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 | I²C, SPI |
| Peripherals | Brown-out Detect/Reset, POR, PWM, WDT |
| Number of I/O | 16 |
| Program Memory Size | 1.75KB (1K x 14) |
| Program Memory Type | FLASH |
| EEPROM Size | 128 x 8 |
| RAM Size | 128 x 8 |
| Voltage - Supply (Vcc/Vdd) | 4V ~ 5.5V |
| Data Converters | A/D 5x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 20-SSOP (0.209", 5.30mm Width) |
| Supplier Device Package | 20-SSOP |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16f818-i-sstsl |

# 1.0 DEVICE OVERVIEW

This document contains device specific information for the operation of the PIC16F818/819 devices. Additional information may be found in the *"PIC® Mid-Range MCU Family Reference Manual"* (DS33023) which may be downloaded from the Microchip web site. The Reference Manual should be considered a complementary document to this data sheet and is highly recommended reading for a better understanding of the device architecture and operation of the peripheral modules.

The PIC16F818/819 belongs to the Mid-Range family of the PIC® devices. The devices differ from each other in the amount of Flash program memory, data memory and data EEPROM (see Table 1-1). A block diagram of the devices is shown in Figure 1-1. These devices contain features that are new to the PIC16 product line:

- Internal RC oscillator with eight selectable frequencies, including 31.25 kHz, 125 kHz, 250 kHz, 500 kHz, 1 MHz, 2 MHz, 4 MHz and 8 MHz. The INTRC can be configured as the system clock via the configuration bits. Refer to **Section 4.5 "Internal Oscillator Block"** and **Section 12.1 "Configuration Bits"** for further details.
- The Timer1 module current consumption has been greatly reduced from 20 μA (previous PIC16 devices) to 1.8 μA typical (32 kHz at 2V), which is ideal for real-time clock applications. Refer to **Section 6.0 "Timer0 Module"** for further details.
- The amount of oscillator selections has increased. The RC and INTRC modes can be selected with an I/O pin configured as an I/O or a clock output ($F_{OSC}/4$). An external clock can be configured with an I/O pin. Refer to **Section 4.0 "Oscillator Configurations"** for further details.

**TABLE 1-1: AVAILABLE MEMORY IN PIC16F818/819 DEVICES**

| Device | Program Flash | Data Memory | Data EEPROM |
|---|---|---|---|
| PIC16F818 | 1K x 14 | 128 x 8 | 128 x 8 |
| PIC16F819 | 2K x14 | 256 x 8 | 256 x 8 |

There are 16 I/O pins that are user configurable on a pin-to-pin basis. Some pins are multiplexed with other device functions. These functions include:

- External Interrupt
- Change on PORTB Interrupt
- Timer0 Clock Input
- Low-Power Timer1 Clock/Oscillator
- Capture/Compare/PWM
- 10-bit, 5-channel Analog-to-Digital Converter
- SPI/I²C
- $\overline{\text{MCLR}}$ (RA5) can be configured as an Input

Table 1-2 details the pinout of the devices with descriptions and details for each pin.

## 3.3 Reading Data EEPROM Memory

To read a data memory location, the user must write the address to the EEADR register, clear the EEPGD control bit (EECON1<7>) and then set control bit, RD (EECON1<0>). The data is available in the very next cycle in the EEDATA register; therefore, it can be read in the next instruction (see Example 3-1). EEDATA will hold this value until another read or until it is written to by the user (during a write operation).

The steps to reading the EEPROM data memory are:

1. Write the address to EEADR. Make sure that the address is not larger than the memory size of the device.
2. Clear the EEPGD bit to point to EEPROM data memory.
3. Set the RD bit to start the read operation.
4. Read the data from the EEDATA register.

**EXAMPLE 3-1: DATA EEPROM READ**

```
BANKSEL EEADR          ; Select Bank of EEADR
MOVF    ADDR, W        ;
MOVWF   EEADR          ; Data Memory Address
                       ; to read
BANKSEL EECON1         ; Select Bank of EECON1
BCF     EECON1, EEPGD  ; Point to Data memory
BSF     EECON1, RD     ; EE Read
BANKSEL EEDATA         ; Select Bank of EEDATA
MOVF    EEDATA, W      ; W = EEDATA
```

## 3.4 Writing to Data EEPROM Memory

To write an EEPROM data location, the user must first write the address to the EEADR register and the data to the EEDATA register. Then, the user must follow a specific write sequence to initiate the write for each byte.

The write will not initiate if the write sequence is not exactly followed (write 55h to EECON2, write AAh to EECON2, then set WR bit) for each byte. We strongly recommend that interrupts be disabled during this code segment (see Example 3-2).

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

After a write sequence has been initiated, clearing the WREN bit will not affect this write cycle. The WR bit will be inhibited from being set unless the WREN bit is set. At the completion of the write cycle, the WR bit is cleared in hardware and the EE Write Complete Interrupt Flag bit (EEIF) is set. The user can either enable this interrupt or poll this bit. EEIF must be cleared by software.

The steps to write to EEPROM data memory are:

1. If step 10 is not implemented, check the WR bit to see if a write is in progress.
2. Write the address to EEADR. Make sure that the address is not larger than the memory size of the device.
3. Write the 8-bit data value to be programmed in the EEDATA register.
4. Clear the EEPGD bit to point to EEPROM data memory.
5. Set the WREN bit to enable program operations.
6. Disable interrupts (if enabled).
7. Execute the special five instruction sequence:
   - Write 55h to EECON2 in two steps (first to W, then to EECON2)
   - Write AAh to EECON2 in two steps (first to W, then to EECON2)
   - Set the WR bit
8. Enable interrupts (if using interrupts).
9. Clear the WREN bit to disable program operations.
10. At the completion of the write cycle, the WR bit is cleared and the EEIF interrupt flag bit is set (EEIF must be cleared by firmware). If step 1 is not implemented, then firmware should check for EEIF to be set, or WR to be clear, to indicate the end of the program cycle.

**EXAMPLE 3-2: DATA EEPROM WRITE**

```
BANKSEL EECON1         ; Select Bank of
                       ; EECON1
BTFSC   EECON1, WR     ; Wait for write
GOTO    $-1            ; to complete
BANKSEL EEADR          ; Select Bank of
                       ; EEADR
MOVF    ADDR, W        ;
MOVWF   EEADR          ; Data Memory
                       ; Address to write
MOVF    VALUE, W       ;
MOVWF   EEDATA         ; Data Memory Value
                       ; to write
BANKSEL EECON1         ; Select Bank of
                       ; EECON1
BCF     EECON1, EEPGD  ; Point to DATA
                       ; memory
BSF     EECON1, WREN   ; Enable writes

BCF     INTCON, GIE    ; Disable INTs.
Required Sequence:
MOVLW   55h            ;
MOVWF   EECON2         ; Write 55h
MOVLW   AAh            ;
MOVWF   EECON2         ; Write AAh
BSF     EECON1, WR     ; Set WR bit to
                       ; begin write
BSF     INTCON, GIE    ; Enable INTs.
BCF     EECON1, WREN   ; Disable writes
```

An example of the complete four-word write sequence is shown in Example 3-5. The initial address is loaded into the EEADRH:EEADR register pair; the four words of data are loaded using indirect addressing, assuming that a row erase sequence has already been performed.

**EXAMPLE 3-5: WRITING TO FLASH PROGRAM MEMORY**

Required Sequence

```
; This write routine assumes the following:

; 1. The 32 words in the erase block have already been erased.
; 2. A valid starting address (the least significant bits = '00') is loaded into EEADRH:EEADR
; 3. This example is starting at 0x100, this is an application dependent setting.
; 4. The 8 bytes (4 words) of data are loaded, starting at an address in RAM called ARRAY.
; 5. This is an example only, location of data to program is application dependent.
; 6. word_block is located in data memory.

        BANKSEL EECON1              ;prepare for WRITE procedure
        BSF     EECON1, EEPGD       ;point to program memory
        BSF     EECON1, WREN        ;allow write cycles
        BCF     EECON1, FREE        ;perform write only

        BANKSEL word_block
        MOVLW   .4
        MOVWF   word_block          ;prepare for 4 words to be written

        BANKSEL EEADRH              ;Start writing at 0x100
        MOVLW   0x01
        MOVWF   EEADRH              ;load HIGH address
        MOVLW   0x00
        MOVWF   EEADR               ;load LOW address
        BANKSEL ARRAY
        MOVLW   ARRAY               ;initialize FSR to start of data
        MOVWF   FSR
LOOP
        BANKSEL EEDATA
        MOVF    INDF, W             ;indirectly load EEDATA
        MOVWF   EEDATA
        INCF    FSR, F              ;increment data pointer
        MOVF    INDF, W             ;indirectly load EEDATH
        MOVWF   EEDATH
        INCF    FSR, F              ;increment data pointer

        BANKSEL EECON1
        MOVLW   0x55                ;required sequence
        MOVWF   EECON2
        MOVLW   0xAA
        MOVWF   EECON2
        BSF     EECON1, WR          ;set WR bit to begin write
        NOP                         ;instructions here are ignored as processor
        NOP

        BANKSEL EEADR
        INCF    EEADR, f            ;load next word address
        BANKSEL word_block
        DECFSZ  word_block, f       ;have 4 words been written?
        GOTO    loop                ;NO, continue with writing

        BANKSEL EECON1
        BCF     EECON1, WREN        ;YES, 4 words complete, disable writes
        BSF     INTCON, GIE         ;enable interrupts
```

# 5.0 I/O PORTS

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

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

## 5.1 PORTA and the TRISA Register

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

> **Note:** On a Power-on Reset, the pins PORTA<4:0> are configured as analog inputs and read as '0'.

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

Pin RA4 is multiplexed with the Timer0 module clock input and with an analog input to become the RA4/AN4/T0CKI pin. The RA4/AN4/T0CKI pin is a Schmitt Trigger input and full CMOS output driver.

Pin RA5 is multiplexed with the Master Clear module input. The RA5/$\overline{MCLR}$/VPP pin is a Schmitt Trigger input.

Pin RA6 is multiplexed with the oscillator module input and external oscillator output. Pin RA7 is multiplexed with the oscillator module input and external oscillator input. Pin RA6/OSC2/CLKO and pin RA7/OSC1/CLKI are Schmitt Trigger inputs and full CMOS output drivers.

Pins RA<1:0> are multiplexed with analog inputs. Pins RA<3:2> are multiplexed with analog inputs and VREF inputs. Pins RA<3:0> have TTL inputs and full CMOS output drivers.

**EXAMPLE 5-1: INITIALIZING PORTA**

```
BANKSEL PORTA    ; select bank of PORTA
CLRF    PORTA    ; Initialize PORTA by
                 ; clearing output
                 ; data latches
BANKSEL ADCON1   ; Select Bank of ADCON1
MOVLW   0x06     ; Configure all pins
MOVWF   ADCON1   ; as digital inputs
MOVLW   0xFF     ; Value used to
                 ; initialize data
                 ; direction
MOVWF   TRISA    ; Set RA<7:0> as inputs
```

**TABLE 5-1: PORTA FUNCTIONS**

| Name | Bit# | Buffer | Function |
|---|---|---|---|
| RA0/AN0 | bit 0 | TTL | Input/output or analog input. |
| RA1/AN1 | bit 1 | TTL | Input/output or analog input. |
| RA2/AN2/VREF- | bit 2 | TTL | Input/output, analog input or VREF-. |
| RA3/AN3/VREF+ | bit 3 | TTL | Input/output, analog input or VREF+. |
| RA4/AN4/T0CKI | bit 4 | ST | Input/output, analog input or external clock input for Timer0. |
| RA5/$\overline{MCLR}$/VPP | bit 5 | ST | Input, Master Clear (Reset) or programming voltage input. |
| RA6/OSC2/CLKO | bit 6 | ST | Input/output, connects to crystal or resonator, oscillator output or 1/4 the frequency of OSC1 and denotes the instruction cycle in RC mode. |
| RA7/OSC1/CLKI | bit 7 | ST/CMOS(1) | Input/output, connects to crystal or resonator or oscillator input. |

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

**Note 1:** This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.

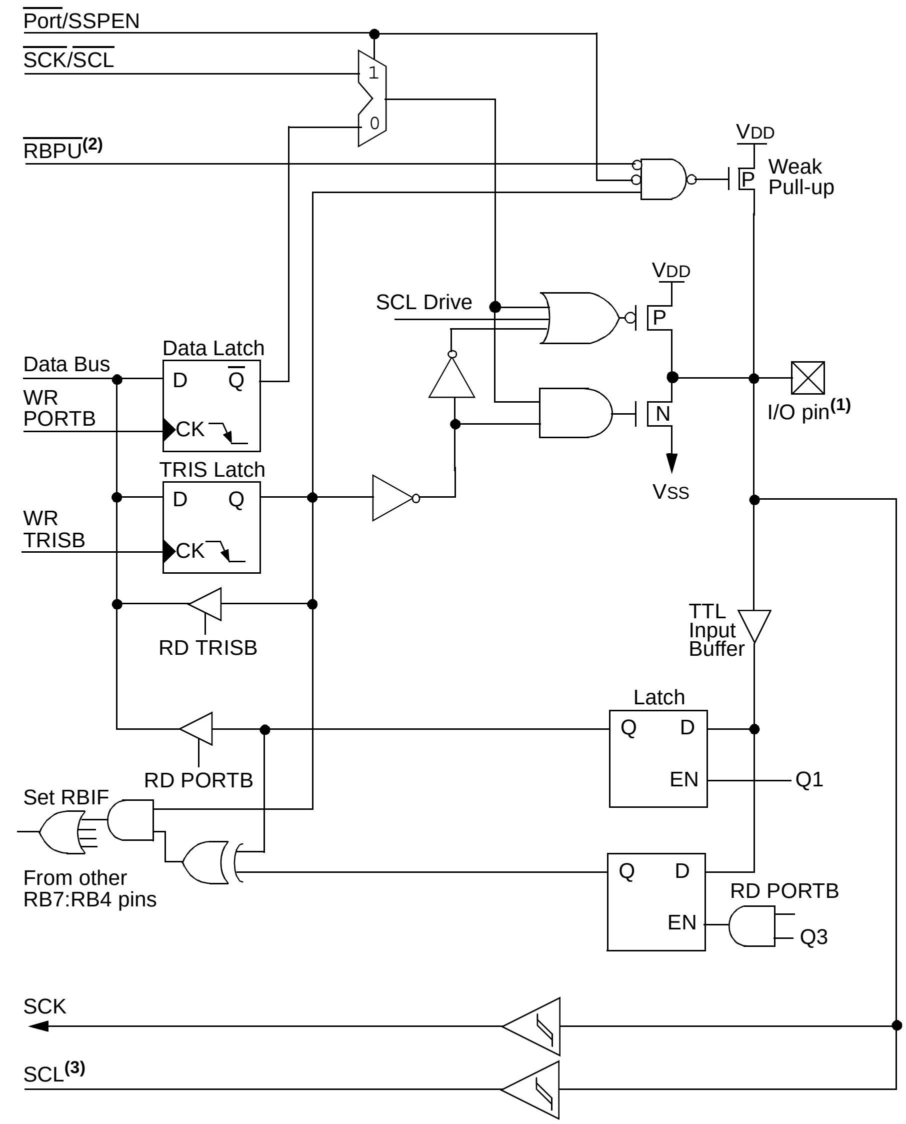
**TABLE 5-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTA**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR, BOR | Value on all other Resets |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 05h | PORTA | RA7 | RA6 | RA5 | RA4 | RA3 | RA2 | RA1 | RA0 | xxx0 0000 | uuu0 0000 |
| 85h | TRISA | TRISA7 | TRISA6 | TRISA5(1) | PORTA Data Direction Register | | | | | 1111 1111 | 1111 1111 |
| 9Fh | ADCON1 | ADFM | ADCS2 | — | — | PCFG3 | PCFG2 | PCFG1 | PCFG0 | 00-- 0000 | 00-- 0000 |

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

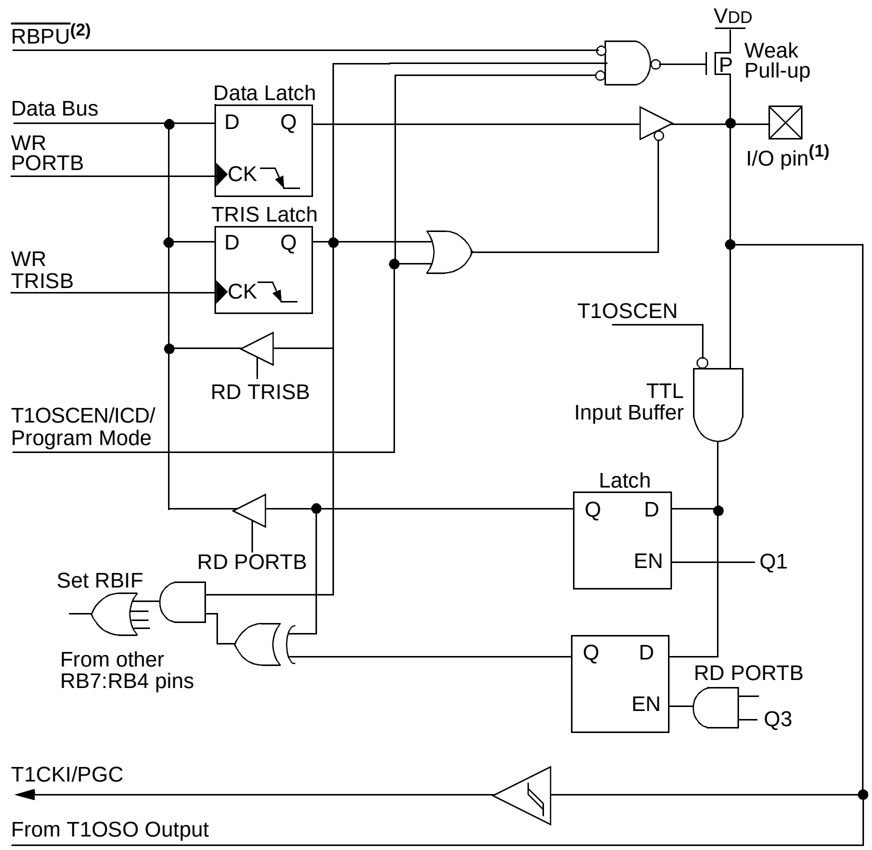
**Note 1:** Pin 5 is an input only; the state of the TRISA5 bit has no effect and will always read '1'.

**FIGURE 5-12: BLOCK DIAGRAM OF RB4 PIN**




**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 RBPU bit.
**3:** The SCL Schmitt Trigger conforms to the $I^2C$™ specification.

**FIGURE 5-14: BLOCK DIAGRAM OF RB6 PIN**




**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 RBPU bit.

## 9.3 PWM Mode

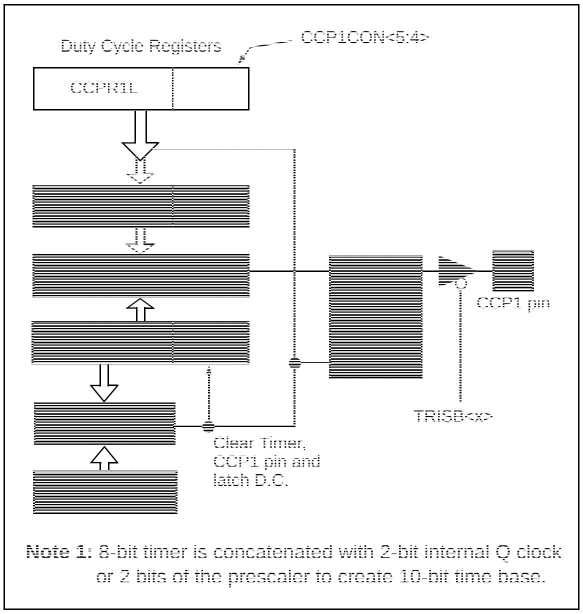
In Pulse-Width Modulation (PWM) mode, the CCP1 pin produces up to a 10-bit resolution PWM output. Since the CCP1 pin is multiplexed with the PORTB data latch, the TRISB<x> bit must be cleared to make the CCP1 pin an output.

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

Figure 9-3 shows a simplified block diagram of the CCP module in PWM mode.

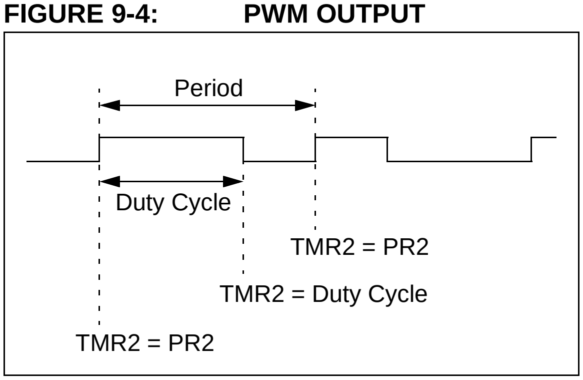
For a step by step procedure on how to set up the CCP module for PWM operation, see **Section 9.3.3 "Setup for PWM Operation"**.

**FIGURE 9-3: SIMPLIFIED PWM BLOCK DIAGRAM**




A PWM output (Figure 9-4) has a time base (period) and a time that the output stays high (duty cycle). The frequency of the PWM is the inverse of the period (1/period).

**FIGURE 9-4: PWM OUTPUT**




### 9.3.1 PWM PERIOD

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

**EQUATION 9-1:**

$$\text{PWM Period} = [(\text{PR2}) + 1] \bullet 4 \bullet T_{OSC} \bullet (\text{TMR2 Prescale Value})$$

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 8.0 "Timer2 Module"**) is not used in the determination of the PWM frequency. The postscaler could be used to have a servo update rate at a different frequency than the PWM output. |
|---|---|

### 9.3.2 PWM DUTY CYCLE

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

**EQUATION 9-2:**

$$\text{PWM Duty Cycle} = (\text{CCPR1L:CCP1CON<5:4>}) \bullet T_{OSC} \bullet (\text{TMR2 Prescale Value})$$

CCPR1L and CCP1CON<5:4> can be written to at any time but the duty cycle value is not latched into CCPR1H until 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 duty cycle. This double-buffering is essential for glitchless PWM operation.

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

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

The Analog-to-Digital (A/D) converter module has five inputs for 18/20 pin devices.

The conversion of an analog input signal results in a corresponding 10-bit digital number. The A/D module has a high and low-voltage reference input that is software selectable to some combination of $V_{DD}$, $V_{SS}$, RA2 or RA3.

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

The A/D module has four registers:

- A/D Result High Register (ADRESH)
- A/D Result Low Register (ADRESL)
- A/D Control Register 0 (ADCON0)
- A/D Control Register 1 (ADCON1)

The ADCON0 register, shown in Register 11-1, controls the operation of the A/D module. The ADCON1 register, shown in Register 11-2, 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/Os.

Additional information on using the A/D module can be found in the *"PIC® Mid-Range MCU Family Reference Manual"* (DS33023).

**REGISTER 11-1: ADCON0: A/D CONTROL REGISTER 0 (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 |
| bit 7 | | | | | | | bit 0 |

bit 7-6 **ADCS1:ADCS0:** A/D Conversion Clock Select bits

If ADCS2 = 0:
00 = FOSC/2
01 = FOSC/8
10 = FOSC/32
11 = FRC (clock derived from the internal A/D module RC oscillator)

If ADCS2 = 1:
00 = FOSC/4
01 = FOSC/16
10 = FOSC/64
11 = FRC (clock derived from the internal A/D module 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 (RA4/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 shut-off and consumes no operating current

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

## 11.5 A/D Operation During Sleep

The A/D module can operate during Sleep mode. This requires that the A/D clock source be set to RC (ADCS1:ADCS0 = 11). When the RC clock source is selected, the A/D module waits one instruction cycle before starting the conversion. This allows the SLEEP instruction to be executed which eliminates all digital switching noise from the conversion. When the conversion is completed, the GO/DONE bit will be cleared and the result loaded into the ADRES register. If the A/D interrupt is enabled, the device will wake-up from Sleep. If the A/D interrupt is not enabled, the A/D module will then be turned off, although the ADON bit will remain set.

When the A/D clock source is another clock option (not RC), a SLEEP instruction will cause the present conversion to be aborted and the A/D module to be turned off, though the ADON bit will remain set.

Turning off the A/D places the A/D module in its lowest current consumption state.

| **Note:** | For the A/D module to operate in Sleep, the A/D clock source must be set to RC (ADCS1:ADCS0 = 11). To perform an A/D conversion in Sleep, ensure the SLEEP instruction immediately follows the instruction that sets the GO/DONE bit. |
|---|---|

## 11.6 Effects of a Reset

A device Reset forces all registers to their Reset state. The A/D module is disabled and any conversion in progress is aborted. All A/D input pins are configured as analog inputs.

The value that is in the ADRESH:ADRESL registers is not modified for a Power-on Reset. The ADRESH:ADRESL registers will contain unknown data after a Power-on Reset.

## 11.7 Use of the CCP Trigger

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

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

**TABLE 11-2: REGISTERS/BITS ASSOCIATED WITH A/D**

| 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 10Bh,18Bh | INTCON | GIE | PEIE | TMR0IE | INTE | RBIE | TMR0IF | 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 | ADRESH | A/D Result Register High Byte | | | | | | | | xxxx xxxx | uuuu uuuu |
| 9Eh | ADRESL | A/D Result Register Low Byte | | | | | | | | xxxx xxxx | uuuu uuuu |
| 1Fh | ADCON0 | ADCS1 | ADCS0 | CHS2 | CHS1 | CHS0 | GO/DONE | — | ADON | 0000 00-0 | 0000 00-0 |
| 9Fh | ADCON1 | ADFM | ADCS2 | — | — | PCFG3 | PCFG2 | PCFG1 | PCFG0 | 00-- 0000 | 00-- 0000 |
| 05h | PORTA | RA7 | RA6 | RA5 | RA4 | RA3 | RA2 | RA1 | RA0 | xxx0 0000 | uuu0 0000 |
| 85h | TRISA | TRISA7 | TRISA6 | TRISA5 | PORTA Data Direction Register | | | | | 1111 1111 | 1111 1111 |

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

## 12.2 Reset

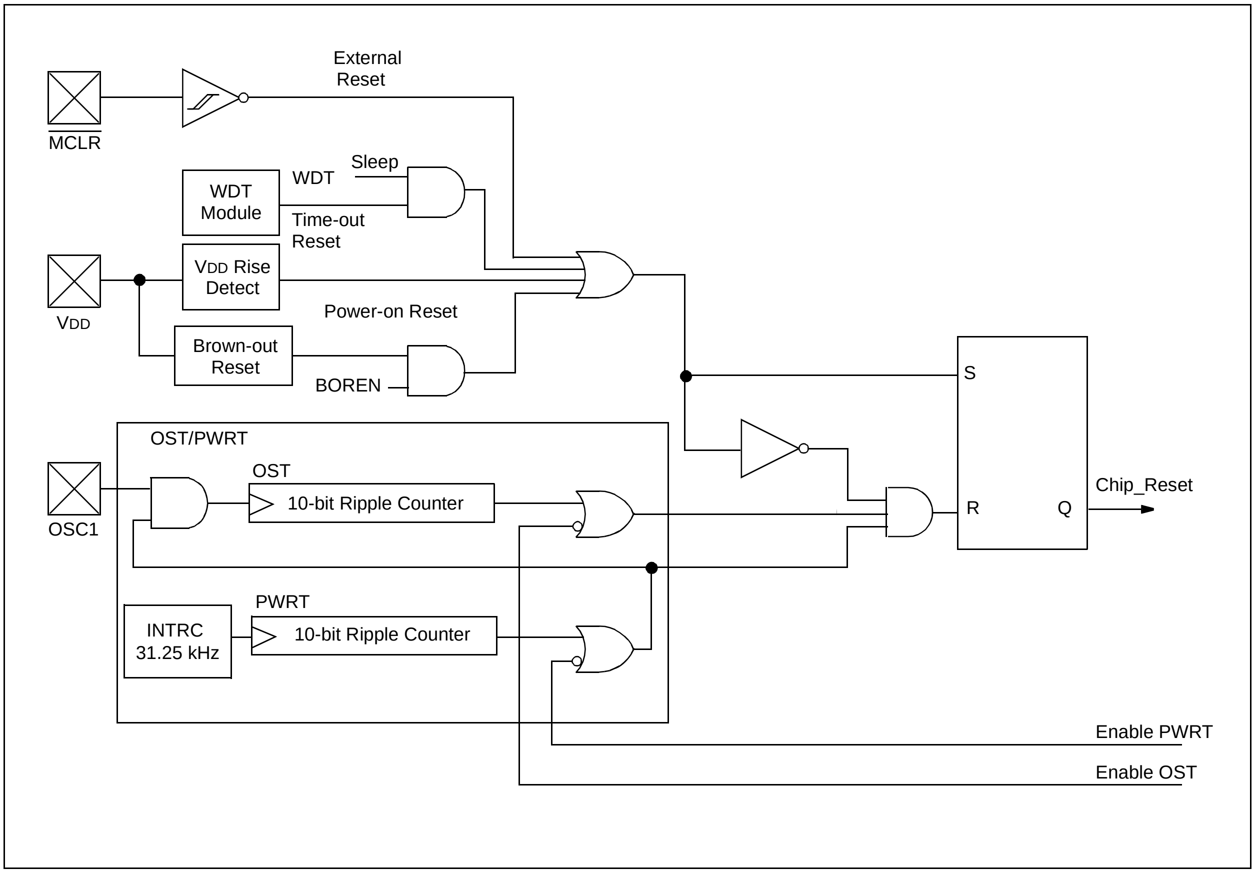
The PIC16F818/819 differentiates between various kinds of Reset:

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

Some registers are not affected in any Reset condition. Their status is unknown on POR and unchanged in any other Reset. Most other registers are reset to a “Reset state” on Power-on Reset (POR), on the $\overline{\text{MCLR}}$ and WDT Reset, on $\overline{\text{MCLR}}$ Reset during Sleep and Brown-out Reset (BOR). They are not affected by a WDT wake-up which is viewed as the resumption of normal operation. The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits are set or cleared differently in different Reset situations as indicated in Table 12-3. These bits are used in software to determine the nature of the Reset. Upon a POR, BOR or wake-up from Sleep, the CPU requires approximately 5-10 μs to become ready for code execution. This delay runs in parallel with any other timers. See Table 12-4 for a full description of Reset states of all registers.

A simplified block diagram of the on-chip Reset circuit is shown in Figure 12-1.

**FIGURE 12-1: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT**



# 15.0 ELECTRICAL CHARACTERISTICS

## Absolute Maximum Ratings †

| | |
|---|---|
| Ambient temperature under bias | -40°C to +125°C |
| Storage temperature | -65°C to +150°C |
| Voltage on any pin with respect to $V_{SS}$ (except $V_{DD}$ and $\overline{MCLR}$) | -0.3V to ($V_{DD}$ + 0.3V) |
| Voltage on $V_{DD}$ with respect to $V_{SS}$ | -0.3 to +7.5V |
| Voltage on $\overline{MCLR}$ with respect to $V_{SS}$ **(Note 2)** | -0.3 to +14V |
| Total power dissipation **(Note 1)** | 1W |
| Maximum current out of $V_{SS}$ pin | 200 mA |
| Maximum current into $V_{DD}$ pin | 200 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 | 100 mA |
| Maximum current sourced by PORTA | 100 mA |
| Maximum current sunk by PORTB | 100 mA |
| Maximum current sourced by PORTB | 100 mA |

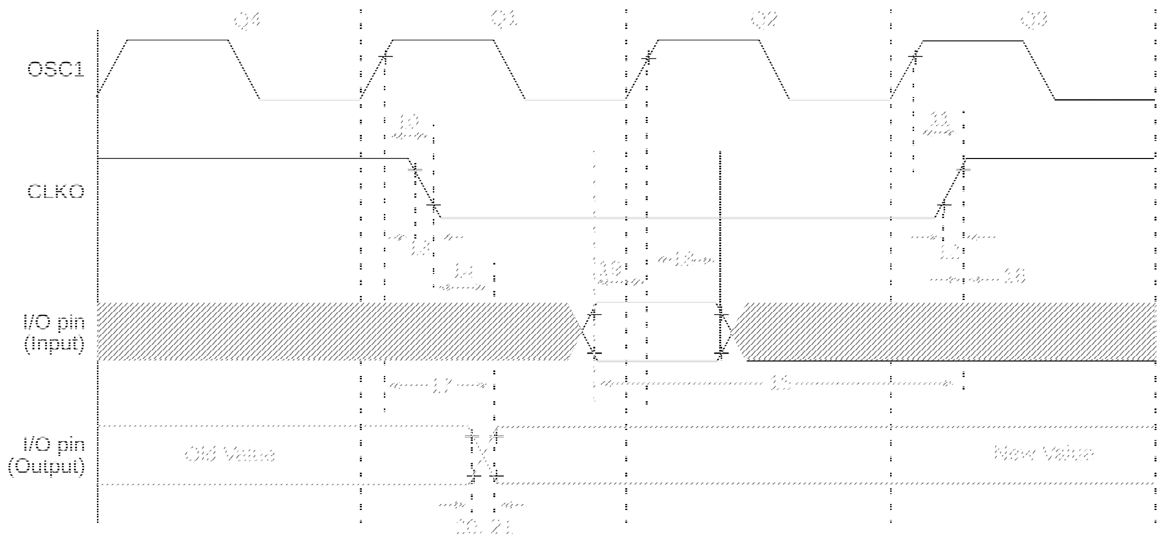
**Note 1:** Power dissipation is calculated as follows: $Pdis = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum(V_{OL} \times I_{OL})$

**2:** Voltage spikes at the $\overline{MCLR}$ pin may cause latch-up. A series resistor of greater than 1 kΩ should be used to pull $\overline{MCLR}$ to $V_{DD}$, rather than tying the pin directly to $V_{DD}$.

† 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 15-5: CLKO AND I/O TIMING**




**Note:** Refer to Figure 15-3 for load conditions.

**TABLE 15-2: CLKO AND I/O TIMING REQUIREMENTS**

| Param No. | Symbol | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 10* | TOSH2CKL | OSC1 ↑ to CLKO ↓ | | — | 75 | 200 | ns | **(Note 1)** |
| 11* | TOSH2CKH | OSC1 ↑ to CLKO ↑ | | — | 75 | 200 | ns | **(Note 1)** |
| 12* | TCKR | CLKO Rise Time | | — | 35 | 100 | ns | **(Note 1)** |
| 13* | TCKF | CLKO Fall Time | | — | 35 | 100 | ns | **(Note 1)** |
| 14* | TCKL2IOV | CLKO ↓ to Port Out Valid | | — | — | 0.5 TCY + 20 | ns | **(Note 1)** |
| 15* | TIOV2CKH | Port In Valid before CLKO ↑ | | TOSC + 200 | — | — | ns | **(Note 1)** |
| 16* | TCKH2IOI | Port In Hold after CLKO ↑ | | 0 | — | — | ns | **(Note 1)** |
| 17* | TOSH2IOV | OSC1 ↑ (Q1 cycle) to Port Out Valid | | — | 100 | 255 | ns | |
| 18* | TOSH2IOI | OSC1 ↑ (Q2 cycle) to Port Input Invalid (I/O in hold time) | PIC16**F**818/819 | 100 | — | — | ns | |
| | | | PIC16**LF**818/819 | 200 | — | — | ns | |
| 19* | TIOV2OSH | Port Input Valid to OSC1 ↑ (I/O in setup time) | | 0 | — | — | ns | |
| 20* | TIOR | Port Output Rise Time | PIC16**F**818/819 | — | 10 | 40 | ns | |
| | | | PIC16**LF**818/819 | — | — | 145 | ns | |
| 21* | TIOF | Port Output Fall Time | PIC16**F**818/819 | — | 10 | 40 | ns | |
| | | | PIC16**LF**818/819 | — | — | 145 | ns | |
| 22††* | TINP | INT pin High or Low Time | | TCY | — | — | ns | |
| 23††* | TRBP | RB7:RB4 Change INT High or Low Time | | TCY | — | — | ns | |

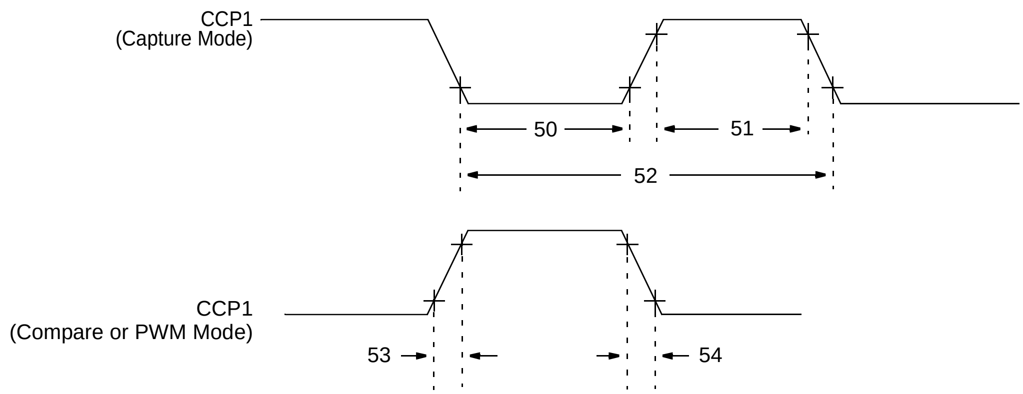
* These parameters are characterized but not tested.

† 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 edges.

**Note 1:** Measurements are taken in RC mode, where CLKO output is 4 x TOSC.

**FIGURE 15-9: CAPTURE/COMPARE/PWM TIMINGS (CCP1)**




**Note:** Refer to Figure 15-3 for load conditions.

**TABLE 15-5: CAPTURE/COMPARE/PWM REQUIREMENTS (CCP1)**

| Param No. | Symbol | Characteristic | | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|---|
| 50* | TCCL | CCP1 Input Low Time | No Prescaler | | 0.5 TCY + 20 | — | — | ns | |
| | | | With Prescaler | PIC16**F**818/819 | 10 | — | — | ns | |
| | | | | PIC16**LF**818/819 | 20 | — | — | ns | |
| 51* | TCCH | CCP1 Input High Time | No Prescaler | | 0.5 TCY + 20 | — | — | ns | |
| | | | With Prescaler | PIC16**F**818/819 | 10 | — | — | ns | |
| | | | | PIC16**LF**818/819 | 20 | — | — | ns | |
| 52* | TCCP | CCP1 Input Period | | | $\frac{3 TCY + 40}{N}$ | — | — | ns | N = prescale value (1,4 or 16) |
| 53* | TCCR | CCP1 Output Rise Time | | PIC16**F**818/819 | — | 10 | 25 | ns | |
| | | | | PIC16**LF**818/819 | — | 25 | 50 | ns | |
| 54* | TCCF | CCP1 Output Fall Time | | PIC16**F**818/819 | — | 10 | 25 | ns | |
| | | | | PIC16**LF**818/819 | — | 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.

**FIGURE 15-10: SPI MASTER MODE TIMING (CKE = 0, SMP = 0)**

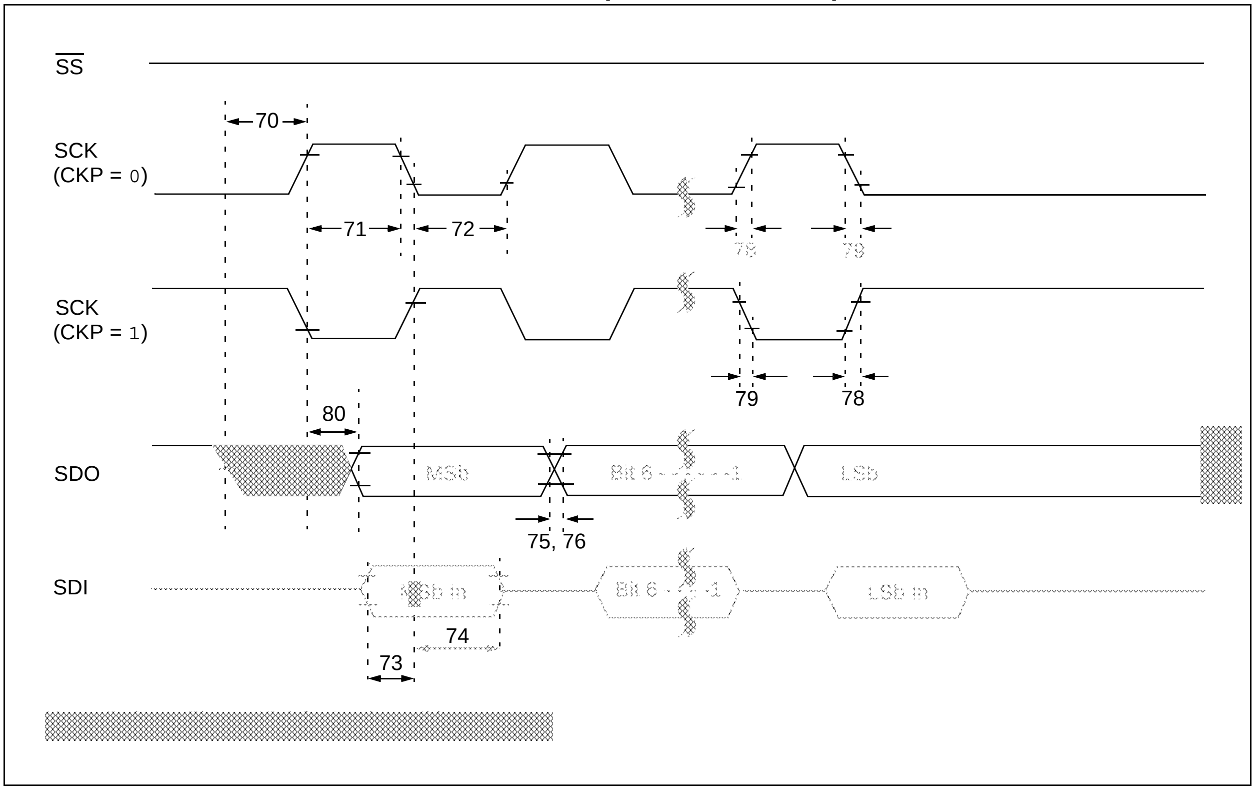
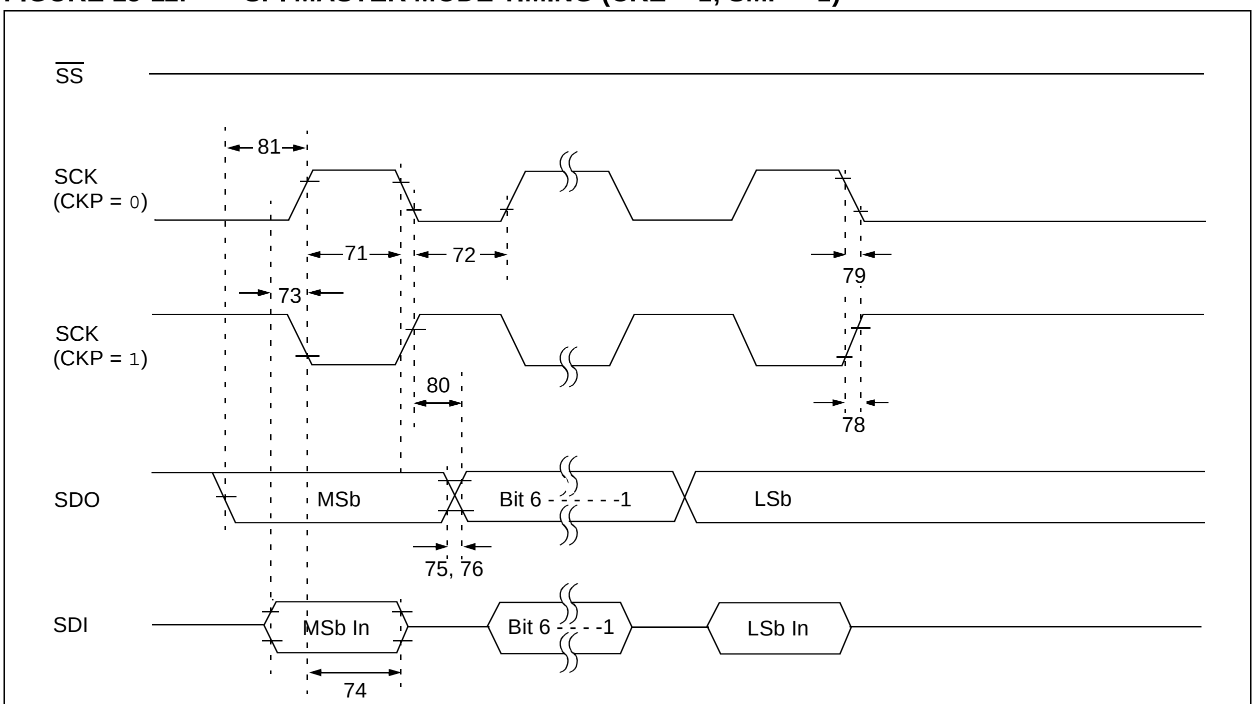



**FIGURE 15-11: SPI MASTER MODE TIMING (CKE = 1, SMP = 1)**




**Note:** Refer to Figure 15-3 for load conditions.

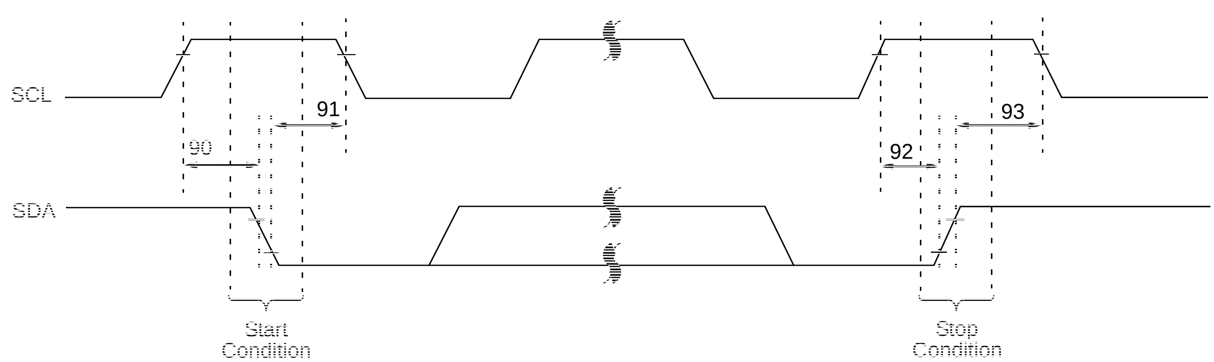
**TABLE 15-6: SPI MODE REQUIREMENTS**

| 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) | | TCY + 20 | — | — | ns | |
| 72* | TSCL | SCK Input Low Time (Slave mode) | | TCY + 20 | — | — | ns | |
| 73* | TDIV2SCH, TDIV2SCL | Setup Time of SDI Data Input to SCK Edge | | 100 | — | — | ns | |
| 74* | TSCH2DIL, TSCL2DIL | Hold Time of SDI Data Input to SCK Edge | | 100 | — | — | ns | |
| 75* | TDOR | SDO Data Output Rise Time | PIC16**F**818/819<br>PIC16**LF**818/819 | —<br>— | 10<br>25 | 25<br>50 | ns<br>ns | |
| 76* | TDOF | SDO Data Output Fall Time | | — | 10 | 25 | ns | |
| 77* | TSSH2DOZ | $\overline{SS}$ ↑ to SDO Output High-Impedance | | 10 | — | 50 | ns | |
| 78* | TSCR | SCK Output Rise Time (Master mode) | PIC16**F**818/819<br>PIC16**LF**818/819 | —<br>— | 10<br>25 | 25<br>50 | ns<br>ns | |
| 79* | TSCF | SCK Output Fall Time (Master mode) | | — | 10 | 25 | ns | |
| 80* | TSCH2DOV, TSCL2DOV | SDO Data Output Valid after SCK Edge | PIC16**F**818/819<br>PIC16**LF**818/819 | —<br>— | —<br>— | 50<br>145 | ns<br>ns | |
| 81* | TDOV2SCH, TDOV2SCL | SDO Data Output Setup to SCK Edge | | TCY | — | — | ns | |
| 82* | TSSL2DOV | SDO Data Output Valid after $\overline{SS}$ ↓ Edge | | — | — | 50 | ns | |
| 83* | TSCH2SSH, TSCL2SSH | $\overline{SS}$ ↑ after SCK Edge | | 1.5 TCY + 40 | — | — | 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.

**FIGURE 15-14: I²C™ BUS START/STOP BITS TIMING**




**Note**: Refer to Figure 15-3 for load conditions.

**FIGURE 16-19: TYPICAL, MINIMUM AND MAXIMUM $V_{OL}$ vs. $I_{OL}$ ($V_{DD}$ = 5V, -40°C TO +125°C)**

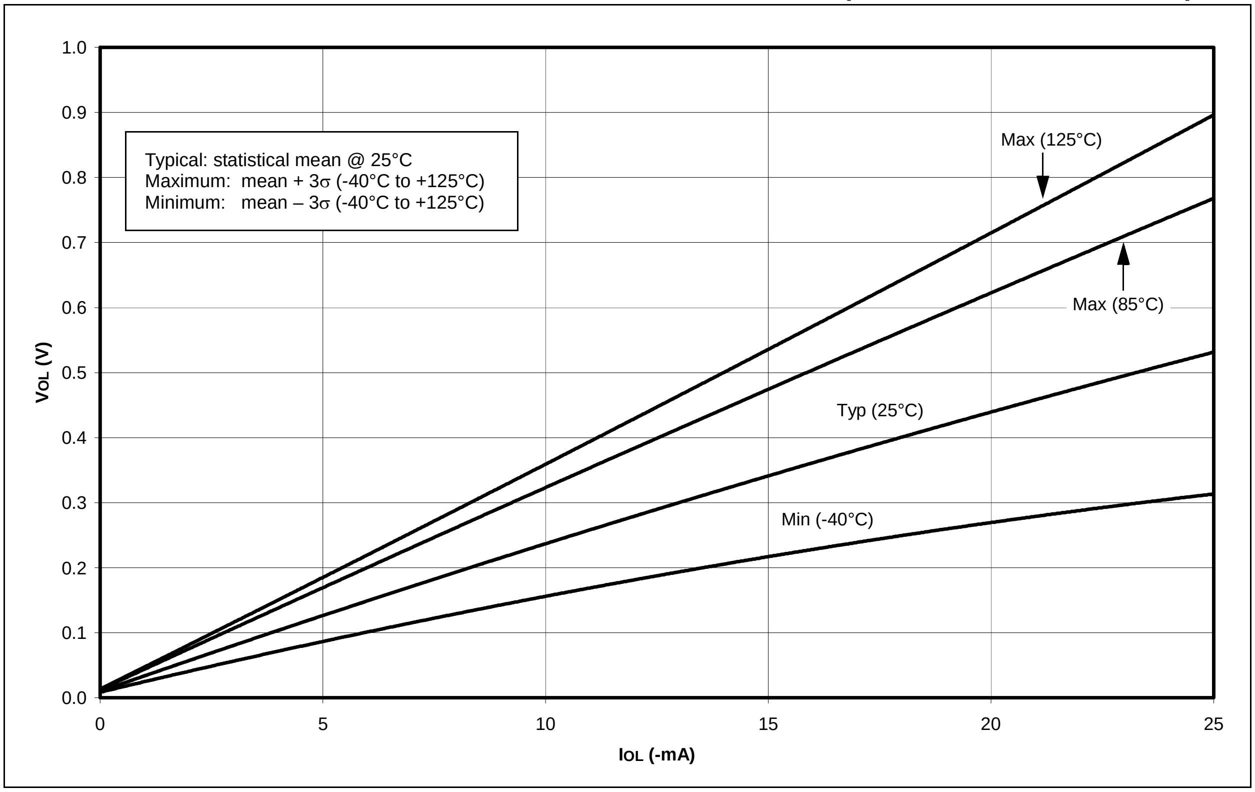



**FIGURE 16-20: TYPICAL, MINIMUM AND MAXIMUM $V_{OL}$ vs. $I_{OL}$ ($V_{DD}$ = 3V, -40°C TO +125°C)**

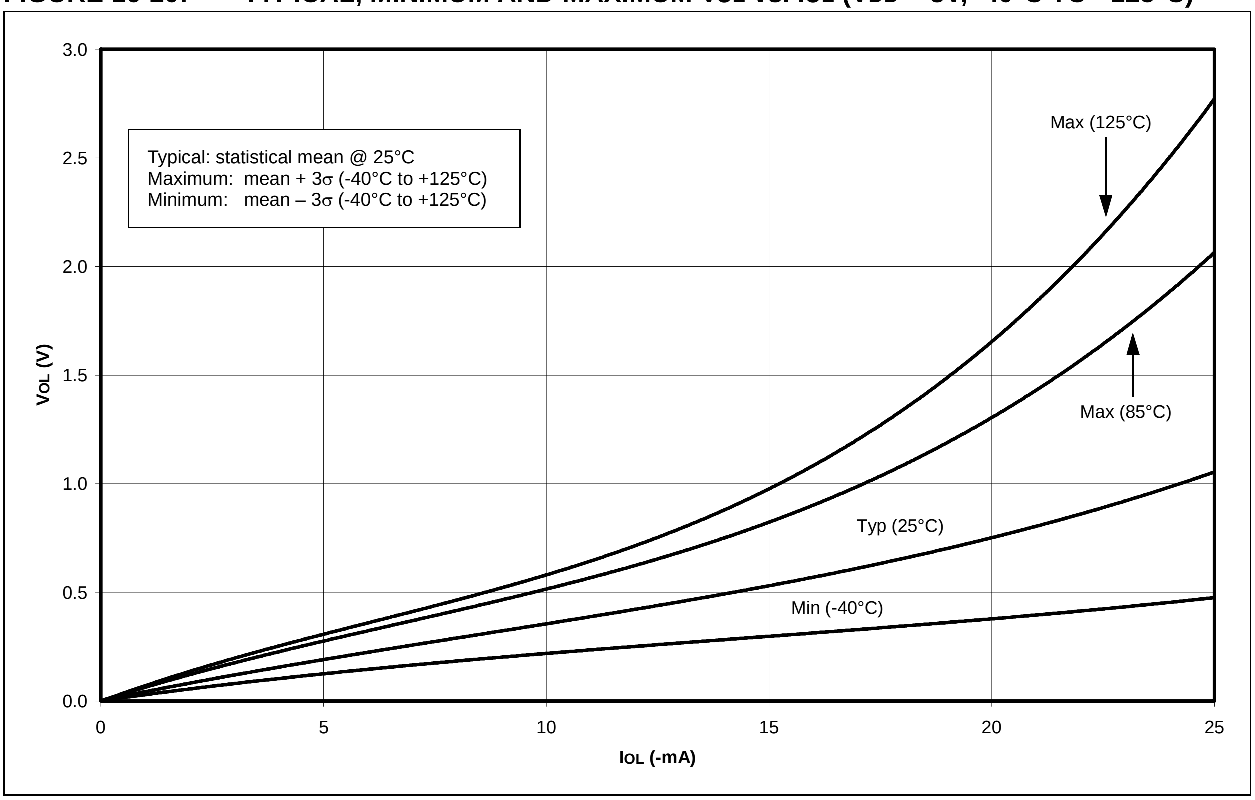
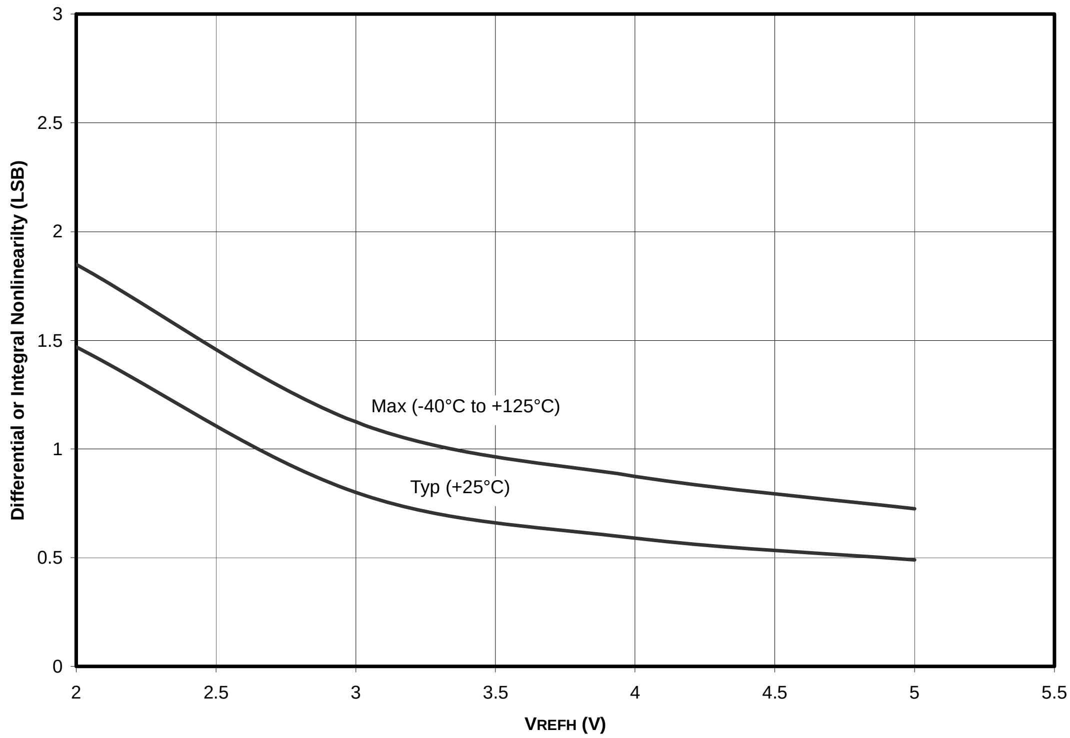
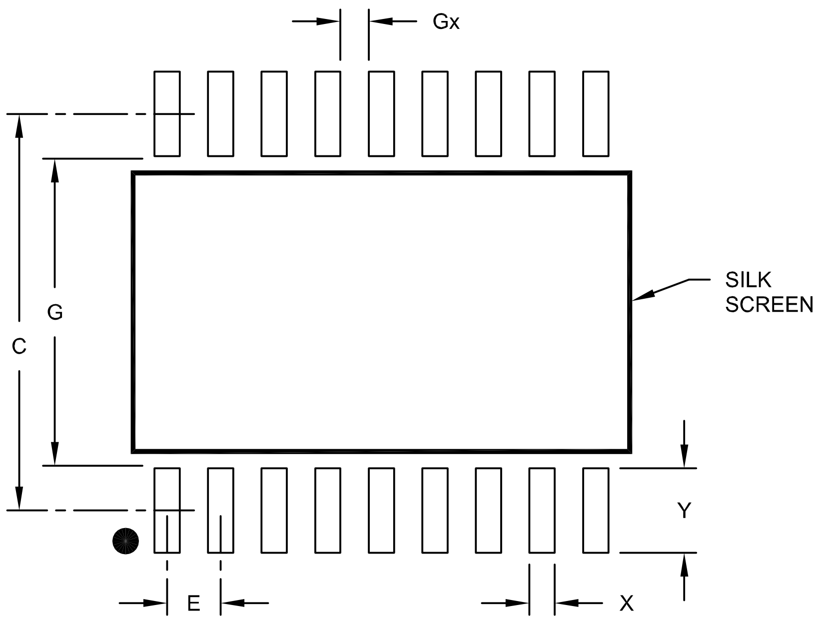


**FIGURE 16-25: A/D NONLINEARITY vs. VREFH (VDD = 5V, -40°C TO +125°C)**



## 18-Lead Plastic Small Outline (SO) - Wide, 7.50 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 | | 9.40 | |
| Contact Pad Width | X | | | 0.60 |
| Contact Pad Length | Y | | | 2.00 |
| Distance Between Pads | Gx | 0.67 | | |
| Distance Between Pads | G | 7.40 | | |

Notes:

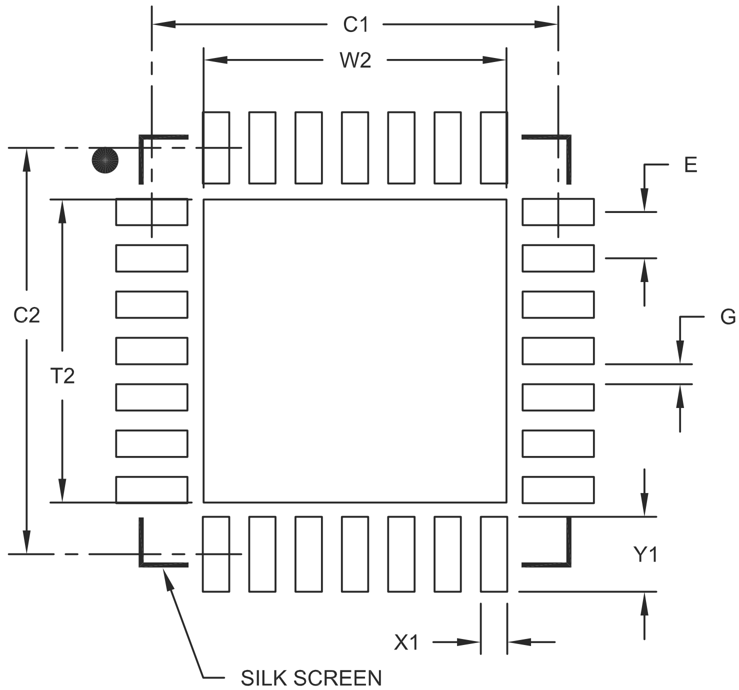
1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2051A

## 28-Lead Plastic Quad Flat, No Lead Package (ML) – 6x6 mm Body [QFN] with 0.55 mm Contact Length

| **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 | | 0.65 BSC | |
| Optional Center Pad Width | W2 | | | 4.25 |
| Optional Center Pad Length | T2 | | | 4.25 |
| Contact Pad Spacing | C1 | | 5.70 | |
| Contact Pad Spacing | C2 | | 5.70 | |
| Contact Pad Width (X28) | X1 | | | 0.37 |
| Contact Pad Length (X28) | Y1 | | | 1.00 |
| Distance Between Pads | G | 0.20 | | |

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2105A

## 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**