**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Active |
| Core Processor | PIC |
| Core Size | 8-Bit |
| Speed | 20MHz |
| Connectivity | - |
| Peripherals | Brown-out Detect/Reset, POR, PWM, WDT |
| Number of I/O | 11 |
| Program Memory Size | 1.75KB (1K x 14) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 64 x 8 |
| Voltage - Supply (Vcc/Vdd) | 2V ~ 5.5V |
| Data Converters | - |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 14-SOIC (0.154", 3.90mm Width) |
| Supplier Device Package | 14-SOIC |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16f610-i-sl |

**PIC16F616/16HV616 14-Pin Diagram (PDIP, SOIC, TSSOP)**

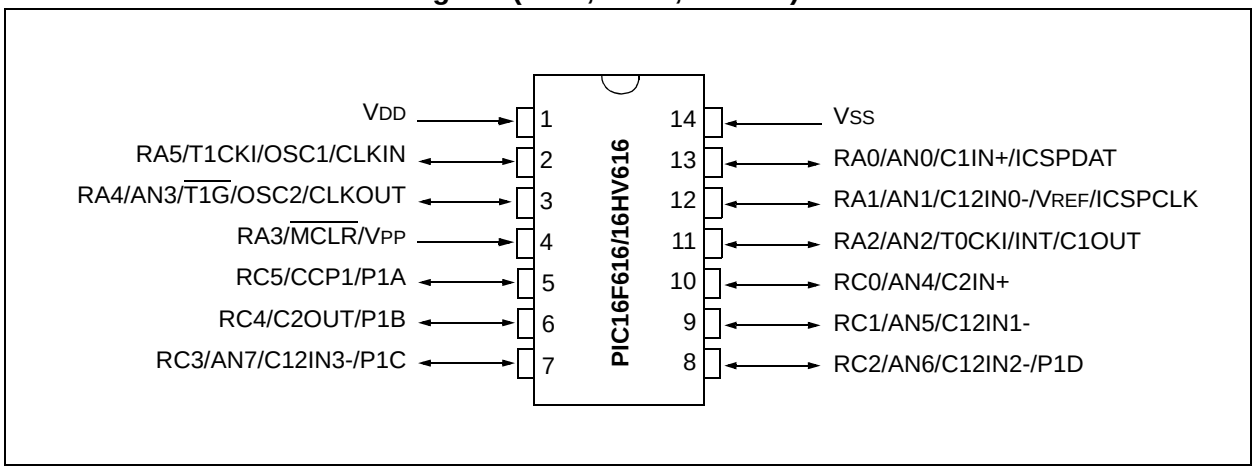



**TABLE 2: PIC16F616/16HV616 14-PIN SUMMARY**

| I/O | Pin | Analog | Comparators | Timer | CCP | Interrupts | Pull-ups | Basic |
|---|---|---|---|---|---|---|---|---|
| RA0 | 13 | AN0 | C1IN+ | — | — | IOC | Y | ICSPDAT |
| RA1 | 12 | AN1/VREF | C12IN0- | — | — | IOC | Y | ICSPCLK |
| RA2 | 11 | AN2 | C1OUT | T0CKI | — | INT/IOC | Y | — |
| RA3(1) | 4 | — | — | — | — | IOC | Y(2) | MCLR/VPP |
| RA4 | 3 | AN3 | — | T1G | — | IOC | Y | OSC2/CLKOUT |
| RA5 | 2 | — | — | T1CKI | — | IOC | Y | OSC1/CLKIN |
| RC0 | 10 | AN4 | C2IN+ | — | — | — | — | — |
| RC1 | 9 | AN5 | C12IN1- | — | — | — | — | — |
| RC2 | 8 | AN6 | C12IN2- | — | P1D | — | — | — |
| RC3 | 7 | AN7 | C12IN3- | — | P1C | — | — | — |
| RC4 | 6 | — | C2OUT | — | P1B | — | — | — |
| RC5 | 5 | — | — | — | CCP1/P1A | — | — | — |
| — | 1 | — | — | — | — | — | — | VDD |
| — | 14 | — | — | — | — | — | — | VSS |

**Note 1:** Input only.

**2:** Only when pin is configured for external MCLR.

## 1.0 DEVICE OVERVIEW

The PIC16F610/616/16HV610/616 is covered by this data sheet. It is available in 14-pin PDIP, SOIC, TSSOP and 16-pin QFN packages.

Block Diagrams and pinout descriptions of the devices are as follows:

- PIC16F610/16HV610 (Figure 1-1, Table 1-1)
- PIC16F616/16HV616 (Figure 1-2, Table 1-2)

**FIGURE 1-1: PIC16F610/16HV610 BLOCK DIAGRAM**

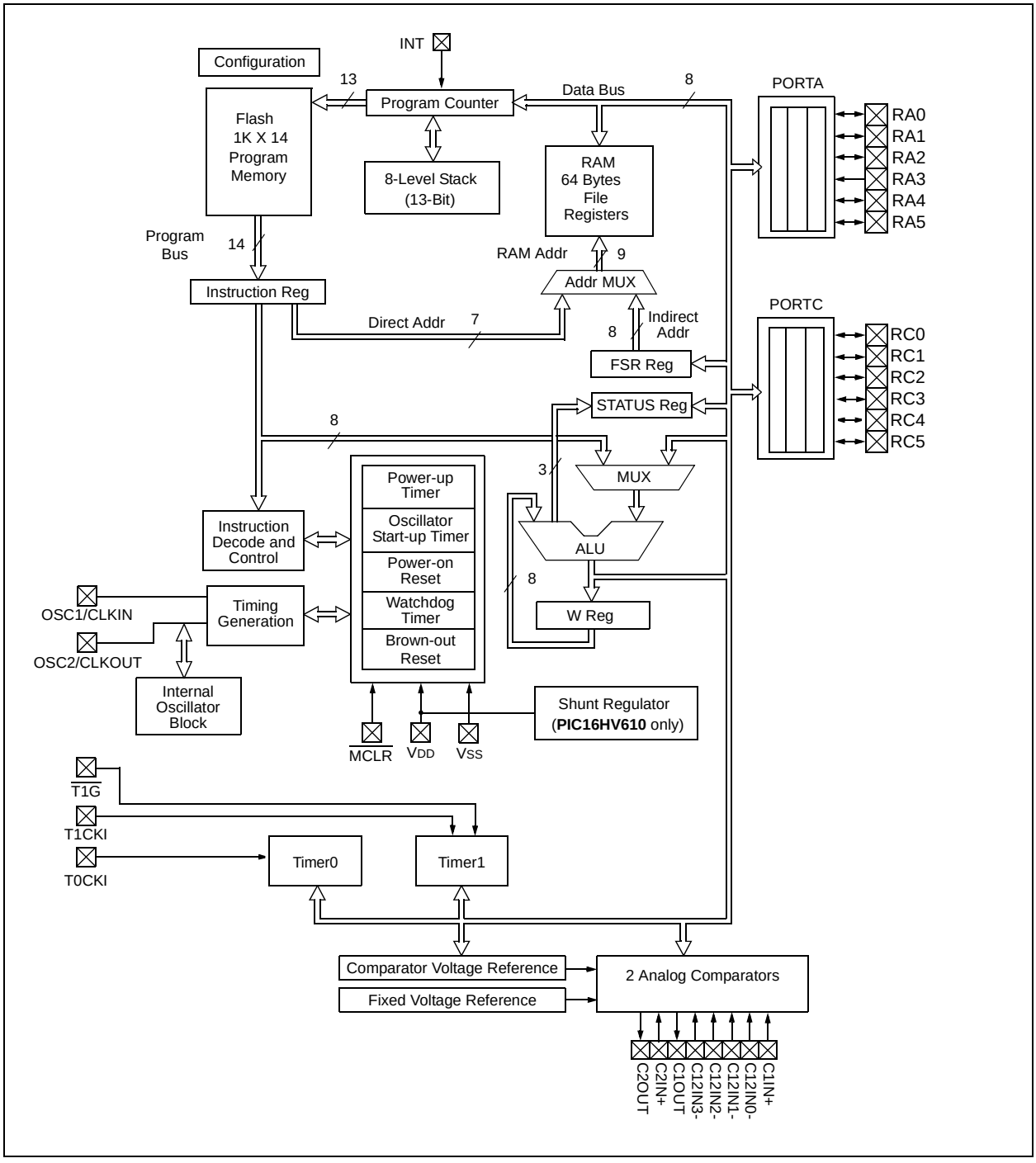
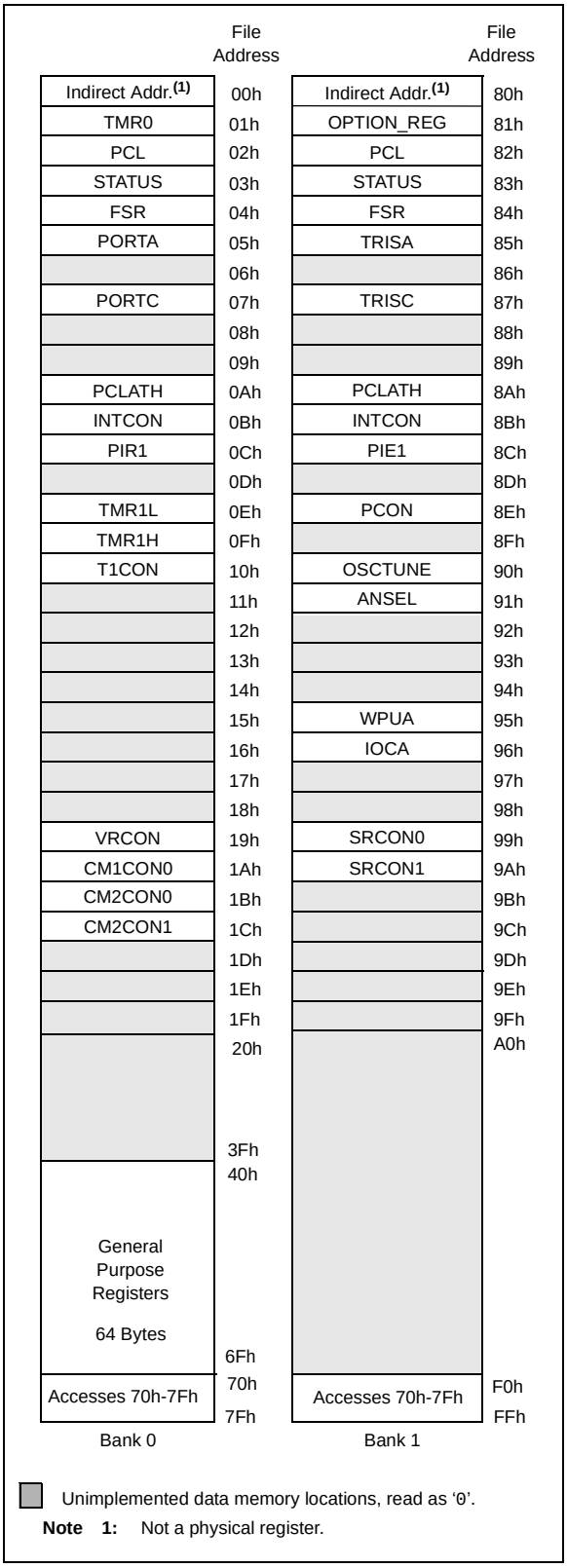


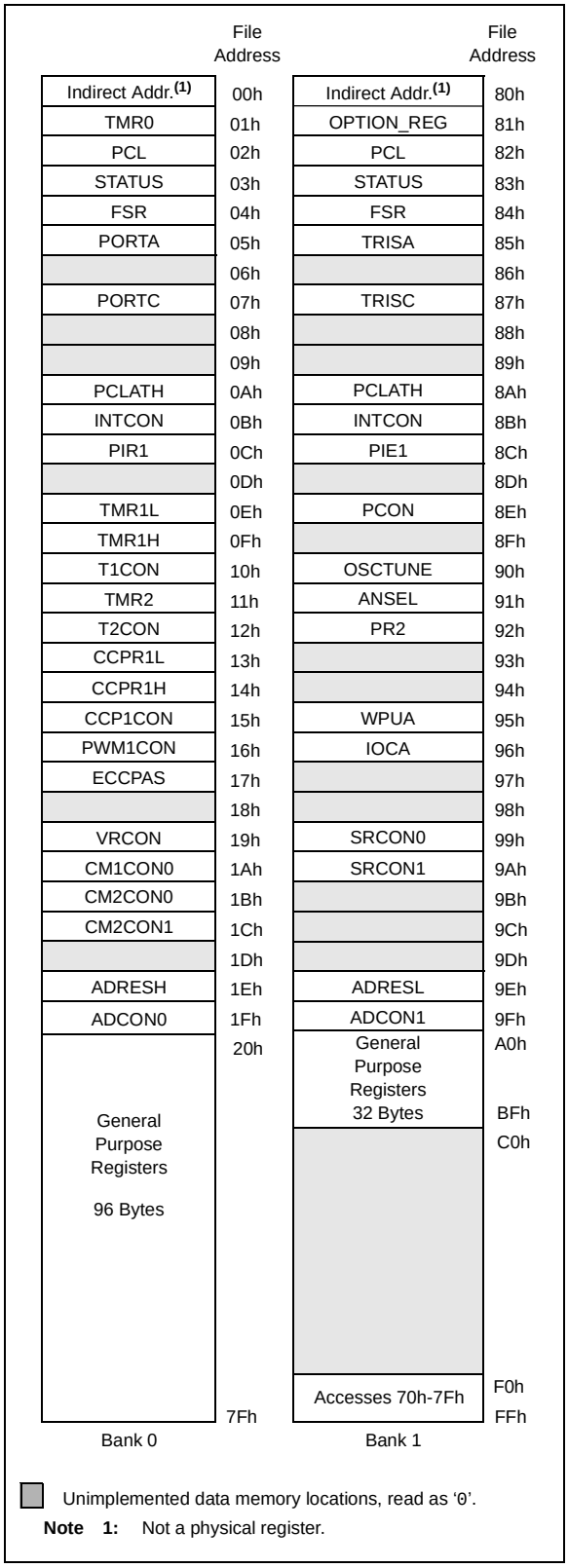
**FIGURE 2-3: DATA MEMORY MAP OF THE PIC16F610/16HV610**




Unimplemented data memory locations, read as '0'.

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

**FIGURE 2-4: DATA MEMORY MAP OF THE PIC16F616/16HV616**



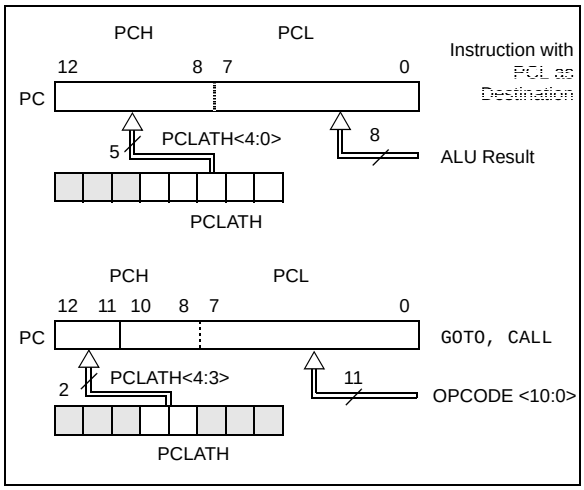

Unimplemented data memory locations, read as '0'.

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

## 2.3 PCL and PCLATH

The Program Counter (PC) is 13 bits wide. The low byte comes from the PCL register, which is a readable and writable register. The high byte (PC<12:8>) is not directly readable or writable and comes from PCLATH. On any Reset, the PC is cleared. Figure 2-5 shows the two situations for the loading of the PC. The upper example in Figure 2-5 shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in Figure 2-5 shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

**FIGURE 2-5: LOADING OF PC IN DIFFERENT SITUATIONS**




### 2.3.1 MODIFYING PCL

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

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

For more information refer to Application Note AN556, "*Implementing a Table Read*" (DS00556).

### 2.3.2 STACK

The PIC16F610/616/16HV610/616 Family has an 8-level x 13-bit wide hardware stack (see Figure 2-1). The stack space is not part of either program or data space and the Stack Pointer is not readable or writable. The PC is PUSHed onto the stack when a CALL instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not affected by a PUSH or POP operation.

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

| | |
|---|---|
| **Note 1:** | There are no Status bits to indicate stack overflow or stack underflow conditions. |
| **2:** | There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, RETURN, RETLW and RETFIE instructions or the vectoring to an interrupt address. |

## 2.4 Indirect Addressing, INDF and FSR Registers

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

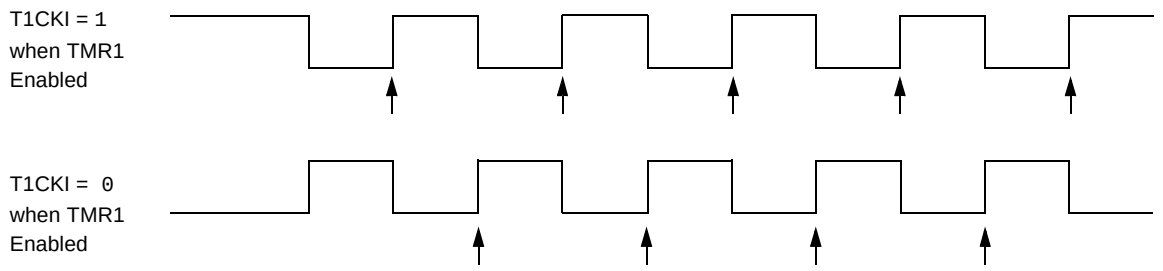
Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the File Select Register (FSR). Reading INDF itself indirectly will produce 00h. Writing to the INDF register indirectly results in a no operation (although Status bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR and the IRP bit of the STATUS register, as shown in Figure 2-7.

A simple program to clear RAM location 40h-4Fh using indirect addressing is shown in Example 2-1.

**EXAMPLE 2-1: INDIRECT ADDRESSING**

```
        MOVLW   0x40    ;initialize pointer
        MOVWF   FSR     ;to RAM
NEXT    CLRF    INDF    ;clear INDF register
        INCF    FSR, F  ;inc pointer
        BTFSS   FSR,4   ;all done?
        GOTO    NEXT    ;no clear next
CONTINUE                ;yes continue
```

**FIGURE 6-2: TIMER1 INCREMENTING EDGE**




**Note 1:** Arrows indicate counter increments.

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

## 6.12 Timer1 Control Register

The Timer1 Control register (T1CON), shown in Register 6-1, is used to control Timer1 and select the various features of the Timer1 module.

**REGISTER 6-1: T1CON: TIMER1 CONTROL REGISTER**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| T1GINV(1) | TMR1GE(2) | T1CKPS1 | T1CKPS0 | T1OSCEN | $\overline{\text{T1SYNC}}$ | TMR1CS | TMR1ON |
| bit 7 | | | | | | | bit 0 |

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

bit 7 **T1GINV:** Timer1 Gate Invert bit(1)
1 = Timer1 gate is active-high (Timer1 counts when gate is high)
0 = Timer1 gate is active-low (Timer1 counts when gate is low)

bit 6 **TMR1GE:** Timer1 Gate Enable bit(2)
If TMR1ON = 0:
This bit is ignored
If TMR1ON = 1:
1 = Timer1 counting is controlled by the Timer1 Gate function
0 = Timer1 is always counting

bit 5-4 **T1CKPS<1:0>:** Timer1 Input Clock Prescale Select bits
11 = 1:8 Prescale Value
10 = 1:4 Prescale Value
01 = 1:2 Prescale Value
00 = 1:1 Prescale Value

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

## 8.10 Comparator SR Latch

The SR latch module provides additional control of the comparator outputs. The module consists of a single SR latch and output multiplexers. The SR latch can be set, reset or toggled by the comparator outputs. The SR latch may also be set or reset, independent of comparator output, by control bits in the SRCON0 control register. The SR latch output multiplexers select whether the latch outputs or the comparator outputs are directed to the I/O port logic for eventual output to a pin.

The SR latch also has a variable clock, which is connected to the set input of the latch. The SRCLKEN bit of SRCON0 enables the SR latch set clock. The clock will periodically pulse the set input of the latch. Control over the frequency of the SR latch set clock is provided by the SRCS<1:0> bits of SRCON1 register.

### 8.10.1 LATCH OPERATION

The latch is a Set-Reset latch that does not depend on a clock source. Each of the Set and Reset inputs are active-high. Each latch input is connected to a comparator output and a software controlled pulse generator. The latch can be set by C1OUT or the PULSS bit of the SRCON0 register. The latch can be reset by C2OUT or the PULSR bit of the SRCON0 register. The latch is reset-dominant, therefore, if both Set and Reset inputs are high the latch will go to the Reset state. Both the PULSS and PULSR bits are self resetting which means that a single write to either of the bits is all that is necessary to complete a latch Set or Reset operation.

### 8.10.2 LATCH OUTPUT

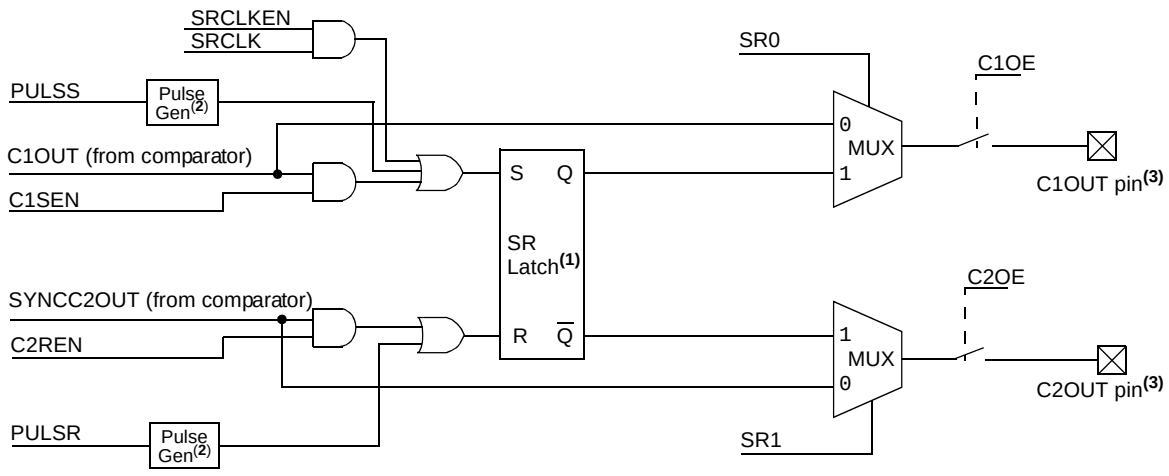
The SR<1:0> bits of the SRCON0 register control the latch output multiplexers and determine four possible output configurations. In these four configurations, the CxOUT I/O port logic is connected to:

- C1OUT and C2OUT
- C1OUT and SR latch $\overline{Q}$
- C2OUT and SR latch Q
- SR latch Q and $\overline{Q}$

After any Reset, the default output configuration is the unlatched C1OUT and C2OUT mode. This maintains compatibility with devices that do not have the SR latch feature.

The applicable TRIS bits of the corresponding ports must be cleared to enable the port pin output drivers. Additionally, the CxOE comparator output enable bits of the CMxCON0 registers must be set in order to make the comparator or latch outputs available on the output pins. The latch configuration enable states are completely independent of the enable states for the comparators.

**FIGURE 8-8: SR LATCH SIMPLIFIED BLOCK DIAGRAM**




**Note 1:** If R = 1 and S = 1 simultaneously, Q = 0, $\overline{Q}$ = 1
**2:** Pulse generator causes a 1 TOSC pulse width.
**3:** Output shown for reference only. See I/O port pin block diagram for more detail.

**REGISTER 8-4: SRCON0: SR LATCH CONTROL 0 REGISTER**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/S-0 | R/S-0 | U-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| SR1(2) | SR0(2) | C1SEN | C2REN | PULSS | PULSR | — | SRCLKEN |
| bit 7 | | | | | | | bit 0 |

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

bit 7 **SR1:** SR Latch Configuration bit(2)
- 1 = C2OUT pin is the latch $\overline{Q}$ output
- 0 = C2OUT pin is the C2 comparator output

bit 6 **SR0:** SR Latch Configuration bits(2)
- 1 = C1OUT pin is the latch Q output
- 0 = C1OUT pin is the C1 Comparator output

bit 5 **C1SEN:** C1 Set Enable bit
- 1 = C1 comparator output sets SR latch
- 0 = C1 comparator output has no effect on SR latch

bit 4 **C2REN:** C2 Reset Enable bit
- 1 = C2 comparator output resets SR latch
- 0 = C2 comparator output has no effect on SR latch

bit 3 **PULSS:** Pulse the SET Input of the SR Latch bit
- 1 = Triggers pulse generator to set SR latch. Bit is immediately reset by hardware.
- 0 = Does not trigger pulse generator

bit 2 **PULSR:** Pulse the Reset Input of the SR Latch bit
- 1 = Triggers pulse generator to reset SR latch. Bit is immediately reset by hardware.
- 0 = Does not trigger pulse generator

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

bit 0 **SRCLKEN:** SR Latch Set Clock Enable bit
- 1 = Set input of SR latch is pulsed with SRCLK
- 0 = Set input of SR latch is not pulsed with the SRCLK

**Note 1:** The C1OUT and C2OUT bits in the CMxCON0 register will always reflect the actual comparator output (not the level on the pin), regardless of the SR latch operation.

**2:** To enable an SR Latch output to the pin, the appropriate CxOE, and TRIS bits must be properly configured.

**REGISTER 8-5: SRCON1: SR LATCH CONTROL 1 REGISTER**

| R/W-0 | R/W-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| SRCS1 | SRCS0 | — | — | — | — | — | — |
| bit 7 | | | | | | | bit 0 |

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

bit 7-6 **SRCS<1:0>:** SR Latch Clock Prescale bits
- 00 = Fosc/16
- 01 = Fosc/32
- 10 = Fosc/64
- 11 = Fosc/128

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

### 9.2.7 ADC REGISTER DEFINITIONS

The following registers are used to control the operation of the ADC.

**REGISTER 9-1: ADCON0: A/D CONTROL REGISTER 0**

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| ADFM | VCFG | CHS3 | CHS2 | CHS1 | CHS0 | GO/$\overline{\text{DONE}}$ | ADON |
| bit 7 | | | | | | | bit 0 |

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

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

bit 6 **VCFG:** Voltage Reference bit
1 = $V_{REF}$ pin
0 = $V_{DD}$

bit 5-2 **CHS<3:0>:** Analog Channel Select bits
0000 = Channel 00 (AN0)
0001 = Channel 01 (AN1)
0010 = Channel 02 (AN2)
0011 = Channel 03 (AN3)
0100 = Channel 04 (AN4)
0101 = Channel 05 (AN5)
0110 = Channel 06 (AN6)
0111 = Channel 07 (AN7)
1000 = Reserved – do not use
1001 = Reserved – do not use
1010 = Reserved – do not use
1011 = Reserved – do not use
1100 = $CV_{REF}$
1101 = 0.6V Fixed Voltage Reference(1)
1110 = 1.2V Fixed Voltage Reference(1)
1111 = Reserved – do not use

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

bit 0 **ADON:** ADC Enable bit
1 = ADC is enabled
0 = ADC is disabled and consumes no operating current

**Note 1:** When the CHS<3:0> bits change to select the 1.2V or 0.6V Fixed Voltage Reference, the reference output voltage will have a transient. If the Comparator module uses this VP6 reference voltage, the comparator output may momentarily change state due to the transient.

#### 10.4.2.1 Direction Change in Full-Bridge Mode

In the Full-Bridge mode, the P1M1 bit in the CCP1CON register allows users to control the forward/reverse direction. When the application firmware changes this direction control bit, the module will change to the new direction on the next PWM cycle.

A direction change is initiated in software by changing the P1M1 bit of the CCP1CON register. The following sequence occurs four Timer2 cycles prior to the end of the current PWM period:

- The modulated outputs (P1B and P1D) are placed in their inactive state.
- The associated unmodulated outputs (P1A and P1C) are switched to drive in the opposite direction.
- PWM modulation resumes at the beginning of the next period.

See Figure 10-12 for an illustration of this sequence.

The Full-Bridge mode does not provide dead-band delay. As one output is modulated at a time, dead-band delay is generally not required. There is a situation where dead-band delay is required. This situation occurs when both of the following conditions are true:

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

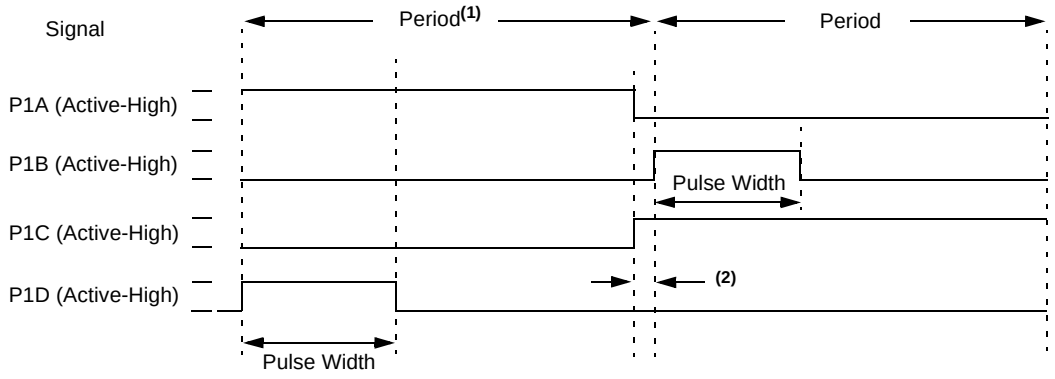
Figure 10-13 shows an example of the PWM direction changing from forward to reverse, at a near 100% duty cycle. In this example, at time t1, the output P1A and P1D become inactive, while output P1C becomes active. Since the turn off time of the power devices is longer than the turn on time, a shoot-through current will flow through power devices QC and QD (see Figure 10-10) for the duration of ‘t’. The same phenomenon will occur to power devices QA and QB for PWM direction change from reverse to forward.

If changing PWM direction at high duty cycle is required for an application, two possible solutions for eliminating the shoot-through current are:

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

Other options to prevent shoot-through current may exist.

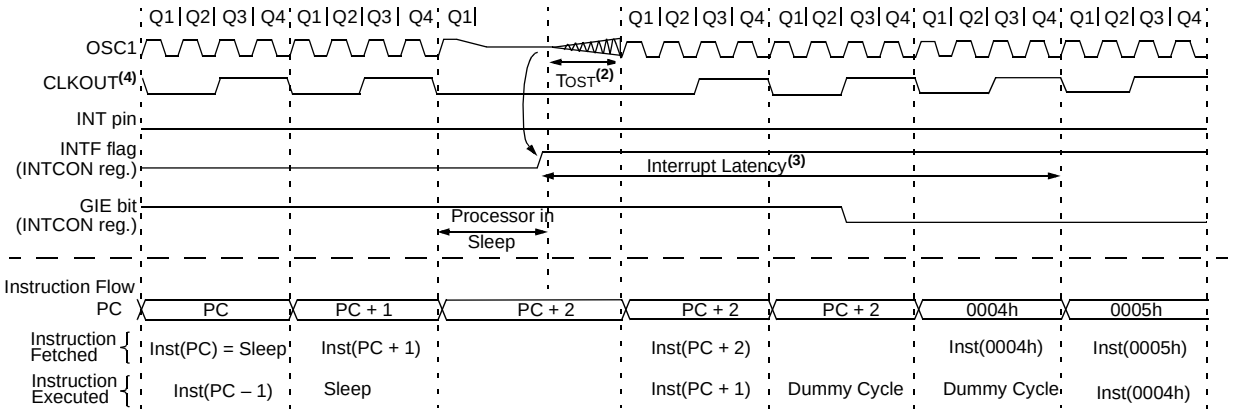
**FIGURE 10-12: EXAMPLE OF PWM DIRECTION CHANGE**




**Note 1:** The direction bit P1M1 of the CCP1CON register is written any time during the PWM cycle.

**2:** When changing directions, the P1A and P1C signals switch before the end of the current PWM cycle. The modulated P1B and P1D signals are inactive at this time. The length of this time is four Timer2 counts.

**FIGURE 12-9: WAKE-UP FROM SLEEP THROUGH INTERRUPT**




**Note 1:** XT, HS or LP Oscillator mode assumed.

**2:** TOST = 1024 TOSC (drawing not to scale). This delay does not apply to EC, INTOSC and RC Oscillator modes.

**3:** GIE = '1' assumed. In this case after wake-up, the processor jumps to 0004h. If GIE = '0', execution will continue in-line.

**4:** CLKOUT is not available in XT, HS, LP or EC Oscillator modes, but shown here for timing reference.

## 12.8 Code Protection

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

| **Note:** | The entire Flash program memory will be erased when the code protection is turned off. See the *Memory Programming Specification* (DS41284) for more information. |
|---|---|

## 12.9 ID Locations

Four memory locations (2000h-2003h) are designated as ID locations where the user can store checksum or other code identification numbers. These locations are not accessible during normal execution but are readable and writable during Program/Verify mode. Only the Least Significant 7 bits of the ID locations are used.

## 13.2 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. |

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

## 14.2 MPLAB C Compilers for Various Device Families

The MPLAB C Compiler code development systems are complete ANSI C compilers for Microchip's PIC18, PIC24 and PIC32 families of microcontrollers and the dsPIC30 and dsPIC33 families of digital signal controllers. These compilers provide powerful integration capabilities, superior code optimization and ease of use.

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

## 14.3 HI-TECH C for Various Device Families

The HI-TECH C Compiler code development systems are complete ANSI C compilers for Microchip's PIC family of microcontrollers and the dsPIC family of digital signal controllers. These compilers provide powerful integration capabilities, omniscient code generation and ease of use.

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

The compilers include a macro assembler, linker, preprocessor, and one-step driver, and can run on multiple platforms.

## 14.4 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

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

The MPASM Assembler features include:

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

## 14.5 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler and the MPLAB C18 C Compiler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

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

The object linker/library features include:

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

## 14.6 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC devices. MPLAB C Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

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

## 15.8 DC Characteristics: PIC16F610/616/16HV610/616- I (Industrial) PIC16F610/616/16HV610/616 - E (Extended)

| DC CHARACTERISTICS | | | Standard Operating Conditions (unless otherwise stated)<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial<br>-40°C ≤ TA ≤ +125°C for extended | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Sym** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| | VIL | **Input Low Voltage** | | | | | |
| | | I/O port: | | | | | |
| D030 | | with TTL buffer | Vss | — | 0.8 | V | 4.5V ≤ VDD ≤ 5.5V |
| D030A | | | Vss | — | 0.15 VDD | V | 2.0V ≤ VDD ≤ 4.5V |
| D031 | | with Schmitt Trigger buffer | Vss | — | 0.2 VDD | V | 2.0V ≤ VDD ≤ 5.5V |
| D032 | | $\overline{MCLR}$, OSC1 (RC mode) | Vss | — | 0.2 VDD | V | |
| D033 | | OSC1 (XT and LP modes) | Vss | — | 0.3 | V | |
| D033A | | OSC1 (HS mode) | Vss | — | 0.3 VDD | V | |
| | VIH | **Input High Voltage** | | | | | |
| | | I/O ports: | | — | | | |
| D040 | | with TTL buffer | 2.0 | — | VDD | V | 4.5V ≤ VDD ≤ 5.5V |
| D040A | | | 0.25 VDD + 0.8 | — | VDD | V | 2.0V ≤ VDD ≤ 4.5V |
| D041 | | with Schmitt Trigger buffer | 0.8 VDD | — | VDD | V | 2.0V ≤ VDD ≤ 5.5V |
| D042 | | $\overline{MCLR}$ | 0.8 VDD | — | VDD | V | |
| D043 | | OSC1 (XT and LP modes) | 1.6 | — | VDD | V | |
| D043A | | OSC1 (HS mode) | 0.7 VDD | — | VDD | V | |
| D043B | | OSC1 (RC mode) | 0.9 VDD | — | VDD | V | **(Note 1)** |
| | IIL | **Input Leakage Current(2,3)** | | | | | |
| D060 | | I/O ports | — | ± 0.1 | ± 1 | µA | VSS ≤ VPIN ≤ VDD,<br>Pin at high-impedance |
| D061 | | RA3/$\overline{MCLR}$(3,4) | — | ± 0.7 | ± 5 | µA | VSS ≤ VPIN ≤ VDD |
| D063 | | OSC1 | — | ± 0.1 | ± 5 | µA | VSS ≤ VPIN ≤ VDD, XT, HS and LP oscillator configuration |
| D070* | IPUR | **PORTA Weak Pull-up Current(5)** | 50 | 250 | 400 | µA | VDD = 5.0V, VPIN = VSS |
| | VOL | **Output Low Voltage** | — | — | 0.6 | V | IOL = 7.0 mA, VDD = 4.5V,<br>-40°C to +125°C |
| D080 | | I/O ports | — | — | 0.6 | V | IOL = 8.5 mA, VDD = 4.5V,<br>-40°C to +85°C |
| | VOH | **Output High Voltage** | VDD – 0.7 | — | — | V | IOH = -2.5 mA, VDD = 4.5V,<br>-40°C to +125°C |
| D090 | | I/O ports(2) | VDD – 0.7 | — | — | V | IOH = -3.0 mA, VDD = 4.5V,<br>-40°C to +85°C |

* These parameters are characterized but not tested.

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

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

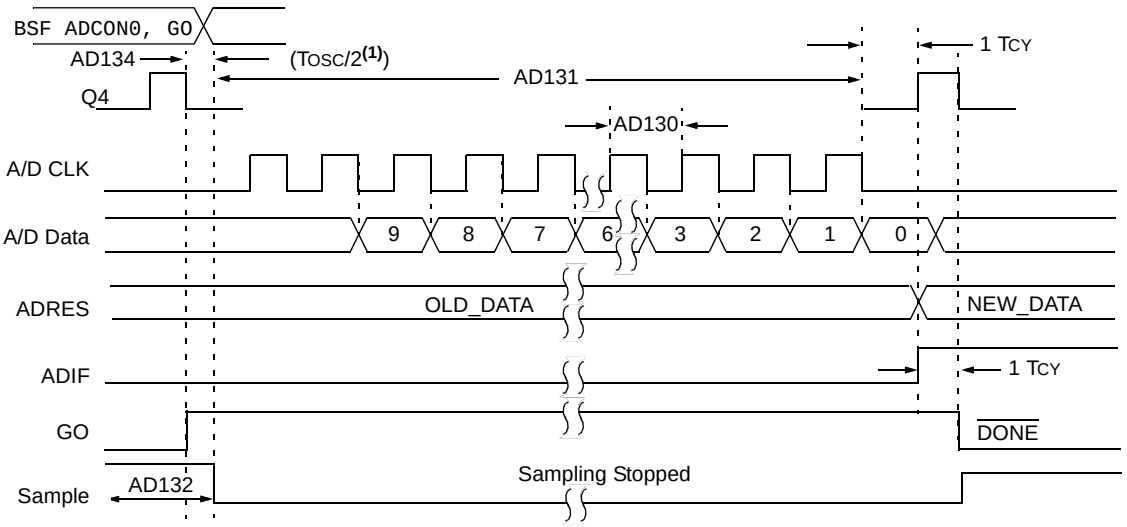
**2:** Negative current is defined as current sourced by the pin.

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

**4:** This specification applies to RA3/$\overline{MCLR}$ configured as RA3 input with internal pull-up disabled.

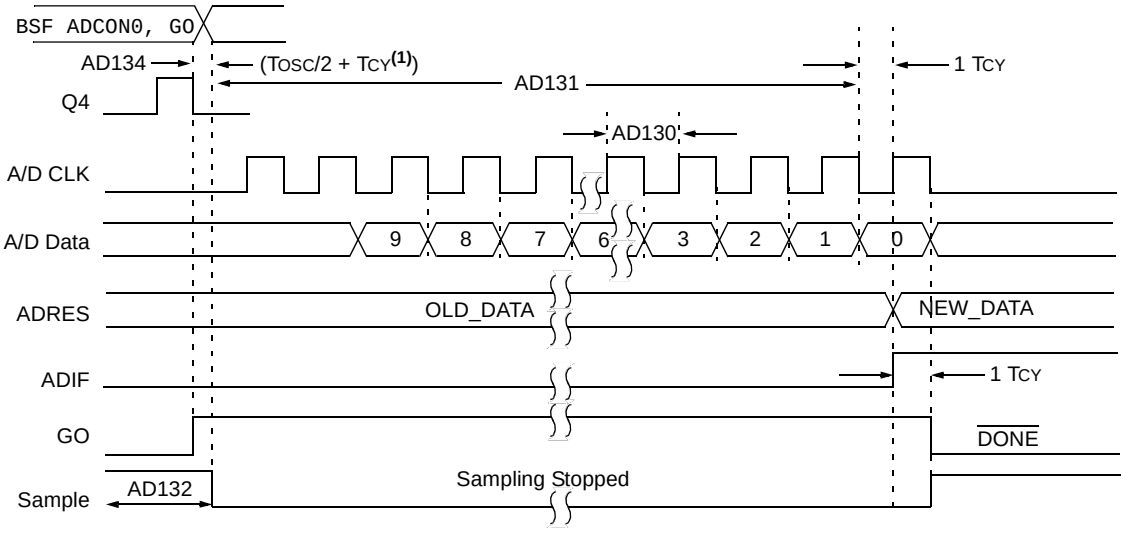
**5:** This specification applies to all weak pull-up pins, including the weak pull-up on RA3/$\overline{MCLR}$. When RA3/$\overline{MCLR}$ is configured as $\overline{MCLR}$ reset pin, the weak pull-up is always enabled.

**FIGURE 15-12: PIC16F616/16HV616 A/D CONVERSION TIMING (NORMAL MODE)**




**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.

**FIGURE 15-13: PIC16F616/16HV616 A/D CONVERSION TIMING (SLEEP MODE)**




**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 15-14: DC CHARACTERISTICS FOR IDD SPECIFICATIONS FOR PIC16F616 – H (High Temp.)**

| Param No. | Device Characteristics | Units | Min | Typ | Max | Condition | |
|---|---|---|---|---|---|---|---|
| | | | | | | VDD | Note |
| D010 | Supply Current (IDD) | μA | — | 13 | 58 | 2.0 | IDD LP OSC (32 kHz) |
| | | | — | 19 | 67 | 3.0 | |
| | | | — | 32 | 92 | 5.0 | |
| D011 | | μA | — | 135 | 316 | 2.0 | IDD XT OSC (1 MHz) |
| | | | — | 185 | 400 | 3.0 | |
| | | | — | 300 | 537 | 5.0 | |
| D012 | | μA | — | 240 | 495 | 2.0 | IDD XT OSC (4 MHz) |
| | | | — | 360 | 680 | 3.0 | |
| | | mA | — | 0.660 | 1.20 | 5.0 | |
| D013 | | μA | — | 75 | 158 | 2.0 | IDD EC OSC (1 MHz) |
| | | | — | 155 | 338 | 3.0 | |
| | | | — | 345 | 792 | 5.0 | |
| D014 | | μA | — | 185 | 357 | 2.0 | IDD EC OSC (4 MHz) |
| | | | — | 325 | 625 | 3.0 | |
| | | mA | — | 0.665 | 1.30 | 5.0 | |
| D016 | | μA | — | 245 | 476 | 2.0 | IDD INTOSC (4 MHz) |
| | | | — | 360 | 672 | 3.0 | |
| | | | — | 620 | 1.10 | 5.0 | |
| D017 | | μA | — | 395 | 757 | 2.0 | IDD INTOSC (8 MHz) |
| | | mA | — | 0.620 | 1.20 | 3.0 | |
| | | | — | 1.20 | 2.20 | 5.0 | |
| D018 | | μA | — | 175 | 332 | 2.0 | IDD EXTRC (4 MHz) |
| | | | — | 285 | 518 | 3.0 | |
| | | | — | 530 | 972 | 5.0 | |
| D019 | | mA | — | 2.20 | 4.10 | 4.5 | IDD HS OSC (20 MHz) |
| | | | — | 2.80 | 4.80 | 5.0 | |

**FIGURE 16-6: PIC16F610/616 IDD INTOSC (4 MHz) vs. VDD**

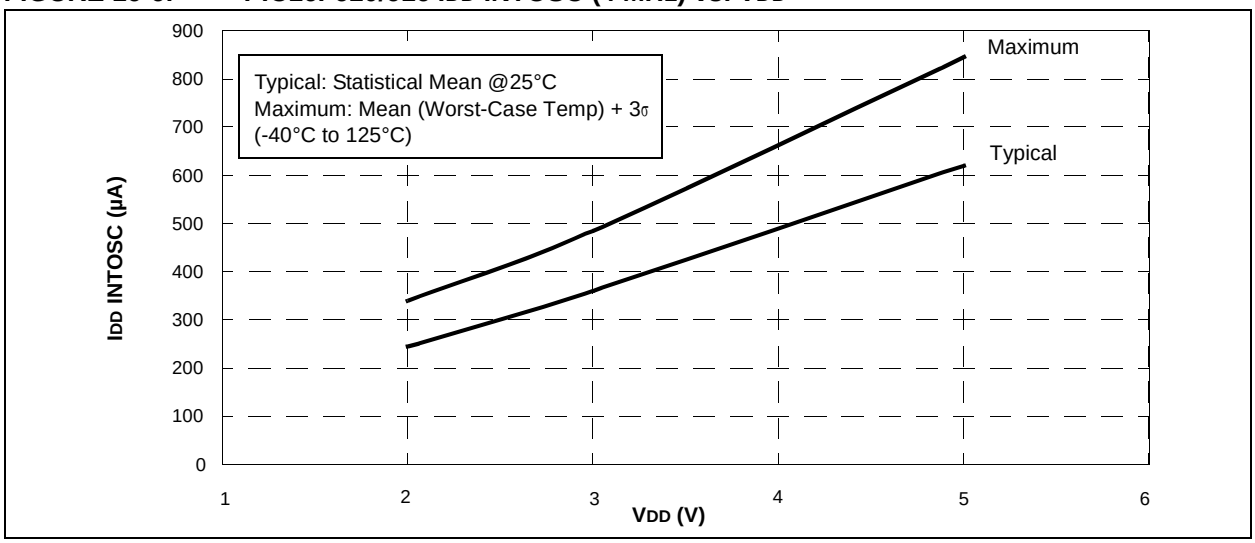



**FIGURE 16-7: PIC16F610/616 IDD INTOSC (8 MHz) vs. VDD**

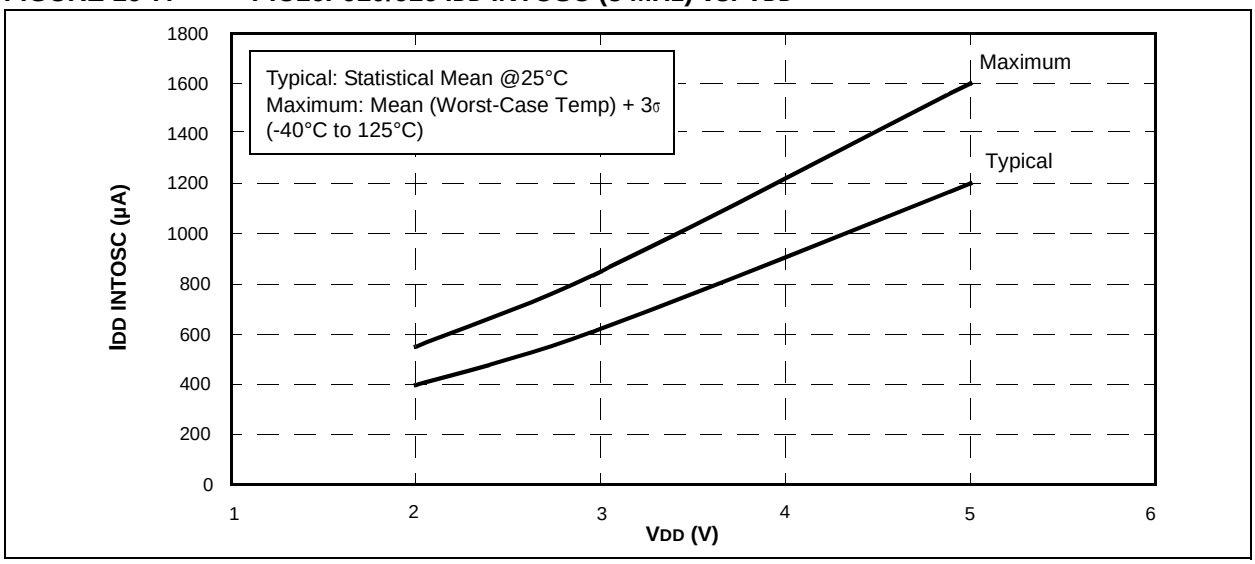


**FIGURE 16-47: TYPICAL HFINTOSC FREQUENCY CHANGE vs. VDD (125°C)**

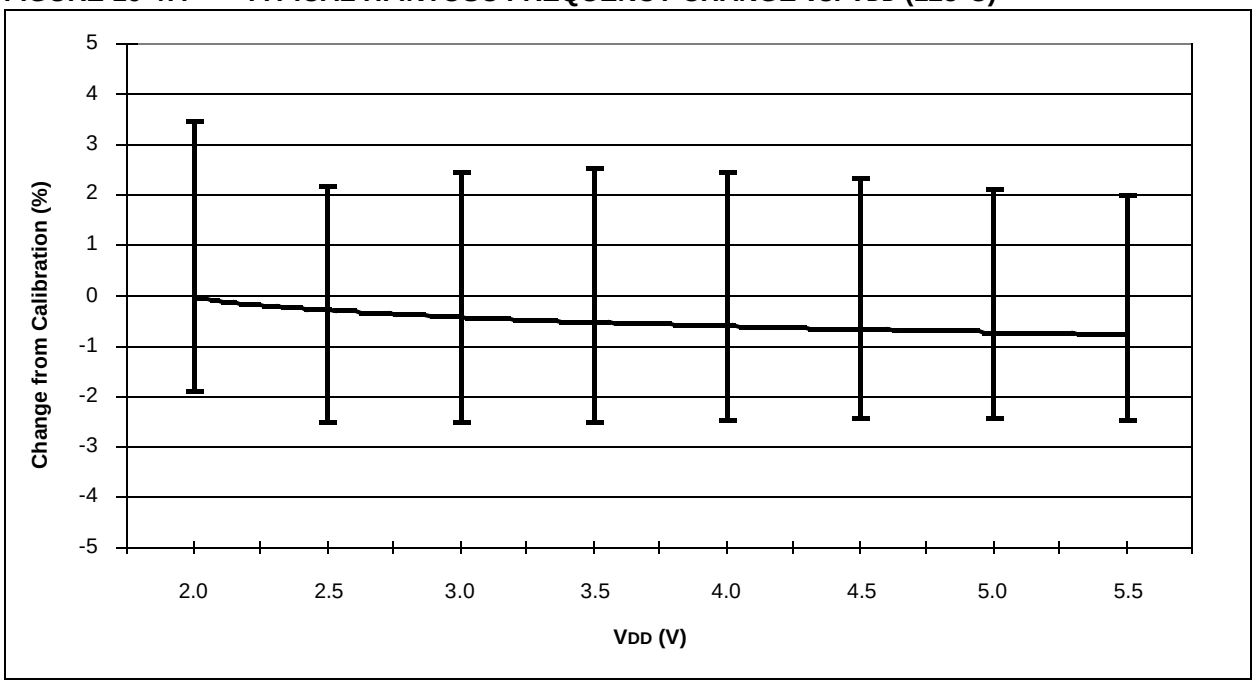
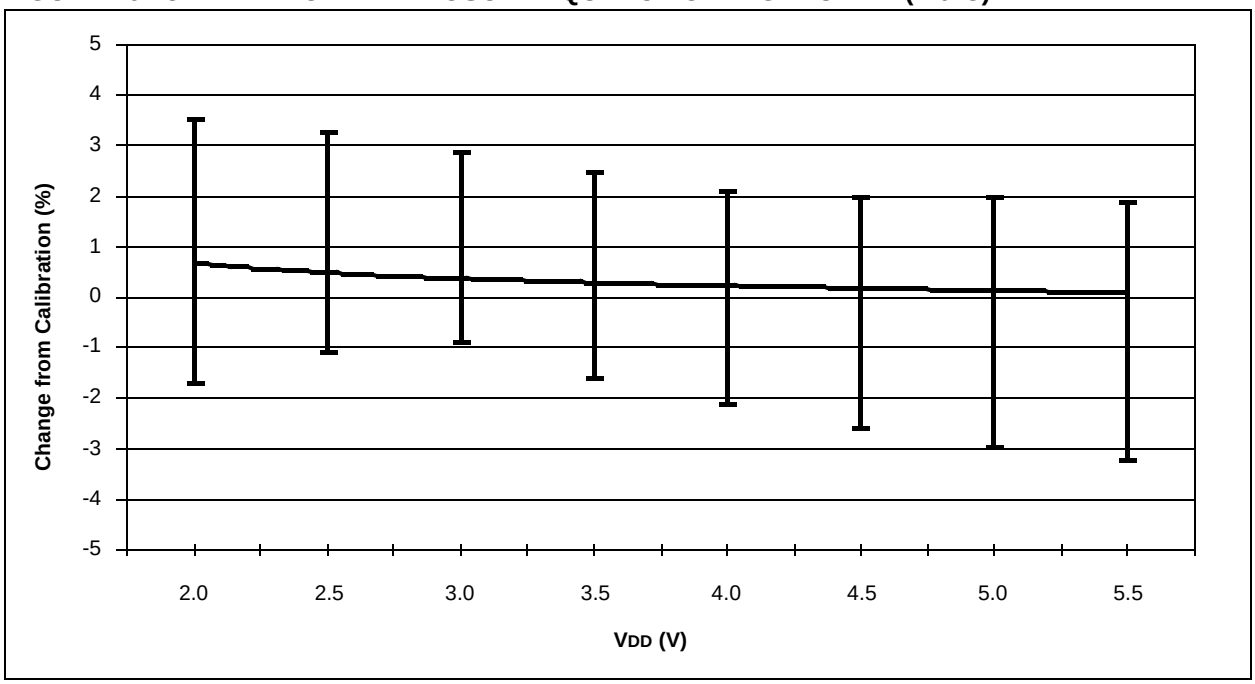
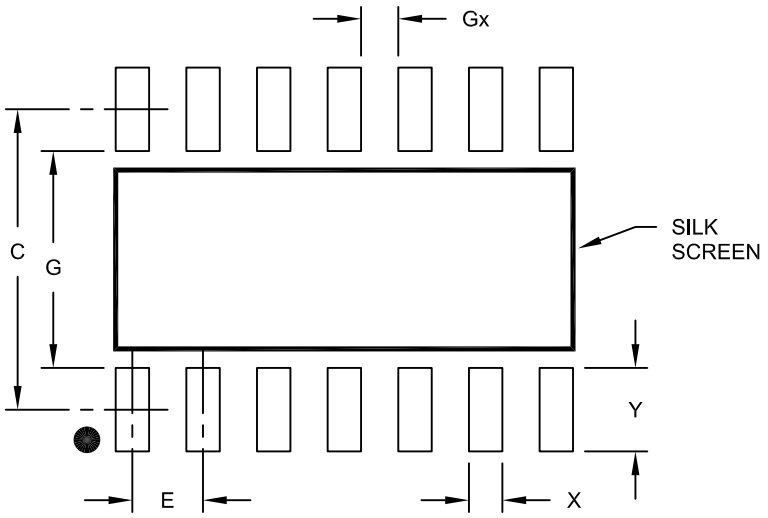



**FIGURE 16-48: TYPICAL HFINTOSC FREQUENCY CHANGE vs. VDD (-40°C)**



14-Lead Plastic Small Outline (SL) - Narrow, 3.90 mm Body [SOIC]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




RECOMMENDED LAND PATTERN

| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Contact Pitch | E | 1.27 BSC | | |
| Contact Pad Spacing | C | | 5.40 | |
| Contact Pad Width | X | | | 0.60 |
| Contact Pad Length | Y | | | 1.50 |
| Distance Between Pads | Gx | 0.67 | | |
| Distance Between Pads | G | 3.90 | | |

Notes:

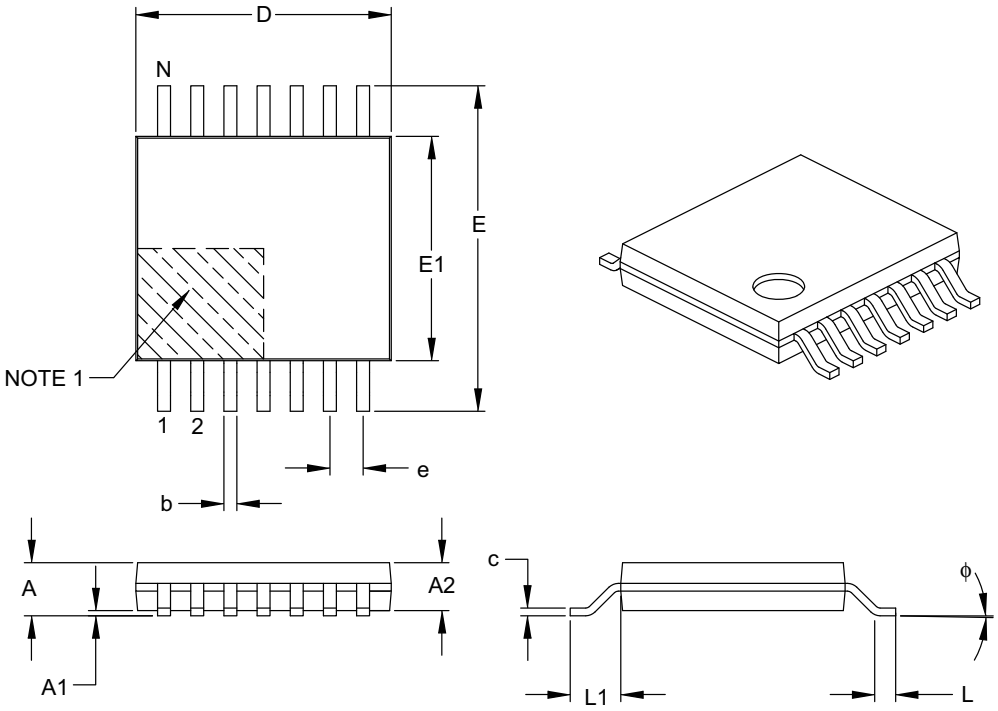
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2065A

## 14-Lead Plastic Thin Shrink Small Outline (ST) – 4.4 mm Body [TSSOP]

**Note:** For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging




| Units | | MILLIMETERS | | |
|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX |
| Number of Pins | N | | 14 | |
| Pitch | e | | 0.65 BSC | |
| Overall Height | A | – | – | 1.20 |
| Molded Package Thickness | A2 | 0.80 | 1.00 | 1.05 |
| Standoff | A1 | 0.05 | – | 0.15 |
| Overall Width | E | | 6.40 BSC | |
| Molded Package Width | E1 | 4.30 | 4.40 | 4.50 |
| Molded Package Length | D | 4.90 | 5.00 | 5.10 |
| Foot Length | L | 0.45 | 0.60 | 0.75 |
| Footprint | L1 | | 1.00 REF | |
| Foot Angle | ϕ | 0° | – | 8° |
| Lead Thickness | c | 0.09 | – | 0.20 |
| Lead Width | b | 0.19 | – | 0.30 |

**Notes:**

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15 mm per side.
3. Dimensioning and tolerancing per ASME Y14.5M.

   BSC: Basic Dimension. Theoretically exact value shown without tolerances.

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-087B

Context Saving ..... 121
Interrupt-on-Change ..... 34
PORTA Interrupt-on-Change ..... 119
RA2/INT ..... 118
Timer0 ..... 119
TMR1 ..... 51
INTOSC Specifications ..... 158, 159
IOCA Register ..... 35

**L**

Load Conditions ..... 156

**M**

MCLR ..... 112
Internal ..... 112
Memory Organization ..... 13
Data ..... 14
Program ..... 13
Microchip Internet Web Site ..... 211
Migrating from other PIC Devices ..... 206
MPLAB ASM30 Assembler, Linker, Librarian ..... 140
MPLAB Integrated Development Environment Software .. 139
MPLAB PM3 Device Programmer ..... 142
MPLAB REAL ICE In-Circuit Emulator System ..... 141
MPLINK Object Linker/MPLIB Object Librarian ..... 140

**O**

OPCODE Field Descriptions ..... 129
Operational Amplifier (OPA) Module
AC Specifications ..... 165
OPTION Register ..... 19, 47
Oscillator
Associated registers ..... 31, 54
Oscillator Module ..... 27
EC ..... 27
HS ..... 27
INTOSC ..... 27
INTOSCIO ..... 27
LP ..... 27
RC ..... 27
RCIO ..... 27
XT ..... 27
Oscillator Parameters ..... 158
Oscillator Specifications ..... 157
Oscillator Start-up Timer (OST)
Specifications ..... 161
OSCTUNE Register ..... 31

**P**

P1A/P1B/P1C/P1D.*See* Enhanced Capture/Compare/PWM (ECCP) ..... 93
Packaging ..... 197
Marking ..... 197
PDIP Details ..... 198
PCL and PCLATH ..... 24
Stack ..... 24
PCON Register ..... 23, 114
PIE1 Register ..... 21
Pin Diagram
PDIP, SOIC, TSSOP ..... 4, 5
QFN ..... 6, 7
Pinout Descriptions
PIC16F610/16HV610 ..... 11
PIC16F616/16HV616 ..... 12
PIR1 Register ..... 22
PORTA ..... 33
Additional Pin Functions ..... 34
ANSEL Register ..... 34
Interrupt-on-Change ..... 34
Weak Pull-Ups ..... 34
Associated registers ..... 41
Pin Descriptions and Diagrams ..... 36
RA0 ..... 36
RA1 ..... 36
RA2 ..... 37
RA3 ..... 38
RA4 ..... 39
RA5 ..... 40
Specifications ..... 159
PORTA Register ..... 33
PORTC ..... 42
Associated registers ..... 44
P1A/P1B/P1C/P1D.*See* Enhanced Capture/Compare/PWM (ECCP) ..... 42
Specifications ..... 159
PORTC Register ..... 42
Power-Down Mode (Sleep) ..... 124
Power-on Reset (POR) ..... 112
Power-up Timer (PWRT) ..... 112
Specifications ..... 161
Precision Internal Oscillator Parameters ..... 159
Prescaler
Shared WDT/Timer0 ..... 46
Switching Prescaler Assignment ..... 46
Program Memory ..... 13
Map and Stack (PIC16F610/16HV610) ..... 13
Map and Stack (PIC16F616/16HV616) ..... 13
Programming, Device Instructions ..... 129
PWM Mode. *See* Enhanced Capture/Compare/PWM ..... 93
PWM1CON Register ..... 105

**R**

Reader Response ..... 212
Read-Modify-Write Operations ..... 129
Registers
ADCON0 (ADC Control 0) ..... 78
ADCON1 (ADC Control 1) ..... 79
ADRESH (ADC Result High) with ADFM = 0) ..... 80
ADRESH (ADC Result High) with ADFM = 1) ..... 80
ADRESL (ADC Result Low) with ADFM = 0) ..... 80
ADRESL (ADC Result Low) with ADFM = 1) ..... 80
ANSEL (Analog Select) ..... 34
CCP1CON (Enhanced CCP1 Control) ..... 85
CM1CON0 (C1 Control) ..... 62
CM2CON0 (C2 Control) ..... 63
CM2CON1 (C2 Control) ..... 65
CONFIG (Configuration Word) ..... 110
Data Memory Map (PIC16F610/16HV610) ..... 15
Data Memory Map (PIC16F616/16HV616) ..... 15
ECCPAS (Enhanced CCP Auto-shutdown Control) . 102
INTCON (Interrupt Control) ..... 20
IOCA (Interrupt-on-Change PORTA) ..... 35
OPTION_REG (OPTION) ..... 19, 47
OSCTUNE (Oscillator Tuning) ..... 31
PCON (Power Control Register) ..... 23
PCON (Power Control) ..... 114
PIE1 (Peripheral Interrupt Enable 1) ..... 21
PIR1 (Peripheral Interrupt Register 1) ..... 22
PORTA ..... 33
PORTC ..... 42
PWM1CON (Enhanced PWM Control) ..... 105
Reset Values ..... 116