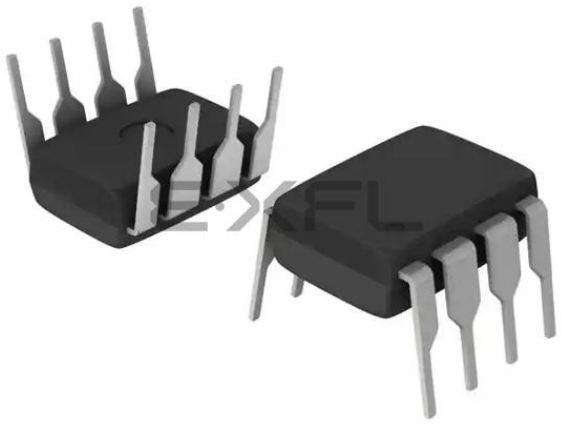


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

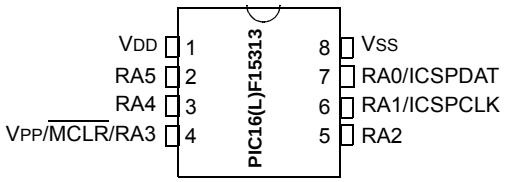
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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	6
Program Memory Size	3.5KB (2K x 14)
Program Memory Type	FLASH
EEPROM Size	224 x 8
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 5x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	8-DIP (0.300", 7.62mm)
Supplier Device Package	8-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf15313-i-p

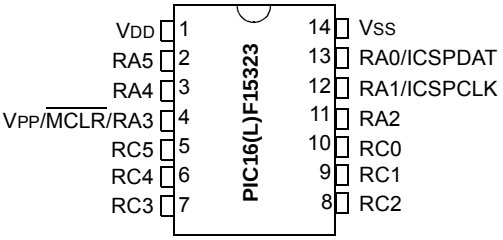
PIN DIAGRAMS

8-PIN PDIP, SOIC, MSOP



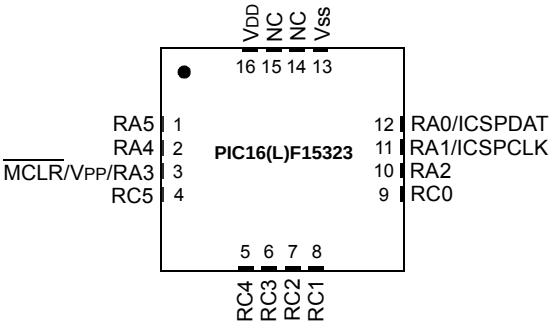
Note: See Table 3 for location of all peripheral functions.

14-PIN PDIP, SOIC, TSSOP



Note: See Table 4 for location of all peripheral functions.

16-PIN UQFN (4X4)



Note 1: See Table 4 for location of all peripheral functions.

2: It is recommended that the exposed bottom pad be connected to VSS.

2.5 External Oscillator Pins

Many microcontrollers have options for at least two oscillators: a high-frequency primary oscillator and a low-frequency secondary oscillator (refer to **Section 9.0 "Oscillator Module (with Fail-Safe Clock Monitor)"** for details). The PIC16(L)F15313/23 devices do not have a secondary oscillator.

The oscillator circuit should be placed on the same side of the board as the device. Place the oscillator circuit close to the respective oscillator pins with no more than 0.5 inch (12 mm) between the circuit components and the pins. The load capacitors should be placed next to the oscillator itself, on the same side of the board.

Use a grounded copper pour around the oscillator circuit to isolate it from surrounding circuits. The grounded copper pour should be routed directly to the MCU ground. Do not run any signal traces or power traces inside the ground pour. Also, if using a two-sided board, avoid any traces on the other side of the board where the crystal is placed.

Layout suggestions are shown in Figure 2-3. In-line packages may be handled with a single-sided layout that completely encompasses the oscillator pins. With fine-pitch packages, it is not always possible to completely surround the pins and components. A suitable solution is to tie the broken guard sections to a mirrored ground layer. In all cases, the guard trace(s) must be returned to ground.

In planning the application's routing and I/O assignments, ensure that adjacent port pins, and other signals in close proximity to the oscillator, are benign (i.e., free of high frequencies, short rise and fall times, and other similar noise).

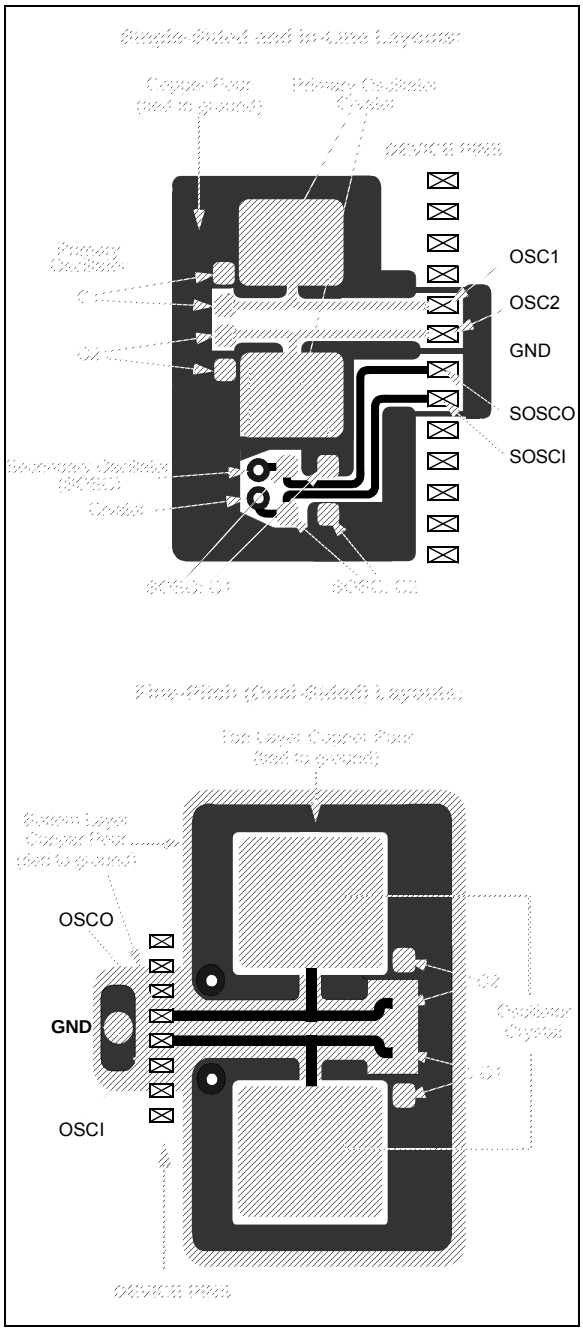
For additional information and design guidance on oscillator circuits, refer to these Microchip Application Notes, available at the corporate website (www.microchip.com):

- *AN826,* "*Crystal Oscillator Basics and Crystal Selection for rfPIC™ and PICmicro® Devices"*
- *AN849, "Basic PICmicro® Oscillator Design"*
- *AN943, "Practical PICmicro® Oscillator Analysis and Design"*
- *AN949, "Making Your Oscillator Work"*

2.6 Unused I/Os

Unused I/O pins should be configured as outputs and driven to a logic low state. Alternatively, connect a 1 kΩ to 10 kΩ resistor to VSS on unused pins and drive the output to logic low.

FIGURE 2-3: SUGGESTED PLACEMENT OF THE OSCILLATOR CIRCUIT



3.0 ENHANCED MID-RANGE CPU

This family of devices contains an enhanced mid-range 8-bit CPU core. The CPU has 48 instructions. Interrupt capability includes automatic context saving.

The hardware stack is 16-levels deep and has Overflow and Underflow Reset capability. Direct, Indirect, and Relative Addressing modes are available. Two File Select Registers (FSRs) provide the ability to read program and data memory.

FIGURE 3-1: CORE DATA PATH DIAGRAM

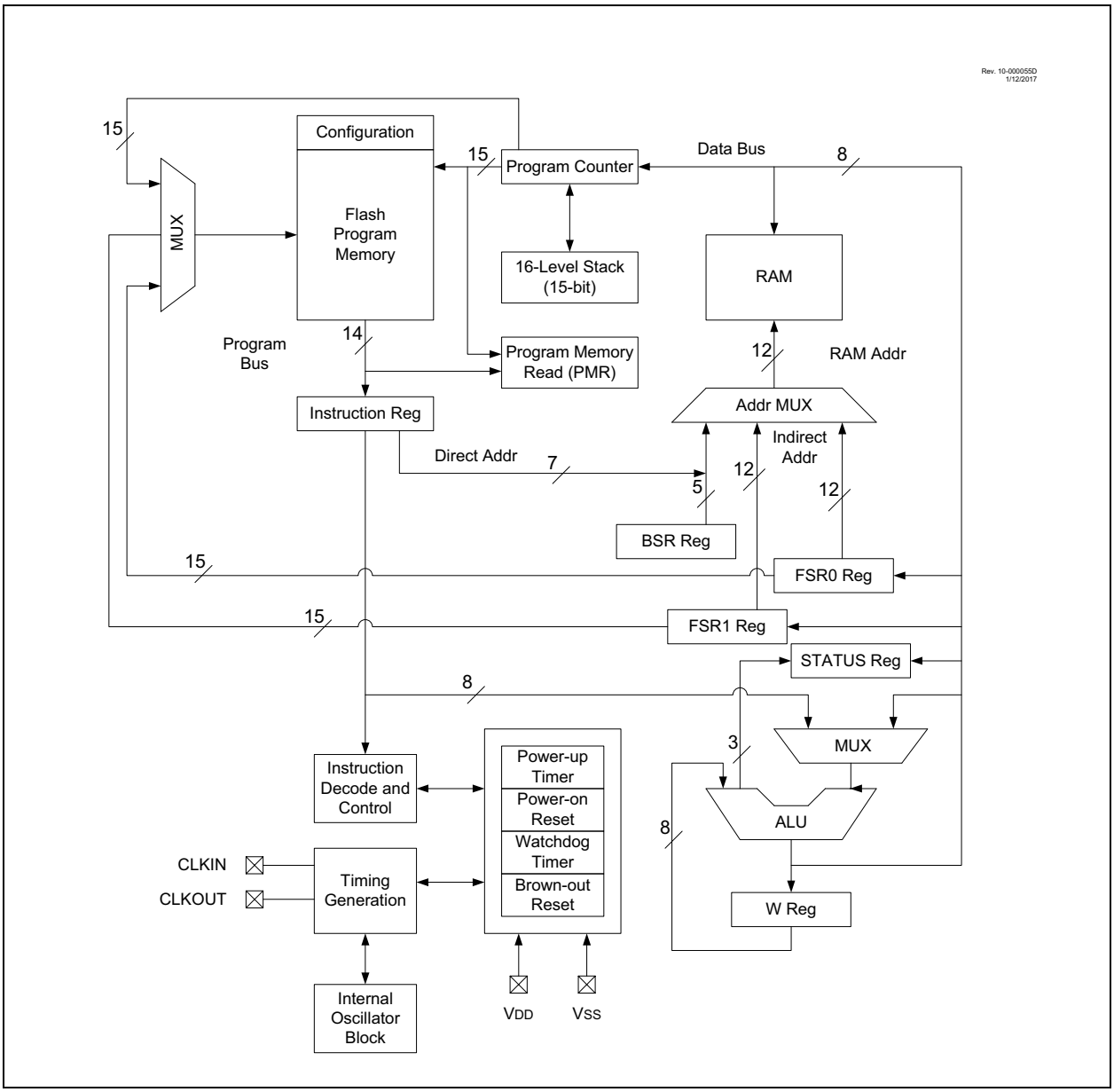


TABLE 4-10: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-63 (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on: MCLR
Bank 63											
CPU CORE REGISTERS; see Table 4-3 for specifics											
1F8Ch — 1FE3h	—	Unimplemented								—	—
1FE4h	STATUS_SHAD	—	—	—	—	—	Z	DC	C	---- -xxx	---- -uuu
1FE5h	WREG_SHAD	Working Register Shadow								xxxx xxxx	uuuu uuuu
1FE6h	BSR_SHAD	—	—	—	Bank Select Register Shadow					---x xxxx	---u uuuu
1FE7h	PCLATH_SHAD	—	Program Counter Latch High Register Shadow							-xxx xxxx	uuuu uuuu
1FE8h	FSR0L_SHAD	Indirect Data Memory Address 0 Low Pointer Shadow								xxxx xxxx	uuuu uuuu
1FE9h	FSR0H_SHAD	Indirect Data Memory Address 0 High Pointer Shadow								xxxx xxxx	uuuu uuuu
1FEAh	FSR1L_SHAD	Indirect Data Memory Address 1 Low Pointer Shadow								xxxx xxxx	uuuu uuuu
1FEBh	FSR1H_SHAD	Indirect Data Memory Address 1 High Pointer Shadow								xxxx xxxx	uuuu uuuu
1FECh	—	Unimplemented								—	—
1FEDh	STKPTR	—	—	—	Current Stack Pointer					---1 1111	---1 1111
1FEEh	TOSL	Top of Stack Low byte								xxxx xxxx	uuuu uuuu
1FEFh	TOSH	—	Top of Stack High byte							-xxx xxxx	-uuu uuuu

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

REGISTER 5-4: CONFIGURATION WORD 4: MEMORY

R/W-1	U-1	R/W-1	U-1	R/W-1	R/W-1
LVP	—	$\overline{\text{WRTSAF}}$(1)	—	$\overline{\text{WRTC}}$(1)	$\overline{\text{WRTB}}$(1)
bit 13	12	11	10	9	bit 8

R/W-1	U-1	U-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{WRTAPP}}$(1)	—	—	$\overline{\text{SAFEN}}$(1)	$\overline{\text{BBEN}}$(1)	BBSIZE2	BBSIZE1	BBSIZE0
bit 7	6	5	4	3	2	1	bit 0

Legend:			
R = Readable bit	P = Programmable bit	x = Bit is unknown	U = Unimplemented bit, read as '1'
'0' = Bit is cleared	'1' = Bit is set	W = Writable bit	n = Value when blank or after Bulk Erase

bit 13 **LVP:** Low Voltage Programming Enable bit
1 = Low voltage programming enabled. $\overline{\text{MCLR}}$/VPP pin function is $\overline{\text{MCLR}}$. MCLRE Configuration bit is ignored.
0 = HV on $\overline{\text{MCLR}}$/VPP must be used for programming.
The LVP bit cannot be written (to zero) while operating from the LVP programming interface. The purpose of this rule is to prevent the user from dropping out of LVP mode while programming from LVP mode, or accidentally eliminating LVP mode from the configuration state.
The preconditioned (erased) state for this bit is critical.

bit 12 **Unimplemented:** Read as '1'

bit 11 **WRTSAF:** Storage Area Flash Write Protection bit
1 = SAF NOT write-protected
0 = SAF write-protected
Unimplemented, if SAF is not supported in the device family and only applicable if $\overline{\text{SAFEN}}$ = 0.

bit 10 **Unimplemented:** Read as '1'

bit 9 **WRTC:** Configuration Register Write Protection bit
1 = Configuration Register NOT write-protected
0 = Configuration Register write-protected

bit 8 **WRTB:** Boot Block Write Protection bit
1 = Boot Block NOT write-protected
0 = Boot Block write-protected
Only applicable if $\overline{\text{BBEN}}$ = 0.

bit 7 **WRTAPP:** Application Block Write Protection bit
1 = Application Block NOT write-protected
0 = Application Block write-protected

bit 6-5 **Unimplemented:** Read as '1'

bit 4 **SAFEN:** SAF Enable bit
1 = SAF disabled
0 = SAF enabled

bit 3 **BBEN:** Boot Block Enable bit
1 = Boot Block disabled
0 = Boot Block enabled

bit 2-0 **BBSIZE<2:0>:** Boot Block Size Selection bits (See Table 5-1)
BBSIZE is used only when $\overline{\text{BBEN}}$ = 0
BBSIZ bits can only be written while $\overline{\text{BBEN}}$ = 1; after $\overline{\text{BBEN}}$ = 0, BBSIZ is write-protected.

Note 1: Bits are implemented as sticky bits. Once protection is enabled, it can only be reset through a Bulk Erase.

REGISTER 10-13: PIR3: PERIPHERAL INTERRUPT REQUEST REGISTER 3

U-0	U-0	R/W/HS-0/0	R/W/HS-0/0	U-0	U-0	R/W/HS-0/0	R/W/HS-0/0
—	—	RC1IF	TX1IF	—	—	BCL1IF	SSP1IF
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	HS = Hardware clearable

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

bit 5 **RC1IF:** EUSART1 Receive Interrupt Flag (read-only) bit **(1)**

1 = The EUSART1 receive buffer is not empty (contains at least one byte)
0 = The EUSART1 receive buffer is empty

bit 4 **TX1IF:** EUSART1 Transmit Interrupt Flag (read-only) bit**(2)**

1 = The EUSART1 transmit buffer contains at least one unoccupied space
0 = The EUSART1 transmit buffer is currently full. The application firmware should not write to TXxREG again, until more room becomes available in the transmit buffer.

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

bit 1 **BCL1IF:** MSSP1 Bus Collision Interrupt Flag bit

1 = A bus collision was detected (must be cleared in software)
0 = No bus collision was detected

bit 0 **SSP1IF:** Synchronous Serial Port (MSSP1) Interrupt Flag bit

1 = The Transmission/Reception/Bus Condition is complete (must be cleared in software)
0 = Waiting for the Transmission/Reception/Bus Condition in progress

Note 1: The RCxIF flag is a read-only bit. To clear the RCxIF flag, the firmware must read from RCxREG enough times to remove all bytes from the receive buffer.

2: The TXxIF flag is a read-only bit, indicating if there is room in the transmit buffer. To clear the TX1IF flag, the firmware must write enough data to TXxREG to completely fill all available bytes in the buffer. The TXxIF flag does not indicate transmit completion (use TRMT for this purpose instead).

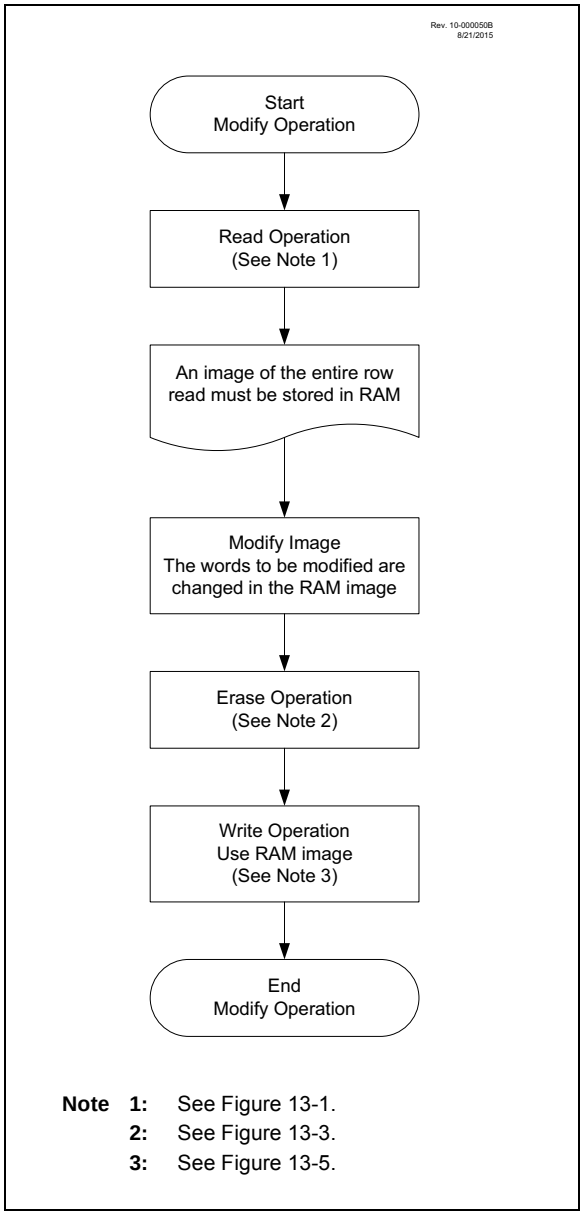
Note: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

13.3.5 MODIFYING FLASH PROGRAM MEMORY

When modifying existing data in a program memory row, and data within that row must be preserved, it must first be read and saved in a RAM image. Program memory is modified using the following steps:

1. Load the starting address of the row to be modified.
2. Read the existing data from the row into a RAM image.
3. Modify the RAM image to contain the new data to be written into program memory.
4. Load the starting address of the row to be rewritten.
5. Erase the program memory row.
6. Load the write latches with data from the RAM image.
7. Initiate a programming operation.

FIGURE 13-6: FLASH PROGRAM MEMORY MODIFY FLOWCHART

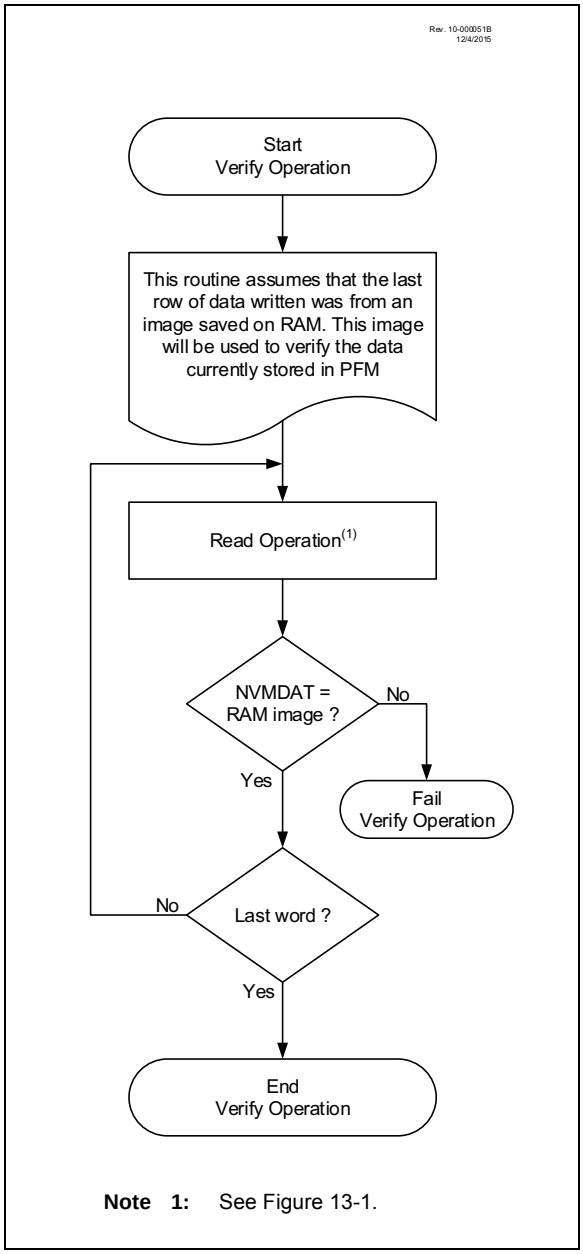


Note 1: See Figure 13-1.
2: See Figure 13-3.
3: See Figure 13-5.

13.3.7 WRITE VERIFY

It is considered good programming practice to verify that program memory writes agree with the intended value. Since program memory is stored as a full row then the stored program memory contents are compared with the intended data stored in RAM after the last write is complete.

FIGURE 13-7: FLASH PROGRAM MEMORY VERIFY FLOWCHART



Note 1: See Figure 13-1.

14.5 Register Definitions: PORTC

REGISTER 14-9: PORTC: PORTC REGISTER

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
—	—	RC5	RC4	RC3	RC2	RC1	RC0
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 **RC<5:0>**: PORTC General Purpose I/O Pin bits(1)
1 = Port pin is ≥ VIH
0 = Port pin is ≤ VIL

Note 1: Writes to PORTC are actually written to corresponding LATC register. The actual I/O pin values are read from the PORTC register.

REGISTER 14-10: TRISC: PORTC TRI-STATE REGISTER

U-0	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
—	—	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
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 **TRISC<5:0>:** PORTC Tri-State Control bits
1 = PORTC pin configured as an input (tri-stated)
0 = PORTC pin configured as an output

REGISTER 14-11: LATC: PORTC DATA LATCH REGISTER

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
—	—	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0
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 **LATC<5:0>**: PORTC Output Latch Value bits((1)

Note 1: Writes to PORTC are actually written to corresponding LATC register. Reads from PORTC register returns actual I/O pin values.

REGISTER 16-4: PMD3: PMD CONTROL 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
—	—	PWM6MD	PWM5MD	PWM4MD	PWM3MD	CCP2MD	CCP1MD
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 7-6 **Unimplemented:** Read as '0'

bit 5 **PWM6MD:** Disable Pulse-Width Modulator PWM6 bit
1 = PWM6 module disabled
0 = PWM6 module enabled

bit 4 **PWM5MD:** Disable Pulse-Width Modulator PWM5 bit
1 = PWM5 module disabled
0 = PWM5 module enabled

bit 3 **PWM4MD:** Disable Pulse-Width Modulator PWM4 bit
1 = PWM4 module disabled
0 = PWM4 module enabled

bit 2 **PWM3MD:** Disable Pulse-Width Modulator PWM3 bit
1 = PWM3 module disabled
0 = PWM3 module enabled

bit 1 **CCP2MD:** Disable CCP2 bit
1 = CCP2 module disabled
0 = CCP2 module enabled

bit 0 **CCP1MD:** Disable CCP1 bit
1 = CCP1 module disabled
0 = CCP1 module enabled

18.3 Register Definitions: FVR Control

REGISTER 18-1: FVRCON: FIXED VOLTAGE REFERENCE CONTROL REGISTER

R/W-0/0	R-q/q	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
FVREN	FVRRDY(1)	TSEN(3)	TSRNG(3)	CDAFVR<1:0>		ADFVR<1:0>	
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 **FVREN:** Fixed Voltage Reference Enable bit
1 = Fixed Voltage Reference is enabled
0 = Fixed Voltage Reference is disabled

bit 6 **FVRRDY:** Fixed Voltage Reference Ready Flag bit(1)
1 = Fixed Voltage Reference output is ready for use
0 = Fixed Voltage Reference output is not ready or not enabled

bit 5 **TSEN:** Temperature Indicator Enable bit(3)
1 = Temperature Indicator is enabled
0 = Temperature Indicator is disabled

bit 4 **TSRNG:** Temperature Indicator Range Selection bit(3)
1 = Temperature in High Range VOUT = 3VT
0 = Temperature in Low Range VOUT = 2VT

bit 3-2 **CDAFVR<1:0>:** Comparator FVR Buffer Gain Selection bits
11 = Comparator FVR Buffer Gain is 4x, (4.096V)(2)
10 = Comparator FVR Buffer Gain is 2x, (2.048V)(2)
01 = Comparator FVR Buffer Gain is 1x, (1.024V)
00 = Comparator FVR Buffer is off

bit 1-0 **ADFVR<1:0>:** ADC FVR Buffer Gain Selection bit
11 = ADC FVR Buffer Gain is 4x, (4.096V)(2)
10 = ADC FVR Buffer Gain is 2x, (2.048V)(2)
01 = ADC FVR Buffer Gain is 1x, (1.024V)
00 = ADC FVR Buffer is off

Note 1: FVRRDY is always '1'.
2: Fixed Voltage Reference output cannot exceed VDD.
3: See **Section 19.0 "Temperature Indicator Module"** for additional information.

REGISTER 23-5: CMOUT: COMPARATOR OUTPUT REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	R-0/0	R-0/0
—	—	—	—	—	—	MC2OUT	MC1OUT
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 7-2 **Unimplemented:** Read as ‘0’

bit 1 **MC2OUT:** Mirror Copy of C2OUT bit

bit 0 **MC1OUT:** Mirror Copy of C1OUT bit

TABLE 23-3: SUMMARY OF REGISTERS ASSOCIATED WITH COMPARATOR MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
CMxCON0	ON	OUT	—	POL	—	—	HYS	SYNC	250
CMxCON1	—	—	—	—	—	—	INTP	INTN	251
CMOUT	—	—	—	—	—	—	MC2OUT	MC1OUT	253
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>		ADFVR<1:0>		210
DAC1CON0	DAC1EN	—	DAC1OE1	DAC1OE2	DAC1PSS<1:0>		—	DAC1NSS	232
DAC1CON1	—	—	—	DAC1R<4:0>					232
INTCON	GIE	PEIE	—					INTEDG	121
PIE2	—	ZCDIE	—	—	—	—	C2IE	C1IE	124
PIR2	—	ZCDIF	—	—	—	—	C2IF	C1IF	132
RxyPPS	—	—	—	RxyPPS<4:0>					192
CLCINxPPS	—	—	CLCIN0PPS<5:0>						191
T1GPPS	—	—	T1GPPS<5:0>						191

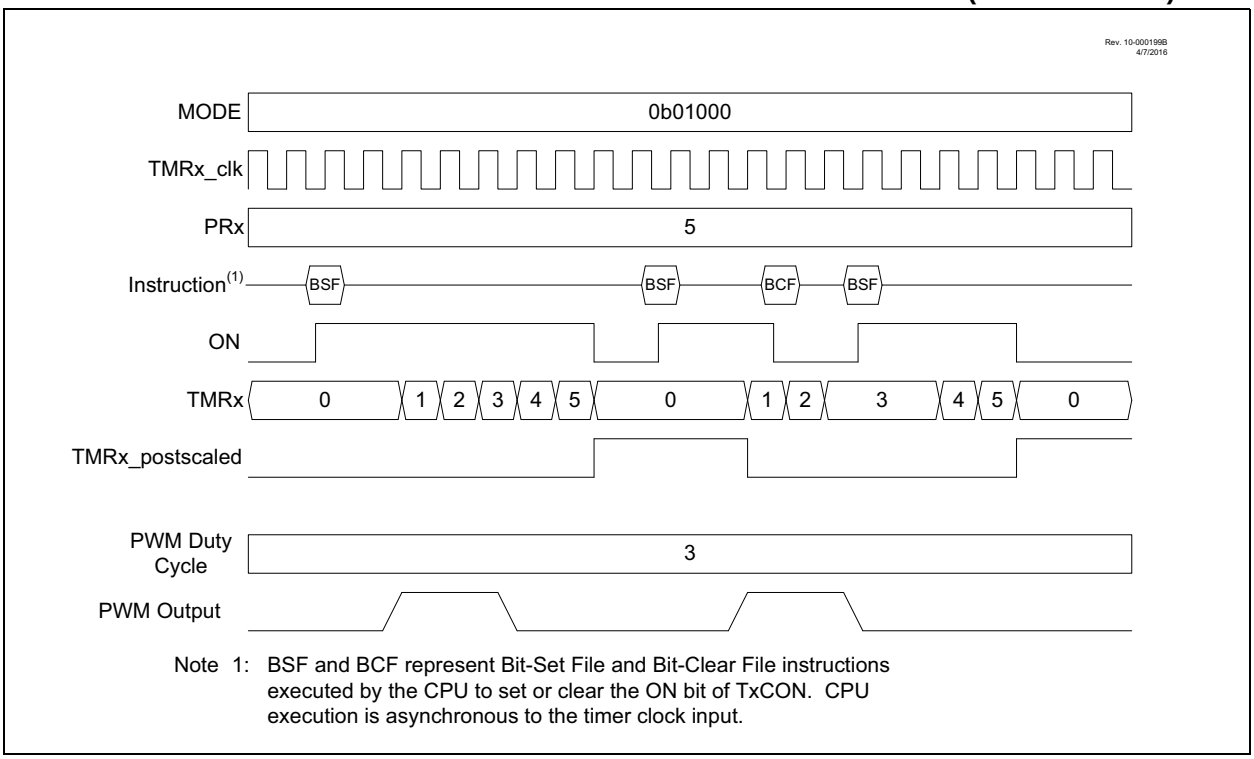
Legend: — = unimplemented location, read as ‘0’. Shaded cells are unused by the comparator module.

27.5.5 SOFTWARE START ONE-SHOT MODE

In One-Shot mode the timer resets and the ON bit is cleared when the timer value matches the PRx period value. The ON bit must be set by software to start another timer cycle. Setting MODE<4:0> = 01000 selects One-Shot mode which is illustrated in Figure 27-8. In the example, ON is controlled by BSF and BCF instructions. In the first case, a BSF instruction sets ON and the counter runs to completion and clears ON. In the second case, a BSF instruction starts the cycle, BCF/BSF instructions turn the counter off and on during the cycle, and then it runs to completion.

When One-Shot mode is used in conjunction with the CCP PWM operation the PWM pulse drive starts concurrent with setting the ON bit. Clearing the ON bit while the PWM drive is active will extend the PWM drive. The PWM drive will terminate when the timer value matches the CCPRx pulse width value. The PWM drive will remain off until software sets the ON bit to start another cycle. If software clears the ON bit after the CCPRx match but before the PRx match then the PWM drive will be extended by the length of time the ON bit remains cleared. Another timing cycle can only be initiated by setting the ON bit after it has been cleared by a PRx period count match.

FIGURE 27-8: SOFTWARE START ONE-SHOT MODE TIMING DIAGRAM (MODE = 01000)



Note 1: BSF and BCF represent Bit-Set File and Bit-Clear File instructions executed by the CPU to set or clear the ON bit of TxCON. CPU execution is asynchronous to the timer clock input.

29.0 PULSE-WIDTH MODULATION (PWM)

The PWMx modules generate Pulse-Width Modulated (PWM) signals of varying frequency and duty cycle.

In addition to the CCP modules, the PIC16(L)F15313/23 devices contain four 10-bit PWM modules (PWM3, PWM4, PWM5 and PWM6). The PWM modules reproduce the PWM capability of the CCP modules.

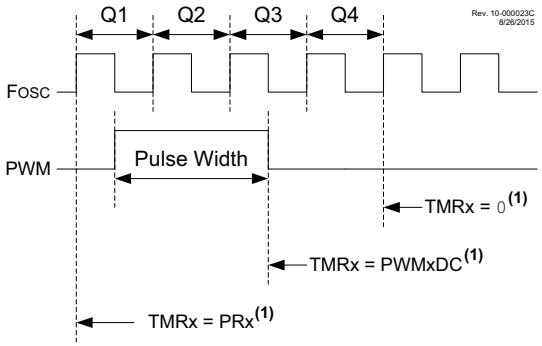
Note:	The PWM3/4/5/6 modules are four instances of the same PWM module design. Throughout this section, the lower case 'x' in register and bit names is a generic reference to the PWM module number (which should be substituted with 3, or 4, or, 5 or 6 during code development). For example, the control register is generically described in this chapter as PWMxCON, but the actual device registers are PWM3CON, PWM4CON, PWM5CON and PWM6CON. Similarly, the PWMxEN bit represents the PWM3EN, PWM4EN, PWM5EN and PWM6EN bits.

Pulse-Width Modulation (PWM) is a scheme that provides power to a load by switching quickly between fully on and fully off states. The PWM signal resembles a square wave where the high portion of the signal is considered the 'on' state (pulse width), and the low portion of the signal is considered the 'off' state. The term duty cycle describes the proportion of the 'on' time to the 'off' time and is expressed in percentages, where 0% is fully off and 100% is fully on. A lower duty cycle corresponds to less power applied and a higher duty cycle corresponds to more power applied. The PWM period is defined as the duration of one complete cycle or the total amount of on and off time combined.

PWM resolution defines the maximum number of steps that can be present in a single PWM period. A higher resolution allows for more precise control of the pulse width time and, in turn, the power that is applied to the load.

Figure 29-1 shows a typical waveform of the PWM signal.

FIGURE 29-1: PWM OUTPUT



Note 1: Timer dependent on PWMTMRS register settings.

FIGURE 30-6: DEAD-BAND OPERATION CWG1DBR = 0X01, CWG1DBF = 0X02

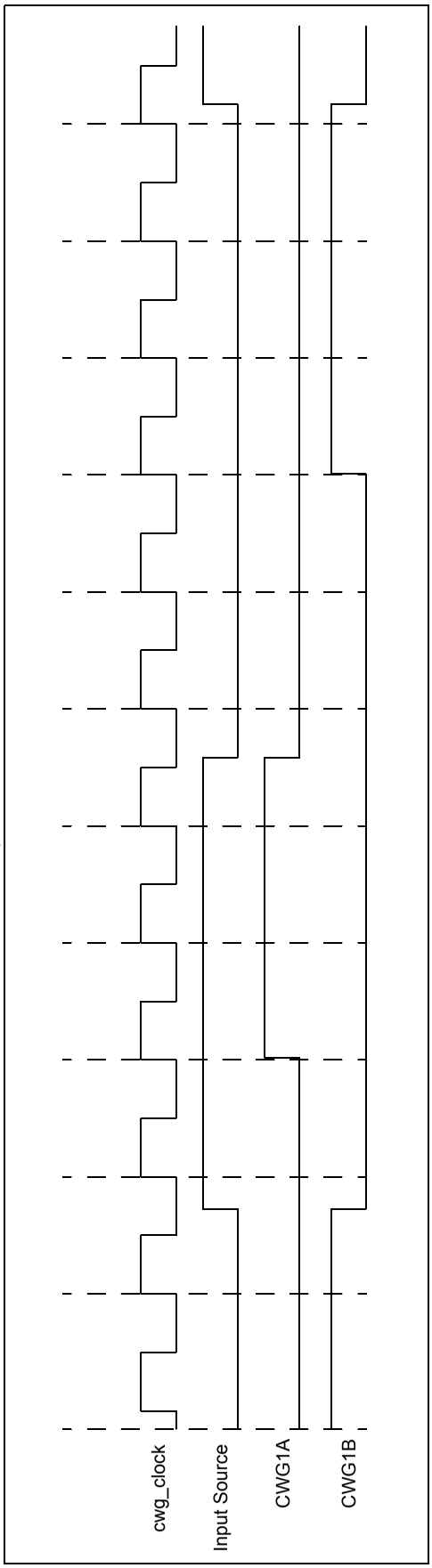
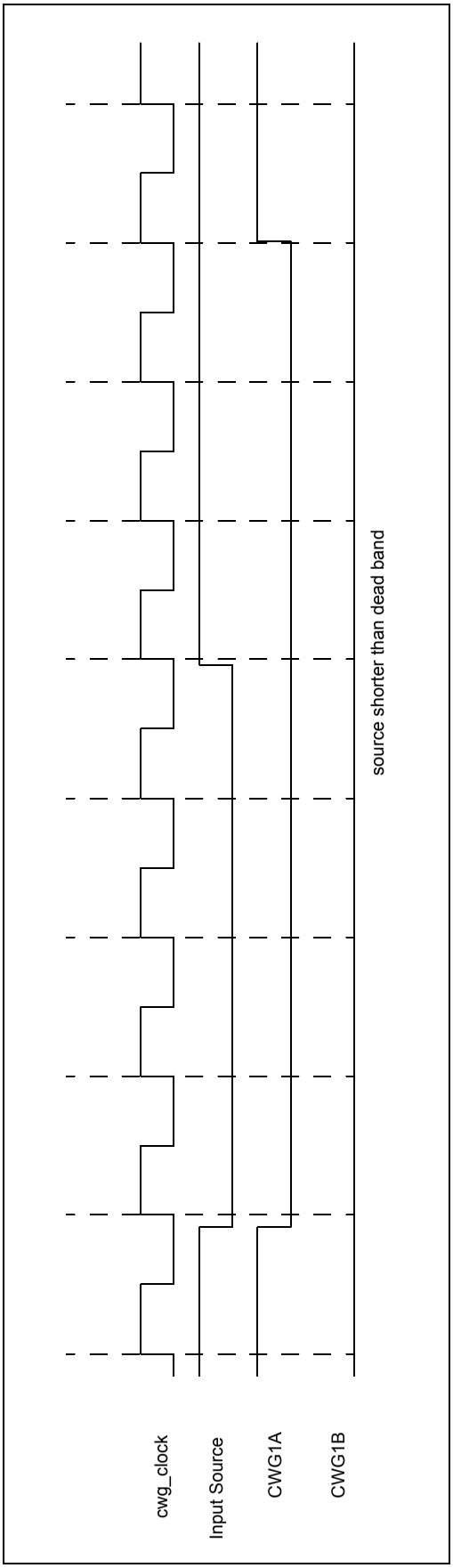


FIGURE 30-7: DEAD-BAND OPERATION, CWG1DBR = 0X03, CWG1DBF = 0X04, SOURCE SHORTER THAN DEAD BAND



30.8 Dead-Band Uncertainty

When the rising and falling edges of the input source are asynchronous to the CWG clock, it creates uncertainty in the dead-band time delay. The maximum uncertainty is equal to one CWG clock period. Refer to Equation 30-1 for more details.

EQUATION 30-1: DEAD-BAND UNCERTAINTY

$$T_{DEADBAND\_UNCERTAINTY} = \frac{1}{Fcwg\_clock}$$

Example:

$FCWG\_CLOCK = 16\ MHz$

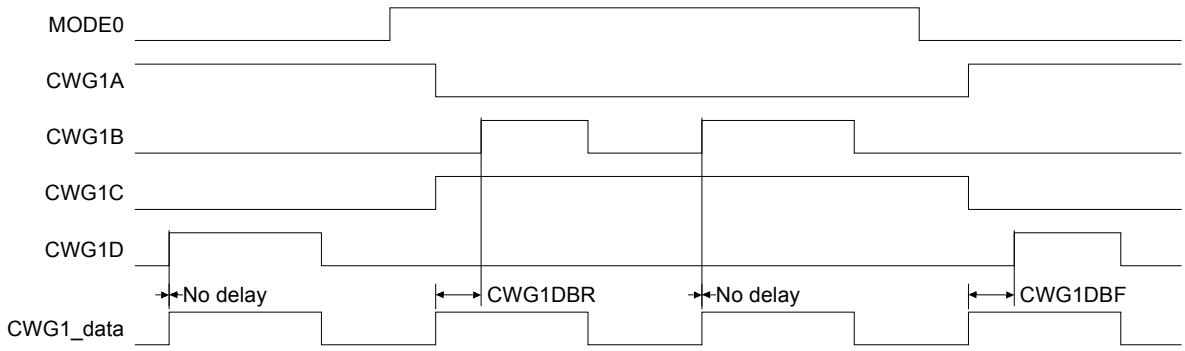
Therefore:

$$T_{DEADBAND\_UNCERTAINTY} = \frac{1}{Fcwg\_clock}$$

$$= \frac{1}{16MHz}$$

$$= 62.5ns$$

FIGURE 30-8: EXAMPLE OF PWM DIRECTION CHANGE



Note 1: WGPOL{ABCD} = 0

2: The direction bit MODE<0> (Register 30-1) can be written any time during the PWM cycle, and takes effect at the next rising CWG1_data.

3: When changing directions, CWG1A and CWG1C switch at rising CWG1_data; modulated CWG1B and CWG1D are held inactive for the dead band duration shown; dead band affects only the first pulse after the direction change.

31.2 CLCx Interrupts

An interrupt will be generated upon a change in the output value of the CLCx when the appropriate interrupt enables are set. A rising edge detector and a falling edge detector are present in each CLC for this purpose.

The CLCxIF bit of the associated PIR5 register will be set when either edge detector is triggered and its associated enable bit is set. The LCxINTP enables rising edge interrupts and the LCxINTN bit enables falling edge interrupts. Both are located in the CLCxCON register.

To fully enable the interrupt, set the following bits:

- CLCxIE bit of the PIE5 register
- LCxINTP bit of the CLCxCON register (for a rising edge detection)
- LCxINTN bit of the CLCxCON register (for a falling edge detection)
- PEIE and GIE bits of the INTCON register

The CLCxIF bit of the PIR5 register, must be cleared in software as part of the interrupt service. If another edge is detected while this flag is being cleared, the flag will still be set at the end of the sequence.

31.3 Output Mirror Copies

Mirror copies of all LCxCON output bits are contained in the CLCxDATA register. Reading this register reads the outputs of all CLCs simultaneously. This prevents any reading skew introduced by testing or reading the LCxOUT bits in the individual CLCxCON registers.

31.4 Effects of a Reset

The CLCxCON register is cleared to zero as the result of a Reset. All other selection and gating values remain unchanged.

31.5 Operation During Sleep

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

The HFINTOSC remains active during Sleep when the CLC module is enabled and the HFINTOSC is selected as an input source, regardless of the system clock source selected.

In other words, if the HFINTOSC is simultaneously selected as the system clock and as a CLC input source, when the CLC is enabled, the CPU will go idle during Sleep, but the CLC will continue to operate and the HFINTOSC will remain active.

This will have a direct effect on the Sleep mode current.

31.6 CLCx Setup Steps

The following steps should be followed when setting up the CLCx:

- Disable CLCx by clearing the LCxEN bit.
- Select desired inputs using CLCxSEL0 through CLCxSEL3 registers (See Table 31-2).
- Clear any associated ANSEL bits.
- Enable the chosen inputs through the four gates using CLCxGLS0, CLCxGLS1, CLCxGLS2, and CLCxGLS3 registers.
- Select the gate output polarities with the LCxGyPOL bits of the CLCxPOL register.
- Select the desired logic function with the LCxMODE<2:0> bits of the CLCxCON register.
- Select the desired polarity of the logic output with the LCxPOL bit of the CLCxPOL register. (This step may be combined with the previous gate output polarity step).
- If driving a device pin, set the desired pin PPS control register and also clear the TRIS bit corresponding to that output.
- If interrupts are desired, configure the following bits:
  - Set the LCxINTP bit in the CLCxCON register for rising event.
  - Set the LCxINTN bit in the CLCxCON register for falling event.
  - Set the CLCxIE bit of the PIE5 register.
  - Set the GIE and PEIE bits of the INTCON register.
- Enable the CLCx by setting the LCxEN bit of the CLCxCON register.

SLEEP **Enter Sleep mode**

Syntax: [*label*] SLEEP

Operands: None

Operation: 00h → WDT,
0 → WDT prescaler,
1 → $\overline{TO}$,
0 → $\overline{PD}$

Status Affected: $\overline{TO}$, $\overline{PD}$

Description: The power-down Status bit, $\overline{PD}$ is cleared. Time-out Status bit, $\overline{TO}$ is set. Watchdog Timer and its prescaler are cleared.
See **Section 11.2 "Sleep Mode"** for more information.

SUBLW **Subtract W from literal**

Syntax: [*label*] SUBLW k

Operands: $0 \le k \le 255$

Operation: k - (W) → (W)

Status Affected: C, DC, Z

Description: The W register is subtracted (2's complement method) from the 8-bit literal 'k'. The result is placed in the W register.

C = 0	W > k
C = 1	W ≤ k
DC = 0	W<3:0> > k<3:0>
DC = 1	W<3:0> ≤ k<3:0>

SUBWF **Subtract W from f**

Syntax: [*label*] SUBWF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) - (W) → (destination)

Status Affected: C, DC, Z

Description: Subtract (2's complement method) W register from register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f.

C = 0	W > f
C = 1	W ≤ f
DC = 0	W<3:0> > f<3:0>
DC = 1	W<3:0> ≤ f<3:0>

SUBWFB **Subtract W from f with Borrow**

Syntax: SUBWFB f {,d}

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f) – (W) – ($\overline{B}$) → dest

Status Affected: C, DC, Z

Description: Subtract W and the BORROW flag (CARRY) from register 'f' (2's complement method). If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f'.

SWAPF **Swap Nibbles in f**

Syntax: [*label*] SWAPF f,d

Operands: $0 \le f \le 127$
$d \in [0,1]$

Operation: (f<3:0>) → (destination<7:4>),
(f<7:4>) → (destination<3:0>)

Status Affected: None

Description: The upper and lower nibbles of register 'f' are exchanged. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed in register 'f'.

TABLE 37-13: ANALOG-TO-DIGITAL CONVERTER (ADC) CONVERSION TIMING SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	**Sym.**	**Characteristic**	**Min.**	**Typ†**	**Max.**	**Units**	**Conditions**
AD20	TAD	ADC Clock Period	1	—	9	µs	The requirement is to set ADCCS correctly to produce this period/frequency.
AD21			1	2	6	µs	Using FRC as the ADC clock source ADOSC = 1
AD22	TCNV	Conversion Time	—	11	—	TAD	Set of GO/DONE bit to Clear of GO/DONE bit
AD23	TACQ	Acquisition Time	—	2	—	µs	
AD24	THCD	Sample and Hold Capacitor Disconnect Time	—	—	—	µs	FOSC-based clock source FRC-based clock source

* These parameters are characterized but not tested.

† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 37-10: ADC CONVERSION TIMING (ADC CLOCK FOSC-BASED)

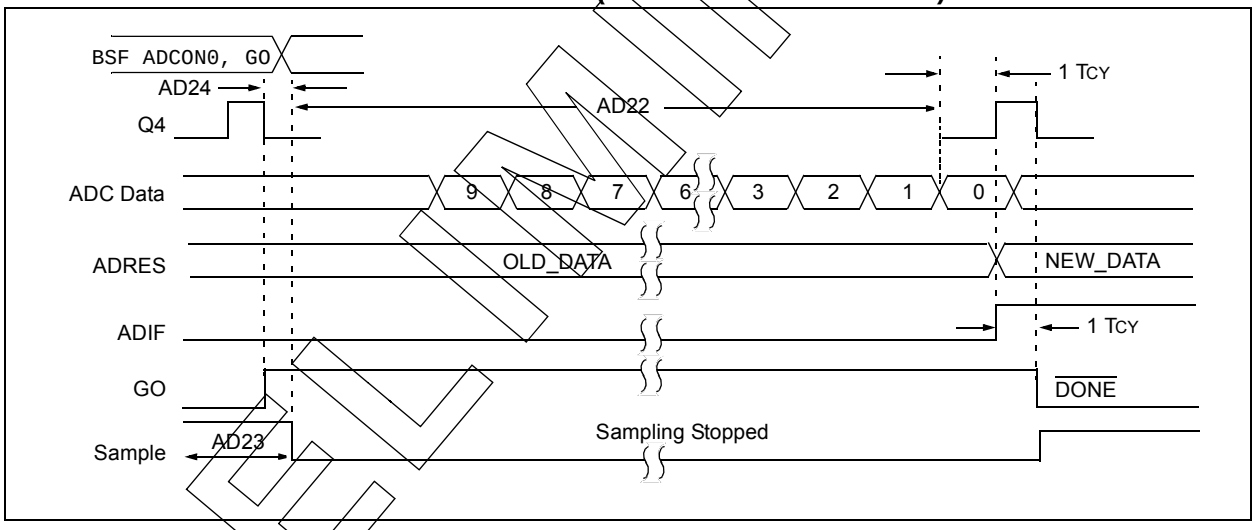
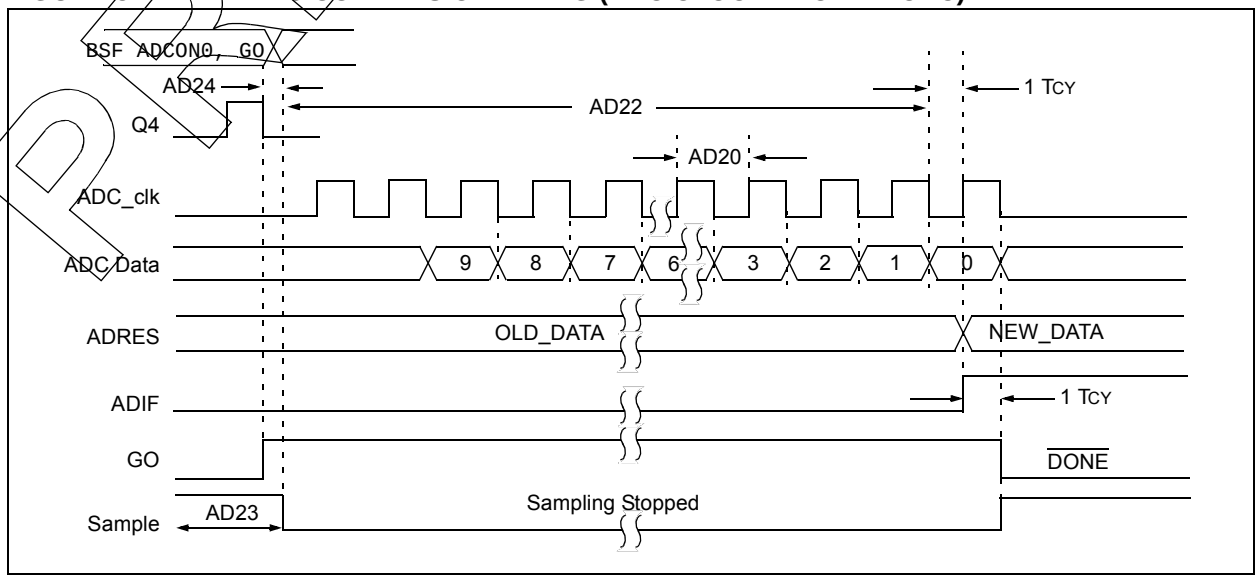
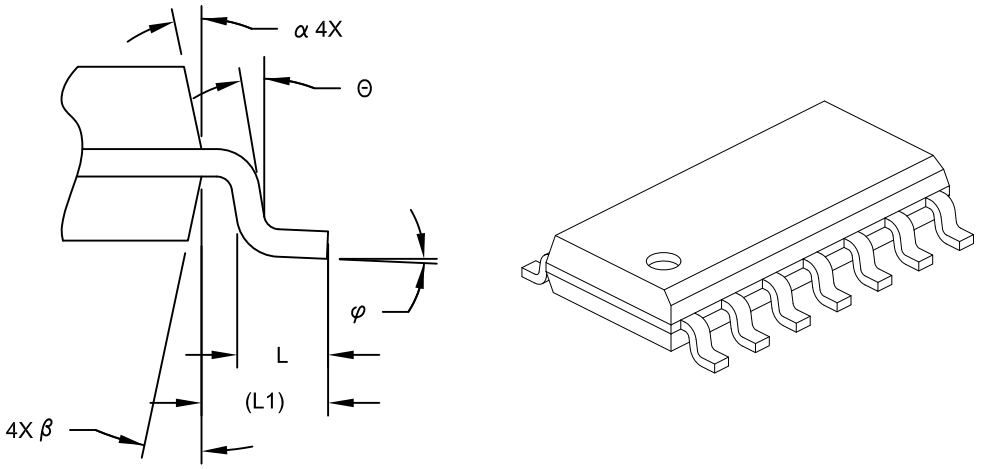


FIGURE 37-11: ADC CONVERSION TIMING (ADC CLOCK FROM ADCRC)



14-Lead Plastic Small Outline (SL) - Narrow, 3.90 mm Body [SOIC]

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



VIEW C

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		14	
Pitch	e		1.27 BSC	
Overall Height	A	-	-	1.75
Molded Package Thickness	A2	1.25	-	-
Standoff §	A1	0.10	-	0.25
Overall Width	E		6.00 BSC	
Molded Package Width	E1		3.90 BSC	
Overall Length	D		8.65 BSC	
Chamfer (Optional)	h	0.25	-	0.50
Foot Length	L	0.40	-	1.27
Footprint	L1		1.04 REF	
Lead Angle	Θ	0°	-	-
Foot Angle	φ	0°	-	8°
Lead Thickness	c	0.10	-	0.25
Lead Width	b	0.31	-	0.51
Mold Draft Angle Top	α	5°	-	15°
Mold Draft Angle Bottom	β	5°	-	15°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic
3. Dimension D does not include mold flash, protrusions or gate burrs, which shall not exceed 0.15 mm per end. Dimension E1 does not include interlead flash or protrusion, which shall not exceed 0.25 mm per side.
4. 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.
5. Datums A & B to be determined at Datum H.

Microchip Technology Drawing No. C04-065C Sheet 2 of 2