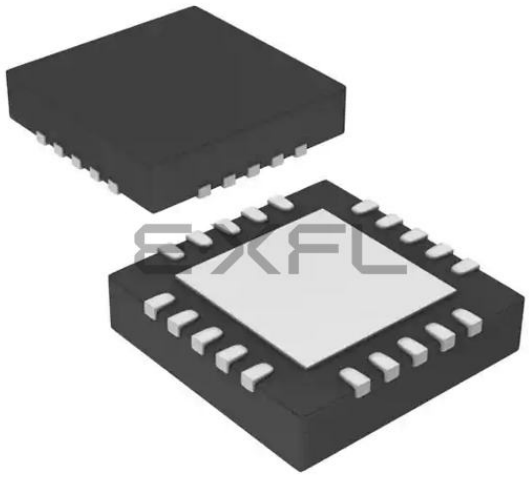


Welcome to E-XFL.COM

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	32MHz
Connectivity	I²C, 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-VFQFN Exposed Pad
Supplier Device Package	20-QFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1619-i-ml

TABLE 3-3: PIC16(L)F1619 MEMORY MAP, BANK 0-7

BANK 0		BANK 1		BANK 2		BANK 3		BANK 4		BANK 5		BANK 6		BANK 7	
000h	Core Registers (Table 3-1)	080h	Core Registers (Table 3-1)	100h	Core Registers (Table 3-1)	180h	Core Registers (Table 3-1)	200h	Core Registers (Table 3-1)	280h	Core Registers (Table 3-1)	300h	Core Registers (Table 3-1)	380h	Core Registers (Table 3-1)
00Bh		08Bh		10Bh		18Bh		20Bh		28Bh		30Bh		38Bh	
00Ch	PORTA	08Ch	TRISA	10Ch	LATA	18Ch	ANSELA	20Ch	WPUA	28Ch	ODCONA	30Ch	SLRCONA	38Ch	INLVLA
00Dh	PORTB	08Dh	TRISB	10Dh	LATB	18Dh	ANSELB	20Eh	WPUB	28Eh	ODCONB	30Eh	SLRCONB	38Eh	INLVLB
00Eh	PORTC	08Eh	TRISC	10Eh	LATC	18Eh	ANSELC	20Eh	WPUC	28Eh	ODCONC	30Eh	SLRCONC	38Eh	INLVLC
00Fh	—	08Fh	—	10Fh	—	18Fh	—	20Fh	—	28Fh	—	30Fh	—	38Fh	—
010h	PIR1	090h	PIE1	110h	—	190h	—	210h	—	290h	—	310h	—	390h	—
011h	PIR2	091h	PIE2	111h	CM1CON0	191h	PMADRL	211h	SSP1BUF	291h	CCP1RL	311h	—	391h	IOCAP
012h	PIR3	092h	PIE3	112h	CM1CON1	192h	PMADRH	212h	SSP1ADD	292h	CCP1RH	312h	—	392h	IOCAN
013h	PIR4	093h	PIE4	113h	CM2CON0	193h	PMDATL	213h	SSP1MSK	293h	CCP1CON	313h	—	393h	IOCAF
014h	PIR5	094h	PIE5	114h	CM2CON1	194h	PMDATH	214h	SSP1STAT	294h	CCP1CAP	314h	—	394h	IOCBP
015h	TMR0	095h	OPTION_REG	115h	CMOUT	195h	PMCON1	215h	SSP1CON	295h	—	315h	—	395h	IOCBN
016h	TMR1L	096h	PCON	116h	BORCON	196h	PMCON2	216h	SSP1CON2	296h	—	316h	—	396h	IOCBF
017h	TMR1H	097h	—	117h	FVRCON	197h	VREGCON	217h	SSP1CON3	297h	—	317h	—	397h	IOCCP
018h	T1CON	098h	OSCTUNE	118h	DAC1CON0	198h	—	218h	—	298h	CCP2RL	318h	—	398h	IOCCN
019h	T1GCON	099h	OSCCON	119h	DAC1CON1	199h	RC1REG	219h	—	299h	CCP2RH	319h	—	399h	IOCCF
01Ah	TMR2	09Ah	OSCSTAT	11Ah	—	19Ah	TX1REG	21Ah	—	29Ah	CCP2CON	31Ah	—	39Ah	—
01Bh	PR2	09Bh	ADRESL	11Bh	—	19Bh	SP1BRGL	21Bh	—	29Bh	CCP2CAP	31Bh	—	39Bh	—
01Ch	T2CON	09Ch	ADRESH	11Ch	ZCD1CON	19Ch	SP1BRGH	21Ch	—	29Ch	—	31Ch	—	39Ch	—
01Dh	T2HLT	09Dh	ADCON0	11Dh	—	19Dh	RC1STA	21Dh	—	29Dh	—	31Dh	—	39Dh	—
01Eh	T2CLKCON	09Eh	ADCON1	11Eh	—	19Eh	TX1STA	21Eh	—	29Eh	CCPTMRS	31Eh	—	39Eh	—
01Fh	T2RST	09Fh	ADCON2	11Fh	—	19Fh	BAUD1CON	21Fh	—	29Fh	—	31Fh	—	39Fh	—
020h	General Purpose Register 96 Bytes	0A0h	General Purpose Register 80 Bytes	120h	General Purpose Register 80 Bytes	1A0h	General Purpose Register 80 Bytes	220h	General Purpose Register 80 Bytes	2A0h	General Purpose Register 80 Bytes	320h	General Purpose Register 80 Bytes	3A0h	General Purpose Register 80 Bytes
07Fh		0EFh		16Fh		1EFh		26Fh		2EFh		36Fh		3EFh	
		0F0h	Common RAM (Accesses 70h – 7Fh)	170h	Common RAM (Accesses 70h – 7Fh)	1F0h	Common RAM (Accesses 70h – 7Fh)	270h	Common RAM (Accesses 70h – 7Fh)	2F0h	Common RAM (Accesses 70h – 7Fh)	370h	Common RAM (Accesses 70h – 7Fh)	3F0h	Common RAM (Accesses 70h – 7Fh)
		0FFh		17Fh		1FFh		27Fh		2FFh		37Fh		3FFh	

Legend: = Unimplemented data memory locations, read as '0'.

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 7											
38Ch	INLVLA	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	--11 1111	--11 1111
38Dh	INLVLB(4)	INLVLB7	INLVLB6	INLVLB5	INLVLB4	—	—	—	—	1111 ----	1111 ----
38Eh	INLVLC	INLVLC7(4)	INLVLC6(4)	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	1111 1111	1111 1111
30Fh	—	Unimplemented								—	—
390h	—	Unimplemented								—	—
391h	IOCAP	—	—	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0	--00 0000	--00 0000
392h	IOCAN	—	—	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0	--00 0000	--00 0000
393h	IOCAF	—	—	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0	--00 0000	--00 0000
394h	IOCBP(4)	IOCBP7	IOCBP6	IOCBP5	IOCBP4	—	—	—	—	0000 ----	0000 ----
395h	IOCBN(4)	IOCBN7	IOCBN6	IOCBN5	IOCBN4	—	—	—	—	0000 ----	0000 ----
396h	IOCBF(4)	IOCBF7	IOCBF6	IOCBF5	IOCBF4	—	—	—	—	0000 ----	0000 ----
397h	IOCCP	IOCCP7(4)	IOCCP6(4)	IOCCP5	IOCCP4	IOCCP3	IOCCP2	IOCCP1	IOCCP0	0000 0000	0000 0000
398h	IOCCN	IOCCN7(4)	IOCCN6(4)	IOCCN5	IOCCN4	IOCCN3	IOCCN2	IOCCN1	IOCCN0	0000 0000	0000 0000
399h	IOCCF	IOCCF7(4)	IOCCF6(4)	IOCCF5	IOCCF4	IOCCF3	IOCCF2	IOCCF1	IOCCF0	0000 0000	0000 0000
39Ah to 39Fh	—	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'.

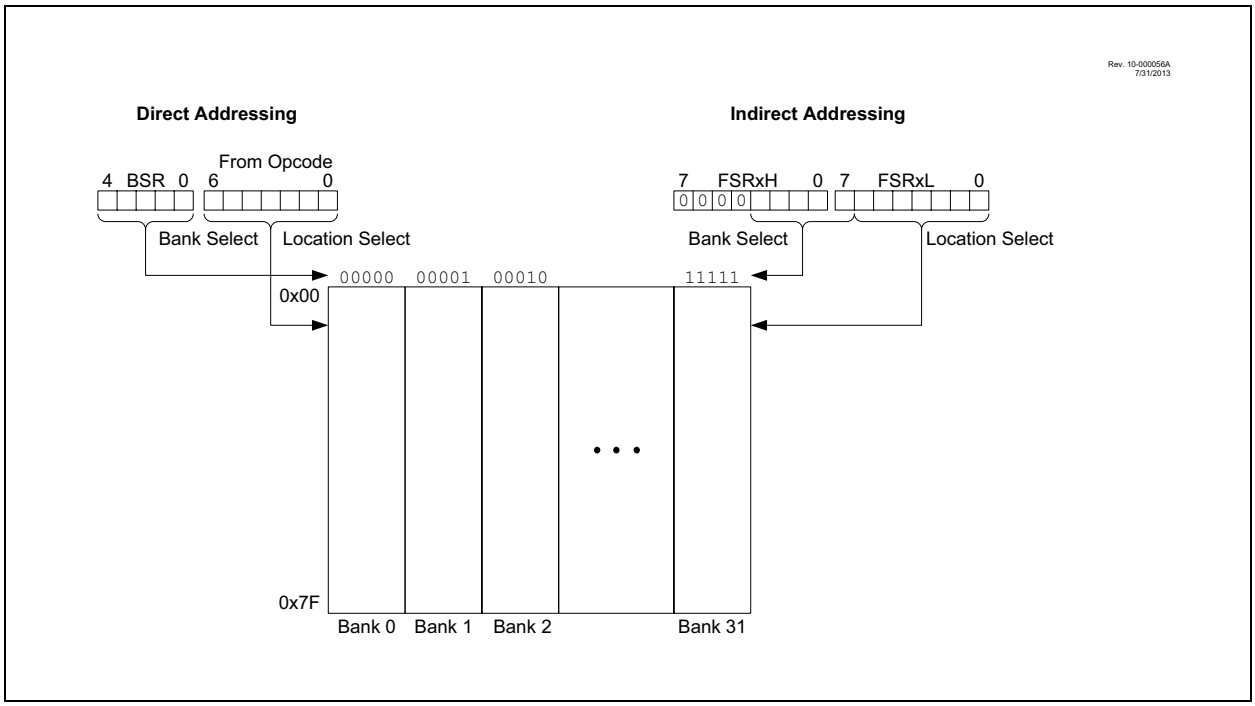
3: PIC16(L)F1615 only.

4: PIC16(L)F1619 only.

3.6.1 TRADITIONAL DATA MEMORY

The traditional data memory is a region from FSR address 0x000 to FSR address 0xFFF. The addresses correspond to the absolute addresses of all SFR, GPR and common registers.

FIGURE 3-9: TRADITIONAL DATA MEMORY MAP



REGISTER 7-4: PIE3: PERIPHERAL INTERRUPT ENABLE REGISTER 3

U-0	U-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
—	—	CWGIE	ZCDIE	CLC4IE	CLC3IE	CLC2IE	CLC1IE
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 **CWGIE**: Complementary Waveform Generator (CWG) Interrupt Enable bit
- 1 = Enables the CWG interrupt
- 0 = Disables the CWG interrupt

bit 4 **ZCDIE**: Zero-Cross Detection (ZCD) Interrupt Enable bit
- 1 = Enables the ZCD interrupt
- 0 = Disables the ZCD interrupt

bit 3 **CLC4IE**: Configurable Logic Block 4 Interrupt Enable bit
- 1 = Enables the CLC 4 interrupt
- 0 = Disables the CLC 4 interrupt

bit 2 **CLC3IE**: Configurable Logic Block 3 Interrupt Enable bit
- 1 = Enables the CLC 3 interrupt
- 0 = Disables the CLC 3 interrupt

bit 1 **CLC2IE**: Configurable Logic Block 2 Interrupt Enable bit
- 1 = Enables the CLC 2 interrupt
- 0 = Disables the CLC 2 interrupt

bit 0 **CLC1IE**: Configurable Logic Block 1 Interrupt Enable bit
- 1 = Enables the CLC 1 interrupt
- 0 = Disables the CLC 1 interrupt

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

8.0 POWER-DOWN MODE (SLEEP)

The Power-Down mode is entered by executing a SLEEP instruction.

Upon entering Sleep mode, the following conditions exist:

1. WDT will be cleared but keeps running, if enabled for operation during Sleep.
2. $\overline{PD}$ bit of the STATUS register is cleared.
3. $\overline{TO}$ bit of the STATUS register is set.
4. CPU clock is disabled.
5. 31 kHz LFINTOSC is unaffected and peripherals that operate from it may continue operation in Sleep.
6. Timer1 and peripherals that operate from Timer1 continue operation in Sleep when the Timer1 clock source selected is:
   - LFINTOSC
   - T1CKI
   - Timer1 oscillator
7. ADC is unaffected, if the dedicated FRC oscillator is selected.
8. I/O ports maintain the status they had before SLEEP was executed (driving high, low or high-impedance).
9. Resets other than WDT are not affected by Sleep mode.

Refer to individual chapters for more details on peripheral operation during Sleep.

To minimize current consumption, the following conditions should be considered:

- I/O pins should not be floating
- External circuitry sinking current from I/O pins
- Internal circuitry sourcing current from I/O pins
- Current draw from pins with internal weak pull-ups
- Modules using 31 kHz LFINTOSC
- CWG modules using HFINTOSC

I/O pins that are high-impedance inputs should be pulled to VDD or VSS externally to avoid switching currents caused by floating inputs.

Examples of internal circuitry that might be sourcing current include the FVR module. See **Section 15.0 "Fixed Voltage Reference (FVR)"** for more information on this module.

8.1 Wake-up from Sleep

The device can wake-up from Sleep through one of the following events:

1. External Reset input on $\overline{MCLR}$ pin, if enabled
2. BOR Reset, if enabled
3. POR Reset
4. Watchdog Timer, if enabled
5. Any external interrupt
6. Interrupts by peripherals capable of running during Sleep (see individual peripheral for more information)

The first three events will cause a device Reset. The last three events are considered a continuation of program execution. To determine whether a device Reset or wake-up event occurred, refer to **Section 6.12 "Determining the Cause of a Reset"**.

When the SLEEP instruction is being executed, the next instruction (PC + 1) is prefetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be enabled. Wake-up will occur regardless of the state of the GIE bit. If the GIE bit is disabled, the device continues execution at the instruction after the SLEEP instruction. If the GIE bit is enabled, the device executes the instruction after the SLEEP instruction, the device will then call the Interrupt Service Routine. In cases where the execution of the instruction following SLEEP is not desirable, the user should have a NOP after the SLEEP instruction.

The WDT is cleared when the device wakes up from Sleep, regardless of the source of wake-up.

8.1.1 WAKE-UP USING INTERRUPTS

When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the execution of a SLEEP instruction
  - SLEEP instruction will execute as a NOP
  - WDT and WDT prescaler will not be cleared
  - $\overline{TO}$ bit of the STATUS register will not be set
  - $\overline{PD}$ bit of the STATUS register will not be cleared
- If the interrupt occurs **during or after** the execution of a SLEEP instruction
  - SLEEP instruction will be completely executed
  - Device will immediately wake-up from Sleep
  - WDT and WDT prescaler will be cleared
  - $\overline{TO}$ bit of the STATUS register will be set
  - $\overline{PD}$ bit of the STATUS register will be cleared

Even if the flag bits were checked before executing a SLEEP instruction, it may be possible for flag bits to become set before the SLEEP instruction completes. To determine whether a SLEEP instruction executed, test the $\overline{PD}$ bit. If the $\overline{PD}$ bit is set, the SLEEP instruction was executed as a NOP.

10.2.1 READING THE FLASH PROGRAM MEMORY

To read a program memory location, the user must:

1. Write the desired address to the PMADRH:PMADRL register pair.
2. Clear the CFGS bit of the PMCON1 register.
3. Then, set control bit RD of the PMCON1 register.

Once the read control bit is set, the program memory Flash controller will use the second instruction cycle to read the data. This causes the second instruction immediately following the "BSF PMCON1,RD" instruction to be ignored. The data is available in the very next cycle, in the PMDATH:PMDATL register pair; therefore, it can be read as two bytes in the following instructions.

PMDATH:PMDATL register pair will hold this value until another read or until it is written to by the user.

Note:	The two instructions following a program memory read are required to be NOPs. This prevents the user from executing a 2-cycle instruction on the next instruction after the RD bit is set.

FIGURE 10-1: FLASH PROGRAM MEMORY READ FLOWCHART

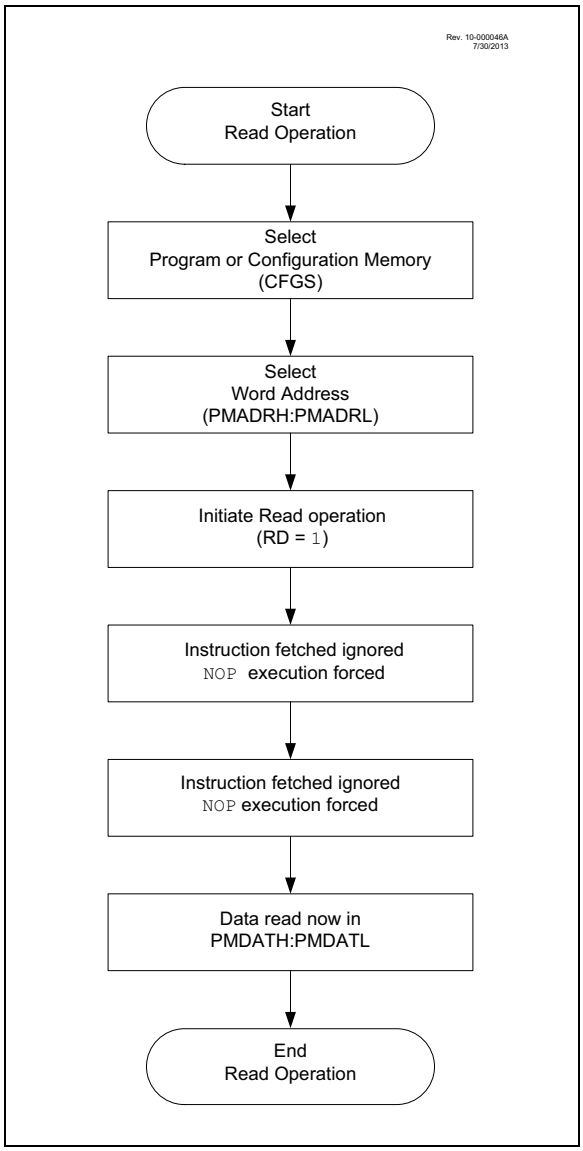


TABLE 12-5: SUMMARY OF CONFIGURATION WORD WITH PORTB

Name	Bits	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	Register on Page
CONFIG1	13:8	—	—	FCMEN	IESO	$\overline{\text{CLKOUTEN}}$	BOREN<1:0>		—	69
	7:0	$\overline{\text{CP}}$	MCLRE	$\overline{\text{PWRTE}}$	WDTE<1:0>		FOSC<2:0>			

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by PORTB.

20.9 Register Definitions: ZCD Control

REGISTER 20-1: ZCDxCON: ZERO CROSS DETECTION CONTROL REGISTER

R/W-q/q	U-0	R-x/x	R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0
ZCDxEN	—	ZCDxOUT	ZCDxPOL	—	—	ZCDxINTP	ZCDxINTN
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 configuration bits

bit 7 **ZCDxEN**: Zero-Cross Detection Enable bit
1 = Zero-cross detect is enabled. ZCD pin is forced to output to source and sink current.
0 = Zero-cross detect is disabled. ZCD pin operates according to PPS and TRIS controls.

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

bit 5 **ZCDxOUT:** Zero-Cross Detection Logic Level bit
ZCDxPOL bit = 0:
1 = ZCD pin is sinking current
0 = ZCD pin is sourcing current
ZCDxPOL bit = 1:
1 = ZCD pin is sourcing current
0 = ZCD pin is sinking current

bit 4 **ZCDxPOL:** Zero-Cross Detection Logic Output Polarity bit
1 = ZCD logic output is inverted
0 = ZCD logic output is not inverted

bit 3-2 **Unimplemented:** Read as '0'

bit 1 **ZCDxINTP:** Zero-Cross Positive Edge Interrupt Enable bit
1 = ZCDIF bit is set on low-to-high ZCDx_output transition
0 = ZCDIF bit is unaffected by low-to-high ZCDx_output transition

bit 0 **ZCDxINTN:** Zero-Cross Negative Edge Interrupt Enable bit
1 = ZCDIF bit is set on high-to-low ZCDx_output transition
0 = ZCDIF bit is unaffected by high-to-low ZCDx_output transition

TABLE 20-1: SUMMARY OF REGISTERS ASSOCIATED WITH THE ZCD MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
PIE3	—	—	CWGIE	ZCDIE	CLC4IE	CLC3IE	CLC2IE	CLC1IE	108
PIR3	—	—	CWGIF	ZCDIF	CLC4IF	CLC3IF	CLC2IF	CLC1IF	113
ZCD1CON	ZCD1EN	—	ZCD1OUT	ZCD1POL	—	—	ZCD1INTP	ZCD1INTN	227

Legend: — = unimplemented, read as '0'. Shaded cells are unused by the ZCD module.

TABLE 20-2: SUMMARY OF CONFIGURATION WORD WITH THE ZCD MODULE

Name	Bits	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	Register on Page
CONFIG2	13:8	—	—	LVP	$\overline{\text{DEBUG}}$	$\overline{\text{LPBOR}}$	BORV	STVREN	PLLEN	71
	7:0	$\overline{\text{ZCD}}$	—	—	—	—	PPS1WAY	WRT<1:0>		

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by the ZCD module.

22.1 Timer1 Operation

The Timer1 module is a 16-bit incrementing counter which is accessed through the TMR1H:TMR1L register pair. Writes to TMR1H or TMR1L directly update the counter.

When used with an internal clock source, the module is a timer and increments on every instruction cycle. When used with an external clock source, the module can be used as either a timer or counter and increments on every selected edge of the external source.

Timer1 is enabled by configuring the TMR1ON and TMR1GE bits in the T1CON and T1GCON registers, respectively. Table 22-1 displays the Timer1 enable selections.

TABLE 22-1: TIMER1 ENABLE SELECTIONS

TMR1ON	TMR1GE	Timer1 Operation
0	0	Off
0	1	Off
1	0	Always On
1	1	Count Enabled

22.2 Clock Source Selection

The TMR1CS<1:0> bits of the T1CON register are used to select the clock source for Timer1. Table 22-2 displays the clock source selections.

22.2.1 INTERNAL CLOCK SOURCE

When the internal clock source is selected, the TMR1H:TMR1L register pair will increment on multiples of FOSC as determined by the Timer1 prescaler.

When the FOSC internal clock source is selected, the Timer1 register value will increment by four counts every instruction clock cycle. Due to this condition, a 2 LSB error in resolution will occur when reading the Timer1 value. To utilize the full resolution of Timer1, an asynchronous input signal must be used to gate the Timer1 clock input.

The following asynchronous sources may be used:

- Asynchronous event on the T1G pin to Timer1 gate
- C1 or C2 comparator input to Timer1 gate

22.2.2 EXTERNAL CLOCK SOURCE

When the external clock source is selected, the Timer1 module may work as a timer or a counter.

When enabled to count, Timer1 is incremented on the rising edge of the external clock input T1CKI. The external clock source can be synchronized to the microcontroller system clock or it can run asynchronously.

Note: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge after any one or more of the following conditions:

- Timer1 enabled after POR
- Write to TMR1H or TMR1L
- Timer1 is disabled
- Timer1 is disabled (TMR1ON = 0) when T1CKI is high then Timer1 is enabled (TMR1ON=1) when T1CKI is low.

TABLE 22-2: CLOCK SOURCE SELECTIONS

TMR1CS<1:0>	Clock Source
11	LFINTOSC
10	External Clocking on T1CKI Pin
01	System Clock (FOSC)
00	Instruction Clock (FOSC/4)

22.3 Timer1 Prescaler

Timer1 has four prescaler options allowing 1, 2, 4 or 8 divisions of the clock input. The T1CKPS bits of the T1CON register control the prescale counter. The prescale counter is not directly readable or writable; however, the prescaler counter is cleared upon a write to TMR1H or TMR1L.

22.4 Timer1 Operation in Asynchronous Counter Mode

If control bit $\overline{\text{T1SYNC}}$ of the T1CON register is set, the external clock input is not synchronized. The timer increments asynchronously to the internal phase clocks. If the external clock source is selected then the timer will continue to run during Sleep and can generate an interrupt on overflow, which will wake-up the processor. However, special precautions in software are needed to read/write the timer (see **Section22.4.1 "Reading and Writing Timer1 in Asynchronous Counter Mode"**).

Note:	When switching from synchronous to asynchronous operation, it is possible to skip an increment. When switching from asynchronous to synchronous operation, it is possible to produce an additional increment.

22.4.1 READING AND WRITING TIMER1 IN ASYNCHRONOUS COUNTER MODE

Reading TMR1H or TMR1L while the timer is running from an external asynchronous clock will ensure a valid read (taken care of in hardware). However, the user should keep in mind that reading the 16-bit timer in two 8-bit values itself, poses certain problems, since the timer may overflow between the reads.

TABLE 22-4: TIMER1 GATE SOURCES

T1GSS	Timer1 Gate Source
00	Timer1 Gate pin (T1G)
01	Overflow of Timer0 (T0_overflow) (TMR0 increments from FFh to 00h)
10	Comparator 1 Output (C1_OUT_sync)(1)
11	Comparator 2 Output (C2_OUT_sync)(1)

Note 1: Optionally synchronized comparator output.

For writes, it is recommended that the user simply stop the timer and write the desired values. A write contention may occur by writing to the timer registers, while the register is incrementing. This may produce an unpredictable value in the TMR1H:TMR1L register pair.

22.5 Timer1 Gate

Timer1 can be configured to count freely or the count can be enabled and disabled using Timer1 gate circuitry. This is also referred to as Timer1 Gate Enable.

Timer1 gate can also be driven by multiple selectable sources.

22.5.1 TIMER1 GATE ENABLE

The Timer1 Gate Enable mode is enabled by setting the TMR1GE bit of the T1GCON register. The polarity of the Timer1 Gate Enable mode is configured using the T1GPOL bit of the T1GCON register.

When Timer1 Gate Enable mode is enabled, Timer1 will increment on the rising edge of the Timer1 clock source. When Timer1 Gate Enable mode is disabled, no incrementing will occur and Timer1 will hold the current count. See Figure 22-3 for timing details.

TABLE 22-3: TIMER1 GATE ENABLE SELECTIONS

T1CLK	T1GPOL	T1G	Timer1 Operation
↑	0	0	Counts
↑	0	1	Holds Count
↑	1	0	Holds Count
↑	1	1	Counts

22.5.2 TIMER1 GATE SOURCE SELECTION

Timer1 gate source selections are shown in Table 22-4. Source selection is controlled by the T1GSS<1:0> bits of the T1GCON register. The polarity for each available source is also selectable. Polarity selection is controlled by the T1GPOL bit of the T1GCON register.

23.5.10 LEVEL-TRIGGERED HARDWARE LIMIT ONE-SHOT MODES

The Level-Triggered Hardware Limit One-Shot modes hold the timer in Reset on an external Reset level and start counting when both the ON bit is set and the external signal is not at the Reset level. If one of either the external signal is not in Reset or the ON bit is set then the other signal being set/made active will start the timer. Reset levels are selected as follows:

- Low Reset level (MODE<4:0> = `10110`)
- High Reset level (MODE<4:0> = `10111`)

When the timer count matches the PRx period count, the timer is reset and the ON bit is cleared. When the ON bit is cleared by either a PRx match or by software control the timer will stay in Reset until both the ON bit is set and the external signal is not at the Reset level.

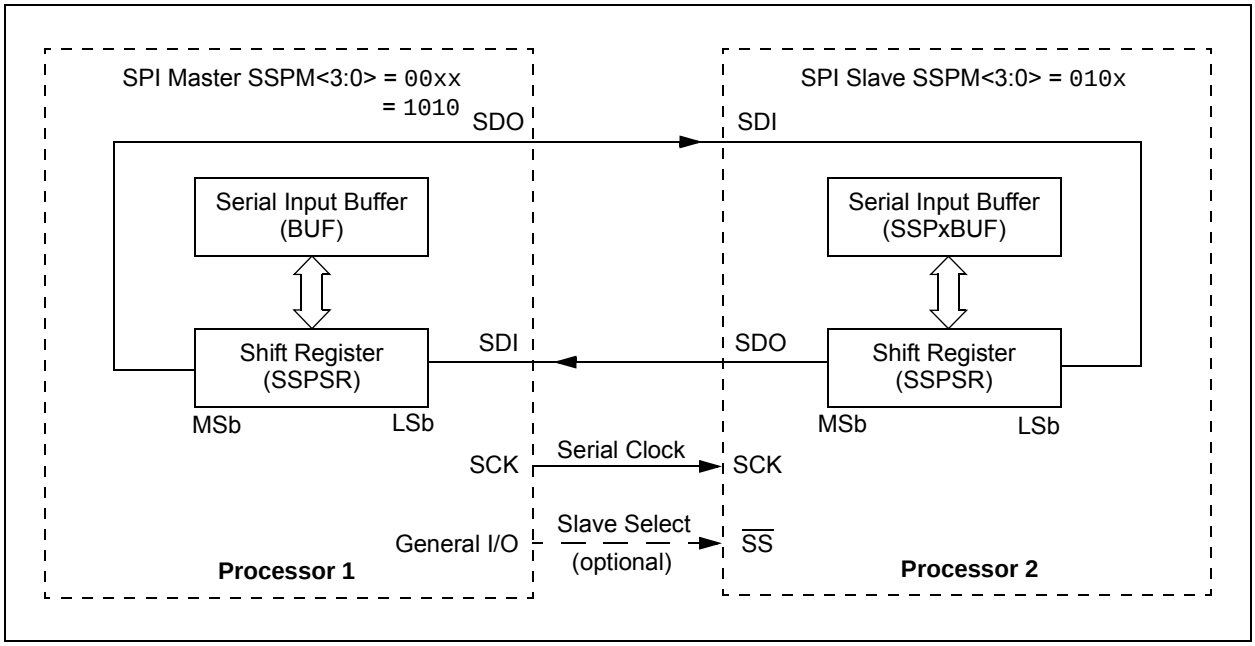
When Level-Triggered Hardware Limit One-Shot modes are used in conjunction with the CCP PWM operation the PWM drive goes active with either the external signal edge or the setting of the ON bit, whichever of the two starts the timer.

The MSSP consists of a transmit/receive shift register (SSPSR) and a buffer register (SSPxBUF). The SSPSR shifts the data in and out of the device, MSb first. The SSPxBUF holds the data that was written to the SSPSR until the received data is ready. Once the eight bits of data have been received, that byte is moved to the SSPxBUF register. Then, the Buffer Full Detect bit, BF of the SSPxSTAT register, and the interrupt flag bit, SSPxIF, are set. This double-buffering of the received data (SSPxBUF) allows the next byte to start reception before reading the data that was just received. Any write to the SSPxBUF register during transmission/reception of data will be ignored and the write collision detect bit WCOL of the SSPxCON1 register, will be set. User software must clear the WCOL bit to allow the following write(s) to the SSPxBUF register to complete successfully.

When the application software is expecting to receive valid data, the SSPxBUF should be read before the next byte of data to transfer is written to the SSPxBUF. The Buffer Full bit, BF of the SSPxSTAT register, indicates when SSPxBUF has been loaded with the received data (transmission is complete). When the SSPxBUF is read, the BF bit is cleared. This data may be irrelevant if the SPI is only a transmitter. Generally, the MSSP interrupt is used to determine when the transmission/reception has completed. If the interrupt method is not going to be used, then software polling can be done to ensure that a write collision does not occur.

The SSPSR is not directly readable or writable and can only be accessed by addressing the SSPxBUF register. Additionally, the SSPxSTAT register indicates the various Status conditions.

FIGURE 24-5: SPI MASTER/SLAVE CONNECTION



24.8 Register Definitions: MSSP Control

REGISTER 24-1: SSP1STAT: SSP STATUS REGISTER

R/W-0/0	R/W-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0
SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF
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 **SMP:** SPI Data Input Sample bit
SPI Master mode:
1 = Input data sampled at end of data output time
0 = Input data sampled at middle of data output time
SPI Slave mode:
SMP must be cleared when SPI is used in Slave mode
In I^2C Master or Slave mode:
1 = Slew rate control disabled for Standard Speed mode (100 kHz and 1 MHz)
0 = Slew rate control enabled for High-Speed mode (400 kHz)

bit 6 **CKE:** SPI Clock Edge Select bit (SPI mode only)
In SPI Master or Slave mode:
1 = Transmit occurs on transition from active to Idle clock state
0 = Transmit occurs on transition from Idle to active clock state
In I^2C™ mode only:
1 = Enable input logic so that thresholds are compliant with SMBus specification
0 = Disable SMBus specific inputs

bit 5 **D/$\overline{A}$:** Data/$\overline{Address}$ bit (I^2C mode only)
1 = Indicates that the last byte received or transmitted was data
0 = Indicates that the last byte received or transmitted was address

bit 4 **P:** Stop bit
(I^2C mode only. This bit is cleared when the MSSP module is disabled, SSPEN is cleared.)
1 = Indicates that a Stop bit has been detected last (this bit is '0' on Reset)
0 = Stop bit was not detected last

bit 3 **S:** Start bit
(I^2C mode only. This bit is cleared when the MSSP module is disabled, SSPEN is cleared.)
1 = Indicates that a Start bit has been detected last (this bit is '0' on Reset)
0 = Start bit was not detected last

bit 2 **R/$\overline{W}$:** Read/$\overline{Write}$ bit information (I^2C mode only)
This bit holds the R/$\overline{W}$ bit information following the last address match. This bit is only valid from the address match to the next Start bit, Stop bit, or not ACK bit.
In I^2C Slave mode:
1 = Read
0 = Write
In I^2C Master mode:
1 = Transmit is in progress
0 = Transmit is not in progress
OR-ing this bit with SEN, RSEN, PEN, RCEN or ACKEN will indicate if the MSSP is in Idle mode.

bit 1 **UA:** Update Address bit (10-bit I^2C mode only)
1 = Indicates that the user needs to update the address in the SSP1ADD register
0 = Address does not need to be updated

bit 0 **BF:** Buffer Full Status bit
Receive (SPI and I^2C modes):
1 = Receive complete, SSP1BUF is full
0 = Receive not complete, SSP1BUF is empty
Transmit (I^2C mode only):
1 = Data transmit in progress (does not include the $\overline{ACK}$ and Stop bits), SSP1BUF is full
0 = Data transmit complete (does not include the $\overline{ACK}$ and Stop bits), SSP1BUF is empty

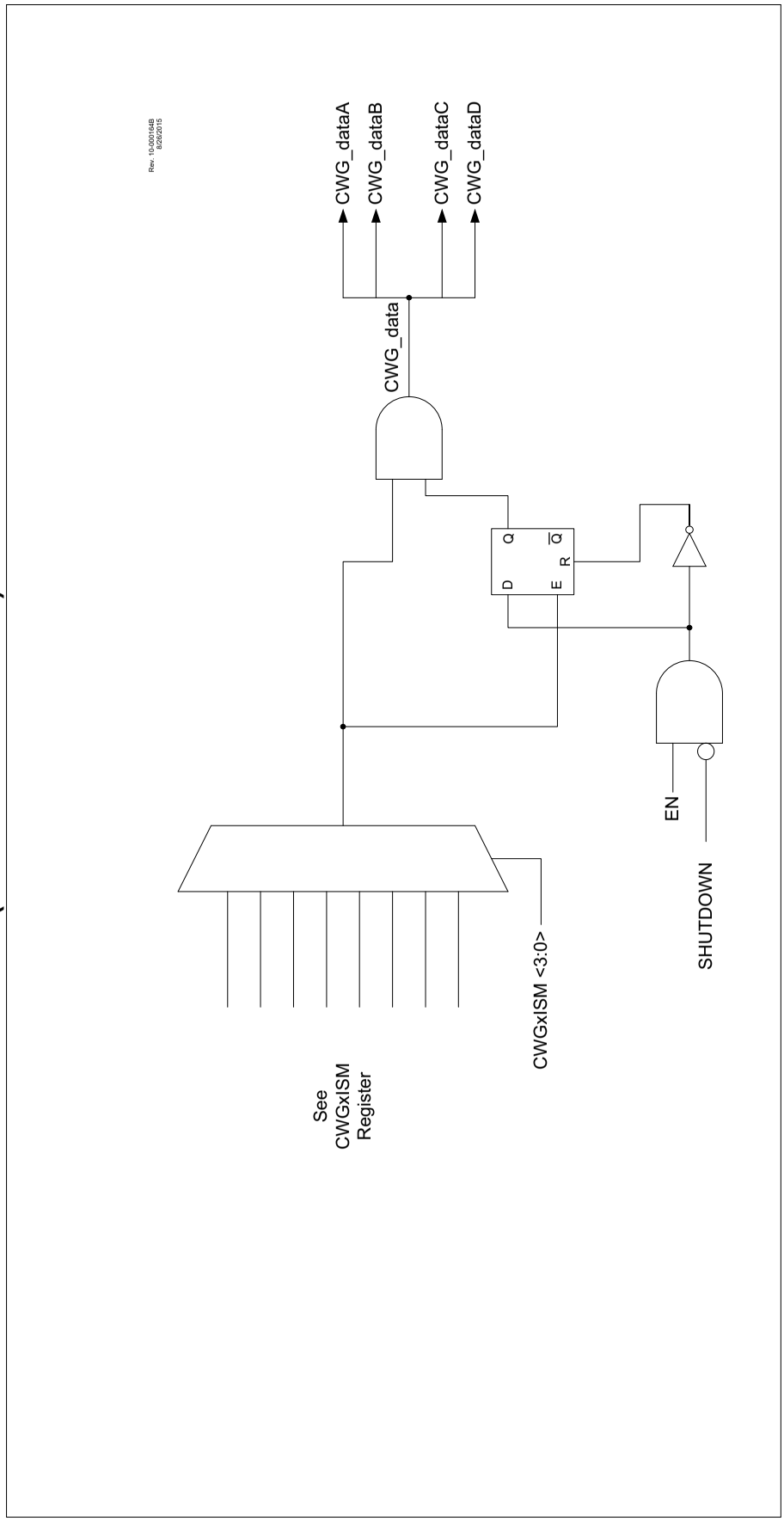


FIGURE 28-4: SIMPLIFIED CWG BLOCK DIAGRAM (OUTPUT STEERING MODES)

28.10 Auto-Shutdown

Auto-shutdown is a method to immediately override the CWG output levels with specific overrides that allow for safe shutdown of the circuit. The shutdown state can be either cleared automatically or held until cleared by software. The auto-shutdown circuit is illustrated in Figure 28-12.

28.10.1 SHUTDOWN

The shutdown state can be entered by either of the following two methods:

- Software generated
- External Input

28.10.1.1 Software Generated Shutdown

Setting the SHUTDOWN bit of the CWGxAS0 register will force the CWG into the shutdown state.

When the auto-restart is disabled, the shutdown state will persist as long as the SHUTDOWN bit is set.

When auto-restart is enabled, the SHUTDOWN bit will clear automatically and resume operation on the next rising edge event.

28.10.2 EXTERNAL INPUT SOURCE

External shutdown inputs provide the fastest way to safely suspend CWG operation in the event of a Fault condition. When any of the selected shutdown inputs goes active, the CWG outputs will immediately go to the selected override levels without software delay. Several input sources can be selected to cause a shutdown condition. All input sources are active-low. The sources are:

- Comparator C1_OUT_sync
- Comparator C2_OUT_sync
- Timer2 – TMR2_postscaled
- Timer4 – TMR4_postscaled
- Timer6 – TMR6_postscaled
- CWGxIN input pin

Shutdown inputs are selected using the CWGxAS1 register (Register 28-6).

Note:	Shutdown inputs are level sensitive, not edge sensitive. The shutdown state cannot be cleared, except by disabling auto-shutdown, as long as the shutdown input level persists.

28.11 Operation During Sleep

The CWG module operates independently from the system clock and will continue to run during Sleep, provided that the clock and input sources selected remain active.

The HFINTOSC remains active during Sleep when all the following conditions are met:

- CWG module is enabled
- Input source is active
- HFINTOSC is selected as the clock source, regardless of the system clock source selected.

In other words, if the HFINTOSC is simultaneously selected as the system clock and the CWG clock source, when the CWG is enabled and the input source is active, then the CPU will go idle during Sleep, but the HFINTOSC will remain active and the CWG will continue to operate. This will have a direct effect on the Sleep mode current.

REGISTER 30-6: SMT2WIN: SMT2 WINDOW INPUT SELECT REGISTER

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	WSEL<4:0>				
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 7-5 **Unimplemented**: Read as ‘0’

bit 4-0 **WSEL<4:0>:** SMT2 Window Selection bits

11111 = Reserved
•
•
•
11000 = Reserved
10111 = MFINTOSC/16
10110 = AT1_perclk
10101 = LFINTOSC
10100 = PWM4_out
10011 = PWM3_out
10010 = Reserved
10001 = SMT1_match
10000 = TMR0_overflow
01111 = TMR5_overflow
01110 = TMR3_overflow
01101 = TMR1_overflow
01100 = LC4_out
01011 = LC3_out
01010 = LC2_out
01001 = LC1_out
01000 = TMR6_postscaled
00111 = TMR4_postscaled
00110 = TMR2_postscaled
00101 = ZCD1_out
00100 = CCP2_out
00011 = CCP1_out
00010 = C2OUT_sync
00001 = C1OUT_sync
00000 = SMTWINx pin

REGISTER 31-19: ATxERRH: ANGULAR TIMER SET POINT ERROR VALUE HIGH REGISTER

R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x
ERR<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 **ERR<15:8>:** Most Significant bits of ATxERR. ATxERR is the error of the measured period value compared to the threshold setting, defined as ATxPER-ATxSTPTP.

REGISTER 31-20: ATxERRL: ANGULAR TIMER SET POINT ERROR VALUE LOW REGISTER

R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x	R-x/x
ERR<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 **ERR<7:0>:** Least Significant bits of ATxERR. ATxERR is the error of the measured period value compared to the threshold setting, defined as ATxPER-ATxSTPTP.

33.0 IN-CIRCUIT SERIAL PROGRAMMING™ (ICSP™)

ICSP™ programming allows customers to manufacture circuit boards with unprogrammed devices. Programming can be done after the assembly process allowing the device to be programmed with the most recent firmware or a custom firmware. Five pins are needed for ICSP™ programming:

- ICSPCLK
- ICSPDAT
- $\overline{MCLR}/V_{PP}$
- V_{DD}
- V_{SS}

In Program/Verify mode the program memory, user IDs and the Configuration Words are programmed through serial communications. The ICSPDAT pin is a bidirectional I/O used for transferring the serial data and the ICSPCLK pin is the clock input. For more information on ICSP™ refer to the "*PIC12(L)F1612/PIC16(L)F161X Memory Programming Specification*" (DS40001720).

33.1 High-Voltage Programming Entry Mode

The device is placed into High-Voltage Programming Entry mode by holding the ICSPCLK and ICSPDAT pins low then raising the voltage on $\overline{MCLR}/V_{PP}$ to V_{IHH}.

33.2 Low-Voltage Programming Entry Mode

The Low-Voltage Programming Entry mode allows the PIC® Flash MCUs to be programmed using V_{DD} only, without high voltage. When the LVP bit of Configuration Words is set to '1', the ICSP Low-Voltage Programming Entry mode is enabled. To disable the Low-Voltage ICSP mode, the LVP bit must be programmed to '0'.

Entry into the Low-Voltage Programming Entry mode requires the following steps:

1. $\overline{MCLR}$ is brought to V_{IL}.
2. A 32-bit key sequence is presented on ICSPDAT, while clocking ICSPCLK.

Once the key sequence is complete, $\overline{MCLR}$ must be held at V_{IL} for as long as Program/Verify mode is to be maintained.

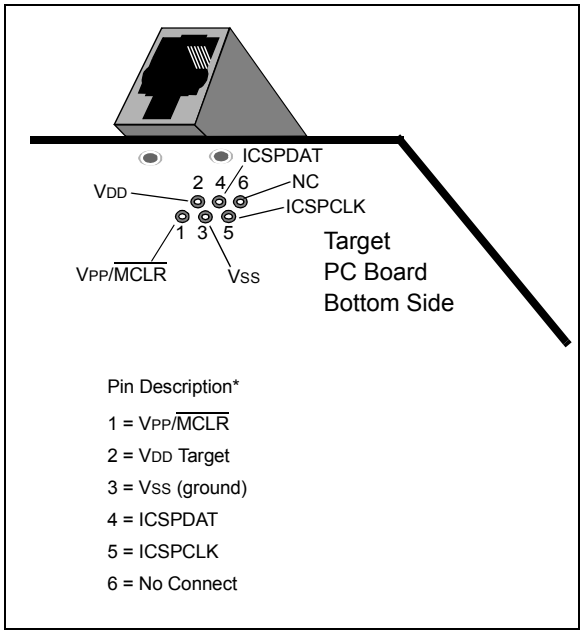
If low-voltage programming is enabled (LVP = 1), the $\overline{MCLR}$ Reset function is automatically enabled and cannot be disabled. See **Section6.5 "MCLR"** for more information.

The LVP bit can only be reprogrammed to '0' by using the High-Voltage Programming mode.

33.3 Common Programming Interfaces

Connection to a target device is typically done through an ICSP™ header. A commonly found connector on development tools is the RJ-11 in the 6P6C (6-pin, 6-connector) configuration. See Figure 33-1.

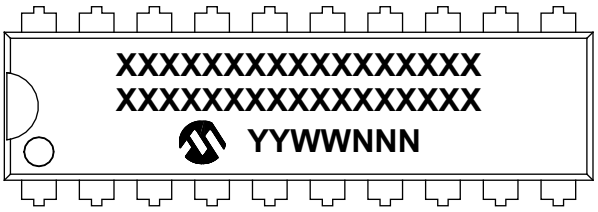
FIGURE 33-1: ICD RJ-11 STYLE CONNECTOR INTERFACE



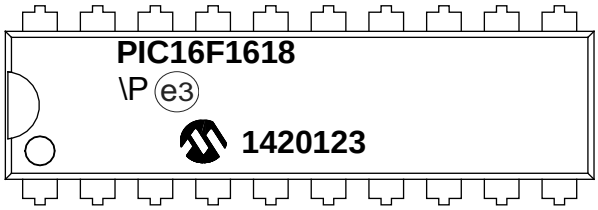
Another connector often found in use with the PICkit™ programmers is a standard 6-pin header with 0.1 inch spacing. Refer to Figure 33-2.

38.1 Package Marking Information (Continued)

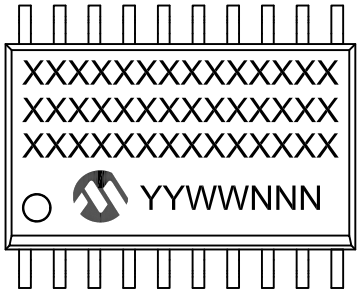
20-Lead PDIP (300 mil)



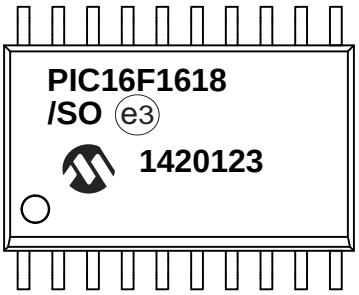
Example



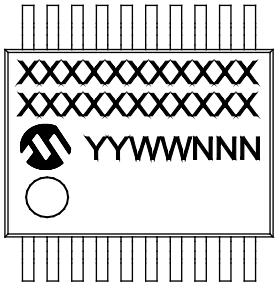
20-Lead SOIC (7.50 mm)



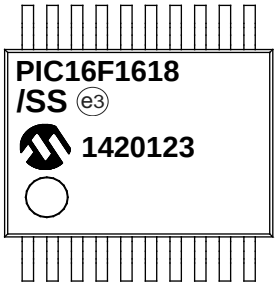
Example



20-Lead SSOP (5.30 mm)

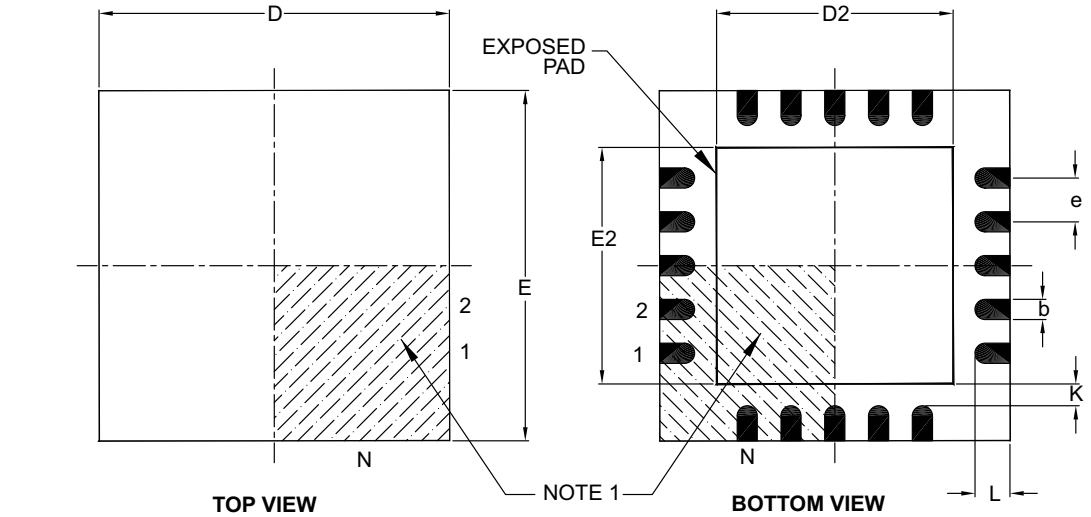


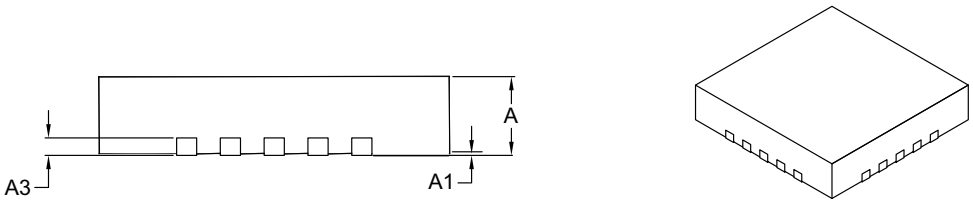
Example



20-Lead Plastic Quad Flat, No Lead Package (ML) – 4x4x0.9 mm 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		20	
Pitch	e		0.50 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	2.60	2.70	2.80
Overall Length	D		4.00 BSC	
Exposed Pad Length	D2	2.60	2.70	2.80
Contact Width	b	0.18	0.25	0.30
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 C04-126B