**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, UART/USART |
| Peripherals | Brown-out Detect/Reset, POR, PWM, WDT |
| Number of I/O | 25 |
| Program Memory Size | 7KB (4K x 14) |
| Program Memory Type | FLASH |
| EEPROM Size | - |
| RAM Size | 368 x 8 |
| Voltage - Supply (Vcc/Vdd) | 4V ~ 5.5V |
| Data Converters | A/D 11x10b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 28-SSOP (0.209", 5.30mm Width) |
| Supplier Device Package | 28-SSOP |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16f737-i-ss |

**TABLE 1-3: PIC16F747 AND PIC16F777 PINOUT DESCRIPTION**

| Pin Name | PDIP Pin # | QFN Pin # | TQFP Pin # | I/O/P Type | Buffer Type | Description |
|---|---|---|---|---|---|---|
| OSC1/CLKI/RA7 | 13 | 32 | 30 | | ST/CMOS(4) | Oscillator crystal or external clock input. |
| OSC1 | | | | I | | Oscillator crystal input or external clock source input. ST buffer when configured in RC mode; otherwise CMOS. |
| CLKI | | | | I | | External clock source input. Always associated with pin function OSC1 (see OSC1/CLKI, OSC2/CLKO pins). |
| RA7 | | | | I/O | ST | Bidirectional I/O pin. |
| OSC2/CLKO/RA6 | 14 | 33 | 31 | | — | Oscillator crystal or clock output. |
| OSC2 | | | | O | | Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. |
| CLKO | | | | O | | In RC mode, OSC2 pin outputs CLKO which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate. |
| RA6 | | | | I/O | ST | Bidirectional I/O pin. |
| $\overline{MCLR}$/VPP/RE3 | 1 | 18 | 18 | | ST | Master Clear (input) or programming voltage (output). |
| $\overline{MCLR}$ | | | | I | | Master Clear (Reset) input. This pin is an active-low Reset to the device. |
| VPP | | | | P | | Programming voltage input. |
| RE3 | | | | I | ST | Digital input only pin. |
| | | | | | | PORTA is a bidirectional I/O port. |
| RA0/AN0 | 2 | 19 | 19 | | TTL | |
| RA0 | | | | I/O | | Digital I/O. |
| AN0 | | | | I | | Analog input 0. |
| RA1/AN1 | 3 | 20 | 20 | | TTL | |
| RA1 | | | | I/O | | Digital I/O. |
| AN1 | | | | I | | Analog input 1. |
| RA2/AN2/VREF-/CVREF | 4 | 21 | 21 | | TTL | |
| RA2 | | | | I/O | | Digital I/O. |
| AN2 | | | | I | | Analog input 2. |
| VREF- | | | | I | | A/D reference voltage input (low). |
| CVREF | | | | I | | Comparator voltage reference output. |
| RA3/AN3/VREF+ | 5 | 22 | 22 | | TTL | |
| RA3 | | | | I/O | | Digital I/O. |
| AN3 | | | | I | | Analog input 3. |
| VREF+ | | | | I | | A/D reference voltage input (high). |
| RA4/T0CKI/C1OUT | 6 | 23 | 23 | | ST | |
| RA4 | | | | I/O | | Digital I/O – Open-drain when configured as output. |
| T0CKI | | | | I | | Timer0 external clock input. |
| C1OUT | | | | O | | Comparator 1 output. |
| RA5/AN4/LVDIN/$\overline{SS}$/C2OUT | 7 | 24 | 24 | | TTL | |
| RA5 | | | | I/O | | Digital I/O. |
| AN4 | | | | I | | Analog input 4. |
| LVDIN | | | | I | | Low-Voltage Detect input. |
| $\overline{SS}$ | | | | I | | SPI slave select input. |
| C2OUT | | | | I | | Comparator 2 output. |

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

**Note 1:** This buffer is a Schmitt Trigger input when configured as an 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 as a general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).

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

**5:** Pin location of CCP2 is determined by the CCPMX bit in Configuration Word Register 1.

**TABLE 1-3: PIC16F747 AND PIC16F777 PINOUT DESCRIPTION (CONTINUED)**

| Pin Name | PDIP Pin # | QFN Pin # | TQFP Pin # | I/O/P Type | Buffer Type | Description |
|---|---|---|---|---|---|---|
| | | | | | | PORTD is a bidirectional I/O port or Parallel Slave Port when interfacing to a microprocessor bus. |
| RD0/PSP0 | 19 | 38 | 38 | | ST/TTL(3) | |
| RD0 | | | | I/O | | Digital I/O. |
| PSP0 | | | | I/O | | Parallel Slave Port data. |
| RD1/PSP1 | 20 | 39 | 39 | | ST/TTL(3) | |
| RD1 | | | | I/O | | Digital I/O. |
| PSP1 | | | | I/O | | Parallel Slave Port data. |
| RD2/PSP2 | 21 | 40 | 40 | | ST/TTL(3) | |
| RD2 | | | | I/O | | Digital I/O. |
| PSP2 | | | | I/O | | Parallel Slave Port data. |
| RD3/PSP3 | 22 | 41 | 41 | | ST/TTL(3) | |
| RD3 | | | | I/O | | Digital I/O. |
| PSP3 | | | | I/O | | Parallel Slave Port data. |
| RD4/PSP4 | 27 | 2 | 2 | | ST/TTL(3) | |
| RD4 | | | | I/O | | Digital I/O. |
| PSP4 | | | | I/O | | Parallel Slave Port data. |
| RD5/PSP5 | 28 | 3 | 3 | | ST/TTL(3) | |
| RD5 | | | | I/O | | Digital I/O. |
| PSP5 | | | | I/O | | Parallel Slave Port data. |
| RD6/PSP6 | 29 | 4 | 4 | | ST/TTL(3) | |
| RD6 | | | | I/O | | Digital I/O. |
| PSP6 | | | | I/O | | Parallel Slave Port data. |
| RD7/PSP7 | 30 | 5 | 5 | | ST/TTL(3) | |
| RD7 | | | | I/O | | Digital I/O. |
| PSP7 | | | | I/O | | Parallel Slave Port data. |
| | | | | | | PORTE is a bidirectional I/O port. |
| RE0/$\overline{RD}$/AN5 | 8 | 25 | 25 | | ST/TTL(3) | |
| RE0 | | | | I/O | | Digital I/O. |
| $\overline{RD}$ | | | | I | | Read control for Parallel Slave Port. |
| AN5 | | | | I | | Analog input 5. |
| RE1/$\overline{WR}$/AN6 | 9 | 26 | 26 | | ST/TTL(3) | |
| RE1 | | | | I/O | | Digital I/O. |
| $\overline{WR}$ | | | | I | | Write control for Parallel Slave Port. |
| AN6 | | | | I | | Analog input 6. |
| RE2/$\overline{CS}$/AN7 | 10 | 27 | 27 | | ST/TTL(3) | |
| RE2 | | | | I/O | | Digital I/O. |
| $\overline{CS}$ | | | | I | | Chip select control for Parallel Slave Port. |
| AN7 | | | | I | | Analog input 7. |
| VSS | — | 31 | — | P | — | Analog ground reference. |
| VSS | 12, 31 | 6, 30 | 6, 29 | P | — | Ground reference for logic and I/O pins. |
| VDD | — | 8 | — | P | — | Analog positive supply. |
| VDD | 11, 32 | 7, 28 | 7, 28 | P | — | Positive supply for logic and I/O pins. |
| NC | — | 13, 29 | 12, 13, 33, 34 | — | — | These pins are not internally connected. These pins should be left unconnected. |

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

**Note 1:** This buffer is a Schmitt Trigger input when configured as an 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 as a general purpose I/O and a TTL input when used in the Parallel Slave Port mode (for interfacing to a microprocessor bus).

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

**5:** Pin location of CCP2 is determined by the CCPMX bit in Configuration Word Register 1.

## 3.0 READING PROGRAM MEMORY

The Flash program memory is readable during normal operation over the entire VDD range. It is indirectly addressed through Special Function Registers (SFR). Up to 14-bit numbers can be stored in memory for use as calibration parameters, serial numbers, packed 7-bit ASCII, etc. Executing a program memory location containing data that forms an invalid instruction results in a NOP.

There are five SFRs used to read the program and memory. These registers are:

- PMCON1
- PMDATA
- PMDATH
- PMADR
- PMADRH

The program memory allows word reads. Program memory access allows for checksum calculation and reading calibration tables.

When interfacing to the program memory block, the PMDATH:PMDATA registers form a two-byte word which holds the 14-bit data for reads. The PMADRH:PMADR registers form a two-byte word which holds the 13-bit address of the Flash location being accessed. These devices can have up to 8K words of program Flash, with an address range from 0h to 3FFFh. The unused upper bits in both the PMDATH and PMADRH registers are not implemented and read as '0's.

### 3.1 PMADR

The address registers can address up to a maximum of 8K words of program Flash.

When selecting a program address value, the MSB of the address is written to the PMADRH register and the LSB is written to the PMADR register. The upper Most Significant bits of PMADRH must always be clear.

### 3.2 PMCON1 Register

PMCON1 is the control register for memory accesses.

The control bit, RD, initiates read operations. This bit cannot be cleared, only set, in software. It is cleared in hardware at the completion of the read operation.

**REGISTER 3-1: PMCON1: PROGRAM MEMORY CONTROL REGISTER 1 (ADDRESS 18Ch)**

| R-1 | U-0 | U-0 | U-0 | U-x | U-0 | U-0 | R/S-0 |
|---|---|---|---|---|---|---|---|
| reserved | — | — | — | — | — | — | RD |
| bit 7 | | | | | | | bit 0 |

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

bit 6-1 **Unimplemented**: Read as '0'

bit 0 **RD**: Read Control bit

1 = Initiates a Flash read, RD is cleared in hardware. The RD bit can only be set (not cleared) in software.

0 = Flash read completed

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

### 4.5.1 INTRC MODES

Using the internal oscillator as the clock source can eliminate the need for up to two external oscillator pins, after which it can be used for digital I/O. Two distinct configurations are available:

- In INTIO1 mode, the OSC2 pin outputs FOSC/4, while OSC1 functions as RA7 for digital input and output.
- In INTIO2 mode, OSC1 functions as RA7 and OSC2 functions as RA6, both for digital input and output.

### 4.5.2 OSCTUNE REGISTER

The internal oscillator's output has been calibrated at the factory but can be adjusted in the application. This is done by writing to the OSCTUNE register (Register 4-1). The tuning sensitivity is constant throughout the tuning range. The OSCTUNE register has a tuning range of ±12.5%.

When the OSCTUNE register is modified, the INTOSC and INTRC frequencies will begin shifting to the new frequency. The INTRC clock will reach the new frequency within 8 clock cycles (approximately 8 * 32 μs = 256 μs); the INTOSC clock will stabilize within 1 ms. Code execution continues during this shift. There is no indication that the shift has occurred. Operation of features that depend on the 31.25 kHz INTRC clock source frequency, such as the WDT, Fail-Safe Clock Monitor and peripherals, will also be affected by the change in frequency.

**REGISTER 4-1: OSCTUNE: OSCILLATOR TUNING REGISTER (ADDRESS 90h)**

| U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---|---|---|---|---|---|---|---|
| — | — | TUN5 | TUN4 | TUN3 | TUN2 | TUN1 | TUN0 |
| bit 7 | | | | | | | bit 0 |

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

bit 5-0 **TUN<5:0>:** Frequency Tuning bits

011111 = Maximum frequency
011110 =
•
•
•
000001 =
000000 = Center frequency. Oscillator module is running at the calibrated frequency.
111111 =
•
•
•
100000 = Minimum frequency

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

## 5.0 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 a 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).

Reading the PORTA register reads the status of the pins, whereas writing to it, will write to the port latch.

The RA4 pin is multiplexed with the Timer0 module clock input and one of the comparator outputs to become the RA4/T0CKI/C1OUT pin. Pins RA6 and RA7 are multiplexed with the main oscillator pins; they are enabled as oscillator or I/O pins by the selection of the main oscillator in Configuration Register 1H (see **Section 15.1 "Configuration Bits"** for details). When they are not used as port pins, RA6 and RA7 and their associated TRIS and LAT bits are read as '0'.

The other PORTA pins are multiplexed with analog inputs, the analog VREF+ and VREF- inputs and the comparator voltage reference output. The operation of pins RA3:RA0 and RA5 as A/D converter inputs is selected by clearing/setting the control bits in the ADCON1 register (A/D Control Register 1). Pins RA0 through RA5 may also be used as comparator inputs or outputs by setting the appropriate bits in the CMCON register.

| **Note:** | On a Power-on Reset, RA5 and RA3:RA0 are configured as analog inputs and read as '0'. RA4 is configured as a digital input. |
|---|---|

The RA4/T0CKI/C1OUT pin is a Schmitt Trigger input and an open-drain output. All other PORTA pins have TTL input levels and full CMOS output drivers.

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.

**EXAMPLE 5-1: INITIALIZING PORTA**

```
BCF     STATUS, RP0   ;
BCF     STATUS, RP1   ; Bank0
CLRF    PORTA         ; Initialize PORTA by
                      ; clearing output
                      ; data latches
BSF     STATUS, RP0   ; Select Bank 1
MOVLW   0x0F          ; Configure all pins
MOVWF   ADCON1        ; as digital inputs
MOVLW   0xCF          ; Value used to
                      ; initialize data
                      ; direction
MOVWF   TRISA         ; Set RA<3:0> as inputs
                      ; RA<5:4> as outputs
                      ; TRISA<7:6>are always
                      ; read as '0'.
```

**FIGURE 5-1: BLOCK DIAGRAM OF RA0/AN0:RA1/AN1 PINS**

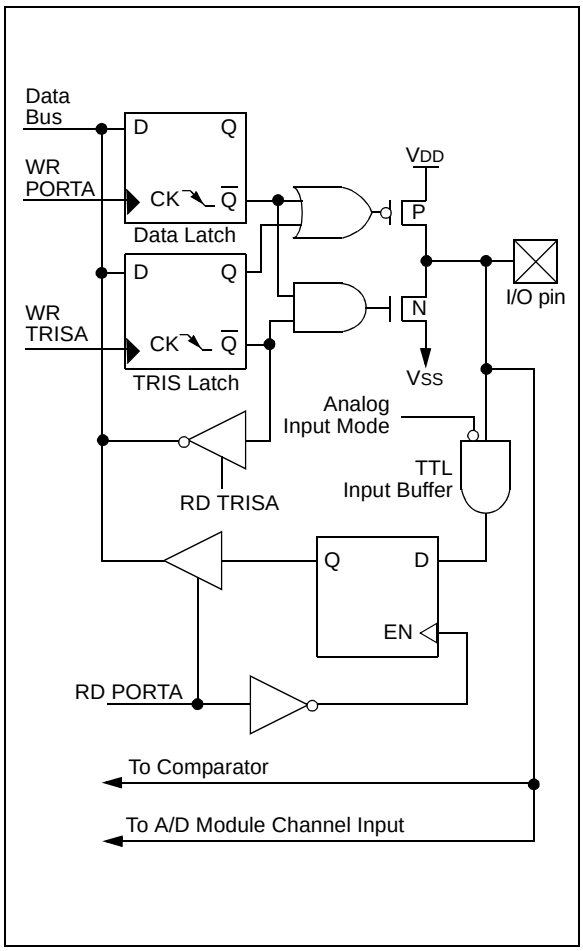
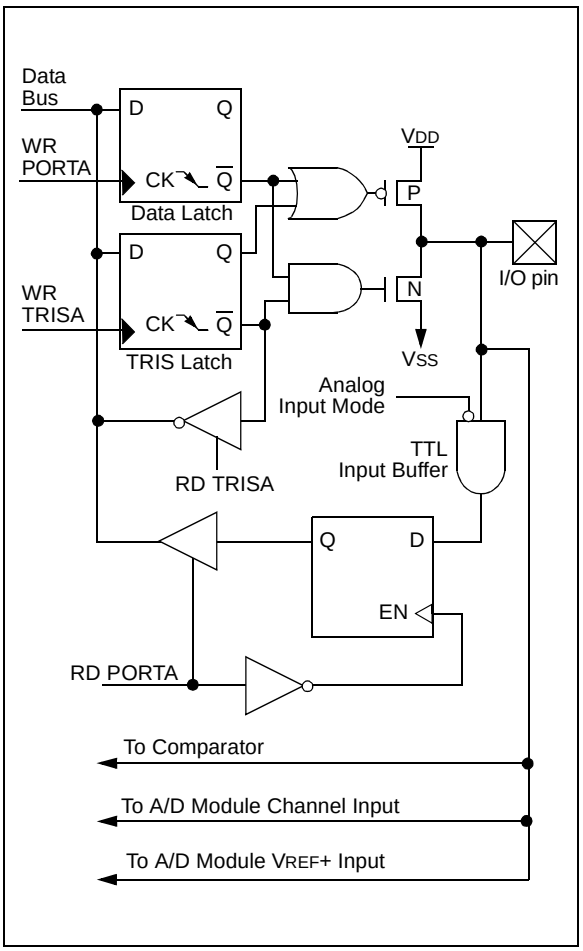



**FIGURE 5-2: BLOCK DIAGRAM OF RA3/AN3/VREF+ PIN**



## 6.0 TIMER0 MODULE

The Timer0 module timer/counter has the following features:

- 8-bit timer/counter
- Readable and writable
- 8-bit software programmable prescaler
- Internal or external clock select
- Interrupt on overflow from FFh to 00h
- Edge select for external clock

Additional information on the Timer0 module is available in the *"PIC® Mid-Range MCU Family Reference Manual"* (DS33023).

Figure 6-1 is a block diagram of the Timer0 module and the prescaler shared with the WDT.

### 6.1 Timer0 Operation

Timer0 operation is controlled through the OPTION_REG register (see Register 2-2). 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.

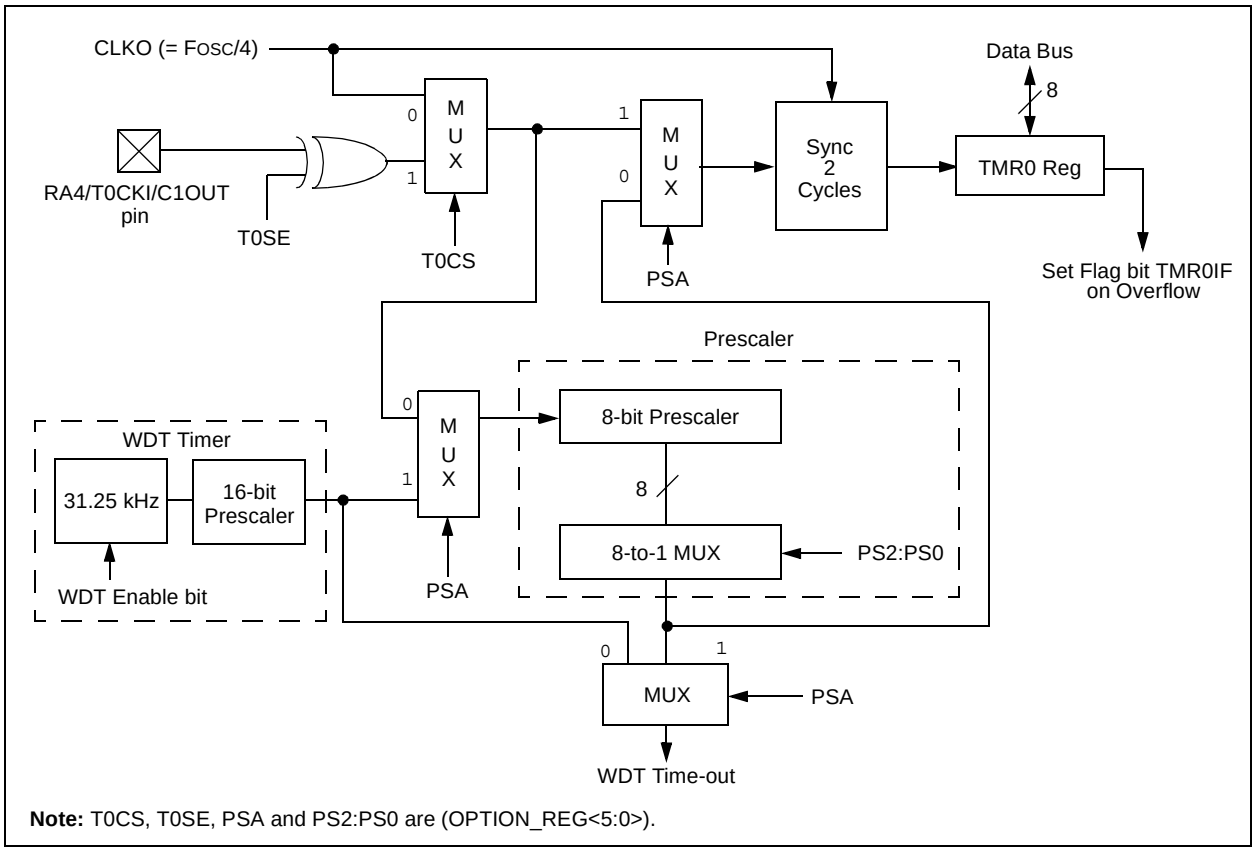
Counter mode is selected by setting bit, T0CS (OPTION_REG<5>). In Counter mode, Timer0 will increment, either on every rising or falling edge of pin RA4/T0CKI/C1OUT. The incrementing edge is determined by the Timer0 Source Edge Select bit, T0SE (OPTION_REG<4>). Clearing bit T0SE selects the rising edge. Restrictions on the external clock input are discussed in detail in **Section 6.3 "Using Timer0 With an External Clock"**.

The prescaler is mutually, exclusively shared between the Timer0 module and the Watchdog Timer. The prescaler is not readable or writable. **Section 6.4 "Prescaler"** details the operation of the prescaler.

### 6.2 Timer0 Interrupt

The TMR0 interrupt is generated when the TMR0 register overflows from FFh to 00h. This overflow sets bit TMR0IF (INTCON<2>). The interrupt can be masked by clearing bit TMR0IE (INTCON<5>). Bit TMR0IF 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 6-1: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER**




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

**REGISTER 10-2: SSPCON: MSSP CONTROL (SPI MODE) REGISTER 1 (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 |
| bit 7 | | | | | | | bit 0 |

bit 7 **WCOL:** Write Collision Detect bit (Transmit mode only)

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

SPI Slave 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. (Must be cleared in software.)

0 = No overflow

**Note:** In Master mode, the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPBUF register.

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

1 = Enables serial port and configures SCK, SDO, SDI and $\overline{SS}$ as serial port pins

0 = Disables serial port and configures these pins as I/O port pins

**Note:** When enabled, these pins must be properly configured as input or output.

bit 4 **CKP:** Clock Polarity Select bit

1 = Idle state for clock is a high level

0 = Idle state for clock is a low level

bit 3-0 **SSPM3:SSPM0:** Synchronous Serial Port Mode Select bits

0101 = SPI Slave mode, clock = SCK pin. $\overline{SS}$ pin control disabled. $\overline{SS}$ can be used as I/O pin.

0100 = SPI Slave mode, clock = SCK pin. $\overline{SS}$ pin control enabled.

0011 = SPI Master mode, clock = TMR2 output/2

0010 = SPI Master mode, clock = FOSC/64

0001 = SPI Master mode, clock = FOSC/16

0000 = SPI Master mode, clock = FOSC/4

**Note:** Bit combinations not specifically listed here are either reserved or implemented in $I^2C$ mode only.

| **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 |

### 10.3.5 MASTER MODE

The master can initiate the data transfer at any time because it controls the SCK. The master determines when the slave (Processor 2, Figure 10-2) is to broadcast data by the software protocol.

In Master mode, the data is transmitted/received as soon as the SSPBUF register is written to. If the SPI is only going to receive, the SDO output could be disabled (programmed as an input). The SSPSR register will continue to shift in the signal present on the SDI pin at the programmed clock rate. As each byte is received, it will be loaded into the SSPBUF register as if it is a normal received byte (interrupts and status bits appropriately set). This could be useful in receiver applications, such as a “Line Activity Monitor” mode.

The clock polarity is selected by appropriately programming the CKP bit (SSPCON<4>). This then, would give waveforms for SPI communication as shown in Figure 10-3, Figure 10-5 and Figure 10-6, where the MSB is transmitted first. In Master mode, the SPI clock rate (bit rate) is user programmable to be one of the following:

- FOSC/4 (or TCY)
- FOSC/16 (or 4 • TCY)
- FOSC/64 (or 16 • TCY)
- Timer2 output/2

This allows a maximum data rate (at 40 MHz) of 10.00 Mbps.

Figure 10-3 shows the waveforms for Master mode. When the CKE bit is set, the SDO data is valid before there is a clock edge on SCK. The change of the input sample is shown based on the state of the SMP bit. The time when the SSPBUF is loaded with the received data is shown.

**FIGURE 10-3: SPI MODE WAVEFORM (MASTER MODE)**

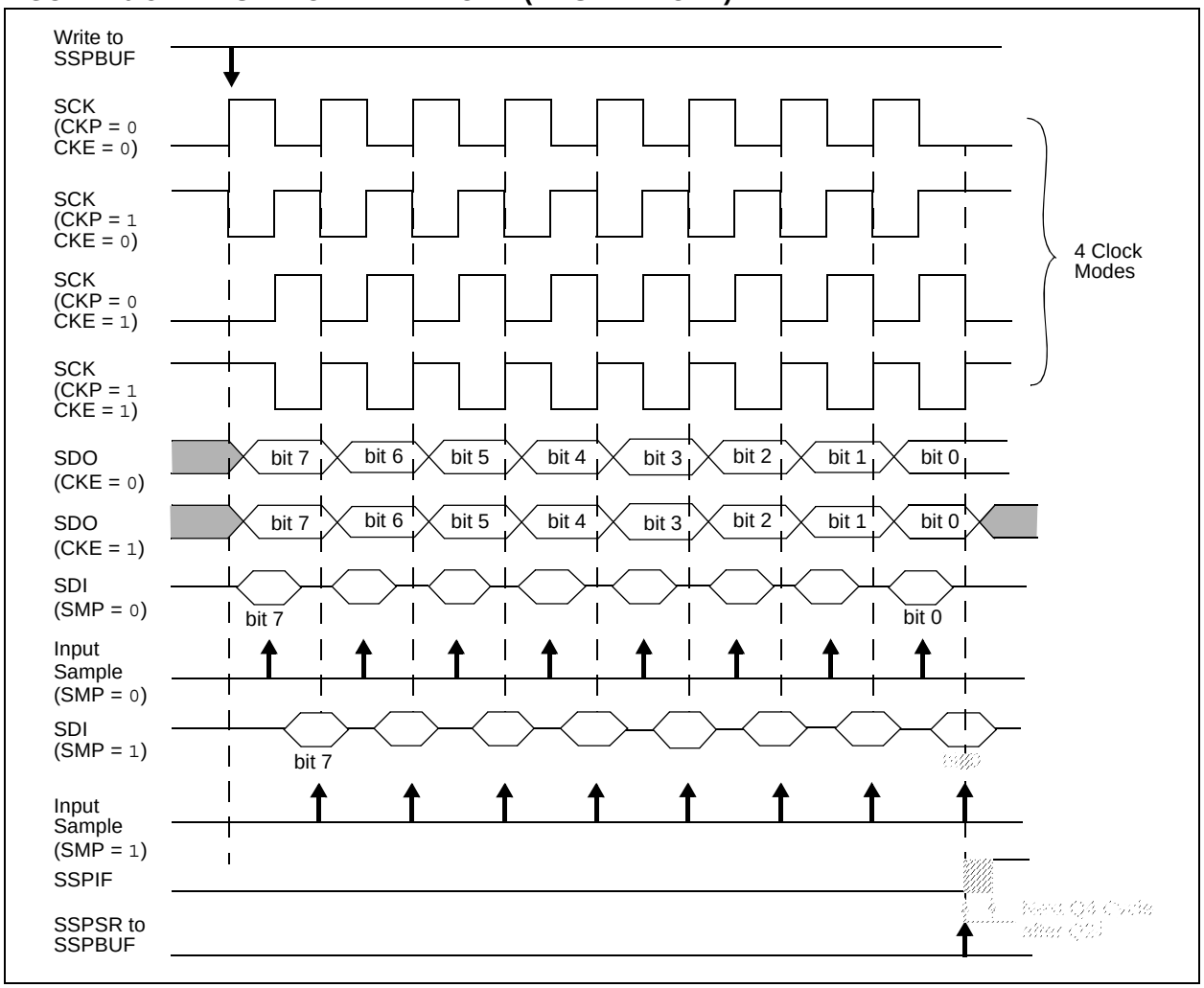
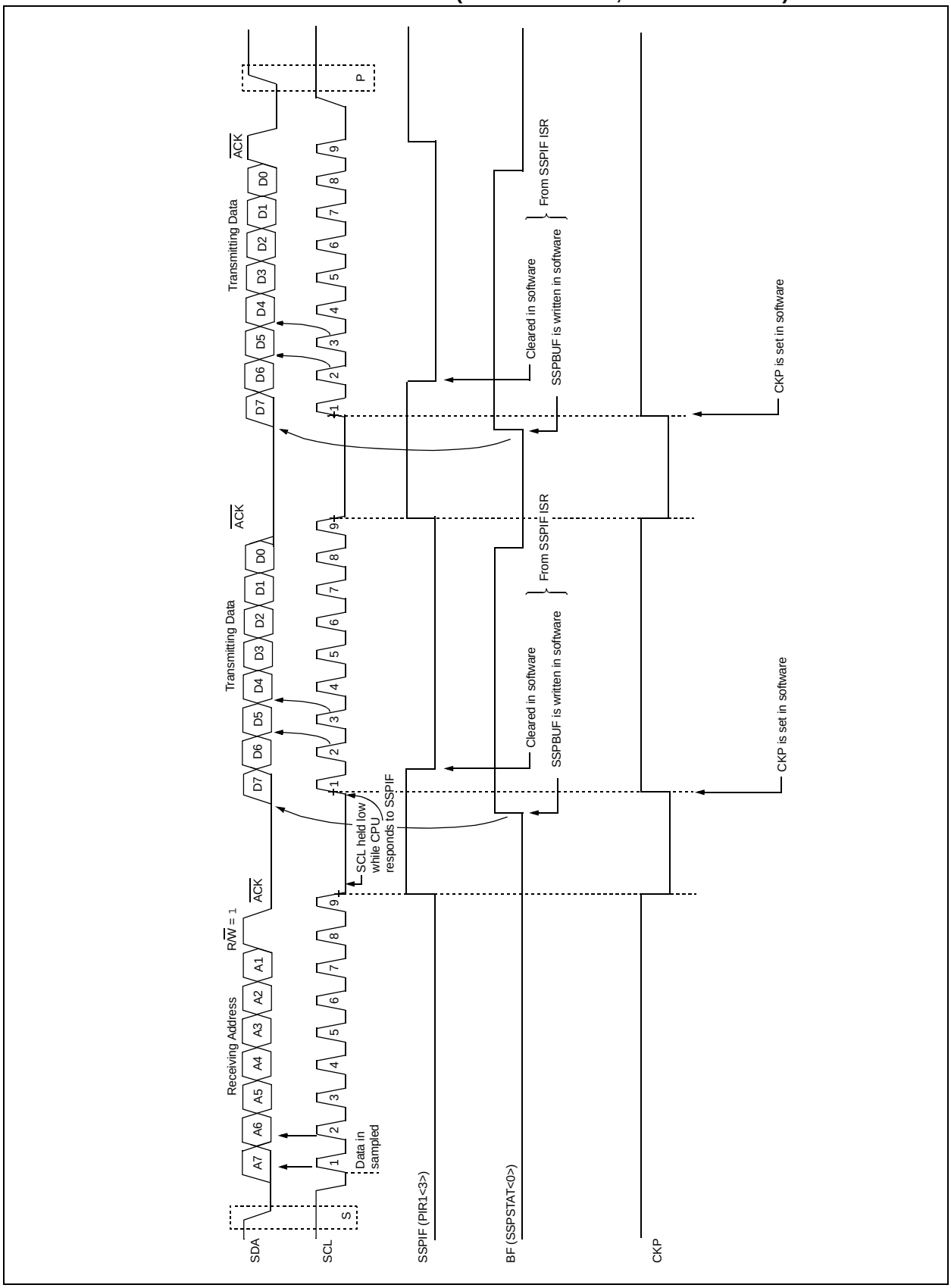


**FIGURE 10-9: I²C™ SLAVE MODE TIMING (TRANSMISSION, 7-BIT ADDRESS)**



#### 10.4.6.1 I²C Master Mode Operation

The master device generates all of the serial clock pulses and the Start and Stop conditions. A transfer is ended with a Stop condition or with a Repeated Start condition. Since the Repeated Start condition is also the beginning of the next serial transfer, the I²C bus will not be released.

In Master Transmitter mode, serial data is output through SDA while SCL outputs the serial clock. The first byte transmitted contains the slave address of the receiving device (7 bits) and the Read/Write (R/$\overline{W}$) bit. In this case, the R/$\overline{W}$ bit will be logic '0'. Serial data is transmitted 8 bits at a time. After each byte is transmitted, an Acknowledge bit is received. Start and Stop conditions are output to indicate the beginning and the end of a serial transfer.

In Master Receive mode, the first byte transmitted contains the slave address of the transmitting device (7 bits) and the R/$\overline{W}$ bit. In this case, the R/$\overline{W}$ bit will be logic '1'. Thus, the first byte transmitted is a 7-bit slave address followed by a '1' to indicate a receive bit. Serial data is received via SDA, while SCL outputs the serial clock. Serial data is received 8 bits at a time. After each byte is received, an Acknowledge bit is transmitted. Start and Stop conditions indicate the beginning and end of transmission.

The Baud Rate Generator used for the SPI mode operation is used to set the SCL clock frequency for either 100 kHz, 400 kHz or 1 MHz I²C operation. See **Section 10.4.7 "Baud Rate Generator"** for more detail.

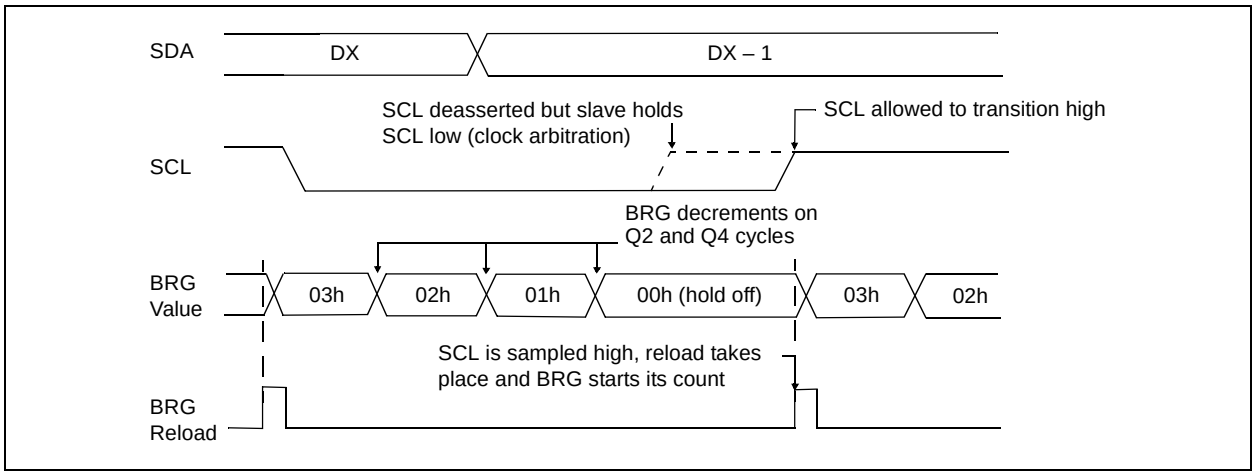
A typical transmit sequence would go as follows:

1. The user generates a Start condition by setting the Start enable bit, SEN (SSPCON2<0>).
2. SSPIF is set. The MSSP module will wait the required Start time before any other operation takes place.
3. The user loads the SSPBUF with the slave address to transmit.
4. Address is shifted out the SDA pin until all 8 bits are transmitted.
5. The MSSP module shifts in the ACK bit from the slave device and writes its value into the SSPCON2 register (SSPCON2<6>).
6. The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSPIF bit.
7. The user loads the SSPBUF with eight bits of data.
8. Data is shifted out the SDA pin until all 8 bits are transmitted.
9. The MSSP module shifts in the ACK bit from the slave device and writes its value into the SSPCON2 register (SSPCON2<6>).
10. The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSPIF bit.
11. The user generates a Stop condition by setting the Stop enable bit, PEN (SSPCON2<2>).
12. Interrupt is generated once the Stop condition is complete.

#### 10.4.7.1 Clock Arbitration

Clock arbitration occurs when the master, during any receive, transmit or Repeated Start/Stop condition, deasserts the SCL pin (SCL allowed to float high). When the SCL pin is allowed to float high, the Baud Rate Generator (BRG) is suspended from counting until the SCL pin is actually sampled high. When the SCL pin is sampled high, the Baud Rate Generator is reloaded with the contents of SSPADD<6:0> and begins counting. This ensures that the SCL high time will always be at least one BRG rollover count in the event that the clock is held low by an external device (Figure 10-18).

**FIGURE 10-18: BAUD RATE GENERATOR TIMING WITH CLOCK ARBITRATION**



### 10.4.9 I²C MASTER MODE REPEATED START CONDITION TIMING

A Repeated Start condition occurs when the RSEN bit (SSPCON2<1>) is programmed high and the I²C logic module is in the Idle state. When the RSEN bit is set, the SCL pin is asserted low. When the SCL pin is sampled low, the Baud Rate Generator is loaded with the contents of SSPADD<5:0> and begins counting. The SDA pin is released (brought high) for one Baud Rate Generator count (TBRG). When the Baud Rate Generator times out, if SDA is sampled high, the SCL pin will be deasserted (brought high). When SCL is sampled high, the Baud Rate Generator is reloaded with the contents of SSPADD<6:0> and begins counting. SDA and SCL must be sampled high for one TBRG. This action is then followed by assertion of the SDA pin (SDA = 0) for one TBRG while SCL is high. Following this, the RSEN bit (SSPCON2<1>) will be automatically cleared and the Baud Rate Generator will not be reloaded, leaving the SDA pin held low. As soon as a Start condition is detected on the SDA and SCL pins, the S bit (SSPSTAT<3>) will be set. The SSPIF bit will not be set until the Baud Rate Generator has timed out.

**Note 1:** If RSEN is programmed while any other event is in progress, it will not take effect.

**2:** A bus collision during the Repeated Start condition occurs if:

- SDA is sampled low when SCL goes from low-to-high.
- SCL goes low before SDA is asserted low. This may indicate that another master is attempting to transmit a data '1'.

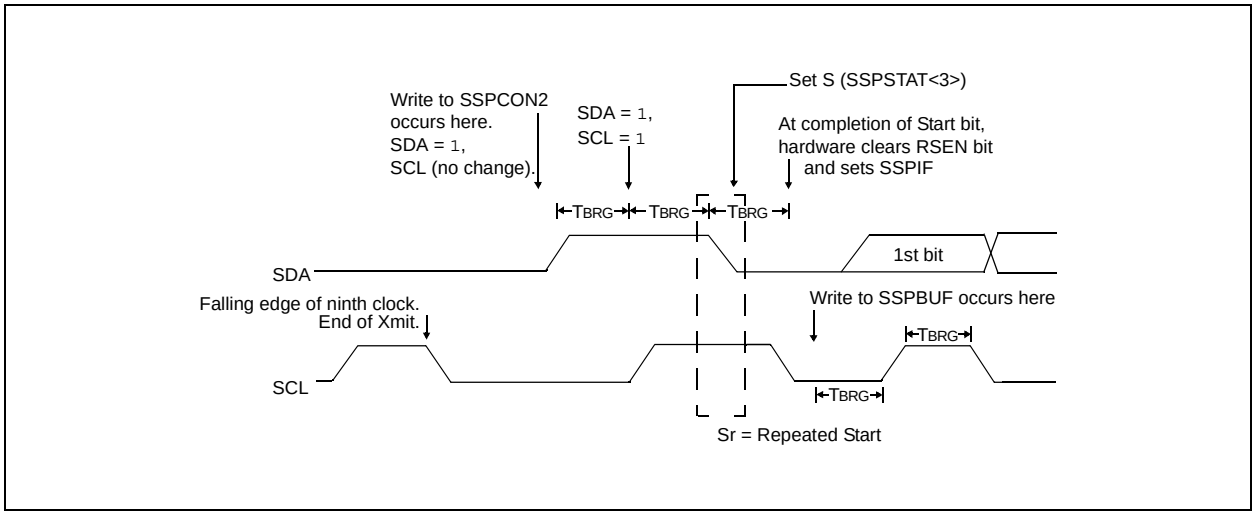
Immediately following the SSPIF bit getting set, the user may write the SSPBUF with the 7-bit address in 7-bit mode or the default first address in 10-bit mode. After the first eight bits are transmitted and an ACK is received, the user may then transmit an additional eight bits of address (10-bit mode) or eight bits of data (7-bit mode).

#### 10.4.9.1 WCOL Status Flag

If the user writes the SSPBUF when a Repeated Start sequence is in progress, the WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

**Note:** Because queueing of events is not allowed, writing of the lower 5 bits of SSPCON2 is disabled until the Repeated Start condition is complete.

**FIGURE 10-20: REPEATED START CONDITION WAVEFORM**



### 10.4.14 SLEEP OPERATION

While in Sleep mode, the $I^2C$ module can receive addresses or data and when an address match or complete byte transfer occurs, wake the processor from Sleep (if the MSSP interrupt is enabled).

### 10.4.15 EFFECT OF A RESET

A Reset disables the MSSP module and terminates the current transfer.

### 10.4.16 MULTI-MASTER MODE

In Multi-Master mode, 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 MSSP module is disabled. Control of the $I^2C$ bus may be taken when the P bit (SSPSTAT<4>) is set or the bus is Idle, with 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 for arbitration to see if the signal level is at the expected output level. This check is performed in hardware with the result placed in the BCLIF bit.

The states where arbitration can be lost are:

- Address Transfer
- Data Transfer
- A Start Condition
- A Repeated Start Condition
- An Acknowledge Condition

### 10.4.17 MULTI-MASTER COMMUNICATION, BUS COLLISION AND BUS ARBITRATION

Multi-Master mode support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA by letting SDA float high and another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin = 0, then a bus collision has taken place. The master will set the Bus Collision Interrupt Flag, BCLIF and reset the $I^2C$ port to its Idle state (Figure 10-25).

If a transmit was in progress when the bus collision occurred, the transmission is halted, the BF flag is cleared, the SDA and SCL lines are deasserted and the SSPBUF can be written to. When the user services the bus collision Interrupt Service Routine and if the $I^2C$ bus is free, the user can resume communication by asserting a Start condition.

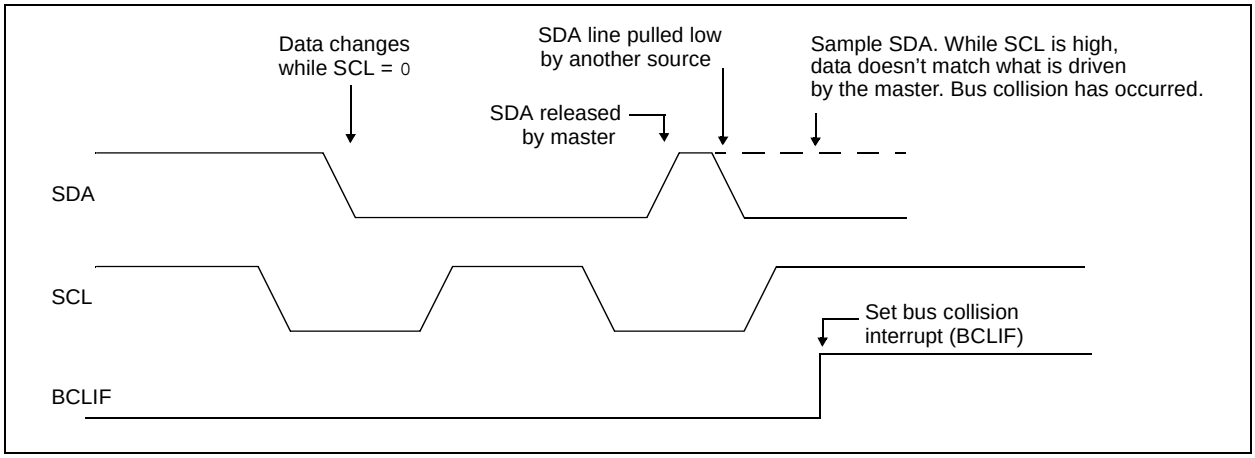
If a Start, Repeated Start, Stop or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are deasserted and the respective control bits in the SSPCON2 register are cleared. When the user services the bus collision Interrupt Service Routine and if the $I^2C$ bus is free, the user can resume communication by asserting a Start condition.

The master will continue to monitor the SDA and SCL pins. If a Stop condition occurs, the SSPIF bit will be set.

A write to the SSPBUF will start the transmission of data at the first data bit, regardless of where the transmitter left off when the bus collision occurred.

In Multi-Master mode, the interrupt generation on the detection of Start and Stop conditions allows the determination of when the bus is free. Control of the $I^2C$ bus can be taken when the P bit is set in the SSPSTAT register or the bus is Idle and the S and P bits are cleared.

**FIGURE 10-25: BUS COLLISION TIMING FOR TRANSMIT AND ACKNOWLEDGE**



## 12.2 Selecting and Configuring Automatic Acquisition Time

The ADCON2 register allows the user to select an acquisition time that occurs each time the GO/DONE bit is set.

When the GO/DONE bit is set, sampling is stopped and a conversion begins. The user is responsible for ensuring the required acquisition time has passed between selecting the desired input channel and setting the GO/DONE bit. This occurs when the ACQT2:ACQT0 bits (ADCON2<5:3>) remain in their Reset state ('000') and is compatible with devices that do not offer programmable acquisition times.

If desired, the ACQT bits can be set to select a programmable acquisition time for the A/D module. When the GO/DONE bit is set, the A/D module continues to sample the input for the selected acquisition time, then automatically begins a conversion. Since the acquisition time is programmed, there may be no need to wait for an acquisition time between selecting a channel and setting the GO/DONE bit.

In either case, when the conversion is completed, the GO/DONE bit is cleared, the ADIF flag is set and the A/D begins sampling the currently selected channel again. If an acquisition time is programmed, there is nothing to indicate if the acquisition time has ended or if the conversion has begun.

## 12.3 Selecting the A/D Conversion Clock

The A/D conversion time per bit is defined as TAD. The A/D conversion requires a minimum 12 TAD per 10-bit conversion. The source of the A/D conversion clock is software selected. The seven possible options for TAD are:

- 2 TOSC
- 4 TOSC
- 8 TOSC
- 16 TOSC
- 32 TOSC
- 64 TOSC
- Internal A/D module, RC oscillator (2-6 μs)

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

Table 12-1 shows the resultant TAD times derived from the device operating frequencies and the A/D clock source selected.

**TABLE 12-1: TAD vs. MAXIMUM DEVICE OPERATING FREQUENCIES (STANDARD DEVICES (F))**

| AD Clock Source (TAD) | | Maximum Device Frequency |
|---|---|---|
| Operation | ADCS2:ADCS1:ADCS0 | |
| 2 TOSC | 000 | 1.25 MHz |
| 4 TOSC | 100 | 2.5 MHz |
| 8 TOSC | 001 | 5 MHz |
| 16 TOSC | 101 | 10 MHz |
| 32 TOSC | 010 | 20 MHz |
| 64 TOSC | 110 | 20 MHz |
| RC(1,2,3) | x11 | **(Note 1)** |

**Note 1:** The RC source has a typical TAD time of 4 μs but can vary between 2-6 μs.

**2:** When the device frequencies are greater than 1 MHz, the RC A/D conversion clock source is only recommended for Sleep operation.

**3:** For extended voltage devices (LF), please refer to **Section 18.0 "Electrical Characteristics"**.

## 17.7 MPLAB SIM Software Simulator

The MPLAB SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC® DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB SIM Software Simulator fully supports symbolic debugging using the MPLAB C Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

## 17.8 MPLAB REAL ICE In-Circuit Emulator System

MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs PIC® Flash MCUs and dsPIC® Flash DSCs with the easy-to-use, powerful graphical user interface of the MPLAB Integrated Development Environment (IDE), included with each kit.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB IDE. In upcoming releases of MPLAB IDE, new devices will be supported, and new features will be added. MPLAB REAL ICE offers significant advantages over competitive emulators including low-cost, full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, a ruggedized probe interface and long (up to three meters) interconnection cables.

## 17.9 MPLAB ICD 3 In-Circuit Debugger System

MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost effective high-speed hardware debugger/programmer for Microchip Flash Digital Signal Controller (DSC) and microcontroller (MCU) devices. It debugs and programs PIC® Flash microcontrollers and dsPIC® DSCs with the powerful, yet easy-to-use graphical user interface of MPLAB Integrated Development Environment (IDE).

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

## 17.10 PICkit 3 In-Circuit Debugger/ Programmer and PICkit 3 Debug Express

The MPLAB PICkit 3 allows debugging and programming of PIC® and dsPIC® Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB Integrated Development Environment (IDE). The MPLAB PICkit 3 is connected to the design engineer's PC using a full speed USB interface and can be connected to the target via an Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the reset line to implement in-circuit debugging and In-Circuit Serial Programming™.

The PICkit 3 Debug Express include the PICkit 3, demo board and microcontroller, hookup cables and CDROM with user's guide, lessons, tutorial, compiler and MPLAB IDE software.

**TABLE 18-15: A/D CONVERTER CHARACTERISTICS: PIC16F7X7 (INDUSTRIAL, EXTENDED) PIC16LF7X7 (INDUSTRIAL)**

| Param No. | Sym | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| A01 | NR | Resolution | | — | — | 10 bits | bit | 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 |
| A06 | EOFF | Offset Error | | — | — | <±2 | LSb | VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF |
| A07 | EGN | Gain Error | | — | — | <±1 | LSb | VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF |
| A10 | — | Monotonicity | | — | guaranteed(3) | — | — | VSS ≤ VAIN ≤ VREF |
| A20 | VREF | Reference Voltage (VREF+ – VREF-) | | 2.0 | — | VDD + 0.3 | V | |
| A21 | VREF+ | Reference Voltage High | | AVDD – 2.5V | — | AVDD + 0.3V | V | |
| A22 | VREF- | Reference Voltage Low | | AVSS – 0.3V | — | VREF+ – 2.0V | V | |
| A25 | VAIN | Analog Input Voltage | | VSS – 0.3V | — | VREF + 0.3V | V | |
| A30 | ZAIN | Recommended Impedance of Analog Voltage Source | | — | — | 2.5 | kΩ | **(Note 4)** |
| A40 | IAD | A/D Conversion Current (VDD) | PIC16F7X7 | — | 220 | — | μA | Average current consumption when A/D is on **(Note 1)** |
| | | | PIC16LF7X7 | — | 90 | — | μA | |
| A50 | IREF | VREF Input Current **(Note 2)** | | — | — | 5 | μA | During VAIN acquisition. Based on differential of VHOLD to VAIN to charge CHOLD, see **Section 12.1 "A/D Acquisition Requirements"**. |
| | | | | — | — | 150 | 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 specification 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.

**4:** Maximum allowed impedance for analog voltage source is 10 kΩ. This requires higher acquisition time.

**FIGURE 19-13: AVERAGE FOSC VS. VDD FOR VARIOUS VALUES OF R (RC MODE, C = 300 pF, +25°C)**

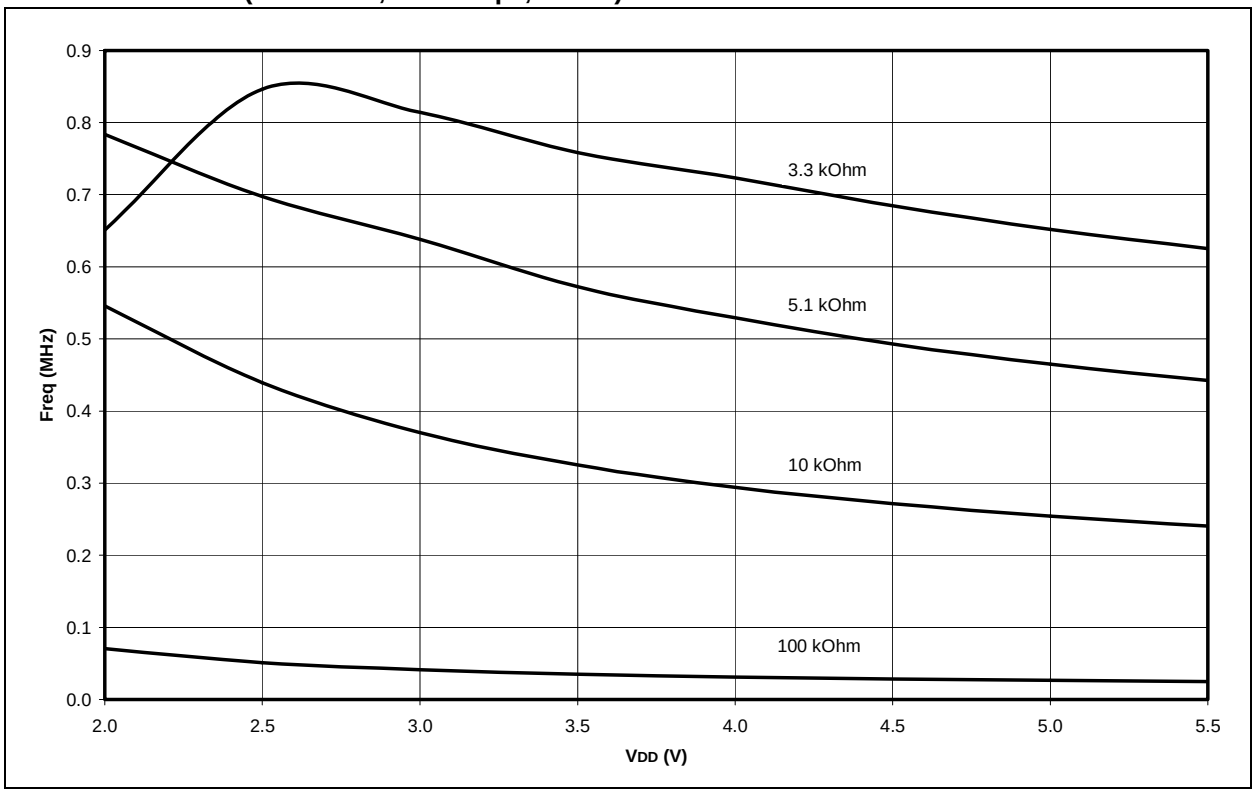



**FIGURE 19-14: ΔIPD TIMER1 OSCILLATOR, -10°C TO +70°C (SLEEP MODE, TMR1 COUNTER DISABLED)**

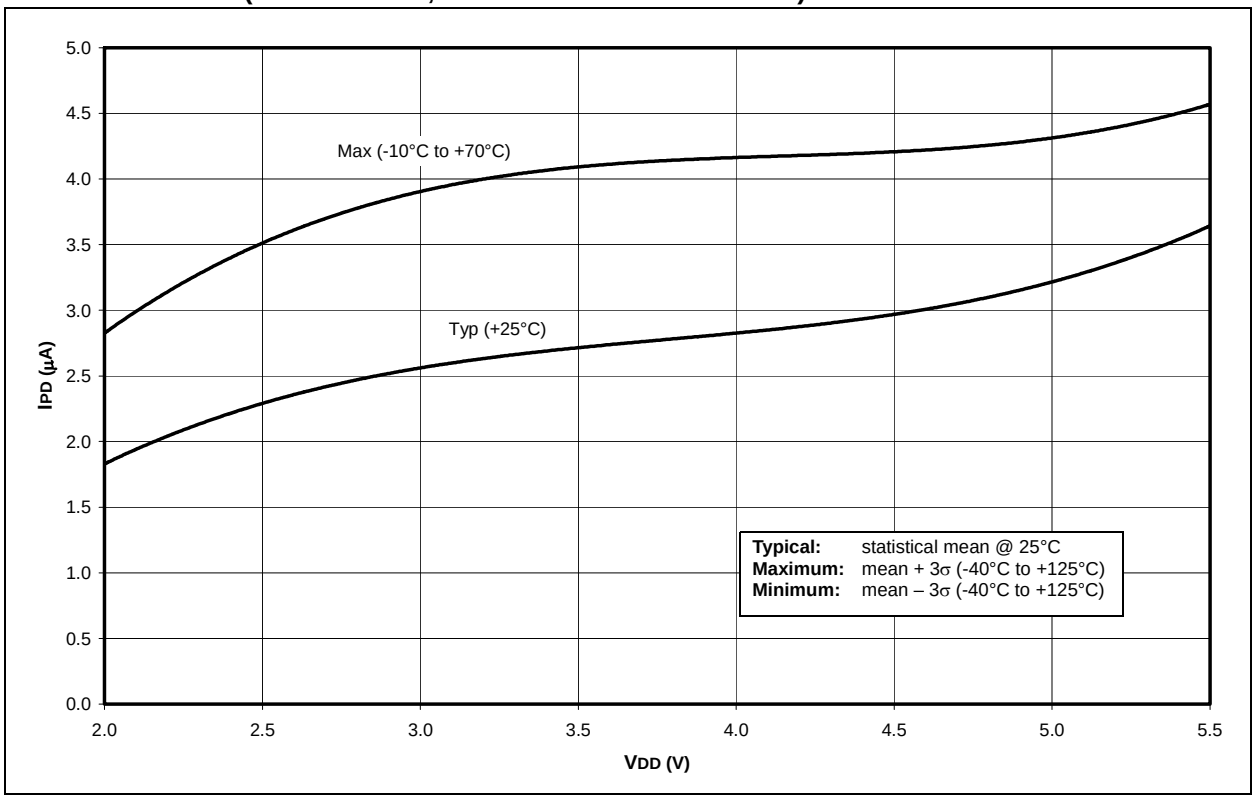


**FIGURE 19-23: MINIMUM AND MAXIMUM VIN vs. VDD (TTL INPUT, -40°C TO +125°C)**

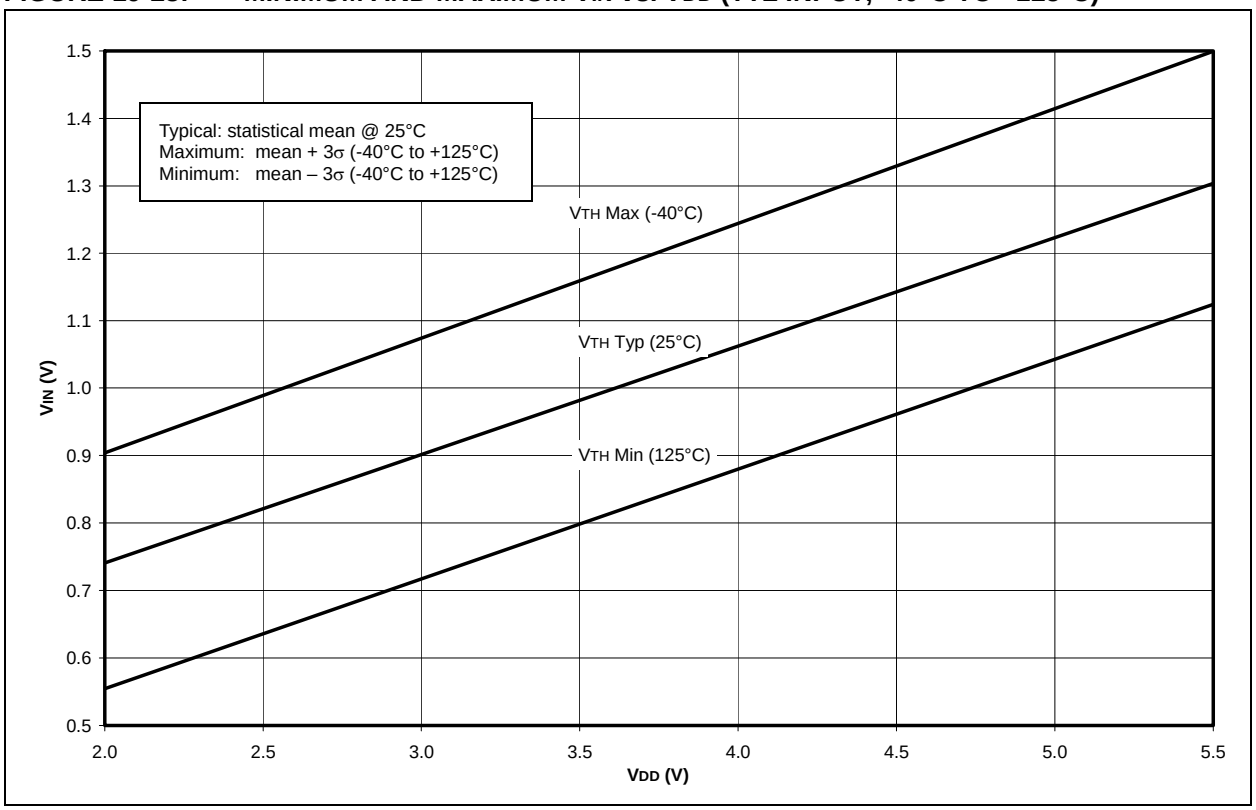
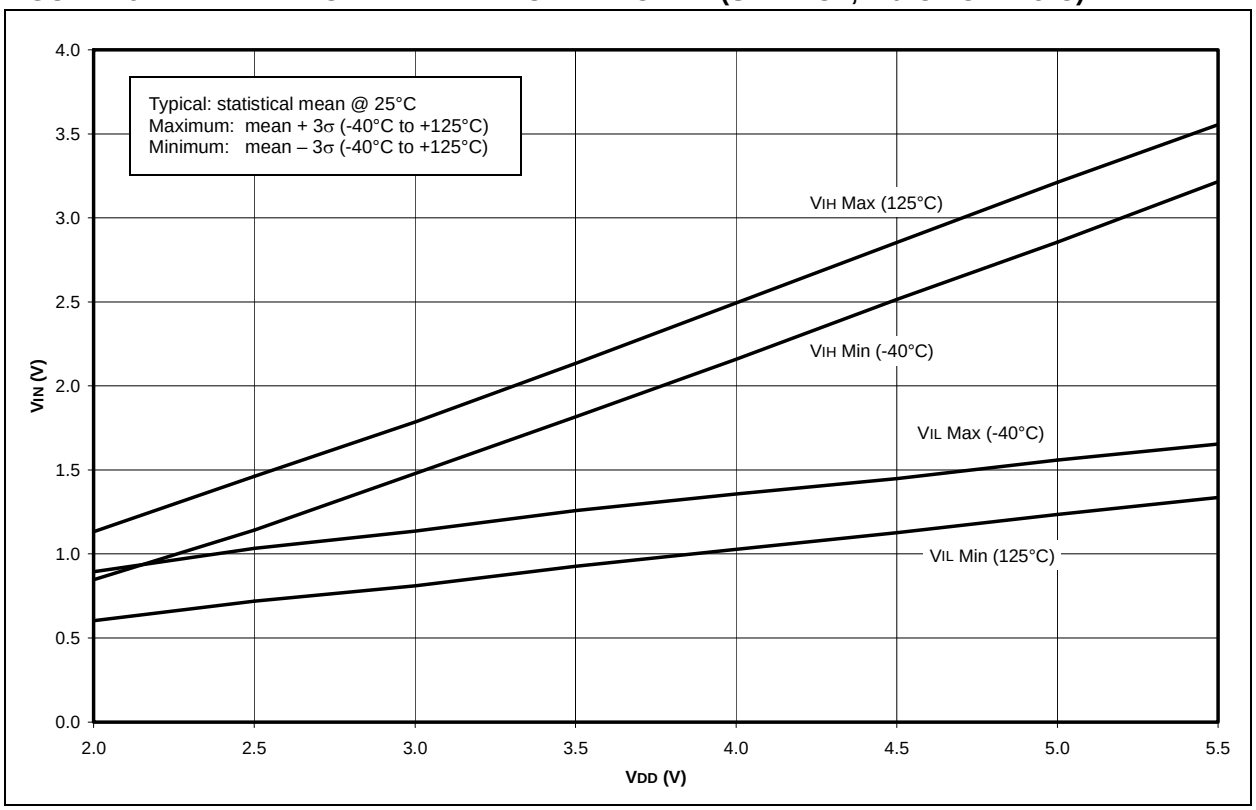
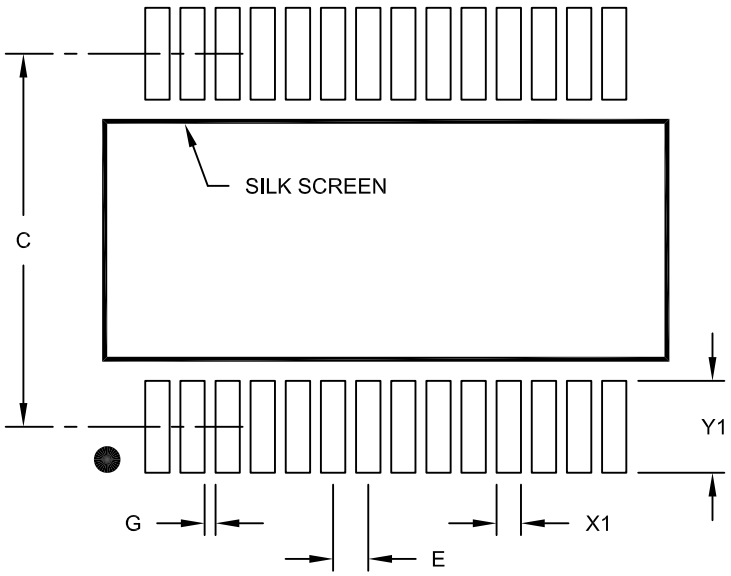



**FIGURE 19-24: MINIMUM AND MAXIMUM VIN vs. VDD (ST INPUT, -40°C TO +125°C)**



28-Lead Plastic Shrink Small Outline (SS) - 5.30 mm Body [SSOP]

**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 | | |
| Contact Pad Spacing | C | | 7.20 | |
| Contact Pad Width (X28) | X1 | | | 0.45 |
| Contact Pad Length (X28) | Y1 | | | 1.75 |
| 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. C04-2073A