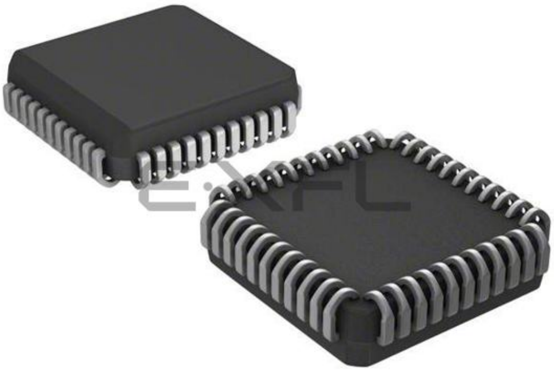
# Microchip Technology - PIC16LF877T-04I/L Datasheet



**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

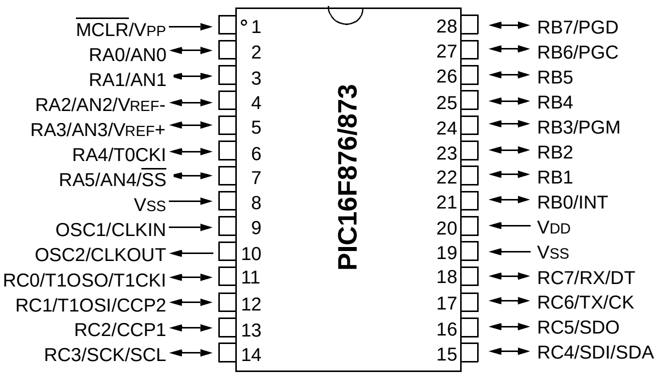
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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"

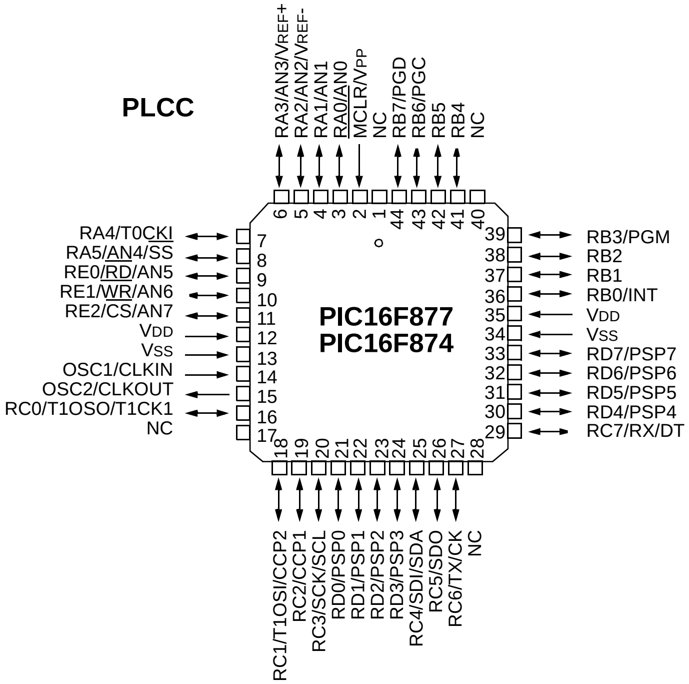
| **Details** | |
|---|---|
| Product Status | Obsolete |
| Core Processor | PIC |
| Core Size | 8-Bit |
| Speed | 4MHz |
| Connectivity | I²C, SPI, UART/USART |
| Peripherals | Brown-out Detect/Reset, POR, PWM, WDT |
| Number of I/O | 33 |
| Program Memory Size | 14KB (8K x 14) |
| Program Memory Type | FLASH |
| EEPROM Size | 256 x 8 |
| RAM Size | 368 x 8 |
| Voltage - Supply (Vcc/Vdd) | 2V ~ 5.5V |
| Data Converters | A/D 8x10b |
| Oscillator Type | External |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 44-LCC (J-Lead) |
| Supplier Device Package | 44-PLCC (16.59x16.59) |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16lf877t-04i-l |

## Pin Diagrams

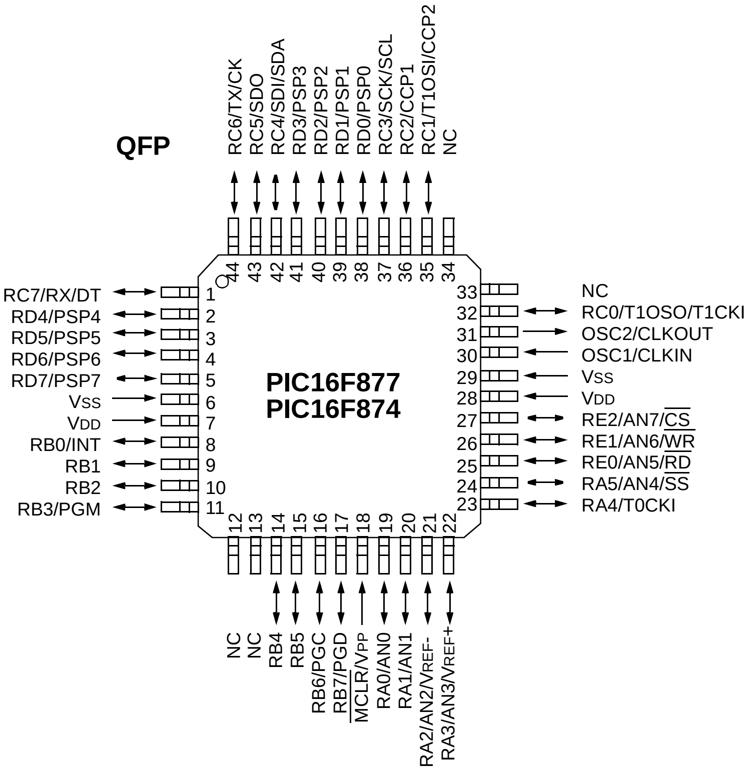
**PDIP, SOIC**



**PLCC**



**QFP**



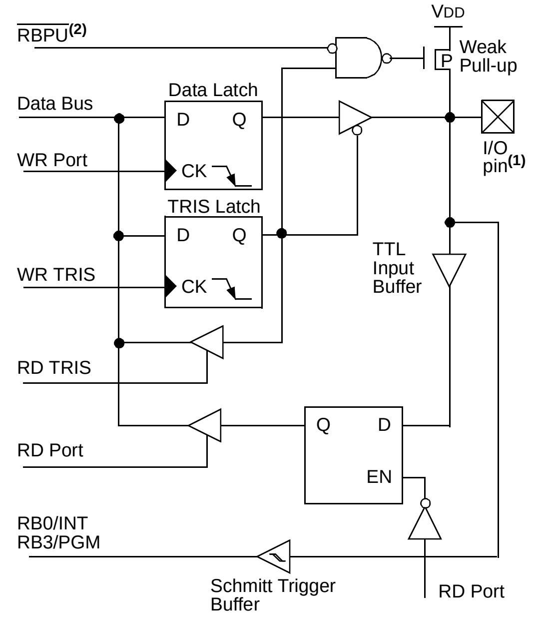
## 3.2 PORTB and the TRISB Register

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

Three pins of PORTB are multiplexed with the Low Voltage Programming function: RB3/PGM, RB6/PGC and RB7/PGD. The alternate functions of these pins are described in the Special Features Section.

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

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



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

**2:** To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the $\overline{\text{RBPU}}$ bit (OPTION_REG<7>).

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

This interrupt can wake the device from SLEEP. The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

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

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

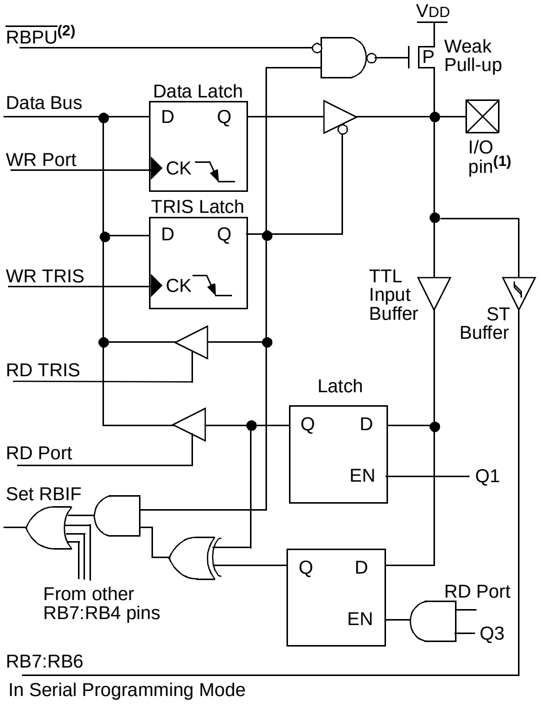
The interrupt-on-change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt-on-change feature. Polling of PORTB is not recommended while using the interrupt-on-change feature.

This interrupt-on-mismatch feature, together with software configureable pull-ups on these four pins, allow easy interface to a keypad and make it possible for wake-up on key depression. Refer to the Embedded Control Handbook, *“Implementing Wake-up on Key Strokes”* (AN552).

RB0/INT is an external interrupt input pin and is configured using the INTEDG bit (OPTION_REG<6>).

RB0/INT is discussed in detail in Section 12.10.1.

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



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

**2:** To enable weak pull-ups, set the appropriate TRIS bit(s) and clear the $\overline{\text{RBPU}}$ bit (OPTION_REG<7>).

## 7.1 Timer2 Prescaler and Postscaler

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

- a write to the TMR2 register
- a write to the T2CON register
- any device RESET (POR, $\overline{\text{MCLR}}$ Reset, WDT Reset, or BOR)

TMR2 is not cleared when T2CON is written.

## 7.2 Output of TMR2

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

### TABLE 7-1: REGISTERS ASSOCIATED WITH TIMER2 AS A TIMER/COUNTER

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on: POR, BOR | Value on all other RESETS |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0Bh,8Bh, 10Bh,18Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | `0000 000x` | `0000 000u` |
| 0Ch | PIR1 | PSPIF[1] | ADIF | RCIF | TXIF | SSPIF | CCP1IF | TMR2IF | TMR1IF | `0000 0000` | `0000 0000` |
| 8Ch | PIE1 | PSPIE[1] | ADIE | RCIE | TXIE | SSPIE | CCP1IE | TMR2IE | TMR1IE | `0000 0000` | `0000 0000` |
| 11h | TMR2 | Timer2 Module's Register | | | | | | | | `0000 0000` | `0000 0000` |
| 12h | T2CON | — | TOUTPS3 | TOUTPS2 | TOUTPS1 | TOUTPS0 | TMR2ON | T2CKPS1 | T2CKPS0 | `-000 0000` | `-000 0000` |
| 92h | PR2 | Timer2 Period Register | | | | | | | | `1111 1111` | `1111 1111` |

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

**Note 1:** Bits PSPIE and PSPIF are reserved on the PIC16F873/876; always maintain these bits clear.

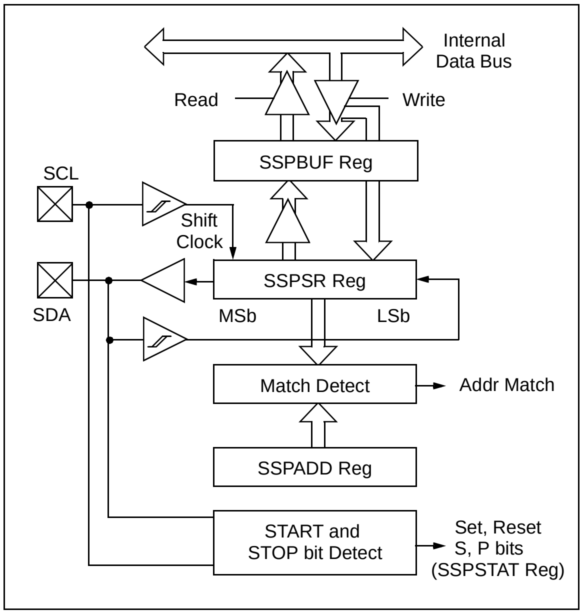
## 9.2 MSSP I$^2$C Operation

The MSSP module in I$^2$C mode, fully implements all master and slave functions (including general call support) and provides interrupts on START and STOP bits in hardware, to determine a free bus (multi-master function). The MSSP module implements the standard mode specifications, as well as 7-bit and 10-bit addressing.

Refer to Application Note AN578, *"Use of the SSP Module in the I$^2$C Multi-Master Environment."*

A "glitch" filter is on the SCL and SDA pins when the pin is an input. This filter operates in both the 100 kHz and 400 kHz modes. In the 100 kHz mode, when these pins are an output, there is a slew rate control of the pin that is independent of device frequency.

**FIGURE 9-5: I$^2$C SLAVE MODE BLOCK DIAGRAM**



Two pins are used for data transfer. These are the SCL pin, which is the clock, and the SDA pin, which is the data. The SDA and SCL pins are automatically configured when the I$^2$C mode is enabled. The SSP module functions are enabled by setting SSP Enable bit SSPEN (SSPCON<5>).

The MSSP module has six registers for I$^2$C operation. They are the:

- SSP Control Register (SSPCON)
- SSP Control Register2 (SSPCON2)
- SSP Status Register (SSPSTAT)
- Serial Receive/Transmit Buffer (SSPBUF)
- SSP Shift Register (SSPSR) - Not directly accessible
- SSP Address Register (SSPADD)

The SSPCON register allows control of the I$^2$C operation. Four mode selection bits (SSPCON<3:0>) allow one of the following I$^2$C modes to be selected:

- I$^2$C Slave mode (7-bit address)
- I$^2$C Slave mode (10-bit address)
- I$^2$C Master mode, clock = OSC/4 (SSPADD +1)
- I$^2$C firmware modes (provided for compatibility to other mid-range products)

Before selecting any I$^2$C mode, the SCL and SDA pins must be programmed to inputs by setting the appropriate TRIS bits. Selecting an I$^2$C mode by setting the SSPEN bit, enables the SCL and SDA pins to be used as the clock and data lines in I$^2$C mode. Pull-up resistors must be provided externally to the SCL and SDA pins for the proper operation of the I$^2$C module.

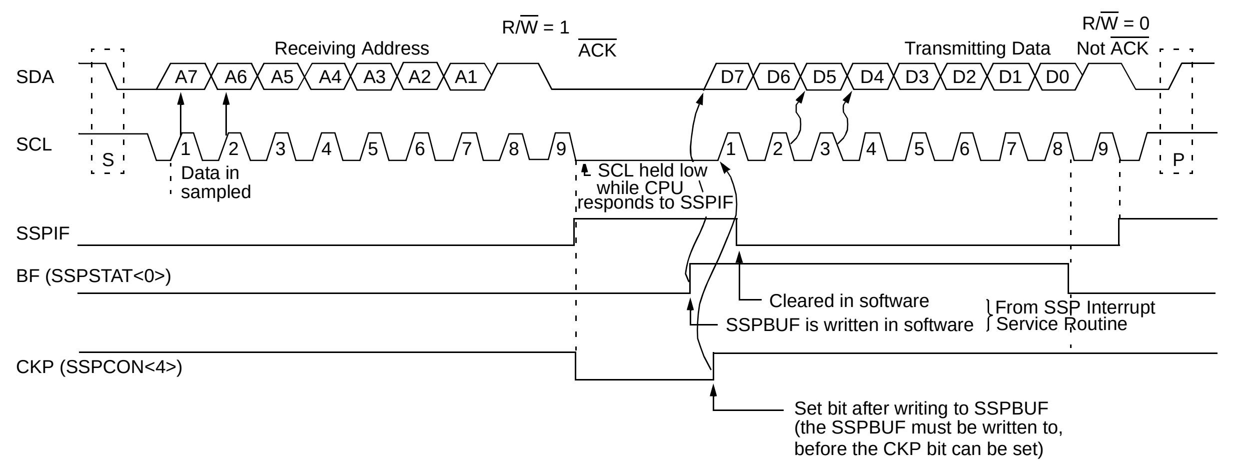
The CKE bit (SSPSTAT<6:7>) sets the levels of the SDA and SCL pins in either Master or Slave mode. When CKE = 1, the levels will conform to the SMBus specification. When CKE = 0, the levels will conform to the I$^2$C specification.

The SSPSTAT register gives the status of the data transfer. This information includes detection of a START (S) or STOP (P) bit, specifies if the received byte was data or address, if the next byte is the completion of 10-bit address, and if this will be a read or write data transfer.

SSPBUF is the register to which the transfer data is written to, or read from. The SSPSR register shifts the data in or out of the device. In receive operations, the SSPBUF and SSPSR create a doubled buffered receiver. This allows reception of the next byte to begin before reading the last byte of received data. When the complete byte is received, it is transferred to the SSPBUF register and flag bit SSPIF is set. If another complete byte is received before the SSPBUF register is read, a receiver overflow has occurred and bit SSPOV (SSPCON<6>) is set and the byte in the SSPSR is lost.

The SSPADD register holds the slave address. In 10-bit mode, the user needs to write the high byte of the address (`1111 0 A9 A8 0`). Following the high byte address match, the low byte of the address needs to be loaded (A7:A0).

**FIGURE 9-7: I$^2$C WAVEFORMS FOR TRANSMISSION (7-BIT ADDRESS)**



### 9.2.2 GENERAL CALL ADDRESS SUPPORT

The addressing procedure for the I$^2$C bus is such that the first byte after the START condition usually determines which device will be the slave addressed by the master. The exception is the general call address, which can address all devices. When this address is used, all devices should, in theory, respond with an acknowledge.

The general call address is one of eight addresses reserved for specific purposes by the I$^2$C protocol. It consists of all 0's with R/$\overline{W}$ = 0.

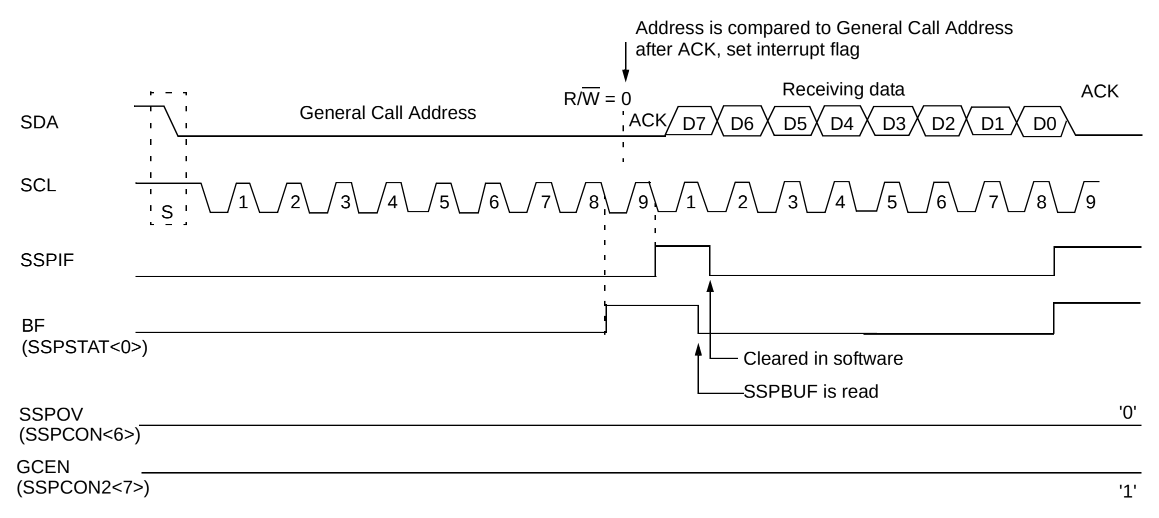
The general call address is recognized when the General Call Enable bit (GCEN) is enabled (SSPCON2<7> is set). Following a START bit detect, 8 bits are shifted into SSPSR and the address is compared against SSPADD. It is also compared to the general call address and fixed in hardware.

If the general call address matches, the SSPSR is transferred to the SSPBUF, the BF flag is set (eighth bit), and on the falling edge of the ninth bit ($\overline{ACK}$ bit), the SSPIF flag is set.

When the interrupt is serviced, the source for the interrupt can be checked by reading the contents of the SSPBUF to determine if the address was device specific, or a general call address.

In 10-bit mode, the SSPADD is required to be updated for the second half of the address to match, and the UA bit is set (SSPSTAT<1>). If the general call address is sampled when GCEN is set, while the slave is configured in 10-bit address mode, then the second half of the address is not necessary, the UA bit will not be set, and the slave will begin receiving data after the Acknowledge (Figure 9-8).

**FIGURE 9-8: SLAVE MODE GENERAL CALL ADDRESS SEQUENCE (7 OR 10-BIT MODE)**



### 10.4.2 USART SYNCHRONOUS SLAVE RECEPTION

The operation of the Synchronous Master and Slave modes is identical, except in the case of the SLEEP mode. Bit SREN is a “don't care” in Slave mode.

If receive is enabled by setting bit CREN prior to the `SLEEP` instruction, then a word may be received during SLEEP. On completely receiving the word, the RSR register will transfer the data to the RCREG register and if enable bit RCIE bit is set, the interrupt generated will wake the chip from SLEEP. If the global interrupt is enabled, the program will branch to the interrupt vector (0004h).

When setting up a Synchronous Slave Reception, follow these steps:

1. Enable the synchronous master serial port by setting bits SYNC and SPEN and clearing bit CSRC.
2. If interrupts are desired, set enable bit RCIE.
3. If 9-bit reception is desired, set bit RX9.
4. To enable reception, set enable bit CREN.
5. Flag bit RCIF will be set when reception is complete and an interrupt will be generated, if enable bit RCIE was set.
6. Read the RCSTA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
7. Read the 8-bit received data by reading the RCREG register.
8. If any error occurred, clear the error by clearing bit CREN.
9. If using interrupts, ensure that GIE and PEIE (bits 7 and 6) of the INTCON register are set.

**TABLE 10-11: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE RECEPTION**

| 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 | T0IE | INTE | RBIE | T0IF | INTF | R0IF | `0000 000x` | `0000 000u` |
| 0Ch | PIR1 | PSPIF[1] | ADIF | RCIF | TXIF | SSPIF | CCP1IF | TMR2IF | TMR1IF | `0000 0000` | `0000 0000` |
| 18h | RCSTA | SPEN | RX9 | SREN | CREN | ADDEN | FERR | OERR | RX9D | `0000 000x` | `0000 000x` |
| 1Ah | RCREG | USART Receive Register | | | | | | | | `0000 0000` | `0000 0000` |
| 8Ch | PIE1 | PSPIE[1] | ADIE | RCIE | TXIE | SSPIE | CCP1IE | TMR2IE | TMR1IE | `0000 0000` | `0000 0000` |
| 98h | TXSTA | CSRC | TX9 | TXEN | SYNC | — | BRGH | TRMT | TX9D | `0000 -010` | `0000 -010` |
| 99h | SPBRG | Baud Rate Generator Register | | | | | | | | `0000 0000` | `0000 0000` |

Legend: `x` = unknown, `-` = unimplemented, read as '0'. Shaded cells are not used for synchronous slave reception.

**Note 1:** Bits PSPIE and PSPIF are reserved on PIC16F873/876 devices, always maintain these bits clear.

## 12.4 Power-On Reset (POR)

A Power-on Reset pulse is generated on-chip when VDD rise is detected (in the range of 1.2V - 1.7V). To take advantage of the POR, tie the $\overline{\text{MCLR}}$ pin directly (or through a resistor) to VDD. This will eliminate external RC components usually needed to create a Power-on Reset. A maximum rise time for VDD is specified. See Electrical Specifications for details.

When the device starts normal operation (exits the RESET condition), device operating parameters (voltage, frequency, temperature,...) must be met to ensure operation. If these conditions are not met, the device must be held in RESET until the operating conditions are met. Brown-out Reset may be used to meet the start-up conditions. For additional information, refer to Application Note, AN007, “Power-up Trouble Shooting”, (DS00007).

## 12.5 Power-up Timer (PWRT)

The Power-up Timer provides a fixed 72 ms nominal time-out on power-up only from the POR. The Power-up Timer operates on an internal RC oscillator. The chip is kept in RESET as long as the PWRT is active. The PWRT’s time delay allows VDD to rise to an acceptable level. A configuration bit is provided to enable/disable the PWRT.

The power-up time delay will vary from chip to chip due to VDD, temperature and process variation. See DC parameters for details (TPWRT, parameter #33).

## 12.6 Oscillator Start-up Timer (OST)

The Oscillator Start-up Timer (OST) provides a delay of 1024 oscillator cycles (from OSC1 input) after the PWRT delay is over (if PWRT is enabled). This helps to ensure that the crystal oscillator or resonator has started and stabilized.

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

## 12.7 Brown-out Reset (BOR)

The configuration bit, BODEN, can enable or disable the Brown-out Reset circuit. If VDD falls below VBOR (parameter D005, about 4V) for longer than TBOR (parameter #35, about 100µS), the brown-out situation will reset the device. If VDD falls below VBOR for less than TBOR, a RESET may not occur.

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

## 12.8 Time-out Sequence

On power-up, the time-out sequence is as follows: The PWRT delay starts (if enabled) when a POR Reset occurs. Then OST starts counting 1024 oscillator cycles when PWRT ends (LP, XT, HS). When the OST ends, the device comes out of RESET.

If $\overline{\text{MCLR}}$ is kept low long enough, the time-outs will expire. Bringing $\overline{\text{MCLR}}$ high will begin execution immediately. This is useful for testing purposes or to synchronize more than one PIC16F87X device operating in parallel.

Table 12-5 shows the RESET conditions for the STATUS, PCON and PC registers, while Table 12-6 shows the RESET conditions for all the registers.

## 12.9 Power Control/Status Register (PCON)

The Power Control/Status Register, PCON, has up to two bits depending upon the device.

Bit0 is Brown-out Reset Status bit, $\overline{\text{BOR}}$. Bit $\overline{\text{BOR}}$ is unknown on a Power-on Reset. It must then be set by the user and checked on subsequent RESETS to see if bit $\overline{\text{BOR}}$ cleared, indicating a BOR occurred. When the Brown-out Reset is disabled, the state of the $\overline{\text{BOR}}$ bit is unpredictable and is, therefore, not valid at any time.

Bit1 is $\overline{\text{POR}}$ (Power-on Reset Status bit). It is cleared on a Power-on Reset and unaffected otherwise. The user must set this bit following a Power-on Reset.

**TABLE 12-3: TIME-OUT IN VARIOUS SITUATIONS**

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

**FIGURE 12-6: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ NOT TIED TO VDD): CASE 1**

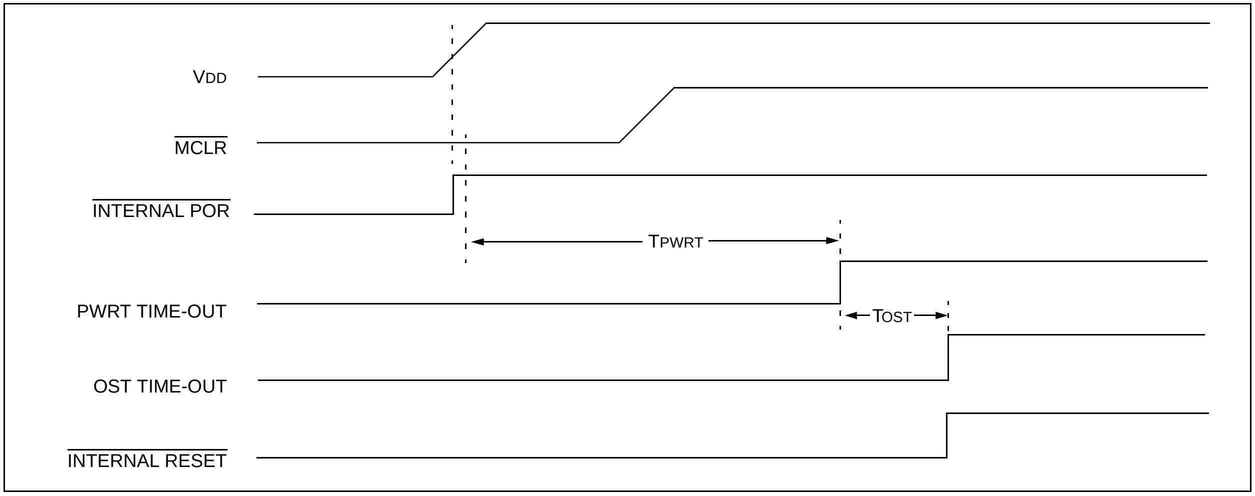


**FIGURE 12-7: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ NOT TIED TO VDD): CASE 2**

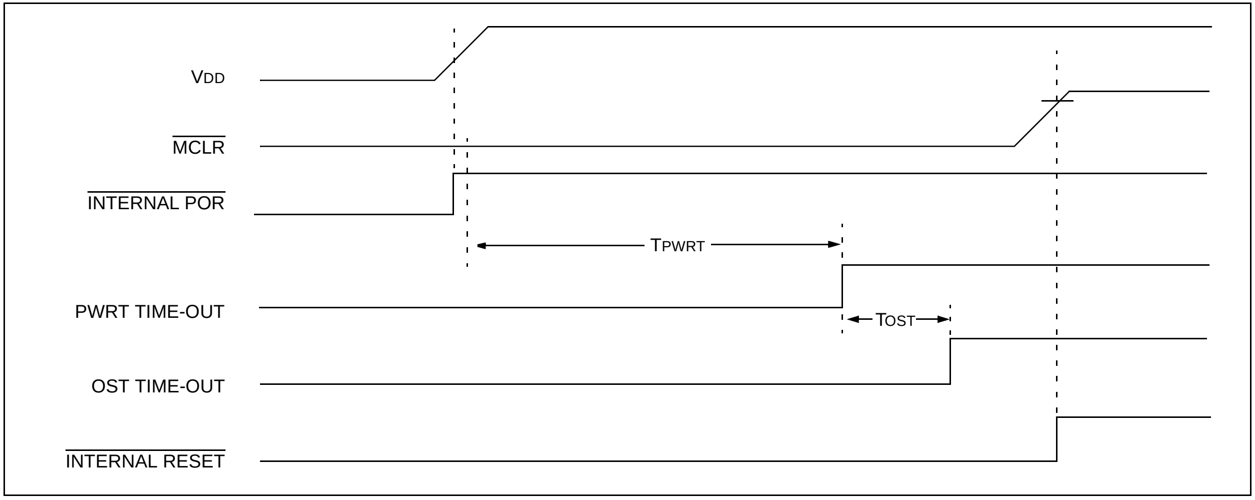


**FIGURE 12-8: SLOW RISE TIME ($\overline{\text{MCLR}}$ TIED TO VDD)**

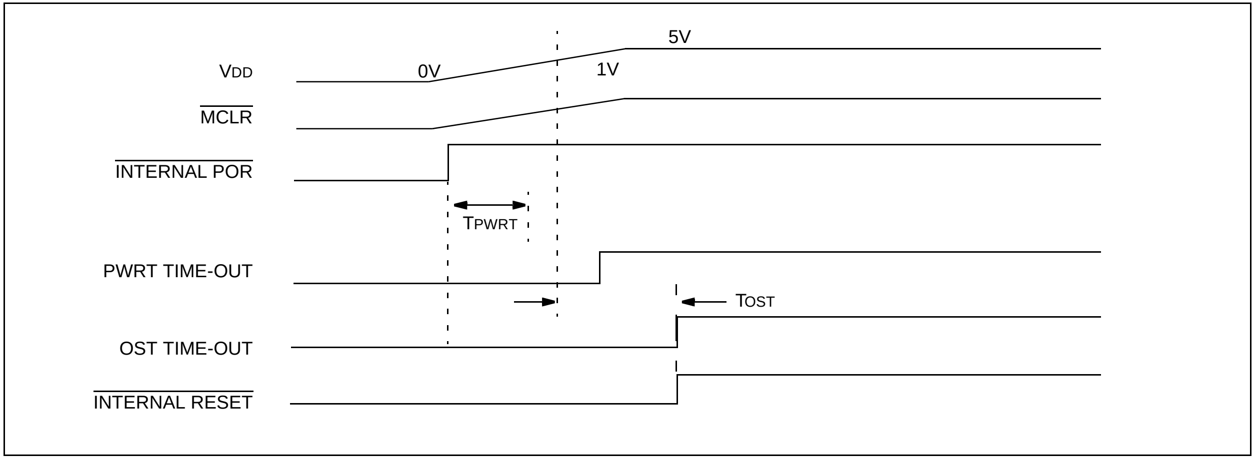


**TABLE 14-1: DEVELOPMENT TOOLS FROM MICROCHIP**

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

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

** Contact Microchip Technology Inc. for availability date.

† Development tool is available on select devices.

## 15.0 ELECTRICAL CHARACTERISTICS

**Absolute Maximum Ratings †**

| | |
|---|---|
| Ambient temperature under bias | -55 to +125°C |
| Storage temperature | -65°C to +150°C |
| Voltage on any pin with respect to VSS (except VDD, $\overline{\text{MCLR}}$. and RA4) | -0.3 V to (VDD + 0.3 V) |
| Voltage on VDD with respect to VSS | -0.3 to +7.5 V |
| Voltage on $\overline{\text{MCLR}}$ with respect to VSS **(Note 2)** | 0 to +14 V |
| Voltage on RA4 with respect to Vss | 0 to +8.5 V |
| Total power dissipation **(Note 1)** | 1.0 W |
| Maximum current out of VSS pin | 300 mA |
| Maximum current into VDD pin | 250 mA |
| Input clamp current, IIK (VI < 0 or VI > VDD) | ± 20 mA |
| Output clamp current, IOK (VO < 0 or VO > VDD) | ± 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, PORTB, and PORTE (combined) **(Note 3)** | 200 mA |
| Maximum current sourced by PORTA, PORTB, and PORTE (combined) **(Note 3)** | 200 mA |
| Maximum current sunk by PORTC and PORTD (combined) **(Note 3)** | 200 mA |
| Maximum current sourced by PORTC and PORTD (combined) **(Note 3)** | 200 mA |

**Note 1:** Power dissipation is calculated as follows: Pdis = VDD x {IDD - ∑ IOH} + ∑ {(VDD - VOH) x IOH} + ∑(VOl x IOL)

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

**3:** PORTD and PORTE are not implemented on PIC16F873/876 devices.

† 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.

## 15.1 DC Characteristics: PIC16F873/874/876/877-04 (Commercial, Industrial) PIC16F873/874/876/877-20 (Commercial, Industrial) PIC16LF873/874/876/877-04 (Commercial, Industrial) (Continued)

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| **PIC16LF873/874/876/877-04**<br>(Commercial, Industrial) | | | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial<br>0°C ≤ TA ≤ +70°C for commercial | | | | |
| **PIC16F873/874/876/877-04**<br>**PIC16F873/874/876/877-20**<br>(Commercial, Industrial) | | | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature -40°C ≤ TA ≤ +85°C for industrial<br>0°C ≤ TA ≤ +70°C for commercial | | | | |
| **Param No.** | **Symbol** | **Characteristic/ Device** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| | IDD | **Supply Current(2,5)** | | | | | |
| D010 | | 16LF87X | — | 0.6 | 2.0 | mA | XT, RC osc configuration<br>FOSC = 4 MHz, VDD = 3.0V |
| D010 | | 16F87X | — | 1.6 | 4 | mA | RC osc configurations<br>FOSC = 4 MHz, VDD = 5.5V |
| D010A | | 16LF87X | — | 20 | 35 | μA | LP osc configuration<br>FOSC = 32 kHz, VDD = 3.0V,<br>WDT disabled |
| D013 | | 16F87X | — | 7 | 15 | mA | HS osc configuration,<br>FOSC = 20 MHz, VDD = 5.5V |
| D015 | ΔIBOR | **Brown-out Reset Current(6)** | — | 85 | 200 | μA | BOR enabled, VDD = 5.0V |

Legend: Rows with standard voltage device data only are shaded for improved readability.

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

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

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

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

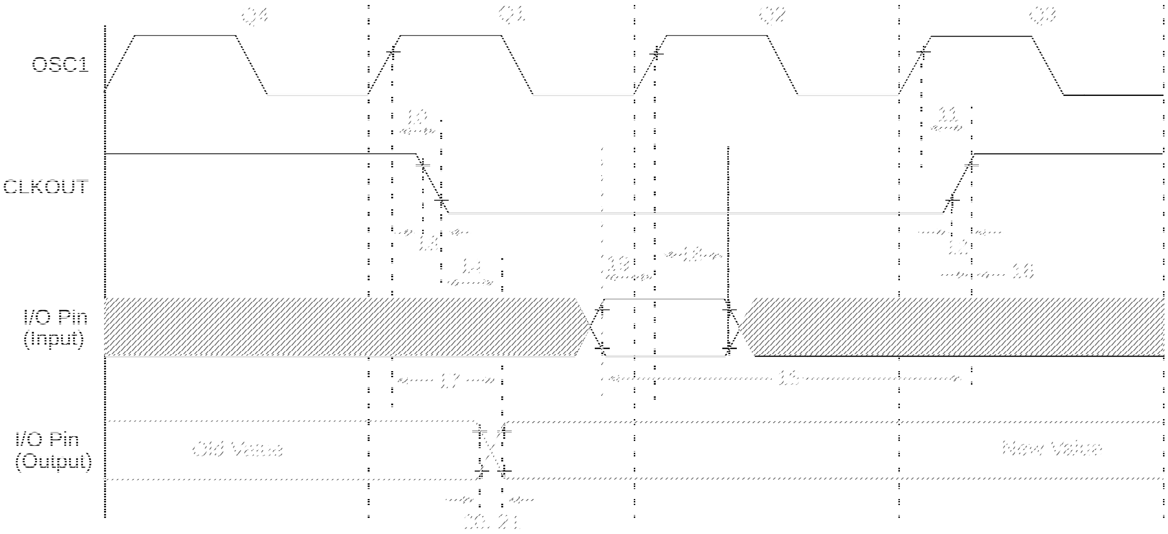
**4:** For RC osc configuration, current through REXT is not included. The current through the resistor can be estimated by the formula Ir = VDD/2REXT (mA) with REXT in kOhm.

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

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

**7:** When BOR is enabled, the device will operate correctly until the VBOR voltage trip point is reached.

## FIGURE 15-7: CLKOUT AND I/O TIMING



**Note:** Refer to Figure 15-5 for load conditions.

## TABLE 15-2: CLKOUT AND I/O TIMING REQUIREMENTS

| Param No. | Symbol | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 10* | TosH2ckL | OSC1↑ to CLKOUT↓ | | — | 75 | 200 | ns | **(Note 1)** |
| 11* | TosH2ckH | OSC1↑ to CLKOUT↑ | | — | 75 | 200 | ns | **(Note 1)** |
| 12* | TckR | CLKOUT rise time | | — | 35 | 100 | ns | **(Note 1)** |
| 13* | TckF | CLKOUT fall time | | — | 35 | 100 | ns | **(Note 1)** |
| 14* | TckL2ioV | CLKOUT ↓ to Port out valid | | — | — | 0.5TCY + 20 | ns | **(Note 1)** |
| 15* | TioV2ckH | Port in valid before CLKOUT ↑ | | TOSC + 200 | — | — | ns | **(Note 1)** |
| 16* | TckH2ioI | Port in hold after CLKOUT ↑ | | 0 | — | — | ns | **(Note 1)** |
| 17* | TosH2ioV | OSC1↑ (Q1 cycle) to Port out valid | | — | 100 | 255 | ns | |
| 18* | TosH2ioI | OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time) | Standard (**F**) | 100 | — | — | ns | |
| | | | Extended (**LF**) | 200 | — | — | ns | |
| 19* | TioV2osH | Port input valid to OSC1↑ (I/O in setup time) | | 0 | — | — | ns | |
| 20* | TioR | Port output rise time | Standard (**F**) | — | 10 | 40 | ns | |
| | | | Extended (**LF**) | — | — | 145 | ns | |
| 21* | TioF | Port output fall time | Standard (**F**) | — | 10 | 40 | ns | |
| | | | Extended (**LF**) | — | — | 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 CLKOUT output is 4 x TOSC.

**FIGURE 15-15: SPI SLAVE MODE TIMING (CKE = 0)**

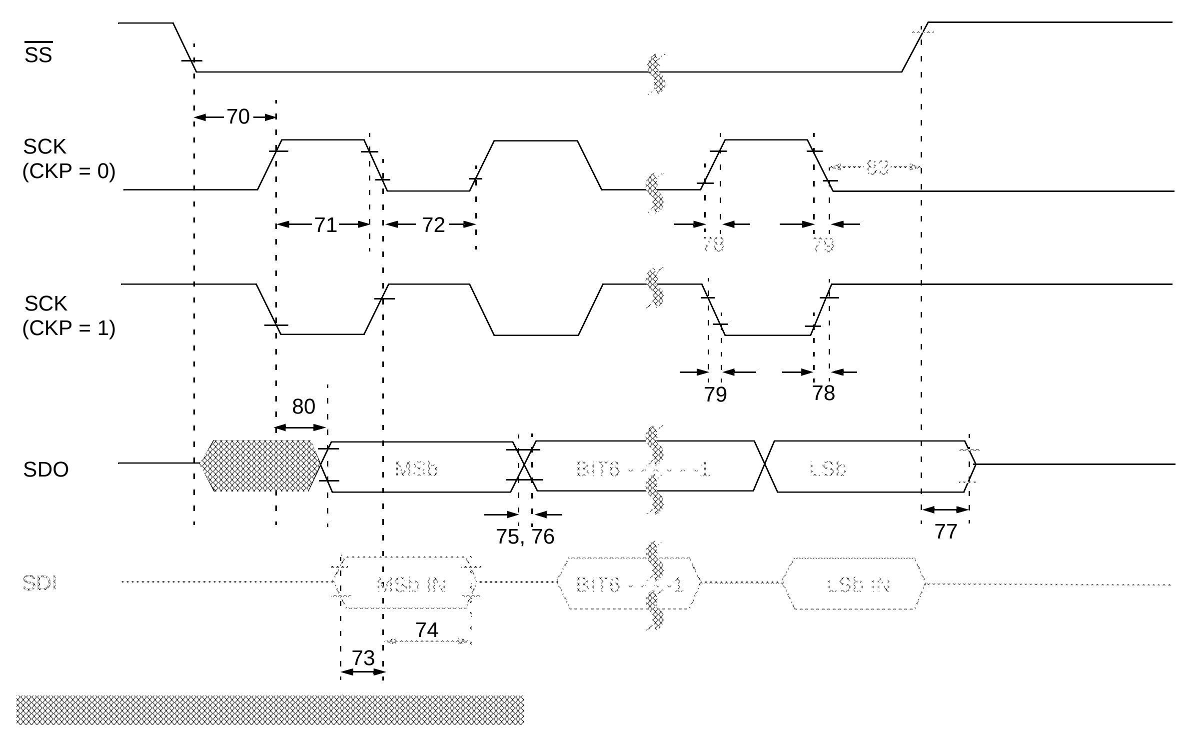
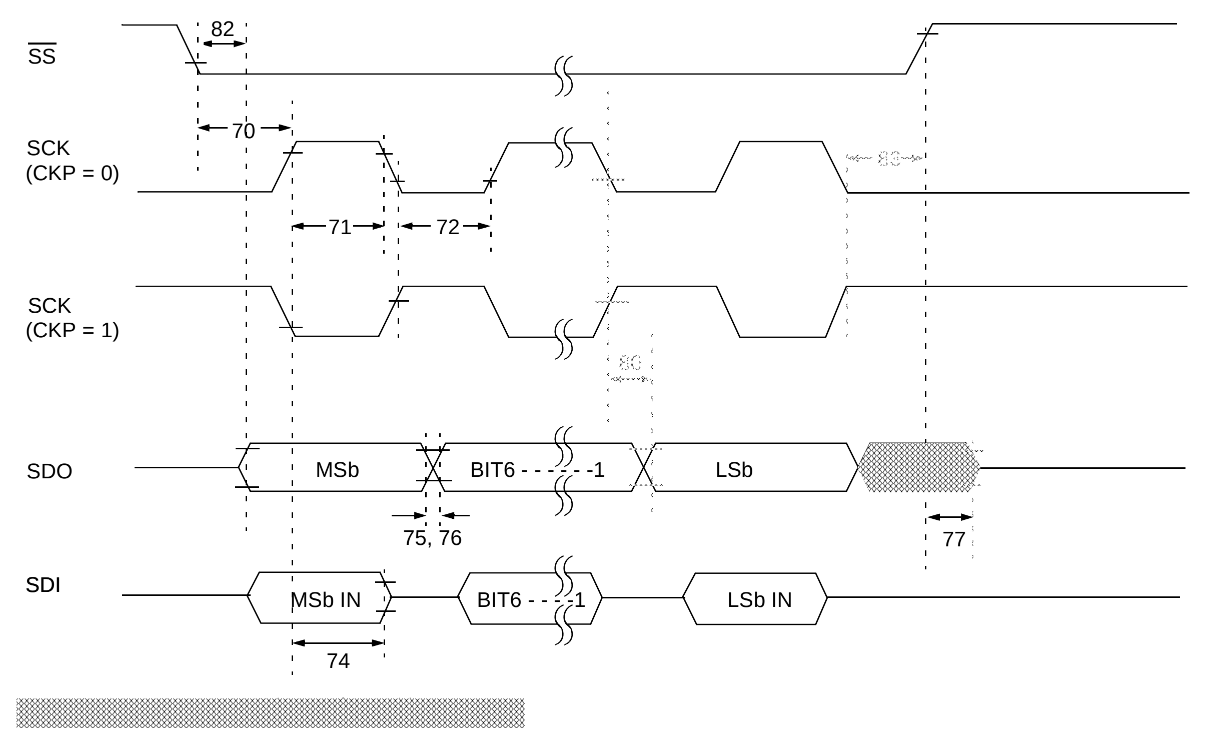


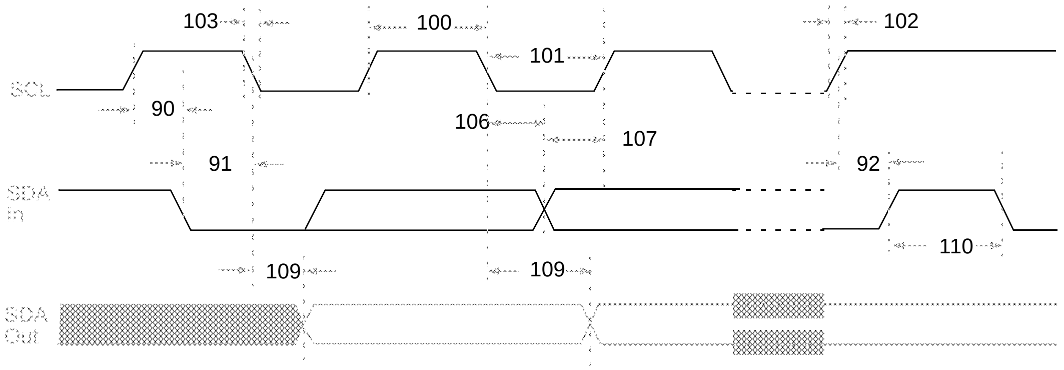
**FIGURE 15-16: SPI SLAVE MODE TIMING (CKE = 1)**



### TABLE 15-8: I$^2$C BUS START/STOP BITS REQUIREMENTS

| Parameter No. | Symbol | Characteristic | | Min | Typ | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 90 | Tsu:sta | START condition | 100 kHz mode | 4700 | — | — | ns | Only relevant for Repeated START condition |
| | | Setup time | 400 kHz mode | 600 | — | — | | |
| 91 | Thd:sta | START condition | 100 kHz mode | 4000 | — | — | ns | After this period, the first clock pulse is generated |
| | | Hold time | 400 kHz mode | 600 | — | — | | |
| 92 | Tsu:sto | STOP condition | 100 kHz mode | 4700 | — | — | ns | |
| | | Setup time | 400 kHz mode | 600 | — | — | | |
| 93 | Thd:sto | STOP condition | 100 kHz mode | 4000 | — | — | ns | |
| | | Hold time | 400 kHz mode | 600 | — | — | | |

### FIGURE 15-18: I$^2$C BUS DATA TIMING



**Note:** Refer to Figure 15-5 for load conditions.

**FIGURE 16-13: TYPICAL AND MAXIMUM $\Delta I_{WDT}$ vs. $V_{DD}$ OVER TEMPERATURE**

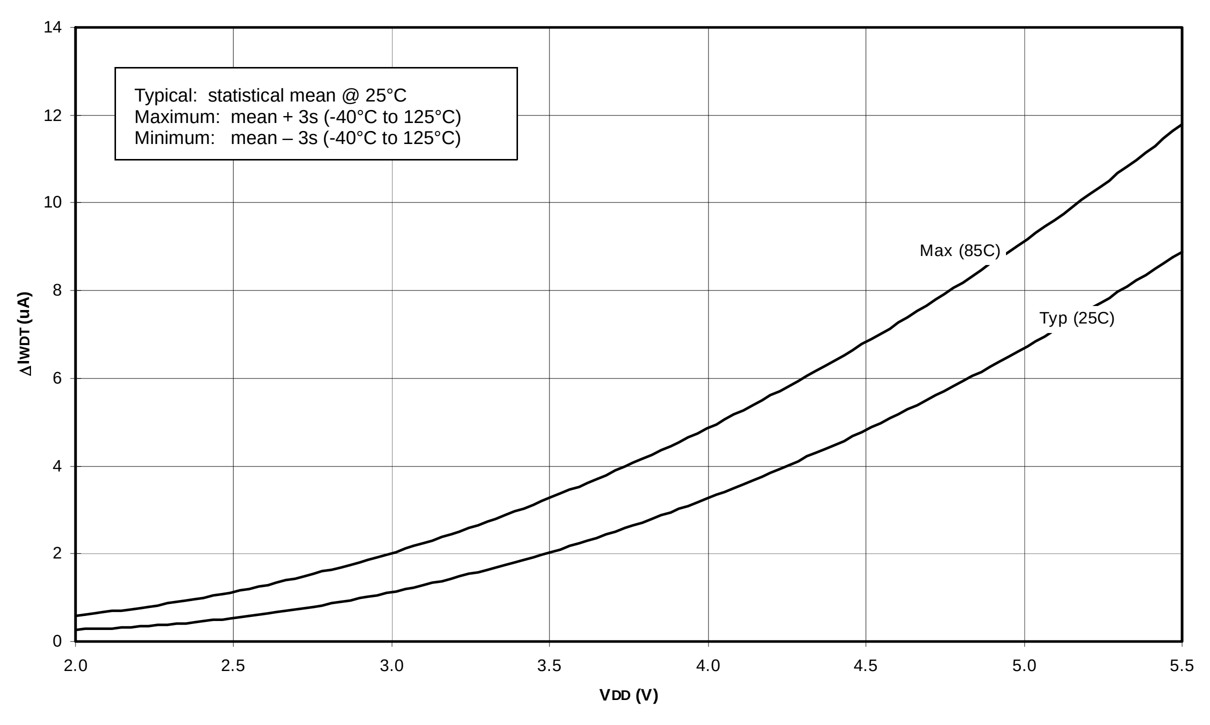
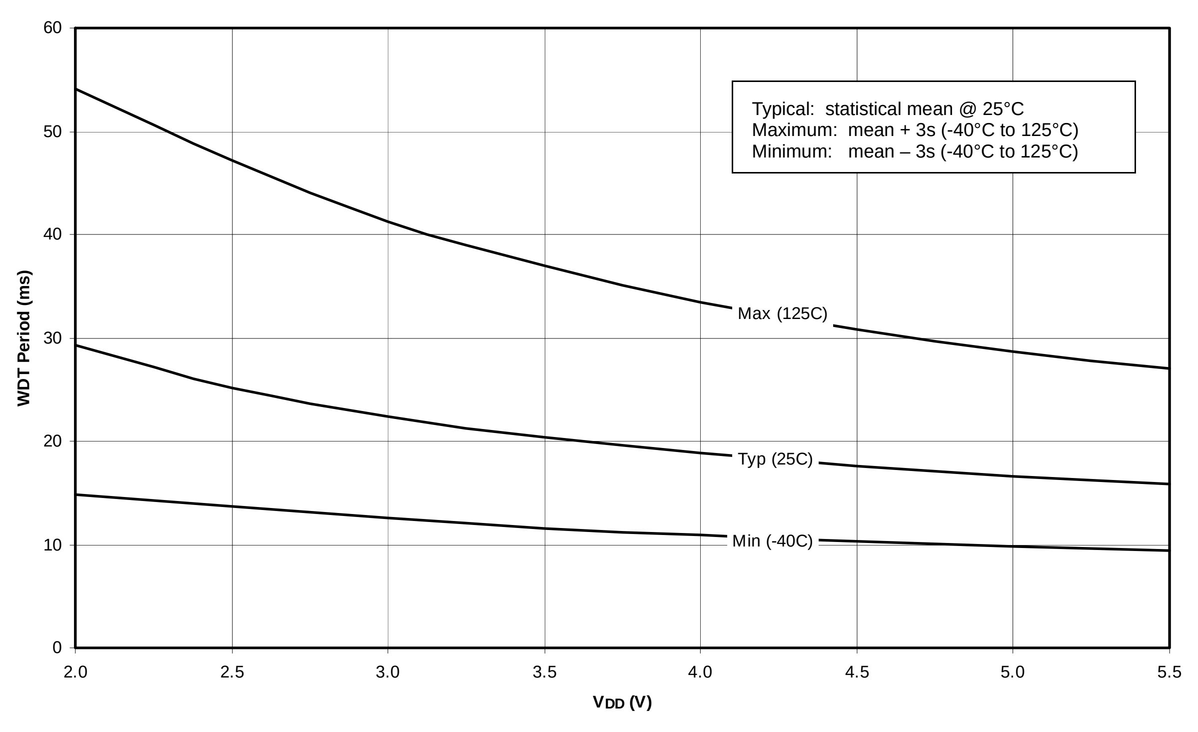


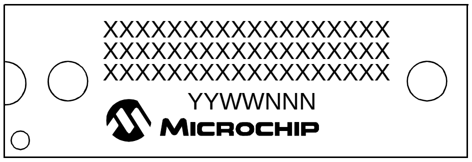
**FIGURE 16-14: TYPICAL, MINIMUM AND MAXIMUM WDT PERIOD vs. $V_{DD}$ (-40°C TO 125°C)**



NOTES:

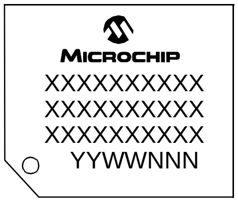
## Package Marking Information (Cont'd)

40-Lead PDIP

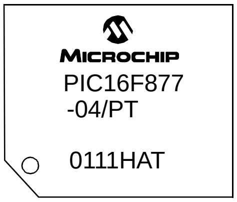


Example

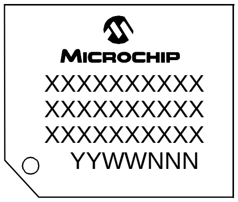
44-Lead TQFP



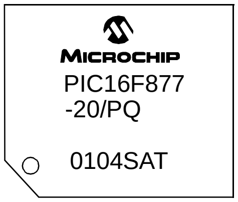
Example



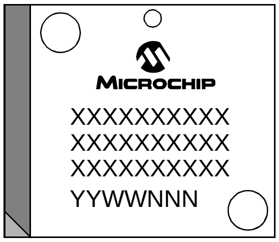
44-Lead MQFP



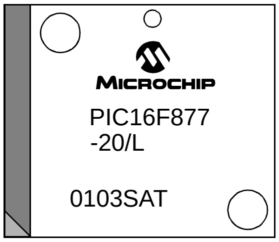
Example



44-Lead PLCC



Example



## READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

Please list the following information, and use this outline to provide us with your comments about this Data Sheet.

To: Technical Publications Manager　　Total Pages Sent

RE: Reader Response

From: Name ____________________

Company ____________________

Address ____________________

City / State / ZIP / Country ____________________

Telephone: (_______) _________ - _________　　FAX: (______) _________ - _________

Application (optional):

Would you like a reply?___Y ___N

Device: **PIC16F87X**　　Literature Number: **DS30292D**

Questions:

1. What are the best features of this document?

____________________

____________________

2. How does this document meet your hardware and software development needs?

____________________

____________________

3. Do you find the organization of this data sheet easy to follow? If not, why?

____________________

____________________

4. What additions to the data sheet do you think would enhance the structure and subject?

____________________

____________________

5. What deletions from the data sheet could be made without affecting the overall usefulness?

____________________

____________________

6. Is there any incorrect or misleading information (what and where)?

____________________

____________________

7. How would you improve this document?

____________________

____________________

8. How would you improve our software, systems, and silicon products?

____________________

____________________

**NOTES:**