# PIC16C62B/72A

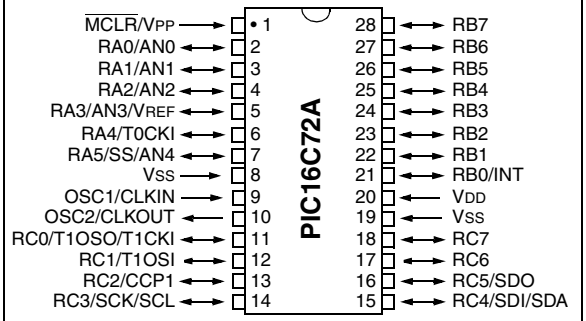
## 28-Pin 8-Bit CMOS Microcontrollers

### Microcontroller Core Features:

- High-performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches, which are two cycle
- Operating speed: DC - 20 MHz clock input
  DC - 200 ns instruction cycle
- 2K x 14 words of Program Memory, 128 x 8 bytes of Data Memory (RAM)
- Interrupt capability
- Eight level deep hardware stack
- Direct, indirect, and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Brown-out detection circuitry for Brown-out Reset (BOR)
- Programmable code-protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low-power, high-speed CMOS EPROM technology
- Fully static design
- In-Circuit Serial Programming™ (ICSP)
- Wide operating voltage range: 2.5V to 5.5V
- High Sink/Source Current 25/25 mA
- Commercial, Industrial and Extended temperature ranges
- Low-power consumption:
  - < 2 mA @ 5V, 4 MHz
  - 22.5 μA typical @ 3V, 32 kHz
  - < 1 μA typical standby current

### Pin Diagram

**SDIP, SOIC, SSOP, Windowed CERDIP**



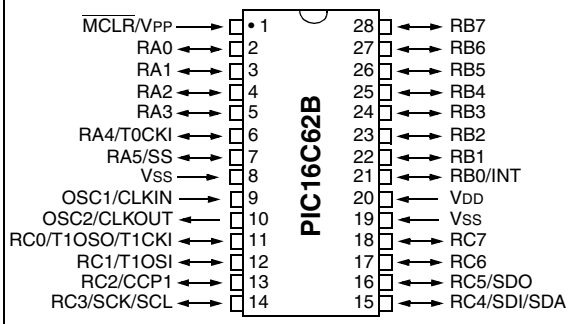

### Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler, can be incremented during sleep via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Capture, Compare, PWM module
- Capture is 16-bit, max. resolution is 12.5 ns, Compare is 16-bit, max. resolution is 200 ns, PWM maximum resolution is 10-bit
- 8-bit multi-channel Analog-to-Digital converter
- Synchronous Serial Port (SSP) with Enhanced SPI and $I^2C$™

© 1998-2013 Microchip Technology Inc.

**Pin Diagrams**

**SDIP, SOIC, SSOP, Windowed CERDIP**




| Key Features<br>PIC® Mid-Range Reference Manual (DS33023) | PIC16C62B | PIC16C72A |
|---|---|---|
| Operating Frequency | DC - 20 MHz | DC - 20 MHz |
| Resets (and Delays) | POR, BOR (PWRT, OST) | POR, BOR (PWRT, OST) |
| Program Memory (14-bit words) | 2K | 2K |
| Data Memory (bytes) | 128 | 128 |
| Interrupts | 7 | 8 |
| I/O Ports | Ports A,B,C | Ports A,B,C |
| Timers | 3 | 3 |
| Capture/Compare/PWM modules | 1 | 1 |
| Serial Communications | SSP | SSP |
| 8-bit Analog-to-Digital Module | — | 5 input channels |

 © 1998-2013 Microchip Technology Inc.

## Table of Contents

1.0 Device Overview ..... 5
2.0 Memory Organization ..... 7
3.0 I/O Ports ..... 19
4.0 Timer0 Module ..... 25
5.0 Timer1 Module ..... 27
6.0 Timer2 Module ..... 31
7.0 Capture/Compare/PWM (CCP) Module ..... 33
8.0 Synchronous Serial Port (SSP) Module ..... 39
9.0 Analog-to-Digital Converter (A/D) Module ..... 49
10.0 Special Features of the CPU ..... 55
11.0 Instruction Set Summary ..... 67
12.0 Development Support ..... 75
13.0 Electrical Characteristics ..... 81
14.0 DC and AC Characteristics Graphs and Tables ..... 103
15.0 Packaging Information ..... 105
Appendix A: Revision History ..... 111
Appendix B: Conversion Considerations ..... 111
Appendix C: Migration from Base-line to Mid-Range Devices ..... 112
Index ..... 113
On-Line Support ..... 117
Reader Response ..... 118
PIC16C62B/72A Product Identification System ..... 119

*To Our Valued Customers*

**Most Current Data Sheet**

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

http://www.microchip.com

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

**New Customer Notification System**

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

**Errata**

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

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

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

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

**Corrections to this Data Sheet**

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

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

We appreciate your assistance in making this a better document.

**NOTES:**

 © 1998-2013 Microchip Technology Inc.

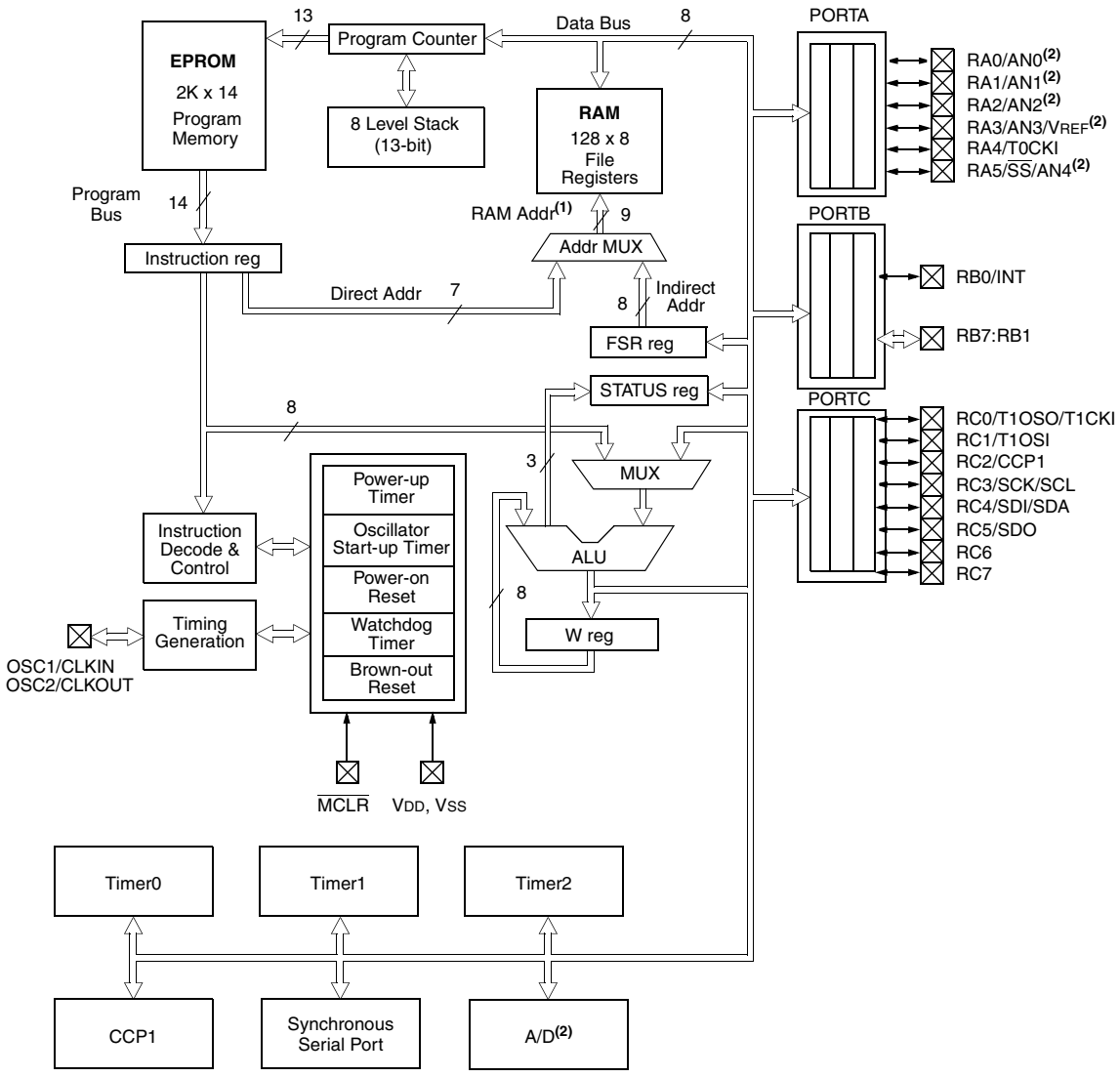
## 1.0 DEVICE OVERVIEW

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

There are two devices (PIC16C62B, PIC16C72A) covered by this datasheet. The PIC16C62B does not have the A/D module implemented.

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

**FIGURE 1-1: PIC16C62B/PIC16C72A BLOCK DIAGRAM**




**Note 1:** Higher order bits are from the STATUS register.
**2:** The A/D module is not available on the PIC16C62B.

**TABLE 1-1 PIC16C62B/PIC16C72A PINOUT DESCRIPTION**

| Pin Name | DIP Pin# | SOIC Pin# | I/O/P Type | Buffer Type | Description |
|---|---|---|---|---|---|
| OSC1/CLKIN | 9 | 9 | I | ST/CMOS(3) | Oscillator crystal input/external clock source input. |
| OSC2/CLKOUT | 10 | 10 | O | — | Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, the OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate. |
| $\overline{MCLR}$/VPP | 1 | 1 | I/P | ST | Master clear (reset) input or programming voltage input. This pin is an active low reset to the device. |
| | | | | | PORTA is a bi-directional I/O port. |
| RA0/AN0(4) | 2 | 2 | I/O | TTL | RA0 can also be analog input 0 |
| RA1/AN1(4) | 3 | 3 | I/O | TTL | RA1 can also be analog input 1 |
| RA2/AN2(4) | 4 | 4 | I/O | TTL | RA2 can also be analog input 2 |
| RA3/AN3/VREF(4) | 5 | 5 | I/O | TTL | RA3 can also be analog input 3 or analog reference voltage |
| RA4/T0CKI | 6 | 6 | I/O | ST | RA4 can also be the clock input to the Timer0 module. Output is open drain type. |
| RA5/$\overline{SS}$/AN4(4) | 7 | 7 | I/O | TTL | RA5 can also be analog input 4 or the slave select for the synchronous serial port. |
| | | | | | PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs. |
| RB0/INT | 21 | 21 | I/O | TTL/ST(1) | RB0 can also be the external interrupt pin. |
| RB1 | 22 | 22 | I/O | TTL | |
| RB2 | 23 | 23 | I/O | TTL | |
| RB3 | 24 | 24 | I/O | TTL | |
| RB4 | 25 | 25 | I/O | TTL | Interrupt on change pin. |
| RB5 | 26 | 26 | I/O | TTL | Interrupt on change pin. |
| RB6 | 27 | 27 | I/O | TTL/ST(2) | Interrupt on change pin. Serial programming clock. |
| RB7 | 28 | 28 | I/O | TTL/ST(2) | Interrupt on change pin. Serial programming data. |
| | | | | | PORTC is a bi-directional I/O port. |
| RC0/T1OSO/T1CKI | 11 | 11 | I/O | ST | RC0 can also be the Timer1 oscillator output or Timer1 clock input. |
| RC1/T1OSI | 12 | 12 | I/O | ST | RC1 can also be the Timer1 oscillator input. |
| RC2/CCP1 | 13 | 13 | I/O | ST | RC2 can also be the Capture1 input/Compare1 output/ PWM1 output. |
| RC3/SCK/SCL | 14 | 14 | I/O | ST | RC3 can also be the synchronous serial clock input/output for both SPI and $I^2C$ modes. |
| RC4/SDI/SDA | 15 | 15 | I/O | ST | RC4 can also be the SPI Data In (SPI mode) or data I/O ($I^2C$ mode). |
| RC5/SDO | 16 | 16 | I/O | ST | RC5 can also be the SPI Data Out (SPI mode). |
| RC6 | 17 | 17 | I/O | ST | |
| RC7 | 18 | 18 | I/O | ST | |
| VSS | 8, 19 | 8, 19 | P | — | Ground reference for logic and I/O pins. |
| VDD | 20 | 20 | P | — | Positive supply for logic and I/O pins. |

Legend: I = input O = output I/O = input/output P = power or program
— = Not used TTL = TTL input ST = Schmitt Trigger 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.
**3:** This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.
**4:** The A/D module is not available on the PIC16C62B.

 © 1998-2013 Microchip Technology Inc.

## 2.0 MEMORY ORGANIZATION

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

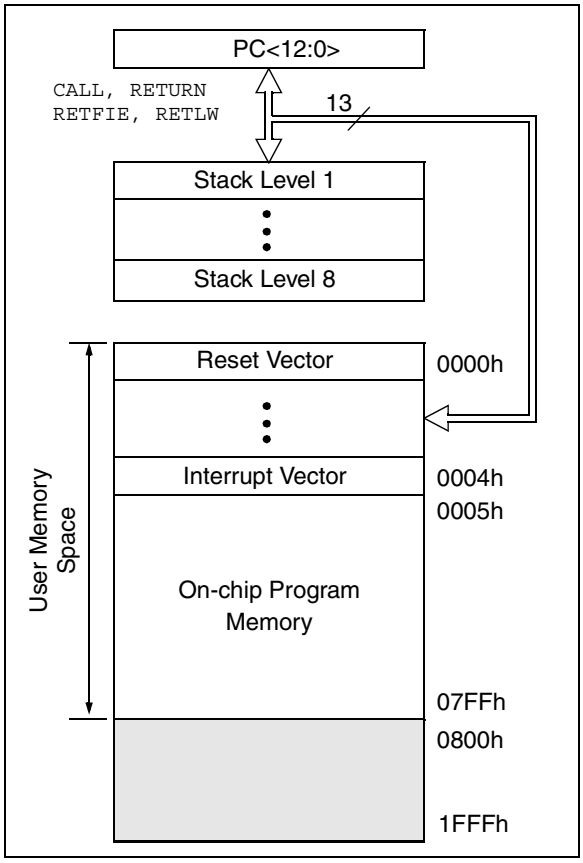
Additional information on device memory may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023).

### 2.1 Program Memory Organization

The PIC16C62B/72A devices have a 13-bit program counter capable of addressing an 8K x 14 program memory space. Each device has 2K x 14 words of program memory. Accessing a location above 07FFh will cause a wraparound.

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

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



## 2.2 Data Memory Organization

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

| RP1(1) | RP0 |
|---|---|

(STATUS<6:5>)

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

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

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

### 2.2.1 GENERAL PURPOSE REGISTER FILE

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

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

| File Address | Bank 0 | Bank 1 | File Address |
|---|---|---|---|
| 00h | INDF(1) | INDF(1) | 80h |
| 01h | TMR0 | OPTION_REG | 81h |
| 02h | PCL | PCL | 82h |
| 03h | STATUS | STATUS | 83h |
| 04h | FSR | FSR | 84h |
| 05h | PORTA | TRISA | 85h |
| 06h | PORTB | TRISB | 86h |
| 07h | PORTC | TRISC | 87h |
| 08h | — | — | 88h |
| 09h | — | — | 89h |
| 0Ah | PCLATH | PCLATH | 8Ah |
| 0Bh | INTCON | INTCON | 8Bh |
| 0Ch | PIR1 | PIE1 | 8Ch |
| 0Dh | — | — | 8Dh |
| 0Eh | TMR1L | PCON | 8Eh |
| 0Fh | TMR1H | — | 8Fh |
| 10h | T1CON | — | 90h |
| 11h | TMR2 | — | 91h |
| 12h | T2CON | PR2 | 92h |
| 13h | SSPBUF | SSPADD | 93h |
| 14h | SSPCON | SSPSTAT | 94h |
| 15h | CCPR1L | — | 95h |
| 16h | CCPR1H | — | 96h |
| 17h | CCP1CON | — | 97h |
| 18h | — | — | 98h |
| 19h | — | — | 99h |
| 1Ah | — | — | 9Ah |
| 1Bh | — | — | 9Bh |
| 1Ch | — | — | 9Ch |
| 1Dh | — | — | 9Dh |
| 1Eh | ADRES(2) | — | 9Eh |
| 1Fh | ADCON0(2) | ADCON1(2) | 9Fh |
| 20h – 7Fh | General Purpose Registers | General Purpose Registers / — | A0h – FFh |

Unimplemented data memory locations, read as '0'.

**Note 1:** Not a physical register.
**2:** These registers are not implemented on the PIC16C62B, read as '0'.


© 1998-2013 Microchip Technology Inc.

### 2.2.2 SPECIAL FUNCTION REGISTERS

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

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

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

| Addr | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on: POR, BOR | Value on all other resets (4) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bank 0** | | | | | | | | | | | |
| 00h | INDF(1) | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | 0000 0000 | 0000 0000 |
| 01h | TMR0 | Timer0 module's register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 02h | PCL(1) | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 0000 0000 |
| 03h | STATUS(1) | IRP(5) | RP1(5) | RP0 | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C | 0001 1xxx | 000q quuu |
| 04h | FSR(1) | Indirect data memory address pointer | | | | | | | | xxxx xxxx | uuuu uuuu |
| 05h | PORTA(6,7) | — | — | PORTA Data Latch when written: PORTA pins when read | | | | | | --0x 0000 | --0u 0000 |
| 06h | PORTB(6,7) | PORTB Data Latch when written: PORTB pins when read | | | | | | | | xxxx xxxx | uuuu uuuu |
| 07h | PORTC(6,7) | PORTC Data Latch when written: PORTC pins when read | | | | | | | | xxxx xxxx | uuuu uuuu |
| 08h-09h | — | Unimplemented | | | | | | | | — | — |
| 0Ah | PCLATH(1,2) | — | — | — | Write Buffer for the upper 5 bits of the Program Counter | | | | | ---0 0000 | ---0 0000 |
| 0Bh | INTCON(1) | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF(3) | — | — | SSPIF | CCP1IF | TMR2IF | TMR1IF | -0-- 0000 | -0-- 0000 |
| 0Dh | — | Unimplemented | | | | | | | | — | — |
| 0Eh | TMR1L | Holding register for the Least Significant Byte of the 16-bit TMR1 register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 0Fh | TMR1H | Holding register for the Most Significant Byte of the 16-bit TMR1 register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 10h | T1CON | — | — | T1CKPS1 | T1CKPS0 | T1OSCEN | $\overline{T1SYNC}$ | TMR1CS | TMR1ON | --00 0000 | --uu uuuu |
| 11h | TMR2 | Timer2 module's register | | | | | | | | 0000 0000 | 0000 0000 |
| 12h | T2CON | — | TOUTPS3 | TOUTPS2 | TOUTPS1 | TOUTPS0 | TMR2ON | T2CKPS1 | T2CKPS0 | -000 0000 | -000 0000 |
| 13h | SSPBUF | Synchronous Serial Port Receive Buffer/Transmit Register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 14h | SSPCON | WCOL | SSPOV | SSPEN | CKP | SSPM3 | SSPM2 | SSPM1 | SSPM0 | 0000 0000 | 0000 0000 |
| 15h | CCPR1L | Capture/Compare/PWM Register1 (LSB) | | | | | | | | xxxx xxxx | uuuu uuuu |
| 16h | CCPR1H | Capture/Compare/PWM Register1 (MSB) | | | | | | | | xxxx xxxx | uuuu uuuu |
| 17h | CCP1CON | — | — | CCP1X | CCP1Y | CCP1M3 | CCP1M2 | CCP1M1 | CCP1M0 | --00 0000 | --00 0000 |
| 18h-1Dh | — | Unimplemented | | | | | | | | — | — |
| 1Eh | ADRES(3) | A/D Result Register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 1Fh | ADCON0(3) | ADCS1 | ADCS0 | CHS2 | CHS1 | CHS0 | GO/$\overline{DONE}$ | — | ADON | 0000 00-0 | 0000 00-0 |

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

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

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

**3:** A/D not implemented on the PIC16C62B, maintain as '0'.

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

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

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

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

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

| Addr | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on: POR, BOR | Value on all other resets (4) |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Bank 1** | | | | | | | | | | | |
| 80h | INDF(1) | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | 0000 0000 | 0000 0000 |
| 81h | OPTION_REG | $\overline{RBPU}$ | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |
| 82h | PCL(1) | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 0000 0000 |
| 83h | STATUS(1) | IRP(5) | RP1(5) | RP0 | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C | 0001 1xxx | 000q quuu |
| 84h | FSR(1) | Indirect data memory address pointer | | | | | | | | xxxx xxxx | uuuu uuuu |
| 85h | TRISA | — | — | PORTA Data Direction Register | | | | | | --11 1111 | --11 1111 |
| 86h | TRISB | PORTB Data Direction Register | | | | | | | | 1111 1111 | 1111 1111 |
| 87h | TRISC | PORTC Data Direction Register | | | | | | | | 1111 1111 | 1111 1111 |
| 88h-89h | — | Unimplemented | | | | | | | | — | — |
| 8Ah | PCLATH(1,2) | — | — | — | Write Buffer for the upper 5 bits of the Program Counter | | | | | ---0 0000 | ---0 0000 |
| 8Bh | INTCON(1) | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 8Ch | PIE1 | — | ADIE(3) | — | — | SSPIE | CCP1IE | TMR2IE | TMR1IE | -0-- 0000 | -0-- 0000 |
| 8Dh | — | Unimplemented | | | | | | | | — | — |
| 8Eh | PCON | — | — | — | — | — | — | $\overline{POR}$ | $\overline{BOR}$ | ---- --qq | ---- --uu |
| 8Fh-91h | — | Unimplemented | | | | | | | | — | — |
| 92h | PR2 | Timer2 Period Register | | | | | | | | 1111 1111 | 1111 1111 |
| 93h | SSPADD | Synchronous Serial Port ($I^2C$ mode) Address Register | | | | | | | | 0000 0000 | 0000 0000 |
| 94h | SSPSTAT | SMP | CKE | D/$\overline{A}$ | P | S | R/$\overline{W}$ | UA | BF | 0000 0000 | 0000 0000 |
| 95h-9Eh | — | Unimplemented | | | | | | | | — | — |
| 9Fh | ADCON1(3) | — | — | — | — | — | PCFG2 | PCFG1 | PCFG0 | ---- -000 | ---- -000 |

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

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

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

**3:** A/D not implemented on the PIC16C62B, maintain as '0'.

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

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

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

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

 © 1998-2013 Microchip Technology Inc.

2.2.2.1 STATUS REGISTER

The STATUS register, shown in Register 2-1, contains the arithmetic status of the ALU, the RESET status and the bank select bits for data memory.

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

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

It is recommended, therefore, that only `BCF`, `BSF`, `SWAPF` and `MOVWF` instructions are used to alter the STATUS register, because these instructions do not affect the Z, C or DC bits from the STATUS register. For other instructions, not affecting any status bits, see the "Instruction Set Summary."

**Note 1:** The IRP and RP1 bits are reserved. Maintain these bits clear to ensure upward compatibility with future products.

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

**REGISTER 2-1: STATUS REGISTER (ADDRESS 03h, 83h)**

| R/W-0 | R/W-0 | R/W-0 | R-1 | R-1 | R/W-x | R/W-x | R/W-x |
|---|---|---|---|---|---|---|---|
| IRP | RP1 | RP0 | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C |
| bit7 | | | | | | | bit0 |

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

bit 7: **IRP**: Register Bank Select bit (used for indirect addressing)
(reserved, maintain clear)

bit 6-5: **RP1:RP0**: Register Bank Select bits (used for direct addressing)
`01` = Bank 1 (80h - FFh)
`00` = Bank 0 (00h - 7Fh)
Each bank is 128 bytes
**Note:** RP1 is reserved, maintain 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) (for $\overline{borrow}$, the polarity is reversed)
`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.

2.2.2.2 OPTION_REG REGISTER

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

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

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

| R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 | R/W-1 |
|---|---|---|---|---|---|---|---|
| $\overline{\text{RBPU}}$ | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 |
| bit7 | | | | | | | bit0 |

R = Readable bit
W = Writable bit
- n = Value at POR reset

bit 7: **$\overline{\text{RBPU}}$**: PORTB Pull-up Enable bit
1 = PORTB pull-ups are disabled
0 = PORTB pull-ups are enabled for all PORTB inputs

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

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

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

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

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

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

 © 1998-2013 Microchip Technology Inc.

2.2.2.3 INTCON REGISTER

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

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

**REGISTER 2-3: INTCON REGISTER (ADDRESS 0Bh, 8Bh)**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-x |
|---|---|---|---|---|---|---|---|
| GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF |
| bit7 | | | | | | | bit0 |

R = Readable bit
W = Writable bit
- n = Value at POR reset

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

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

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

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

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

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

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

bit 0: **RBIF**: RB Port Change Interrupt Flag bit
1 = At least one of the RB7:RB4 input pins have changed state (clear by reading PORTB)
0 = None of the RB7:RB4 input pins have changed state

2.2.2.4 PIE1 REGISTER

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

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

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

| U-0 | R/W-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | ADIE(1) | — | — | SSPIE | CCP1IE | TMR2IE | TMR1IE |
| bit7 | | | | | | | bit0 |

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

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

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

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

bit 3: **SSPIE**: Synchronous Serial Port Interrupt Enable bit
1 = Enables the SSP interrupt
0 = Disables the SSP interrupt

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

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

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

**Note 1:** The PIC16C62B does not have an A/D module. This bit location is reserved on these devices. Always maintain this bit clear.

 © 1998-2013 Microchip Technology Inc.

2.2.2.5 PIR1 REGISTER

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

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

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

| U-0 | R/W-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | ADIF(1) | — | — | SSPIF | CCP1IF | TMR2IF | TMR1IF |
| bit7 | | | | | | | bit0 |

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

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

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

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

bit 3: **SSPIF**: Synchronous Serial Port Interrupt Flag bit
1 = The transmission/reception is complete (must be cleared in software)
0 = Waiting to transmit/receive

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

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

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

**Note 1:** The PIC16C62B does not have an A/D module. This bit location is reserved on these devices. Always maintain this bit clear.

2.2.2.6 PCON REGISTER

The Power Control register (PCON) contains flag bits to allow differentiation between a Power-on Reset (POR), Brown-Out Reset (BOR) and resets from other sources. .

| | |
|---|---|
| **Note:** | On Power-on Reset, the state of the $\overline{BOR}$ bit is unknown and is not predictable. If the BODEN bit in the configuration word is set, the user must first set the BOR bit on a POR, and check it on subsequent resets. If BOR is cleared while POR remains set, a Brown-out reset has occurred. If the BODEN bit is clear, the BOR bit may be ignored. |

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

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-q |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | — | $\overline{POR}$ | $\overline{BOR}$ |
| bit7 | | | | | | | bit0 |

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

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

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

bit 0: **$\overline{BOR}$**: 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)

 © 1998-2013 Microchip Technology Inc.

## 2.3 PCL and PCLATH

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

### 2.3.1 STACK

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

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

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

## 2.4 Program Memory Paging

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

## 2.5 Indirect Addressing, INDF and FSR Registers

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

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

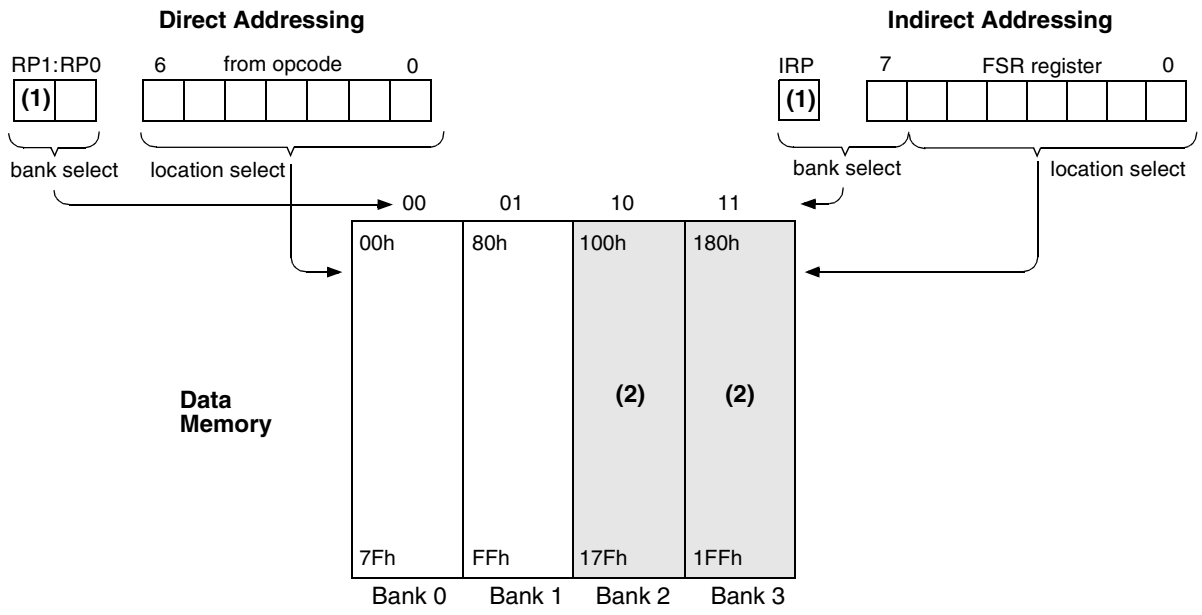
A simple program to clear RAM locations 20h-2Fh using indirect addressing is shown in Example 2-1.

**EXAMPLE 2-1: HOW TO CLEAR RAM USING INDIRECT ADDRESSING**

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

An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit (STATUS<7>), as shown in Figure 2-3. However, IRP is not used in the PIC16C62B/72A.

**FIGURE 2-3: DIRECT/INDIRECT ADDRESSING**




**Note 1:** Maintain clear for upward compatibility with future products.

**2:** Not implemented.

 © 1998-2013 Microchip Technology Inc.

## 3.0 I/O PORTS

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

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

### 3.1 PORTA and the TRISA Register

PORTA is a 6-bit wide bi-directional port. The corresponding data direction register is TRISA. Setting a TRISA bit (=1) will make the corresponding PORTA pin an input, i.e., put the corresponding output driver in a hi-impedance mode. Clearing a TRISA bit (=0) will make the corresponding PORTA pin an output, (i.e., put the contents of the output latch on the selected pin).

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

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

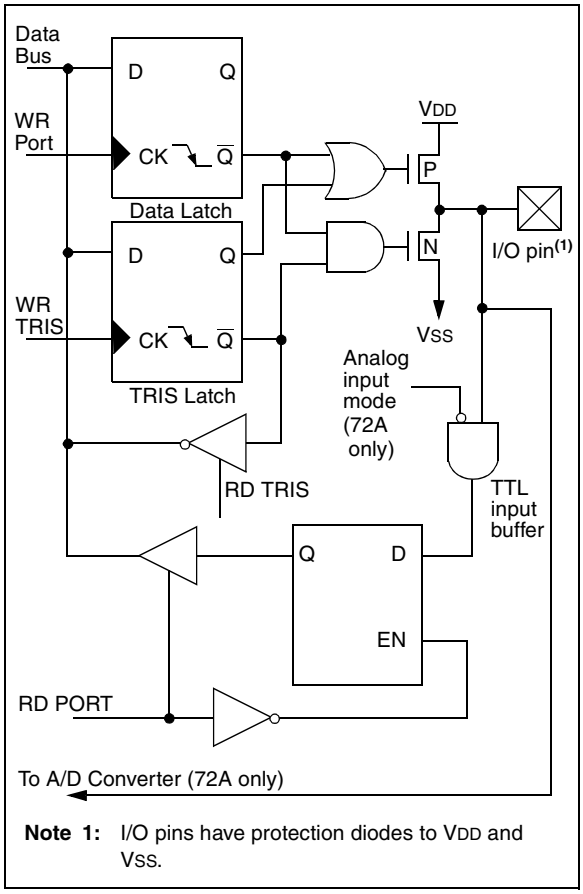
Pin RA5 is multiplexed with the SSP to become the RA5/$\overline{SS}$ pin.

On the PIC16C72A device, other PORTA pins are multiplexed with analog inputs and analog VREF input. The operation of each pin is selected by clearing/setting the control bits in the ADCON1 register (A/D Control Register1).

| **Note:** | On a Power-on Reset, pins with analog functions are configured as analog inputs with digital input buffers disabled . A digital read of these pins will return '0'. |
|---|---|

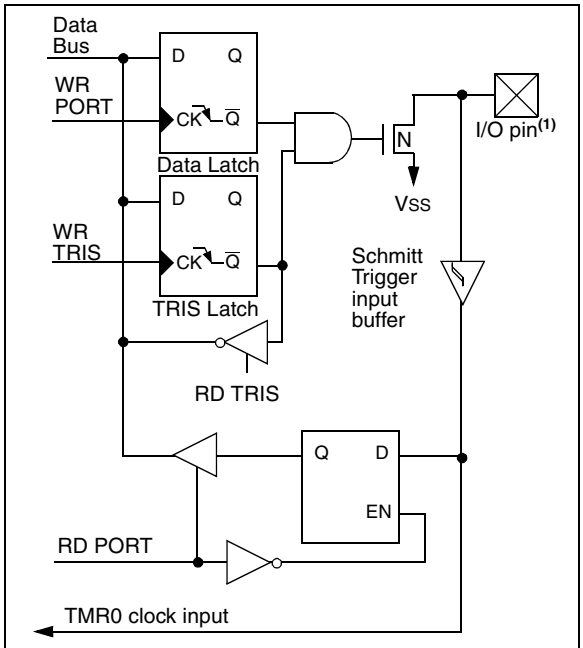
The TRISA register controls the direction of the RA pins, even when they are being used as analog inputs. The user must ensure the bits in the TRISA register are maintained set when using them as analog inputs.

**FIGURE 3-1: BLOCK DIAGRAM OF RA3:RA0 AND RA5 PINS**




**Note 1:** I/O pins have protection diodes to VDD and VSS.

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




**Note 1:** I/O pin has protection diodes to VSS only.

**TABLE 3-1 PORTA FUNCTIONS**

| Name | Bit# | Buffer | Function |
|---|---|---|---|
| RA0/AN0 | bit0 | TTL | Input/output or analog input(1) |
| RA1/AN1 | bit1 | TTL | Input/output or analog input(1) |
| RA2/AN2 | bit2 | TTL | Input/output or analog input(1) |
| RA3/AN3/VREF | bit3 | TTL | Input/output or analog input(1) or VREF(1) |
| RA4/T0CKI | bit4 | ST | Input/output or external clock input for Timer0<br>Output is open drain type |
| RA5/$\overline{SS}$/AN4 | bit5 | TTL | Input/output or slave select input for synchronous serial port or analog input(1) |

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

**Note 1:** The PIC16C62B does not implement the A/D module.

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

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other resets |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 05h | PORTA (for PIC16C72A only) | — | — | RA5 | RA4 | RA3 | RA2 | RA1 | RA0 | --0x 0000 | --0u 0000 |
| 05h | PORTA (for PIC16C62B only) | — | — | RA5 | RA4 | RA3 | RA2 | RA1 | RA0 | --xx xxxx | --uu uuuu |
| 85h | TRISA | — | — | PORTA Data Direction Register | | | | | | --11 1111 | --11 1111 |
| 9Fh | ADCON1(1) | — | — | — | — | — | PCFG2 | PCFG1 | PCFG0 | ---- -000 | ---- -000 |

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

**Note 1:** The PIC16C62B does not implement the A/D module. Maintain this register clear.

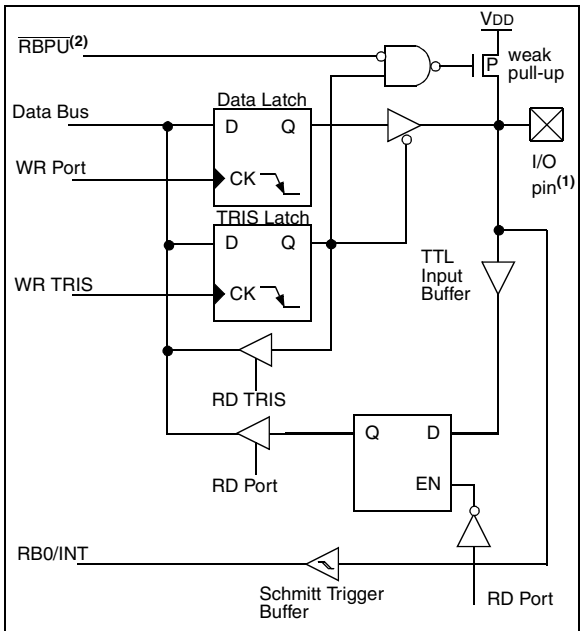
 © 1998-2013 Microchip Technology Inc.

### 3.2 PORTB and the TRISB Register

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

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

**FIGURE 3-3: BLOCK DIAGRAM OF RB3:RB0 PINS**




**Note 1:** I/O pins have diode protection to VDD and VSS.
**2:** To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the $\overline{\text{RBPU}}$ bit (OPTION_REG<7>).

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

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

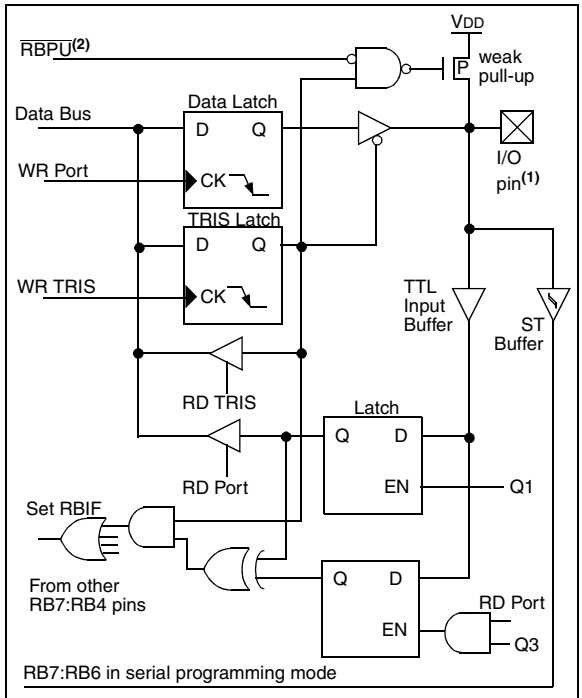
a) Any read or write of PORTB. This will end the mismatch condition.
b) Clear flag bit RBIF.

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.

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.

RB0/INT is an external interupt pin and is configured using the INTEDG bit (OPTION_REG<6>). RB0/INT is discussed in detail in Section 10.10.1.

**FIGURE 3-4: BLOCK DIAGRAM OF RB7:RB4 PINS**




**Note 1:** I/O pins have diode protection to VDD and VSS.
**2:** To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the $\overline{\text{RBPU}}$ bit (OPTION_REG<7>).

**TABLE 3-3 PORTB FUNCTIONS**

| Name | Bit# | Buffer | Function |
|---|---|---|---|
| RB0/INT | bit0 | TTL/ST(1) | Input/output pin or external interrupt input.<br>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).<br>Internal software programmable weak pull-up. |
| RB5 | bit5 | TTL | Input/output pin (with interrupt on change).<br>Internal software programmable weak pull-up. |
| RB6 | bit6 | TTL/ST(2) | Input/output pin (with interrupt on change).<br>Internal software programmable weak pull-up. Serial programming clock. |
| RB7 | bit7 | TTL/ST(2) | Input/output pin (with interrupt on change).<br>Internal software programmable weak pull-up. Serial programming data. |

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

**Note 1:** This buffer is a Schmitt Trigger input when configured as the external interrupt.

**2:** This buffer is a Schmitt Trigger input when used in serial programming mode.

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

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on: POR, BOR | Value on all other resets |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 06h | PORTB | RB7 | RB6 | RB5 | RB4 | RB3 | RB2 | RB1 | RB0 | xxxx xxxx | uuuu uuuu |
| 86h | TRISB | PORTB Data Direction Register | | | | | | | | 1111 1111 | 1111 1111 |
| 81h | OPTION_REG | $\overline{\text{RBPU}}$ | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |

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

 © 1998-2013 Microchip Technology Inc.

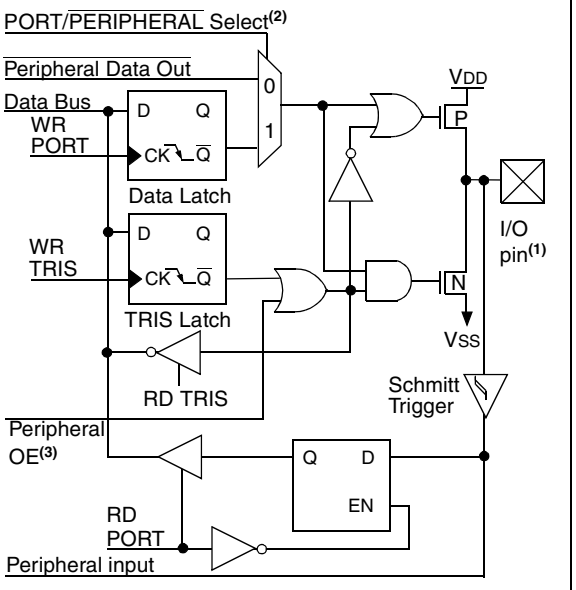
### 3.3 PORTC and the TRISC Register

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

PORTC is multiplexed with several peripheral functions (Table 3-5). PORTC pins have Schmitt Trigger input buffers.

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

**FIGURE 3-5: PORTC BLOCK DIAGRAM (PERIPHERAL OUTPUT OVERRIDE)**




**Note 1:** I/O pins have diode protection to VDD and VSS.
**2:** Port/Peripheral select signal selects between port data and peripheral output.
**3:** Peripheral OE (output enable) is only activated if peripheral select is active.

**TABLE 3-5 PORTC FUNCTIONS**

| Name | Bit# | Buffer Type | Function | TRISC Override |
|---|---|---|---|---|
| RC0/T1OSO/T1CKI | bit0 | ST | Input/output port pin or Timer1 oscillator output/Timer1 clock input | Yes |
| RC1/T1OSI | bit1 | ST | Input/output port pin or Timer1 oscillator input | Yes |
| RC2/CCP1 | bit2 | ST | Input/output port pin or Capture1 input/Compare1 output/PWM1 output | No |
| RC3/SCK/SCL | bit3 | ST | RC3 can also be the synchronous serial clock for both SPI and $I^2C$ modes. | No |
| RC4/SDI/SDA | bit4 | ST | RC4 can also be the SPI Data In (SPI mode) or data I/O ($I^2C$ mode). | No |
| RC5/SDO | bit5 | ST | Input/output port pin or Synchronous Serial Port data output | No |
| RC6 | bit6 | ST | Input/output port pin | No |
| RC7 | bit7 | ST | Input/output port pin | No |

Legend: ST = Schmitt Trigger input

**TABLE 3-6 SUMMARY OF REGISTERS ASSOCIATED WITH PORTC**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on: POR, BOR | Value on all other resets |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 07h | PORTC | RC7 | RC6 | RC5 | RC4 | RC3 | RC2 | RC1 | RC0 | xxxx xxxx | uuuu uuuu |
| 87h | TRISC | PORTC Data Direction Register | | | | | | | | 1111 1111 | 1111 1111 |

Legend: x = unknown, u = unchanged.

 © 1998-2013 Microchip Technology Inc.

## 4.0 TIMER0 MODULE

The Timer0 module timer/counter has the following features:

- 8-bit timer/counter
  - Read and write
  - INT on overflow
- 8-bit software programmable prescaler
- INT or EXT clock select
  - EXT clock edge select

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

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

### 4.1 Timer0 Operation

Timer0 can operate as a timer or as a counter.

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

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

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

Additional information on external clock requirements is available in the Electrical Specifications section of this manual, and in the PIC® MCU Mid-Range Reference Manual, (DS33023).

### 4.2 Prescaler

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

The prescaler is not readable or writable.

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

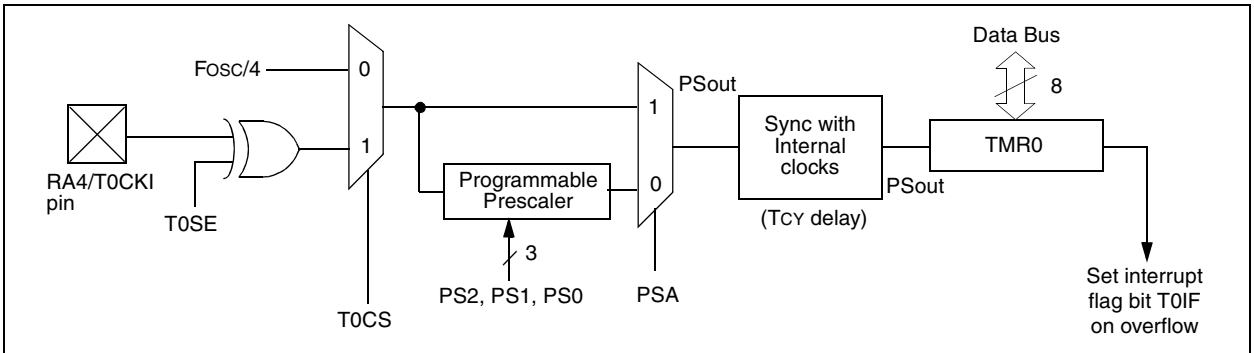
Clearing bit PSA will assign the prescaler to the Timer0 module. When the prescaler is assigned to the Timer0 module, prescale values of 1:2, 1:4, ..., 1:256 are selectable.

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

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

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

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




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

4.2.1 SWITCHING PRESCALER ASSIGNMENT

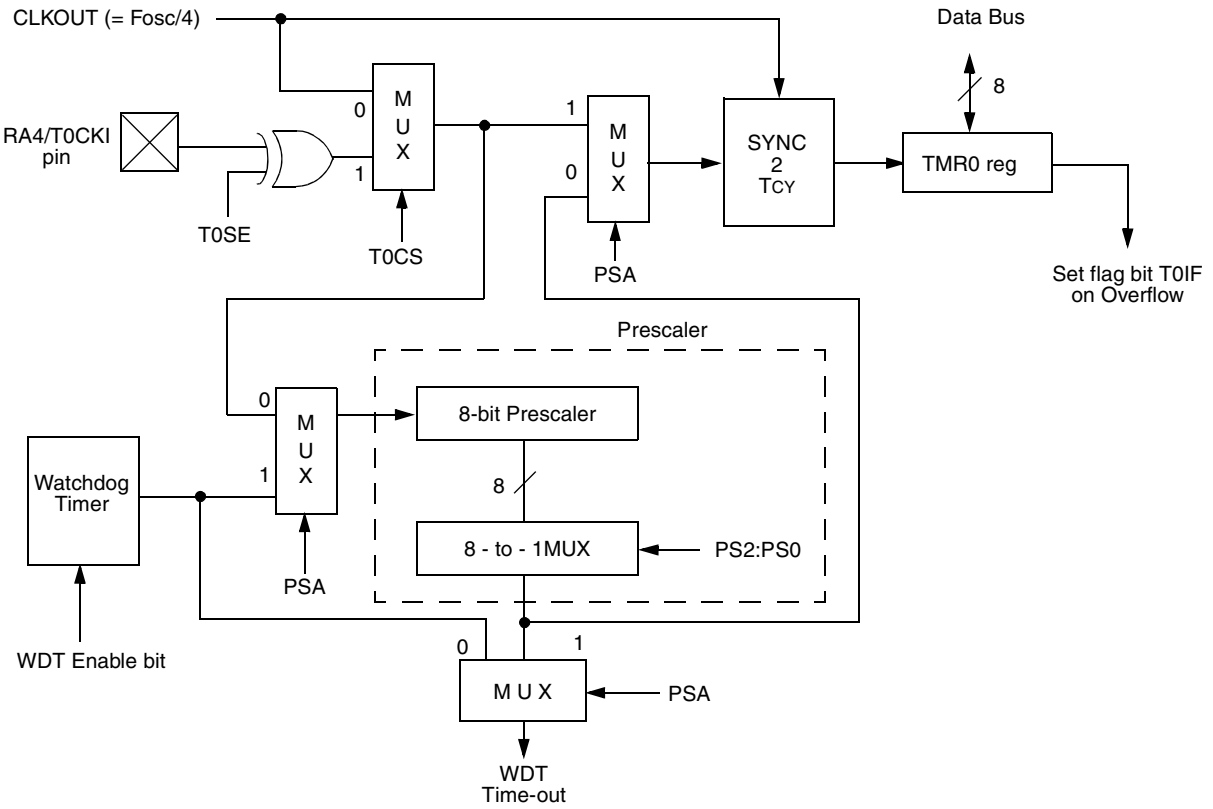
The prescaler assignment is fully under software control, (i.e., it can be changed “on-the-fly” during program execution).

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

## 4.3 Timer0 Interrupt

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

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




**Note:** T0CS, T0SE, PSA, PS2:PS0 are (OPTION_REG<5:0>).

**TABLE 4-1 REGISTERS ASSOCIATED WITH TIMER0**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on: POR, BOR | Value on all other resets |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 01h | TMR0 | Timer0 module's register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 0Bh,8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 81h | OPTION_REG | $\overline{\text{RBPU}}$ | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |
| 85h | TRISA | — | — | PORTA Data Direction Register | | | | | | --11 1111 | --11 1111 |

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by Timer0.

 © 1998-2013 Microchip Technology Inc.

## 5.0 TIMER1 MODULE

The Timer1 module timer/counter has the following features:

- 16-bit timer/counter
- Readable and writable
- Internal or external clock select
- Interrupt on overflow from FFFFh to 0000h
- Reset from CCP module trigger

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

Figure 5-1 is a simplified block diagram of the Timer1 module.

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

### 5.1 Timer1 Operation

Timer1 can operate in one of these modes:

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

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

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

When the Timer1 oscillator is enabled (T1OSCEN is set), the RC1/T1OSI and RC0/T1OSO/T1CKI pins become inputs. That is, the TRISC<1:0> value is ignored.

Timer1 also has an internal “reset input”. This reset can be generated by the CCP module as a special event trigger (Section 7.0).

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

| U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | T1CKPS1 | T1CKPS0 | T1OSCEN | $\overline{\text{T1SYNC}}$ | TMR1CS | TMR1ON |
| bit7 | | | | | | | bit0 |

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

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

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

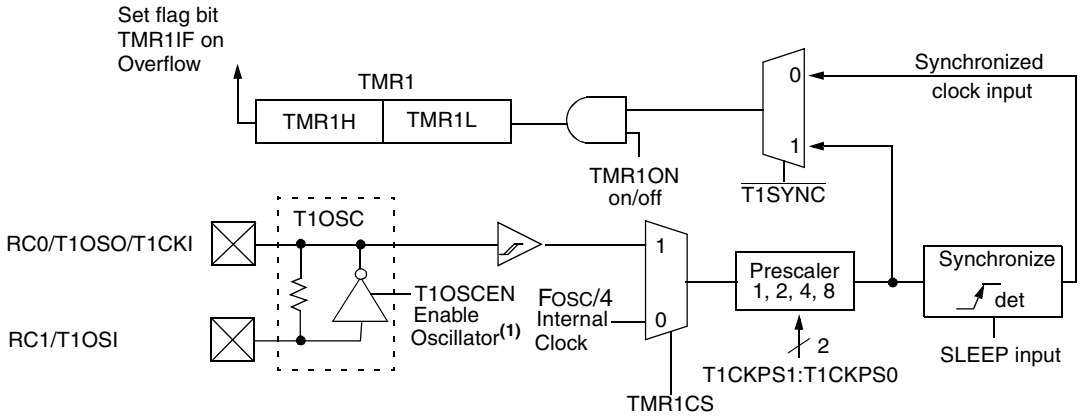
bit 3: **T1OSCEN**: Timer1 Oscillator Enable Control bit
1 = Oscillator is enabled (TRISC<1:0> ignored)
0 = Oscillator is shut off
(The oscillator is turned off to reduce power drain

bit 2: **$\overline{\text{T1SYNC}}$**: Timer1 External Clock Input Synchronization Control bit
TMR1CS = 1
1 = Do not synchronize external clock input
0 = Synchronize external clock input
TMR1CS = 0
This bit is ignored. Timer1 uses the internal clock when TMR1CS = 0.

bit 1: **TMR1CS**: Timer1 Clock Source Select bit
1 = External clock from pin RC0/T1OSO/T1CKI (on the rising edge)
0 = Internal clock (FOSC/4)

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

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




**Note 1:** When the T1OSCEN bit is cleared, the inverter and feedback resistor are turned off. This eliminates power drain.

 © 1998-2013 Microchip Technology Inc.

## 5.2 Timer1 Oscillator

A crystal oscillator circuit is built-in between pins T1OSI (input) and T1OSO (amplifier output). It is enabled by setting control bit T1OSCEN (T1CON<3>). When the Timer1 oscillator is enabled, RC0 and RC1 pins become T1OSO and T1OSI inputs, overriding TRISC<1:0>.

The oscillator is a low power oscillator rated up to 200 kHz. It will continue to run during SLEEP. It is primarily intended for a 32 kHz crystal. Table 5-1 shows the capacitor selection for the Timer1 oscillator.

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

**TABLE 5-1 CAPACITOR SELECTION FOR THE TIMER1 OSCILLATOR**

| Osc Type | Freq | C1 | C2 |
|---|---|---|---|
| LP | 32 kHz | 33 pF | 33 pF |
| | 100 kHz | 15 pF | 15 pF |
| | 200 kHz | 15 pF | 15 pF |
| **These values are for design guidance only.** | | | |
| **Crystals Tested:** | | | |
| 32.768 kHz | Epson C-001R32.768K-A | | ± 20 PPM |
| 100 kHz | Epson C-2 100.00 KC-P | | ± 20 PPM |
| 200 kHz | STD XTL 200.000 kHz | | ± 20 PPM |

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

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

PRELIMINARY

## 5.3 Timer1 Interrupt

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

## 5.4 Resetting Timer1 using a CCP Trigger Output

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

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

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

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

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

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

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other resets |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0Bh,8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | SSPIF | CCP1IF | TMR2IF | TMR1IF | -0-- 0000 | -0-- 0000 |
| 8Ch | PIE1 | — | ADIE | — | — | SSPIE | CCP1IE | TMR2IE | TMR1IE | -0-- 0000 | -0-- 0000 |
| 0Eh | TMR1L | Holding register for the Least Significant Byte of the 16-bit TMR1 register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 0Fh | TMR1H | Holding register for the Most Significant Byte of the 16-bit TMR1 register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 10h | T1CON | — | — | T1CKPS1 | T1CKPS0 | T1OSCEN | $\overline{\text{T1SYNC}}$ | TMR1CS | TMR1ON | --00 0000 | --uu uuuu |

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

**NOTES:**

 © 1998-2013 Microchip Technology Inc.

## 6.0 TIMER2 MODULE

The Timer2 module timer has the following features:

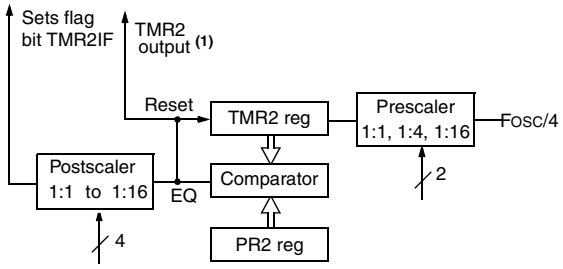
- 8-bit timer (TMR2 register)
  - Readable and writable
- 8-bit period register (PR2)
  - Readable and writable
- Software programmable prescaler (1:1, 1:4, 1:16)
- Software programmable postscaler (1:1 to 1:16)
- Interrupt on match (TMR2 = PR2)
- Timer2 can be used by SSP and CCP

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

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

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

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




**Note 1:** TMR2 register output can be software selected by the SSP Module as a baud clock.

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

| U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | TOUTPS3 | TOUTPS2 | TOUTPS1 | TOUTPS0 | TMR2ON | T2CKPS1 | T2CKPS0 |
| bit7 | | | | | | | bit0 |

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

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

bit 6-3: **TOUTPS3:TOUTPS0**: Timer2 Output Postscale Select bits
0000 = 1:1 Postscale
0001 = 1:2 Postscale
0010 = 1:3 Postscale
•
•
•
1111 = 1:16 Postscale

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

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

### 6.1 Timer2 Operation

The Timer2 output is also used by the CCP module to generate the PWM "On-Time", and the PWM period with a match with PR2.

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

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

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

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

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

TMR2 is not cleared when T2CON is written.

### 6.2 Timer2 Interrupt

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

### 6.3 Output of TMR2

The output of TMR2 (before the postscaler) is fed to the Synchronous Serial Port module, which optionally uses it to generate shift clock.

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

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other resets |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0Bh,8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | SSPIF | CCP1IF | TMR2IF | TMR1IF | -00- 0000 | 0000 0000 |
| 8Ch | PIE1 | — | ADIE | — | — | SSPIE | CCP1IE | TMR2IE | TMR1IE | -0-- 0000 | 0000 0000 |
| 11h | TMR2 | Timer2 module's register | | | | | | | | 0000 0000 | 0000 0000 |
| 12h | T2CON | — | TOUTPS3 | TOUTPS2 | TOUTPS1 | TOUTPS0 | TMR2ON | T2CKPS1 | T2CKPS0 | -000 0000 | -000 0000 |
| 92h | PR2 | Timer2 Period Register | | | | | | | | 1111 1111 | 1111 1111 |

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

 © 1998-2013 Microchip Technology Inc.

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

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

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

Additional information on the CCP module is available in the PIC® MCU Mid-Range Reference Manual, (DS33023).

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

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

**TABLE 7-2 INTERACTION OF TWO CCP MODULES**

| CCPx Mode | CCPy Mode | Interaction |
|---|---|---|
| Capture | Capture | Same TMR1 time-base. |
| Capture | Compare | The compare should be configured for the special event trigger, which clears TMR1. |
| Compare | Compare | The compare(s) should be configured for the special event trigger, which clears TMR1. |
| PWM | PWM | The PWMs will have the same frequency and update rate (TMR2 interrupt). |
| PWM | Capture | None. |
| PWM | Compare | None. |

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

| U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | CCP1X | CCP1Y | CCP1M3 | CCP1M2 | $\overline{\text{CCP1M1}}$ | $\overline{\text{CCP1M0}}$ |
| bit7 | | | | | | | bit0 |

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

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

bit 5-4: **CCP1X:CCP1Y**: PWM Least Significant bits
Capture Mode: Unused
Compare Mode: Unused
PWM Mode: These bits are the two LSbs of the PWM duty cycle. The eight MSbs are found in CCPR1L.

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

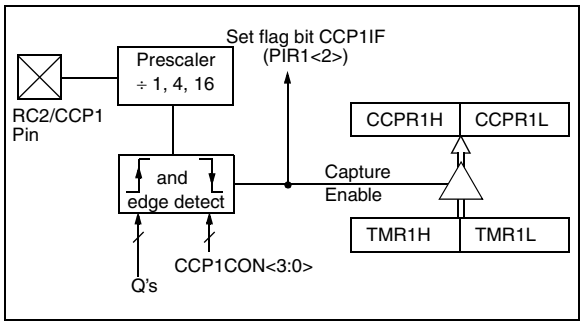
## 7.1 Capture Mode

In Capture mode, CCPR1H:CCPR1L captures the 16-bit value of the TMR1 register, when an event occurs on pin RC2/CCP1. An event is defined as:

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

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

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




### 7.1.1 CCP PIN CONFIGURATION

In Capture mode, the RC2/CCP1 pin should be configured as an input by setting the TRISC<2> bit.

| **Note:** | If the RC2/CCP1 is configured as an output, a write to the port can cause a capture condition. |
|---|---|

### 7.1.2 TIMER1 MODE SELECTION

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

### 7.1.3 SOFTWARE INTERRUPT

When the Capture mode is changed, a false capture interrupt may be generated. The user should clear CCP1IE (PIE1<2>) before changing the capture mode to avoid false interrupts. Clear the interrupt flag bit, CCP1IE before setting CCP1IE.

### 7.1.4 CCP PRESCALER

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

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

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

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

 © 1998-2013 Microchip Technology Inc.

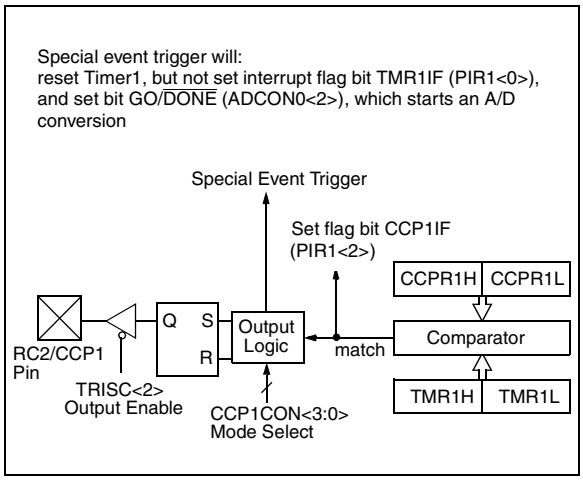
## 7.2 Compare Mode

In Compare mode, the 16-bit CCPR1 register value is constantly compared against the TMR1 register pair value. When a match occurs, the RC2/CCP1 pin is:

- driven High
- driven Low
- remains Unchanged

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

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




### 7.2.1 CCP PIN CONFIGURATION

The user must configure the RC2/CCP1 pin as an output by clearing the TRISC<2> bit.

| **Note:** | Clearing the CCP1CON register will force the RC2/CCP1 compare output latch to the default low level. This is not the data latch. |
|---|---|

### 7.2.2 TIMER1 MODE SELECTION

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

### 7.2.3 SOFTWARE INTERRUPT MODE

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

### 7.2.4 SPECIAL EVENT TRIGGER

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

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

The special trigger output of CCP1 resets the TMR1 register pair and starts an A/D conversion (if the A/D module is enabled).

**TABLE 7-3 REGISTERS ASSOCIATED WITH CAPTURE, COMPARE, AND TIMER1**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other resets |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0Bh,8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | SSPIF | CCP1IF | TMR2IF | TMR1IF | -0-- 0000 | -0-- 0000 |
| 8Ch | PIE1 | — | ADIE | — | — | SSPIE | CCP1IE | TMR2IE | TMR1IE | -0-- 0000 | -0-- 0000 |
| 87h | TRISC | PORTC Data Direction Register | | | | | | | | 1111 1111 | 1111 1111 |
| 0Eh | TMR1L | Holding register for the Least Significant Byte of the 16-bit TMR1 register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 0Fh | TMR1H | Holding register for the Most Significant Byte of the 16-bit TMR1register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 10h | T1CON | — | — | T1CKPS1 | T1CKPS0 | T1OSCEN | $\overline{\text{T1SYNC}}$ | TMR1CS | TMR1ON | --00 0000 | --uu uuuu |
| 15h | CCPR1L | Capture/Compare/PWM register1 (LSB) | | | | | | | | xxxx xxxx | uuuu uuuu |
| 16h | CCPR1H | Capture/Compare/PWM register1 (MSB) | | | | | | | | xxxx xxxx | uuuu uuuu |
| 17h | CCP1CON | — | — | CCP1X | CCP1Y | CCP1M3 | CCP1M2 | CCP1M1 | CCP1M0 | --00 0000 | --00 0000 |

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

## 7.3 PWM Mode

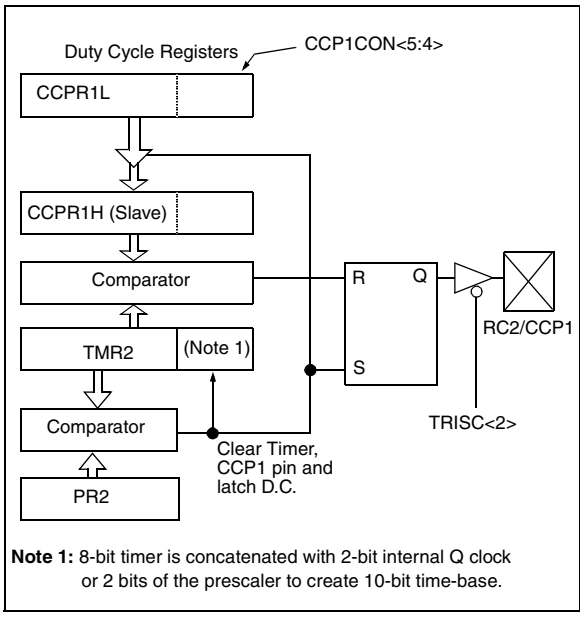
In Pulse Width Modulation (PWM) mode, the CCP1 pin produces up to a 10-bit resolution PWM output. Since the CCP1 pin is multiplexed with the PORTC data latch, the TRISC<2> bit must be cleared to make the CCP1 pin an output.

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

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

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

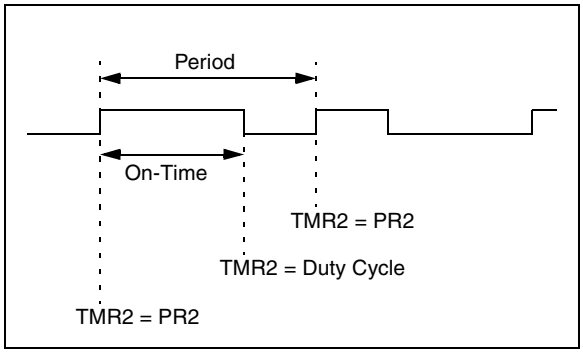
**FIGURE 7-3: SIMPLIFIED PWM BLOCK DIAGRAM**




**Note 1:** 8-bit timer is concatenated with 2-bit internal Q clock or 2 bits of the prescaler to create 10-bit time-base.

A PWM output (Figure 7-4) has a time base (period) and a time that the output stays high (on-time). The frequency of the PWM is the inverse of the period (1/period).

**FIGURE 7-4: PWM OUTPUT**




### 7.3.1 PWM PERIOD

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

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

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

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

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

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

### 7.3.2 PWM ON-TIME

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

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

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

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

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

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

$$\text{Resolution} = \frac{\log\left(\frac{Fosc}{Fpwm}\right)}{\log(2)} \text{ bits}$$

| **Note:** | If the PWM on-time value is larger than the PWM period, the CCP1 pin will not be cleared. |
|---|---|

For an example PWM period and on-time calculation, see the PIC® MCU Mid-Range Reference Manual, (DS33023).

 © 1998-2013 Microchip Technology Inc.

### 7.3.3 SET-UP FOR PWM OPERATION

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

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

**TABLE 7-4 EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 20 MHz**

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

**TABLE 7-5 REGISTERS ASSOCIATED WITH PWM AND TIMER2**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other resets |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0Bh,8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | SSPIF | CCP1IF | TMR2IF | TMR1IF | -0-- 0000 | -0-- 0000 |
| 8Ch | PIE1 | — | ADIE | — | — | SSPIE | CCP1IE | TMR2IE | TMR1IE | -0-- 0000 | -0-- 0000 |
| 87h | TRISC | PORTC Data Direction Register | | | | | | | | 1111 1111 | 1111 1111 |
| 11h | TMR2 | Timer2 module's register | | | | | | | | 0000 0000 | 0000 0000 |
| 92h | PR2 | Timer2 module's period register | | | | | | | | 1111 1111 | 1111 1111 |
| 12h | T2CON | — | TOUTPS3 | TOUTPS2 | TOUTPS1 | TOUTPS0 | TMR2ON | T2CKPS1 | T2CKPS0 | -000 0000 | -000 0000 |
| 15h | CCPR1L | Capture/Compare/PWM register1 (LSB) | | | | | | | | xxxx xxxx | uuuu uuuu |
| 16h | CCPR1H | Capture/Compare/PWM register1 (MSB) | | | | | | | | xxxx xxxx | uuuu uuuu |
| 17h | CCP1CON | — | — | CCP1X | CCP1Y | CCP1M3 | CCP1M2 | CCP1M1 | CCP1M0 | --00 0000 | --00 0000 |

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

**NOTES:**

 © 1998-2013 Microchip Technology Inc.

# 8.0 SYNCHRONOUS SERIAL PORT (SSP) MODULE

## 8.1 SSP Module Overview

The Synchronous Serial Port (SSP) module is a serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be Serial EEPROMs, shift registers, display drivers, A/D converters, etc. The SSP module can operate in one of two modes:

- Serial Peripheral Interface (SPI)
- Inter-Integrated Circuit ($I^2C$)

For more information on SSP operation (including an $I^2C$ Overview), refer to the PIC® MCU Mid-Range Reference Manual, (DS33023). Also, refer to Application Note AN578, *"Use of the SSP Module in the $I^2C$ Multi-Master Environment."*

## 8.2 SPI Mode

This section contains register definitions and operational characteristics of the SPI module.

Additional information on SPI operation may be found in the PIC® MCU Mid-Range Reference Manual, (DS33023).

### 8.2.1 OPERATION OF SSP MODULE IN SPI MODE

A block diagram of the SSP Module in SPI Mode is shown in Figure 8-1.

The SPI mode allows 8-bits of data to be synchronously transmitted and received simultaneously. To accomplish communication, three pins are used:

- Serial Data Out (SDO)RC5/SDO
- Serial Data In (SDI)RC4/SDI/SDA
- Serial Clock (SCK)RC3/SCK/SCL

Additionally, a fourth pin may be used when in a slave mode of operation:

- Slave Select ($\overline{SS}$)RA5/$\overline{SS}$/AN4

When initializing the SPI, several options need to be specified. This is done by programming the appropriate control bits in the SSPCON register (SSPCON<5:0>) and SSPSTAT<7:6>. These control bits allow the following to be specified:

- Master Operation (SCK is the clock output)
- Slave Mode (SCK is the clock input)
- Clock Polarity (Idle state of SCK)
- Clock Edge (Output data on rising/falling edge of SCK)
- Clock Rate (master operation only)
- Slave Select Mode (Slave mode only)

To enable the serial port, SSP Enable bit, SSPEN (SSPCON<5>) must be set. To reset or reconfigure SPI mode, clear bit SSPEN, re-initialize the SSPCON register, and then set bit SSPEN. This configures the SDI, SDO, SCK and $\overline{SS}$ pins as serial port pins. For the pins to behave as the serial port function, they must have their data direction bits (in the TRISC register) appropriately programmed. That is:

- SDI must have TRISC<4> set
- SDO must have TRISC<5> cleared
- SCK (master operation) must have TRISC<3> cleared
- SCK (Slave mode) must have TRISC<3> set
- $\overline{SS}$ must have TRISA<5> set (if used)

| | |
|---|---|
| **Note:** | When the SPI is in Slave Mode with $\overline{SS}$ pin control enabled, (SSPCON<3:0> = 0100) the SPI module will reset if the $\overline{SS}$ pin is set to VDD. |
| **Note:** | If the SPI is used in Slave Mode with CKE = '1', then the $\overline{SS}$ pin control must be enabled. |

**FIGURE 8-1: SSP BLOCK DIAGRAM (SPI MODE)**

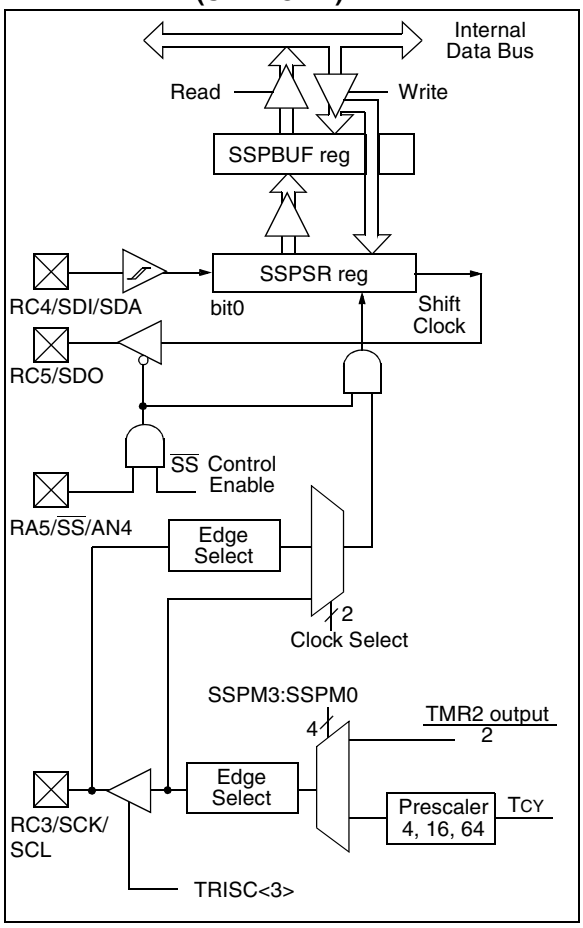


**TABLE 8-1 REGISTERS ASSOCIATED WITH SPI OPERATION**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other resets |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0Bh,8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | SSPIF | CCP1IF | TMR2IF | TMR1IF | -0-- 0000 | -0-- 0000 |
| 8Ch | PIE1 | — | ADIE | — | — | SSPIE | CCP1IE | TMR2IE | TMR1IE | -0-- 0000 | -0-- 0000 |
| 13h | SSPBUF | Synchronous Serial Port Receive Buffer/Transmit Register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 14h | SSPCON | WCOL | SSPOV | SSPEN | CKP | SSPM3 | SSPM2 | SSPM1 | SSPM0 | 0000 0000 | 0000 0000 |
| 94h | SSPSTAT | SMP | CKE | D/$\overline{A}$ | P | S | R/$\overline{W}$ | UA | BF | 0000 0000 | 0000 0000 |
| 85h | TRISA | — | — | PORTA Data Direction Register | | | | | | --11 1111 | --11 1111 |
| 87h | TRISC | PORTC Data Direction Register | | | | | | | | 1111 1111 | 1111 1111 |

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by the SSP in SPI mode.

 © 1998-2013 Microchip Technology Inc.

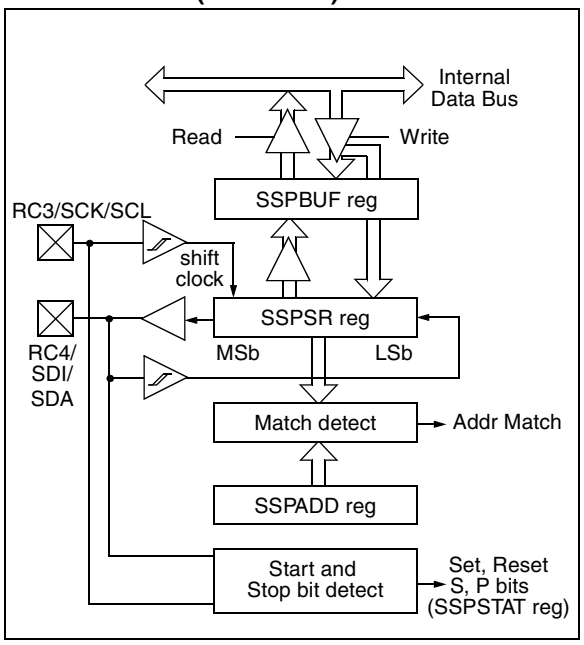
## 8.3 SSP I²C Operation

The SSP module in I²C mode fully implements all slave functions, except general call support, and provides interrupts on start and stop bits in hardware to support firmware implementations of the master functions. The SSP module implements the standard mode specifications, as well as 7-bit and 10-bit addressing.

Two pins are used for data transfer. These are the RC3/SCK/SCL pin, which is the clock (SCL), and the RC4/SDI/SDA pin, which is the data (SDA). The user must configure these pins as inputs or outputs through the TRISC<4:3> bits.

The SSP module functions are enabled by setting SSP Enable bit SSPEN (SSPCON<5>).

**FIGURE 8-2: SSP BLOCK DIAGRAM (I²C MODE)**




The SSP module has five registers for I²C operation. These are the:

- SSP Control Register (SSPCON)
- SSP Status Register (SSPSTAT)
- Serial Receive/Transmit Buffer (SSPBUF)
- SSP Shift Register (SSPSR) - Not accessible
- SSP Address Register (SSPADD)

The SSPCON register allows control of the I²C operation. Four mode selection bits (SSPCON<3:0>) allow one of the following I²C modes to be selected:

- I²C Slave mode (7-bit address)
- I²C Slave mode (10-bit address)
- I²C Slave mode (7-bit address), with start and stop bit interrupts enabled for firmware master mode support
- I²C Slave mode (10-bit address), with start and stop bit interrupts enabled for firmware master mode support
- I²C start and stop bit interrupts enabled for firmware master mode support, slave mode idle

Selection of any I²C mode, with the SSPEN bit set, forces the SCL and SDA pins to be operated as open drain outputs, provided these pins are programmed to inputs by setting the appropriate TRISC bits.

Additional information on SSP I²C operation may be found in the PIC® MCU Mid-Range Reference Manual, (DS33023).

### 8.3.1 SLAVE MODE

In slave mode, the SCL and SDA pins must be configured as inputs (TRISC<4:3> set). The SSP module will override the input state with the output data when required (slave-transmitter).

When an address is matched or the data transfer after an address match is received, the hardware automatically will generate the acknowledge ($\overline{ACK}$) pulse, and load the SSPBUF register with the received value in the SSPSR register.

There are certain conditions that will cause the SSP module not to give this $\overline{ACK}$ pulse. This happens if either of the following conditions occur:

a) The buffer full bit BF (SSPSTAT<0>) was set before the transfer was completed.

b) The overflow bit SSPOV (SSPCON<6>) was set before the transfer was completed.

In this case, the SSPSR register value is not loaded into the SSPBUF, but bit SSPIF (PIR1<3>) is set. Table 8-2 shows what happens when a data transfer byte is received, given the status of bits BF and SSPOV. The shaded cells show the condition where user software did not properly clear the overflow condition. Flag bit BF is cleared by reading the SSPBUF register, while bit SSPOV is cleared through software.

The SCL clock input must have a minimum high and low for proper operation. The high and low times of the I²C specification, as well as the requirement of the SSP module, is shown in timing parameter #100, THIGH, and parameter #101, TLOW.

8.3.1.1 ADDRESSING

Once the SSP module has been enabled, it waits for a START condition to occur. Following the START condition, 8 bits are shifted into the SSPSR register. All incoming bits are sampled with the rising edge of the clock (SCL) line. The value of register SSPSR<7:1> is compared to the value of the SSPADD register. The address is compared on the falling edge of the eighth clock (SCL) pulse. If the addresses match and the BF and SSPOV bits are clear, the following events occur:

a) The SSPSR register value is loaded into the SSPBUF register.
b) The buffer full bit, BF is set.
c) An $\overline{ACK}$ pulse is generated.
d) SSP interrupt flag bit, SSPIF (PIR1<3>), is set (interrupt is generated if enabled) on the falling edge of the ninth SCL pulse.

In 10-bit address mode, two address bytes need to be received by the slave. The five Most Significant bits (MSbs) of the first address byte specify if this is a 10-bit address. Bit R/$\overline{W}$ (SSPSTAT<2>) must specify a write so the slave device will receive the second address byte. For a 10-bit address, the first byte would equal '1111 0 A9 A8 0', where A9 and A8 are the two MSbs of the address. The sequence of events for 10-bit address is as follows, with steps 7- 9 for slave-transmitter:

1. Receive first (high) byte of Address (bits SSPIF, BF, and bit UA (SSPSTAT<1>) are set).
2. Update the SSPADD register with second (low) byte of Address (clears bit UA and releases the SCL line).
3. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.
4. Receive second (low) byte of Address (bits SSPIF, BF, and UA are set).
5. Update the SSPADD register with the first (high) byte of Address, if match releases SCL line, this will clear bit UA.
6. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.
7. Receive repeated START condition.
8. Receive first (high) byte of Address (bits SSPIF and BF are set).
9. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.

**TABLE 8-2 DATA TRANSFER RECEIVED BYTE ACTIONS**

| Status Bits as Data Transfer is Received | | | | |
|---|---|---|---|---|
| **BF** | **SSPOV** | **SSPSR → SSPBUF** | **Generate $\overline{ACK}$ Pulse** | **Set bit SSPIF (SSP Interrupt occurs if enabled)** |
| 0 | 0 | Yes | Yes | Yes |
| 1 | 0 | No | No | Yes |
| 1 | 1 | No | No | Yes |
| 0 | 1 | Yes | No | Yes |

**Note:** Shaded cells show the conditions where the user software did not properly clear the overflow condition.

 © 1998-2013 Microchip Technology Inc.

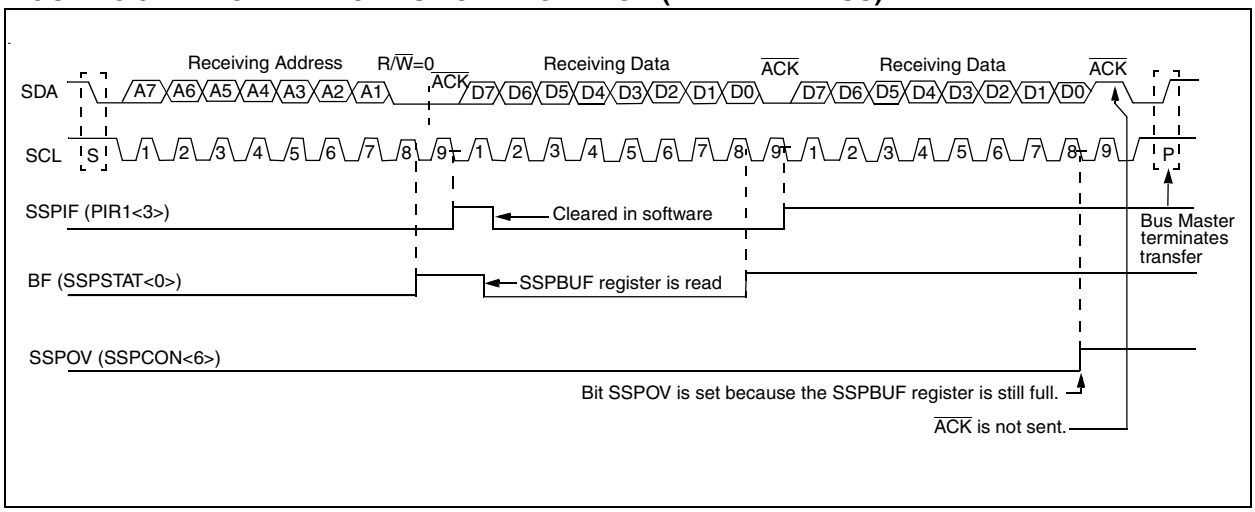
8.3.1.2 RECEPTION

When the R/$\overline{W}$ bit of the address byte is clear and an address match occurs, the R/$\overline{W}$ bit of the SSPSTAT register is cleared. The received address is loaded into the SSPBUF register.

When the address byte overflow condition exists, then no acknowledge ($\overline{ACK}$) pulse is given. An overflow condition is defined as either bit BF (SSPSTAT<0>) is set or bit SSPOV (SSPCON<6>) is set.

An SSP interrupt is generated for each data transfer byte. Flag bit SSPIF (PIR1<3>) must be cleared in software. The SSPSTAT register is used to determine the status of the byte.

**FIGURE 8-3: $I^2C$ WAVEFORMS FOR RECEPTION (7-BIT ADDRESS)**



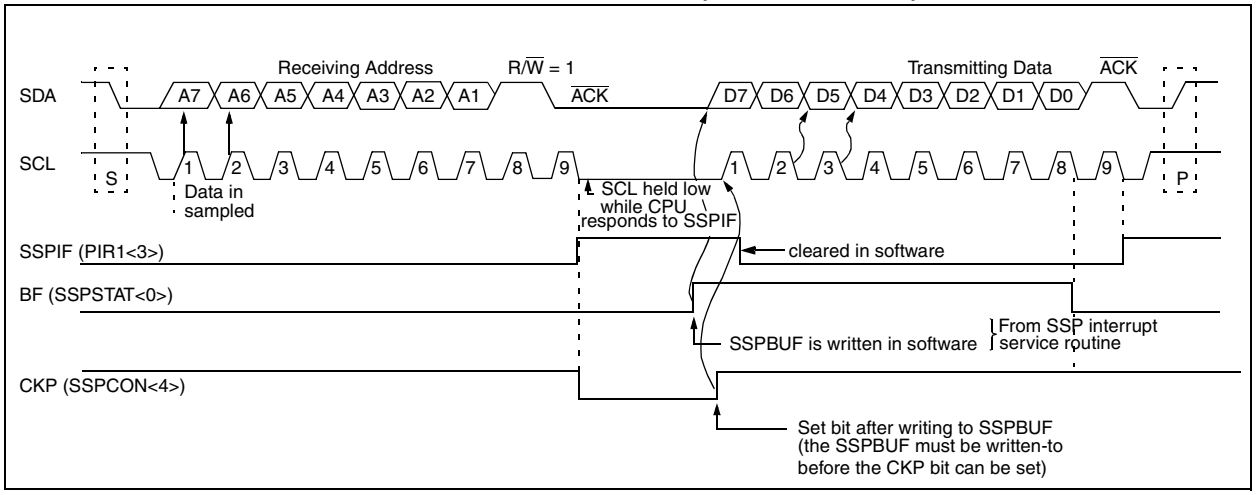
8.3.1.3 TRANSMISSION

When the R/$\overline{W}$ bit of the incoming address byte is set and an address match occurs, the R/$\overline{W}$ bit of the SSPSTAT register is set. The received address is loaded into the SSPBUF register. The $\overline{ACK}$ pulse will be sent on the ninth bit and the CKP will be cleared by hardware, holding SCL low. Slave devices cause the master to wait by holding the SCL line low. The transmit data is loaded into the SSPBUF register, which in turn loads the SSPSR register. When bit CKP (SSPCON<4>) is set, pin RC3/SCK/SCL releases SCL. When the SCL line goes high, the master may resume operating the SCL line and receiving data. The master must monitor the SCL pin prior to asserting another clock pulse. The slave devices may be holding off the master by stretching the clock. The eight data bits are shifted out on the falling edge of the SCL input. This ensures that the SDA signal is valid during the SCL high time (Figure 8-4).

An SSP interrupt is generated for each data transfer byte. Flag bit SSPIF must be cleared in software, and the SSPSTAT register used to determine the status of the byte. Flag bit SSPIF is set on the falling edge of the ninth clock pulse.

As a slave-transmitter, the $\overline{ACK}$ pulse from the master-receiver is latched on the rising edge of the ninth SCL input pulse. If the SDA line was high (not $\overline{ACK}$), then the data transfer is complete. When the $\overline{ACK}$ is latched by the slave, the slave logic is reset (resets SSPSTAT register) and the slave then monitors for another occurrence of the START bit. If the SDA line was low ($\overline{ACK}$), the transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. Then pin RC3/SCK/SCL should be enabled by setting bit CKP.

**FIGURE 8-4: I²C WAVEFORMS FOR TRANSMISSION (7-BIT ADDRESS)**




 © 1998-2013 Microchip Technology Inc.

### 8.3.2 MASTER OPERATION

Master operation is supported in firmware using interrupt generation on the detection of the START and STOP conditions. The STOP (P) and START (S) bits are cleared by a reset or when the SSP module is disabled. The STOP (P) and START (S) bits will toggle based on the START and STOP conditions. Control of the $I^2C$ bus may be taken when the P bit is set, or the bus is idle and both the S and P bits are clear.

In master operation, the SCL and SDA lines are manipulated in software by clearing the corresponding TRISC<4:3> bit(s). The output level is always low, irrespective of the value(s) in PORTC<4:3>. So when transmitting data, a '1' data bit must have the TRISC<4> bit set (input) and a '0' data bit must have the TRISC<4> bit cleared (output). The same scenario is true for the SCL line with the TRISC<3> bit.

The following events will cause SSP Interrupt Flag bit, SSPIF, to be set (SSP Interrupt if enabled):

- START condition
- STOP condition
- Byte transfer completed

Master operation can be done with either the slave mode idle (SSPM3:SSPM0 = 1011) or with the slave active. When both master operation and slave modes are used, the software needs to differentiate the source(s) of the interrupt.

For more information on master operation, see *AN554 - Software Implementation of $I^2C$ Bus Master*.

### 8.3.3 MULTI-MASTER OPERATION

In multi-master operation, the interrupt generation on the detection of the START and STOP conditions allows the determination of when the bus is free. The STOP (P) and START (S) bits are cleared from a reset or when the SSP module is disabled. The STOP (P) and START (S) bits will toggle based on the START and STOP conditions. Control of the $I^2C$ bus may be taken when bit P (SSPSTAT<4>) is set, or the bus is idle and both the S and P bits clear. When the bus is busy, enabling the SSP Interrupt will generate the interrupt when the STOP condition occurs.

In multi-master operation, the SDA line must be monitored to see if the signal level is the expected output level. This check only needs to be done when a high level is output. If a high level is expected and a low level is present, the device needs to release the SDA and SCL lines (set TRISC<4:3>). There are two stages where this arbitration can be lost, these are:

- Address Transfer
- Data Transfer

When the slave logic is enabled, the slave continues to receive. If arbitration was lost during the address transfer stage, communication to the device may be in progress. If addressed, an $\overline{ACK}$ pulse will be generated. If arbitration was lost during the data transfer stage, the device will need to re-transfer the data at a later time.

For more information on master operation, see *AN578 - Use of the SSP Module in the of $I^2C$ Multi-Master Environment*.

**TABLE 8-3 REGISTERS ASSOCIATED WITH $I^2C$ OPERATION**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other resets |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0Bh, 8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | SSPIF | CCP1IF | TMR2IF | TMR1IF | -0-- 0000 | -0-- 0000 |
| 8Ch | PIE1 | — | ADIE | — | — | SSPIE | CCP1IE | TMR2IE | TMR1IE | -0-- 0000 | -0-- 0000 |
| 13h | SSPBUF | Synchronous Serial Port Receive Buffer/Transmit Register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 93h | SSPADD | Synchronous Serial Port ($I^2C$ mode) Address Register | | | | | | | | 0000 0000 | 0000 0000 |
| 14h | SSPCON | WCOL | SSPOV | SSPEN | CKP | SSPM3 | SSPM2 | SSPM1 | SSPM0 | 0000 0000 | 0000 0000 |
| 94h | SSPSTAT | SMP(1) | CKE(1) | D/$\overline{A}$ | P | S | R/$\overline{W}$ | UA | BF | 0000 0000 | 0000 0000 |
| 87h | TRISC | PORTC Data Direction register | | | | | | | | 1111 1111 | 1111 1111 |

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'.
Shaded cells are not used by SSP module in SPI mode.

**Note 1:** Maintain these bits clear in $I^2C$ mode.

**REGISTER 8-1: SSPSTAT: SYNC SERIAL PORT STATUS REGISTER (ADDRESS 94h)**

| R/W-0 | R/W-0 | R-0 | R-0 | R-0 | R-0 | R-0 | R-0 |
|---|---|---|---|---|---|---|---|
| SMP | CKE | D/$\overline{A}$ | P | S | R/$\overline{W}$ | UA | BF |
| bit7 | | | | | | | bit0 |

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

bit 7: **SMP:** SPI data input sample phase
SPI Master Operation
1 = Input data sampled at end of data output time
0 = Input data sampled at middle of data output time
SPI Slave Mode
SMP must be cleared when SPI is used in slave mode
$I^2C$ Mode
This bit must be maintained clear

bit 6: **CKE**: SPI Clock Edge Select
SPI Mode
CKP = 0
1 = Data transmitted on rising edge of SCK
0 = Data transmitted on falling edge of SCK
CKP = 1
1 = Data transmitted on falling edge of SCK
0 = Data transmitted on rising edge of SCK
$I^2C$ Mode
This bit must be maintained clear

bit 5: **D/$\overline{A}$**: Data/$\overline{Address}$ bit ($I^2C$ mode only)
1 = Indicates that the last byte received or transmitted was data
0 = Indicates that the last byte received or transmitted was address

bit 4: **P**: Stop bit ($I^2C$ mode only. This bit is cleared when the SSP module is disabled, or when the Start bit is detected last, SSPEN is cleared)
1 = Indicates that a stop bit has been detected last (this bit is '0' on RESET)
0 = Stop bit was not detected last

bit 3: **S**: Start bit ($I^2C$ mode only. This bit is cleared when the SSP module is disabled, or when the Stop bit is detected last, SSPEN is cleared)
1 = Indicates that a start bit has been detected last (this bit is '0' on RESET)
0 = Start bit was not detected last

bit 2: **R/$\overline{W}$**: Read/Write bit information ($I^2C$ mode only)
This bit holds the R/W bit information following the last address match. This bit is only valid from the address match to the next start bit, stop bit, or $\overline{ACK}$ bit.
1 = Read
0 = Write

bit 1: **UA**: Update Address (10-bit $I^2C$ mode only)
1 = Indicates that the user needs to update the address in the SSPADD register
0 = Address does not need to be updated

bit 0: **BF**: Buffer Full Status bit

Receive (SPI and $I^2C$ modes)
1 = Receive complete, SSPBUF is full
0 = Receive not complete, SSPBUF is empty

Transmit ($I^2C$ mode only)
1 = Transmit in progress, SSPBUF is full
0 = Transmit complete, SSPBUF is empty

 © 1998-2013 Microchip Technology Inc.

**REGISTER 8-2: SSPCON: SYNC SERIAL PORT CONTROL REGISTER (ADDRESS 14h)**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| WCOL | SSPOV | SSPEN | CKP | SSPM3 | SSPM2 | SSPM1 | SSPM0 |
| bit7 | | | | | | | bit0 |

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

bit 7: **WCOL**: Write Collision Detect bit
1 = The SSPBUF register is written while it is still transmitting the previous word (must be cleared in software)
0 = No collision

bit 6: **SSPOV**: Receive Overflow Indicator bit

In SPI mode
1 = A new byte is received while the SSPBUF register is still holding the previous data. In case of overflow, the data in SSPSR is lost. Overflow can only occur in slave mode. The user must read the SSPBUF, even if only transmitting data, to avoid setting overflow. In master operation, the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPBUF register.
0 = No overflow

In $I^2C$ mode
1 = A byte is received while the SSPBUF register is still holding the previous byte. SSPOV is a "don't care" in transmit mode. SSPOV must be cleared in software in either mode.
0 = No overflow

bit 5: **SSPEN**: Synchronous Serial Port Enable bit

In SPI mode
1 = Enables serial port and configures SCK, SDO, and SDI as serial port pins
0 = Disables serial port and configures these pins as I/O port pins

In $I^2C$ mode
1 = Enables the serial port and configures the SDA and SCL pins as serial port pins
0 = Disables serial port and configures these pins as I/O port pins
In both modes, when enabled, these pins must be properly configured as input or output.

bit 4: **CKP**: Clock Polarity Select bit
In SPI mode
1 = Idle state for clock is a high level
0 = Idle state for clock is a low level
In $I^2C$ mode
SCK release control
1 = Enable clock
0 = Holds clock low (clock stretch)

bit 3-0: **SSPM3:SSPM0**: Synchronous Serial Port Mode Select bits
0000 = SPI master operation, clock = FOSC/4
0001 = SPI master operation, clock = FOSC/16
0010 = SPI master operation, clock = FOSC/64
0011 = SPI master operation, clock = TMR2 output/2
0100 = SPI slave mode, clock = SCK pin. $\overline{SS}$ pin control enabled.
0101 = SPI slave mode, clock = SCK pin. $\overline{SS}$ pin control disabled. $\overline{SS}$ can be used as I/O pin
0110 = $I^2C$ slave mode, 7-bit address
0111 = $I^2C$ slave mode, 10-bit address
1011 = $I^2C$ firmware controlled master operation (slave idle)
1110 = $I^2C$ slave mode, 7-bit address with start and stop bit interrupts enabled
1111 = $I^2C$ slave mode, 10-bit address with start and stop bit interrupts enabled

**NOTES:**

 © 1998-2013 Microchip Technology Inc.

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

**Note:** This section applies to the PIC16C72A only.

The analog-to-digital (A/D) converter module has five input channels.

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

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

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

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

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

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

The ADCON0 register, shown in Figure 9-1, controls the operation of the A/D module. The ADCON1 register, shown in Figure 9-2, configures the functions of the port pins. The port pins can be configured as analog inputs (RA3 can also be a voltage reference) or as digital I/O.

**REGISTER 9-1: ADCON0 REGISTER (ADDRESS 1Fh)**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | U-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| ADCS1 | ADCS0 | CHS2 | CHS1 | CHS0 | GO/$\overline{\text{DONE}}$ | — | ADON |
| bit7 | | | | | | | bit0 |

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

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

bit 5-3: **CHS2:CHS0**: Analog Channel Select bits
000 = channel 0, (RA0/AN0)
001 = channel 1, (RA1/AN1)
010 = channel 2, (RA2/AN2)
011 = channel 3, (RA3/AN3)
100 = channel 4, (RA5/AN4)

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

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

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

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

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

| U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | — | — | — | PCFG2 | PCFG1 | PCFG0 |
| bit7 | | | | | | | bit0 |

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

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

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

| PCFG2:PCFG0 | RA0 | RA1 | RA2 | RA5 | RA3 | VREF |
|---|---|---|---|---|---|---|
| 000 | A | A | A | A | A | VDD |
| 001 | A | A | A | A | VREF | RA3 |
| 010 | A | A | A | A | A | VDD |
| 011 | A | A | A | A | VREF | RA3 |
| 100 | A | A | D | D | A | VDD |
| 101 | A | A | D | D | VREF | RA3 |
| 11x | D | D | D | D | D | VDD |

A = Analog input

D = Digital I/O

 © 1998-2013 Microchip Technology Inc.

When the A/D conversion is complete, the result is loaded into the ADRES register, the GO/$\overline{\text{DONE}}$ bit, ADCON0<2>, is cleared, and the A/D interrupt flag bit, ADIF, is set. The block diagram of the A/D module is shown in Figure 9-1.

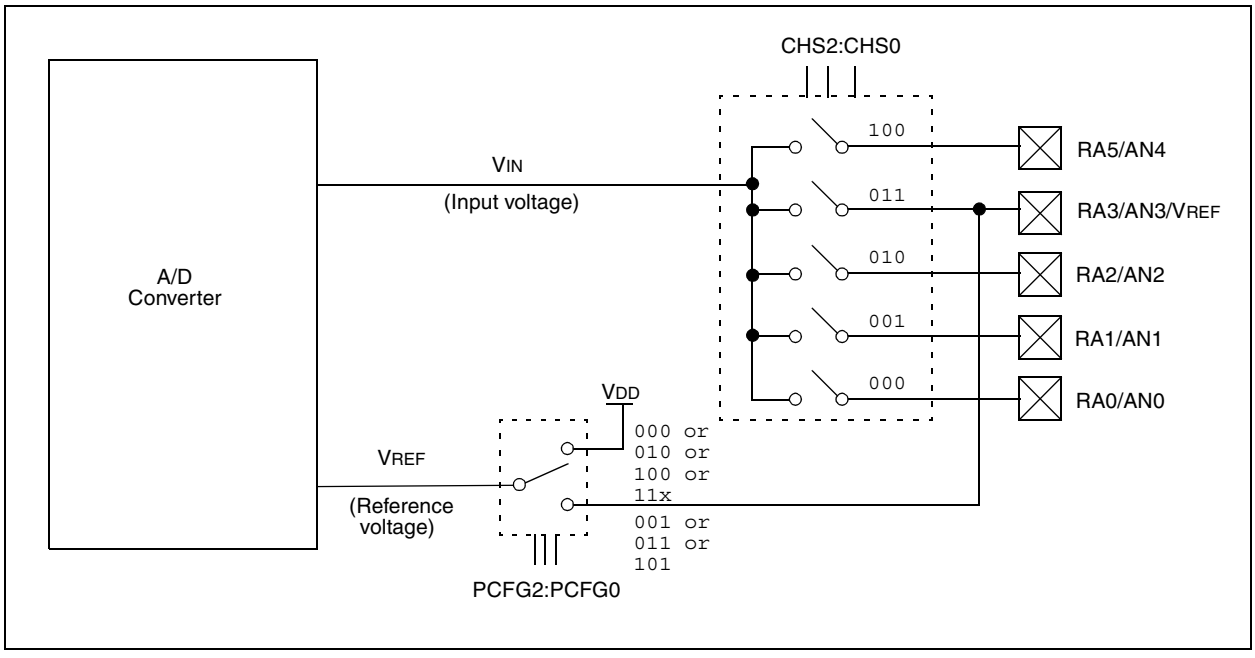
The value that is in the ADRES register is not modified for a Power-on Reset. The ADRES register will contain unknown data after a Power-on Reset.

After the A/D module has been configured as desired, the selected channel must be acquired before the conversion is started. The analog input channels must have their corresponding TRIS bits selected as an input. To determine acquisition time, see Section 9.1. After this acquisition time has elapsed, the A/D conversion can be started. The following steps should be followed for doing an A/D conversion:

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

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

**FIGURE 9-1: A/D BLOCK DIAGRAM**



### 9.1 A/D Acquisition Requirements

For the A/D converter to meet its specified accuracy, the charge holding capacitor (CHOLD) must be allowed to fully charge to the input channel voltage level. The analog input model is shown in Figure 9-2. The source impedance (RS) and the internal sampling switch (RSS) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch (RSS) impedance varies over the device voltage (VDD). The source impedance affects the offset voltage at the analog input (due to pin leakage current). **The maximum recommended impedance for analog sources is 10 kΩ**. After the analog input channel is selected (changed), this acquisition must pass before the conversion can be started.

To calculate the minimum acquisition time, TACQ, see Equation 9-1. This equation calculates the acquisition time to within 1/2 LSb error (512 steps for the A/D). The 1/2 LSb error is the maximum error allowed for the A/D to meet its specified accuracy.

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

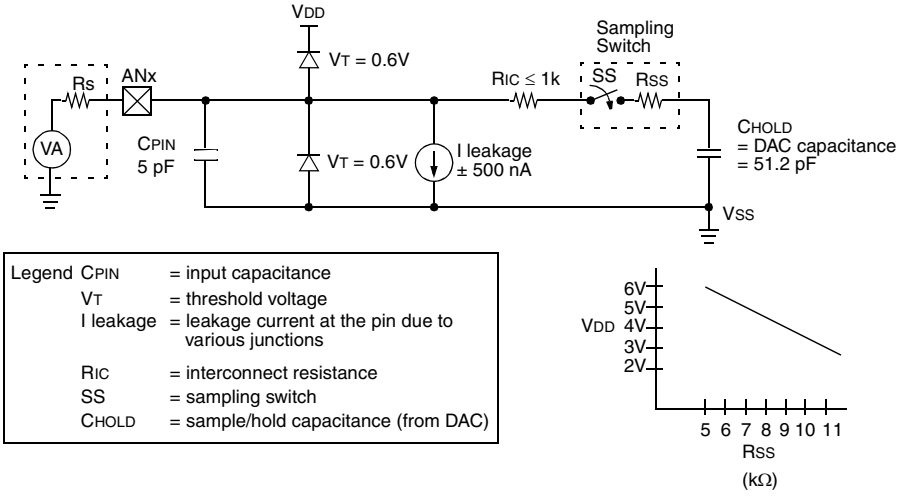
In general;

Assuming RS = 10kΩ
Vdd = 3.0V (RSS = 10kΩ)
Temp. = 50°C (122°F)

TACQ ≈ 13.0 μSec

By increasing VDD and reducing RS and Temp., TACQ can be substantially reduced.

**FIGURE 9-2: ANALOG INPUT MODEL**




**EQUATION 9-1: ACQUISITION TIME**

TACQ = Amplifier Settling Time +
Hold Capacitor Charging Time +
Temperature Coefficient

= TAMP + TC + TCOFF
TAMP = 5μS
TC = - (51.2pF)(1kΩ + RSS + RS) In(1/511)
TCOFF = (Temp -25°C)(0.05μS/°C)

 © 1998-2013 Microchip Technology Inc.

### 9.2 Selecting the A/D Conversion Clock

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

- 2TOSC
- 8TOSC
- 32TOSC
- Internal RC oscillator

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

The A/D module can operate during sleep mode, but the RC oscillator must be selected as the A/D clock source prior to the SLEEP instruction.

Table 9-1 shows the resultant TAD times derived from the device operating frequencies and the A/D clock source selected.

### 9.3 Configuring Analog Port Pins

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

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

**Note 1:** When reading the port register, all pins configured as analog input channels will read as cleared (a low level). Pins configured as digital inputs, will convert an analog input. Analog levels on a digitally configured input will not affect the conversion accuracy.

**Note 2:** Analog levels on any pin that is defined as a digital input (including the AN4:AN0 pins) may cause the input buffer to consume current that is out of the devices specification.

**TABLE 9-1 TAD vs. DEVICE OPERATING FREQUENCIES**

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

Legend: Shaded cells are outside of recommended range.

**Note 1:** The RC source has a typical TAD time of 4 μs.

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

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

**4:** When device frequency is greater than 1 MHz, the RC A/D conversion clock source is recommended for sleep operation only.

**5:** For extended voltage devices (LC), please refer to Electrical Specifications section.

### 9.4 A/D Conversions

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

### 9.5 Use of the CCP Trigger

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

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

**TABLE 9-2 SUMMARY OF A/D REGISTERS**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other Resets |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0Bh,8Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | SSPIF | CCP1IF | TMR2IF | TMR1IF | -0-- 0000 | -0-- 0000 |
| 8Ch | PIE1 | — | ADIE | — | — | SSPIE | CCP1IE | TMR2IE | TMR1IE | -0-- 0000 | -0-- 0000 |
| 1Eh | ADRES | A/D Result Register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 1Fh | ADCON0 | ADCS1 | ADCS0 | CHS2 | CHS1 | CHS0 | GO/$\overline{\text{DONE}}$ | — | ADON | 0000 00-0 | 0000 00-0 |
| 9Fh | ADCON1 | — | — | — | — | — | PCFG2 | PCFG1 | PCFG0 | ---- -000 | ---- -000 |
| 05h | PORTA | — | — | RA5 | RA4 | RA3 | RA2 | RA1 | RA0 | --0x 0000 | --0u 0000 |
| 85h | TRISA | — | — | PORTA Data Direction Register | | | | | | --11 1111 | --11 1111 |

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used for A/D conversion.

 © 1998-2013 Microchip Technology Inc.

## 10.0 SPECIAL FEATURES OF THE CPU

The PIC16C62B/72A devices have a host of features intended to maximize system reliability, minimize cost through elimination of external components, provide power saving operating modes and offer code protection. These are:

- Oscillator Mode Selection
- Reset
  - Power-on Reset (POR)
  - Power-up Timer (PWRT)
  - Oscillator Start-up Timer (OST)
  - Brown-out Reset (BOR)
- Interrupts
- Watchdog Timer (WDT)
- SLEEP
- Code protection
- ID locations
- In-circuit serial programming™ (ICSP)

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

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

Additional information on special features is available in the PIC® MCU Mid-Range Reference Manual, (DS33023).

### 10.1 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.

**FIGURE 10-1: CONFIGURATION WORD**

| CP1 | CP0 | CP1 | CP0 | CP1 | CP0 | — | BODEN | CP1 | CP0 | $\overline{\text{PWRTE}}$ | WDTE | FOSC1 | FOSC0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| bit13 | | | | | | | | | | | | | bit0 |

Register: CONFIG
Address: 2007h

bit 13-8 5-4: **CP1:CP0**: Code Protection bits (2)
- 11 = Code protection off
- 10 = Upper half of program memory code protected
- 01 = Upper 3/4th of program memory code protected
- 00 = All memory is code protected

bit 7: **Unimplemented**: Read as '1'

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

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

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

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

**Note 1:** Enabling Brown-out Reset automatically enables Power-up Timer (PWRT), regardless of the value of bit $\overline{\text{PWRTE}}$.
All of the CP1:CP0 pairs must be given the same value to enable the code protection scheme listed.

## 10.2 Oscillator Configurations

### 10.2.1 OSCILLATOR TYPES

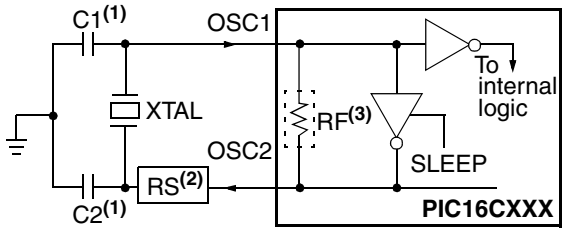
The PIC16CXXX can be operated in four different oscillator modes. The user can program two configuration bits (FOSC1 and FOSC0) to select one of these four modes:

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

### 10.2.2 CRYSTAL OSCILLATOR/CERAMIC RESONATORS

In XT, LP or HS modes, a crystal or ceramic resonator is connected to the OSC1/CLKIN and OSC2/CLKOUT pins to establish oscillation (Figure 10-2). The PIC16CXXX oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may give a frequency out of the crystal manufacturers specifications. When in XT, LP or HS modes, the device can use an external clock source to drive the OSC1/CLKIN pin (Figure 10-3).

**FIGURE 10-2: CRYSTAL/CERAMIC RESONATOR OPERATION (HS, XT OR LP OSC CONFIGURATION)**




**Note1:** See Table 10-1 and Table 10-2 for recommended values of C1 and C2.
**2:** A series resistor (RS) may be required for AT strip cut crystals.
**3:** RF varies with the crystal chosen.

**FIGURE 10-3: EXTERNAL CLOCK INPUT OPERATION (HS, XT OR LP OSC CONFIGURATION)**

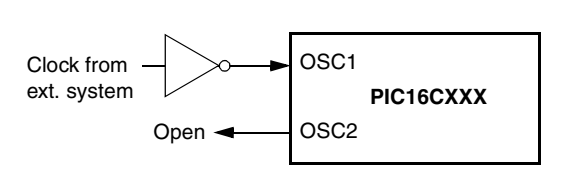



**TABLE 10-1 CERAMIC RESONATORS**

| Ranges Tested: | | | |
|---|---|---|---|
| **Mode** | **Freq** | **OSC1** | **OSC2** |
| XT | 455 kHz<br>2.0 MHz<br>4.0 MHz | 68 - 100 pF<br>15 - 68 pF<br>15 - 68 pF | 68 - 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 |
| **These values are for design guidance only.** See notes at bottom of page. | | | |
| **Resonators Used:** | | | |
| 455 kHz | Panasonic EFO-A455K04B | ± 0.3% | |
| 2.0 MHz | Murata Erie CSA2.00MG | ± 0.5% | |
| 4.0 MHz | Murata Erie CSA4.00MG | ± 0.5% | |
| 8.0 MHz | Murata Erie CSA8.00MT | ± 0.5% | |
| 16.0 MHz | Murata Erie CSA16.00MX | ± 0.5% | |
| Resonators did not have built-in capacitors. | | | |

**TABLE 10-2 CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR**

| Osc Type | Crystal Freq | Cap. Range C1 | Cap. Range C2 |
|---|---|---|---|
| LP | 32 kHz | 33 pF | 33 pF |
| | 200 kHz | 15 pF | 15 pF |
| XT | 200 kHz | 47-68 pF | 47-68 pF |
| | 1 MHz | 15 pF | 15 pF |
| | 4 MHz | 15 pF | 15 pF |
| HS | 4 MHz | 15 pF | 15 pF |
| | 8 MHz | 15-33 pF | 15-33 pF |
| | 20 MHz | 15-33 pF | 15-33 pF |
| **These values are for design guidance only.** See notes at bottom of page. | | | |
| **Crystals Used** | | | |
| 32 kHz | Epson C-001R32.768K-A | ± 20 PPM | |
| 200 kHz | STD XTL 200.000KHz | ± 20 PPM | |
| 1 MHz | ECS ECS-10-13-1 | ± 50 PPM | |
| 4 MHz | ECS ECS-40-20-1 | ± 50 PPM | |
| 8 MHz | EPSON CA-301 8.000M-C | ± 30 PPM | |
| 20 MHz | EPSON CA-301 20.000M-C | ± 30 PPM | |

**Note 1:** Higher capacitance increases the stability of the oscillator, but also increases the start-up time.
**2:** Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components.
**3:** Rs may be required in HS mode, as well as XT mode, to avoid overdriving crystals with low drive level specification.
**4:** Oscillator performance should be verified when migrating between devices (including PIC16C62A to PIC16C62B and PIC16C72 to PIC16C72A)

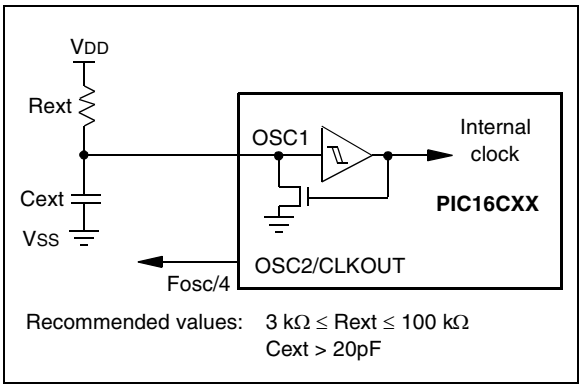
PRELIMINARY

 © 1998-2013 Microchip Technology Inc.

### 10.2.3 RC OSCILLATOR

For timing insensitive applications, the "RC" device option offers additional cost savings. The RC oscillator frequency is a function of the supply voltage, the resistor (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 10-4 shows how the R/C combination is connected to the PIC16CXXX.

**FIGURE 10-4: RC OSCILLATOR MODE**




## 10.3 Reset

The PIC16CXXX differentiates between various kinds of reset:

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

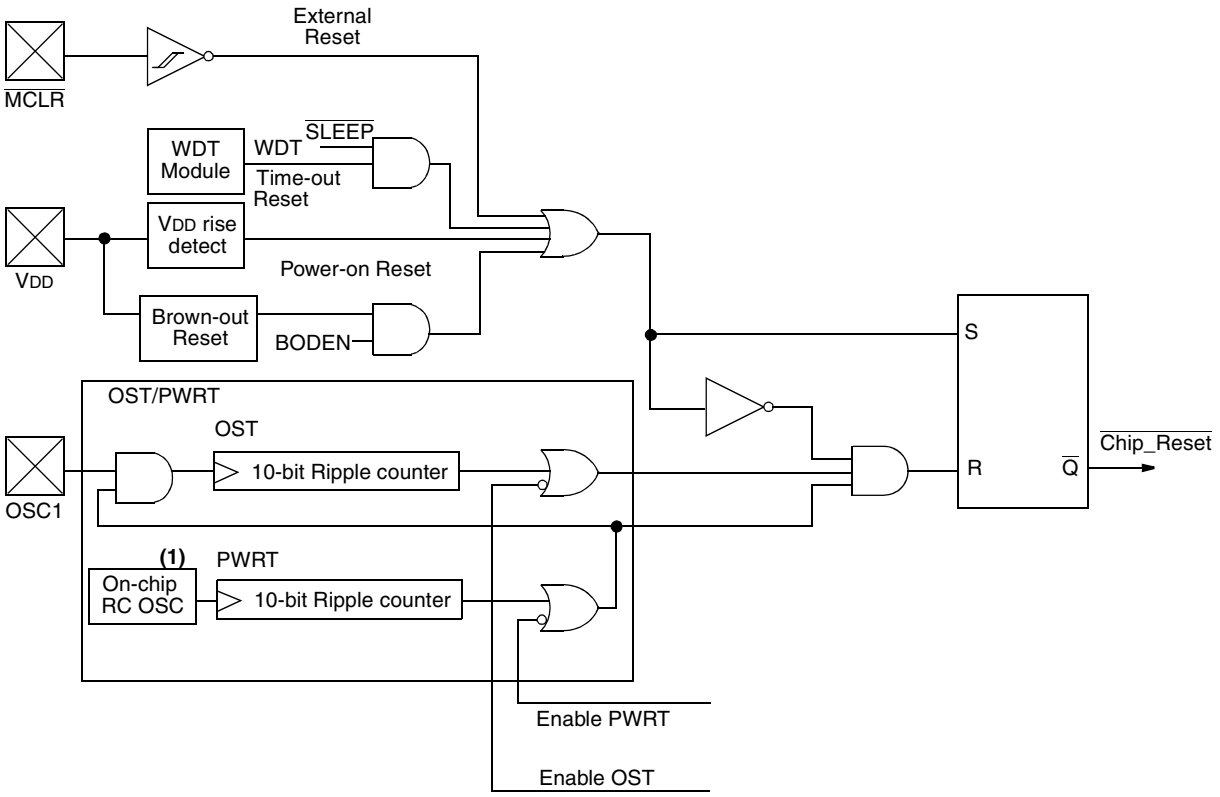
Some registers are not affected in any reset condition; their status is unknown on POR and unchanged by any other reset. Most other registers are reset to a "reset state" on Power-on Reset (POR), on the $\overline{\text{MCLR}}$ and WDT Reset, on $\overline{\text{MCLR}}$ reset during SLEEP, and on Brown-out Reset (BOR). They are not affected by a WDT Wake-up from SLEEP, which is viewed as the resumption of normal operation. The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits are set or cleared depending on the reset situation, as indicated in Table 10-4. These bits are used in software to determine the nature of the reset. See Table 10-6 for a full description of reset states of all registers.

A simplified block diagram of the on-chip reset circuit is shown in Figure 10-5.

The PIC devices have a $\overline{\text{MCLR}}$ noise filter in the $\overline{\text{MCLR}}$ reset path. The filter will ignore small pulses. However, a valid $\overline{\text{MCLR}}$ pulse must meet the minimum pulse width (TmcL, Specification #30).

No internal reset source (WDT, BOR, POR) willdrive the $\overline{\text{MCLR}}$ pin low.

**FIGURE 10-5: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT**




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

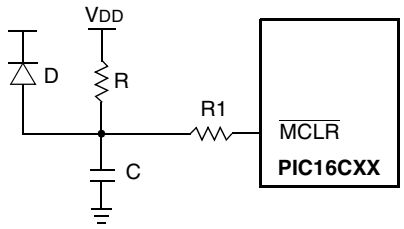
 © 1998-2013 Microchip Technology Inc.

### 10.4 Power-On Reset (POR)

A Power-on Reset pulse is generated on-chip when VDD rise is detected (in the range of 1.5V - 2.1V). To take advantage of the POR, just tie the MCLR pin directly (or through a resistor) to VDD. This will eliminate external RC components usually needed to create a Power-on Reset. A maximum rise time for VDD is specified (SVDD, parameter D004). For a slow rise time, see Figure 10-6.

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

**FIGURE 10-6: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW VDD POWER-UP)**




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

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

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

### 10.5 Power-up Timer (PWRT)

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

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

### 10.6 Oscillator Start-up Timer (OST)

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

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

**Note:** The OST delay may not occur when the device wakes from SLEEP.

### 10.7 Brown-Out Reset (BOR)

The configuration bit, BODEN, can enable or disable the Brown-Out Reset circuit. If VPP falls below Vbor (parameter #35, about 100μS), the brown-out situation will reset the device. If VDD falls below VBOR for less than TBOR, a reset may not occur.

Once the brown-out occurs, the device will remain in brown-out reset until VDD rises above VBOR. The power-up timer then keeps the device in reset for TPWRT (parameter #33, about 72mS). If VDD should fall below VBOR during TPWRT, the brown-out reset process will restart when VDD rises above VBOR with the power-up timer reset. The power-up timer is always enabled when the brown-out reset circuit is enabled, regardless of the state of the PWRT configuration bit.

### 10.8 Time-out Sequence

When a POR reset occurs, the PWRT delay starts (if enabled). When PWRT ends, the OST counts 1024 oscillator cycles (LP, XT, HS modes only). When OST completes, the device comes out of reset. The total time-out will vary based on oscillator configuration and the status of the PWRT. For example, in RC mode with the PWRT disabled, there will be no time-out at all.

If $\overline{MCLR}$ is kept low long enough, the time-outs will expire. Bringing $\overline{MCLR}$ high will begin execution immediately. This is useful for testing purposes or to synchronize more than one PIC16CXXX device operating in parallel.

Table 10-5 shows the reset conditions for the STATUS, PCON and PC registers, while Table 10-6 shows the reset conditions for all the registers.

### 10.9 Power Control/Status Register (PCON)

The $\overline{BOR}$ bit is unknown on Power-on Reset. If the Brown-out Reset circuit is used, the $\overline{BOR}$ bit must be set by the user and checked on subsequent resets to see if it was cleared, indicating a Brown-out has occurred.

$\overline{POR}$ (Power-on Reset Status bit) is cleared on a Power-on Reset and unaffected otherwise. The user

**Status Register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| IRP | RP1 | RP0 | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C |

**PCON Register**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | | $\overline{POR}$ | $\overline{BOR}$ |

**TABLE 10-3 TIME-OUT IN VARIOUS SITUATIONS**

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

**TABLE 10-4 STATUS BITS AND THEIR SIGNIFICANCE**

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

**TABLE 10-5 RESET 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 1uuu | ---- --uu |
| WDT Wake-up | PC + 1 | uuu0 0uuu | ---- --uu |
| Brown-out Reset | 000h | 0001 1uuu | ---- --u0 |
| Interrupt wake-up from SLEEP | PC + 1(1) | uuu1 0uuu | ---- --uu |

Legend: u = unchanged, x = unknown, - = unimplemented bit read as '0'.

**Note 1:** When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

 © 1998-2013 Microchip Technology Inc.

**TABLE 10-6 INITIALIZATION CONDITIONS FOR ALL REGISTERS**

| Register | Applicable Devices | | Power-on Reset, Brown-out Reset | $\overline{\text{MCLR}}$ Resets WDT Reset | Wake-up via WDT or Interrupt |
|---|---|---|---|---|---|
| W | 62B | 72A | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| INDF | 62B | 72A | N/A | N/A | N/A |
| TMR0 | 62B | 72A | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PCL | 62B | 72A | 0000h | 0000h | PC + 1(2) |
| STATUS | 62B | 72A | 0001 1xxx | 000q quuu(3) | uuuq quuu(3) |
| FSR | 62B | 72A | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PORTA(4) | 62B | 72A | --0x 0000 | --0u 0000 | --uu uuuu |
| PORTB(5) | 62B | 72A | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PORTC(5) | 62B | 72A | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PCLATH | 62B | 72A | ---0 0000 | ---0 0000 | ---u uuuu |
| INTCON | 62B | 72A | 0000 000x | 0000 000u | uuuu uuuu(1) |
| PIR1 | 62B | 72A | ---- 0000 | ---- 0000 | ---- uuuu(1) |
| | 62B | 72A | -0-- 0000 | -0-- 0000 | -u-- uuuu(1) |
| TMR1L | 62B | 72A | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| TMR1H | 62B | 72A | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| T1CON | 62B | 72A | --00 0000 | --uu uuuu | --uu uuuu |
| TMR2 | 62B | 72A | 0000 0000 | 0000 0000 | uuuu uuuu |
| T2CON | 62B | 72A | -000 0000 | -000 0000 | -uuu uuuu |
| SSPBUF | 62B | 72A | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| SSPCON | 62B | 72A | 0000 0000 | 0000 0000 | uuuu uuuu |
| CCPR1L | 62B | 72A | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| CCPR1H | 62B | 72A | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| CCP1CON | 62B | 72A | --00 0000 | --00 0000 | --uu uuuu |
| ADRES | 62B | 72A | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| ADCON0 | 62B | 72A | 0000 00-0 | 0000 00-0 | uuuu uu-u |
| OPTION_REG | 62B | 72A | 1111 1111 | 1111 1111 | uuuu uuuu |
| TRISA | 62B | 72A | --11 1111 | --11 1111 | --uu uuuu |
| TRISB | 62B | 72A | 1111 1111 | 1111 1111 | uuuu uuuu |
| TRISC | 62B | 72A | 1111 1111 | 1111 1111 | uuuu uuuu |
| PIE1 | 62B | 72A | ---- 0000 | ---- 0000 | ---- uuuu |
| | 62B | 72A | -0-- 0000 | -0-- 0000 | -u-- uuuu |
| PCON | 62B | 72A | ---- --0q | ---- --uq | ---- --uq |
| PR2 | 62B | 72A | 1111 1111 | 1111 1111 | 1111 1111 |
| SSPADD | 62B | 72A | 0000 0000 | 0000 0000 | uuuu uuuu |
| SSPSTAT | 62B | 72A | 0000 0000 | 0000 0000 | uuuu uuuu |
| ADCON1 | 62B | 72A | ---- -000 | ---- -000 | ---- -uuu |

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

**Note 1:** One or more bits in INTCON and/or PIR1 will be affected (to cause wake-up).

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

**3:** See Table 10-5 for reset value for specific condition.

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

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

### 10.10 Interrupts

The interrupt control register (INTCON) records individual interrupt requests in flag bits. It also has individual and global interrupt enable bits.

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

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

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

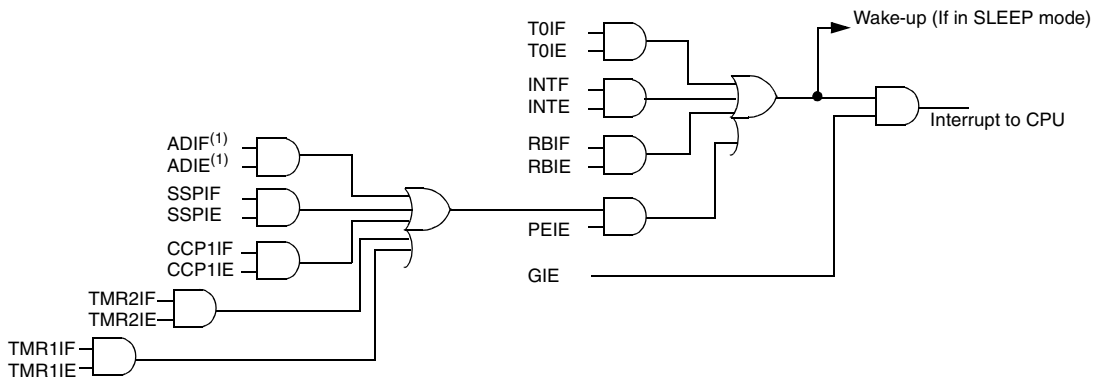
The RB0/INT pin interrupt, the RB port change interrupt and the TMR0 overflow interrupt flags are contained in the INTCON register.

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

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

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

**FIGURE 10-7: INTERRUPT LOGIC**




**Note 1:** The A/D module is not implemented on the PIC16C62B.

 © 1998-2013 Microchip Technology Inc.

10.10.1 INT INTERRUPT

The external interrupt on RB0/INT pin is edge triggered: either rising, if bit INTEDG (OPTION_REG<6>) is set, or falling, if the INTEDG bit is clear. When a valid edge appears on the RB0/INT pin, flag bit INTF (INTCON<1>) is set. This interrupt can be disabled by clearing enable bit INTE (INTCON<4>). Flag bit INTF must be cleared in software in the interrupt service routine before re-enabling this interrupt. The INT interrupt can wake-up the processor from SLEEP, if bit INTE was set prior to going into SLEEP. The status of global interrupt enable bit GIE decides whether or not the processor branches to the interrupt vector following wake-up. See Section 10.13 for details on SLEEP mode.

10.10.2 TMR0 INTERRUPT

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

10.10.3 PORTB INTCON CHANGE

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

## 10.11 Context Saving During Interrupts

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

Example 10-1 stores and restores the W and STATUS registers. The register, W_TEMP, must be defined in each bank and must be defined at the same offset from the bank base address (i.e., if W_TEMP is defined at 0x20 in bank 0, it must also be defined at 0xA0 in bank 1).

The example:

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

**EXAMPLE 10-1: SAVING STATUS, W, AND PCLATH REGISTERS IN RAM**

```
MOVWF    W_TEMP           ;Copy W to TEMP register, could be bank one or zero
SWAPF    STATUS,W         ;Swap status to be saved into W
CLRF     STATUS           ;bank 0, regardless of current bank, Clears IRP,RP1,RP0
MOVWF    STATUS_TEMP      ;Save status to bank zero STATUS_TEMP register
:
:(ISR)
:
SWAPF    STATUS_TEMP,W    ;Swap STATUS_TEMP register into W
                          ;(sets bank to original state)
MOVWF    STATUS           ;Move W into STATUS register
SWAPF    W_TEMP,F         ;Swap W_TEMP
SWAPF    W_TEMP,W         ;Swap W_TEMP into W
```

### 10.12 Watchdog Timer (WDT)

The Watchdog Timer is a free running on-chip RC oscillator which does not require any external components. This RC oscillator is separate from the RC oscillator of the OSC1/CLKIN pin. The WDT will run, even if the clock on the OSC1/CLKIN and OSC2/CLKOUT 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 (Watchdog Timer Reset). If the device is in SLEEP mode, a WDT time-out causes the device to wake-up and continue with normal operation (Watchdog Timer Wake-up). The $\overline{TO}$ bit in the STATUS register will be cleared upon a Watchdog Timer time-out.

The WDT can be permanently disabled by clearing configuration bit WDTE (Section 10.1).

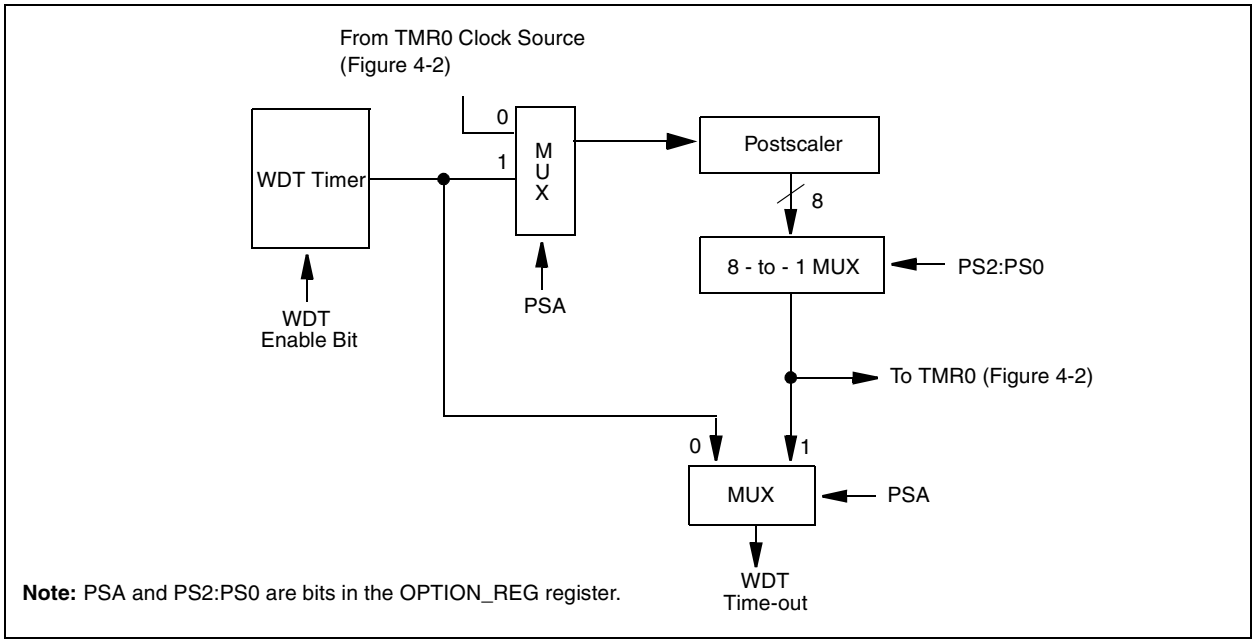
The WDT time-out period (TWDT, parameter #31) is multiplied by the prescaler ratio, when the prescaler is assigned to the WDT. The prescaler assignment (assigned to either the WDT or Timer0) and prescaler ratio are set in the OPTION_REG register.

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

.

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

**FIGURE 10-8: WATCHDOG TIMER BLOCK DIAGRAM**




**Note:** PSA and PS2:PS0 are bits in the OPTION_REG register.

**FIGURE 10-9: SUMMARY OF WATCHDOG TIMER REGISTERS**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---|---|---|---|---|---|---|---|---|---|
| 2007h | Config. bits | | BODEN | CP1 | CP0 | $\overline{PWRTE}$ | WDTE | FOSC1 | FOSC0 |
| 81h | OPTION_REG | $\overline{RBPU}$ | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 |

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

 © 1998-2013 Microchip Technology Inc.

## 10.13 Power-down Mode (SLEEP)

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

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

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

The $\overline{MCLR}$ pin must be at a logic high level (VIHMC, parameter D042).

### 10.13.1 WAKE-UP FROM SLEEP

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

1. External reset input on $\overline{MCLR}$ pin.
2. Watchdog Timer Wake-up (if WDT was enabled).
3. Interrupt from INT pin, RB port change, or some Peripheral Interrupts.

External $\overline{MCLR}$ Reset will cause a device reset. All other events are considered a continuation of program execution and cause a "wake-up". The $\overline{TO}$ and $\overline{PD}$ bits in the STATUS register can be used to determine the cause of device reset. The $\overline{PD}$ bit, which is set on power-up, is cleared when SLEEP is invoked. The $\overline{TO}$ bit is cleared if a WDT time-out occurred (and caused wake-up).

The following peripheral interrupts can wake the device from SLEEP:

1. TMR1 interrupt. Timer1 must be operating as an asynchronous counter.
2. CCP capture mode interrupt.
3. Special event trigger (Timer1 in asynchronous mode using an external clock. CCP1 is in compare mode).
4. SSP (Start/Stop) bit detect interrupt.
5. SSP transmit or receive in slave mode (SPI/$I^2C$).
6. USART RX or TX (synchronous slave mode).

Other peripherals cannot generate interrupts since during SLEEP, no on-chip clocks are present.

When the SLEEP instruction is being executed, the next instruction (PC + 1) is pre-fetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be set (enabled). Wake-up is regardless of the state of the GIE bit. If the GIE bit is clear (disabled), the device resumes 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, a NOP should follow the SLEEP instruction.

### 10.13.2 WAKE-UP USING INTERRUPTS

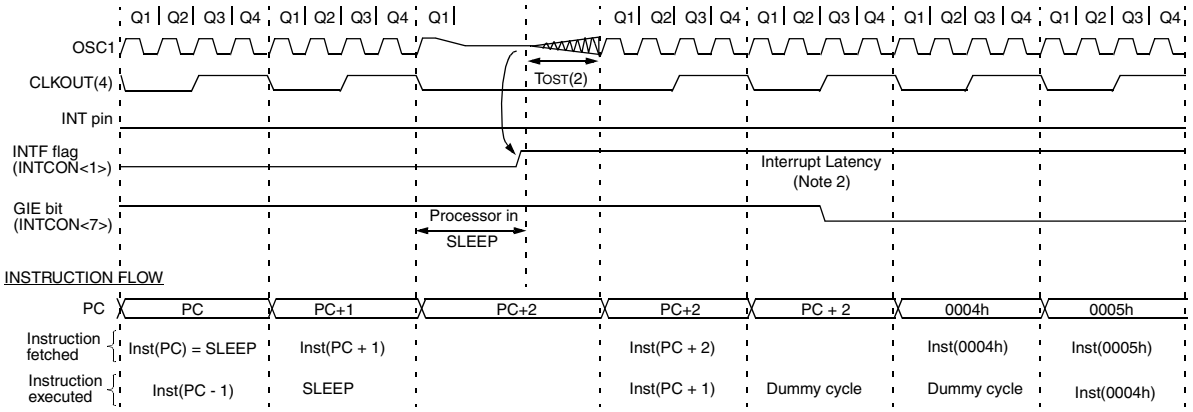
When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the execution of a SLEEP instruction, the SLEEP instruction will complete as a NOP. Therefore, the WDT and WDT postscaler will not be cleared, the $\overline{TO}$ bit will not be set and $\overline{PD}$ bits will not be cleared.
- If the interrupt occurs **during or after** the execution of a SLEEP instruction, the device will immediately wake up from sleep. The SLEEP instruction will be completely executed before the wake-up. Therefore, the WDT and WDT postscaler will be cleared, the $\overline{TO}$ bit will be set and the $\overline{PD}$ bit will be cleared.

Even if the flag bits were checked before executing a SLEEP instruction, it may be possible for flag bits to become set before the SLEEP instruction completes. To determine whether a SLEEP instruction executed, test the $\overline{PD}$ bit. If the $\overline{PD}$ bit is set, the SLEEP instruction was executed as a NOP.

To ensure that the WDT is cleared, a CLRWDT instruction should be executed before a SLEEP instruction.

**FIGURE 10-10: 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.

## 10.14 Program Verification/Code Protection

If the code protection bits 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. |
|---|---|

## 10.15 ID Locations

Four memory locations (2000h - 2003h) are designated as ID locations where the user can store checksum or other code-identification numbers. These locations are not accessible during normal execution, but are readable and writable during program/verify. It is recommended that only the 4 least significant bits of the ID location are used.

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

## 10.16 In-Circuit Serial Programming™

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

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

 © 1998-2013 Microchip Technology Inc.

## 11.0 INSTRUCTION SET SUMMARY

Each PIC16CXXX instruction is a 14-bit word divided into an OPCODE which specifies the instruction type and one or more operands which further specify the operation of the instruction. The PIC16CXX instruction set summary in Table 11-2 lists **byte-oriented**, **bit-oriented**, and **literal and control** operations. Table 11-1 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 11-1 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 |
| PC | Program Counter |
| $\overline{TO}$ | Time-out bit |
| $\overline{PD}$ | Power-down bit |
| Z | Zero bit |
| DC | Digit Carry bit |
| C | Carry bit |

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

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

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

Table 11-2 lists the instructions recognized by the MPASM assembler.

Figure 11-1 shows the general formats that the instructions can have.

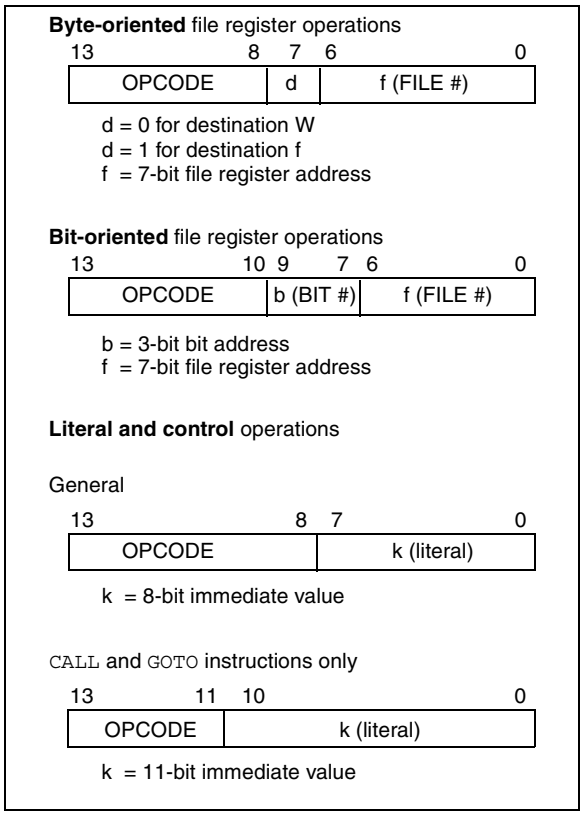
**Note:** To maintain upward compatibility with future PIC16CXXX products, do not use the OPTION and TRIS instructions.

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

0xhh

where h signifies a hexadecimal digit.

**FIGURE 11-1: GENERAL FORMAT FOR INSTRUCTIONS**




A description of each instruction is available in the PIC® MCU Mid-Range Reference Manual, (DS33023).

### TABLE 11-2 PIC16CXXX INSTRUCTION SET

| Mnemonic, Operands | | Description | Cycles | 14-Bit Opcode MSb | | | LSb | Status Affected | Notes |
|---|---|---|---|---|---|---|---|---|---|
| **BYTE-ORIENTED FILE REGISTER OPERATIONS** | | | | | | | | | |
| **ADDWF** | **f, d** | Add W and f | 1 | 00 | 0111 | dfff | ffff | C,DC,Z | 1,2 |
| **ANDWF** | **f, d** | AND W with f | 1 | 00 | 0101 | dfff | ffff | Z | 1,2 |
| **CLRF** | **f** | Clear f | 1 | 00 | 0001 | 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.

 © 1998-2013 Microchip Technology Inc.

## 11.1 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 |
| Description: | The contents of the W register are added to the eight bit literal 'k' and the result is placed in the W register. |

| ADDWF | Add W and f |
|---|---|
| Syntax: | [*label*] ADDWF f,d |
| Operands: | $0 \le f \le 127$<br>$d \in [0,1]$ |
| Operation: | (W) + (f) → (destination) |
| Status Affected: | C, DC, Z |
| Description: | Add the contents of the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'. |

| ANDLW | AND Literal with W |
|---|---|
| Syntax: | [*label*] ANDLW k |
| Operands: | $0 \le k \le 255$ |
| Operation: | (W) .AND. (k) → (W) |
| Status Affected: | Z |
| Description: | The contents of W register are AND'ed with the eight bit literal 'k'. The result is placed in the W register. |

| ANDWF | AND W with f |
|---|---|
| Syntax: | [*label*] ANDWF f,d |
| Operands: | $0 \le f \le 127$<br>$d \in [0,1]$ |
| Operation: | (W) .AND. (f) → (destination) |
| Status Affected: | Z |
| Description: | AND the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'. |

| BCF | Bit Clear f |
|---|---|
| Syntax: | [*label*] BCF f,b |
| Operands: | $0 \le f \le 127$<br>$0 \le b \le 7$ |
| Operation: | 0 → (f<b>) |
| Status Affected: | None |
| Description: | Bit 'b' in register 'f' is cleared. |

| BSF | Bit Set f |
|---|---|
| Syntax: | [*label*] BSF f,b |
| Operands: | $0 \le f \le 127$<br>$0 \le b \le 7$ |
| Operation: | 1 → (f<b>) |
| Status Affected: | None |
| Description: | Bit 'b' in register 'f' is set. |

| BTFSS | Bit Test f, Skip if Set |
|---|---|
| Syntax: | [*label*] BTFSS f,b |
| Operands: | 0 ≤ f ≤ 127<br>0 ≤ b < 7 |
| Operation: | skip if (f<b>) = 1 |
| Status Affected: | None |
| Description: | If bit 'b' in register 'f' is '0', then the next instruction is executed.<br>If bit 'b' is '1', then the next instruction is discarded and a NOP is executed instead, making this a 2TCY instruction. |

| BTFSC | Bit Test, Skip if Clear |
|---|---|
| Syntax: | [*label*] BTFSC f,b |
| Operands: | 0 ≤ f ≤ 127<br>0 ≤ b ≤ 7 |
| Operation: | skip if (f<b>) = 0 |
| Status Affected: | None |
| Description: | If bit 'b' in register 'f' is '1', then the next instruction is executed.<br>If bit 'b' in register 'f' is '0', then the next instruction is discarded, and a NOP is executed instead, making this a 2TCY instruction. |

| CALL | Call Subroutine |
|---|---|
| Syntax: | [ *label* ] CALL k |
| Operands: | 0 ≤ k ≤ 2047 |
| Operation: | (PC)+ 1→ TOS,<br>k → PC<10:0>,<br>(PCLATH<4:3>) → PC<12:11> |
| Status Affected: | None |
| Description: | Call Subroutine. First, return address (PC+1) is pushed onto the stack. The eleven bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a two cycle instruction. |

| CLRF | Clear f |
|---|---|
| Syntax: | [*label*] CLRF f |
| Operands: | 0 ≤ f ≤ 127 |
| Operation: | 00h → (f)<br>1 → Z |
| Status Affected: | Z |
| Description: | The contents of register 'f' are cleared and the Z bit is set. |

| CLRW | Clear W |
|---|---|
| Syntax: | [ *label* ] CLRW |
| Operands: | None |
| Operation: | 00h → (W)<br>1 → Z |
| Status Affected: | Z |
| Description: | W register is cleared. Zero bit (Z) is set. |

| CLRWDT | Clear Watchdog Timer |
|---|---|
| Syntax: | [ *label* ] CLRWDT |
| Operands: | None |
| Operation: | 00h → WDT<br>0 → WDT prescaler,<br>1 → $\overline{TO}$<br>1 → $\overline{PD}$ |
| Status Affected: | $\overline{TO}$, $\overline{PD}$ |
| Description: | CLRWDT instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set. |

 © 1998-2013 Microchip Technology Inc.

| COMF | Complement f |
|---|---|
| Syntax: | [ *label* ] COMF f,d |
| Operands: | $0 \le f \le 127$<br>$d \in [0,1]$ |
| Operation: | $(\bar{f}) \rightarrow$ (destination) |
| Status Affected: | Z |
| 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'. |

| GOTO | Unconditional Branch |
|---|---|
| Syntax: | [ *label* ] GOTO k |
| Operands: | $0 \le k \le 2047$ |
| Operation: | $k \rightarrow$ PC<10:0><br>PCLATH<4:3> $\rightarrow$ PC<12:11> |
| Status Affected: | None |
| Description: | GOTO is an unconditional branch. The eleven bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. GOTO is a two cycle instruction. |

| DECF | Decrement f |
|---|---|
| Syntax: | [*label*] DECF f,d |
| Operands: | $0 \le f \le 127$<br>$d \in [0,1]$ |
| Operation: | (f) - 1 $\rightarrow$ (destination) |
| Status Affected: | Z |
| Description: | Decrement register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'. |

| INCF | Increment f |
|---|---|
| Syntax: | [ *label* ] INCF f,d |
| Operands: | $0 \le f \le 127$<br>$d \in [0,1]$ |
| Operation: | (f) + 1 $\rightarrow$ (destination) |
| Status Affected: | Z |
| Description: | The contents of register 'f' are incremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'. |

| DECFSZ | Decrement f, Skip if 0 |
|---|---|
| Syntax: | [ *label* ] DECFSZ f,d |
| Operands: | $0 \le f \le 127$<br>$d \in [0,1]$ |
| Operation: | (f) - 1 $\rightarrow$ (destination);<br>skip if result = 0 |
| Status Affected: | None |
| Description: | The contents of register 'f' are decremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'.<br>If the result is 1, the next instruction, is executed. If the result is 0, then a NOP is executed instead making it a 2TCY instruction. |

| INCFSZ | Increment f, Skip if 0 |
|---|---|
| Syntax: | [ *label* ] INCFSZ f,d |
| Operands: | $0 \le f \le 127$<br>$d \in [0,1]$ |
| Operation: | (f) + 1 $\rightarrow$ (destination),<br>skip if result = 0 |
| Status Affected: | None |
| Description: | The contents of register 'f' are incremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'.<br>If the result is 1, the next instruction is executed. If the result is 0, a NOP is executed instead making it a 2TCY instruction. |

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

| IORWF | Inclusive OR W with f |
|---|---|
| Syntax: | [ *label* ] IORWF f,d |
| Operands: | $0 \le f \le 127$<br>$d \in [0,1]$ |
| Operation: | (W) .OR. (f) → (destination) |
| Status Affected: | Z |
| Description: | Inclusive OR the W register with register 'f'. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'. |

| MOVF | Move f |
|---|---|
| Syntax: | [ *label* ] MOVF f,d |
| Operands: | $0 \le f \le 127$<br>$d \in [0,1]$ |
| Operation: | (f) → (destination) |
| Status Affected: | Z |
| Description: | The contents of register f is moved to a destination dependant upon the status of d. If d = 0, destination is W register. If d = 1, the destination is file register f itself. d = 1 is useful to test a file register since status flag Z is affected. |

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

| MOVWF | Move W to f |
|---|---|
| Syntax: | [ *label* ] MOVWF f |
| Operands: | $0 \le f \le 127$ |
| Operation: | (W) → (f) |
| Status Affected: | None |
| Description: | Move data from W register to register 'f'. |

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

 © 1998-2013 Microchip Technology Inc.

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

Syntax: [ *label* ] RETFIE

Operands: None

Operation: TOS → PC,
1 → GIE

Status Affected: None

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

Syntax: [ *label* ] RETLW k

Operands: 0 ≤ k ≤ 255

Operation: k → (W);
TOS → PC

Status Affected: None

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.

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

Syntax: [ *label* ] RETURN

Operands: None

Operation: TOS → PC

Status Affected: None

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.

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

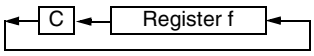
Syntax: [ *label* ] RLF f,d

Operands: 0 ≤ f ≤ 127
d ∈ [0,1]

Operation: See description below

Status Affected: C

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'.




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

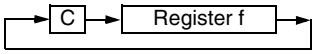
Syntax: [ *label* ] RRF f,d

Operands: 0 ≤ f ≤ 127
d ∈ [0,1]

Operation: See description below

Status Affected: C

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'.




**SLEEP**

Syntax: [ *label* ] SLEEP

Operands: None

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

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

Description: The power-down status bit, $\overline{PD}$ is cleared. Time-out status bit, $\overline{TO}$ is set. Watchdog Timer and its prescaler are cleared.
The processor is put into SLEEP mode with the oscillator stopped. See Section 10.13 for more details.

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

Syntax: [ *label* ] SUBLW k

Operands: $0 \le k \le 255$

Operation: k - (W) → (W)

Status Affected: C, DC, Z

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

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

Syntax: [ *label* ] SUBWF f,d

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

Operation: (f) - (W) → (destination)

Status Affected: C, DC, Z

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'.

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

Syntax: [ *label* ] SWAPF f,d

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

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

Status Affected: None

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

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

Syntax: [*label*] XORLW k

Operands: $0 \le k \le 255$

Operation: (W) .XOR. k → (W)

Status Affected: Z

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

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

Syntax: [*label*] XORWF f,d

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

Operation: (W) .XOR. (f) → (destination)

Status Affected: Z

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

 © 1998-2013 Microchip Technology Inc.

## 12.0 DEVELOPMENT SUPPORT

The PIC® 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/MPLIB Linker/Librarian
- Simulators
  - MPLAB-SIM Software Simulator
- Emulators
  - MPLAB-ICE Real-Time In-Circuit Emulator
  - PICMASTER®/PICMASTER-CE In-Circuit Emulator
  - ICEPIC™
- In-Circuit Debugger
  - MPLAB-ICD for PIC16F877
- Device Programmers
  - PRO MATE® II Universal Programmer
  - PICSTART® Plus Entry-Level Prototype Programmer
- Low-Cost Demonstration Boards
  - SIMICE
  - PICDEM-1
  - PICDEM-2
  - PICDEM-3
  - PICDEM-17
  - SEEVAL®
  - KEELOQ®

### 12.1 MPLAB Integrated Development Environment Software

  - The MPLAB IDE software brings an ease of software development previously unseen in the 8-bit microcontroller market. MPLAB is a Windows®-based application which contains:
- Multiple functionality
  - editor
  - simulator
  - programmer (sold separately)
  - emulator (sold separately)
- A full featured editor
- A project manager
- Customizable tool bar and key mapping
- A status bar
- On-line help

MPLAB allows you to:

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

The ability to use MPLAB with Microchip's simulator, MPLAB-SIM, allows a consistent platform and the ability to easily switch from the cost-effective simulator to the full featured emulator with minimal retraining.

### 12.2 MPASM Assembler

MPASM is a full featured universal macro assembler for all PIC MCUs. It can produce absolute code directly in the form of HEX files for device programmers, or it can generate relocatable objects for MPLINK.

MPASM has a command line interface and a Windows shell and can be used as a standalone application on a Windows 3.x or greater system. MPASM generates relocatable object files, Intel standard HEX files, MAP files to detail memory usage and symbol reference, an absolute LST file which contains source lines and generated machine code, and a COD file for MPLAB debugging.

MPASM features include:

- MPASM and MPLINK are integrated into MPLAB projects.
- MPASM allows user defined macros to be created for streamlined assembly.
- MPASM allows conditional assembly for multi purpose source files.
- MPASM directives allow complete control over the assembly process.

### 12.3 MPLAB-C17 and MPLAB-C18 C Compilers

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

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

### 12.4 MPLINK/MPLIB Linker/Librarian

MPLINK is a relocatable linker for MPASM and MPLAB-C17 and MPLAB-C18. It can link relocatable objects from assembly or C source files along with precompiled libraries using directives from a linker script.

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

MPLINK features include:

- MPLINK works with MPASM and MPLAB-C17 and MPLAB-C18.
- MPLINK allows all memory areas to be defined as sections to provide link-time flexibility.

MPLIB features include:

- MPLIB makes linking easier because single libraries can be included instead of many smaller files.
- MPLIB helps keep code maintainable by grouping related modules together.
- MPLIB commands allow libraries to be created and modules to be added, listed, replaced, deleted, or extracted.

### 12.5 MPLAB-SIM Software Simulator

The MPLAB-SIM Software Simulator allows code development in a PC host environment by simulating the PIC series microcontrollers on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a file or user-defined key press to any of the pins. The execution can be performed in single step, execute until break, or trace mode.

MPLAB-SIM fully supports symbolic debugging using MPLAB-C17 and MPLAB-C18 and MPASM. The Software Simulator offers the flexibility to develop and debug code outside of the laboratory environment making it an excellent multi-project software development tool.

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

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

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

The MPLAB-ICE Emulator System has been designed as a real-time emulation system with advanced features that are generally found on more expensive development tools. The PC platform and Microsoft® Windows 3.x/95/98 environment were chosen to best make these features available to you, the end user.

MPLAB-ICE 2000 is a full-featured emulator system with enhanced trace, trigger, and data monitoring features. Both systems use the same processor modules and will operate across the full operating speed range of the PIC MCU.

### 12.7 PICMASTER/PICMASTER CE

The PICMASTER system from Microchip Technology is a full-featured, professional quality emulator system. This flexible in-circuit emulator provides a high-quality, universal platform for emulating Microchip 8-bit PIC microcontrollers (MCUs). PICMASTER systems are sold worldwide, with a CE compliant model available for European Union (EU) countries.

### 12.8 ICEPIC

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

### 12.9 MPLAB-ICD In-Circuit Debugger

Microchip's In-Circuit Debugger, MPLAB-ICD, is a powerful, low-cost run-time development tool. This tool is based on the flash PIC16F877 and can be used to develop for this and other PIC microcontrollers from the PIC16CXXX family. MPLAB-ICD utilizes the In-Circuit Debugging capability built into the PIC16F87X. This feature, along with Microchip's In-Circuit Serial Programming protocol, offers cost-effective in-circuit flash programming and debugging from the graphical user interface of the MPLAB Integrated Development Environment. This enables a designer to develop and debug source code by watching variables, single-stepping and setting break points. Running at full speed enables testing hardware in real-time. The MPLAB-ICD is also a programmer for the flash PIC16F87X family.

### 12.10 PRO MATE II Universal Programmer

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

The PRO MATE II has programmable $V_{DD}$ and $V_{PP}$ supplies which allows it to verify programmed memory at $V_{DD}$ min and $V_{DD}$ max for maximum reliability. It has an LCD display for instructions and error messages, keys to enter commands and a modular detachable socket assembly to support various package types. In

 © 1998-2013 Microchip Technology Inc.

stand-alone mode the PRO MATE II can read, verify or program PIC devices. It can also set code-protect bits in this mode.

### 12.11 PICSTART Plus Entry Level Development System

The PICSTART programmer is an easy-to-use, low-cost prototype programmer. It connects to the PC via one of the COM (RS-232) ports. MPLAB Integrated Development Environment software makes using the programmer simple and efficient.

PICSTART Plus supports all PIC devices with up to 40 pins. Larger pin count devices such as the PIC16C92X, and PIC17C76X may be supported with an adapter socket. PICSTART Plus is CE compliant.

### 12.12 SIMICE Entry-Level Hardware Simulator

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

### 12.13 PICDEM-1 Low-Cost PIC MCU Demonstration Board

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

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

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

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

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

### 12.16 PICDEM-17

The PICDEM-17 is an evaluation board that demonstrates the capabilities of several Microchip microcontrollers, including PIC17C752, PIC17C756, PIC17C762, and PIC17C766. All necessary hardware is included to run basic demo programs, which are supplied on a 3.5-inch disk. A programmed sample is included, and the user may erase it and program it with the other sample programs using the PRO MATE II or PICSTART Plus device programmers and easily debug

and test the sample code. In addition, PICDEM-17 supports down-loading of programs to and executing out of external FLASH memory on board. The PICDEM-17 is also usable with the MPLAB-ICE or PICMASTER emulator, and all of the sample programs can be run and modified using either emulator. Additionally, a generous prototype area is available for user hardware.

### 12.17 SEEVAL Evaluation and Programming System

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

### 12.18 KEELOQ Evaluation and Programming Tools

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

 © 1998-2013 Microchip Technology Inc.

**TABLE 12-1: DEVELOPMENT TOOLS FROM MICROCHIP**

| | | PIC12CXXX | PIC14000 | PIC16C5X | PIC16C6X | PIC16CXXX | PIC16F62X | PIC16C7X | PIC16C7XX | PIC16C8X | PIC16F8XX | PIC16C9XX | PIC17C4X | PIC17C7XX | PIC18CXX2 | 24CXX/ 25CXX/ 93CXX | HCSXXX | MCRFXXX | MCP2510 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Software Tools | MPLAB™ Integrated Development Environment | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | |
| | MPLAB™ C17 Compiler | | | | | | | | | | | | ✓ | ✓ | | | | | |
| | MPLAB™ C18 Compiler | | | | | | | | | | | | | | ✓ | | | | |
| | MPASM/MPLINK | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | |
| Emulators | MPLAB™-ICE | ✓ | ✓ | ✓ | ✓ | ✓ | ✓** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | |
| | PICMASTER/PICMASTER-CE | ✓ | ✓ | ✓ | ✓ | ✓ | | ✓ | ✓ | ✓ | | ✓ | ✓ | ✓ | | | | | |
| | ICEPIC™ Low-Cost In-Circuit Emulator | ✓ | | ✓ | ✓ | ✓ | | ✓ | ✓ | ✓ | | ✓ | | | | | | | |
| Debugger | MPLAB-ICD In-Circuit Debugger | | | | ✓* | | | ✓* | | | ✓ | | | | | | | | |
| Programmers | PICSTART®Plus Low-Cost Universal Dev. Kit | ✓ | ✓ | ✓ | ✓ | ✓ | ✓** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | |
| | PRO MATE® II Universal Programmer | ✓ | ✓ | ✓ | ✓ | ✓ | ✓** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | |
| Demo Boards and Eval Kits | SIMICE | ✓ | | ✓ | | | | | | | | | | | | | | | |
| | PICDEM-1 | | | ✓ | | ✓ | | ✓† | | ✓ | | | ✓ | | | | | | |
| | PICDEM-2 | | | | ✓† | | | ✓† | | | | | | | ✓ | | | | |
| | PICDEM-3 | | | | | | | | | | | ✓ | | | | | | | |
| | PICDEM-14A | | ✓ | | | | | | | | | | | | | | | | |
| | PICDEM-17 | | | | | | | | | | | | | ✓ | | | | | |
| | KEELOQ® Evaluation Kit | | | | | | | | | | | | | | | | ✓ | | |
| | KEELOQ Transponder Kit | | | | | | | | | | | | | | | | ✓ | | |
| | microID™ Programmer's Kit | | | | | | | | | | | | | | | | | ✓ | |
| | 125 kHz microID Developer's Kit | | | | | | | | | | | | | | | | | ✓ | |
| | 125 kHz Anticollision microID Developer's Kit | | | | | | | | | | | | | | | | | ✓ | |
| | 13.56 MHz Anticollision microID Developer's Kit | | | | | | | | | | | | | | | | | ✓ | |
| | MCP2510 CAN Developer's Kit | | | | | | | | | | | | | | | | | | ✓ |

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

** Contact Microchip Technology Inc. for availability date.

† Development tool is available on select devices.

**NOTES:**

 © 1998-2013 Microchip Technology Inc.

## 13.0 ELECTRICAL CHARACTERISTICS

**Absolute Maximum Ratings** (†)

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

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

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

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

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

Voltage on RA4 with respect to Vss ..... 0V to +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 (combined) ..... 200 mA

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

Maximum current sunk by PORTC ..... 200 mA

Maximum current sourced by PORTC ..... 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 $\overline{MCLR}/V_{PP}$ pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100Ω should be used when applying a "low" level to the $\overline{MCLR}/V_{PP}$ pin, rather than pulling this pin directly to $V_{SS}$.

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

**FIGURE 13-1: PIC16C62B/72A-20 VOLTAGE-FREQUENCY GRAPH**

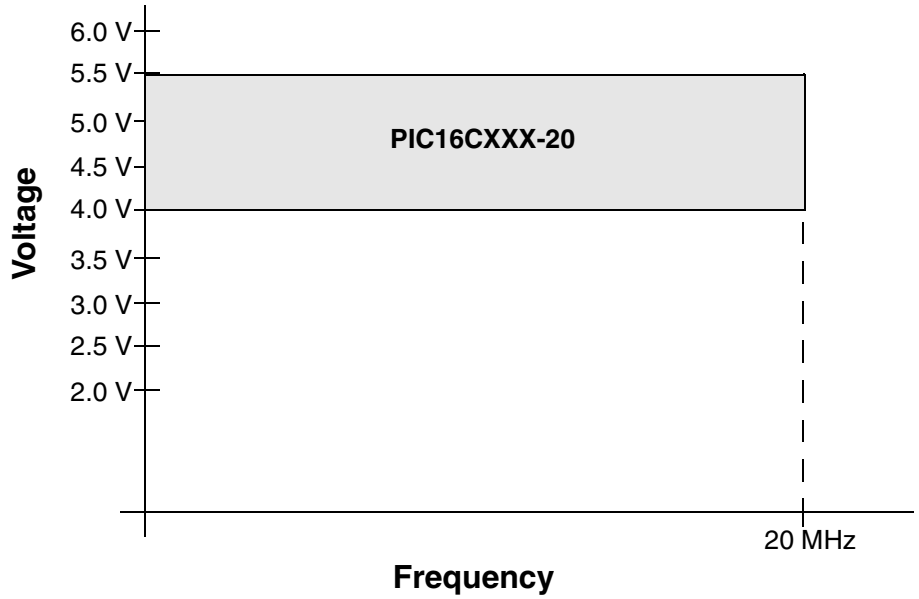
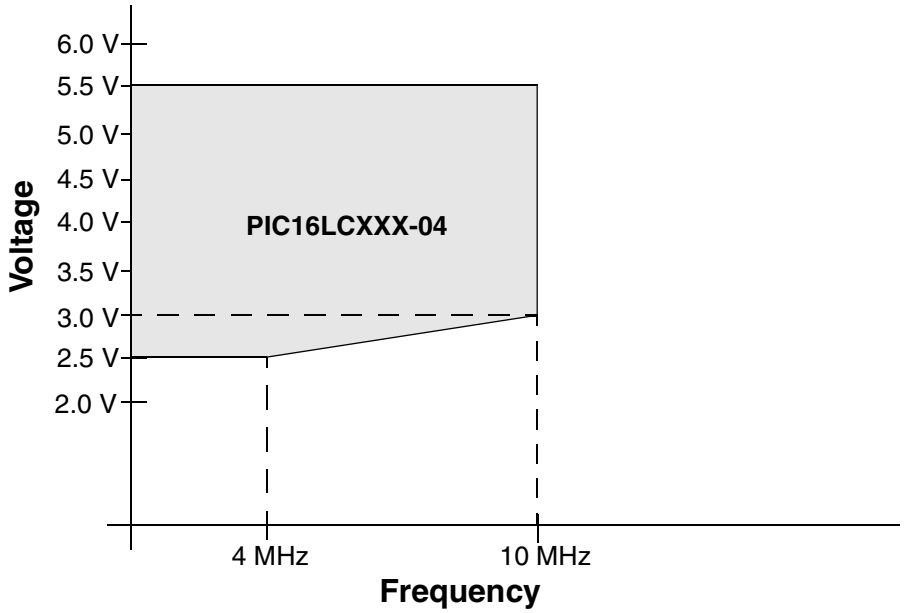



**FIGURE 13-2: PIC16LC62B/72A AND PIC16C62B/72A/JW VOLTAGE-FREQUENCY GRAPH**

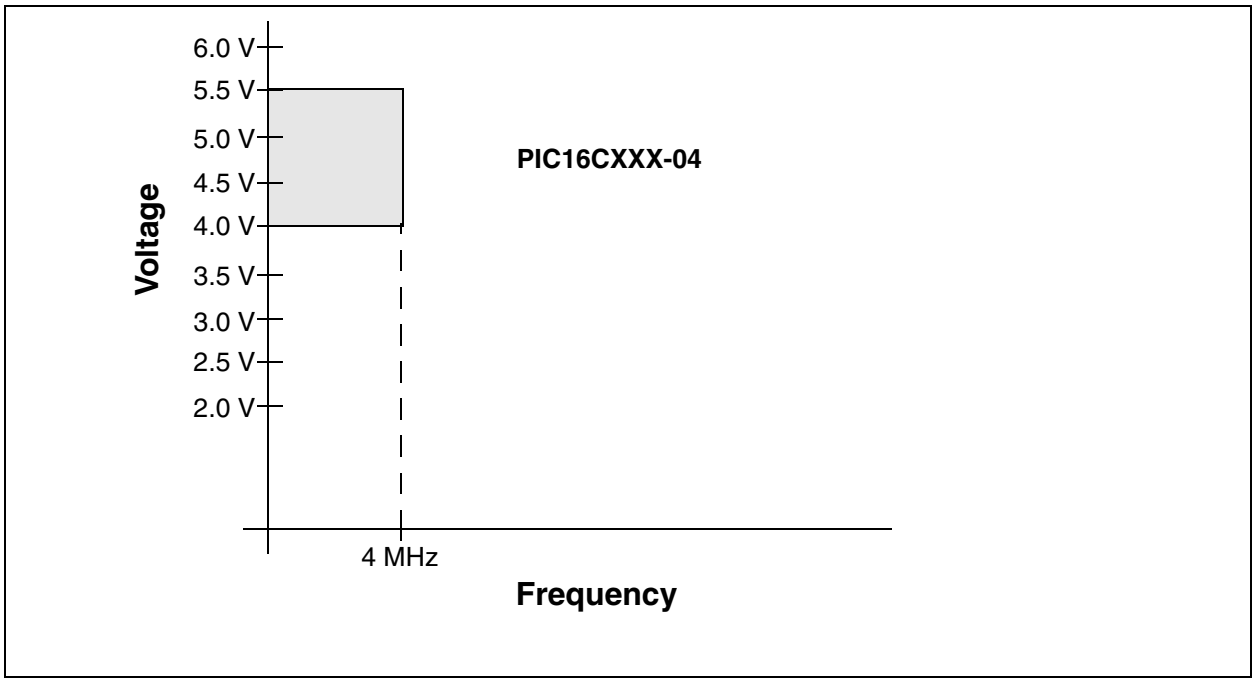



FMAX = (12.0 MHz/V) (VDDAPPMIN - 2.5 V) + 4 MHz

**Note:** VDDAPPMIN is the minimum voltage of the PIC® device in the application. Fmax is no greater than 10 MHz.

 © 1998-2013 Microchip Technology Inc.

**FIGURE 13-3: PIC16C62B/72A-04 VOLTAGE-FREQUENCY GRAPH**



### 13.1 DC Characteristics: PIC16C62B/72A-04 (Commercial, Industrial, Extended) PIC16C62B/72A-20 (Commercial, Industrial, Extended)

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature 0°C ≤ TA ≤ +70°C for commercial<br>-40°C ≤ TA ≤ +85°C for industrial<br>-40°C ≤ TA ≤+125°C for extended | | | | |
|---|---|---|---|---|---|---|---|
| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| D001 | VDD | Supply Voltage | 4.0 | - | 5.5 | V | XT, RC and LP osc mode |
| D001A | | | 4.5 | - | 5.5 | V | HS osc mode |
| | | | VBOR* | - | 5.5 | V | BOR enabled (Note 7) |
| D002* | VDR | RAM Data Retention **Voltage** (Note 1) | - | 1.5 | - | V | |
| D003 | VPOR | **VDD Start Voltage** to ensure internal Power-on Reset signal | - | VSS | - | V | See section on Power-on Reset for details |
| D004* | SVDD | **VDD Rise Rate** to ensure internal Power-on Reset signal | 0.05 | - | - | V/ms | PWRT enabled ($\overline{\text{PWRTE}}$ bit clear) |
| D004A* | | | TBD | - | - | | PWRT disabled ($\overline{\text{PWRTE}}$ bit set) See section on Power-on Reset for details |
| D005 | VBOR | Brown-out Reset voltage trip point | 3.65 | - | 4.35 | V | BODEN bit set |
| D010 | IDD | **Supply Current** (Note 2, 5) | - | 2.7 | 5 | mA | XT, RC osc modes FOSC = 4 MHz, VDD = 5.5V (Note 4) |
| D013 | | | - | 10 | 20 | mA | HS osc mode FOSC = 20 MHz, VDD = 5.5V |
| D020 | IPD | **Power-down Current** (Note 3, 5) | - | 10.5 | 42 | µA | VDD = 4.0V, WDT enabled,-40°C to +85°C |
| | | | - | 1.5 | 16 | µA | VDD = 4.0V, WDT disabled, 0°C to +70°C |
| D021 | | | - | 1.5 | 19 | µA | VDD = 4.0V, WDT disabled,-40°C to +85°C |
| D021B | | | - | 2.5 | 19 | µA | VDD = 4.0V, WDT disabled,-40°C to +125°C |
| | | Module Differential Current (Note 6) | | | | | |
| D022* | ΔIWDT | Watchdog Timer | - | 6.0 | 20 | µA | WDTE BIT SET, VDD = 4.0V |
| D022A* | ΔIBOR | Brown-out Reset | - | TBD | 200 | µA | BODEN bit set, VDD = 5.0V |

* These parameters are characterized but not tested.

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

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

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

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

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

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

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

**7:** This is the voltage where the device enters the Brown-out Reset. When BOR is enabled, the device will perform a brown-out reset when VDD falls below VBOR.

 © 1998-2013 Microchip Technology Inc.

### 13.2 DC Characteristics: PIC16LC62B/72A-04 (Commercial, Industrial)

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

* These parameters are characterized but not tested.

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

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

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

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

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

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

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

**7:** This is the voltage where the device enters the Brown-out Reset. When BOR is enabled, the device will perform a brown-out reset when VDD falls below VBOR.

### 13.3 DC Characteristics: PIC16C62B/72A-04 (Commercial, Industrial, Extended) PIC16C62B/72A-20 (Commercial, Industrial, Extended) PIC16LC62B/72A-04 (Commercial, Industrial)

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature 0°C ≤ TA ≤ +70°C for commercial<br>-40°C ≤ TA ≤ +85°C for industrial<br>-40°C ≤ TA ≤+125°C for extended<br>Operating voltage VDD range as described in DC spec Section 13.1 and Section 13.2 | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| | | **Input Low Voltage** | | | | | |
| | VIL | I/O ports | | | | | |
| D030 | | with TTL buffer | VSS | - | 0.15VDD | V | For entire VDD range |
| D030A | | | Vss | - | 0.8V | V | 4.5V ≤ VDD ≤ 5.5V |
| D031 | | with Schmitt Trigger buffer | VSS | - | 0.2VDD | V | |
| D032 | | $\overline{MCLR}$, OSC1 (in RC mode) | Vss | - | 0.2VDD | V | |
| D033 | | OSC1 (in XT, HS and LP modes) | Vss | - | 0.3VDD | V | Note1 |
| | | **Input High Voltage** | | | | | |
| | VIH | I/O ports | | - | | | |
| D040 | | with TTL buffer | 2.0 | - | VDD | V | 4.5V ≤ VDD ≤ 5.5V |
| D040A | | | 0.25VDD + 0.8V | - | Vdd | V | For entire VDD range |
| D041 | | with Schmitt Trigger buffer | 0.8VDD | - | VDD | V | For entire VDD range |
| D042 | | MCLR | 0.8VDD | - | VDD | V | |
| D042A | | OSC1 (XT, HS and LP modes) | 0.7VDD | - | VDD | V | Note1 |
| D043 | | OSC1 (in RC mode) | 0.9VDD | - | Vdd | V | |
| | | **Input Leakage Current** (Notes 2, 3) | | | | | |
| D060 | IIL | I/O ports | - | - | ±1 | µA | Vss ≤ VPIN ≤ VDD, Pin at hi-impedance |
| D061 | | $\overline{MCLR}$, RA4/T0CKI | - | - | ±5 | µA | Vss ≤ VPIN ≤ VDD |
| D063 | | OSC1 | - | - | ±5 | µA | Vss ≤ VPIN ≤ VDD, XT, HS and LP osc modes |
| D070 | IPURB | **PORTB weak pull-up current** | 50 | 250 | 400 | µA | VDD = 5V, VPIN = VSS |
| | | **Output Low Voltage** | | | | | |
| D080 | VOL | I/O ports | - | - | 0.6 | V | IOL = 8.5 mA, VDD = 4.5V, -40°C to +85°C |

* These parameters are characterized but not tested.

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

**Note 1:** In RC oscillator mode, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the device be driven with external clock in RC mode.

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

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

 © 1998-2013 Microchip Technology Inc.

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature 0°C ≤ TA ≤ +70°C for commercial<br>-40°C ≤ TA ≤ +85°C for industrial<br>-40°C ≤ TA ≤+125°C for extended<br>Operating voltage VDD range as described in DC spec Section 13.1 and Section 13.2 | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| D083 | | OSC2/CLKOUT (RC osc mode) | - | - | 0.6 | V | IOL = 7.0 mA, VDD = 4.5V, -40°C to +125°C |
| | | | - | - | 0.6 | V | IOL = 1.6 mA, VDD = 4.5V, -40°C to +85°C |
| | | | - | - | 0.6 | V | IOL = 1.2 mA, VDD = 4.5V, -40°C to +125°C |
| | | **Output High Voltage** | | | | | |
| D090 | VOH | I/O ports (Note 3) | VDD-0.7 | - | - | V | IOH = -3.0 mA, VDD = 4.5V, -40°C to +85°C |
| | | | VDD-0.7 | - | - | V | IOH = -2.5 mA, VDD = 4.5V, -40°C to +125°C |
| D092 | | OSC2/CLKOUT (RC osc mode) | VDD-0.7 | - | - | V | IOH = -1.3 mA, VDD = 4.5V, -40°C to +85°C |
| | | | VDD-0.7 | - | - | V | IOH = -1.0 mA, VDD = 4.5V, -40°C to +125°C |
| D150* | VOD | **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 is used to drive OSC1. |
| D101 | CIO | All I/O pins and OSC2 (in RC mode) | - | - | 50 | pF | |
| D102 | Cb | SCL, SDA in $I^2C$ mode | - | - | 400 | pF | |

* These parameters are characterized but not tested.

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

**Note 1:** In RC oscillator mode, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the device be driven with external clock in RC mode.

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

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

## 13.4 AC (Timing) Characteristics

### 13.4.1 TIMING PARAMETER SYMBOLOGY

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

1. TppS2ppS
2. TppS
3. TCC:ST ($I^2C$ specifications only)
4. Ts ($I^2C$ specifications only)

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

Lowercase letters (pp) and their meanings:

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

Uppercase letters and their meanings:

| | | | |
|---|---|---|---|
| **S** | | | |
| F | Fall | P | Period |
| H | High | R | Rise |
| I | Invalid (Hi-impedance) | V | Valid |
| L | Low | Z | Hi-impedance |
| **$I^2C$ only** | | | |
| AA | output access | High | High |
| BUF | Bus free | Low | Low |

TCC:ST ($I^2C$ specifications only)

| | | | |
|---|---|---|---|
| **CC** | | | |
| HD | Hold | SU | Setup |
| **ST** | | | |
| DAT | DATA input hold | STO | STOP condition |
| STA | START condition | | |

 © 1998-2013 Microchip Technology Inc.

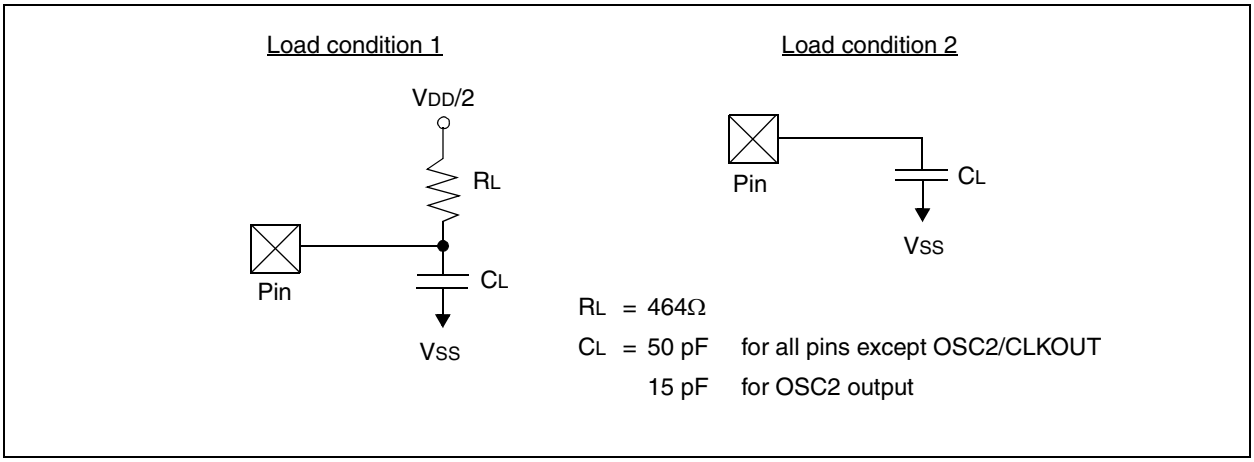
13.4.2 TIMING CONDITIONS

The temperature and voltages specified in Table 13-1 apply to all timing specifications unless otherwise noted. Figure 13-4 specifies the load conditions for the timing specifications.

**TABLE 13-1: TEMPERATURE AND VOLTAGE SPECIFICATIONS - AC**

| | | | | |
|---|---|---|---|---|
| AC CHARACTERISTICS | Standard Operating Conditions (unless otherwise stated) | | | |
| | Operating temperature | 0°C | ≤ TA ≤ +70°C | for commercial |
| | | -40°C | ≤ TA ≤ +85°C | for industrial |
| | | -40°C | ≤ TA ≤+125°C | for extended |
| | Operating voltage VDD range as described in DC spec Section 13.1 and Section 13.2. LC parts operate for commercial/industrial temp's only. | | | |

**FIGURE 13-4: LOAD CONDITIONS FOR DEVICE TIMING SPECIFICATIONS**



13.4.3 TIMING DIAGRAMS AND SPECIFICATIONS

**FIGURE 13-5: EXTERNAL CLOCK TIMING**

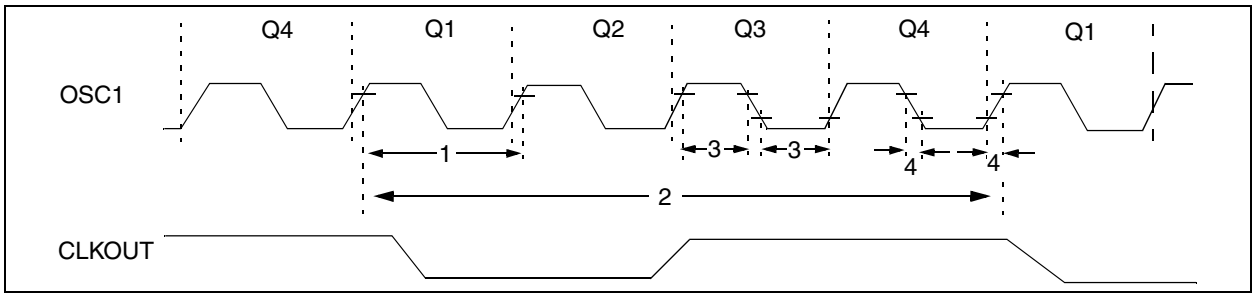



**TABLE 13-2: EXTERNAL CLOCK TIMING REQUIREMENTS**

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

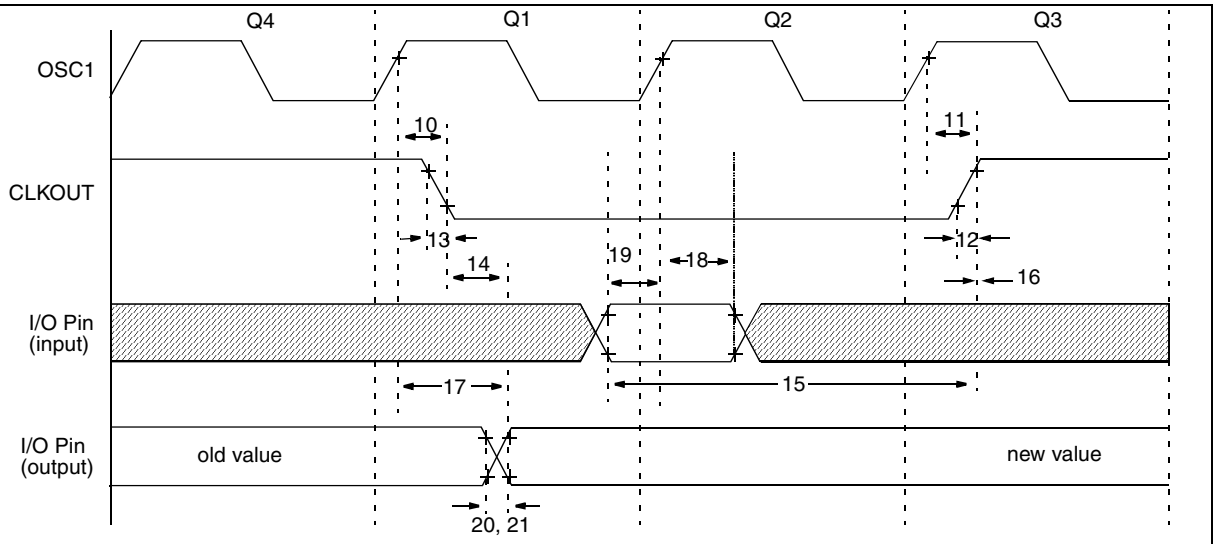
* These parameters are characterized but not tested.

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

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

 © 1998-2013 Microchip Technology Inc.

**FIGURE 13-6: CLKOUT AND I/O TIMING**




**Note:** Refer to Figure 13-4 for load conditions.

**TABLE 13-3: CLKOUT AND I/O TIMING REQUIREMENTS**

| Param No. | Sym | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 10* | TosH2ckL | OSC1↑ to CLKOUT↓ | | — | 75 | 200 | ns | Note 1 |
| 11* | TosH2ckH | OSC1↑ to CLKOUT↑ | | — | 75 | 200 | ns | Note 1 |
| 12* | TckR | CLKOUT rise time | | — | 35 | 100 | ns | Note 1 |
| 13* | TckF | CLKOUT fall time | | — | 35 | 100 | ns | Note 1 |
| 14* | TckL2ioV | CLKOUT ↓ to Port out valid | | — | — | 0.5TCY + 20 | ns | Note 1 |
| 15* | TioV2ckH | Port in valid before CLKOUT ↑ | | Tosc + 200 | — | — | ns | Note 1 |
| 16* | TckH2ioI | Port in hold after CLKOUT ↑ | | 0 | — | — | ns | Note 1 |
| 17* | TosH2ioV | OSC1↑ (Q1 cycle) to Port out valid | | — | 50 | 150 | ns | |
| 18* | TosH2ioI | OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time) | **PIC16CXX** | 100 | — | — | ns | |
| 18A* | | | **PIC16LCXX** | 200 | — | — | ns | |
| 19* | TioV2osH | Port input valid to OSC1↑ (I/O in setup time) | | 0 | — | — | ns | |
| 20* | TioR | Port output rise time | **PIC16CXX** | — | 10 | 40 | ns | |
| 20A* | | | **PIC16LCXX** | — | — | 80 | ns | |
| 21* | TioF | Port output fall time | **PIC16CXX** | — | 10 | 40 | ns | |
| 21A* | | | **PIC16LCXX** | — | — | 80 | ns | |
| 22††* | Tinp | INT pin high or low time | | TCY | — | — | ns | |
| 23††* | Trbp | RB7:RB4 change INT high or low time | | TCY | — | — | ns | |

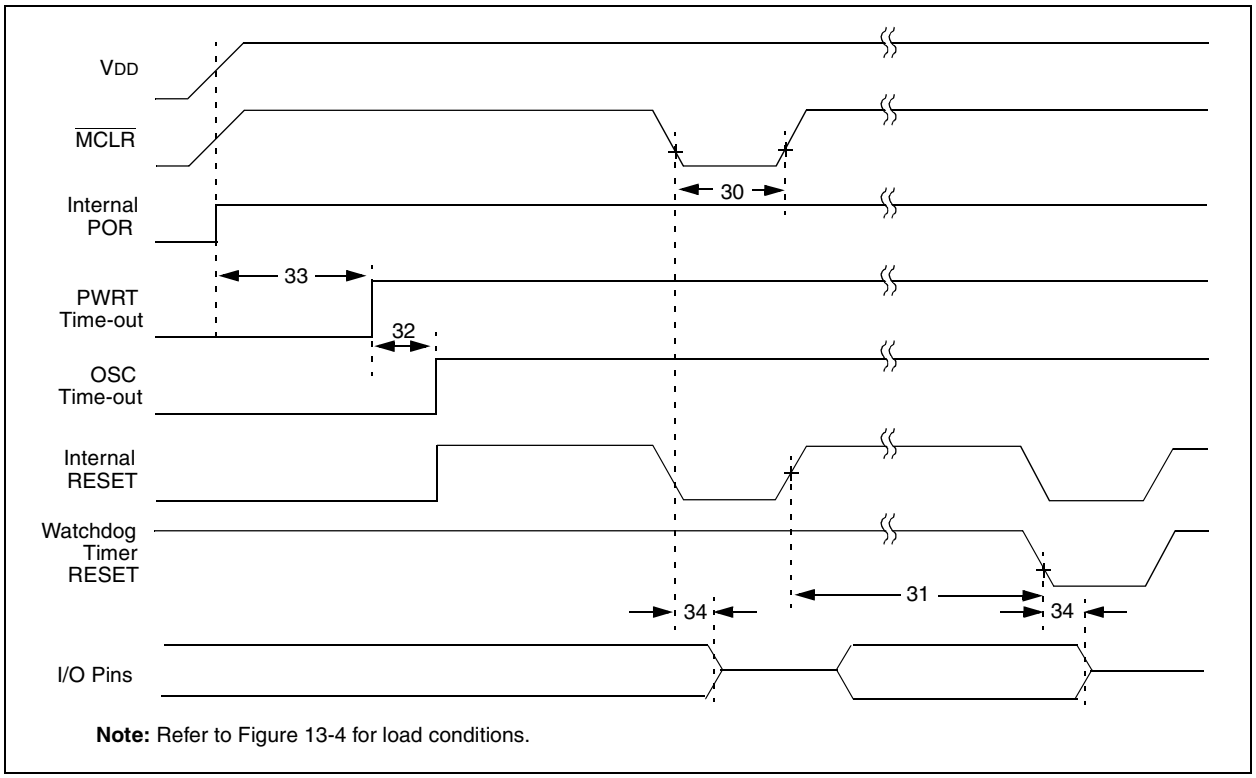
* These parameters are characterized but not tested.

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

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

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

**FIGURE 13-7: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING**




**Note:** Refer to Figure 13-4 for load conditions.

**FIGURE 13-8: BROWN-OUT RESET TIMING**

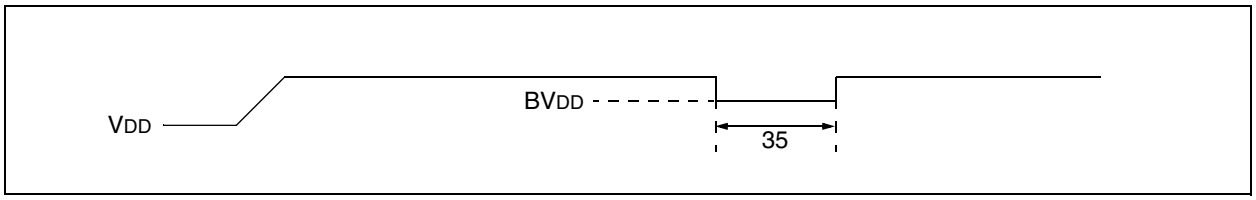



**TABLE 13-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER AND BROWN-OUT RESET REQUIREMENTS**

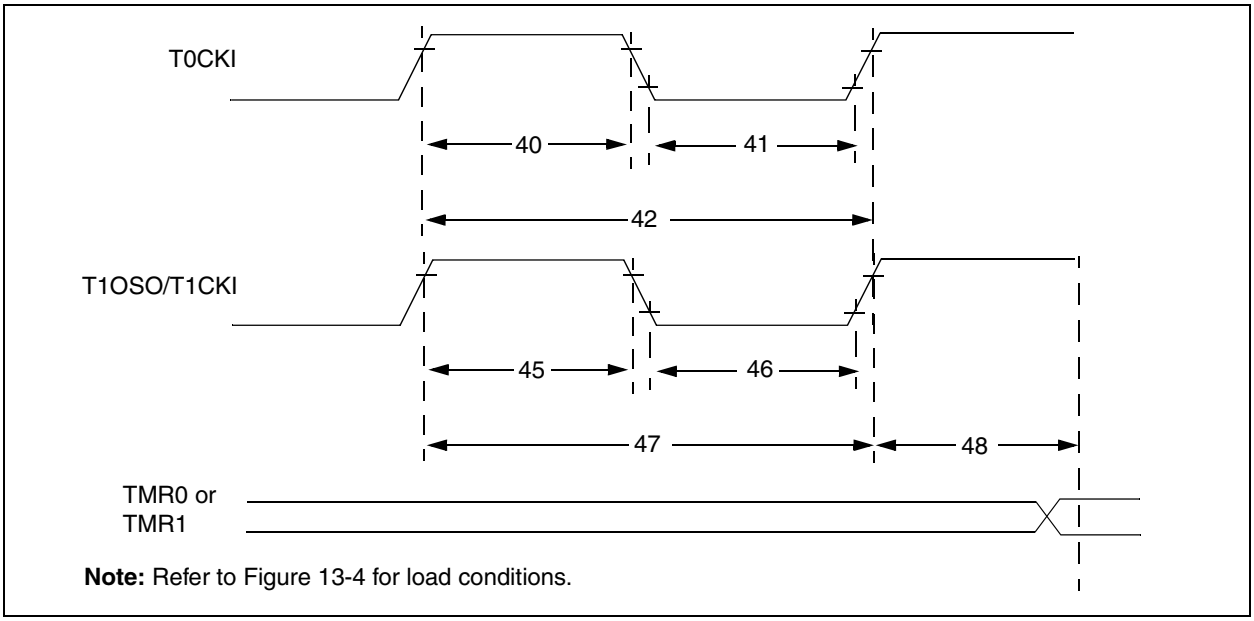
| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| 30 | TmcL | $\overline{MCLR}$ Pulse Width (low) | 2 | — | — | μs | VDD = 5V, -40°C to +125°C |
| 31* | Twdt | Watchdog Timer Time-out Period (No Prescaler) | 7 | 18 | 33 | ms | VDD = 5V, -40°C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

* These parameters are characterized but not tested.

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

 © 1998-2013 Microchip Technology Inc.

**FIGURE 13-9: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS**




**Note:** Refer to Figure 13-4 for load conditions.

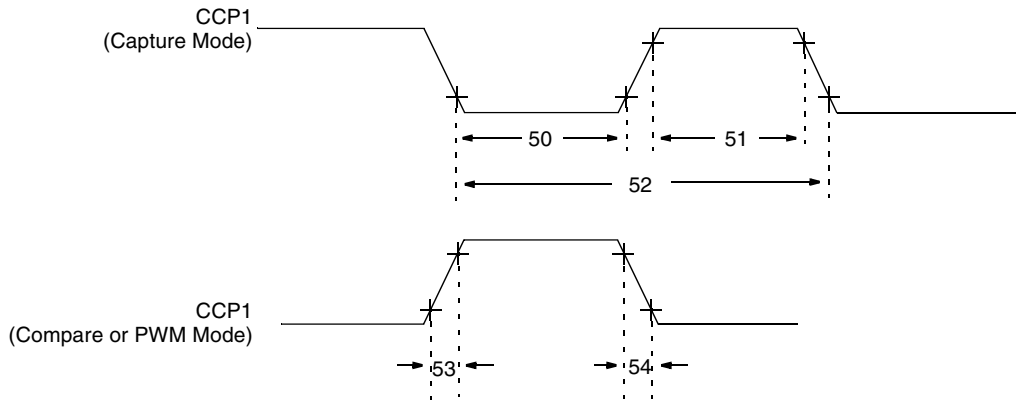
**TABLE 13-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS**

| Param No. | Sym | Characteristic | | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|---|
| 40* | Tt0H | T0CKI High Pulse Width | | No Prescaler | 0.5TCY + 20 | — | — | ns | Must also meet parameter 42 |
| | | | | With Prescaler | 10 | — | — | ns | |
| 41* | Tt0L | T0CKI Low Pulse Width | | No Prescaler | 0.5TCY + 20 | — | — | ns | Must also meet parameter 42 |
| | | | | With Prescaler | 10 | — | — | ns | |
| 42* | Tt0P | T0CKI Period | | No Prescaler | TCY + 40 | — | — | ns | |
| | | | | With Prescaler | Greater of: 20 or TCY + 40 / N | — | — | ns | N = prescale value (2, 4,..., 256) |
| 45* | Tt1H | T1CKI High Time | Synchronous, Prescaler = 1 | | 0.5TCY + 20 | — | — | ns | Must also meet parameter 47 |
| | | | Synchronous, Prescaler = 2,4,8 | **PIC16CXX** | 15 | — | — | ns | |
| | | | | **PIC16LCXX** | 25 | — | — | ns | |
| | | | Asynchronous | **PIC16CXX** | 30 | — | — | ns | |
| | | | | **PIC16LCXX** | 50 | — | — | ns | |
| 46* | Tt1L | T1CKI Low Time | Synchronous, Prescaler = 1 | | 0.5TCY + 20 | — | — | ns | Must also meet parameter 47 |
| | | | Synchronous, Prescaler = 2,4,8 | **PIC16CXX** | 15 | — | — | ns | |
| | | | | **PIC16LCXX** | 25 | — | — | ns | |
| | | | Asynchronous | **PIC16CXX** | 30 | — | — | ns | |
| | | | | **PIC16LCXX** | 50 | — | — | ns | |
| 47* | Tt1P | T1CKI input period | Synchronous | **PIC16CXX** | GREATER OF: 30 OR TCY + 40 / N | — | — | ns | N = prescale value (1, 2, 4, 8) |
| | | | | **PIC16LCXX** | GREATER OF: 50 OR TCY + 40 / N | | | | N = prescale value (1, 2, 4, 8) |
| | | | Asynchronous | **PIC16CXX** | 60 | — | — | ns | |
| | | | | **PIC16LCXX** | 100 | — | — | ns | |
| | Ft1 | Timer1 oscillator input frequency range (oscillator enabled by setting bit T1OSCEN) | | | DC | — | 200 | kHz | |
| 48 | TCKEZtmr1 | Delay from external clock edge to timer increment | | | 2Tosc | — | 7Tosc | — | |

* These parameters are characterized but not tested.

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

**FIGURE 13-10: CAPTURE/COMPARE/PWM TIMINGS**




**Note:** Refer to Figure 13-4 for load conditions.

**TABLE 13-6: CAPTURE/COMPARE/PWM REQUIREMENTS**

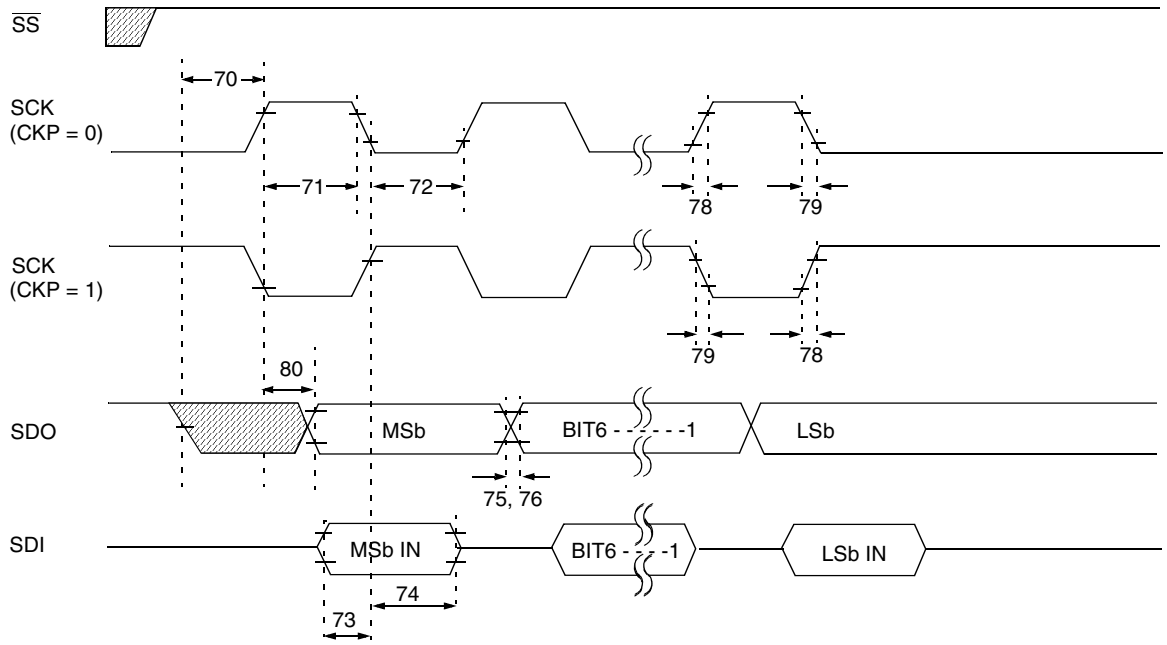
| Param No. | Sym | Characteristic | | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|---|
| 50* | TccL | CCP1 input low time | No Prescaler | | 0.5TCY + 20 | — | — | ns | |
| | | | With Prescaler | **PIC16CXX** | 10 | — | — | ns | |
| | | | | **PIC16LCXX** | 20 | — | — | ns | |
| 51* | TccH | CCP1 input high time | No Prescaler | | 0.5TCY + 20 | — | — | ns | |
| | | | With Prescaler | **PIC16CXX** | 10 | — | — | ns | |
| | | | | **PIC16LCXX** | 20 | — | — | ns | |
| 52* | TccP | CCP1 input period | | | $\frac{3T_{CY} + 40}{N}$ | — | — | ns | N = prescale value (1,4, or 16) |
| 53* | TccR | CCP1 output rise time | | **PIC16CXX** | — | 10 | 25 | ns | |
| | | | | **PIC16LCXX** | — | 25 | 45 | ns | |
| 54* | TccF | CCP1 output fall time | | **PIC16CXX** | — | 10 | 25 | ns | |
| | | | | **PIC16LCXX** | — | 25 | 45 | ns | |

* These parameters are characterized but not tested.

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

 © 1998-2013 Microchip Technology Inc.

**FIGURE 13-11: EXAMPLE SPI MASTER MODE TIMING (CKE = 0)**




**Note:** Refer to Figure 13-4 for load conditions.

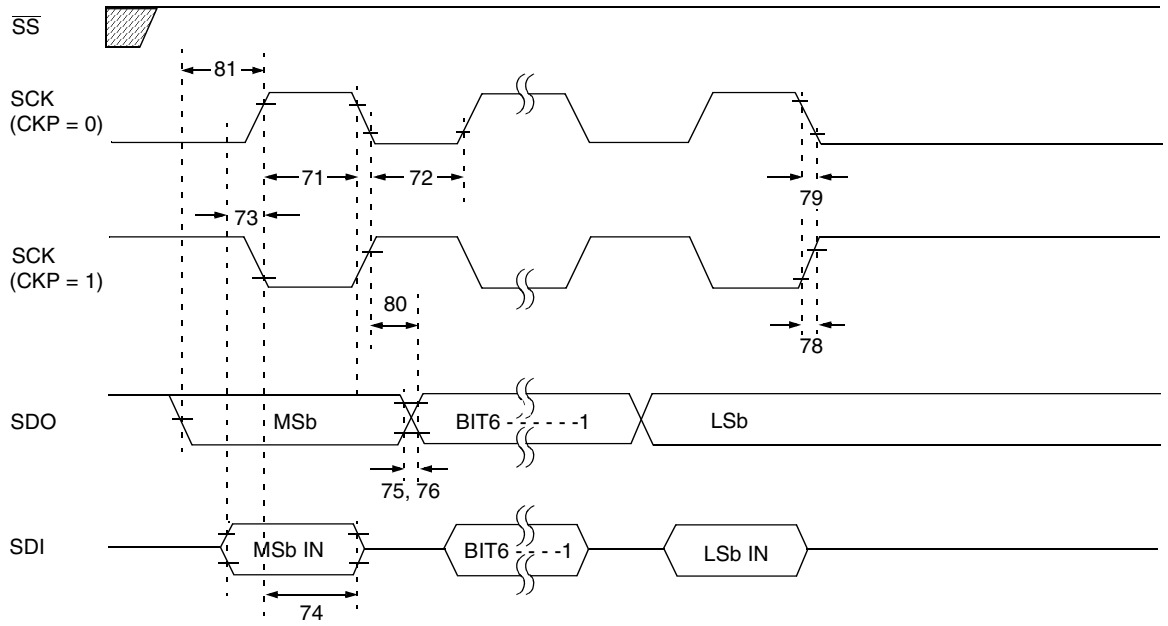
**TABLE 13-7: EXAMPLE SPI MODE REQUIREMENTS (MASTER MODE, CKE = 0)**

| Param. No. | Symbol | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 70 | TssL2scH, TssL2scL | $\overline{SS}\downarrow$ to SCK↓ or SCK↑ input | | TCY | — | — | ns | |
| 71 | TscH | SCK input high time (slave mode) | Continuous | 1.25TCY + 30 | — | — | ns | |
| 71A | | | Single Byte | 40 | — | — | ns | Note 1 |
| 72 | TscL | SCK input low time (slave mode) | Continuous | 1.25TCY + 30 | — | — | ns | |
| 72A | | | Single Byte | 40 | — | — | ns | Note 1 |
| 73 | TdiV2scH, TdiV2scL | Setup time of SDI data input to SCK edge | | 100 | — | — | ns | |
| 73A | TB2B | Last clock edge of Byte1 to the 1st clock edge of Byte2 | | 1.5TCY + 40 | — | — | ns | Note 1 |
| 74 | TscH2diL, TscL2diL | Hold time of SDI data input to SCK edge | | 100 | — | — | ns | |
| 75 | TdoR | SDO data output rise time | **PIC16CXX** | — | 10 | 25 | ns | |
| | | | **PIC16LCXX** | — | 20 | 45 | ns | |
| 76 | TdoF | SDO data output fall time | | — | 10 | 25 | ns | |
| 78 | TscR | SCK output rise time (master mode) | **PIC16CXX** | — | 10 | 25 | ns | |
| | | | **PIC16LCXX** | — | 20 | 45 | ns | |
| 79 | TscF | SCK output fall time (master mode) | | — | 10 | 25 | ns | |
| 80 | TscH2doV, TscL2doV | SDO data output valid after SCK edge | **PIC16CXX** | — | — | 50 | ns | |
| | | | **PIC16LCXX** | — | — | 100 | ns | |

† 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:** Specification 73A is only required if specifications 71A and 72A are used.

**FIGURE 13-12: EXAMPLE SPI MASTER MODE TIMING (CKE = 1)**




**Note:** Refer to Figure 13-4 for load conditions.

**TABLE 13-8: EXAMPLE SPI MODE REQUIREMENTS (MASTER MODE, CKE = 1)**

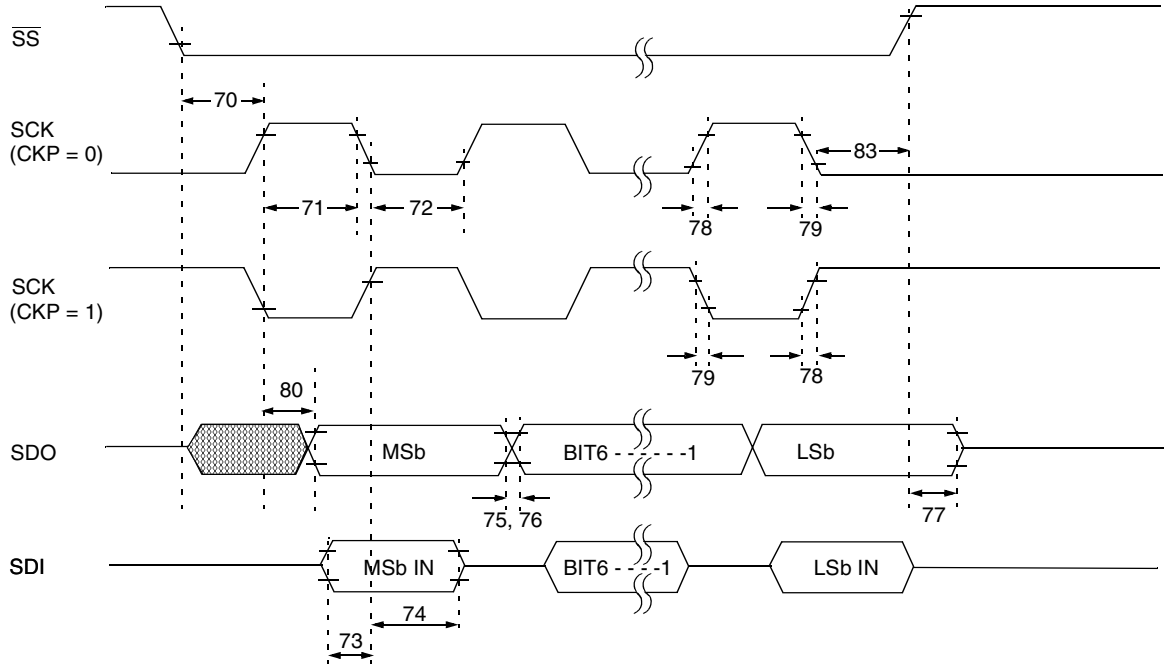
| Param. No. | Symbol | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 71 | TscH | SCK input high time (slave mode) | Continuous | 1.25TCY + 30 | — | — | ns | |
| 71A | | | Single Byte | 40 | — | — | ns | Note 1 |
| 72 | TscL | SCK input low time (slave mode) | Continuous | 1.25TCY + 30 | — | — | ns | |
| 72A | | | Single Byte | 40 | — | — | ns | Note 1 |
| 73 | TdiV2scH, TdiV2scL | Setup time of SDI data input to SCK edge | | 100 | — | — | ns | |
| 73A | TB2B | Last clock edge of Byte1 to the 1st clock edge of Byte2 | | 1.5TCY + 40 | — | — | ns | Note 1 |
| 74 | TscH2diL, TscL2diL | Hold time of SDI data input to SCK edge | | 100 | — | — | ns | |
| 75 | TdoR | SDO data output rise time | **PIC16CXX** | — | 10 | 25 | ns | |
| | | | **PIC16LCXX** | | 20 | 45 | ns | |
| 76 | TdoF | SDO data output fall time | | — | 10 | 25 | ns | |
| 78 | TscR | SCK output rise time (master mode) | **PIC16CXX** | — | 10 | 25 | ns | |
| | | | **PIC16LCXX** | | 20 | 45 | ns | |
| 79 | TscF | SCK output fall time (master mode) | | — | 10 | 25 | ns | |
| 80 | TscH2doV, TscL2doV | SDO data output valid after SCK edge | **PIC16CXX** | — | — | 50 | ns | |
| | | | **PIC16LCXX** | | — | 100 | ns | |
| 81 | TdoV2scH, TdoV2scL | SDO data output setup to SCK edge | | TCY | — | — | ns | |

† 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:** Specification 73A is only required if specifications 71A and 72A are used.

 © 1998-2013 Microchip Technology Inc.

**FIGURE 13-13: EXAMPLE SPI SLAVE MODE TIMING (CKE = 0)**




**Note:** Refer to Figure 13-4 for load conditions.

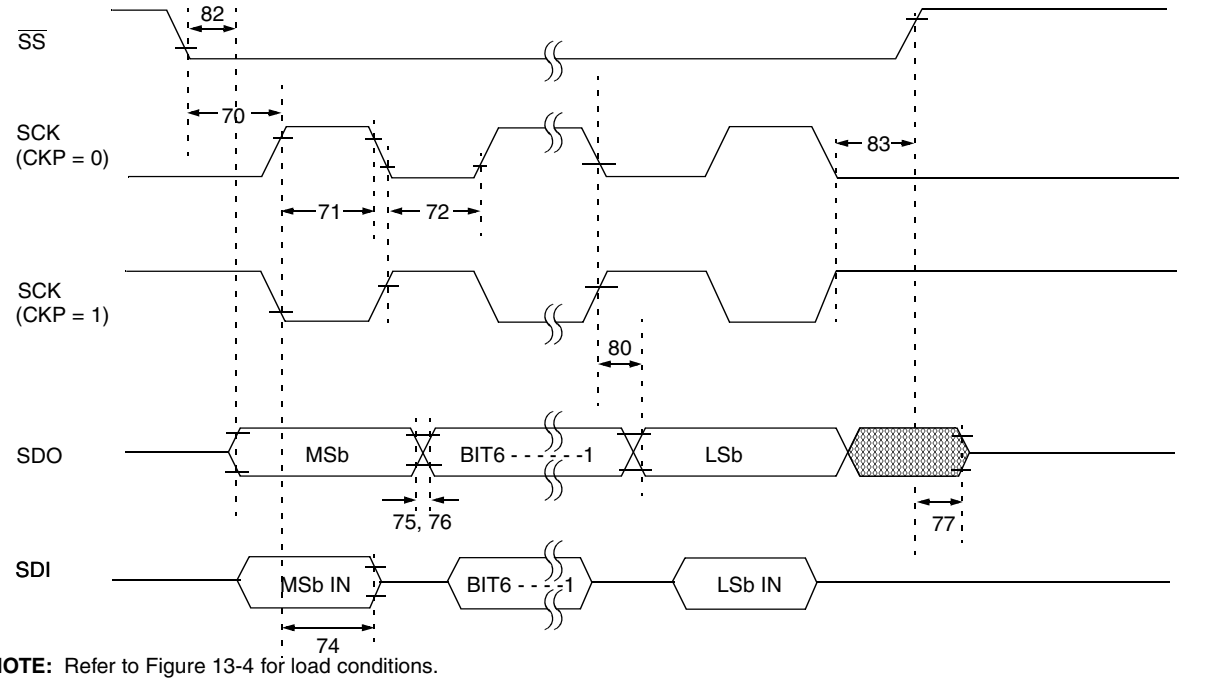
**TABLE 13-9: EXAMPLE SPI MODE REQUIREMENTS (SLAVE MODE TIMING (CKE = 0)**

| Param. No. | Symbol | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 70 | TssL2scH, TssL2scL | $\overline{SS}$↓ to SCK↓ or SCK↑ input | | TCY | — | — | ns | |
| 71 | TscH | SCK input high time (slave mode) | Continuous | 1.25TCY + 30 | — | — | ns | |
| 71A | | | Single Byte | 40 | — | — | ns | Note 1 |
| 72 | TscL | SCK input low time (slave mode) | Continuous | 1.25TCY + 30 | — | — | ns | |
| 72A | | | Single Byte | 40 | — | — | ns | Note 1 |
| 73 | TdiV2scH, TdiV2scL | Setup time of SDI data input to SCK edge | | 100 | — | — | ns | |
| 73A | TB2B | Last clock edge of Byte1 to the 1st clock edge of Byte2 | | 1.5TCY + 40 | — | — | ns | Note 1 |
| 74 | TscH2diL, TscL2diL | Hold time of SDI data input to SCK edge | | 100 | — | — | ns | |
| 75 | TdoR | SDO data output rise time | **PIC16CXX** | — | 10 | 25 | ns | |
| | | | **PIC16LCXX** | | 20 | 45 | ns | |
| 76 | TdoF | SDO data output fall time | | — | 10 | 25 | ns | |
| 77 | TssH2doZ | $\overline{SS}$↑ to SDO output hi-impedance | | 10 | — | 50 | ns | |
| 78 | TscR | SCK output rise time (master mode) | **PIC16CXX** | — | 10 | 25 | ns | |
| | | | **PIC16LCXX** | | 20 | 45 | ns | |
| 79 | TscF | SCK output fall time (master mode) | | — | 10 | 25 | ns | |
| 80 | TscH2doV, TscL2doV | SDO data output valid after SCK edge | **PIC16CXX** | — | — | 50 | ns | |
| | | | **PIC16LCXX** | | — | 100 | ns | |
| 83 | TscH2ssH, TscL2ssH | $\overline{SS}$ ↑ after SCK edge | | 1.5TCY + 40 | — | — | ns | |

† 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:** Specification 73A is only required if specifications 71A and 72A are used.

**FIGURE 13-14: EXAMPLE SPI SLAVE MODE TIMING (CKE = 1)**




**NOTE:** Refer to Figure 13-4 for load conditions.

**TABLE 13-10: EXAMPLE SPI SLAVE MODE REQUIREMENTS (CKE = 1)**

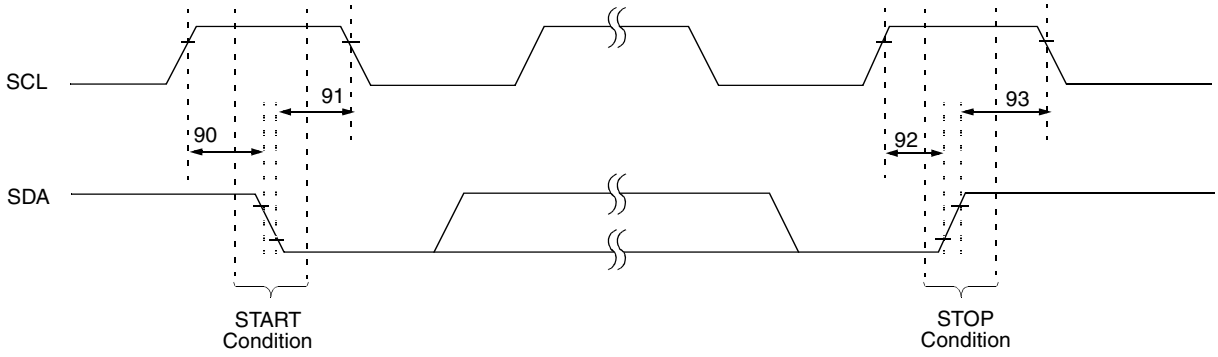
| Param. No. | Symbol | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 70 | TssL2scH, TssL2scL | $\overline{SS}$↓ to SCK↓ or SCK↑ input | | TCY | — | — | ns | |
| 71 | TscH | SCK input high time (slave mode) | Continuous | 1.25TCY + 30 | — | — | ns | |
| 71A | | | Single Byte | 40 | — | — | ns | Note 1 |
| 72 | TscL | SCK input low time (slave mode) | Continuous | 1.25TCY + 30 | — | — | ns | |
| 72A | | | Single Byte | 40 | — | — | ns | Note 1 |
| 73A | TB2B | Last clock edge of Byte1 to the 1st clock edge of Byte2 | | 1.5TCY + 40 | — | — | ns | Note 1 |
| 74 | TscH2diL, TscL2diL | Hold time of SDI data input to SCK edge | | 100 | — | — | ns | |
| 75 | TdoR | SDO data output rise time | **PIC16CXX** | — | 10 | 25 | ns | |
| | | | **PIC16LCXX** | | 20 | 45 | ns | |
| 76 | TdoF | SDO data output fall time | | — | 10 | 25 | ns | |
| 77 | TssH2doZ | $\overline{SS}$↑ to SDO output hi-impedance | | 10 | — | 50 | ns | |
| 78 | TscR | SCK output rise time (master mode) | **PIC16CXX** | — | 10 | 25 | ns | |
| | | | **PIC16LCXX** | — | 20 | 45 | ns | |
| 79 | TscF | SCK output fall time (master mode) | | — | 10 | 25 | ns | |
| 80 | TscH2doV, TscL2doV | SDO data output valid after SCK edge | **PIC16CXX** | — | — | 50 | ns | |
| | | | **PIC16LCXX** | — | — | 100 | ns | |
| 82 | TssL2doV | SDO data output valid after $\overline{SS}$↓ edge | **PIC16CXX** | — | — | 50 | ns | |
| | | | **PIC16LCXX** | — | — | 100 | ns | |
| 83 | TscH2ssH, TscL2ssH | $\overline{SS}$ ↑ after SCK edge | | 1.5TCY + 40 | — | — | ns | |

† 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:** Specification 73A is only required if specifications 71A and 72A are used.

 © 1998-2013 Microchip Technology Inc.

**FIGURE 13-15: $I^2C$ BUS START/STOP BITS TIMING**



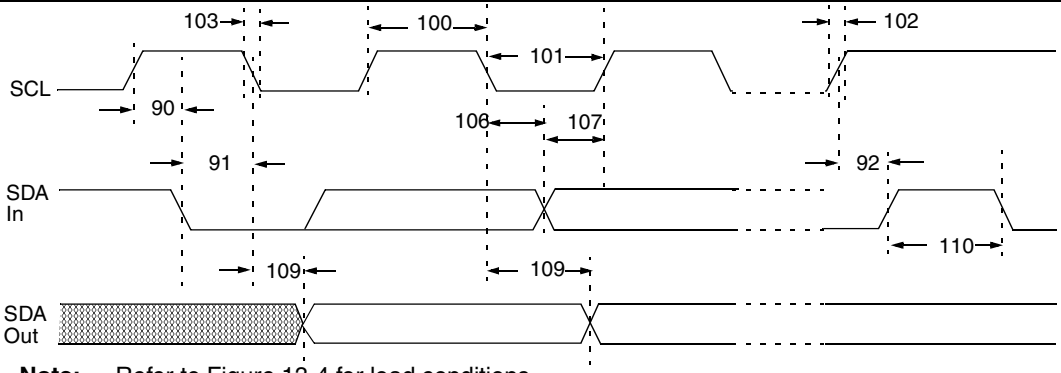

**Note:** Refer to Figure 13-4 for load conditions.

**TABLE 13-11: $I^2C$ BUS START/STOP BITS REQUIREMENTS**

| Parameter No. | Sym | Characteristic | | Min | Typ | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 90* | TSU:STA | START condition | 100 kHz mode | 4700 | — | — | ns | Only relevant for repeated START condition |
| | | Setup time | 400 kHz mode | 600 | — | — | | |
| 91* | THD:STA | START condition | 100 kHz mode | 4000 | — | — | ns | After this period the first clock pulse is generated |
| | | Hold time | 400 kHz mode | 600 | — | — | | |
| 92* | TSU:STO | STOP condition | 100 kHz mode | 4700 | — | — | ns | |
| | | Setup time | 400 kHz mode | 600 | — | — | | |
| 93 | THD:STO | STOP condition | 100 kHz mode | 4000 | — | — | ns | |
| | | Hold time | 400 kHz mode | 600 | — | — | | |

* These parameters are characterized but not tested.

**FIGURE 13-16: I²C BUS DATA TIMING**




**Note:** Refer to Figure 13-4 for load conditions.

**TABLE 13-12: I²C BUS DATA REQUIREMENTS**

| Param. No. | Sym | Characteristic | | Min | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|
| 100* | THIGH | Clock high time | 100 kHz mode | 4.0 | — | µs | Device must operate at a minimum of 1.5 MHz |
| | | | 400 kHz mode | 0.6 | — | µs | Device must operate at a minimum of 10 MHz |
| | | | SSP Module | 1.5TCY | — | | |
| 101* | TLOW | Clock low time | 100 kHz mode | 4.7 | — | µs | Device must operate at a minimum of 1.5 MHz |
| | | | 400 kHz mode | 1.3 | — | µs | Device must operate at a minimum of 10 MHz |
| | | | SSP Module | 1.5TCY | — | | |
| 102* | TR | SDA and SCL rise time | 100 kHz mode | — | 1000 | ns | |
| | | | 400 kHz mode | 20 + 0.1Cb | 300 | ns | Cb is specified to be from 10-400 pF |
| 103* | TF | SDA and SCL fall time | 100 kHz mode | — | 300 | ns | |
| | | | 400 kHz mode | 20 + 0.1Cb | 300 | ns | Cb is specified to be from 10-400 pF |
| 90* | TSU:STA | START condition setup time | 100 kHz mode | 4.7 | — | µs | Only relevant for repeated START condition |
| | | | 400 kHz mode | 0.6 | — | µs | |
| 91* | THD:STA | START condition hold time | 100 kHz mode | 4.0 | — | µs | After this period the first clock pulse is generated |
| | | | 400 kHz mode | 0.6 | — | µs | |
| 106* | THD:DAT | Data input hold time | 100 kHz mode | 0 | — | ns | |
| | | | 400 kHz mode | 0 | 0.9 | µs | |
| 107* | TSU:DAT | Data input setup time | 100 kHz mode | 250 | — | ns | Note 2 |
| | | | 400 kHz mode | 100 | — | ns | |
| 92* | TSU:STO | STOP condition setup time | 100 kHz mode | 4.7 | — | µs | |
| | | | 400 kHz mode | 0.6 | — | µs | |
| 109* | TAA | Output valid from clock | 100 kHz mode | — | 3500 | ns | Note 1 |
| | | | 400 kHz mode | — | — | ns | |
| 110* | TBUF | Bus free time | 100 kHz mode | 4.7 | — | µs | Time the bus must be free before a new transmission can start |
| | | | 400 kHz mode | 1.3 | — | µs | |
| | Cb | Bus capacitive loading | | — | 400 | pF | |

* These parameters are characterized but not tested.

**Note 1:** As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCL to avoid unintended generation of START or STOP conditions.

**2:** A fast-mode (400 kHz) I²C-bus device can be used in a standard-mode (100 kHz) I²C-bus system, but the requirement Tsu:DAT ≥ 250 ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line TR max.+tsu;DAT = 1000 + 250 = 1250 ns (according to the standard-mode I²C bus specification) before the SCL line is released.

 © 1998-2013 Microchip Technology Inc.

**TABLE 13-13: A/D CONVERTER CHARACTERISTICS: PIC16C72A-04 (COMMERCIAL, INDUSTRIAL, EXTENDED) PIC16C72A-20 (COMMERCIAL, INDUSTRIAL, EXTENDED) PIC16LC72A-04 (COMMERCIAL, INDUSTRIAL)**

| Param No. | Sym | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| A01 | NR | Resolution | | — | — | 8-bits | bit | VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF |
| A02 | EABS | Total Absolute error | | — | — | < ± 1 | LSB | VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF |
| A03 | EIL | Integral linearity error | | — | — | < ± 1 | LSB | VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF |
| A04 | EDL | Differential linearity error | | — | — | < ± 1 | LSB | VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF |
| A05 | EFS | Full scale error | | — | — | < ± 1 | LSB | VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF |
| A06 | EOFF | Offset error | | — | — | < ± 1 | LSB | VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF |
| A10 | — | Monotonicity | | — | guaranteed (Note 3) | — | — | VSS ≤ VAIN ≤ VREF |
| A20 | VREF | Reference voltage | | 2.5V | — | VDD + 0.3 | V | |
| A25 | VAIN | Analog input voltage | | VSS - 0.3 | — | VREF + 0.3 | V | |
| A30 | ZAIN | Recommended impedance of analog voltage source | | — | — | 10.0 | kΩ | |
| A40 | IAD | A/D conversion current (VDD) | **PIC16CXX** | — | 180 | — | μA | Average current consumption when A/D is on. (Note 1) |
| | | | **PIC16LCXX** | — | 90 | — | μA | |
| A50 | IREF | VREF input current (Note 2) | | 10 | — | 1000 | μA | During VAIN acquisition. Based on differential of VHOLD to VAIN to charge CHOLD, see Section 9.1. |
| | | | | — | — | 10 | μA | During A/D conversion cycle |

* These parameters are characterized but not tested.

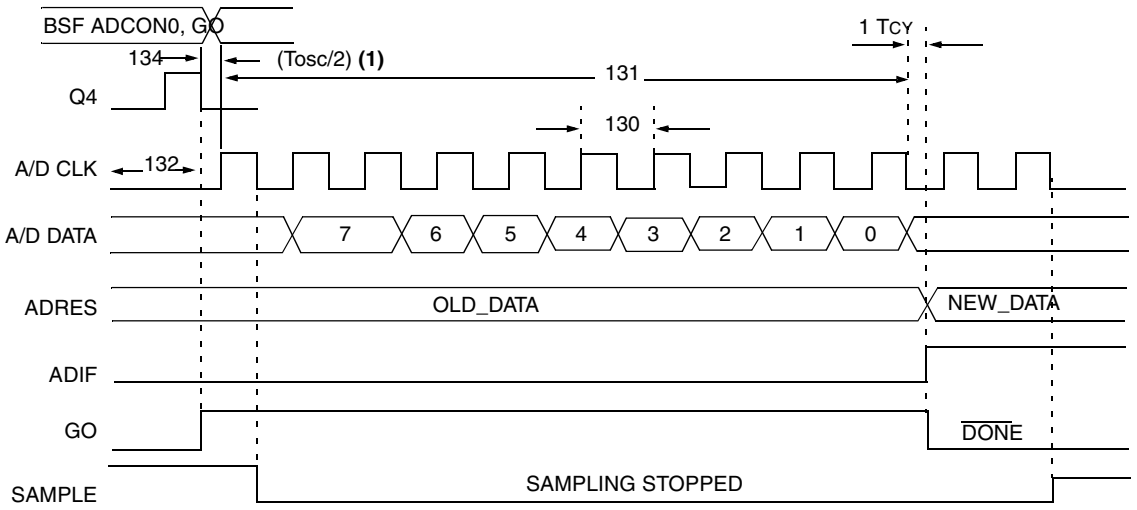
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

**Note 1:** When A/D is off, it will not consume any current other than minor leakage current. The power-down current spec includes any such leakage from the A/D module.

**2:** VREF current is from RA3 pin or VDD pin, whichever is selected as reference input.

**3:** The A/D conversion result never decreases with an increase in the Input Voltage and has no missing codes.

**FIGURE 13-17: A/D CONVERSION TIMING**




**Note 1:** If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

**TABLE 13-14: A/D CONVERSION REQUIREMENTS**

| Param No. | Sym | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 130 | TAD | A/D clock period | **PIC16CXX** | 1.6 | — | — | µs | TOSC based, VREF ≥ 3.0V |
| | | | **PIC16LCXX** | 2.0 | — | — | µs | TOSC based, VREF full range |
| | | | **PIC16CXX** | 2.0 | 4.0 | 6.0 | µs | A/D RC Mode |
| | | | **PIC16LCXX** | 3.0 | 6.0 | 9.0 | µs | A/D RC Mode |
| 131 | TCNV | Conversion time (not including S/H time) (Note 1) | | 11 | — | 11 | TAD | |
| 132 | TACQ | Acquisition time | | Note 2 | 20 | — | µs | |
| | | | | 5* | — | — | µs | The minimum time is the amplifier settling time. This may be used if the "new" input voltage has not changed by more than 1 LSb (i.e., 20.0 mV @ 5.12V) from the last sampled voltage (as stated on CHOLD). |
| 134 | TGO | Q4 to A/D clock start | | — | TOSC/2 | — | — | If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed. |
| 135 | Tswc | Switching from convert → sample time | | 1.5 | — | — | TAD | |

* These parameters are characterized but not tested.

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

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

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

 © 1998-2013 Microchip Technology Inc.

## 14.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES

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

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

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

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

Data is not available at this time but you may reference the *PIC16C72 Series Data Sheet* (DS39016,) DC and AC characteristic section, which contains data similar to what is expected.

**NOTES:**

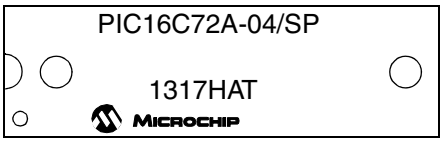
 © 1998-2013 Microchip Technology Inc.

## 15.0 PACKAGING INFORMATION

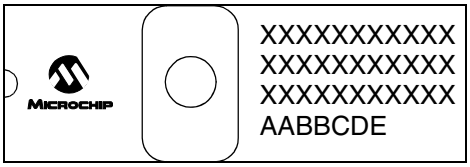
### 15.1 Package Marking Information

28-Lead PDIP (Skinny DIP)

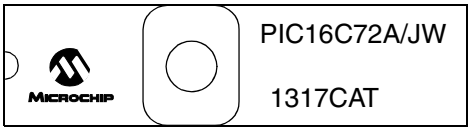
Example



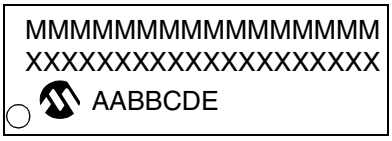

28-Lead CERDIP Windowed



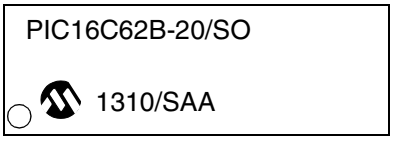

Example



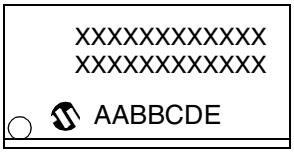

28-Lead SOIC



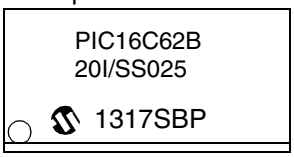

Example




28-Lead SSOP




Example

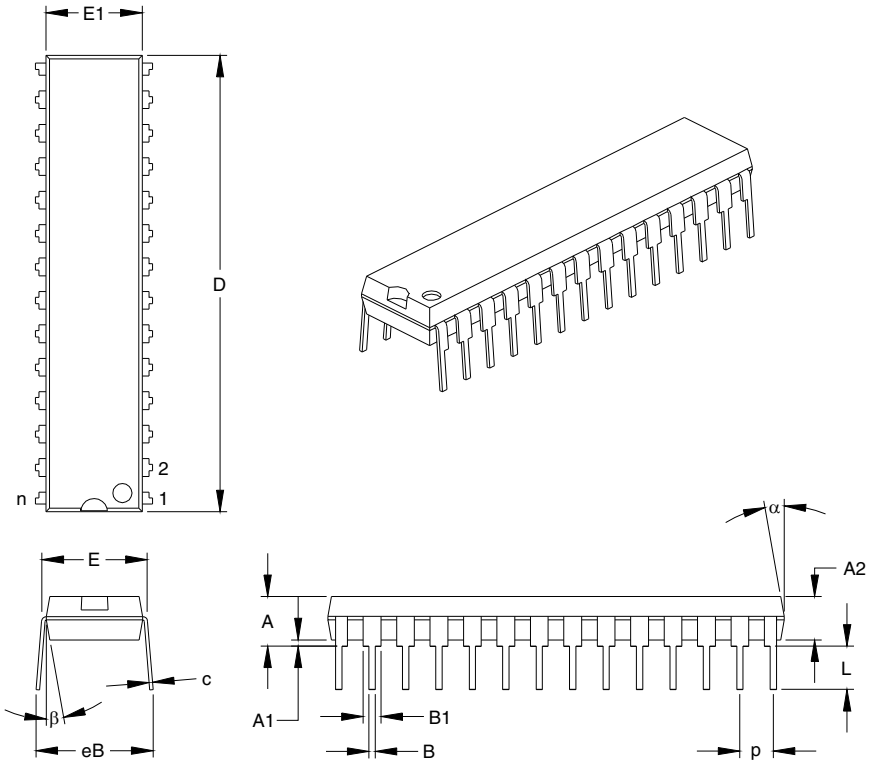



| | | |
|---|---|---|
| **Legend:** | MM...M | Microchip part number information |
| | XX...X | Customer specific information* |
| | AA | Year code (last 2 digits of calendar year) |
| | BB | Week code (week of January 1 is week ‘01’) |
| | C | Facility code of the plant at which wafer is manufactured<br>O = Outside Vendor<br>C = 5” Line<br>S = 6” Line<br>H = 8” Line |
| | D | Mask revision number |
| | E | Assembly code of the plant or country of origin in which part was assembled |

**Note**: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line thus limiting the number of available characters for customer specific information.

* Standard OTP marking consists of Microchip part number, year code, week code, facility code, mask rev#, and assembly code. For OTP marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

## 15.2 28-Lead Skinny Plastic Dual In-line (SP) – 300 mil (PDIP)




| Units | | INCHES* | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 28 | | | 28 | |
| Pitch | p | | .100 | | | 2.54 | |
| Top to Seating Plane | A | .140 | .150 | .160 | 3.56 | 3.81 | 4.06 |
| Molded Package Thickness | A2 | .125 | .130 | .135 | 3.18 | 3.30 | 3.43 |
| 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 | .279 | .307 | .335 | 7.09 | 7.80 | 8.51 |
| Overall Length | D | 1.345 | 1.365 | 1.385 | 34.16 | 34.67 | 35.18 |
| 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 | .040 | .053 | .065 | 1.02 | 1.33 | 1.65 |
| Lower Lead Width | B | .016 | .019 | .022 | 0.41 | 0.48 | 0.56 |
| Overall Row Spacing | eB | .320 | .350 | .430 | 8.13 | 8.89 | 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

Notes:

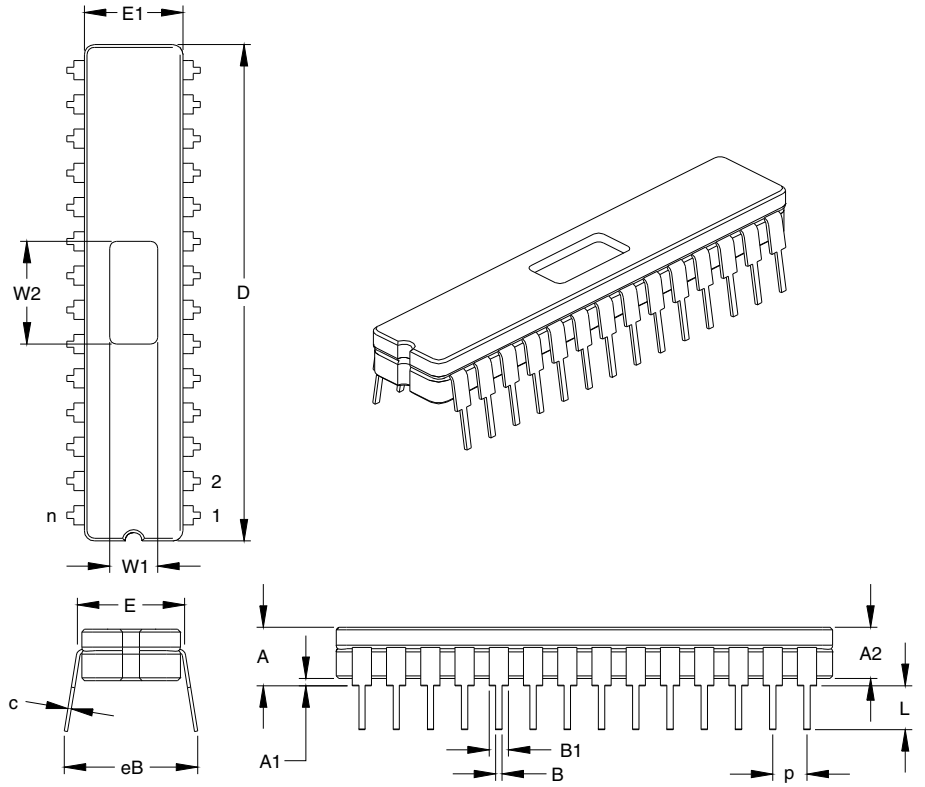
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010” (0.254mm) per side.

JEDEC Equivalent: MO-095

Drawing No. C04-070

 © 1998-2013 Microchip Technology Inc.

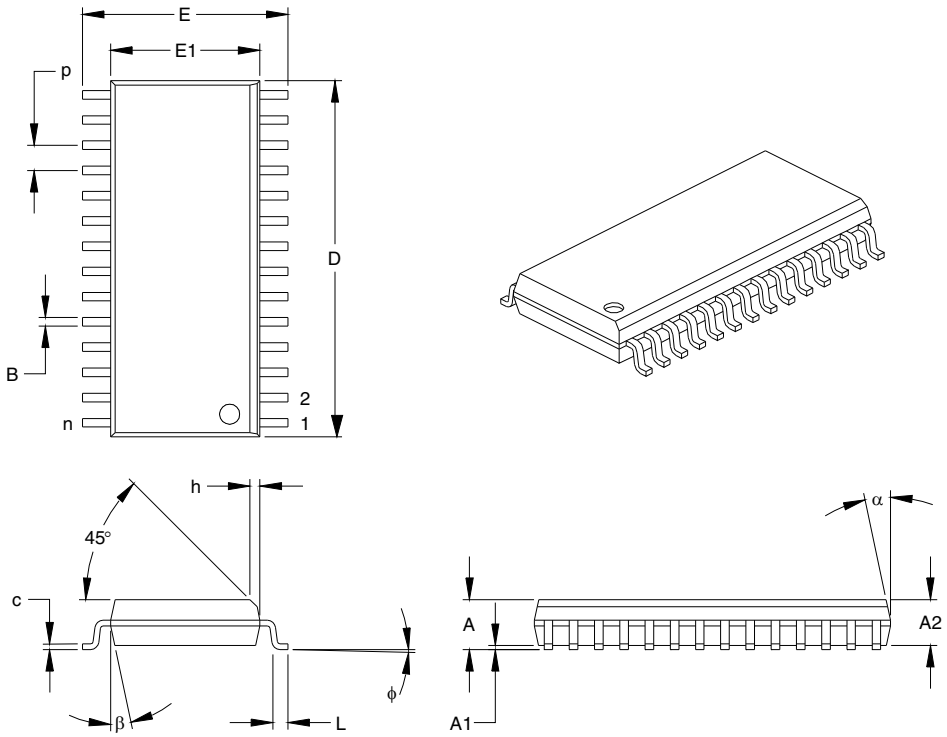
## 15.3 28-Lead Ceramic Dual In-line with Window (JW) – 300 mil (CERDIP)




| Units | | INCHES* | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 28 | | | 28 | |
| 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 | 1.430 | 1.458 | 1.485 | 36.32 | 37.02 | 37.72 |
| Tip to Seating Plane | L | .135 | .140 | .145 | 3.43 | 3.56 | 3.68 |
| Lead Thickness | c | .008 | .010 | .012 | 0.20 | 0.25 | 0.30 |
| Upper Lead Width | B1 | .050 | .058 | .065 | 1.27 | 1.46 | 1.65 |
| 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 | .290 | .300 | .310 | 7.37 | 7.62 | 7.87 |

*Controlling Parameter
JEDEC Equivalent: MO-058
Drawing No. C04-080

## 15.4 28-Lead Plastic Small Outline (SO) – Wide, 300 mil (SOIC)




| Units | | INCHES* | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 28 | | | 28 | |
| 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 | .288 | .295 | .299 | 7.32 | 7.49 | 7.59 |
| Overall Length | D | .695 | .704 | .712 | 17.65 | 17.87 | 18.08 |
| 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 Top | ϕ | 0 | 4 | 8 | 0 | 4 | 8 |
| Lead Thickness | c | .009 | .011 | .013 | 0.23 | 0.28 | 0.33 |
| 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 |

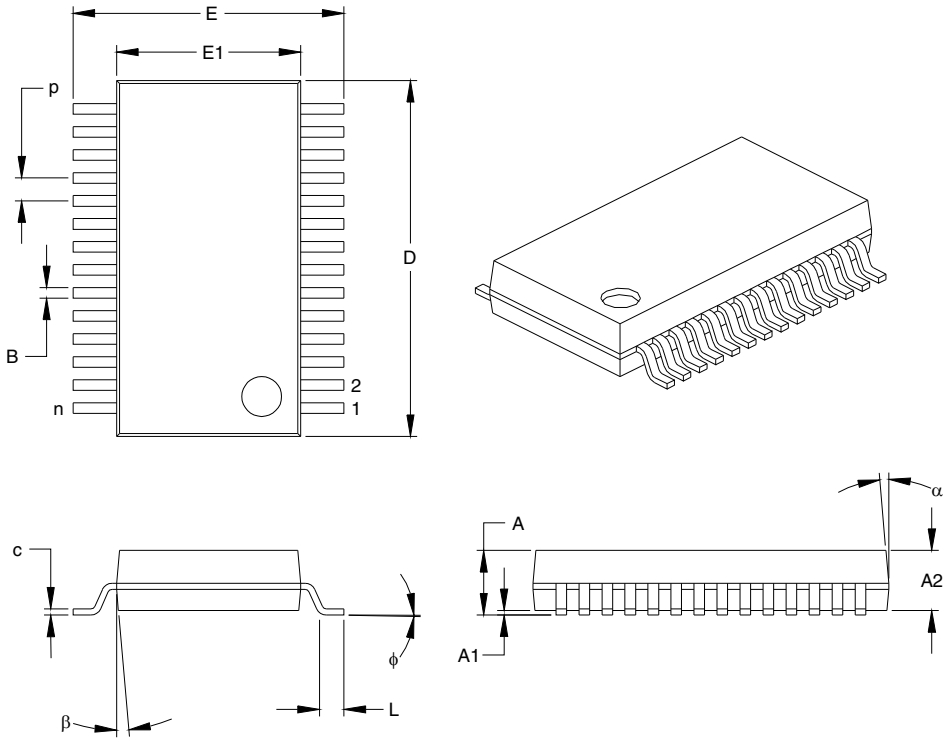
*Controlling Parameter

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. C04-052

 © 1998-2013 Microchip Technology Inc.

## 15.5 28-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 | | 28 | | | 28 | |
| Pitch | p | | .026 | | | 0.66 | |
| 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 | .319 | 7.59 | 7.85 | 8.10 |
| Molded Package Width | E1 | .201 | .207 | .212 | 5.11 | 5.25 | 5.38 |
| Overall Length | D | .396 | .402 | .407 | 10.06 | 10.20 | 10.34 |
| 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

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-150

Drawing No. C04-073

**NOTES:**

 © 1998-2013 Microchip Technology Inc.

## APPENDIX A: REVISION HISTORY

| Version | Date | Revision Description |
|---|---|---|
| A | 7/98 | This is a new data sheet. However, the devices described in this data sheet are the upgrades to the devices found in the *PIC16C6X Data Sheet*, DS30234, and the *PIC16C7X Data Sheet*, DS30390. |

## APPENDIX B: CONVERSION CONSIDERATIONS

Considerations for converting from previous versions of devices to the ones listed in this data sheet are listed in Table B-1.

**TABLE B-1: CONVERSION CONSIDERATIONS**

| Difference | PIC16C62A/72 | PIC16C62B/72A |
|---|---|---|
| Voltage Range | 2.5V - 6.0V | 2.5V - 5.5V |
| SSP module | Basic SSP (2 mode SPI) | SSP (4 mode SPI) |
| CCP module | CCP does not reset TMR1 when in special event trigger mode. | N/A |
| Timer1 module | Writing to TMR1L register can cause overflow in TMR1H register. | N/A |

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

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

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

1. Instruction word length is increased to 14-bits. This allows larger page sizes both in program memory (2K now as opposed to 512 before) and register file (128 bytes now versus 32 bytes before).
2. A PC high latch register (PCLATH) is added to handle program memory paging. Bits PA2, PA1, PA0 are removed from STATUS register.
3. Data memory paging is redefined slightly. STATUS register is modified.
4. Four new instructions have been added: `RETURN`, `RETFIE`, `ADDLW`, and `SUBLW`. Two instructions `TRIS` and `OPTION` are being phased out although they are kept for compatibility with PIC16C5X.
5. OPTION_REG and TRIS registers are made addressable.
6. Interrupt capability is added. Interrupt vector is at 0004h.
7. Stack size is increased to 8 deep.
8. Reset vector is changed to 0000h.
9. Reset of all registers is revisited. Five different reset (and wake-up) types are recognized. Registers are reset differently.
10. Wake up from SLEEP through interrupt is added.
11. Two separate timers, Oscillator Start-up Timer (OST) and Power-up Timer (PWRT) are included for more reliable power-up. These timers are invoked selectively to avoid unnecessary delays on power-up and wake-up.
12. PORTB has weak pull-ups and interrupt on change feature.
13. T0CKI pin is also a port pin (RA4) now.
14. FSR is made a full eight bit register.
15. “In-circuit serial programming” is made possible. The user can program PIC16CXX devices using only five pins: $V_{DD}$, $V_{SS}$, $\overline{MCLR}/V_{PP}$, RB6 (clock) and RB7 (data in/out).
16. PCON status register is added with a Power-on Reset status bit ($\overline{POR}$).
17. Code protection scheme is enhanced such that portions of the program memory can be protected, while the remainder is unprotected.
18. Brown-out protection circuitry has been added. Controlled by configuration word bit BODEN. Brown-out reset ensures the device is placed in a reset condition if $V_{DD}$ dips below a fixed setpoint.

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

1. Remove any program memory page select operations (PA2, PA1, PA0 bits) for `CALL`, `GOTO`.
2. Revisit any computed jump operations (write to PC or add to PC, etc.) to make sure page bits are set properly under the new scheme.
3. Eliminate any data memory page switching. Redefine data variables to reallocate them.
4. Verify all writes to STATUS, OPTION, and FSR registers since these have changed.
5. Change reset vector to 0000h.

 © 1998-2013 Microchip Technology Inc.

## INDEX


### A

A/D ..... 49
A/D Converter Enable (ADIE Bit) ..... 14
A/D Converter Flag (ADIF Bit) ..... 15, 51
A/D Converter Interrupt, Configuring ..... 51
ADCON0 Register ..... 9, 49
ADCON1 Register ..... 10, 49, 50
ADRES Register ..... 9, 49, 51
Analog Port Pins ..... 6
Analog Port Pins, Configuring ..... 53
Block Diagram ..... 51
Block Diagram, Analog Input Model ..... 52
Channel Select (CHS2:CHS0 Bits) ..... 49
Clock Select (ADCS1:ADCS0 Bits) ..... 49
Configuring the Module ..... 51
Conversion Clock (TAD) ..... 53
Conversion Status (GO/$\overline{\text{DONE}}$ Bit) ..... 49, 51
Conversions ..... 54
Converter Characteristics ..... 101
Module On/Off (ADON Bit) ..... 49
Port Configuration Control (PCFG2:PCFG0 Bits) ..... 50
Sampling Requirements ..... 52
Special Event Trigger (CCP) ..... 35, 54
Timing Diagram ..... 102
Absolute Maximum Ratings ..... 81
ADCON0 Register ..... 9, 49
ADCS1:ADCS0 Bits ..... 49
ADON Bit ..... 49
CHS2:CHS0 Bits ..... 49
GO/$\overline{\text{DONE}}$ Bit ..... 49, 51
ADCON1 Register ..... 10, 49, 50
PCFG2:PCFG0 Bits ..... 50
ADRES Register ..... 9, 49, 51
Architecture
PIC16C62B/PIC16C72A Block Diagram ..... 5
Assembler
MPASM Assembler ..... 75

### B

Banking, Data Memory ..... 8, 11
Brown-out Reset (BOR) ..... 55, 57, 59, 60, 61
BOR Enable (BODEN Bit) ..... 55
BOR Status ($\overline{\text{BOR}}$ Bit) ..... 16
Timing Diagram ..... 92

### C

Capture (CCP Module) ..... 34
Block Diagram ..... 34
CCP Pin Configuration ..... 34
CCPR1H:CCPR1L Registers ..... 34
Changing Between Capture Prescalers ..... 34
Software Interrupt ..... 34
Timer1 Mode Selection ..... 34
Capture/Compare/PWM
Interaction of Two CCP Modules ..... 33
Capture/Compare/PWM (CCP) ..... 33
CCP1CON Register ..... 9, 33
CCPR1H Register ..... 9, 33
CCPR1L Register ..... 9, 33
Enable (CCP1IE Bit) ..... 14
Flag (CCP1IF Bit) ..... 15
RC2/CCP1 Pin ..... 6
Timer Resources ..... 33
Timing Diagram ..... 94
CCP1CON Register ..... 33
CCP1M3:CCP1M0 Bits ..... 33
CCP1X:CCP1Y Bits ..... 33
Code Protection ..... 55, 66
CP1:CP0 Bits ..... 55
Compare (CCP Module) ..... 35
Block Diagram ..... 35
CCP Pin Configuration ..... 35
CCPR1H:CCPR1L Registers ..... 35
Software Interrupt ..... 35
Special Event Trigger ..... 29, 35, 54
Timer1 Mode Selection ..... 35
Configuration Bits ..... 55
Conversion Considerations ..... 111

### D

Data Memory ..... 8
Bank Select (RP1:RP0 Bits) ..... 8, 11
General Purpose Registers ..... 8
Register File Map ..... 8
Special Function Registers ..... 9
DC Characteristics ..... 84, 86
Development Support ..... 75
Direct Addressing ..... 18

### E

Electrical Characteristics ..... 81
Errata ..... 3
External Power-on Reset Circuit ..... 59

### F

Firmware Instructions ..... 67

### I

I/O Ports ..... 19
I$^2$C (SSP Module) ..... 41
$\overline{\text{ACK}}$ Pulse ..... 41, 42, 43, 44, 45
Addressing ..... 42
Block Diagram ..... 41
Buffer Full Status (BF Bit) ..... 46
Clock Polarity Select (CKP Bit) ..... 47
Data/Address (D/$\overline{\text{A}}$ Bit) ..... 46
Master Mode ..... 45
Mode Select (SSPM3:SSPM0 Bits) ..... 47
Multi-Master Mode ..... 45
Read/Write Bit Information (R/$\overline{\text{W}}$ Bit) ..... 42, 43, 44, 46
Receive Overflow Indicator (SSPOV Bit) ..... 47
Reception ..... 43
Reception Timing Diagram ..... 43
Slave Mode ..... 41
Start (S Bit) ..... 45, 46
Stop (P Bit) ..... 45, 46
Synchronous Serial Port Enable (SSPEN Bit) ..... 47
Timing Diagram, Data ..... 100
Timing Diagram, Start/Stop Bits ..... 99
Transmission ..... 44
Update Address (UA Bit) ..... 46
ID Locations ..... 55, 66
In-Circuit Serial Programming (ICSP) ..... 55, 66
Indirect Addressing ..... 18
FSR Register ..... 8, 9, 18
INDF Register ..... 9
Instruction Format ..... 67

Instruction Set ..... 67
ADDLW ..... 69
ADDWF ..... 69
ANDLW ..... 69
ANDWF ..... 69
BCF ..... 69
BSF ..... 69
BTFSC ..... 70
BTFSS ..... 70
CALL ..... 70
CLRF ..... 70
CLRW ..... 70
CLRWDT ..... 70
COMF ..... 71
DECF ..... 71
DECFSZ ..... 71
GOTO ..... 71
INCF ..... 71
INCFSZ ..... 71
IORLW ..... 72
IORWF ..... 72
MOVF ..... 72
MOVLW ..... 72
MOVWF ..... 72
NOP ..... 72
RETFIE ..... 73
RETLW ..... 73
RETURN ..... 73
RLF ..... 73
RRF ..... 73
SLEEP ..... 73
SUBLW ..... 74
SUBWF ..... 74
SWAPF ..... 74
XORLW ..... 74
XORWF ..... 74
Summary Table ..... 68
INTCON Register ..... 9, 13
GIE Bit ..... 13
INTE Bit ..... 13
INTF Bit ..... 13
PEIE Bit ..... 13
RBIE Bit ..... 13
RBIF Bit ..... 13, 21
T0IE Bit ..... 13
T0IF Bit ..... 13
Interrupt Sources ..... 55, 62
A/D Conversion Complete ..... 51
Block Diagram ..... 62
Capture Complete (CCP) ..... 34
Compare Complete (CCP) ..... 35
Interrupt on Change (RB7:RB4 ) ..... 21
RB0/INT Pin, External ..... 6, 63
SSP Receive/Transmit Complete ..... 39
TMR0 Overflow ..... 26, 63
TMR1 Overflow ..... 27, 29
TMR2 to PR2 Match ..... 32
TMR2 to PR2 Match (PWM) ..... 31, 36
Interrupts, Context Saving During ..... 63
Interrupts, Enable Bits
A/D Converter Enable (ADIE Bit) ..... 14
CCP1 Enable (CCP1IE Bit) ..... 14
Global Interrupt Enable (GIE Bit) ..... 13, 62
Interrupt on Change (RB7:RB4)
Enable (RBIE Bit) ..... 13, 63
Peripheral Interrupt Enable (PEIE Bit) ..... 13
RB0/INT Enable (INTE Bit) ..... 13
SSP Enable (SSPIE Bit) ..... 14
TMR0 Overflow Enable (T0IE Bit) ..... 13
TMR1 Overflow Enable (TMR1IE Bit) ..... 14
TMR2 to PR2 Match Enable (TMR2IE Bit) ..... 14
Interrupts, Flag Bits
A/D Converter Flag (ADIF Bit) ..... 15, 51
CCP1 Flag (CCP1IF Bit) ..... 15, 34, 35
Interrupt on Change (RB7:RB4)
Flag (RBIF Bit) ..... 13, 21, 63
RB0/INT Flag (INTF Bit) ..... 13
SSP Flag (SSPIF Bit) ..... 15
TMR0 Overflow Flag (T0IF Bit) ..... 13, 63
TMR1 Overflow Flag (TMR1IF Bit) ..... 15
TMR2 to PR2 Match Flag (TMR2IF Bit) ..... 15

## K

KeeLoq® Evaluation and Programming Tools ..... 78

## M

Master Clear ($\overline{MCLR}$) ..... 6
$\overline{MCLR}$ Reset, Normal Operation ..... 57, 60, 61
$\overline{MCLR}$ Reset, SLEEP ..... 57, 60, 61
Memory Organization
Data Memory ..... 8
Program Memory ..... 7
MPLAB Integrated Development Environment Software .. 75

## O

OPCODE Field Descriptions ..... 67
OPTION_REG Register ..... 10, 12
INTEDG Bit ..... 12
PS2:PS0 Bits ..... 12, 25
PSA Bit ..... 12, 25
$\overline{RBPU}$ Bit ..... 12
T0CS Bit ..... 12, 25
T0SE Bit ..... 12, 25
OSC1/CLKIN Pin ..... 6
OSC2/CLKOUT Pin ..... 6
Oscillator Configuration ..... 55, 56
HS ..... 56, 60
LP ..... 56, 60
RC ..... 6, 57, 60
Selection (FOSC1:FOSC0 Bits) ..... 55
XT ..... 56, 60
Oscillator, Timer1 ..... 27, 29
Oscillator, WDT ..... 64

## P

Packaging ..... 105
Paging, Program Memory ..... 7, 17
PCON Register ..... 16, 60
$\overline{BOR}$ Bit ..... 16
$\overline{POR}$ Bit ..... 16
PICDEM-1 Low-Cost PIC MCU Demo Board ..... 77
PICDEM-2 Low-Cost PIC16CXX Demo Board ..... 77
PICDEM-3 Low-Cost PIC16CXXX Demo Board ..... 77
PICSTART® Plus Entry Level Development System ..... 77
PIE1 Register ..... 10, 14
ADIE Bit ..... 14
CCP1IE Bit ..... 14
SSPIE Bit ..... 14
TMR1IE Bit ..... 14
TMR2IE Bit ..... 14
Pinout Descriptions
PIC16C62B/PIC16C72A ..... 6

 © 1998-2013 Microchip Technology Inc.

PIR1 Register ..... 9, 15
ADIF Bit ..... 15
CCP1IF Bit ..... 15
SSPIF Bit ..... 15
TMR1IF Bit ..... 15
TMR2IF Bit ..... 15
Pointer, FSR ..... 18
PORTA ..... 6
Analog Port Pins ..... 6
PORTA Register ..... 9, 19
RA3:RA0 and RA5 Port Pins ..... 19
RA4/T0CKI Pin ..... 6, 19
RA5/SS/AN4 Pin ..... 6, 39
TRISA Register ..... 10, 19
PORTB ..... 6
PORTB Register ..... 9, 21
Pull-up Enable (RBPU Bit) ..... 12
RB0/INT Edge Select (INTEDG Bit) ..... 12
RB0/INT Pin, External ..... 6, 63
RB3:RB0 Port Pins ..... 21
RB7:RB4 Interrupt on Change ..... 63
RB7:RB4 Interrupt on Change Enable (RBIE Bit) ..... 13, 63
RB7:RB4 Interrupt on Change Flag (RBIF Bit) ..... 13, 21, 63
RB7:RB4 Port Pins ..... 21
TRISB Register ..... 10, 21
PORTC ..... 6
Block Diagram ..... 23
PORTC Register ..... 9, 23
RC0/T1OSO/T1CKI Pin ..... 6
RC1/T1OSI Pin ..... 6
RC2/CCP1 Pin ..... 6
RC3/SCK/SCL Pin ..... 6, 39
RC4/SDI/SDA Pin ..... 6, 39
RC5/SDO Pin ..... 6, 39
RC6 Pin ..... 6
RC7 Pin ..... 6
TRISC Register ..... 10, 23
Postscaler, Timer2
Select (TOUTPS3:TOUTPS0 Bits) ..... 31
Postscaler, WDT ..... 25
Assignment (PSA Bit) ..... 12, 25
Block Diagram ..... 26
Rate Select (PS2:PS0 Bits) ..... 12, 25
Switching Between Timer0 and WDT ..... 26
Power-on Reset (POR) ..... 55, 57, 59, 60, 61
Oscillator Start-up Timer (OST) ..... 55, 59
POR Status (POR Bit) ..... 16
Power Control (PCON) Register ..... 60
Power-down (PD Bit) ..... 11, 57
Power-on Reset Circuit, External ..... 59
Power-up Timer (PWRT) ..... 55, 59
PWRT Enable (PWRTE Bit) ..... 55
Time-out (TO Bit) ..... 11, 57
Time-out Sequence ..... 60
Timing Diagram ..... 92
Prescaler, Capture ..... 34
Prescaler, Timer0 ..... 25
Assignment (PSA Bit) ..... 12, 25
Block Diagram ..... 26
Rate Select (PS2:PS0 Bits) ..... 12, 25
Switching Between Timer0 and WDT ..... 26
Prescaler, Timer1 ..... 28
Select (T1CKPS1:T1CKPS0 Bits) ..... 27
Prescaler, Timer2 ..... 36
Select (T2CKPS1:T2CKPS0 Bits) ..... 31
PRO MATE® II Universal Programmer ..... 77
Program Counter
PCL Register ..... 9, 17
PCLATH Register ..... 9, 17, 63
Reset Conditions ..... 60
Program Memory ..... 7
Interrupt Vector ..... 7
Paging ..... 7, 17
Program Memory Map ..... 7
Reset Vector ..... 7
Program Verification ..... 66
Programming Pin (Vpp) ..... 6
Programming, Device Instructions ..... 67
PWM (CCP Module) ..... 36
Block Diagram ..... 36
CCPR1H:CCPR1L Registers ..... 36
Duty Cycle ..... 36
Example Frequencies/Resolutions ..... 37
Output Diagram ..... 36
Period ..... 36
Set-Up for PWM Operation ..... 37
TMR2 to PR2 Match ..... 31, 36
TMR2 to PR2 Match Enable (TMR2IE Bit) ..... 14
TMR2 to PR2 Match Flag (TMR2IF Bit) ..... 15

## Q

Q-Clock ..... 36

## R

Register File ..... 8
Register File Map ..... 8
Reset ..... 55, 57
Block Diagram ..... 58
Reset Conditions for All Registers ..... 61
Reset Conditions for PCON Register ..... 60
Reset Conditions for Program Counter ..... 60
Reset Conditions for STATUS Register ..... 60
Timing Diagram ..... 92
Revision History ..... 111

## S

SEEVAL® Evaluation and Programming System ..... 78
SLEEP ..... 55, 57, 65
Software Simulator (MPLAB-SIM) ..... 76
Special Features of the CPU ..... 55
Special Function Registers ..... 9
Speed, Operating ..... 1
SPI (SSP Module)
Block Diagram ..... 39
Buffer Full Status (BF Bit) ..... 46
Clock Edge Select (CKE Bit) ..... 46
Clock Polarity Select (CKP Bit) ..... 47
Data Input Sample Phase (SMP Bit) ..... 46
Mode Select (SSPM3:SSPM0 Bits) ..... 47
Receive Overflow Indicator (SSPOV Bit) ..... 47
Serial Clock (RC3/SCK/SCL) ..... 39
Serial Data In (RC4/SDI/SDA) ..... 39
Serial Data Out (RC5/SDO) ..... 39
Slave Select (RA5/SS/AN4) ..... 39
Synchronous Serial Port Enable (SSPEN Bit) ..... 47

SSP ..... 39
Enable (SSPIE Bit) ..... 14
Flag (SSPIF Bit) ..... 15
RA5/SS/AN4 Pin ..... 6
RC3/SCK/SCL Pin ..... 6
RC4/SDI/SDA Pin ..... 6
RC5/SDO Pin ..... 6
SSPADD Register ..... 10
SSPBUF Register ..... 9
SSPCON Register ..... 9, 47
SSPSTAT Register ..... 10, 46
TMR2 Output for Clock Shift ..... 32
Write Collision Detect (WCOL Bit) ..... 47
SSPCON Register ..... 47
CKP Bit ..... 47
SSPEN Bit ..... 47
SSPM3:SSPM0 Bits ..... 47
SSPOV Bit ..... 47
WCOL Bit ..... 47
SSPSTAT Register ..... 46
BF Bit ..... 46
CKE Bit ..... 46
D/A Bit ..... 46
P bit ..... 45, 46
R/W Bit ..... 42, 43, 44, 46
S Bit ..... 45, 46
SMP Bit ..... 46
UA Bit ..... 46
Stack ..... 17
STATUS Register ..... 9, 11, 63
C Bit ..... 11
DC Bit ..... 11
IRP Bit ..... 11
PD Bit ..... 11, 57
RP1:RP0 Bits ..... 11
TO Bit ..... 11, 57
Z Bit ..... 11

## T

T1CON Register ..... 9, 27
T1CKPS1:T1CKPS0 Bits ..... 27
T1OSCEN Bit ..... 27
T1SYNC Bit ..... 27
TMR1CS Bit ..... 27
TMR1ON Bit ..... 27
T2CON Register ..... 9, 31
T2CKPS1:T2CKPS0 Bits ..... 31
TMR2ON Bit ..... 31
TOUTPS3:TOUTPS0 Bits ..... 31
Timer0 ..... 25
Block Diagram ..... 25
Clock Source Edge Select (T0SE Bit) ..... 12, 25
Clock Source Select (T0CS Bit) ..... 12, 25
Overflow Enable (T0IE Bit) ..... 13
Overflow Flag (T0IF Bit) ..... 13, 63
Overflow Interrupt ..... 26, 63
RA4/T0CKI Pin, External Clock ..... 6
Timing Diagram ..... 93
TMR0 Register ..... 9
Timer1 ..... 27
Block Diagram ..... 28
Capacitor Selection ..... 29
Clock Source Select (TMR1CS Bit) ..... 27
External Clock Input Sync (T1SYNC Bit) ..... 27
Module On/Off (TMR1ON Bit) ..... 27
Oscillator ..... 27, 29
Oscillator Enable (T1OSCEN Bit) ..... 27
Overflow Enable (TMR1IE Bit) ..... 14
Overflow Flag (TMR1IF Bit) ..... 15
Overflow Interrupt ..... 27, 29
RC0/T1OSO/T1CKI Pin ..... 6
RC1/T1OSI ..... 6
Special Event Trigger (CCP) ..... 29, 35
T1CON Register ..... 9, 27
Timing Diagram ..... 93
TMR1H Register ..... 9
TMR1L Register ..... 9
Timer2
Block Diagram ..... 32
PR2 Register ..... 10, 31, 36
SSP Clock Shift ..... 32
T2CON Register ..... 9, 31
TMR2 Register ..... 9, 31
TMR2 to PR2 Match Enable (TMR2IE Bit) ..... 14
TMR2 to PR2 Match Flag (TMR2IF Bit) ..... 15
TMR2 to PR2 Match Interrupt ..... 31, 32, 36
Timing Diagrams
$I^2C$ Reception (7-bit Address) ..... 43
Wake-up from SLEEP via Interrupt ..... 66
Timing Diagrams and Specifications ..... 90
A/D Conversion ..... 102
Brown-out Reset (BOR) ..... 92
Capture/Compare/PWM (CCP) ..... 94
CLKOUT and I/O ..... 91
External Clock ..... 90
$I^2C$ Bus Data ..... 100
$I^2C$ Bus Start/Stop Bits ..... 99
Oscillator Start-up Timer (OST) ..... 92
Power-up Timer (PWRT) ..... 92
Reset ..... 2
Timer0 and Timer1 ..... 93
Watchdog Timer (WDT) ..... 92

## W

W Register ..... 63
Wake-up from SLEEP ..... 55, 65
Interrupts ..... 60, 61
MCLR Reset ..... 61
Timing Diagram ..... 66
WDT Reset ..... 61
Watchdog Timer (WDT) ..... 55, 64
Block Diagram ..... 64
Enable (WDTE Bit) ..... 55, 64
Programming Considerations ..... 64
RC Oscillator ..... 64
Timing Diagram ..... 92
WDT Reset, Normal Operation ..... 57, 60, 61
WDT Reset, SLEEP ..... 57, 60, 61
WWW, On-Line Support ..... 3

 © 1998-2013 Microchip Technology Inc.

## THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

## CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

## CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

**Technical support is available through the web site at: http://microchip.com/support**

## 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: 24xx010  Literature Number: DS35008C

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?


© 1998-2013 Microchip Technology Inc.

## PIC16C62B/72A 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**

| | | | |
|---|---|---|---|
| **Pattern:** | QTP, SQTP, Code or Special Requirements | | |
| **Package:** | JW | = | Windowed CERDIP |
| | SO | = | SOIC |
| | SP | = | Skinny plastic dip |
| | P | = | PDIP |
| | SS | = | SSOP |
| **Temperature Range:** | - | = | 0°C to +70°C |
| | I | = | -40°C to +85°C |
| | E | = | -40°C to +125°C |
| **Frequency Range:** | 04 | = | 4 MHz |
| | 10 | = | 10 MHz |
| | 20 | = | 20 MHz |
| **Device** | PIC16C62B: | | VDD range 4.0V to 5.5V |
| | PIC16C62BT: | | VDD range 4.0V to 5.5V (Tape/Reel) |
| | PIC16LC62B: | | VDD range 2.5V to 5.5V |
| | PIC16LC62BT: | | VDD range 2.5V to 5.5V (Tape/Reel) |

**Examples**

a) PIC16C72A-04/P 301 Commercial Temp., PDIP Package, 4 MHz, normal VDD limits, QTP pattern #301

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

### Sales and Support

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

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

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

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

 © 1913 Microchip Technology Inc.

**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 provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE**.** Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights.

**Trademarks**

The Microchip name and logo, the Microchip logo, dsPIC, FlashFlex, KEELOQ, KEELOQ logo, MPLAB, PIC, PICmicro, PICSTART, PIC$^{32}$ logo, rfPIC, SST, SST Logo, SuperFlash and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MTP, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

Analog-for-the-Digital Age, Application Maestro, BodyCom, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, HI-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Omniscient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICtail, REAL ICE, rfLAB, Select Mode, SQI, Serial Quad I/O, Total Endurance, TSHARC, UniWinDriver, WiperLock, ZENA and Z-Scale are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

GestIC and ULPP are registered trademarks of Microchip Technology Germany II GmbH & Co. & KG, a subsidiary of Microchip Technology Inc., in other countries.

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

© 1998-2013, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

Printed on recycled paper.

ISBN: 9781620769270

**QUALITY MANAGEMENT SYSTEM**
**CERTIFIED BY DNV**
**ISO/TS 16949**

*Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.*

# 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:
http://www.microchip.com/support
Web Address:
www.microchip.com

**Atlanta**
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

**Boston**
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

**Chicago**
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

**Cleveland**
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

**Dallas**
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

**Detroit**
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

**Indianapolis**
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

**Los Angeles**
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

**Santa Clara**
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

**Toronto**
Mississauga, Ontario, Canada
Tel: 905-673-0699
Fax: 905-673-6509

## ASIA/PACIFIC

**Asia Pacific Office**
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

**Australia - Sydney**
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

**China - Beijing**
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

**China - Chengdu**
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

**China - Chongqing**
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

**China - Hangzhou**
Tel: 86-571-2819-3187
Fax: 86-571-2819-3189

**China - Hong Kong SAR**
Tel: 852-2943-5100
Fax: 852-2401-3431

**China - Nanjing**
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

**China - Qingdao**
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

**China - Shanghai**
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

**China - Shenyang**
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

**China - Shenzhen**
Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

**China - Wuhan**
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

**China - Xian**
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

**China - Xiamen**
Tel: 86-592-2388138
Fax: 86-592-2388130

**China - Zhuhai**
Tel: 86-756-3210040
Fax: 86-756-3210049

## ASIA/PACIFIC

**India - Bangalore**
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

**India - New Delhi**
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

**India - Pune**
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

**Japan - Osaka**
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

**Japan - Tokyo**
Tel: 81-3-6880- 3770
Fax: 81-3-6880-3771

**Korea - Daegu**
Tel: 82-53-744-4301
Fax: 82-53-744-4302

**Korea - Seoul**
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

**Malaysia - Kuala Lumpur**
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

**Malaysia - Penang**
Tel: 60-4-227-8870
Fax: 60-4-227-4068

**Philippines - Manila**
Tel: 63-2-634-9065
Fax: 63-2-634-9069

**Singapore**
Tel: 65-6334-8870
Fax: 65-6334-8850

**Taiwan - Hsin Chu**
Tel: 886-3-5778-366
Fax: 886-3-5770-955

**Taiwan - Kaohsiung**
Tel: 886-7-213-7828
Fax: 886-7-330-9305

**Taiwan - Taipei**
Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

**Thailand - Bangkok**
Tel: 66-2-694-1351
Fax: 66-2-694-1350

## EUROPE

**Austria - Wels**
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

**Denmark - Copenhagen**
Tel: 45-4450-2828
Fax: 45-4485-2829

**France - Paris**
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

**Germany - Munich**
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

**Italy - Milan**
Tel: 39-0331-742611
Fax: 39-0331-466781

**Netherlands - Drunen**
Tel: 31-416-690399
Fax: 31-416-690340

**Spain - Madrid**
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

**UK - Wokingham**
Tel: 44-118-921-5869
Fax: 44-118-921-5820

11/29/12


© 1998-2013 Microchip Technology Inc.

## Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

- View PIC16C72AT-20I/SO on WIN SOURCE
- Microchip Technology Information

## Optimize Your Supply Chain with WIN SOURCE Solutions

- Global Sourcing Solution
- Obsolete Management
- Cost Control Management
- Shortage Management
- Alternative Solution
- Excess Inventory Management