



Welcome to [E-XFL.COM](https://www.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	32MHz
Connectivity	I <sup>2</sup> C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	18
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 12x10b; D/A 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SOIC
Purchase URL	<a href="https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1619t-i-so">https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1619t-i-so</a>

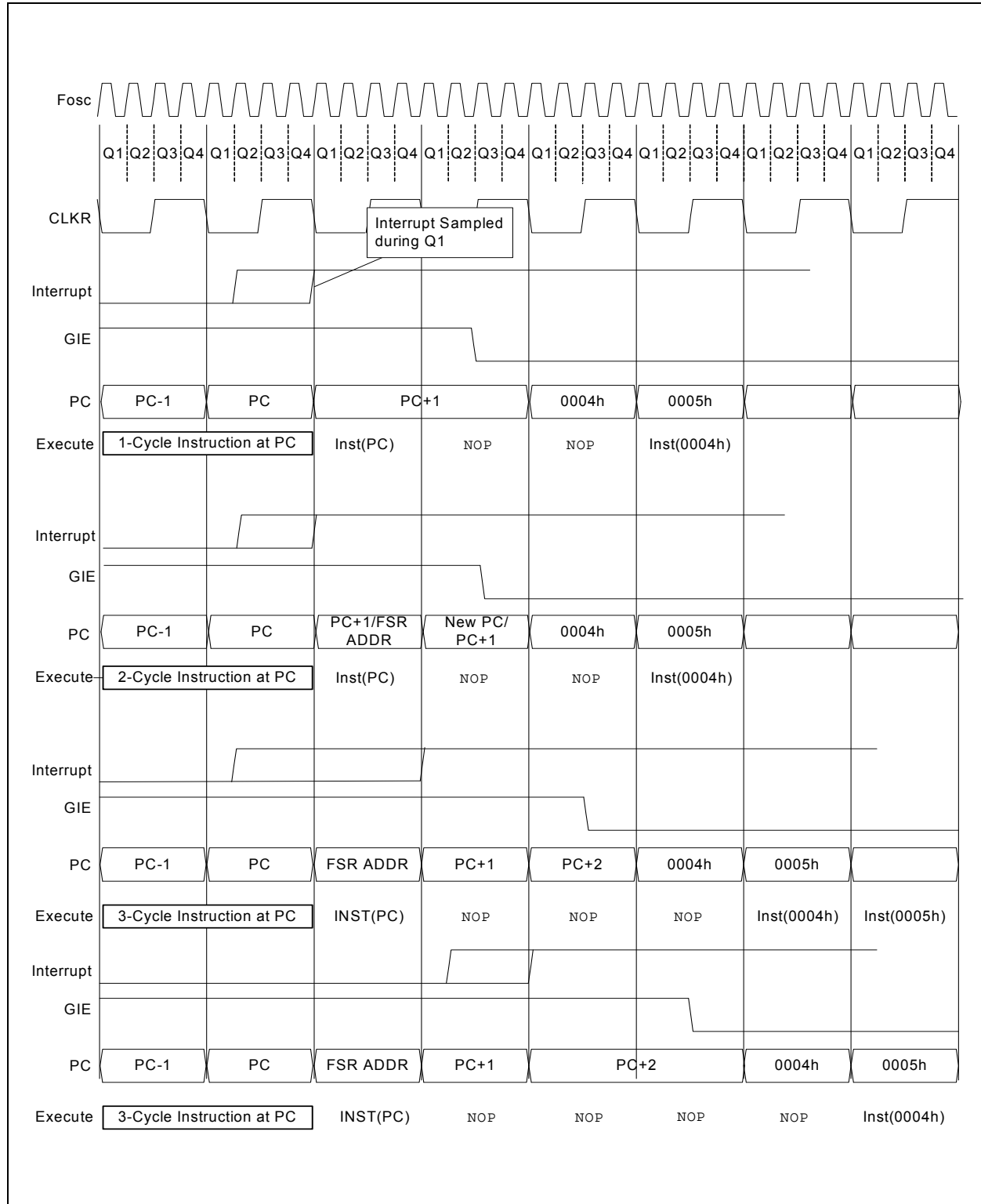
**TABLE 3-14: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)**

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other Resets
Bank 14											
70Ch to 710h	—	Unimplemented								—	—
711h	WDTCN0	—	—	WDTPS<4:0>					SEN	--qq qqqq	--qq qqqq
712h	WDTCN1	—	WDTCS<2:0>			—	WINDOW<2:0>			-qqq -qqq	-qqq -qqq
713h	WDTPSL	PSCNT<7:0>								0000 0000	0000 0000
714h	WDTPSH	PSCNT<15:8>								0000 0000	0000 0000
715h	WDTTMR	WDTTMR<4:0>					STATE	PSCNT<17:16>		0000 0000	0000 0000
716h	—	Unimplemented								—	—
717h	—	Unimplemented								—	—
718h	SCANLADRL	LADR<7:0>								0000 0000	0000 0000
719h	SCANLADRH	LADR<15:8>								0000 0000	0000 0000
71Ah	SCANHADRL	HADR<7:0>								1111 1111	1111 1111
71Bh	SCANHADRH	HADR<15:8>								1111 1111	1111 1111
71Ch	SCANCON0	EN	SCANGO	BUSY	INVALID	INTM	—	MODE<1:0>		0000 0-00	0000 0-00
71Dh	SCANTRIG					TSEL<3:0>			---- 0000	---- 0000	
71Eh	—	Unimplemented								—	—
71Fh	—	Unimplemented								—	—

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

- Note**
- 1: PIC16F1615/9 only.
  - 2: Unimplemented, read as '1'.
  - 3: PIC16(L)F1615 only.
  - 4: PIC16(L)F1619 only.

**FIGURE 7-2: INTERRUPT LATENCY**



## REGISTER 7-5: PIE4: PERIPHERAL INTERRUPT ENABLE REGISTER 4

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
SCANIE	CRCIE	SMT2PWAIE	SMT2PRAIE	SMT2IE	SMT1PWAIE	SMT1PRAIE	SMT1IE
bit 7							bit 0

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

u = Bit is unchanged

x = Bit is unknown

-n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set

'0' = Bit is cleared

- bit 7 **SCANIE:** Scanner Interrupt Enable bit  
 1 = Enables the scanner interrupt  
 0 = Disables the scanner interrupt
- bit 6 **CRCIE:** CRC Interrupt Enable bit  
 1 = Enables the CRC interrupt  
 0 = Disables the CRC interrupt
- bit 5 **SMT2PWAIE:** SMT2 Pulse Width Acquisition Interrupt Enable bit  
 1 = Enables the SMT2 acquisition interrupt  
 0 = Disables the SMT2 acquisition interrupt
- bit 4 **SMT2PRAIE:** SMT2 Period Acquisition Interrupt Enable bit  
 1 = Enables the SMT2 acquisition interrupt  
 0 = Disables the SMT2 acquisition interrupt
- bit 3 **SMT2IE:** SMT2 Match Interrupt Enable bit  
 1 = Enables the SMT2 period match interrupt  
 0 = Disables the SMT2 period match interrupt
- bit 2 **SMT1PWAIE:** SMT1 Pulse Width Acquisition Interrupt Enable bit  
 1 = Enables the SMT1 acquisition interrupt  
 0 = Disables the SMT1 acquisition interrupt
- bit 1 **SMT1PRAIE:** SMT1 Period Acquisition Interrupt Enable bit  
 1 = Enables the SMT1 acquisition interrupt  
 0 = Disables the SMT1 acquisition interrupt
- bit 0 **SMT1IE:** SMT1 Match Interrupt Enable bit  
 1 = Enables the SMT1 period match interrupt  
 0 = Disables the SMT1 period match interrupt

**Note:** Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

## 12.1 PORTA Registers

### 12.1.1 DATA REGISTER

PORTA is a 6-bit wide, bidirectional port. The corresponding data direction register is TRISA (Register 12-2). Setting a TRISA bit (= 1) will make the corresponding PORTA pin an input (i.e., disable the output driver). Clearing a TRISA bit (= 0) will make the corresponding PORTA pin an output (i.e., enables output driver and puts the contents of the output latch on the selected pin). The exception is RA3, which is input-only and its TRIS bit will always read as '1'. Example 12-1 shows how to initialize an I/O port.

Reading the PORTA register (Register 12-1) 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 (LATA).

### 12.1.2 DIRECTION CONTROL

The TRISA register (Register 12-2) controls the PORTA pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISA register are maintained set when using them as analog inputs. I/O pins configured as analog input always read '0'.

### 12.1.3 OPEN-DRAIN CONTROL

The ODCONA register (Register 12-6) controls the open-drain feature of the port. Open-drain operation is independently selected for each pin. When an ODCONA bit is set, the corresponding port output becomes an open-drain driver capable of sinking current only. When an ODCONA bit is cleared, the corresponding port output pin is the standard push-pull drive capable of sourcing and sinking current.

### 12.1.4 SLEW RATE CONTROL

The SLRCONA register (Register 12-7) controls the slew rate option for each port pin. Slew rate control is independently selectable for each port pin. When an SLRCONA bit is set, the corresponding port pin drive is slew rate limited. When an SLRCONA bit is cleared, The corresponding port pin drive slews at the maximum rate possible.

### 12.1.5 INPUT THRESHOLD CONTROL

The INLVLA register (Register 12-8) controls the input voltage threshold for each of the available PORTA input pins. A selection between the Schmitt Trigger CMOS or the TTL Compatible thresholds is available. The input threshold is important in determining the value of a read of the PORTA register and also the level at which an interrupt-on-change occurs, if that feature is enabled. See 35.3 "DC Characteristics" for more information on threshold levels.

**Note:** Changing the input threshold selection should be performed while all peripheral modules are disabled. Changing the threshold level during the time a module is active may inadvertently generate a transition associated with an input pin, regardless of the actual voltage level on that pin.

### 12.1.6 ANALOG CONTROL

The ANSELA register (Register 12-4) is used to configure the Input mode of an I/O pin to analog. Setting the appropriate ANSELA bit high will cause all digital reads on the pin to be read as '0' and allow analog functions on the pin to operate correctly.

The state of the ANSELA bits has no effect on digital output functions. A pin with TRIS clear and ANSEL set will still operate as a digital output, but the Input mode will be analog. This can cause unexpected behavior when executing read-modify-write instructions on the affected port.

**Note:** The ANSELA bits default to the Analog mode after Reset. To use any pins as digital general purpose or peripheral inputs, the corresponding ANSEL bits must be initialized to '0' by user software.

#### EXAMPLE 12-1: INITIALIZING PORTA

```
BANKSEL PORTA      ;
CLRF PORTA          ;Init PORTA
BANKSEL LATA        ;Data Latch
CLRF LATA           ;
BANKSEL ANSELA      ;
CLRF ANSELA         ;digital I/O
BANKSEL TRISA       ;
MOVLW B'00111000'  ;Set RA<5:3> as inputs
MOVWF TRISA         ;and set RA<2:0> as
                   ;outputs
```

## 19.2 Comparator Control

Each comparator has two control registers: CMxCON0 and CMxCON1.

The CMxCON0 registers (see Register 19-1) contain Control and Status bits for the following:

- Enable
- Output selection
- Output polarity
- Speed/Power selection
- Hysteresis enable
- Output synchronization

The CMxCON1 registers (see Register 19-2) contain Control bits for the following:

- Interrupt enable
- Interrupt edge polarity
- Positive input channel selection
- Negative input channel selection

### 19.2.1 COMPARATOR ENABLE

Setting the CxON bit of the CMxCON0 register enables the comparator for operation. Clearing the CxON bit disables the comparator resulting in minimum current consumption.

### 19.2.2 COMPARATOR OUTPUT SELECTION

The output of the comparator can be monitored by reading either the CxOUT bit of the CMxCON0 register or the MCxOUT bit of the CMOUT register. In order to make the output available for an external connection, the following conditions must be true:

- CxOE bit of the CMxCON0 register must be set
- Corresponding TRIS bit must be cleared
- CxON bit of the CMxCON0 register must be set

**Note 1:** The CxOE bit of the CMxCON0 register overrides the PORT data latch. Setting the CxON bit of the CMxCON0 register has no impact on the port override.

**2:** The internal output of the comparator is latched with each instruction cycle. Unless otherwise specified, external outputs are not latched.

### 19.2.3 COMPARATOR OUTPUT POLARITY

Inverting the output of the comparator is functionally equivalent to swapping the comparator inputs. The polarity of the comparator output can be inverted by setting the CxPOL bit of the CMxCON0 register. Clearing the CxPOL bit results in a non-inverted output.

Table 19-2 shows the output state versus input conditions, including polarity control.

**TABLE 19-2: COMPARATOR OUTPUT STATE VS. INPUT CONDITIONS**

Input Condition	CxPOL	CxOUT
$CxVN > CxVP$	0	0
$CxVN < CxVP$	0	1
$CxVN > CxVP$	1	1
$CxVN < CxVP$	1	0

### 19.2.4 COMPARATOR SPEED/POWER SELECTION

The trade-off between speed or power can be optimized during program execution with the CxSP control bit. The default state for this bit is '1' which selects the Normal Speed mode. Device power consumption can be optimized at the cost of slower comparator propagation delay by clearing the CxSP bit to '0'.

## 24.6 I<sup>2</sup>C Master Mode

Master mode is enabled by setting and clearing the appropriate SSPM bits in the SSPxCON1 register and by setting the SSPEN bit. In Master mode, the SDA and SCK pins must be configured as inputs. The MSSP peripheral hardware will override the output driver TRIS controls when necessary to drive the pins low.

Master mode of operation is supported by interrupt generation on the detection of the Start and Stop conditions. The Stop (P) and Start (S) bits are cleared from a Reset or when the MSSP module is disabled. Control of the I<sup>2</sup>C bus may be taken when the P bit is set, or the bus is Idle.

In Firmware Controlled Master mode, user code conducts all I<sup>2</sup>C bus operations based on Start and Stop bit condition detection. Start and Stop condition detection is the only active circuitry in this mode. All other communication is done by the user software directly manipulating the SDA and SCL lines.

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

- Start condition detected
- Stop condition detected
- Data transfer byte transmitted/received
- Acknowledge transmitted/received
- Repeated Start generated

**Note 1:** The MSSP module, when configured in I<sup>2</sup>C Master mode, does not allow queuing of events. For instance, the user is not allowed to initiate a Start condition and immediately write the SSPxBUF register to initiate transmission before the Start condition is complete. In this case, the SSPxBUF will not be written to and the WCOL bit will be set, indicating that a write to the SSPxBUF did not occur

- 2:** When in Master mode, Start/Stop detection is masked and an interrupt is generated when the SEN/PEN bit is cleared and the generation is complete.

### 24.6.1 I<sup>2</sup>C 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 I<sup>2</sup>C 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/W) bit. In this case, the R/W bit will be logic '0'. Serial data is transmitted eight 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/W bit. In this case, the R/W bit will be logic '1'. Thus, the first byte transmitted is a 7-bit slave address followed by a '1' to indicate the receive bit. Serial data is received via SDA, while SCL outputs the serial clock. Serial data is received eight 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.

A Baud Rate Generator is used to set the clock frequency output on SCL. See **Section 24.7 "Baud Rate Generator"** for more detail.

**TABLE 25-2: SUMMARY OF REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION**

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	—	—	—	ANSA4	—	ANSA2	ANSA1	ANSA0	160
ANSELB <sup>(1)</sup>	—	—	ANSB5	ANSB4	—	—	—	—	167
ANSELC	ANSC7 <sup>(1)</sup>	ANSC6 <sup>(1)</sup>	—	—	ANSC3	ANSC2	ANSC1	ANSC0	174
BAUD1CON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	331
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	105
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	105
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	111
RC1REG	EUSART Receive Data Register								324*
RC1STA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	330
RxyPPS	—	—	—	RxyPPS<4:0>					180
SP1BRGL	BRG<7:0>								332
SP1BRGH	BRG<15:8>								332
TRISA	—	—	TRISA5	TRISA4	— <sup>(2)</sup>	TRISA2	TRISA1	TRISA0	159
TRISB <sup>(1)</sup>	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	166
TRISC	TRISC7 <sup>(1)</sup>	TRISC6 <sup>(1)</sup>	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	173
TX1STA	CSRC	TX9	TXEN	SYNC	SEnDB	BRGH	TRMT	TX9D	329

**Legend:** — = unimplemented location, read as '0'. Shaded cells are not used for asynchronous reception.

\* Page provides register information.

**Note 1:** PIC16(L)F1619 only.

**2:** Unimplemented, read as '1'.



## REGISTER 26-3: CCPRxL: CCPx LOW BYTE REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
CCPR<7:0>							
bit 7				bit 0			

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

u = Bit is unchanged

x = Bit is unknown

-n/n = Value at POR and BOR/Value at all other Reset

'1' = Bit is set

'0' = Bit is cleared

bit 7-0

MODE = Capture Mode

**CCPRxL<7:0>**: LSB of captured TMR1 value

MODE = Compare Mode

**CCPRxL<7:0>**: LSB compared to TMR1 value

MODE = PWM Mode && FMT = 0

**CCPRxL<7:0>**: CCPW<7:0> — Pulse width Least Significant eight bits

MODE = PWM Mode && FMT = 1

**CCPRxL<7:6>**: CCPW<1:0> — Pulse width Least Significant two bits

**CCPRxL<5:0>**: Not used

## 28.5 Dead-Band Control

The dead-band control provides non-overlapping PWM signals to prevent shoot-through current in PWM switches. Dead-band operation is employed for Half-Bridge and Full-Bridge modes. The CWG contains two 6-bit dead-band counters. One is used for the rising edge of the input source control in Half-Bridge mode or for reverse dead-band Full-Bridge mode. The other is used for the falling edge of the input source control in Half-Bridge mode or for forward dead band in Full-Bridge mode.

Dead band is timed by counting CWG clock periods from zero up to the value in the rising or falling dead-band counter registers. See CWGxDBR and CWGxDBF registers, respectively.

### 28.5.1 DEAD-BAND FUNCTIONALITY IN HALF-BRIDGE MODE

In Half-Bridge mode, the dead-band counters dictate the delay between the falling edge of the normal output and the rising edge of the inverted output. This can be seen in Figure 28-9.

### 28.5.2 DEAD-BAND FUNCTIONALITY IN FULL-BRIDGE MODE

In Full-Bridge mode, the dead-band counters are used when undergoing a direction change. The MODE<0> bit of the CWGxCON0 register can be set or cleared while the CWG is running, allowing for changes from Forward to Reverse mode. The CWGxA and CWGxC signals will change immediately upon the first rising input edge following a direction change, but the modulated signals (CWGxB or CWGxD, depending on the direction of the change) will experience a delay dictated by the dead-band counters. This is demonstrated in Figure 28-3.

## 28.6 Rising Edge and Reverse Dead Band

CWGxDBR controls the rising edge dead-band time at the leading edge of CWGxA (Half-Bridge mode) or the leading edge of CWGxB (Full-Bridge mode). The CWGxDBR value is double-buffered. When EN = 0, the CWGxDBR register is loaded immediately when CWGxDBR is written. When EN = 1, then software must set the LD bit of the CWGxCON0 register, and the buffer will be loaded at the next falling edge of the CWG input signal. If the input source signal is not present for enough time for the count to be completed, no output will be seen on the respective output.

## 28.7 Falling Edge and Forward Dead Band

CWGxDBF controls the dead-band time at the leading edge of CWGxB (Half-Bridge mode) or the leading edge of CWGxD (Full-Bridge mode). The CWGxDBF value is double-buffered. When EN = 0, the CWGxDBF register is loaded immediately when CWGxDBF is written. When EN = 1 then software must set the LD bit of the CWGxCON0 register, and the buffer will be loaded at the next falling edge of the CWG input signal. If the input source signal is not present for enough time for the count to be completed, no output will be seen on the respective output. Refer to Figure 28.6 and Figure 28-7 for examples.

## 28.12 Configuring the CWG

The following steps illustrate how to properly configure the CWG.

1. Ensure that the TRIS control bits corresponding to the desired CWG pins for your application are set so that the pins are configured as inputs.
2. Clear the EN bit, if not already cleared.
3. Set desired mode of operation with the MODE bits.
4. Set desired dead-band times, if applicable to mode, with the CWGxDBR and CWGxDBF registers.
5. Setup the following controls in the CWGxAS0 and CWGxAS1 registers.
  - a. Select the desired shutdown source.
  - b. Select both output overrides to the desired levels (this is necessary even if not using auto-shutdown because start-up will be from a shutdown state).
  - c. Set which pins will be affected by auto-shutdown with the CWGxAS1 register.
  - d. Set the SHUTDOWN bit and clear the REN bit.
6. Select the desired input source using the CWGxISM register.
7. Configure the following controls.
  - a. Select desired clock source using the CWGxCLKCON register.
  - b. Select the desired output polarities using the CWGxCON1 register.
  - c. Set the output enables for the desired outputs.
8. Set the EN bit.
9. Clear TRIS control bits corresponding to the desired output pins to configure these pins as outputs.
10. If auto-restart is to be used, set the REN bit and the SHUTDOWN bit will be cleared automatically. Otherwise, clear the SHUTDOWN bit to start the CWG.

### 28.12.1 PIN OVERRIDE LEVELS

The levels driven to the output pins, while the shutdown input is true, are controlled by the LSB and LSAC bits of the CWGxAS0 register. LSB<1:0> controls the CWGxB and D override levels and LSAC<1:0> controls the CWGxA and C override levels. The control bit logic level corresponds to the output logic drive level while in the shutdown state. The polarity control does not affect the override level.

### 28.12.2 AUTO-SHUTDOWN RESTART

After an auto-shutdown event has occurred, there are two ways to resume operation:

- Software controlled
- Auto-restart

The restart method is selected with the REN bit of the CWGxAS0 register. Waveforms of software controlled and automatic restarts are shown in Figure 28-13 and Figure 28-14.

#### 28.12.2.1 Software Controlled Restart

When the REN bit of the CWGxAS0 register is cleared, the CWG must be restarted after an auto-shutdown event by software. Clearing the shutdown state requires all selected shutdown inputs to be low, otherwise the SHUTDOWN bit will remain set. The overrides will remain in effect until the first rising edge event after the SHUTDOWN bit is cleared. The CWG will then resume operation.

#### 28.12.2.2 Auto-Restart

When the REN bit of the CWGxAS0 register is set, the CWG will restart from the auto-shutdown state automatically. The SHUTDOWN bit will clear automatically when all shutdown sources go low. The overrides will remain in effect until the first rising edge event after the SHUTDOWN bit is cleared. The CWG will then resume operation.

## REGISTER 29-3: CLCxSEL0: MULTIPLEXER DATA 0 SELECT REGISTERS

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	—	LCxD1S<5:0>					
bit 7							bit 0

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

u = Bit is unchanged

x = Bit is unknown

-n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set

'0' = Bit is cleared

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

bit 5-0 **LCxD1S<5:0>**: Input Data 1 Selection Control bits

See Table 29-1 for signal names associated with inputs.

## REGISTER 29-4: CLCxSEL1: MULTIPLEXER DATA 1 SELECT REGISTERS

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	—	LCxD2S<5:0>					
bit 7							bit 0

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

u = Bit is unchanged

x = Bit is unknown

-n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set

'0' = Bit is cleared

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

bit 5-0 **LCxD2S<5:0>**: Input Data 2 Selection Control bits

See Table 29-1 for signal names associated with inputs.

## REGISTER 29-5: CLCxSEL2: MULTIPLEXER DATA 2 SELECT REGISTERS

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	—	LCxD3S<5:0>					
bit 7							bit 0

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

u = Bit is unchanged

x = Bit is unknown

-n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set

'0' = Bit is cleared

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

bit 5-0 **LCxD3S<5:0>**: Input Data 3 Selection Control bits

See Table 29-1 for signal names associated with inputs.

## 30.0 SIGNAL MEASUREMENT TIMER (SMT)

The SMT is a 24-bit counter with advanced clock and gating logic, which can be configured for measuring a variety of digital signal parameters such as pulse width, frequency and duty cycle, and the time difference between edges on two signals.

Features of the SMT include:

- 24-bit timer/counter
  - Four 8-bit registers (SMTxTMRL/H/U)
  - Readable and writable
  - Optional 16-bit operating mode
- Two 24-bit measurement capture registers
- One 24-bit period match register
- Multi-mode operation, including relative timing measurement
- Interrupt on period match
- Multiple clock, gate and signal sources
- Interrupt on acquisition complete
- Ability to read current input values

**Note:** These devices implement two SMT modules. All references to SMTx apply to SMT1 and SMT2.

## REGISTER 31-17: ATxSTPTH: ANGULAR TIMER SET POINT HIGH REGISTER <sup>(1)</sup>

U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	STPT<14:8>						
bit 7							bit 0

### Legend:

R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	q = Value depends on condition

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

bit 6-0      **STPT<14:8>:** Set Point Most Significant bits. ATxSTPT determines the threshold setting that the period is compared against for error calculation.

**Note 1:** Writes to ATxSTPTH are double buffered. The value written to this register is held until a write to ATxSTPTL occurs, at which point the value will be latched into the register

## REGISTER 31-18: ATxSTPTL: ANGULAR TIMER SET POINT LOW REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
STPT<7:0>							
bit 7							bit 0

### Legend:

R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	q = Value depends on condition

bit 7-0      **STPT<7:0>:** Set Point Least Significant bits. ATxSTPT determines the threshold setting that the period is compared against for error calculation.

## REGISTER 32-14: PIDxOUTHL: PID OUTPUT HIGH LOW REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
OUT<23:16>							
bit 7				bit 0			

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

u = Bit is unchanged

x = Bit is unknown

-n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set

'0' = Bit is cleared

q = Value depends on condition

bit 7-0

**OUT<23:16>:** Bits <23:16> of OUT. OUT is the output value of the PID after completing the designated calculation on the specified inputs.

## REGISTER 32-15: PIDxOUTLH: PID OUTPUT LOW HIGH REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
OUT<15:8>							
bit 7				bit 0			

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

u = Bit is unchanged

x = Bit is unknown

-n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set

'0' = Bit is cleared

q = Value depends on condition

bit 7-0

**OUT<15:8>:** Bits <15:8> of OUT. OUT is the output value of the PID after completing the designated calculation on the specified inputs.

## REGISTER 32-16: PIDxOUTLL: PID OUTPUT LOW LOW REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
OUT<7:0>							
bit 7				bit 0			

### Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

u = Bit is unchanged

x = Bit is unknown

-n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set

'0' = Bit is cleared

q = Value depends on condition

bit 7-0

**OUT<7:0>:** Bits <7:0> of OUT. OUT is the output value of the PID after completing the designated calculation on the specified inputs.

**TABLE 34-3: ENHANCED MID-RANGE INSTRUCTION SET**

Mnemonic, Operands		Description	Cycles	14-Bit Opcode				Status Affected	Notes
				MSb		LSb			
BYTE-ORIENTED FILE REGISTER OPERATIONS									
ADDWF	f, d	Add W and f	1	00	0111	dfff	ffff	C, DC, Z	2
ADDWFC	f, d	Add with Carry W and f	1	11	1101	dfff	ffff	C, DC, Z	2
ANDWF	f, d	AND W with f	1	00	0101	dfff	ffff	Z	2
ASRF	f, d	Arithmetic Right Shift	1	11	0111	dfff	ffff	C, Z	2
LSLF	f, d	Logical Left Shift	1	11	0101	dfff	ffff	C, Z	2
LSRF	f, d	Logical Right Shift	1	11	0110	dfff	ffff	C, Z	2
CLRF	f	Clear f	1	00	0001	1fff	ffff	Z	2
CLRW	—	Clear W	1	00	0001	0000	00xx	Z	
COMF	f, d	Complement f	1	00	1001	dfff	ffff	Z	2
DECF	f, d	Decrement f	1	00	0011	dfff	ffff	Z	2
INCF	f, d	Increment f	1	00	1010	dfff	ffff	Z	2
IORWF	f, d	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	2
MOVF	f, d	Move f	1	00	1000	dfff	ffff	Z	2
MOVWF	f	Move W to f	1	00	0000	1fff	ffff		2
RLF	f, d	Rotate Left f through Carry	1	00	1101	dfff	ffff	C	2
RRF	f, d	Rotate Right f through Carry	1	00	1100	dfff	ffff	C	2
SUBWF	f, d	Subtract W from f	1	00	0010	dfff	ffff	C, DC, Z	2
SUBWFB	f, d	Subtract with Borrow W from f	1	11	1011	dfff	ffff	C, DC, Z	2
SWAPF	f, d	Swap nibbles in f	1	00	1110	dfff	ffff		2
XORWF	f, d	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	2
BYTE ORIENTED SKIP OPERATIONS									
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1, 2
INCFSZ	f, d	Increment f, Skip if 0	1(2)	00	1111	dfff	ffff		1, 2
BIT-ORIENTED FILE REGISTER OPERATIONS									
BCF	f, b	Bit Clear f	1	01	00bb	bfff	ffff		2
BSF	f, b	Bit Set f	1	01	01bb	bfff	ffff		2
BIT-ORIENTED SKIP OPERATIONS									
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01	10bb	bfff	ffff		1, 2
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01	11bb	bfff	ffff		1, 2
LITERAL OPERATIONS									
ADDLW	k	Add literal and W	1	11	1110	kkkk	kkkk	C, DC, Z	
ANDLW	k	AND literal with W	1	11	1001	kkkk	kkkk	Z	
IORLW	k	Inclusive OR literal with W	1	11	1000	kkkk	kkkk	Z	
MOVLB	k	Move literal to BSR	1	00	0000	001k	kkkk		
MOVLW	k	Move literal to PCLATH	1	11	0001	1kkk	kkkk		
MOVLW	k	Move literal to W	1	11	0000	kkkk	kkkk		
SUBLW	k	Subtract W from literal	1	11	1100	kkkk	kkkk	C, DC, Z	
XORLW	k	Exclusive OR literal with W	1	11	1010	kkkk	kkkk	Z	

**Note 1:** If the Program Counter (PC) is modified, or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

**2:** If this instruction addresses an INDF register and the MSb of the corresponding FSR is set, this instruction will require one additional instruction cycle.



<b>GOTO</b>	<b>Unconditional Branch</b>
Syntax:	[ <i>label</i> ] GOTO <i>k</i>
Operands:	$0 \leq k \leq 2047$
Operation:	$k \rightarrow PC<10:0>$ $PCLATH<6:3> \rightarrow PC<14:11>$
Status Affected:	None
Description:	GOTO is an unconditional branch. The 11-bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. GOTO is a 2-cycle instruction.

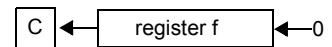
<b>INCF</b>	<b>Increment f</b>
Syntax:	[ <i>label</i> ] INCF <i>f,d</i>
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	$(f) + 1 \rightarrow (\text{destination})$
Status Affected:	Z
Description:	The contents of register 'f' are incremented. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

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

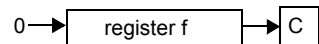
<b>IORLW</b>	<b>Inclusive OR literal with W</b>
Syntax:	[ <i>label</i> ] IORLW <i>k</i>
Operands:	$0 \leq k \leq 255$
Operation:	$(W) .OR. k \rightarrow (W)$
Status Affected:	Z
Description:	The contents of the W register are OR'ed with the 8-bit literal 'k'. The result is placed in the W register.

<b>IORWF</b>	<b>Inclusive OR W with f</b>
Syntax:	[ <i>label</i> ] IORWF <i>f,d</i>
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	$(W) .OR. (f) \rightarrow (\text{destination})$
Status Affected:	Z
Description:	Inclusive OR the W register with register 'f'. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

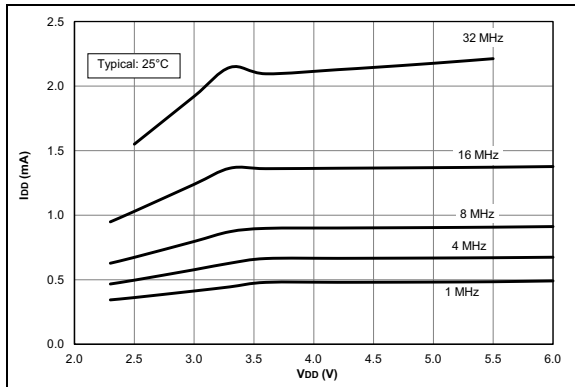
<b>LSLF</b>	<b>Logical Left Shift</b>
Syntax:	[ <i>label</i> ] LSLF <i>f,d</i>
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	$(f<7>) \rightarrow C$ $(f<6:0>) \rightarrow \text{dest}<7:1>$ $0 \rightarrow \text{dest}<0>$
Status Affected:	C, Z
Description:	The contents of register 'f' are shifted one bit to the left through the Carry flag. A '0' is shifted into the LSb. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is stored back in register 'f'.



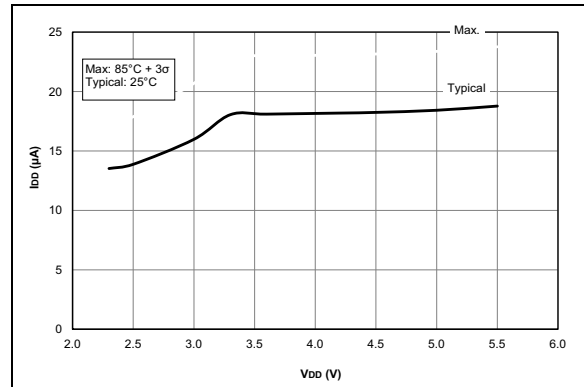
<b>LSRF</b>	<b>Logical Right Shift</b>
Syntax:	[ <i>label</i> ] LSRF <i>f,d</i>
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	$0 \rightarrow \text{dest}<7>$ $(f<7:1>) \rightarrow \text{dest}<6:0>$ , $(f<0>) \rightarrow C$
Status Affected:	C, Z
Description:	The contents of register 'f' are shifted one bit to the right through the Carry flag. A '0' is shifted into the MSb. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is stored back in register 'f'.



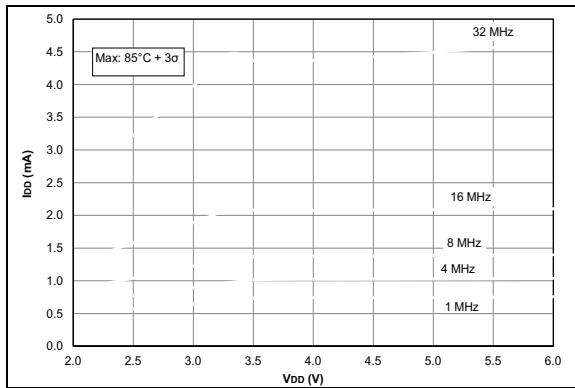
**Note:** Unless otherwise noted,  $V_{IN} = 5V$ ,  $F_{OSC} = 500\text{ kHz}$ ,  $C_{IN} = 0.1\text{ }\mu\text{F}$ ,  $T_A = 25^\circ\text{C}$ .



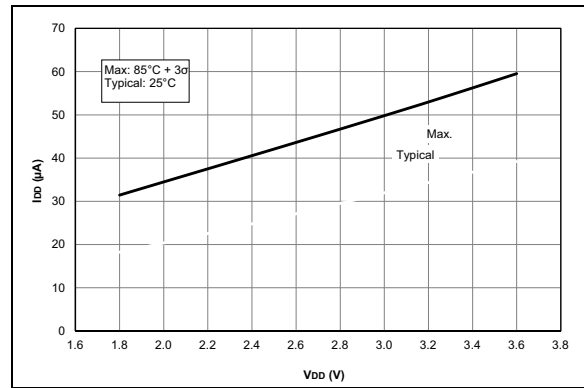
**FIGURE 36-7:**  $I_{DD}$  Typical, EC Oscillator MP Mode, PIC16F1615/9 Only.



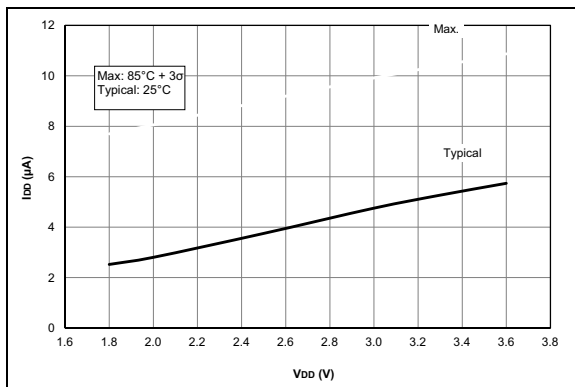
**FIGURE 36-10:**  $I_{DD}$  Maximum, EC Oscillator HP Mode, PIC16LF1615/9 Only.



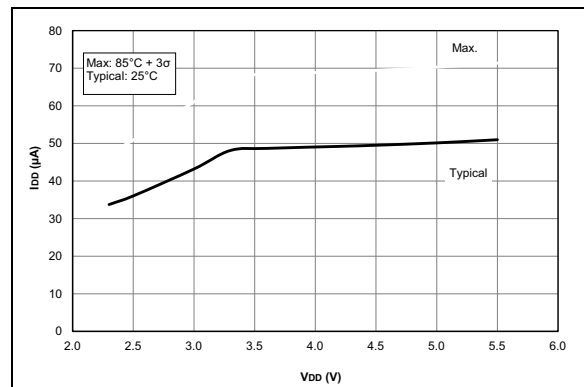
**FIGURE 36-8:**  $I_{DD}$  Maximum, EC Oscillator MP Mode, PIC16F1615/9 Only.



**FIGURE 36-11:**  $I_{DD}$  Typical, EC Oscillator HP Mode, PIC16F1615/9 Only.



**FIGURE 36-9:**  $I_{DD}$  Typical, EC Oscillator HP Mode, PIC16LF1615/9 Only.



**FIGURE 36-12:**  $I_{DD}$  Maximum, EC Oscillator HP Mode, PIC16F1615/9 Only.

## 37.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers (MCU) and dsPIC® digital signal controllers (DSC) are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB® X IDE Software
- Compilers/Assemblers/Linkers
  - MPLAB XC Compiler
  - MPASM™ Assembler
  - MPLINK™ Object Linker/  
MPLIB™ Object Librarian
  - MPLAB Assembler/Linker/Librarian for  
Various Device Families
- Simulators
  - MPLAB X SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers/Programmers
  - MPLAB ICD 3
  - PICKit™ 3
- Device Programmers
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards,  
Evaluation Kits and Starter Kits
- Third-party development tools

## 37.1 MPLAB X Integrated Development Environment Software

The MPLAB X IDE is a single, unified graphical user interface for Microchip and third-party software, and hardware development tool that runs on Windows®, Linux and Mac OS® X. Based on the NetBeans IDE, MPLAB X IDE is an entirely new IDE with a host of free software components and plug-ins for high-performance application development and debugging. Moving between tools and upgrading from software simulators to hardware debugging and programming tools is simple with the seamless user interface.

With complete project management, visual call graphs, a configurable watch window and a feature-rich editor that includes code completion and context menus, MPLAB X IDE is flexible and friendly enough for new users. With the ability to support multiple tools on multiple projects with simultaneous debugging, MPLAB X IDE is also suitable for the needs of experienced users.

Feature-Rich Editor:

- Color syntax highlighting
- Smart code completion makes suggestions and provides hints as you type
- Automatic code formatting based on user-defined rules
- Live parsing

User-Friendly, Customizable Interface:

- Fully customizable interface: toolbars, toolbar buttons, windows, window placement, etc.
- Call graph window

Project-Based Workspaces:

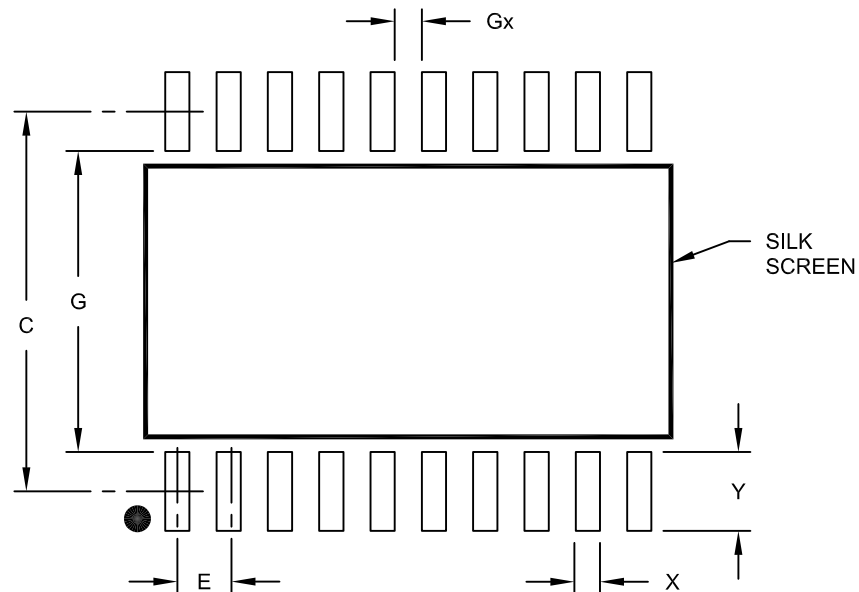
- Multiple projects
- Multiple tools
- Multiple configurations
- Simultaneous debugging sessions

File History and Bug Tracking:

- Local file history feature
- Built-in support for Bugzilla issue tracker

## 20-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 (X20)	X			0.60
Contact Pad Length (X20)	Y			1.95
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	7.45		

#### Notes:

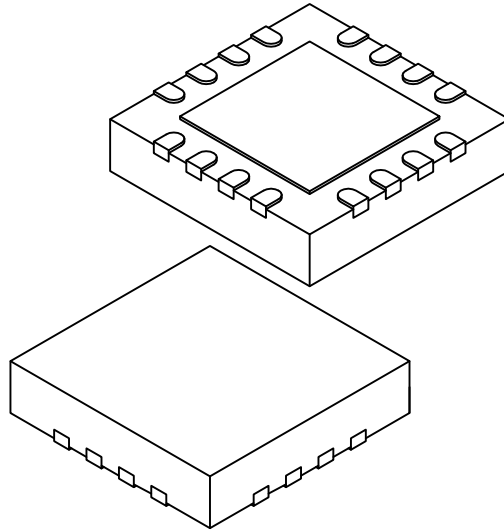
1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2094A

## 16-Lead Plastic Quad Flat, No Lead Package (ML) - 4x4x0.9mm Body [QFN]

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



		Units	MILLIMETERS		
Dimension Limits			MIN	NOM	MAX
Number of Pins	N		16		
Pitch	e		0.65 BSC		
Overall Height	A		0.80	0.90	1.00
Standoff	A1		0.00	0.02	0.05
Contact Thickness	A3		0.20 REF		
Overall Width	E		4.00 BSC		
Exposed Pad Width	E2		2.50	2.65	2.80
Overall Length	D		4.00 BSC		
Exposed Pad Length	D2		2.50	2.65	2.80
Contact Width	b		0.25	0.30	0.35
Contact Length	L		0.30	0.40	0.50
Contact-to-Exposed Pad	K		0.20	-	-

### Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

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

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

Microchip Technology Drawing C04-127D Sheet 2 of 2