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

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


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 10.1 Instruction Descriptions

**ADDLW** — **Add Literal and W**

Syntax: [ *label* ] ADDLW k

Operands: $0 \le k \le 255$

Operation: (W) + k → (W)

Status Affected: C, DC, Z

Encoding:

| | | | |
|---|---|---|---|
| 11 | 111x | kkkk | kkkk |

Description: The contents of the W register are added to the eight bit literal 'k' and the result is placed in the W register.

Words: 1

Cycles: 1

Example
```
ADDLW   0x15
```
Before Instruction
W = 0x10
After Instruction
W = 0x25

**ADDWF** — **Add W and f**

Syntax: [ *label* ] ADDWF f,d

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

Operation: (W) + (f) → (dest)

Status Affected: C, DC, Z

Encoding:

| | | | |
|---|---|---|---|
| 00 | 0111 | dfff | ffff |

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

Words: 1

Cycles: 1

Example
```
ADDWF   FSR, 0
```
Before Instruction
W = 0x17
FSR = 0xC2
After Instruction
W = 0xD9
FSR = 0xC2

**ANDLW** — **AND Literal with W**

Syntax: [ *label* ] ANDLW k

Operands: $0 \le k \le 255$

Operation: (W) .AND. (k) → (W)

Status Affected: Z

Encoding:

| | | | |
|---|---|---|---|
| 11 | 1001 | kkkk | kkkk |

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

Words: 1

Cycles: 1

Example
```
ANDLW   0x5F
```
Before Instruction
W = 0xA3
After Instruction
W = 0x03

**ANDWF** — **AND W with f**

Syntax: [ *label* ] ANDWF f,d

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

Operation: (W) .AND. (f) → (dest)

Status Affected: Z

Encoding:

| | | | |
|---|---|---|---|
| 00 | 0101 | dfff | ffff |

Description: AND the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

Words: 1

Cycles: 1

Example
```
ANDWF   FSR, 1
```
Before Instruction
W = 0x17
FSR = 0xC2
After Instruction
W = 0x17
FSR = 0x02



Downloaded from Arrow.com.

**BCF** **Bit Clear f**

Syntax: [ *label* ] BCF f,b

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

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

Status Affected: None

Encoding:

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

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

Words: 1

Cycles: 1

Example

```
BCF     FLAG_REG, 7
```

Before Instruction
FLAG_REG = 0xC7

After Instruction
FLAG_REG = 0x47

**BSF** **Bit Set f**

Syntax: [ *label* ] BSF f,b

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

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

Status Affected: None

Encoding:

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

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

Words: 1

Cycles: 1

Example

```
BSF     FLAG_REG,   7
```

Before Instruction
FLAG_REG = 0x0A

After Instruction
FLAG_REG = 0x8A

**BTFSC** **Bit Test, Skip if Clear**

Syntax: [ *label* ] BTFSC f,b

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

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

Status Affected: None

Encoding:

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

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

Words: 1

Cycles: 1(2)

Example

```
HERE    BTFSC   FLAG,1
FALSE   GOTO    PROCESS_CO
TRUE    •       DE
        •
        •
```

Before Instruction
PC = address HERE

After Instruction
if FLAG<1> = 0,
PC = address TRUE
if FLAG<1>=1,
PC = address FALSE


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

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

Syntax: [ *label* ] BTFSS f,b

Operands: 0 ≤ f ≤ 127
0 ≤ b < 7

Operation: skip if (f<b>) = 1

Status Affected: None

Encoding:

| | | | |
|---|---|---|---|
| 01 | 11bb | bfff | ffff |

Description: If bit 'b' in register 'f' is '1', then the next instruction is skipped.
If bit 'b' is '1', then the next instruction fetched during the current instruction execution, is discarded and a NOP is executed instead, making this a two-cycle instruction.

Words: 1

Cycles: 1(2)

Example

```
HERE    BTFSS   FLAG,1
FALSE   GOTO    PROCESS_CO
TRUE    •       DE
        •
        •
```

Before Instruction
PC = address HERE
After Instruction
if FLAG<1> = 0,
PC = address FALSE
if FLAG<1> = 1,
PC = address TRUE

**CALL** **Call Subroutine**

Syntax: [ *label* ] CALL k

Operands: 0 ≤ k ≤ 2047

Operation: (PC)+ 1→ TOS,
k → PC<10:0>,
(PCLATH<4:3>) → PC<12:11>

Status Affected: None

Encoding:

| | | | |
|---|---|---|---|
| 10 | 0kkk | kkkk | kkkk |

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.

Words: 1

Cycles: 2

Example

```
HERE    CALL
                THER
                E
```

Before Instruction
PC = Address HERE
After Instruction
PC = Address THERE
TOS = Address HERE+1

**CLRF** **Clear f**

Syntax: [ *label* ] CLRF f

Operands: 0 ≤ f ≤ 127

Operation: 00h → (f)
1 → Z

Status Affected: Z

Encoding:

| | | | |
|---|---|---|---|
| 00 | 0001 | 1fff | ffff |

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

Words: 1

Cycles: 1

Example

```
CLRF    FLAG_REG
```

Before Instruction
FLAG_REG = 0x5A
After Instruction
FLAG_REG = 0x00
Z = 1



Downloaded from Arrow.com.

**CLRW** **Clear W**

Syntax: [ *label* ] CLRW

Operands: None

Operation: 00h → (W)
1 → Z

Status Affected: Z

Encoding:

| | | | |
|---|---|---|---|
| 00 | 0001 | 0000 | 0011 |

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

Words: 1

Cycles: 1

Example
```
CLRW
```
Before Instruction
W = 0x5A
After Instruction
W = 0x00
Z = 1

**CLRWDT** **Clear Watchdog Timer**

Syntax: [ *label* ] CLRWDT

Operands: None

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

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

Encoding:

| | | | |
|---|---|---|---|
| 00 | 0000 | 0110 | 0100 |

Description: CLRWDT instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. STATUS bits $\overline{TO}$ and $\overline{PD}$ are set.

Words: 1

Cycles: 1

Example
```
CLRWDT
```
Before Instruction
WDT counter = ?
After Instruction
WDT counter = 0x00
WDT prescaler = 0
$\overline{TO}$ = 1
$\overline{PD}$ = 1

**COMF** **Complement f**

Syntax: [ *label* ] COMF f,d

Operands: 0 ≤ f ≤ 127
d ∈ [0,1]

Operation: ($\bar{f}$) → (dest)

Status Affected: Z

Encoding:

| | | | |
|---|---|---|---|
| 00 | 1001 | dfff | ffff |

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

Words: 1

Cycles: 1

Example
```
COMF    REG1,0
```
Before Instruction
REG1 = 0x13
After Instruction
REG1 = 0x13
W = 0xEC

**DECF** **Decrement f**

Syntax: [ *label* ] DECF f,d

Operands: 0 ≤ f ≤ 127
d ∈ [0,1]

Operation: (f) - 1 → (dest)

Status Affected: Z

Encoding:

| | | | |
|---|---|---|---|
| 00 | 0011 | dfff | ffff |

Description: Decrement register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

Words: 1

Cycles: 1

Example
```
DECF    CNT, 1
```
Before Instruction
CNT = 0x01
Z = 0
After Instruction
CNT = 0x00
Z = 1


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

**DECFSZ** **Decrement f, Skip if 0**

Syntax: [ *label* ] DECFSZ f,d

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

Operation: (f) - 1 → (dest); skip if result = 0

Status Affected: None

Encoding:

| 00 | 1011 | dfff | ffff |
|---|---|---|---|

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

Words: 1

Cycles: 1(2)

Example

```
HERE     DECFSZ   CNT, 1
         GOTO     LOOP
CONTINUE •
         •
         •
```

Before Instruction
PC = address HERE
After Instruction
CNT = CNT - 1
if CNT = 0,
PC = address CONTINUE
if CNT ≠ 0,
PC = address HERE+1

**GOTO** **Unconditional Branch**

Syntax: [ *label* ] GOTO k

Operands: $0 \le k \le 2047$

Operation: k → PC<10:0>
PCLATH<4:3> → PC<12:11>

Status Affected: None

Encoding:

| 10 | 1kkk | kkkk | kkkk |
|---|---|---|---|

Description: GOTO is an unconditional branch. The eleven bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. GOTO is a two-cycle instruction.

Words: 1

Cycles: 2

Example

```
GOTO THERE
```

After Instruction
PC = Address THERE

**INCF** **Increment f**

Syntax: [ *label* ] INCF f,d

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

Operation: (f) + 1 → (dest)

Status Affected: Z

Encoding:

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

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

Words: 1

Cycles: 1

Example

```
INCF    CNT, 1
```

Before Instruction
CNT = 0xFF
Z = 0
After Instruction
CNT = 0x00
Z = 1



Downloaded from Arrow.com.

**INCFSZ** **Increment f, Skip if 0**

Syntax: [ *label* ] INCFSZ f,d

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

Operation: (f) + 1 → (dest), skip if result = 0

Status Affected: None

Encoding:

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

Description: The contents of register 'f' are incremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'.
If the result is 0, the next instruction, which is already fetched, is discarded. A NOP is executed instead making it a two-cycle instruction.

Words: 1

Cycles: 1(2)

Example

```
HERE     INCFSZ     CNT,  1
         GOTO       LOOP
CONTINUE •
         •
         •
```

Before Instruction
PC = address HERE
After Instruction
CNT = CNT + 1
if CNT= 0,
PC = address CONTINUE
if CNT≠ 0,
PC = address HERE +1

**IORLW** **Inclusive OR Literal with W**

Syntax: [ *label* ] IORLW k

Operands: $0 \le k \le 255$

Operation: (W) .OR. k → (W)

Status Affected: Z

Encoding:

| 11 | 1000 | kkkk | kkkk |
|---|---|---|---|

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

Words: 1

Cycles: 1

Example

```
IORLW   0x35
```

Before Instruction
W = 0x9A
After Instruction
W = 0xBF
Z = 1

**IORWF** **Inclusive OR W with f**

Syntax: [ *label* ] IORWF f,d

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

Operation: (W) .OR. (f) → (dest)

Status Affected: $\overline{Z}$

Encoding:

| 00 | 0100 | dfff | ffff |
|---|---|---|---|

Description: Inclusive OR the W register with register 'f'. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.

Words: 1

Cycles: 1

Example

```
IORWF          RESULT, 0
```

Before Instruction
RESULT = 0x13
W = 0x91
After Instruction
RESULT = 0x13
W = 0x93
Z = 1

**MOVLW** **Move Literal to W**

Syntax: [ *label* ] MOVLW k

Operands: $0 \le k \le 255$

Operation: k → (W)

Status Affected: None

Encoding:

| 11 | 00xx | kkkk | kkkk |
|---|---|---|---|

Description: The eight bit literal 'k' is loaded into W register. The don't cares will assemble as 0's.

Words: 1

Cycles: 1

Example

```
MOVLW   0x5A
```

After Instruction
W = 0x5A


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

**MOVF** **Move f**

Syntax: [ *label* ] MOVF f,d

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

Operation: (f) → (dest)

Status Affected: Z

Encoding:

| | | | |
|---|---|---|---|
| 00 | 1000 | dfff | ffff |

Description: The contents of register f is moved to a destination dependent upon the status of d. If d = 0, destination is W register. If d = 1, the destination is file register f itself. d = 1 is useful to test a file register since status flag Z is affected.

Words: 1

Cycles: 1

Example

```
MOVF    FSR, 0
```

After Instruction
W = value in FSR register
Z = 1

**MOVWF** **Move W to f**

Syntax: [ *label* ] MOVWF f

Operands: $0 \le f \le 127$

Operation: (W) → (f)

Status Affected: None

Encoding:

| | | | |
|---|---|---|---|
| 00 | 0000 | 1fff | ffff |

Description: Move data from W register to register 'f'.

Words: 1

Cycles: 1

Example

```
MOVWF   OPTION
```

Before Instruction
OPTION = 0xFF
W = 0x4F
After Instruction
OPTION = 0x4F
W = 0x4F

**NOP** **No Operation**

Syntax: [ *label* ] NOP

Operands: None

Operation: No operation

Status Affected: None

Encoding:

| | | | |
|---|---|---|---|
| 00 | 0000 | 0xx0 | 0000 |

Description: No operation.

Words: 1

Cycles: 1

Example

```
NOP
```

**OPTION** **Load Option Register**

Syntax: [ *label* ] OPTION

Operands: None

Operation: (W) → OPTION

Status Affected: None

Encoding:

| | | | |
|---|---|---|---|
| 00 | 0000 | 0110 | 0010 |

Description: The contents of the W register are loaded in the OPTION register. This instruction is supported for code compatibility with PIC16C5X products. Since OPTION is a readable/writable register, the user can directly address it.

Words: 1

Cycles: 1

Example

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



Downloaded from Arrow.com.

**RETFIE** **Return from Interrupt**

Syntax: [ *label* ] RETFIE

Operands: None

Operation: TOS → PC,
1 → GIE

Status Affected: None

Encoding:

| | | | |
|---|---|---|---|
| 00 | 0000 | 0000 | 1001 |

Description: Return from Interrupt. Stack is POPed and Top of Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a two-cycle instruction.

Words: 1

Cycles: 2

Example

```
RETFIE
```

After Interrupt
PC = TOS
GIE = 1

**RETLW** **Return with Literal in W**

Syntax: [ *label* ] RETLW k

Operands: 0 ≤ k ≤ 255

Operation: k → (W);
TOS → PC

Status Affected: None

Encoding:

| | | | |
|---|---|---|---|
| 11 | 01xx | kkkk | kkkk |

Description: The W register is loaded with the eight bit literal 'k'. The program counter is loaded from the top of the stack (the return address). This is a two-cycle instruction.

Words: 1

Cycles: 2

Example

```
CALL TABLE;W contains table
        ;offset value
TABLE   •       ;W now has table value
        •
        •
ADDWF PC    ;W = offset
RETLW k1    ;Begin table
RETLW k2    ;
        •
        •
        •
RETLW kn    ; End of table
```

Before Instruction
W = 0x07
After Instruction
W = value of k8

**RETURN** **Return from Subroutine**

Syntax: [ *label* ] RETURN

Operands: None

Operation: TOS → PC

Status Affected: None

Encoding:

| | | | |
|---|---|---|---|
| 00 | 0000 | 0000 | 1000 |

Description: Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two-cycle instruction.

Words: 1

Cycles: 2

Example

```
RETURN
```

After Interrupt
PC = TOS


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

**RLF** **Rotate Left f through Carry**

Syntax: [ *label* ] RLF f,d

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

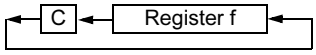
Operation: See description below

Status Affected: C

Encoding:

| 00 | 1101 | dfff | ffff |
|---|---|---|---|

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




Words: 1

Cycles: 1

Example

```
RLF     REG1,0
```

Before Instruction

```
REG1    =   1110 0110
C       =   0
```

After Instruction

```
REG1    =   1110 0110
W       =   1100 1100
C       =   1
```

**RRF** **Rotate Right f through Carry**

Syntax: [ *label* ] RRF f,d

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

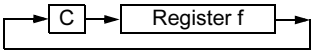
Operation: See description below

Status Affected: C

Encoding:

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

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




Words: 1

Cycles: 1

Example

```
RRF     REG1,
        0
```

Before Instruction

```
REG1    =   1110 0110
C       =   0
```

After Instruction

```
REG1    =   1110 0110
W       =   0111 0011
C       =   0
```

**SLEEP**

Syntax: [ *label* ] SLEEP

Operands: None

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

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

Encoding:

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

Description: The power-down STATUS bit, $\overline{PD}$ is cleared. Time-out STATUS bit, $\overline{TO}$ is set. Watch-dog Timer and its prescaler are cleared.
The processor is put into SLEEP mode with the oscillator stopped. See Section 9.8 for more details.

Words: 1

Cycles: 1

Example:

```
SLEEP
```



Downloaded from Arrow.com.

**SUBLW** **Subtract W from Literal**

Syntax: [ *label* ] SUBLW k

Operands: $0 \le k \le 255$

Operation: k - (W) → (W)

Status Affected: C, DC, Z

Encoding:

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

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

Words: 1

Cycles: 1

Example 1:
```
SUBLW   0x02
```
Before Instruction
W = 1
C = ?
After Instruction
W = 1
C = 1; result is positive

Example 2: Before Instruction
W = 2
C = ?
After Instruction
W = 0
C = 1; result is zero

Example 3: Before Instruction
W = 3
C = ?
After Instruction
W = 0xFF
C = 0; result is negative

**SUBWF** **Subtract W from f**

Syntax: [ *label* ] SUBWF f,d

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

Operation: (f) - (W) → (dest)

Status Affected: C, DC, Z

Encoding:

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

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

Words: 1

Cycles: 1

Example 1:
```
SUBWF   REG1,1
```
Before Instruction
REG1= 3
W = 2
C = ?
After Instruction
REG1= 1
W = 2
C = 1; result is positive

Example 2: Before Instruction
REG1= 2
W = 2
C = ?
After Instruction
REG1= 0
W = 2
C = 1; result is zero

Example 3: Before Instruction
REG1= 1
W = 2
C = ?
After Instruction
REG1= 0xFF
W = 2
C = 0; result is negative


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

**SWAPF** **Swap Nibbles in f**

Syntax: [ *label* ] SWAPF f,d

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

Operation: (f<3:0>) → (dest<7:4>),
(f<7:4>) → (dest<3:0>)

Status Affected: None

Encoding:

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

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

Words: 1

Cycles: 1

Example
```
SWAPF REG, 0
```
Before Instruction
REG1 = 0xA5
After Instruction
REG1 = 0xA5
W = 0x5A

**TRIS** **Load TRIS Register**

Syntax: [ *label* ] TRIS f

Operands: $5 \le f \le 7$

Operation: (W) → TRIS register f;

Status Affected: None

Encoding:

| 00 | 0000 | 0110 | 0fff |
|---|---|---|---|

Description: The instruction is supported for code compatibility with the PIC16C5X products. Since TRIS registers are readable and writable, the user can directly address them.

Words: 1

Cycles: 1

Example

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

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

Syntax: [ *label* ] XORLW k

Operands: $0 \le k \le 255$

Operation: (W) .XOR. k → (W)

Status Affected: Z

Encoding:

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

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

Words: 1

Cycles: 1

Example
```
XORLW 0xAF
```
Before Instruction
W = 0xB5
After Instruction
W = 0x1A

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

Syntax: [ *label* ] XORWF f,d

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

Operation: (W) .XOR. (f) → (dest)

Status Affected: Z

Encoding:

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

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

Words: 1

Cycles: 1

Example
```
XORWF REG 1
```
Before Instruction
REG = 0xAF
W = 0xB5
After Instruction
REG = 0x1A
W = 0xB5



Downloaded from Arrow.com.

**NOTES:**


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 11.0 DEVELOPMENT SUPPORT

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

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

### 11.1 MPLAB Integrated Development Environment Software

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

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

The MPLAB IDE allows you to:

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

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

### 11.2 MPASM Assembler

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

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

The MPASM assembler features include:

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



Downloaded from Arrow.com.

## 11.3 MPLAB C17 and MPLAB C18 C Compilers

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

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

## 11.4 MPLINK Object Linker/ MPLIB Object Librarian

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

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

The object linker/library features include:

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

## 11.5 MPLAB C30 C Compiler

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

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

## 11.6 MPLAB ASM30 Assembler, Linker, and Librarian

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

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

## 11.7 MPLAB SIM Software Simulator

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

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

## 11.8 MPLAB SIM30 Software Simulator

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

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


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

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

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

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

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

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

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

The MPLAB ICD 4000 is a premium emulator system, providing the features of MPLAB ICE 2000, but with increased emulation memory and high speed performance for dsPIC30F and PIC18XXXX devices. Its advanced emulator features include complex triggering and timing, up to 2 Mb of emulation memory, and the ability to view variables in real-time.

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

## 11.11 MPLAB ICD 2 In-Circuit Debugger

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

## 11.12 PRO MATE II Universal Device Programmer

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

## 11.13 PICSTART Plus Development Programmer

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



Downloaded from Arrow.com.

## 11.14 PICDEM 1 PICmicro Demonstration Board

The PICDEM 1 demonstration board demonstrates the capabilities of the PIC16C5X (PIC16C54 to PIC16C58A), PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 and PIC17C44. All necessary hardware and software is included to run basic demo programs. The sample microcontrollers provided with the PICDEM 1 demonstration board can be programmed with a PRO MATE II device programmer, or a PICSTART Plus development programmer. The PICDEM 1 demonstration board can be connected to the MPLAB ICE in-circuit emulator for testing. A prototype area extends the circuitry for additional application components. Features include an RS-232 interface, a potentiometer for simulated analog input, push button switches and eight LEDs.

## 11.15 PICDEM.net Internet/Ethernet Demonstration Board

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

## 11.16 PICDEM 2 Plus Demonstration Board

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

## 11.17 PICDEM 3 PIC16C92X Demonstration Board

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

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

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

## 11.19 PICDEM 17 Demonstration Board

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


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 11.20 PICDEM 18R PIC18C601/801 Demonstration Board

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

## 11.21 PICDEM LIN PIC16C43X Demonstration Board

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

## 11.22 PICkit™ 1 FLASH Starter Kit

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

## 11.23 PICDEM USB PIC16C7X5 Demonstration Board

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

## 11.24 Evaluation and Programming Tools

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

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

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



Downloaded from Arrow.com.

**NOTES:**


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 12.0 ELECTRICAL SPECIFICATIONS

**Absolute Maximum Ratings †**

Ambient Temperature under bias ..... -40° to +125°C

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

Voltage on any pin with respect to $V_{SS}$ (except $V_{DD}$ and $\overline{MCLR}$) .....-0.6V to $V_{DD}$ +0.6V

Voltage on $V_{DD}$ with respect to $V_{SS}$ ..... 0 to +7.5V

Voltage on $\overline{MCLR}$ with respect to $V_{SS}$ **(Note 2)** .....0 to +14V

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

Total power Dissipation **(Note 1)**.....1.0W

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

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

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

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

Maximum Output Current sunk by any I/O pin .....25 mA

Maximum Output Current sourced by any I/O pin.....25 mA

Maximum Current sunk by PORTA and PORTB.....200 mA

Maximum Current sourced by PORTA and PORTB.....200 mA

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

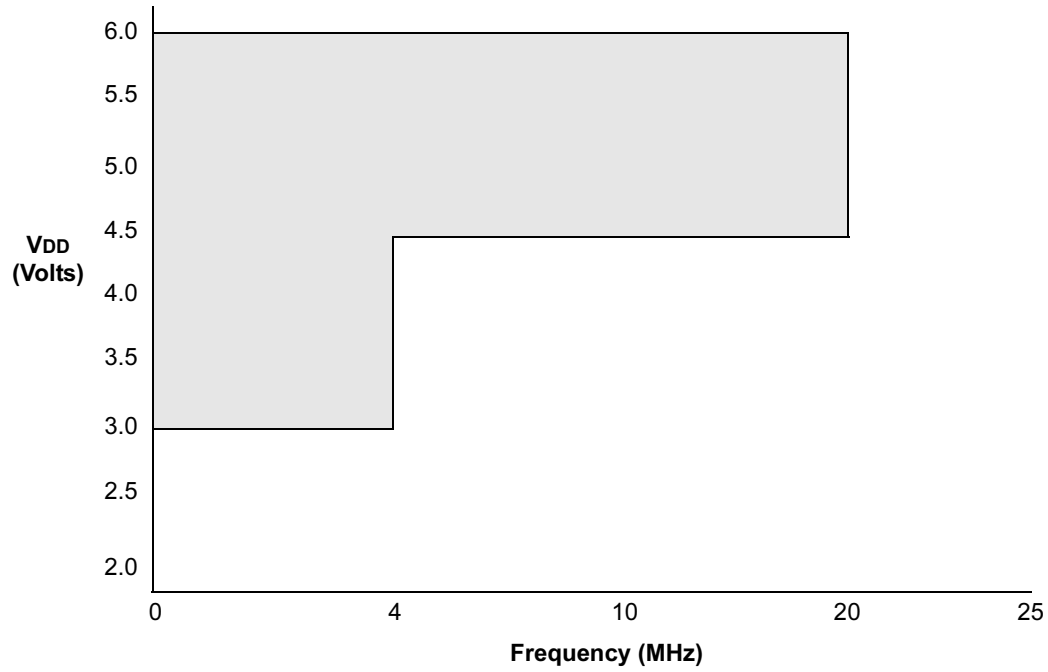
**2:** Voltage spikes below $V_{SS}$ at the $\overline{MCLR}$ pin, inducing currents greater than 80 mA, may cause latchup. Thus, a series resistor of 50-100Ω should be used when applying a "low" level to the $\overline{MCLR}$ pin rather than pulling this pin directly to $V_{SS}$.

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



Downloaded from Arrow.com.

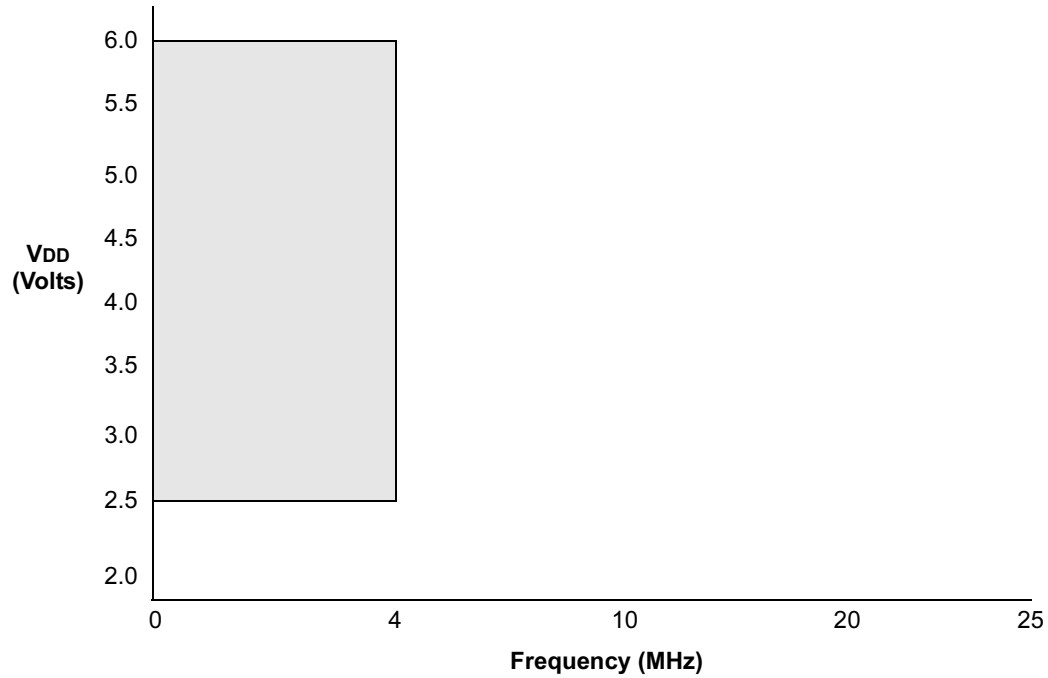
**FIGURE 12-1: PIC16C62X VOLTAGE-FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C**




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

**2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.

**FIGURE 12-2: PIC16LC62X VOLTAGE-FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C**



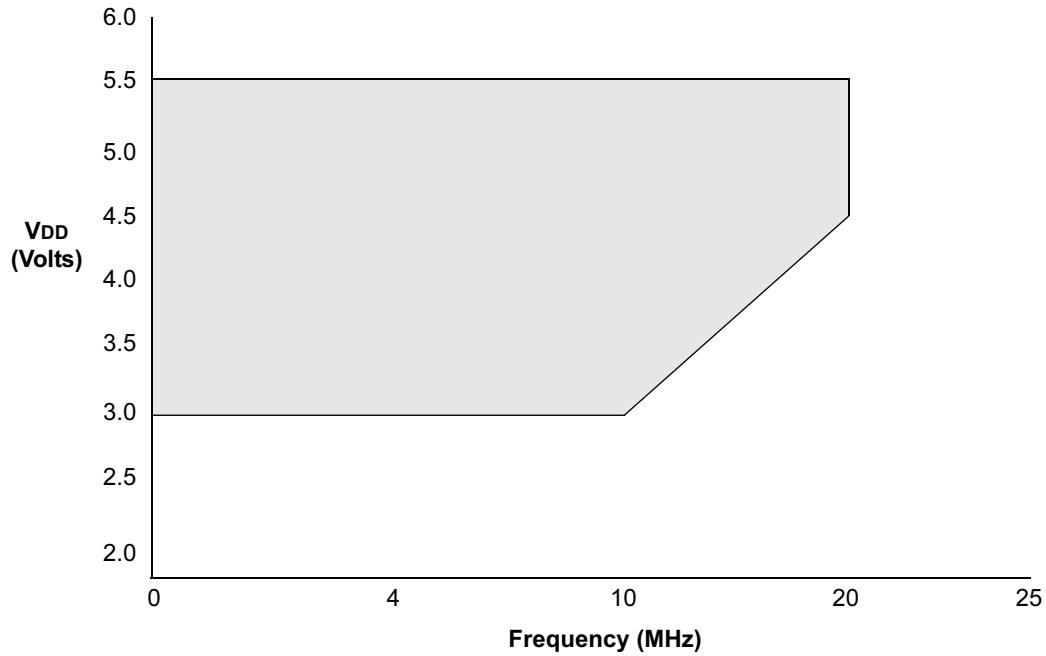

**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.

**2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

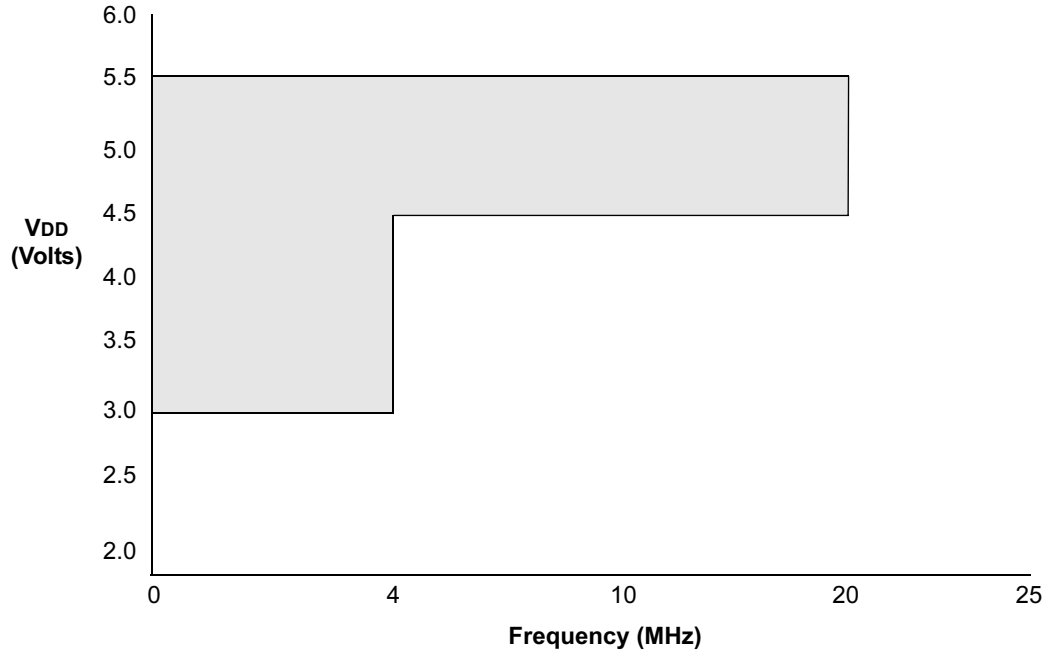
**FIGURE 12-3: PIC16C62XA VOLTAGE-FREQUENCY GRAPH, 0°C ≤ TA ≤ +70°C**




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

**2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.

**FIGURE 12-4: PIC16C62XA VOLTAGE-FREQUENCY GRAPH, -40°C ≤ TA ≤ 0°C, +70°C ≤ TA ≤ +125°C**



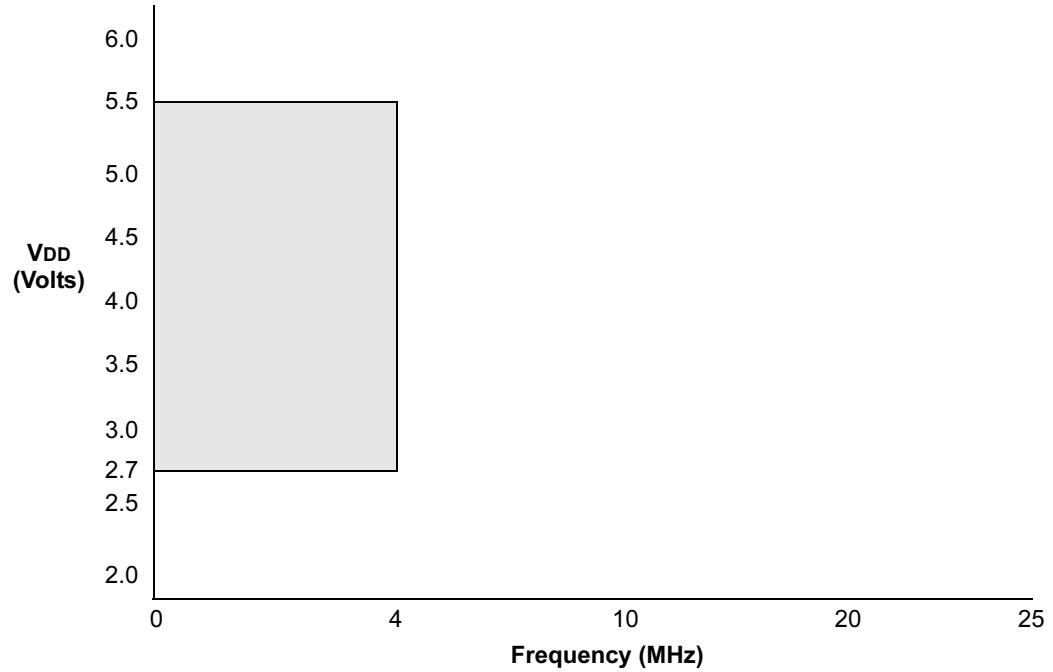

**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.

**2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.



Downloaded from Arrow.com.

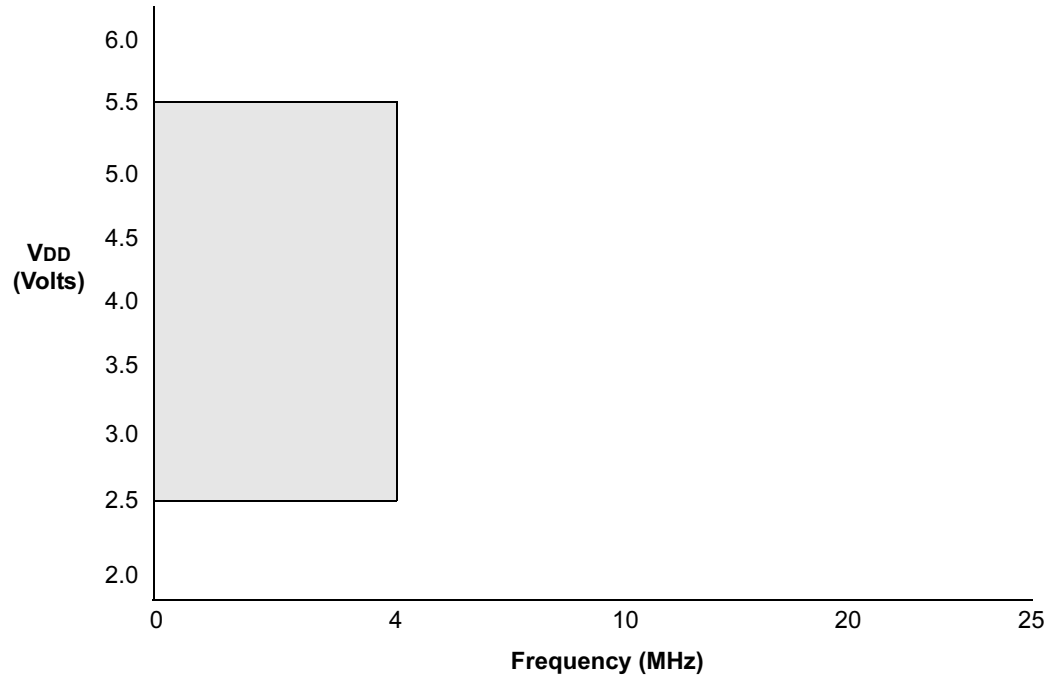
**FIGURE 12-5: PIC16LC620A/LC621A/LC622A VOLTAGE-FREQUENCY GRAPH, -40°C ≤ TA ≤ 0°C**




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

**2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.

**FIGURE 12-6: PIC16LC620A/LC621A/LC622A VOLTAGE-FREQUENCY GRAPH, 0°C ≤ TA ≤ +125°C**



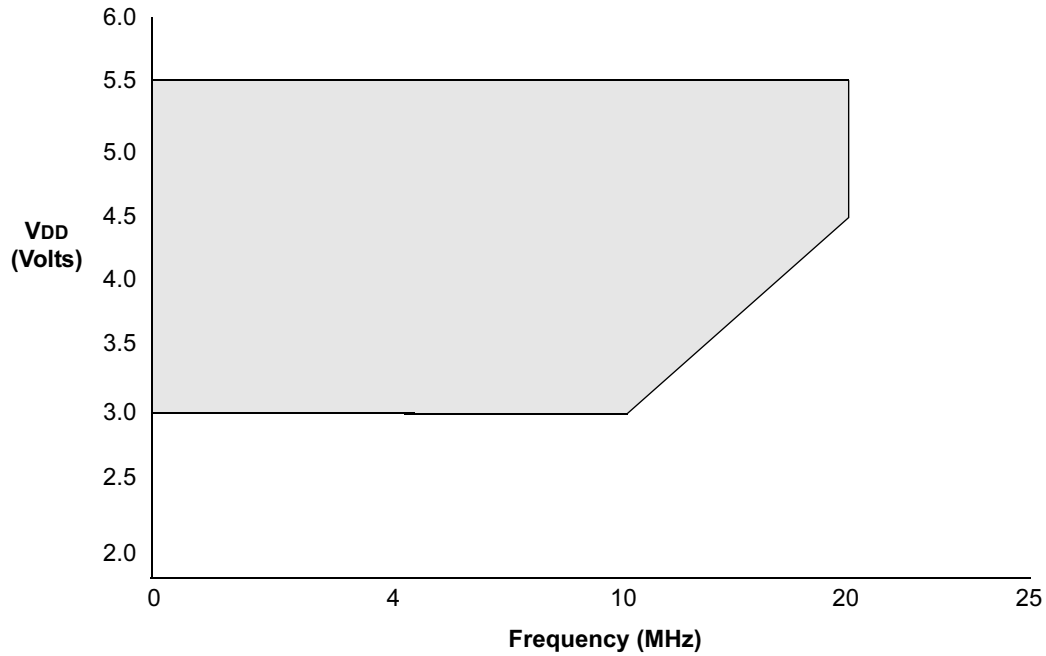

**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.

**2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

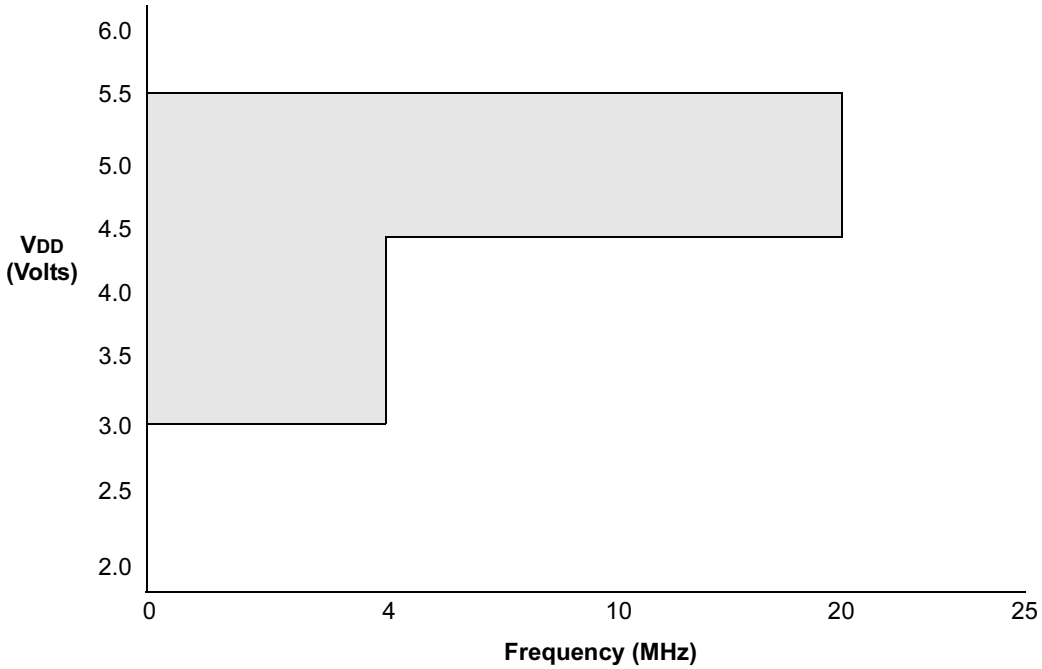
**FIGURE 12-7: PIC16CR62XA VOLTAGE-FREQUENCY GRAPH, 0°C ≤ TA ≤ +70°C**




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

**2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.

**FIGURE 12-8: PIC16CR62XA VOLTAGE-FREQUENCY GRAPH, -40°C ≤ TA ≤ 0°C, +70°C ≤ TA ≤ +125°C**



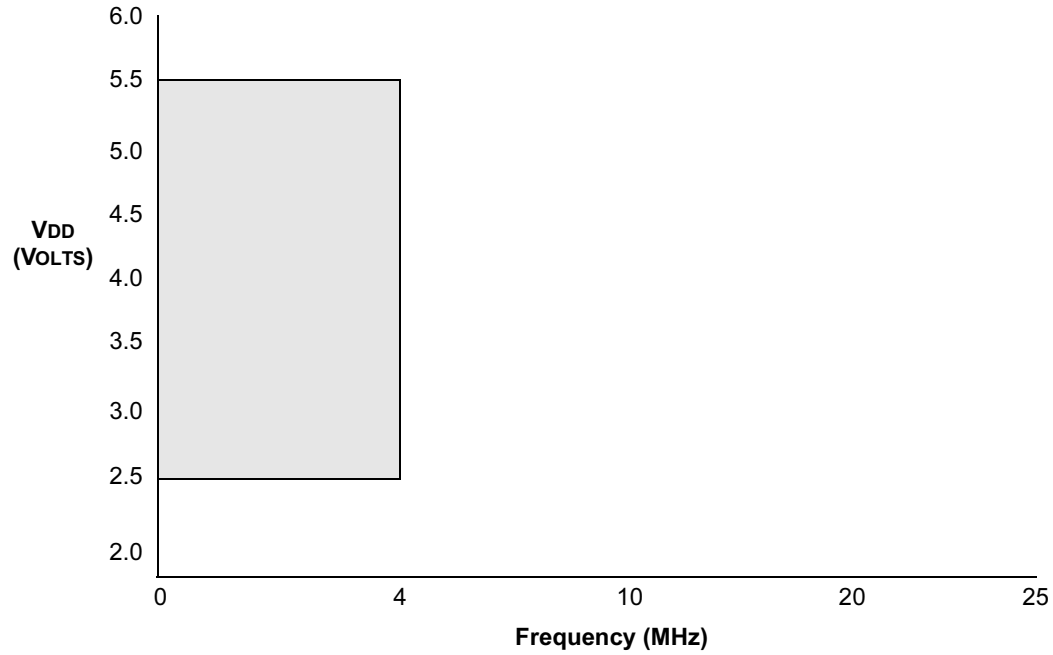

**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.

**2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.



Downloaded from Arrow.com.

**FIGURE 12-9: PIC16LCR62XA VOLTAGE-FREQUENCY GRAPH, -40°C ≤ TA ≤ +125°C**



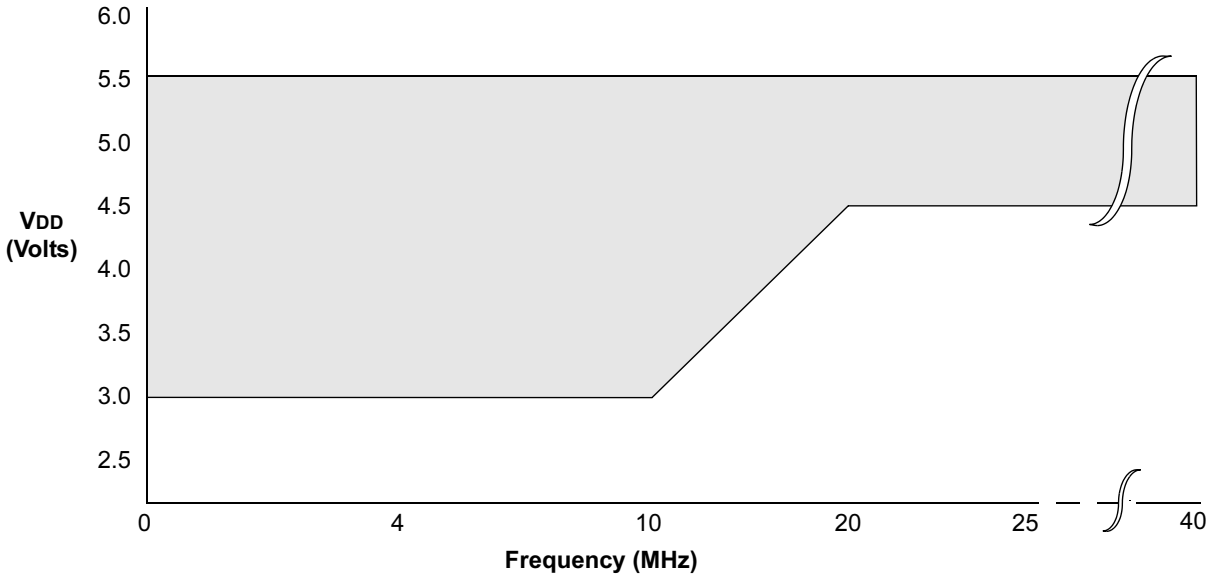

**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.

**2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

**FIGURE 12-10: PIC16C620A/C621A/C622A/CR620A - 40 VOLTAGE-FREQUENCY GRAPH, 0°C ≤ TA ≤ +70°C**




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

**2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.

**3:** Operation between 20 to 40 MHz requires the following:

- VDD between 4.5V. and 5.5V
- OSC1 externally driven
- OSC2 not connected
- HS mode
- Commercial temperatures

Devices qualified for 40 MHz operation have -40 designation (ex: PIC16C620A-40/P).



Downloaded from Arrow.com.

## 12.1 DC Characteristics: PIC16C62X-04 (Commercial, Industrial, Extended) PIC16C62X-20 (Commercial, Industrial, Extended) PIC16LC62X-04 (Commercial, Industrial, Extended)

| | |
|---|---|
| **PIC16C62X** | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended |
| **PIC16LC62X** | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended<br>Operating voltage VDD range is the PIC16C62X range. |

| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| D001 | VDD | Supply Voltage | 3.0 | — | 6.0 | V | See Figures 12-1, 12-2, 12-3, 12-4, and 12-5 |
| D001 | VDD | Supply Voltage | 2.5 | — | 6.0 | V | See Figures 12-1, 12-2, 12-3, 12-4, and 12-5 |
| D002 | VDR | RAM Data Retention Voltage(1) | — | 1.5* | — | V | Device in SLEEP mode |
| D002 | VDR | RAM Data Retention Voltage(1) | — | 1.5* | — | V | Device in SLEEP mode |
| D003 | VPOR | VDD start voltage to ensure Power-on Reset | — | Vss | — | V | See section on Power-on Reset for details |
| D003 | VPOR | VDD start voltage to ensure Power-on Reset | — | VSS | — | V | See section on Power-on Reset for details |
| D004 | SVDD | VDD rise rate to ensure Power-on Reset | 0.05* | — | — | V/ms | See section on Power-on Reset for details |
| D004 | SVDD | VDD rise rate to ensure Power-on Reset | 0.05* | — | — | V/ms | See section on Power-on Reset for details |
| D005 | VBOR | Brown-out Detect Voltage | 3.7 | 4.0 | 4.3 | V | BOREN configuration bit is cleared |
| D005 | VBOR | Brown-out Detect Voltage | 3.7 | 4.0 | 4.3 | V | BOREN configuration bit is cleared |
| D010 | IDD | Supply Current(2) | — | 1.8 | 3.3 | mA | FOSC = 4 MHz, VDD = 5.5V, WDT disabled, XT mode, (**Note 4**)* |
| | | | — | 35 | 70 | µA | FOSC = 32 kHz, VDD = 4.0V, WDT disabled, LP mode |
| | | | — | 9.0 | 20 | mA | FOSC = 20 MHz, VDD = 5.5V, WDT disabled, HS mode |
| D010 | IDD | Supply Current(2) | — | 1.4 | 2.5 | mA | FOSC = 2.0 MHz, VDD = 3.0V, WDT disabled, XT mode, (**Note 4**) |
| | | | — | 26 | 53 | µA | FOSC = 32 kHz, VDD = 3.0V, WDT disabled, LP mode |
| D020 | IPD | Power-down Current(3) | — | 1.0 | 2.5<br>15 | µA<br>µA | VDD=4.0V, WDT disabled<br>(125°C) |
| D020 | IPD | Power-down Current(3) | — | 0.7 | 2 | µA | VDD=3.0V, WDT disabled |

* These parameters are characterized but not tested.

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

**Note 1:** This is the limit to which VDD can be lowered without losing RAM data.

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

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

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

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


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 12.1 DC Characteristics: PIC16C62X-04 (Commercial, Industrial, Extended) PIC16C62X-20 (Commercial, Industrial, Extended) PIC16LC62X-04 (Commercial, Industrial, Extended) (CONT.)

| PIC16C62X | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended |
|---|---|
| **PIC16LC62X** | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended<br>Operating voltage VDD range is the PIC16C62X range. |

| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| D022 | ΔIWDT | WDT Current(5) | — | 6.0 | 20 | µA | VDD=4.0V |
| | | | | | 25 | µA | (125°C) |
| D022A | ΔIBOR | Brown-out Reset Current(5) | — | 350 | 425 | µA | BOD enabled, VDD = 5.0V |
| D023 | ΔICOMP | Comparator Current for each Comparator(5) | — | — | 100 | µA | VDD = 4.0V |
| D023A | ΔIVREF | VREF Current(5) | — | — | 300 | µA | VDD = 4.0V |
| D022 | ΔIWDT | WDT Current(5) | — | 6.0 | 15 | µA | VDD=3.0V |
| D022A | ΔIBOR | Brown-out Reset Current(5) | — | 350 | 425 | µA | BOD enabled, VDD = 5.0V |
| D023 | ΔICOMP | Comparator Current for each Comparator(5) | — | — | 100 | µA | VDD = 3.0V |
| D023A | ΔIVREF | VREF Current(5) | — | — | 300 | µA | VDD = 3.0V |
| 1A | FOSC | LP Oscillator Operating Frequency | 0 | — | 200 | kHz | All temperatures |
| | | RC Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
| | | XT Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
| | | HS Oscillator Operating Frequency | 0 | — | 20 | MHz | All temperatures |
| 1A | FOSC | LP Oscillator Operating Frequency | 0 | — | 200 | kHz | All temperatures |
| | | RC Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
| | | XT Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
| | | HS Oscillator Operating Frequency | 0 | — | 20 | MHz | All temperatures |

* These parameters are characterized but not tested.

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

**Note 1:** This is the limit to which VDD can be lowered without losing RAM data.

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

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

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

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



Downloaded from Arrow.com.

## 12.2 DC Characteristics: PIC16C62XA-04 (Commercial, Industrial, Extended) PIC16C62XA-20 (Commercial, Industrial, Extended) PIC16LC62XA-04 (Commercial, Industrial, Extended)

| PIC16C62XA | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended |
|---|---|
| PIC16LC62XA | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended |

| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| D001 | VDD | Supply Voltage | 3.0 | — | 5.5 | V | See Figures 12-1, 12-2, 12-3, 12-4, and 12-5 |
| D001 | VDD | Supply Voltage | 2.5 | — | 5.5 | V | See Figures 12-1, 12-2, 12-3, 12-4, and 12-5 |
| D002 | VDR | RAM Data Retention Voltage(1) | — | 1.5* | — | V | Device in SLEEP mode |
| D002 | VDR | RAM Data Retention Voltage(1) | — | 1.5* | — | V | Device in SLEEP mode |
| D003 | VPOR | VDD start voltage to ensure Power-on Reset | — | VSS | — | V | See section on Power-on Reset for details |
| D003 | VPOR | VDD start voltage to ensure Power-on Reset | — | VSS | — | V | See section on Power-on Reset for details |
| D004 | SVDD | VDD rise rate to ensure Power-on Reset | 0.05* | — | — | V/ms | See section on Power-on Reset for details |
| D004 | SVDD | VDD rise rate to ensure Power-on Reset | 0.05* | — | — | V/ms | See section on Power-on Reset for details |
| D005 | VBOR | Brown-out Detect Voltage | 3.7 | 4.0 | 4.35 | V | BOREN configuration bit is cleared |
| D005 | VBOR | Brown-out Detect Voltage | 3.7 | 4.0 | 4.35 | V | BOREN configuration bit is cleared |

* These parameters are characterized but not tested.

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

**Note 1:** This is the limit to which VDD can be lowered without losing RAM data.

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

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

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

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

**6:** Commercial temperature range only.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 12.2 DC Characteristics: PIC16C62XA-04 (Commercial, Industrial, Extended) PIC16C62XA-20 (Commercial, Industrial, Extended) PIC16LC62XA-04 (Commercial, Industrial, Extended) (CONT.)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **PIC16C62XA** | | | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended | | | | |
| **PIC16LC62XA** | | | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended | | | | |
| **Param. No.** | **Sym** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| D010 | IDD | Supply Current(2, 4) | — | 1.2 | 2.0 | mA | FOSC = 4 MHz, VDD = 5.5V, WDT disabled, XT mode, **(Note 4)*** |
| | | | — | 0.4 | 1.2 | mA | FOSC = 4 MHz, VDD = 3.0V, WDT disabled, XT mode, **(Note 4)*** |
| | | | — | 1.0 | 2.0 | mA | FOSC = 10 MHz, VDD = 3.0V, WDT disabled, HS mode, **(Note 6)** |
| | | | — | 4.0 | 6.0 | mA | FOSC = 20 MHz, VDD = 4.5V, WDT disabled, HS mode |
| | | | — | 4.0 | 7.0 | mA | FOSC = 20 MHz, VDD = 5.5V, WDT disabled*, HS mode |
| | | | — | 35 | 70 | µA | FOSC = 32 kHz, VDD = 3.0V, WDT disabled, LP mode |
| D010 | IDD | Supply Current(2) | — | 1.2 | 2.0 | mA | FOSC = 4 MHz, VDD = 5.5V, WDT disabled, XT mode, **(Note 4)*** |
| | | | — | — | 1.1 | mA | FOSC = 4 MHz, VDD = 2.5V, WDT disabled, XT mode, **(Note 4)** |
| | | | — | 35 | 70 | µA | FOSC = 32 kHz, VDD = 2.5V, WDT disabled, LP mode |
| D020 | IPD | Power-down Current(3) | — | — | 2.2 | µA | VDD = 3.0V |
| | | | — | — | 5.0 | µA | VDD = 4.5V* |
| | | | — | — | 9.0 | µA | VDD = 5.5V |
| | | | — | — | 15 | µA | VDD = 5.5V Extended Temp. |
| D020 | IPD | Power-down Current(3) | — | — | 2.0 | µA | VDD = 2.5V |
| | | | — | — | 2.2 | µA | VDD = 3.0V* |
| | | | — | — | 9.0 | µA | VDD = 5.5V |
| | | | — | — | 15 | µA | VDD = 5.5V Extended Temp. |

* These parameters are characterized but not tested.

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

**Note 1:** This is the limit to which VDD can be lowered without losing RAM data.

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

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

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

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

**6:** Commercial temperature range only.



Downloaded from Arrow.com.

## 12.2 DC Characteristics: PIC16C62XA-04 (Commercial, Industrial, Extended) PIC16C62XA-20 (Commercial, Industrial, Extended) PIC16LC62XA-04 (Commercial, Industrial, Extended (CONT.)

| PIC16C62XA | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended |
|---|---|
| PIC16LC62XA | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended |

| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| D022 | ΔIWDT | WDT Current(5) | — | 6.0 | 10 | µA | VDD = 4.0V |
| | | | | | 12 | µA | (125°C) |
| D022A | ΔIBOR | Brown-out Reset Current(5) | — | 75 | 125 | µA | BOD enabled, VDD = 5.0V |
| D023 | ΔICOMP | Comparator Current for each Comparator(5) | — | 30 | 60 | µA | VDD = 4.0V |
| D023A | ΔIVREF | VREF Current(5) | — | 80 | 135 | µA | VDD = 4.0V |
| D022 | ΔIWDT | WDT Current(5) | — | 6.0 | 10 | µA | VDD=4.0V |
| | | | | | 12 | µA | (125°C) |
| D022A | ΔIBOR | Brown-out Reset Current(5) | — | 75 | 125 | µA | BOD enabled, VDD = 5.0V |
| D023 | ΔICOMP | Comparator Current for each Comparator(5) | — | 30 | 60 | µA | VDD = 4.0V |
| D023A | ΔIVREF | VREF Current(5) | — | 80 | 135 | µA | VDD = 4.0V |
| 1A | FOSC | LP Oscillator Operating Frequency | 0 | — | 200 | kHz | All temperatures |
| | | RC Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
| | | XT Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
| | | HS Oscillator Operating Frequency | 0 | — | 20 | MHz | All temperatures |
| 1A | FOSC | LP Oscillator Operating Frequency | 0 | — | 200 | kHz | All temperatures |
| | | RC Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
| | | XT Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
| | | HS Oscillator Operating Frequency | 0 | — | 20 | MHz | All temperatures |

* These parameters are characterized but not tested.

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

**Note 1:** This is the limit to which VDD can be lowered without losing RAM data.

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

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

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

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

**6:** Commercial temperature range only.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 12.3 DC CHARACTERISTICS: PIC16CR62XA-04 (Commercial, Industrial, Extended) PIC16CR62XA-20 (Commercial, Industrial, Extended) PIC16LCR62XA-04 (Commercial, Industrial, Extended)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **PIC16CR62XA-04<br>PIC16CR62XA-20** | | | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended | | | | |
| **PIC16LCR62XA-04** | | | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended | | | | |
| **Param. No.** | **Sym** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| D001 | VDD | Supply Voltage | 3.0 | — | 5.5 | V | See Figures 12-7, 12-8, 12-9 |
| D001 | VDD | Supply Voltage | 2.5 | — | 5.5 | V | See Figures 12-7, 12-8, 12-9 |
| D002 | VDR | RAM Data Retention Voltage(1) | — | 1.5* | — | V | Device in SLEEP mode |
| D002 | VDR | RAM Data Retention Voltage(1) | — | 1.5* | — | V | Device in SLEEP mode |
| D003 | VPOR | VDD start voltage to ensure Power-on Reset | — | VSS | — | V | See section on Power-on Reset for details |
| D003 | VPOR | VDD start voltage to ensure Power-on Reset | — | VSS | — | V | See section on Power-on Reset for details |
| D004 | SVDD | VDD rise rate to ensure Power-on Reset | 0.05* | — | — | V/ms | See section on Power-on Reset for details |
| D004 | SVDD | VDD rise rate to ensure Power-on Reset | 0.05* | — | — | V/ms | See section on Power-on Reset for details |
| D005 | VBOR | Brown-out Detect Voltage | 3.7 | 4.0 | 4.35 | V | BOREN configuration bit is cleared |
| D005 | VBOR | Brown-out Detect Voltage | 3.7 | 4.0 | 4.35 | V | BOREN configuration bit is cleared |
| D010 | IDD | Supply Current(2) | — | 1.2 | 1.7 | mA | FOSC = 4 MHz, VDD = 5.5V, WDT disabled, XT mode, (**Note 4**)* |
| | | | — | 500 | 900 | µA | FOSC = 4 MHz, VDD = 3.0V, WDT disabled, XT mode, (**Note 4**) |
| | | | — | 1.0 | 2.0 | mA | FOSC = 10 MHz, VDD = 3.0V, WDT disabled, HS mode, (**Note 6**) |
| | | | — | 4.0 | 7.0 | mA | FOSC = 20 MHz, VDD = 5.5V, WDT disabled*, HS |
| | | | — | 3.0 | 6.0 | mA | mode |
| | | | — | 35 | 70 | µA | FOSC = 20 MHz, VDD = 4.5V, WDT disabled, HS mode<br>FOSC = 32 kHz, VDD = 3.0V, WDT disabled, LP mode |
| D010 | IDD | Supply Current(2) | — | 1.2 | 1.7 | mA | FOSC = 4.0 MHz, VDD = 5.5V, WDT disabled, XT mode, (**Note 4**)* |
| | | | — | 400 | 800 | µA | FOSC = 4.0 MHz, VDD = 2.5V, WDT disabled, XT mode (**Note 4**) |
| | | | — | 35 | 70 | µA | FOSC = 32 kHz, VDD = 2.5V, WDT disabled, LP mode |



Downloaded from Arrow.com.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **PIC16CR62XA-04**<br>**PIC16CR62XA-20** | | | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended | | | | |
| **PIC16LCR62XA-04** | | | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended | | | | |
| **Param. No.** | **Sym** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |

* These parameters are characterized but not tested.

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

**Note 1:** This is the limit to which VDD can be lowered without losing RAM data.

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

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

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

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

**6:** Commercial temperature range only.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 12.3 DC CHARACTERISTICS: PIC16CR62XA-04 (Commercial, Industrial, Extended) PIC16CR62XA-20 (Commercial, Industrial, Extended) PIC16LCR62XA-04 (Commercial, Industrial, Extended) (CONT.)

| | |
|---|---|
| **PIC16CR62XA-04**<br>**PIC16CR62XA-20** | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended |
| **PIC16LCR62XA-04** | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended |

| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| D020 | IPD | Power-down Current(3) | — | 200 | 950 | nA | VDD = 3.0V |
| | | | — | 0.400 | 1.8 | µA | VDD = 4.5V* |
| | | | — | 0.600 | 2.2 | µA | VDD = 5.5V |
| | | | — | 5.0 | 9.0 | µA | VDD = 5.5V Extended Temp. |
| D020 | IPD | Power-down Current(3) | — | 200 | 850 | nA | VDD = 2.5V |
| | | | — | 200 | 950 | nA | VDD = 3.0V* |
| | | | — | 0.600 | 2.2 | µA | VDD = 5.5V |
| | | | — | 5.0 | 9.0 | µA | VDD = 5.5V Extended |
| D022 | ΔIWDT | WDT Current(5) | — | 6.0 | 10 | µA | VDD=4.0V |
| | | | | | 12 | µA | (125°C) |
| D022A | ΔIBOR | Brown-out Reset Current(5) | — | 75 | 125 | µA | BOD enabled, VDD = 5.0V |
| D023 | ΔICOMP | Comparator Current for each Comparator(5) | — | 30 | 60 | µA | VDD = 4.0V |
| D023A | ΔIVREF | VREF Current(5) | — | 80 | 135 | µA | VDD = 4.0V |
| D022 | ΔIWDT | WDT Current(5) | — | 6.0 | 10 | µA | VDD=4.0V |
| | | | | | 12 | µA | (125°C) |
| D022A | ΔIBOR | Brown-out Reset Current(5) | — | 75 | 125 | µA | BOD enabled, VDD = 5.0V |
| D023 | ΔICOMP | Comparator Current for each Comparator(5) | — | 30 | 60 | µA | VDD = 4.0V |
| D023A | ΔIVREF | VREF Current(5) | — | 80 | 135 | µA | VDD = 4.0V |
| 1A | FOSC | LP Oscillator Operating Frequency | 0 | — | 200 | kHz | All temperatures |
| | | RC Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
| | | XT Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
| | | HS Oscillator Operating Frequency | 0 | — | 20 | MHz | All temperatures |
| 1A | FOSC | LP Oscillator Operating Frequency | 0 | — | 200 | kHz | All temperatures |
| | | RC Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
| | | XT Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
| | | HS Oscillator Operating Frequency | 0 | — | 20 | MHz | All temperatures |

* These parameters are characterized but not tested.

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

**Note 1:** This is the limit to which VDD can be lowered without losing RAM data.

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

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

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

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

**6:** Commercial temperature range only.



Downloaded from Arrow.com.

## 12.4 DC Characteristics: PIC16C62X/C62XA/CR62XA (Commercial, Industrial, Extended) PIC16LC62X/LC62XA/LCR62XA (Commercial, Industrial, Extended)

| PIC16C62X/C62XA/CR62XA | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended |
|---|---|
| PIC16LC62X/LC62XA/LCR62XA | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended |

| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| | VIL | **Input Low Voltage** | | | | | |
| | | I/O ports | | | | | |
| D030 | | with TTL buffer | VSS | — | 0.8V<br>0.15 VDD | V | VDD = 4.5V to 5.5V<br>otherwise |
| D031 | | with Schmitt Trigger input | VSS | — | 0.2 VDD | V | |
| D032 | | $\overline{MCLR}$, RA4/T0CKI,OSC1 (in RC mode) | Vss | — | 0.2 VDD | V | **(Note 1)** |
| D033 | | OSC1 (in XT and HS) | Vss | — | 0.3 VDD | V | |
| | | OSC1 (in LP) | Vss | — | 0.6 VDD-1.0 | V | |
| | VIL | **Input Low Voltage** | | | | | |
| | | I/O ports | | | | | |
| D030 | | with TTL buffer | VSS | — | 0.8V<br>0.15 VDD | V | VDD = 4.5V to 5.5V<br>otherwise |
| D031 | | with Schmitt Trigger input | VSS | — | 0.2 VDD | V | |
| D032 | | $\overline{MCLR}$, RA4/T0CKI,OSC1 (in RC mode) | Vss | — | 0.2 VDD | V | **(Note 1)** |
| D033 | | OSC1 (in XT and HS) | Vss | — | 0.3 VDD | V | |
| | | OSC1 (in LP) | Vss | — | 0.6 VDD-1.0 | V | |
| | VIH | Input High Voltage | | | | | |
| | | I/O ports | | | | | |
| D040 | | with TTL buffer | 2.0V<br>0.25 VDD + 0.8V | — | VDD<br>VDD | V | VDD = 4.5V to 5.5V<br>otherwise |
| D041 | | with Schmitt Trigger input | 0.8 VDD | — | VDD | | |
| D042 | | $\overline{MCLR}$ RA4/T0CKI | 0.8 VDD | — | VDD | V | |
| D043<br>D043A | | OSC1 (XT, HS and LP)<br>OSC1 (in RC mode) | 0.7 VDD<br>0.9 VDD | — | VDD | V | **(Note 1)** |

* These parameters are characterized but not tested.

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

**Note 1:** In RC oscillator configuration, the OSC1 pin is a Schmitt Trigger input. It is not recommended that the PIC16C62X(A) be driven with external clock in RC mode.

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

**3:** Negative current is defined as coming out of the pin.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 12.4 DC Characteristics: PIC16C62X/C62XA/CR62XA (Commercial, Industrial, Extended) PIC16LC62X/LC62XA/LCR62XA (Commercial, Industrial, Extended) (CONT.)

| PIC16C62X/C62XA/CR62XA | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended |
|---|---|
| **PIC16LC62X/LC62XA/LCR62XA** | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended |

| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| | VIH | Input High Voltage | | | | | |
| | | I/O ports | | | | | |
| D040 | | with TTL buffer | 2.0V<br>0.25 VDD + 0.8V | — | VDD<br>VDD | V | VDD = 4.5V to 5.5V<br>otherwise |
| D041 | | with Schmitt Trigger input | 0.8 VDD | — | VDD | | |
| D042 | | $\overline{MCLR}$ RA4/T0CKI | 0.8 VDD | — | VDD | V | |
| D043<br>D043A | | OSC1 (XT, HS and LP)<br>OSC1 (in RC mode) | 0.7 VDD<br>0.9 VDD | — | VDD | V | **(Note 1)** |
| D070 | IPURB | PORTB weak pull-up current | 50 | 200 | 400 | µA | VDD = 5.0V, VPIN = VSS |
| D070 | IPURB | PORTB weak pull-up current | 50 | 200 | 400 | µA | VDD = 5.0V, VPIN = VSS |
| | IIL | **Input Leakage Current(2, 3)**<br>I/O ports (Except PORTA) | | | ±1.0 | µA | VSS ≤ VPIN ≤ VDD, pin at hi-impedance |
| D060 | | PORTA | — | — | ±0.5 | µA | Vss ≤ VPIN ≤ VDD, pin at hi-impedance |
| D061 | | RA4/T0CKI | — | — | ±1.0 | µA | Vss ≤ VPIN ≤ VDD |
| D063 | | OSC1, $\overline{MCLR}$ | — | — | ±5.0 | µA | Vss ≤ VPIN ≤ VDD, XT, HS and LP osc configuration |
| | IIL | **Input Leakage Current(2, 3)**<br>I/O ports (Except PORTA) | | | ±1.0 | µA | VSS ≤ VPIN ≤ VDD, pin at hi-impedance |
| D060 | | PORTA | — | — | ±0.5 | µA | Vss ≤ VPIN ≤ VDD, pin at hi-impedance |
| D061 | | RA4/T0CKI | — | — | ±1.0 | µA | Vss ≤ VPIN ≤ VDD |
| D063 | | OSC1, $\overline{MCLR}$ | — | — | ±5.0 | µA | Vss ≤ VPIN ≤ VDD, XT, HS and LP osc configuration |
| | VOL | Output Low Voltage | | | | | |
| D080 | | I/O ports | — | — | 0.6 | V | IOL = 8.5 mA, VDD = 4.5V, -40° to +85°C |
| | | | — | — | 0.6 | V | IOL = 7.0 mA, VDD = 4.5V, +125°C |
| D083 | | OSC2/CLKOUT (RC only) | — | — | 0.6 | V | IOL = 1.6 mA, VDD = 4.5V, -40° to +85°C |
| | | | — | — | 0.6 | V | IOL = 1.2 mA, VDD = 4.5V, +125°C |

* These parameters are characterized but not tested.

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

**Note 1:** In RC oscillator configuration, the OSC1 pin is a Schmitt Trigger input. It is not recommended that the PIC16C62X(A) be driven with external clock in RC mode.

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

**3:** Negative current is defined as coming out of the pin.



Downloaded from Arrow.com.

## 12.4 DC Characteristics: PIC16C62X/C62XA/CR62XA (Commercial, Industrial, Extended) PIC16LC62X/LC62XA/LCR62XA (Commercial, Industrial, Extended) (CONT.)

| PIC16C62X/C62XA/CR62XA | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended |
|---|---|
| **PIC16LC62X/LC62XA/LCR62XA** | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial and<br>0°C ≤ TA ≤ +70°C for commercial and<br>-40°C ≤ TA ≤ +125°C for extended |

| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| | VOL | Output Low Voltage | | | | | |
| D080 | | I/O ports | — | — | 0.6 | V | IOL = 8.5 mA, VDD = 4.5V, -40° to +85°C |
| | | | — | — | 0.6 | V | IOL = 7.0 mA, VDD = 4.5V, +125°C |
| D083 | | OSC2/CLKOUT (RC only) | — | — | 0.6 | V | IOL = 1.6 mA, VDD = 4.5V, -40° to +85°C |
| | | | — | — | 0.6 | V | IOL = 1.2 mA, VDD = 4.5V, +125°C |
| | VOH | Output High Voltage(3) | | | | | |
| D090 | | I/O ports (Except RA4) | VDD-0.7 | — | — | V | IOH = -3.0 mA, VDD = 4.5V, -40° to +85°C |
| | | | VDD-0.7 | — | — | V | IOH = -2.5 mA, VDD = 4.5V, +125°C |
| D092 | | OSC2/CLKOUT (RC only) | VDD-0.7 | — | — | V | IOH = -1.3 mA, VDD = 4.5V, -40° to +85°C |
| | | | VDD-0.7 | — | — | V | IOH = -1.0 mA, VDD = 4.5V, +125°C |
| | VOH | Output High Voltage(3) | | | | | |
| D090 | | I/O ports (Except RA4) | VDD-0.7 | — | — | V | IOH = -3.0 mA, VDD = 4.5V, -40° to +85°C |
| | | | VDD-0.7 | — | — | V | IOH = -2.5 mA, VDD = 4.5V, +125°C |
| D092 | | OSC2/CLKOUT (RC only) | VDD-0.7 | — | — | V | IOH = -1.3 mA, VDD = 4.5V, -40° to +85°C |
| | | | VDD-0.7 | — | — | V | IOH = -1.0 mA, VDD = 4.5V, +125°C |
| *D150 | VOD | Open-Drain High Voltage | | | 10*<br>8.5* | V | RA4 pin PIC16C62X, PIC16LC62X<br>RA4 pin PIC16C62XA, PIC16LC62XA,<br>PIC16CR62XA, PIC16LCR62XA |
| *D150 | VOD | Open-Drain High Voltage | | | 10*<br>8.5* | V | RA4 pin PIC16C62X, PIC16LC62X<br>RA4 pin PIC16C62XA, PIC16LC62XA,<br>PIC16CR62XA, PIC16LCR62XA |
| | | Capacitive Loading Specs on Output Pins | | | | | |
| D100 | COSC2 | OSC2 pin | | | 15 | pF | In XT, HS and LP modes when external clock used to drive OSC1. |
| D101 | CIO | All I/O pins/OSC2 (in RC mode) | | | 50 | pF | |
| | | Capacitive Loading Specs on Output Pins | | | | | |
| D100 | COSC2 | OSC2 pin | | | 15 | pF | In XT, HS and LP modes when external clock used to drive OSC1. |
| D101 | CIO | All I/O pins/OSC2 (in RC mode) | | | 50 | pF | |

* These parameters are characterized but not tested.

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

**Note 1:** In RC oscillator configuration, the OSC1 pin is a Schmitt Trigger input. It is not recommended that the PIC16C62X(A) be driven with external clock in RC mode.

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

**3:** Negative current is defined as coming out of the pin.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 12.5 DC CHARACTERISTICS: PIC16C620A/C621A/C622A-40(7) (Commercial) PIC16CR620A-40(7) (Commercial)

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated) Operating temperature 0°C ≤ TA ≤ +70°C for commercial | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| D001 | VDD | Supply Voltage | 3.0 | — | 5.5 | V | FOSC = DC to 20 MHz |
| D002 | VDR | RAM Data Retention Voltage(1) | — | 1.5* | — | V | Device in SLEEP mode |
| D003 | VPOR | VDD start voltage to ensure Power-on Reset | — | VSS | — | V | See section on Power-on Reset for details |
| D004 | SVDD | VDD rise rate to ensure Power-on Reset | 0.05 * | — | — | V/ms | See section on Power-on Reset for details |
| D005 | VBOR | Brown-out Detect Voltage | 3.65 | 4.0 | 4.35 | V | BOREN configuration bit is cleared |
| D010 | IDD | Supply Current(2,4) | — | 1.2 | 2.0 | mA | FOSC = 4 MHz, VDD = 5.5V, WDT disabled, XT OSC mode, **(Note 4)*** |
| | | | — | 0.4 | 1.2 | mA | FOSC = 4 MHz, VDD = 3.0V, WDT disabled, XT OSC mode, **(Note 4)** |
| | | | — | 1.0 | 2.0 | mA | FOSC = 10 MHz, VDD = 3.0V, WDT disabled, HS OSC mode, **(Note 6)** |
| | | | — | 4.0 | 6.0 | mA | FOSC = 20 MHz, VDD = 4.5V, WDT disabled, HS OSC mode |
| | | | — | 4.0 | 7.0 | mA | FOSC = 20 MHz, VDD = 5.5V, WDT disabled*, HS OSC mode |
| | | | — | 35 | 70 | µA | FOSC = 32 kHz, VDD = 3.0V, WDT disabled, LP OSC mode |
| D020 | IPD | Power Down Current(3) | — | — | 2.2 | µA | VDD = 3.0V |
| | | | — | — | 5.0 | µA | VDD = 4.5V* |
| | | | — | — | 9.0 | µA | VDD = 5.5V |
| | | | — | — | 15 | µA | VDD = 5.5V Extended |
| D022 | ΔIWDT | WDT Current(5) | — | 6.0 | 10 | µA | VDD = 4.0V |
| | | | | | 12 | µA | (125°C) |
| D022A | ΔIBOR | Brown-out Reset Current(5) | — | 75 | 125 | µA | BOD enabled, VDD = 5.0V |
| D023 | ΔICOMP | Comparator Current for each Comparator(5) | — | 30 | 60 | µA | VDD = 4.0V |
| D023A | ΔIVREF | VREF Current(5) | — | 80 | 135 | µA | VDD = 4.0V |
| | ΔIEE Write | Operating Current | — | | 3 | mA | VCC = 5.5V, SCL = 400 kHz |
| | ΔIEE Read | Operating Current | — | | 1 | mA | |
| | ΔIEE | Standby Current | — | | 30 | µA | VCC = 3.0V, EE VDD = VCC |
| | ΔIEE | Standby Current | — | | 100 | µA | VCC = 3.0V, EE VDD = VCC |
| 1A | FOSC | LP Oscillator Operating Frequency | 0 | — | 200 | kHz | All temperatures |
| | | RC Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
| | | XT Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
| | | HS Oscillator Operating Frequency | 0 | — | 20 | MHz | All temperatures |

* 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.

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

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

**4:** For RC OSC configuration, current through REXT is not included. The current through the resistor can be estimated by the formula Ir = VDD/2REXT (mA) with REXT in kΩ.

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

**6:** Commercial temperature range only.

**7:** See Section 12.1 and Section 12.3 for 16C62X and 16CR62X devices for operation between 20 MHz and 40 MHz for valid modified characteristics.



Downloaded from Arrow.com.

## 12.5 DC CHARACTERISTICS: PIC16C620A/C621A/C622A-40(7) (Commercial) PIC16CR620A-40(7) (Commercial)

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature 0°C ≤ TA ≤ +70°C for commercial | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym** | **Characteristic** | **Min** | **Typ†** | **Max** | **Unit** | **Conditions** |
| | VIL | **Input Low Voltage** | | | | | |
| | | I/O ports | | | | | |
| D030 | | with TTL buffer | VSS | — | 0.8V<br>0.15VDD | V | VDD = 4.5V to 5.5V, otherwise |
| D031 | | with Schmitt Trigger input | VSS | | 0.2VDD | V | |
| D032 | | MCLR, RA4/T0CKI, OSC1 (in RC mode) | VSS | — | 0.2VDD | V | **(Note 1)** |
| D033 | | OSC1 (in XT and HS) | VSS | — | 0.3VDD | V | |
| | | OSC1 (in LP) | VSS | — | 0.6VDD - 1.0 | V | |
| | VIH | **Input High Voltage** | | | | | |
| | | I/O ports | | | | | |
| D040 | | with TTL buffer | 2.0V<br>0.25 VDD + 0.8 | — | VDD<br>VDD | V | VDD = 4.5V to 5.5V, otherwise |
| D041 | | with Schmitt Trigger input | 0.8 VDD | | VDD | | |
| D042 | | MCLR RA4/T0CKI | 0.8 VDD | — | VDD | V | |
| D043 | | OSC1 (XT, HS and LP) | 0.7 VDD | — | VDD | V | |
| D043A | | OSC1 (in RC mode) | 0.9 VDD | | | | **(Note 1)** |
| D070 | IPURB | **PORTB Weak Pull-up Current** | 50 | 200 | 400 | µA | VDD = 5.0V, VPIN = VSS |
| | IIL | **Input Leakage Current(2, 3)** | | | | | |
| | | I/O ports (except PORTA) | | | ±1.0 | µA | VSS ≤ VPIN ≤ VDD, pin at hi-impedance |
| D060 | | PORTA | — | — | ±0.5 | µA | Vss ≤ VPIN ≤ VDD, pin at hi-impedance |
| D061 | | RA4/T0CKI | — | — | ±1.0 | µA | Vss ≤ VPIN ≤ VDD |
| D063 | | OSC1, MCLR | — | — | ±5.0 | µA | Vss ≤ VPIN ≤ VDD, XT, HS and LP OSC configuration |
| | VOL | **Output Low Voltage** | | | | | |
| D080 | | I/O ports | — | — | 0.6 | V | IOL = 8.5 mA, VDD = 4.5V, -40° to +85°C |
| | | | — | — | 0.6 | V | IOL = 7.0 mA, VDD = 4.5V, +125°C |
| D083 | | OSC2/CLKOUT (RC only) | — | — | 0.6 | V | IOL = 1.6 mA, VDD = 4.5V, -40° to +85°C |
| | | | — | — | 0.6 | V | IOL = 1.2 mA, VDD = 4.5V, +125°C |
| | VOH | **Output High Voltage(3)** | | | | | |
| D090 | | I/O ports (except RA4) | VDD-0.7 | — | — | V | IOH = -3.0 mA, VDD = 4.5V, -40° to +85°C |
| | | | VDD-0.7 | — | — | V | IOH = -2.5 mA, VDD = 4.5V, +125°C |
| D092 | | OSC2/CLKOUT (RC only) | VDD-0.7 | — | — | V | IOH = -1.3 mA, VDD = 4.5V, -40° to +85°C |
| | | | VDD-0.7 | — | — | V | IOH = -1.0 mA, VDD = 4.5V, +125°C |
| *D150 | VOD | **Open Drain High Voltage** | | | 8.5 | V | RA4 pin |
| | | **Capacitive Loading Specs on Output Pins** | | | | | |
| D100 | COSC2 | OSC2 pin | | | 15 | pF | In XT, HS and LP modes when external clock used to drive OSC1. |
| D101 | CIO | All I/O pins/OSC2 (in RC mode) | | | 50 | pF | |

* 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.

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

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

**4:** For RC OSC configuration, current through REXT is not included. The current through the resistor can be estimated by the formula Ir = VDD/2REXT (mA) with REXT in kΩ.

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

**6:** Commercial temperature range only.

**7:** See Section 12.1 and Section 12.3 for 16C62X and 16CR62X devices for operation between 20 MHz and 40 MHz for valid modified characteristics.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 12.6 DC Characteristics: PIC16C620A/C621A/C622A-40(3) (Commercial) PIC16CR620A-40(3) (Commercial)

| DC CHARACTERISTICS<br>Power Supply Pins | | | | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature 0°C ≤ TA ≤ +70°C for commercial | | |
|---|---|---|---|---|---|---|
| **Characteristic** | **Sym** | **Min** | **Typ(1)** | **Max** | **Units** | **Conditions** |
| Supply Voltage | VDD | 4.5 | — | 5.5 | V | HS Option from 20 - 40 MHz |
| Supply Current(2) | IDD | —<br>— | 5.5<br>7.7 | 11.5<br>16 | mA<br>mA | FOSC = 40 MHz, VDD = 4.5V, HS mode<br>FOSC = 40 MHz, VDD = 5.5V, HS mode |
| HS Oscillator Operating Frequency | FOSC | 20 | — | 40 | MHz | OSC1 pin is externally driven, OSC2 pin not connected |
| Input Low Voltage OSC1 | VIL | VSS | — | 0.2VDD | V | HS mode, OSC1 externally driven |
| Input High Voltage OSC1 | VIH | 0.8VDD | — | VDD | V | HS mode, OSC1 externally driven |

* These parameters are characterized but not tested.

**Note 1:** Data in the Typical ("Typ") column is based on characterization results at 25°C. This data is for design guidance only and is not tested.

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

a) 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 VSS, T0CKI = VDD, $\overline{MCLR}$ = VDD; WDT disabled, HS mode with OSC2 not connected.

**3:** For device operation between DC and 20 MHz. See Table 12-1 and Table 12-2.

## 12.7 AC Characteristics: PIC16C620A/C621A/C622A-40(2) (Commercial) PIC16CR620A-40(2) (Commercial)

| AC CHARACTERISTICS<br>All Pins Except Power Supply Pins | | | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature 0°C ≤ TA ≤ +70°C for commercial | | | |
|---|---|---|---|---|---|---|
| **Characteristic** | **Sym** | **Min** | **Typ(1)** | **Max** | **Units** | **Conditions** |
| External CLKIN Frequency | FOSC | 20 | — | 40 | MHz | HS mode, OSC1 externally driven |
| External CLKIN Period | TOSC | 25 | — | 50 | ns | HS mode (40), OSC1 externally driven |
| Clock in (OSC1) Low or High Time | TOSL, TOSH | 6 | — | — | ns | HS mode, OSC1 externally driven |
| Clock in (OSC1) Rise or Fall Time | TOSR, TOSF | — | — | 6.5 | ns | HS mode, OSC1 externally driven |
| OSC1↑ (Q1 cycle) to Port out valid | TOSH2IOV | — | — | 100 | ns | — |
| OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time) | TOSH2IOI | 50 | — | — | ns | — |

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

**2:** For device operation between DC and 20 MHz. See Table 12-1 and Table 12-2.



Downloaded from Arrow.com.

### TABLE 12-1: COMPARATOR SPECIFICATIONS

Operating Conditions: $V_{DD}$ range as described in Table 12-1, -40°C<$T_A$<+125°C. Current consumption is specified in Table 12-1.

| Characteristics | Sym | Min | Typ | Max | Units | Comments |
|---|---|---|---|---|---|---|
| Input offset voltage | | | ± 5.0 | ± 10 | mV | |
| Input common mode voltage | | 0 | | $V_{DD}$ - 1.5 | V | |
| CMRR | | +55* | | | δβ | |
| Response Time(1) | | | 150* | 400*<br>600* | ns<br>ns | PIC16C62X(A)<br>PIC16LC62X |
| Comparator mode change to output valid | | | | 10* | μs | |

* These parameters are characterized but not tested.

**Note 1:** Response time measured with one comparator input at ($V_{DD}$ - 1.5)/2, while the other input transitions from $V_{SS}$ to $V_{DD}$.

### TABLE 12-2: VOLTAGE REFERENCE SPECIFICATIONS

Operating Conditions:$V_{DD}$ range as described in Table 12-1, -40°C<$T_A$<+125°C. Current consumption is specified in Table 12-1.

| Characteristics | Sym | Min | Typ | Max | Units | Comments |
|---|---|---|---|---|---|---|
| Resolution | | | $V_{DD}$/24<br>$V_{DD}$/32 | | LSB<br>LSB | Low Range ($V_{RR}$=1)<br>High Range ($V_{RR}$=0) |
| Absolute Accuracy | | | | ±1/4<br>±1/2 | LSB<br>LSB | Low Range ($V_{RR}$=1)<br>High Range ($V_{RR}$=0) |
| Unit Resistor Value (R) | | | 2K* | | Ω | Figure 8-1 |
| Settling Time(1) | | | | 10* | μs | |

* These parameters are characterized but not tested.

**Note 1:** Settling time measured while $V_{RR}$ = 1 and $V_R$<3:0> transitions from 0000 to 1111.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 12.8 Timing Parameter Symbology

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

1. TppS2ppS
2. TppS

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

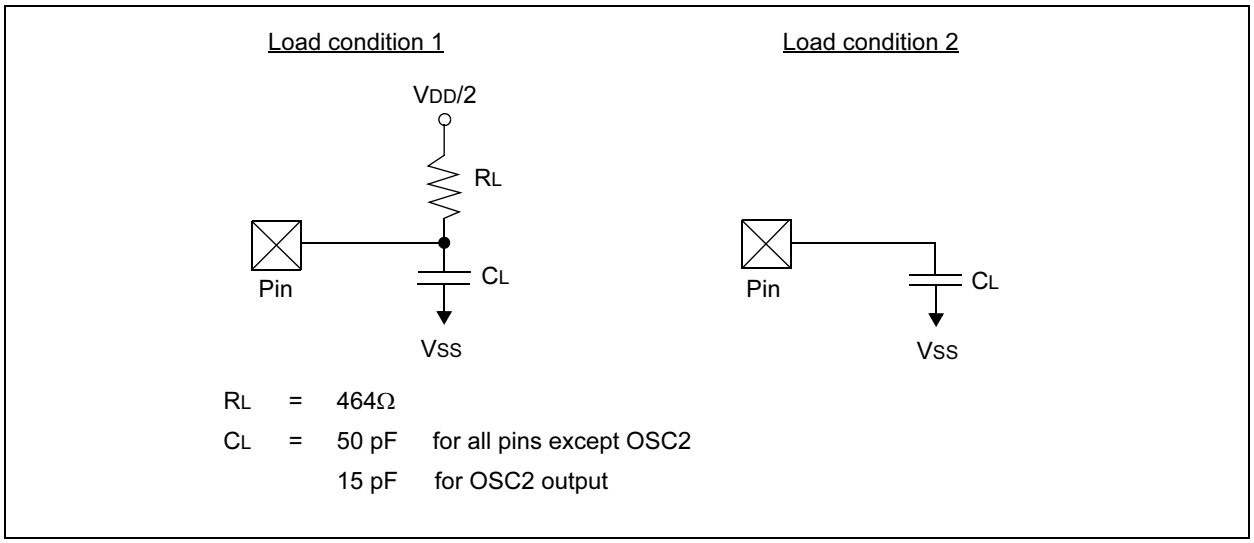
Lowercase subscripts (pp) and their meanings:

| | | | |
|---|---|---|---|
| **pp** | | | |
| ck | CLKOUT | osc | OSC1 |
| io | I/O port | t0 | T0CKI |
| mc | $\overline{\text{MCLR}}$ | | |

Uppercase letters and their meanings:

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

**FIGURE 12-11: LOAD CONDITIONS**






Downloaded from Arrow.com.

## 12.9 Timing Diagrams and Specifications

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

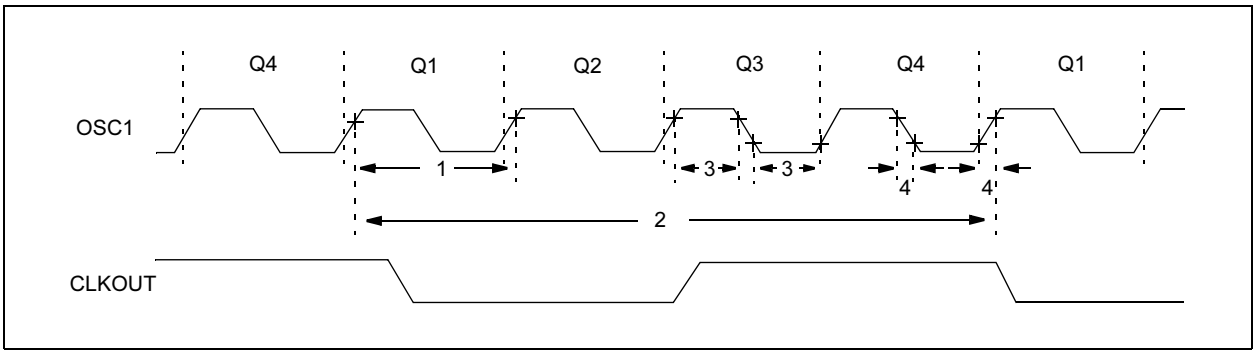



**TABLE 12-3: EXTERNAL CLOCK TIMING REQUIREMENTS**

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| 1A | Fosc | External CLKIN Frequency(1) | DC | — | 4 | MHz | XT and RC Osc mode, VDD=5.0V |
| | | | DC | — | 20 | MHz | HS Osc mode |
| | | | DC | — | 200 | kHz | LP Osc mode |
| | | Oscillator Frequency(1) | DC | — | 4 | MHz | RC Osc mode, VDD=5.0V |
| | | | 0.1 | — | 4 | MHz | XT Osc mode |
| | | | 1 | — | 20 | MHz | HS Osc mode |
| | | | DC | — | 200 | kHz | LP Osc mode |
| 1 | Tosc | External CLKIN Period(1) | 250 | — | — | ns | XT and RC Osc mode |
| | | | 50 | — | — | ns | HS Osc mode |
| | | | 5 | — | — | µs | LP Osc mode |
| | | Oscillator Period(1) | 250 | — | — | ns | RC Osc mode |
| | | | 250 | — | 10,000 | ns | XT Osc mode |
| | | | 50 | — | 1,000 | ns | HS Osc mode |
| | | | 5 | — | — | µs | LP Osc mode |
| 2 | TCY | Instruction Cycle Time(1) | 1.0 | FOSC/4 | DC | µs | TCYS=FOSC/4 |
| 3* | TosL, TosH | External Clock in (OSC1) High or Low Time | 100* | — | — | ns | XT oscillator, TOSC L/H duty cycle |
| | | | 2* | — | — | µs | LP oscillator, TOSC L/H duty cycle |
| | | | 20* | — | — | ns | HS oscillator, TOSC L/H duty cycle |
| 4* | TosR, TosF | External Clock in (OSC1) Rise or Fall Time | 25* | — | — | ns | XT oscillator |
| | | | 50* | — | — | ns | LP oscillator |
| | | | 15* | — | — | ns | HS oscillator |

**2:** * These parameters are characterized but not tested.

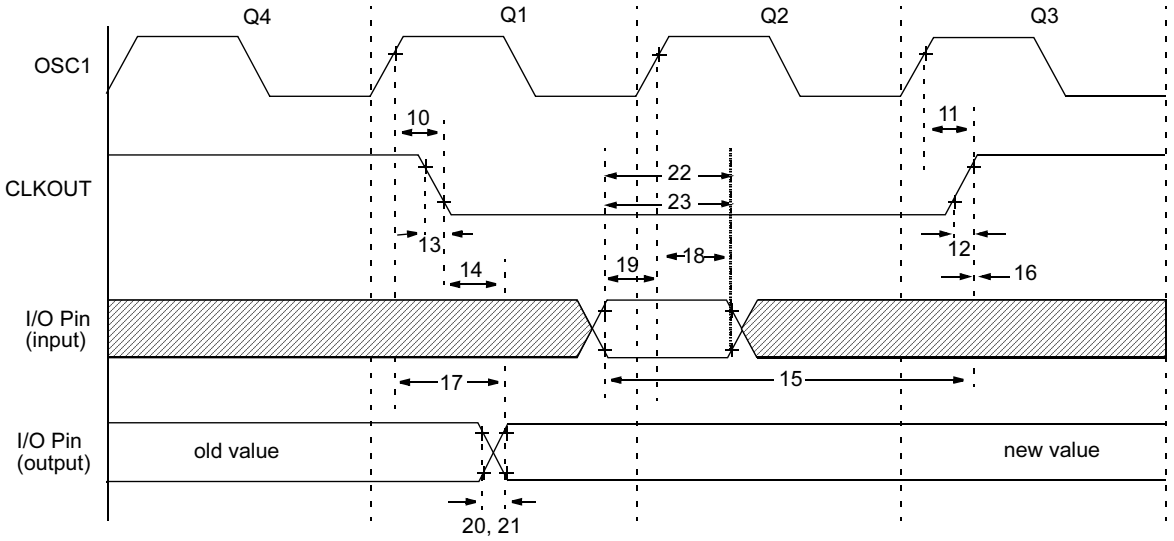
**3:** † Data in "Typ" column is at 5.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

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


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

**FIGURE 12-13: CLKOUT AND I/O TIMING**




**Note:** All tests must be done with specified capacitance loads (Figure 12-11) 50 pF on I/O pins and CLKOUT.



Downloaded from Arrow.com.

**TABLE 12-4: CLKOUT AND I/O TIMING REQUIREMENTS**

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| 10* | TosH2ckL | OSC1↑ to CLKOUT↓(1) | —<br>— | 75<br>— | 200<br>400 | ns<br>ns | PIC16C62X(A)<br>PIC16LC62X(A)<br>PIC16CR62XA<br>PIC16LCR62XA |
| 11* | TosH2ckH | OSC1↑ to CLKOUT↑(1) | —<br>— | 75<br>— | 200<br>400 | ns<br>ns | PIC16C62X(A)<br>PIC16LC62X(A)<br>PIC16CR62XA<br>PIC16LCR62XA |
| 12* | TckR | CLKOUT rise time(1) | —<br>— | 35<br>— | 100<br>200 | ns<br>ns | PIC16C62X(A)<br>PIC16LC62X(A)<br>PIC16CR62XA<br>PIC16LCR62XA |
| 13* | TckF | CLKOUT fall time(1) | —<br>— | 35<br>— | 100<br>200 | ns<br>ns | PIC16C62X(A)<br>PIC16LC62X(A)<br>PIC16CR62XA<br>PIC16LCR62XA |
| 14* | TckL2ioV | CLKOUT ↓ to Port out valid(1) | — | — | 20 | ns | |
| 15* | TioV2ckH | Port in valid before CLKOUT ↑(1) | TOSC +200 ns<br>TOSC +400 ns | —<br>— | —<br>— | ns<br>ns | PIC16C62X(A)<br>PIC16LC62X(A)<br>PIC16CR62XA<br>PIC16LCR62XA |
| 16* | TckH2iol | Port in hold after CLKOUT ↑(1) | 0 | — | — | ns | |
| 17* | TosH2ioV | OSC1↑ (Q1 cycle) to Port out valid | —<br>— | 50 | 150<br>300 | ns<br>ns | PIC16C62X(A)<br>PIC16LC62X(A)<br>PIC16CR62XA<br>PIC16LCR62XA |
| 18* | TosH2iol | OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time) | 100<br>200 | —<br>— | —<br>— | ns<br>ns | PIC16C62X(A)<br>PIC16LC62X(A)<br>PIC16CR62XA<br>PIC16LCR62XA |
| 19* | TioV2osH | Port input valid to OSC1↑ (I/O in setup time) | 0 | — | — | ns | |
| 20* | TioR | Port output rise time | —<br>— | 10<br>— | 40<br>80 | ns<br>ns | PIC16C62X(A)<br>PIC16LC62X(A)<br>PIC16CR62XA<br>PIC16LCR62XA |
| 21* | TioF | Port output fall time | —<br>— | 10<br>— | 40<br>80 | ns<br>ns | PIC16C62X(A)<br>PIC16LC62X(A)<br>PIC16CR62XA<br>PIC16LCR62XA |
| 22* | Tinp | RB0/INT pin high or low time | 25<br>40 | —<br>— | —<br>— | ns<br>ns | PIC16C62X(A)<br>PIC16LC62X(A)<br>PIC16CR62XA<br>PIC16LCR62XA |
| 23 | Trbp | RB<7:4> change interrupt high or low time | TCY | — | — | ns | |

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


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

**FIGURE 12-14: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING**

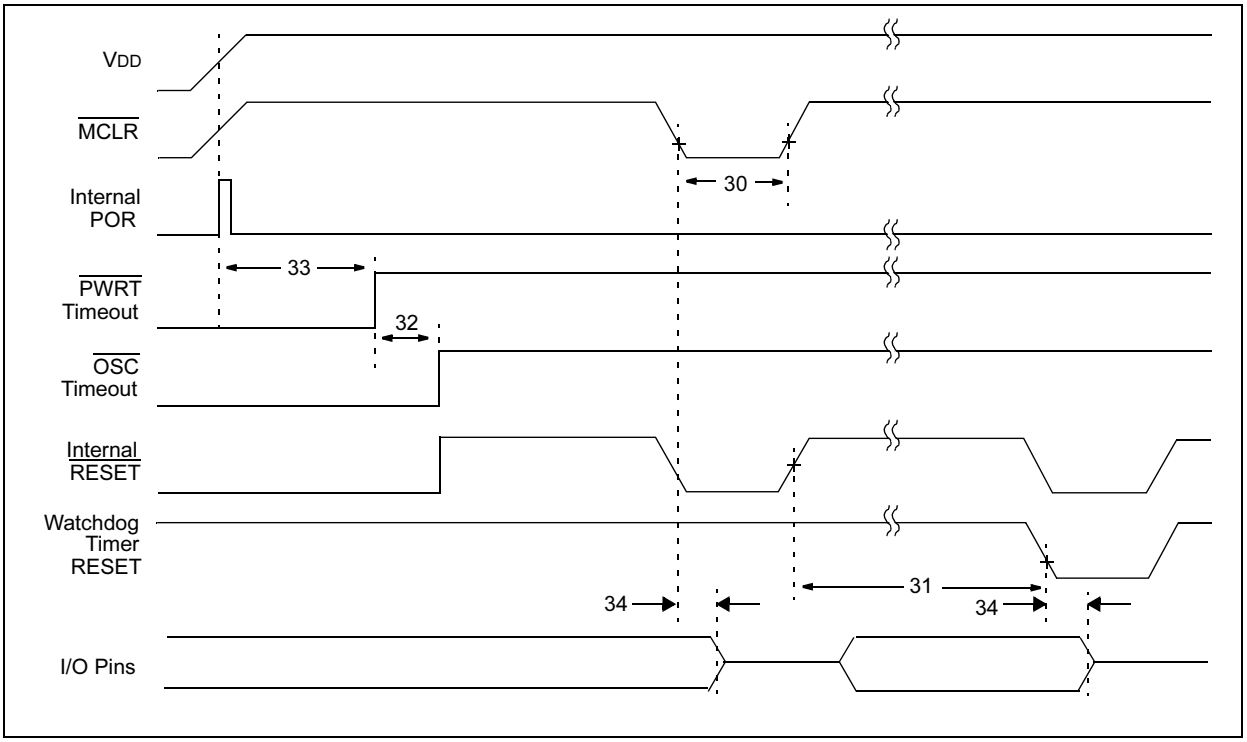



**FIGURE 12-15: BROWN-OUT RESET TIMING**

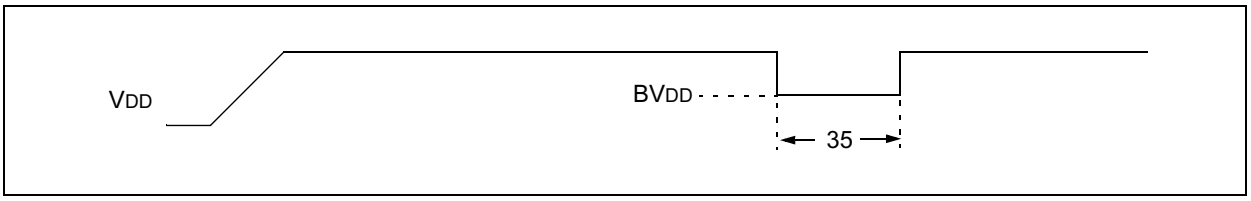



**TABLE 12-5: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER REQUIREMENTS**

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| 30 | TmcL | $\overline{\text{MCLR}}$ Pulse Width (low) | 2000 | — | — | ns | -40° to +85°C |
| 31 | Twdt | Watchdog Timer Time-out Period (No Prescaler) | 7* | 18 | 33* | ms | VDD = 5.0V, -40° to +85°C |
| 32 | Tost | Oscillation Start-up Timer Period | — | 1024 TOSC | — | — | TOSC = OSC1 period |
| 33 | Tpwrt | Power-up Timer Period | 28* | 72 | 132* | ms | VDD = 5.0V, -40° to +85°C |
| 34 | TIOZ | I/O hi-impedance from $\overline{\text{MCLR}}$ low | | — | 2.0 | µs | |
| 35 | TBOR | Brown-out Reset Pulse Width | 100* | — | — | µs | 3.7V ≤ VDD ≤ 4.3V |

* 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.



Downloaded from Arrow.com.

**FIGURE 12-16: TIMER0 CLOCK TIMING**

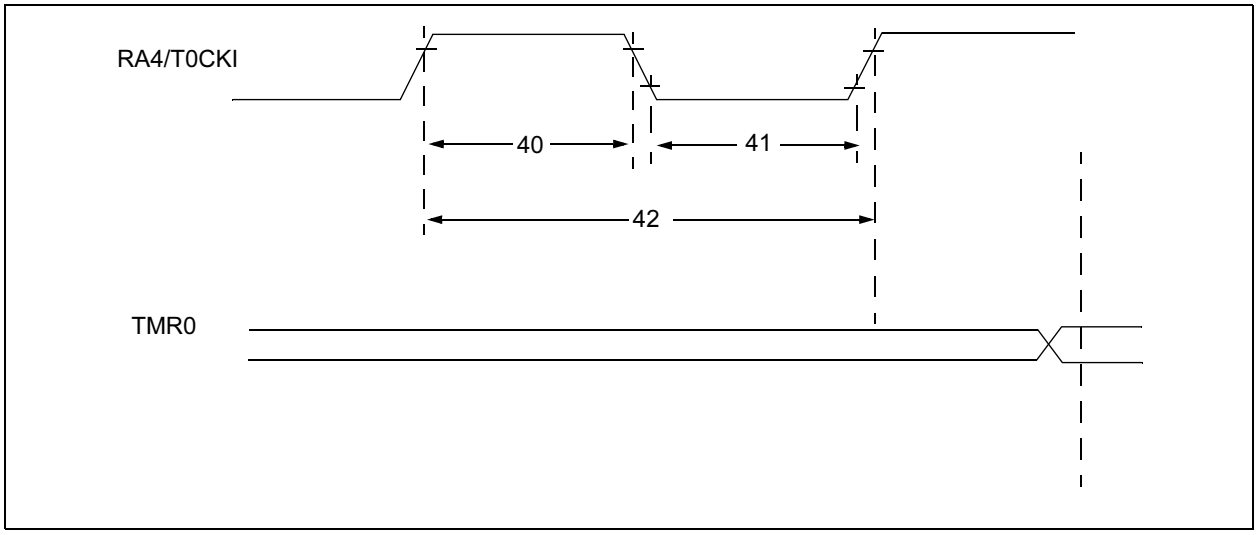



**TABLE 12-6: TIMER0 CLOCK REQUIREMENTS**

| Parameter No. | Sym | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 40 | Tt0H | T0CKI High Pulse Width | No Prescaler | 0.5 TCY + 20* | — | — | ns | |
| | | | With Prescaler | 10* | — | — | ns | |
| 41 | Tt0L | T0CKI Low Pulse Width | No Prescaler | 0.5 TCY + 20* | — | — | ns | |
| | | | With Prescaler | 10* | — | — | ns | |
| 42 | Tt0P | T0CKI Period | | TCY + 40*<br>N | — | — | ns | N = prescale value (1, 2, 4, ..., 256) |

* 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.


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 13.0 DEVICE CHARACTERIZATION INFORMATION

The graphs and tables provided in this section are for design guidance and are not tested. In some graphs or tables, the data presented is outside specified operating range (e.g., outside specified VDD range). This is for information only and devices will operate properly only within the specified range.

The data presented in this section is a statistical summary of data collected on units from different lots over a period of time. “Typical” represents the mean of the distribution, while “max” or “min” represents (mean + 3σ) and (mean – 3σ) respectively, where σ is standard deviation.

**FIGURE 13-1: IDD VS. FREQUENCY (XT MODE, VDD = 5.5V)**

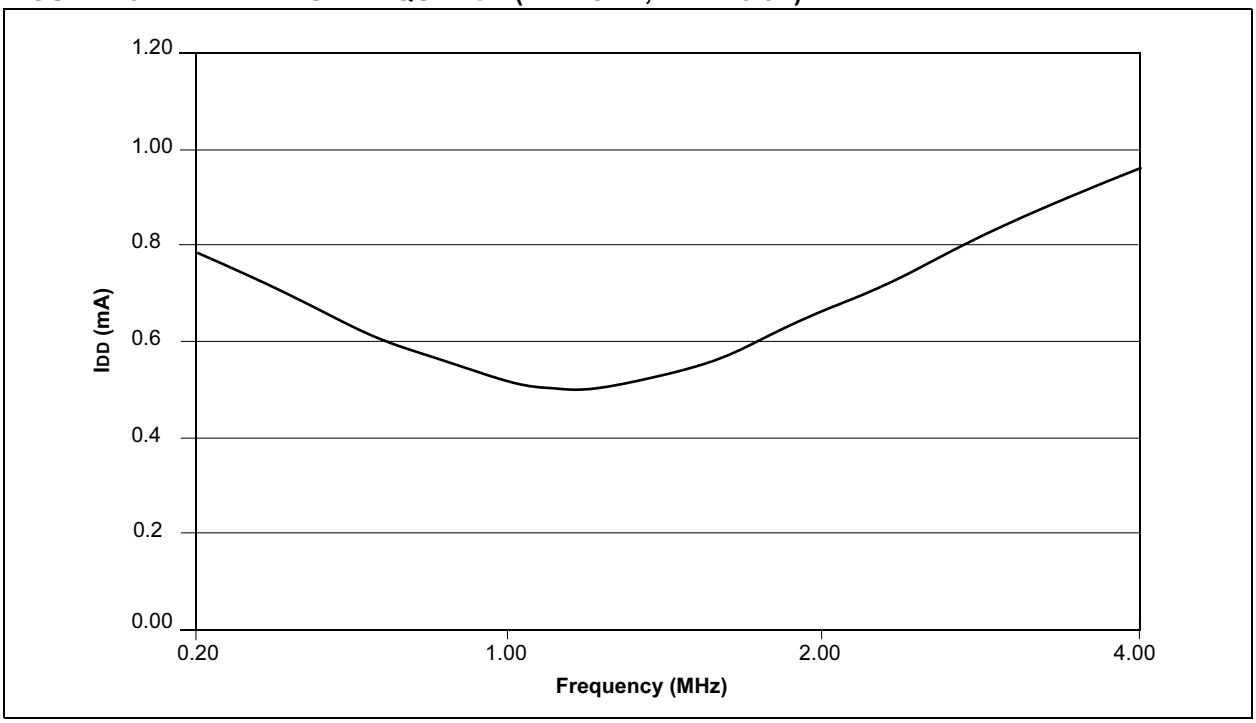
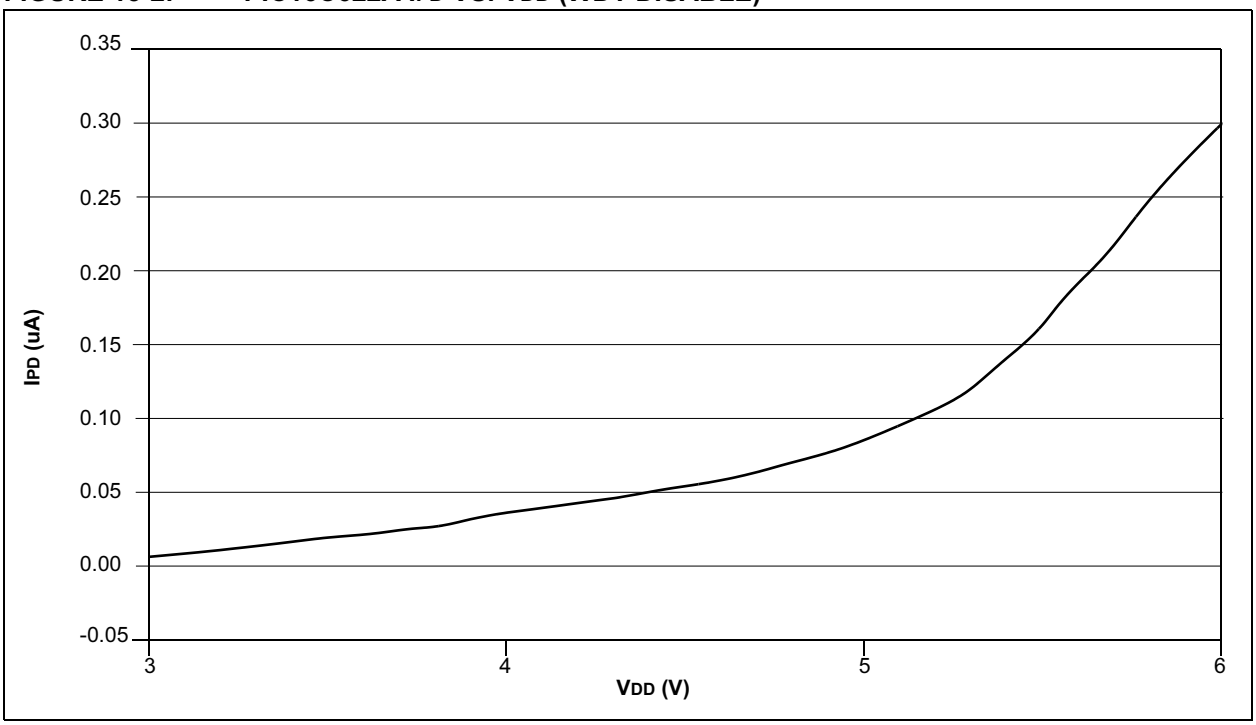



**FIGURE 13-2: PIC16C622A IPD VS. VDD (WDT DISABLE)**






Downloaded from Arrow.com.

**FIGURE 13-3: IDD VS. VDD (XT OSC 4 MHZ)**

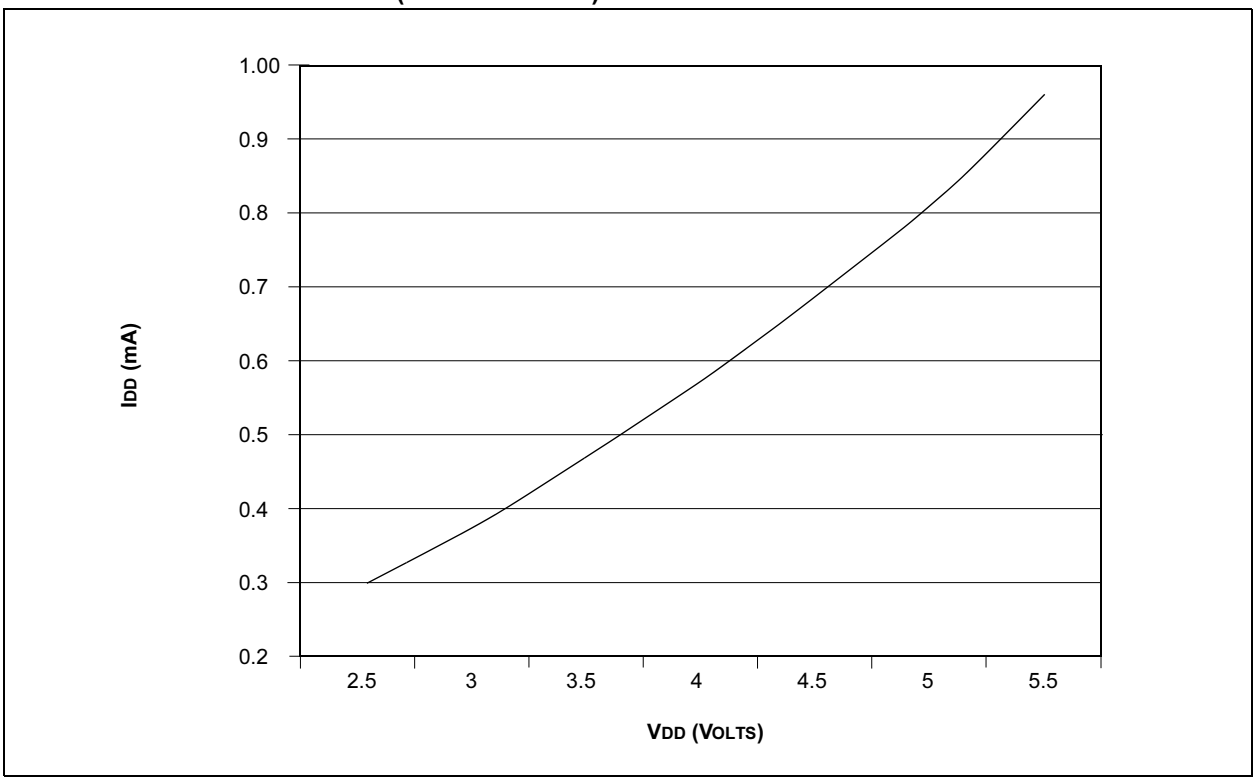
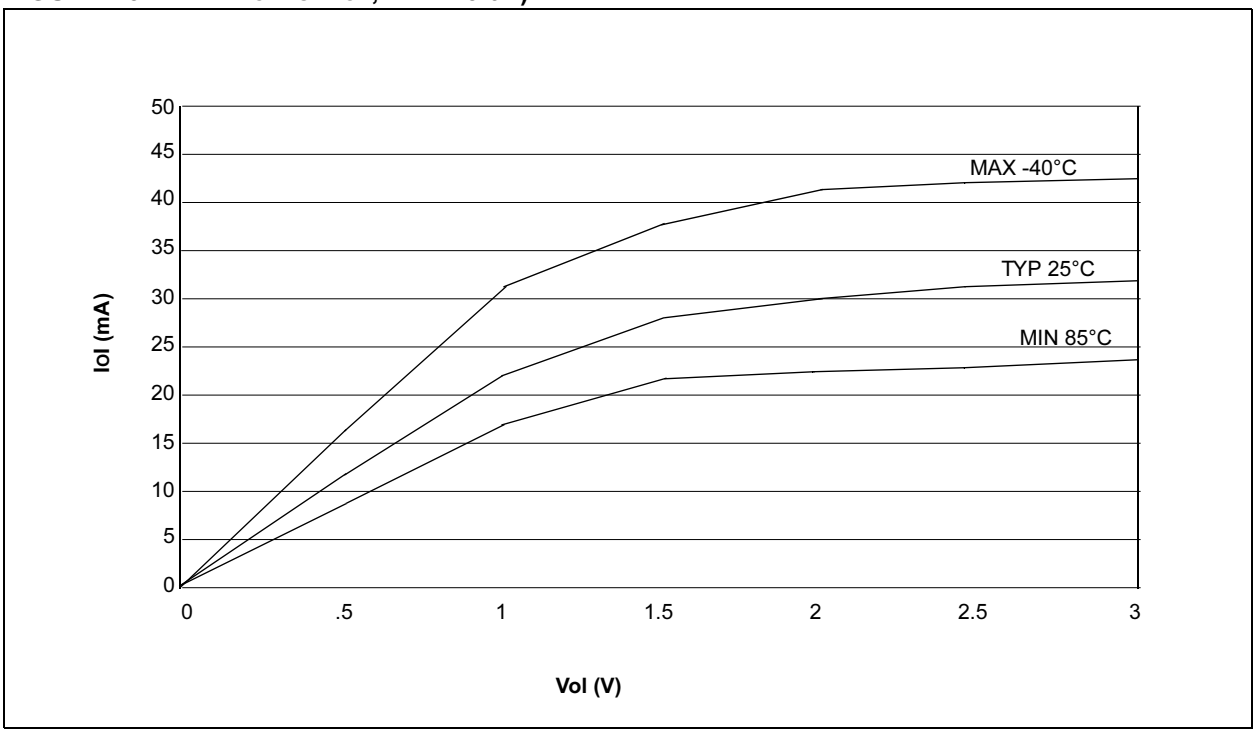



**FIGURE 13-4: Iol vs. VOL, VDD = 3.0V)**





© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

**FIGURE 13-5: IOH VS. VOH, VDD = 3.0V)**

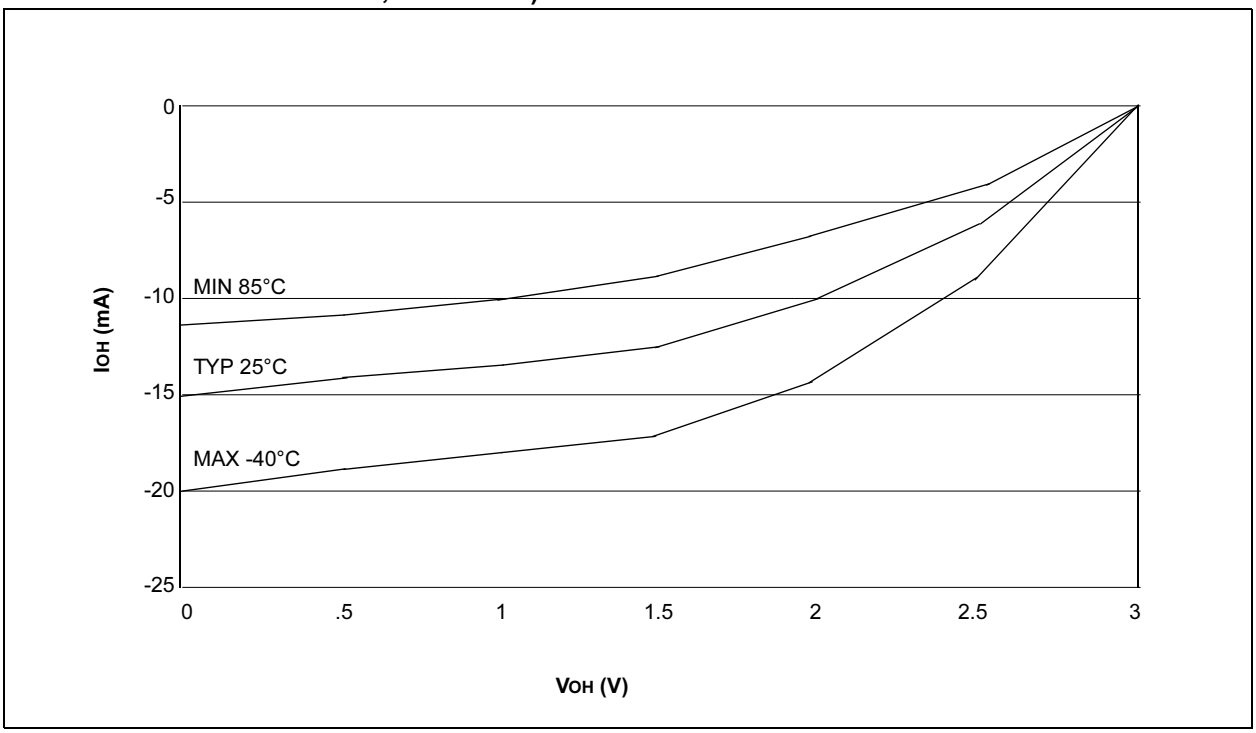
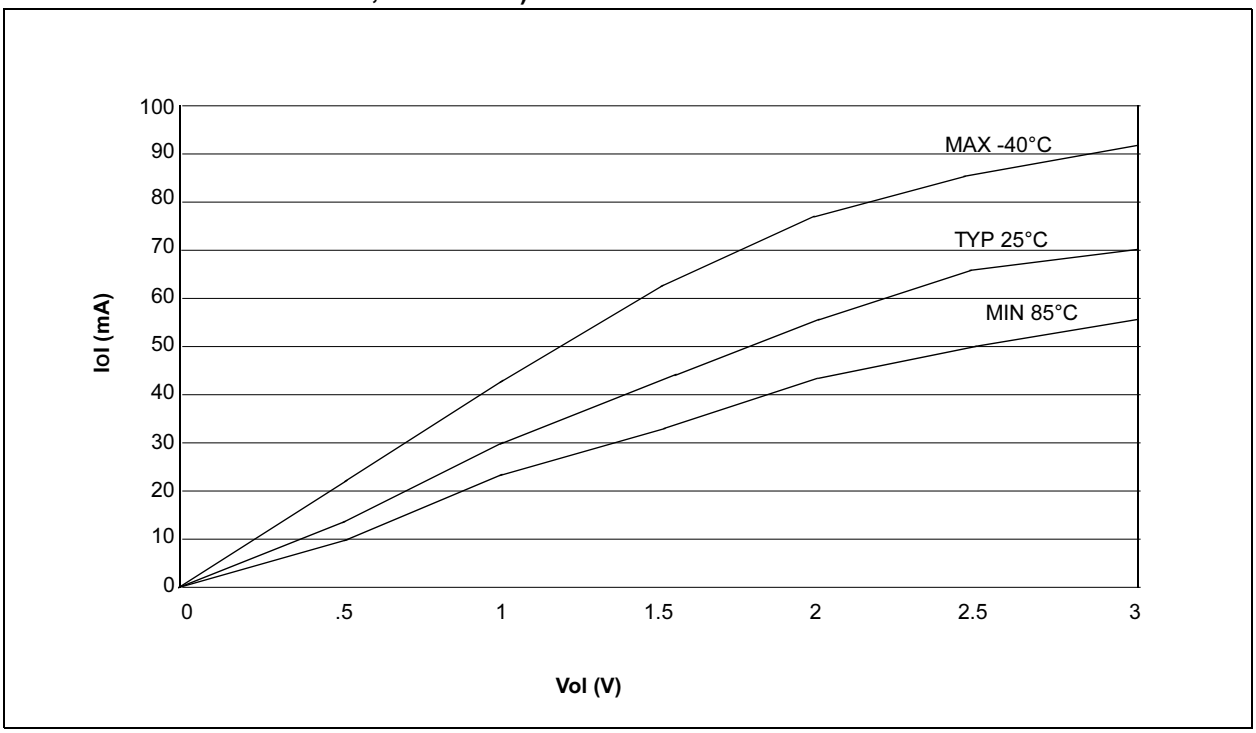


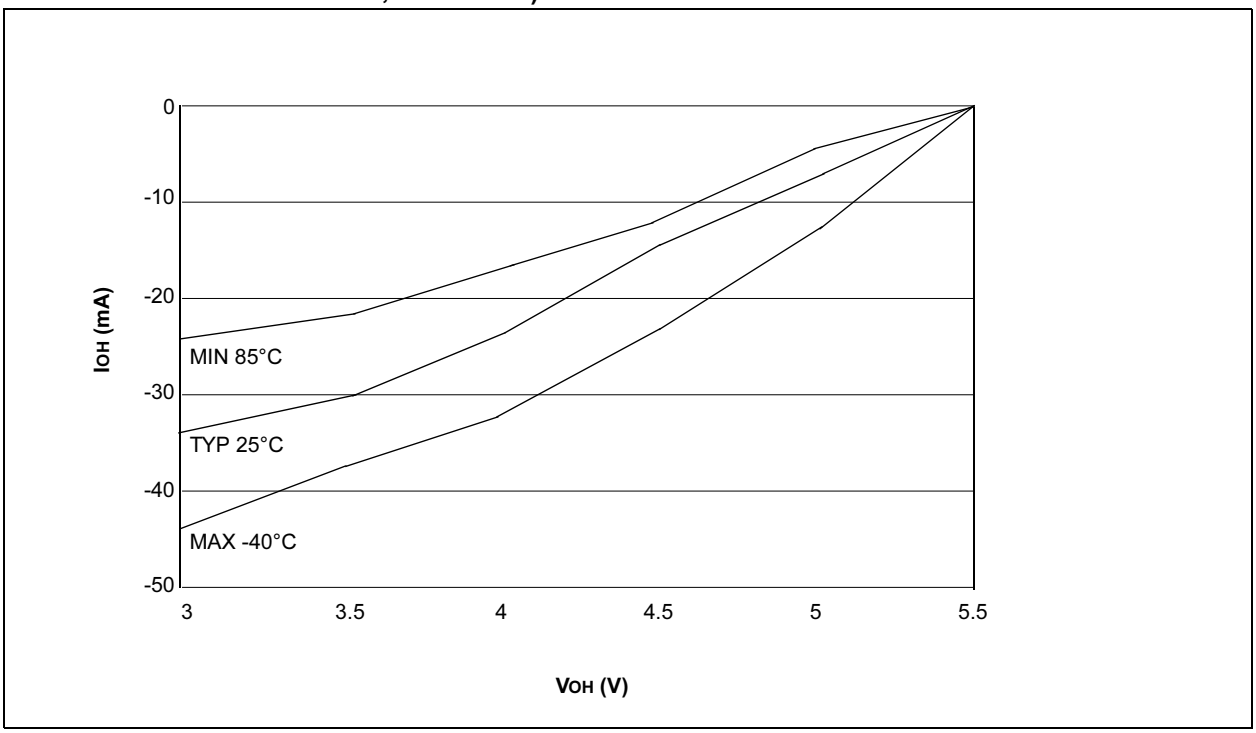

**FIGURE 13-6: IOl VS. VOL, VDD = 5.5V)**






Downloaded from Arrow.com.

**FIGURE 13-7: IOH VS. VOH, VDD = 5.5V)**



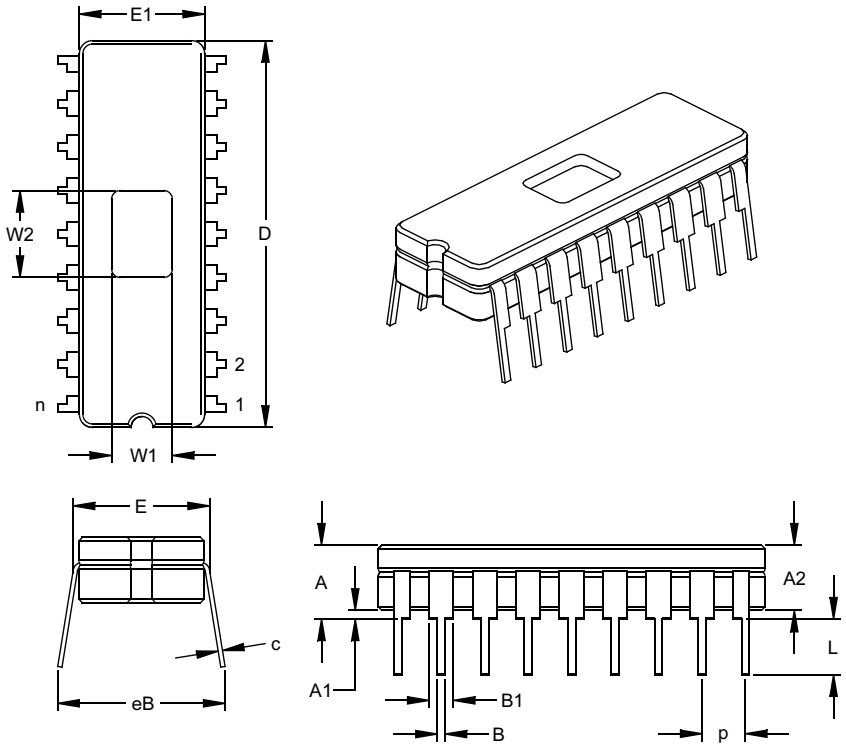


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## 14.0 PACKAGING INFORMATION

### 18-Lead Ceramic Dual In-line with Window (JW) – 300 mil (CERDIP)



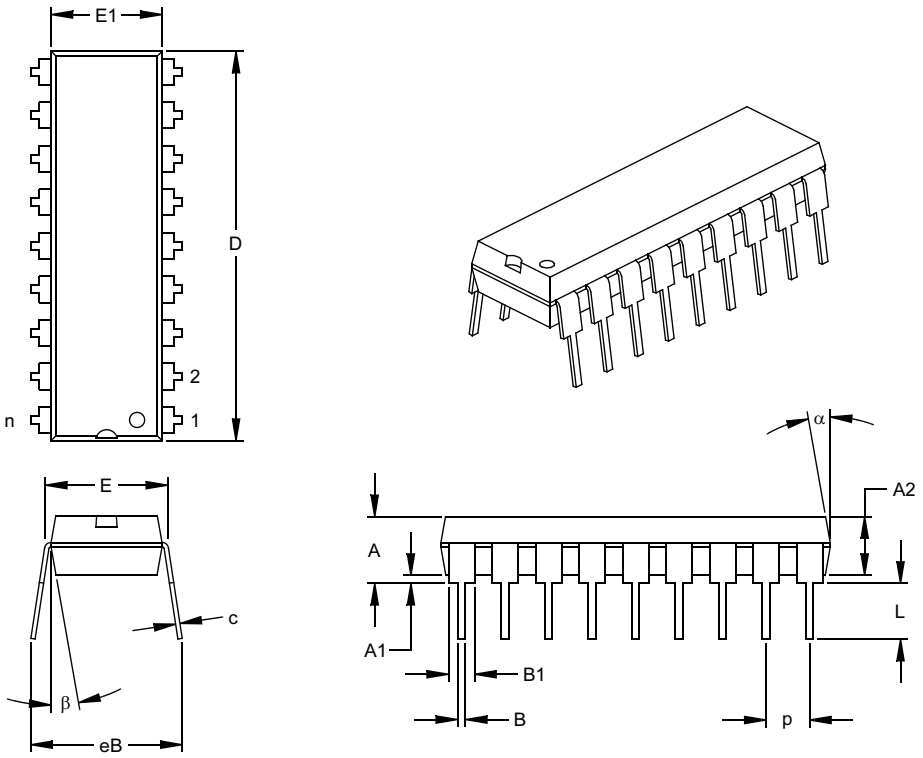

| Units | | INCHES* | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 18 | | | 18 | |
| Pitch | p | | .100 | | | 2.54 | |
| Top to Seating Plane | A | .170 | .183 | .195 | 4.32 | 4.64 | 4.95 |
| Ceramic Package Height | A2 | .155 | .160 | .165 | 3.94 | 4.06 | 4.19 |
| Standoff | A1 | .015 | .023 | .030 | 0.38 | 0.57 | 0.76 |
| Shoulder to Shoulder Width | E | .300 | .313 | .325 | 7.62 | 7.94 | 8.26 |
| Ceramic Pkg. Width | E1 | .285 | .290 | .295 | 7.24 | 7.37 | 7.49 |
| Overall Length | D | .880 | .900 | .920 | 22.35 | 22.86 | 23.37 |
| Tip to Seating Plane | L | .125 | .138 | .150 | 3.18 | 3.49 | 3.81 |
| Lead Thickness | c | .008 | .010 | .012 | 0.20 | 0.25 | 0.30 |
| Upper Lead Width | B1 | .050 | .055 | .060 | 1.27 | 1.40 | 1.52 |
| Lower Lead Width | B | .016 | .019 | .021 | 0.41 | 0.47 | 0.53 |
| Overall Row Spacing § | eB | .345 | .385 | .425 | 8.76 | 9.78 | 10.80 |
| Window Width | W1 | .130 | .140 | .150 | 3.30 | 3.56 | 3.81 |
| Window Length | W2 | .190 | .200 | .210 | 4.83 | 5.08 | 5.33 |

* Controlling Parameter
§ Significant Characteristic
JEDEC Equivalent: MO-036
Drawing No. C04-010



Downloaded from Arrow.com.

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



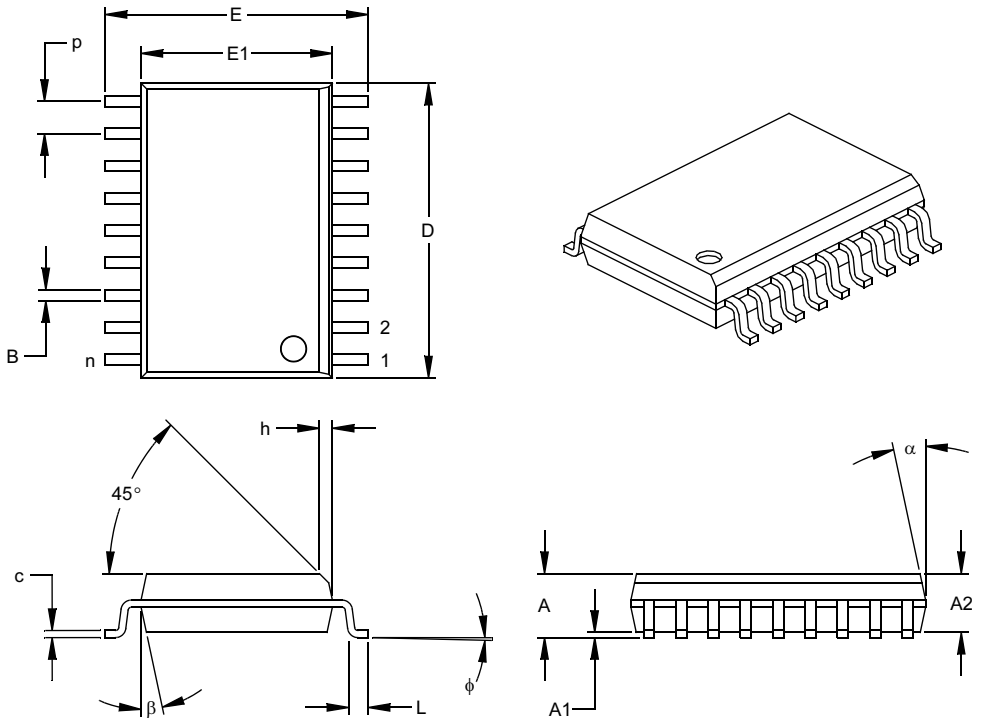

| Units | | INCHES* | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 18 | | | 18 | |
| Pitch | p | | .100 | | | 2.54 | |
| Top to Seating Plane | A | .140 | .155 | .170 | 3.56 | 3.94 | 4.32 |
| Molded Package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

* Controlling Parameter
§ Significant Characteristic
Notes:
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010” (0.254mm) per side.
JEDEC Equivalent: MS-001
Drawing No. C04-007


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

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




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

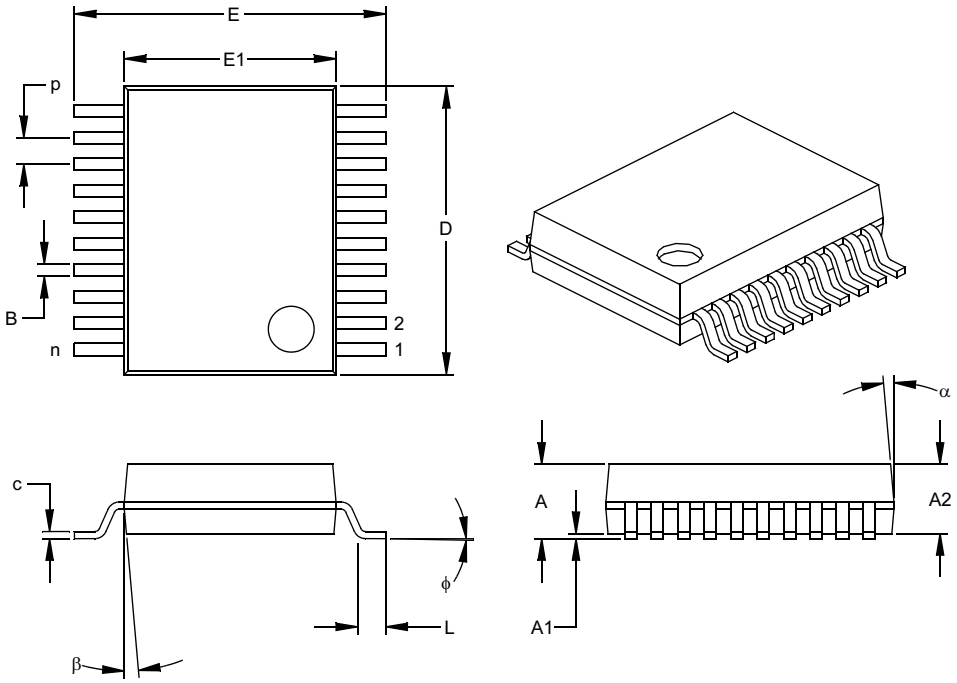
* Controlling Parameter
§ Significant Characteristic

Notes:
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010” (0.254mm) per side.
JEDEC Equivalent: MS-013
Drawing No. C04-051



Downloaded from Arrow.com.

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




| Units | | INCHES* | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 20 | | | 20 | |
| Pitch | p | | .026 | | | 0.65 | |
| Overall Height | A | .068 | .073 | .078 | 1.73 | 1.85 | 1.98 |
| Molded Package Thickness | A2 | .064 | .068 | .072 | 1.63 | 1.73 | 1.83 |
| Standoff § | A1 | .002 | .006 | .010 | 0.05 | 0.15 | 0.25 |
| Overall Width | E | .299 | .309 | .322 | 7.59 | 7.85 | 8.18 |
| Molded Package Width | E1 | .201 | .207 | .212 | 5.11 | 5.25 | 5.38 |
| Overall Length | D | .278 | .284 | .289 | 7.06 | 7.20 | 7.34 |
| Foot Length | L | .022 | .030 | .037 | 0.56 | 0.75 | 0.94 |
| Lead Thickness | c | .004 | .007 | .010 | 0.10 | 0.18 | 0.25 |
| Foot Angle | ϕ | 0 | 4 | 8 | 0.00 | 101.60 | 203.20 |
| Lead Width | B | .010 | .013 | .015 | 0.25 | 0.32 | 0.38 |
| Mold Draft Angle Top | α | 0 | 5 | 10 | 0 | 5 | 10 |
| Mold Draft Angle Bottom | β | 0 | 5 | 10 | 0 | 5 | 10 |

* Controlling Parameter
§ Significant Characteristic

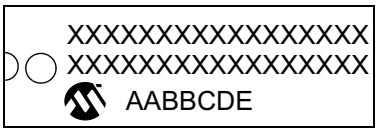
Notes:
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010” (0.254mm) per side.
JEDEC Equivalent: MO-150
Drawing No. C04-072


© 2003 Microchip Technology Inc.

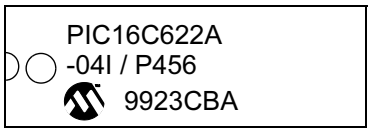
Downloaded from Arrow.com.

## 14.1 Package Marking Information

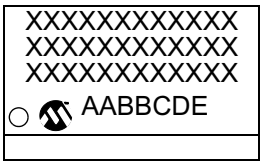
**18-Lead PDIP**



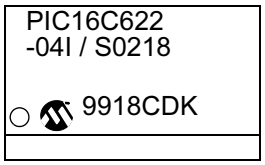

**Example**



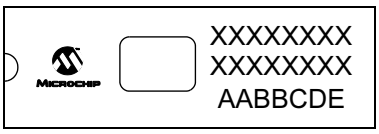

**18-Lead SOIC (.300")**



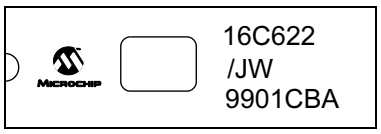

**Example**



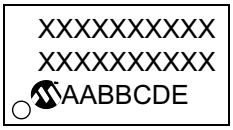

**18-Lead CERDIP Windowed**



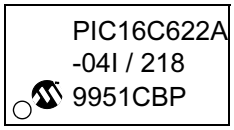

**Example**




**20-Lead SSOP**




**Example**




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

**Note**: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line thus limiting the number of available characters for customer specific information.

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



Downloaded from Arrow.com.

**NOTES:**


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## APPENDIX A: ENHANCEMENTS

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

1. Instruction word length is increased to 14 bits. This allows larger page sizes both in program memory (4K now as opposed to 512 before) and register file (up to 128 bytes now versus 32 bytes before).
2. A PC high latch register (PCLATH) is added to handle program memory paging. PA2, PA1, PA0 bits are removed from STATUS register.
3. Data memory paging is slightly redefined. STATUS register is modified.
4. Four new instructions have been added: RETURN, RETFIE, ADDLW, and SUBLW. Two instructions TRIS and OPTION are being phased out, although they are kept for compatibility with PIC16C5X.
5. OPTION and TRIS registers are made addressable.
6. Interrupt capability is added. Interrupt vector is at 0004h.
7. Stack size is increased to 8 deep.
8. RESET vector is changed to 0000h.
9. RESET of all registers is revisited. Five different RESET (and wake-up) types are recognized. Registers are reset differently.
10. Wake-up from SLEEP through interrupt is added.
11. Two separate timers, Oscillator Start-up Timer (OST) and Power-up Timer (PWRT) are included for more reliable power-up. These timers are invoked selectively to avoid unnecessary delays on power-up and wake-up.
12. PORTB has weak pull-ups and interrupt-on-change feature.
13. Timer0 clock input, T0CKI pin is also a port pin (RA4/T0CKI) and has a TRIS bit.
14. FSR is made a full 8-bit register.
15. “In-circuit programming” is made possible. The user can program PIC16CXX devices using only five pins: VDD, VSS, VPP, RB6 (clock) and RB7 (data in/out).
16. PCON STATUS register is added with a Power-on-Reset ($\overline{\text{POR}}$) STATUS bit and a Brown-out Reset STATUS bit ($\overline{\text{BOD}}$).
17. Code protection scheme is enhanced such that portions of the program memory can be protected, while the remainder is unprotected.
18. PORTA inputs are now Schmitt Trigger inputs.
19. Brown-out Reset reset has been added.
20. Common RAM registers F0h-FFh implemented in bank1.

## APPENDIX B: COMPATIBILITY

To convert code written for PIC16C5X to PIC16CXX, the user should take the following steps:

1. Remove any program memory page select operations (PA2, PA1, PA0 bits) for CALL, GOTO.
2. Revisit any computed jump operations (write to PC or add to PC, etc.) to make sure page bits are set properly under the new scheme.
3. Eliminate any data memory page switching. Redefine data variables to reallocate them.
4. Verify all writes to STATUS, OPTION, and FSR registers since these have changed.
5. Change RESET vector to 0000h.



Downloaded from Arrow.com.

**NOTES:**


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## INDEX


### A

ADDLW Instruction ..... 63
ADDWF Instruction ..... 63
ANDLW Instruction ..... 63
ANDWF Instruction ..... 63
Architectural Overview ..... 9
Assembler
    MPASM Assembler ..... 75

### B

BCF Instruction ..... 64
Block Diagram
    TIMER0 ..... 31
    TMR0/WDT PRESCALER ..... 34
Brown-Out Detect (BOD) ..... 50
BSF Instruction ..... 64
BTFSC Instruction ..... 64
BTFSS Instruction ..... 65

### C

C Compilers
    MPLAB C17 ..... 76
    MPLAB C18 ..... 76
    MPLAB C30 ..... 76
CALL Instruction ..... 65
Clocking Scheme/Instruction Cycle ..... 12
CLRF Instruction ..... 65
CLRW Instruction ..... 66
CLRWDT Instruction ..... 66
Code Protection ..... 60
COMF Instruction ..... 66
Comparator Configuration ..... 38
Comparator Interrupts ..... 41
Comparator Module ..... 37
Comparator Operation ..... 39
Comparator Reference ..... 39
Configuration Bits ..... 46
Configuring the Voltage Reference ..... 43
Crystal Operation ..... 47

### D

Data Memory Organization ..... 14
DC Characteristics ..... 87, 101
    PIC16C717/770/771 ..... 88, 89, 90, 91, 96, 97, 98
DECF Instruction ..... 66
DECFSZ Instruction ..... 67
Demonstration Boards
    PICDEM 1 ..... 78
    PICDEM 17 ..... 78
    PICDEM 18R PIC18C601/801 ..... 79
    PICDEM 2 Plus ..... 78
    PICDEM 3 PIC16C92X ..... 78
    PICDEM 4 ..... 78
    PICDEM LIN PIC16C43X ..... 79
    PICDEM USB PIC16C7X5 ..... 79
    PICDEM.net Internet/Ethernet ..... 78
Development Support ..... 75

### E

Errata ..... 3
Evaluation and Programming Tools ..... 79
External Crystal Oscillator Circuit ..... 48

### G

General purpose Register File ..... 14
GOTO Instruction ..... 67

### I

I/O Ports ..... 25
I/O Programming Considerations ..... 30
ID Locations ..... 60
INCF Instruction ..... 67
INCFSZ Instruction ..... 68
In-Circuit Serial Programming ..... 60
Indirect Addressing, INDF and FSR Registers ..... 24
Instruction Flow/Pipelining ..... 12
Instruction Set
    ADDLW ..... 63
    ADDWF ..... 63
    ANDLW ..... 63
    ANDWF ..... 63
    BCF ..... 64
    BSF ..... 64
    BTFSC ..... 64
    BTFSS ..... 65
    CALL ..... 65
    CLRF ..... 65
    CLRW ..... 66
    CLRWDT ..... 66
    COMF ..... 66
    DECF ..... 66
    DECFSZ ..... 67
    GOTO ..... 67
    INCF ..... 67
    INCFSZ ..... 68
    IORLW ..... 68
    IORWF ..... 68
    MOVF ..... 69
    MOVLW ..... 68
    MOVWF ..... 69
    NOP ..... 69
    OPTION ..... 69
    RETFIE ..... 70
    RETLW ..... 70
    RETURN ..... 70
    RLF ..... 71
    RRF ..... 71
    SLEEP ..... 71
    SUBLW ..... 72
    SUBWF ..... 72
    SWAPF ..... 73
    TRIS ..... 73
    XORLW ..... 73
    XORWF ..... 73
Instruction Set Summary ..... 61
INT Interrupt ..... 56
INTCON Register ..... 20
Interrupts ..... 55
IORLW Instruction ..... 68
IORWF Instruction ..... 68

### M

MOVF Instruction ..... 69
MOVLW Instruction ..... 68
MOVWF Instruction ..... 69
MPLAB ASM30 Assembler, Linker, Librarian ..... 76
MPLAB ICD 2 In-Circuit Debugger ..... 77
MPLAB ICE 2000 High Performance Universal In-Circuit Emulator ..... 77
MPLAB ICE 4000 High Performance Universal In-Circuit Emulator ..... 77
MPLAB Integrated Development Environment Software ..... 75
MPLINK Object Linker/MPLIB Object Librarian ..... 76




Downloaded from Arrow.com.

**N**

NOP Instruction ..... 69

**O**

One-Time-Programmable (OTP) Devices ..... 7
OPTION Instruction ..... 69
OPTION Register ..... 19
Oscillator Configurations ..... 47
Oscillator Start-up Timer (OST) ..... 50

**P**

Package Marking Information ..... 117
Packaging Information ..... 113
PCL and PCLATH ..... 23
PCON Register ..... 22
PICkit 1 FLASH Starter Kit ..... 79
PICSTART Plus Development Programmer ..... 77
PIE1 Register ..... 21
PIR1 Register ..... 21
Port RB Interrupt ..... 56
PORTA ..... 25
PORTB ..... 28
Power Control/Status Register (PCON) ..... 51
Power-Down Mode (SLEEP) ..... 59
Power-On Reset (POR) ..... 50
Power-up Timer (PWRT) ..... 50
Prescaler ..... 34
PRO MATE II Universal Device Programmer ..... 77
Program Memory Organization ..... 13

**Q**

Quick-Turnaround-Production (QTP) Devices ..... 7

**R**

RC Oscillator ..... 48
Reset ..... 49
RETFIE Instruction ..... 70
RETLW Instruction ..... 70
RETURN Instruction ..... 70
RLF Instruction ..... 71
RRF Instruction ..... 71

**S**

Serialized Quick-Turnaround-Production (SQTP) Devices ... 7
SLEEP Instruction ..... 71
Software Simulator (MPLAB SIM) ..... 76
Software Simulator (MPLAB SIM30) ..... 76
Special Features of the CPU ..... 45
Special Function Registers ..... 17
Stack ..... 23
Status Register ..... 18
SUBLW Instruction ..... 72
SUBWF Instruction ..... 72
SWAPF Instruction ..... 73

**T**

Timer0
- TIMER0 ..... 31
- TIMER0 (TMR0) Interrupt ..... 31
- TIMER0 (TMR0) Module ..... 31
- TMR0 with External Clock ..... 33

Timer1
- Switching Prescaler Assignment ..... 35

Timing Diagrams and Specifications ..... 104
TMR0 Interrupt ..... 56
TRIS Instruction ..... 73
TRISA ..... 25
TRISB ..... 28

**V**

Voltage Reference Module ..... 43
VRCON Register ..... 43

**W**

Watchdog Timer (WDT) ..... 58
WWW, On-Line Support ..... 3

**X**

XORLW Instruction ..... 73
XORWF Instruction ..... 73



© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## ON-LINE SUPPORT

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

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

### Connecting to the Microchip Internet Web Site

The Microchip web site is available at the following URL:

**www.microchip.com**

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

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

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

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

## SYSTEMS INFORMATION AND UPGRADE HOT LINE

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

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

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

092002



Downloaded from Arrow.com.

## READER RESPONSE

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

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

To: Technical Publications Manager Total Pages Sent _____

RE: Reader Response

From: Name _____

Company _____

Address _____

City / State / ZIP / Country _____

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

Application (optional):

Would you like a reply? ___Y ___N

Device: PIC16C62X Literature Number: DS30235J

Questions:

1. What are the best features of this document?

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

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

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

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

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

7. How would you improve this document?


© 2003 Microchip Technology Inc.

Downloaded from Arrow.com.

## PRODUCT IDENTIFICATION SYSTEM

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

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

| | |
|---|---|
| Device | PIC16C62X: VDD range 3.0V to 6.0V<br>PIC16C62XT: VDD range 3.0V to 6.0V (Tape and Reel)<br>PIC16C62XA: VDD range 3.0V to 5.5V<br>PIC16C62XAT: VDD range 3.0V to 5.5V (Tape and Reel)<br>PIC16LC62X: VDD range 2.5V to 6.0V<br>PIC16LC62XT: VDD range 2.5V to 6.0V (Tape and Reel)<br>PIC16LC62XA: VDD range 2.5V to 5.5V<br>PIC16LC62XAT: VDD range 2.5V to 5.5V (Tape and Reel)<br>PIC16CR620A: VDD range 2.5V to 5.5V<br>PIC16CR620AT: VDD range 2.5V to 5.5V (Tape and Reel)<br>PIC16LCR620A: VDD range 2.0V to 5.5V<br>PIC16LCR620AT: VDD range 2.0V to 5.5V (Tape and Reel) |
| Frequency Range | 04 200 kHz (LP osc)<br>04 4 MHz (XT and RC osc)<br>20 20 MHz (HS osc) |
| Temperature Range | - = 0°C to +70°C<br>I = -40°C to +85°C<br>E = -40°C to +125°C |
| Package | P = PDIP<br>SO = SOIC (Gull Wing, 300 mil body)<br>SS = SSOP (209 mil)<br>JW* = Windowed CERDIP |
| Pattern | 3-Digit Pattern Code for QTP (blank otherwise) |

**Examples:**

a) PIC16C621A - 04/P 301 = Commercial temp., PDIP package, 4 MHz, normal VDD limits, QTP pattern #301.

b) PIC16LC622- 04I/SO = Industrial temp., SOIC package, 200 kHz, extended VDD limits.

* JW Devices are UV erasable and can be programmed to any device configuration. JW Devices meet the electrical requirement of each oscillator type.

### 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.



Downloaded from Arrow.com.

# WORLDWIDE SALES AND SERVICE

## AMERICAS

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

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

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

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

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

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

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

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

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

**San Jose**
Microchip Technology Inc.
2107 North First Street, Suite 590
San Jose, CA 95131
Tel: 408-436-7950 Fax: 408-436-7955

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

## ASIA/PACIFIC

**Australia**
Microchip Technology Australia Pty Ltd
Marketing Support Division
Suite 22, 41 Rawson Street
Epping 2121, NSW
Australia
Tel: 61-2-9868-6733 Fax: 61-2-9868-6755

**China - Beijing**
Microchip Technology Consulting (Shanghai) Co., Ltd., Beijing Liaison Office
Unit 915
Bei Hai Wan Tai Bldg.
No. 6 Chaoyangmen Beidajie
Beijing, 100027, No. China
Tel: 86-10-85282100 Fax: 86-10-85282104

**China - Chengdu**
Microchip Technology Consulting (Shanghai) Co., Ltd., Chengdu Liaison Office
Rm. 2401-2402, 24th Floor,
Ming Xing Financial Tower
No. 88 TIDU Street
Chengdu 610016, China
Tel: 86-28-86766200 Fax: 86-28-86766599

**China - Fuzhou**
Microchip Technology Consulting (Shanghai) Co., Ltd., Fuzhou Liaison Office
Unit 28F, World Trade Plaza
No. 71 Wusi Road
Fuzhou 350001, China
Tel: 86-591-7503506 Fax: 86-591-7503521

**China - Hong Kong SAR**
Microchip Technology Hongkong Ltd.
Unit 901-6, Tower 2, Metroplaza
223 Hing Fong Road
Kwai Fong, N.T., Hong Kong
Tel: 852-2401-1200 Fax: 852-2401-3431

**China - Shanghai**
Microchip Technology Consulting (Shanghai) Co., Ltd.
Room 701, Bldg. B
Far East International Plaza
No. 317 Xian Xia Road
Shanghai, 200051
Tel: 86-21-6275-5700 Fax: 86-21-6275-5060

**China - Shenzhen**
Microchip Technology Consulting (Shanghai) Co., Ltd., Shenzhen Liaison Office
Rm. 1812, 18/F, Building A, United Plaza
No. 5022 Binhe Road, Futian District
Shenzhen 518033, China
Tel: 86-755-82901380 Fax: 86-755-82966626

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

**India**
Microchip Technology Inc.
India Liaison Office
Marketing Support Division
Divyasree Chambers
1 Floor, Wing A (A3/A4)
No. 11, O'Shaugnessey Road
Bangalore, 560 025, India
Tel: 91-80-2290061 Fax: 91-80-2290062

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

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

**Singapore**
Microchip Technology Singapore Pte Ltd.
200 Middle Road
#07-02 Prime Centre
Singapore, 188980
Tel: 65-6334-8870 Fax: 65-6334-8850

**Taiwan**
Microchip Technology (Barbados) Inc., Taiwan Branch
11F-3, No. 207
Tung Hua North Road
Taipei, 105, Taiwan
Tel: 886-2-2717-7175 Fax: 886-2-2545-0139

## EUROPE

**Austria**
Microchip Technology Austria GmbH
Durisolstrasse 2
A-4600 Wels
Austria
Tel: 43-7242-2244-399
Fax: 43-7242-2244-393

**Denmark**
Microchip Technology Nordic ApS
Regus Business Centre
Lautrup hoj 1-3
Ballerup DK-2750 Denmark
Tel: 45 4420 9895 Fax: 45 4420 9910

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

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

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

**United Kingdom**
Microchip Ltd.
505 Eskdale Road
Winnersh Triangle
Wokingham
Berkshire, England RG41 5TU
Tel: 44 118 921 5869 Fax: 44-118 921-5820

03/25/03


© 2003 Microchip Technology Inc.
Downloaded from Arrow.com.