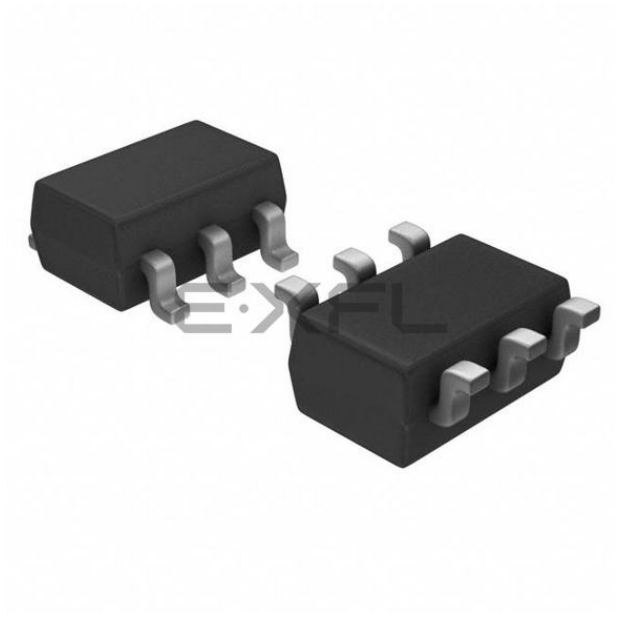


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	PIC
Core Size	8-Bit
Speed	16MHz
Connectivity	-
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	3
Program Memory Size	896B (512 x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	64 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 3x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	SOT-23-6
Supplier Device Package	SOT-23-6
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic10lf322-i-ot

FIGURE 1: 6-PIN DIAGRAM, PIC10(L)F320/322

SOT-23

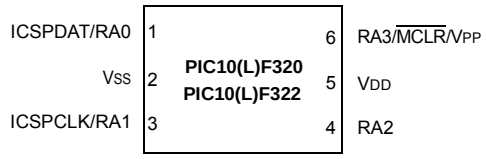


FIGURE 2: 8-PIN DIAGRAM, PIC10(L)F320/322

PDIP, DFN

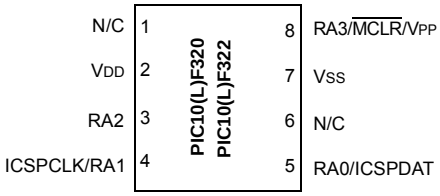


TABLE 1: 6 AND 8-PIN ALLOCATION TABLE, PIC10(L)F320/322

I/O	6-Pin	8-Pin	Analog	Timer	PWM	Interrupts	Pull-ups	CWG	NCO	CLC	Basic	ICSP
RA0	1	5	AN0	—	PWM1	IOC0	Y	CWG1A	—	CLC1IN0	—	ICSPDAT
RA1	3	4	AN1	—	PWM2	IOC1	Y	CWG1B	NCO1CLK	CLC1	CLKIN	ICSPCLK
RA2	4	3	AN2	T0CKI	—	INT/IOC2	Y	$\overline{\text{CWG1FLT}}$	NCO1	CLC1IN1	CLKR	
RA3	6	8	—	—	—	IOC3	Y	—	—	—	$\overline{\text{MCLR}}$	VPP
N/C	—	1	—	—	—	—	—	—	—	—	—	—
N/C	—	6	—	—	—	—	—	—	—	—	—	—
VDD	5	2	—	—	—	—	—	—	—	—	VDD	—
VSS	2	7	—	—	—	—	—	—	—	—	VSS	—

2.2.3 DEVICE MEMORY MAPS

The memory maps for PIC10(L)F320/322 are as shown in Table 2-2.

TABLE 2-2: PIC10(L)F320/322 MEMORY MAP (BANK 0)

INDF(*)	00h	PMADRL	20h		40h		60h
TMR0	01h	PMADRH	21h				
PCL	02h	PMDATL	22h				
STATUS	03h	PMDATH	23h				
FSR	04h	PMCON1	24h				
PORTA	05h	PMCON2	25h				
TRISA	06h	CLKRCON	26h				
LATA	07h	NCO1ACCL	27h				
ANSELA	08h	NCO1ACCH	28h				
WPUA	09h	NCO1ACCU	29h				
PCLATH	0Ah	NCO1INCL	2Ah				
INTCON	0Bh	NCO1INCH	2Bh				
PIR1	0Ch	Reserved	2Ch				
PIE1	0Dh	NCO1CON	2Dh				
OPTION_REG	0Eh	NCO1CLK	2Eh	General Purpose Registers		General Purpose Registers	
PCON	0Fh	Reserved	2Fh				
OSCCON	10h	WDTCON	30h				
TMR2	11h	CLC1CON	31h	32 Bytes		32 Bytes	
PR2	12h	CLC1SEL1	32h				
T2CON	13h	CLC1SEL2	33h				
PWM1DCL	14h	CLC1POL	34h				
PWM1DCH	15h	CLC1GLS0	35h				
PWM1CON	16h	CLC1GLS1	36h				
PWM2DCL	17h	CLC1GLS2	37h				
PWM2DCH	18h	CLC1GLS3	38h				
PWM2CON	19h	CWG1CON0	39h				
IOCAP	1Ah	CWG1CON1	3Ah				
IOCAN	1Bh	CWG1CON2	3Bh				
IOCAF	1Ch	CWG1DBR	3Ch				
FVRCON	1Dh	CWG1DBF	3Dh				
ADRES	1Eh	VREGCON	3Eh				
ADCON	1Fh	BORCON	3Fh		5Fh		7Fh

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

* = Not a physical register.

REGISTER 3-1: CONFIG: CONFIGURATION WORD (CONTINUED)

bit 5 **$\overline{\text{PWRTE}}$:** Power-up Timer Enable bit(1)
1 = PWRT disabled
0 = PWRT enabled

bit 4-3 **WDTE<1:0>:** Watchdog Timer Enable bit
11 = WDT enabled
10 = WDT enabled while running and disabled in Sleep
01 = WDT controlled by the SWDTEN bit in the WDTCON register
00 = WDT disabled

bit 2-1 **BOREN<1:0>:** Brown-out Reset Enable bits
11 = Brown-out Reset enabled; SBOREN bit is ignored
10 = Brown-out Reset enabled while running, disabled in Sleep; SBOREN bit is ignored
01 = Brown-out Reset controlled by the SBOREN bit in the BORCON register
00 = Brown-out Reset disabled; SBOREN bit is ignored

bit 0 **FOSC:** Oscillator Selection bit
1 = EC on CLKIN pin
0 = INTOSC oscillator I/O function available on CLKIN pin

Note 1: Enabling Brown-out Reset does not automatically enable Power-up Timer.
2: Once enabled, code-protect can only be disabled by bulk erasing the device.
3: See V_{BOR} parameter for specific trip point voltages.

TABLE 8-3: SUMMARY OF REGISTERS ASSOCIATED WITH WATCHDOG TIMER

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
OSCCON	—	IRCF<2:0>			HFIOFR	—	LFIOFR	HFIOFS	26
STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	13
WDTCON	—	—	WDTPS<4:0>					SWDTEN	48

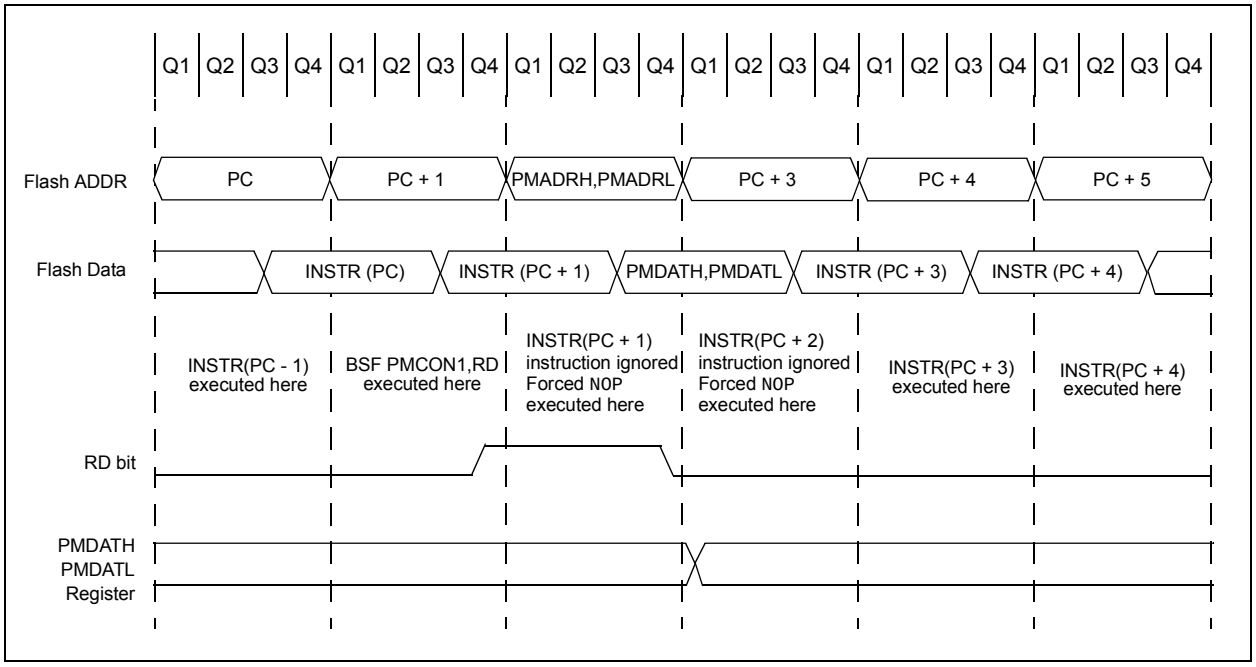
Legend: x = unknown, u = unchanged, – = unimplemented locations read as '0'. Shaded cells are not used by Watchdog Timer.

TABLE 8-4: SUMMARY OF CONFIGURATION WORD WITH WATCHDOG TIMER

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
CONFIG	13:8	—	—	—	WRT<1:0>		BORV	LPBOR	LVP	20
	7:0	$\overline{CP}$	MCLRE	$\overline{PWRTE}$	WDTE<1:0>		BOREN<1:0>		FOSC	

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

FIGURE 9-2: FLASH PROGRAM MEMORY READ CYCLE EXECUTION



EXAMPLE 9-1: FLASH PROGRAM MEMORY READ

```
* This code block will read 1 word of program
* memory at the memory address:
    PROG_ADDR_HI: PROG_ADDR_LO
*   data will be returned in the variables;
*   PROG_DATA_HI, PROG_DATA_LO

    BANKSEL  PMADRL           ; not required on devices with 1 Bank of SFRs
    MOVLW    PROG_ADDR_LO     ;
    MOVWF    PMADRL           ; Store LSB of address
    MOVLW    PROG_ADDR_HI     ;
    MOVWF    PMADRH           ; Store MSB of address

    BCF      PMCON1,CFGS      ; Do not select Configuration Space
    BSF      PMCON1,RD        ; Initiate read
    NOP                       ; Ignored (Figure 9-2)
    NOP                       ; Ignored (Figure 9-2)

    MOVF     PMDATL,W         ; Get LSB of word
    MOVWF    PROG_DATA_LO     ; Store in user location
    MOVF     PMDATH,W         ; Get MSB of word
    MOVWF    PROG_DATA_HI     ; Store in user location
```

EXAMPLE 9-3: WRITING TO FLASH PROGRAM MEMORY

```
    ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
    ; This write routine assumes the following:
    ;        A valid starting address (the least significant bits = '00')
    ;        is loaded in ADDRH:ADDRL
    ;        ADDRH, ADDRL and DATADDR are all located in data memory
    ;
   BANKSEL         PMADRH
   MOVF    ADDRH,W      ;Load initial address
   MOVWF   PMADRH       ;
   MOVF    ADDRL,W      ;
   MOVWF   PMADRL       ;
   MOVF    DATAADDR,W   ;Load initial data address
   MOVWF   FSR          ;
LOOP MOVF  INDF,W       ;Load first data byte into lower
   MOVWF   PMDATL       ;
   INCF    FSR,F        ;Next byte
   MOVF    INDF,W       ;Load second data byte into upper
   MOVWF   PMDATH       ;
   INCF    FSR,F        ;
   BANKSEL PMCON1
   BSF     PMCON1,WREN  ;Enable writes
   BCF     INTCON,GIE   ;Disable interrupts (if using)
   BTFSC   INTCON,GIE   ;See AN576
   GOTO    $-2
   ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
   ;       Required Sequence
   MOVLW   55h          ;Start of required write sequence:
   MOVWF   PMCON2       ;Write 55h
   MOVLW   0AAh         ;
   MOVWF   PMCON2       ;Write 0AAh
   BSF     PMCON1,WR    ;Set WR bit to begin write
   NOP                  ;Required to transfer data to the buffer
   NOP                  ;registers
   ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
   BCF     PMCON1,WREN  ;Disable writes
   BSF     INTCON,GIE   ;Enable interrupts (comment out if not using interrupts)
   BANKSEL PMADRL
   MOVF    PMADRL, W
   INCF    PMADRL,F     ;Increment address
   ANDLW   0x03         ;Indicates when sixteen words have been programmed
   SUBLW   0x03         ;Change value for different size write blocks
                        ;0x0F = 16 words
                        ;0x0B = 12 words
                        ;0x07 = 8 words
                        ;0x03 = 4 words
   BTFSS   STATUS,Z     ;Exit on a match,
   GOTO    LOOP         ;Continue if more data needs to be written
```

12.3 FVR Control Registers

REGISTER 12-1: FVRCON: FIXED VOLTAGE REFERENCE CONTROL REGISTER

R/W-0/0	R-q/q	R/W-0/0	R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0
FVREN	FVRRDY(1)	TSEN(3)	TSRNG(3)	—	—	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 = $V_{OUT} = V_{DD} - 4V_T$ (High Range)
0 = $V_{OUT} = V_{DD} - 2V_T$ (Low Range)

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

bit 1-0 **ADFVR<1:0>:** ADC Fixed Voltage Reference Selection bit
11 = ADC Fixed Voltage Reference Peripheral output is 4x (4.096V)(2)
10 = ADC Fixed Voltage Reference Peripheral output is 2x (2.048V)(2)
01 = ADC Fixed Voltage Reference Peripheral output is 1x (1.024V)
00 = ADC Fixed Voltage Reference Peripheral output is off.

Note 1: FVRRDY indicates the true state of the FVR.
2: Fixed Voltage Reference output cannot exceed V_{DD}.
3: See **Section 14.0 "Temperature Indicator Module"** for additional information.

TABLE 12-2: SUMMARY OF REGISTERS ASSOCIATED WITH FIXED VOLTAGE REFERENCE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on page
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	—	—	ADFVR<1:0>		78

Legend: Shaded cells are not used with the Fixed Voltage Reference.

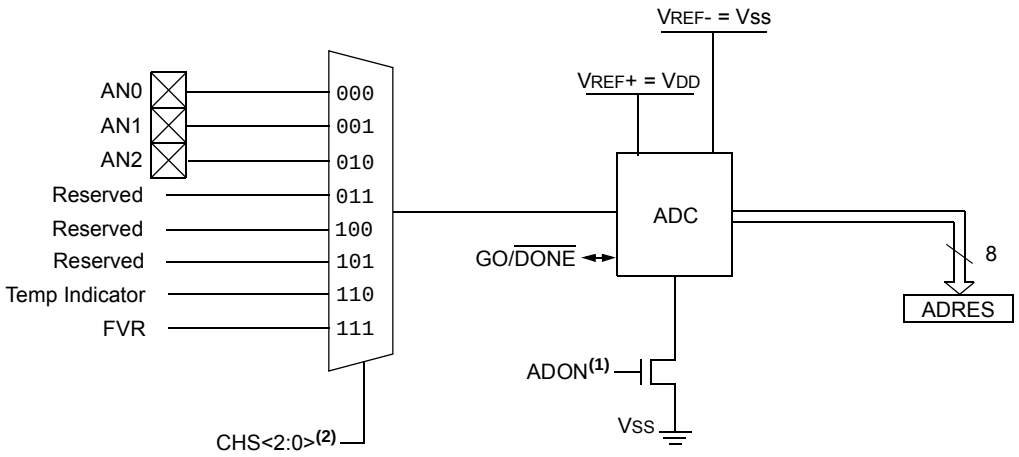
15.0 ANALOG-TO-DIGITAL CONVERTER (ADC) MODULE

The Analog-to-Digital Converter (ADC) converts an analog input signal to an 8-bit binary representation of that signal. This device uses three analog input channels, which are multiplexed into a single sample and hold circuit. The output of the sample and hold is connected to the input of the converter. The converter generates an 8-bit binary result via successive approximation and stores the conversion result into the ADC result register (ADRES). Figure 15-1 shows the block diagram of the ADC.

The ADC voltage reference is software selectable to be internally generated.

The ADC can generate an interrupt upon completion of a conversion. This interrupt can be used to wake-up the device from Sleep.

FIGURE 15-1: ADC SIMPLIFIED BLOCK DIAGRAM



Note 1: When ADON = 0, all multiplexer inputs are disconnected.

2: See ADCON register (Register 15-1) for detailed analog channel selection per device.

15.1.5 INTERRUPTS

The ADC module allows for the ability to generate an interrupt upon completion of an Analog-to-Digital conversion. The ADC Interrupt Flag is the ADIF bit in the PIR1 register. The ADC Interrupt Enable is the ADIE bit in the PIE1 register. The ADIF bit must be cleared in software.

Note:	The ADIF bit is set at the completion of every conversion, regardless of whether or not the ADC interrupt is enabled.

This interrupt can be generated while the device is operating or while in Sleep. If the device is in Sleep, the interrupt will wake-up the device. Upon waking from Sleep, the next instruction following the SLEEP instruction is always executed. If the user is attempting to wake-up from Sleep and resume in-line code execution, the GIE and PEIE bits of the INTCON register must be disabled. If the GIE and PEIE bits of the INTCON register are enabled, execution will switch to the Interrupt Service Routine.

15.2 ADC Operation

15.2.1 STARTING A CONVERSION

To enable the ADC module, the ADON bit of the ADCON register must be set to a '1'. Setting the GO/$\overline{\text{DONE}}$ bit of the ADCON register to a '1' will start the Analog-to-Digital conversion.

Note:	The GO/$\overline{\text{DONE}}$ bit should not be set in the same instruction that turns on the ADC. Refer to **Section 15.2.5 "A/D Conversion Procedure"**.

15.2.2 COMPLETION OF A CONVERSION

When the conversion is complete, the ADC module will:

- Clear the GO/$\overline{\text{DONE}}$ bit
- Set the ADIF Interrupt Flag bit
- Update the ADRES register with new conversion result

15.2.3 TERMINATING A CONVERSION

If a conversion must be terminated before completion, the GO/$\overline{\text{DONE}}$ bit can be cleared in software. The ADRES register will be updated with the partially complete Analog-to-Digital conversion sample. Incomplete bits will match the last bit converted.

Note:	A device Reset forces all registers to their Reset state. Thus, the ADC module is turned off and any pending conversion is terminated.

15.2.4 ADC OPERATION DURING SLEEP

The ADC module can operate during Sleep. This requires the ADC clock source to be set to the FRC option. When the FRC clock source is selected, the ADC waits one additional instruction before starting the conversion. This allows the SLEEP instruction to be executed, which can reduce system noise during the conversion. If the ADC interrupt is enabled, the device will wake-up from Sleep when the conversion completes. If the ADC interrupt is disabled, the ADC module is turned off after the conversion completes, although the ADON bit remains set.

When the ADC clock source is something other than FRC, a SLEEP instruction causes the present conversion to be aborted and the ADC module is turned off, although the ADON bit remains set.

15.2.5 A/D CONVERSION PROCEDURE

This is an example procedure for using the ADC to perform an Analog-to-Digital conversion:

1. Configure Port:
   - Disable pin output driver (Refer to the TRIS register)
   - Configure pin as analog (Refer to the ANSEL register)
   - Disable weak pull-ups either globally (Refer to the OPTION_REG register) or individually (Refer to the appropriate WPUX register)
2. Configure the ADC module:
   - Select ADC conversion clock
   - Select ADC input channel
   - Turn on ADC module
3. Configure ADC interrupt (optional):
   - Clear ADC interrupt flag
   - Enable ADC interrupt
   - Enable peripheral interrupt
   - Enable global interrupt**(1)**
4. Wait the required acquisition time**(2)**.
5. Start conversion by setting the GO/$\overline{\text{DONE}}$ bit.
6. Wait for ADC conversion to complete by one of the following:
   - Polling the GO/$\overline{\text{DONE}}$ bit
   - Waiting for the ADC interrupt (interrupts enabled)
7. Read ADC Result.
8. Clear the ADC interrupt flag (required if interrupt is enabled).

Note 1: The global interrupt can be disabled if the user is attempting to wake-up from Sleep and resume in-line code execution.

2: Refer to **Section 15.4 "A/D Acquisition Requirements"**.

REGISTER 15-2: ADRES: ADC RESULT 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
ADRES<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	

bit 7-0 **ADRES<7:0>**: ADC Result Register bits
8-bit result

16.1.3 SOFTWARE PROGRAMMABLE PRESCALER

A single software programmable prescaler is available for use with Timer0. The prescaler assignment is controlled by the PSA bit of the OPTION_REG register. To assign the prescaler to Timer0, the PSA bit must be cleared to a '0'.

There are eight prescaler options for the Timer0 module ranging from 1:2 to 1:256. The prescale values are selectable via the PS<2:0> bits of the OPTION_REG register.

The prescaler is not readable or writable. When assigned to the Timer0 module, all instructions writing to the TMR0 register will clear the prescaler.

16.1.4 TIMER0 INTERRUPT

Timer0 will generate an interrupt when the TMR0 register overflows from FFh to 00h. The TMR0IF interrupt flag bit of the INTCON register is set every time the TMR0 register overflows, regardless of whether or not the Timer0 interrupt is enabled. The TMR0IF bit can only be cleared in software. The Timer0 interrupt enable is the TMR0IE bit of the INTCON register.

Note:	The Timer0 interrupt cannot wake the processor from Sleep since the timer is frozen during Sleep.

16.1.5 8-BIT COUNTER MODE SYNCHRONIZATION

When in 8-Bit Counter mode, the incrementing edge on the T0CKI pin must be synchronized to the instruction clock. Synchronization can be accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the instruction clock. The high and low periods of the external clocking source must meet the timing requirements as shown in **Section 24.0 "Electrical Specifications"**.

REGISTER 18-2: PWMxDCH: PWM DUTY CYCLE HIGH BITS

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
PWMxDCH<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	

bit 7-0 **PWMxDCH<7:0>:** PWM Duty Cycle Most Significant bits
These bits are the MSbs of the PWM duty cycle. The two LSbs are found in the PWMxDCL Register.

REGISTER 18-3: PWMxDCL: PWM DUTY CYCLE LOW BITS

R/W-x/u	R/W-x/u	U-0	U-0	U-0	U-0	U-0	U-0
PWMxDCL<7:6>		—	—	—	—	—	—
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 **PWMxDCL<7:6>:** PWM Duty Cycle Least Significant bits
These bits are the LSbs of the PWM duty cycle. The MSbs are found in the PWMxDCH Register.

bit 5-0 **Unimplemented:** Read as ‘0’

TABLE 18-3: SUMMARY OF REGISTERS ASSOCIATED WITH PWM

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	—	—	—	—	—	ANSA2	ANSA1	ANSA0	70
LATA	—	—	—	—	—	LATA2	LATA1	LATA0	70
PORTA	—	—	—	—	RA3	RA2	RA1	RA0	69
PR2	Timer2 module Period Register								96
PWM1CON	PWM1EN	PWM1OE	PWM1OUT	PWM1POL	—	—	—	—	102
PWM1DCH	PWM1DCH<7:0>								103
PWM1DCL	PWM1DCL<7:6>		—	—	—	—	—	—	103
PWM2CON	PWM2EN	PWM2OE	PWM2OUT	PWM2POL	—	—	—	—	102
PWM2DCH	PWM2DCH<7:0>								103
PWM2DCL	PWM2DCL<7:6>		—	—	—	—	—	—	103
T2CON	—	TOUTPS<3:0>				TMR2ON	T2CKPS<1:0>		97
TMR2	Timer2 module Register								96
TRISA	—	—	—	—	—	TRISA2	TRISA1	TRISA0	69

Legend: - = Unimplemented locations, read as ‘0’, u = unchanged, x = unknown. Shaded cells are not used by the PWM.

REGISTER 19-2: CLCxPOL: SIGNAL POLARITY CONTROL REGISTER

R/W-x/u	U-0	U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
LCxPOL	—	—	—	LCxG4POL	LCxG3POL	LCxG2POL	LCxG1POL
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 **LCxPOL:** LCOUT Polarity Control bit
1 = The output of the logic cell is inverted
0 = The output of the logic cell is not inverted

bit 6-4 **Unimplemented:** Read as ‘0’

bit 3 **LCxG4POL:** Gate 4 Output Polarity Control bit
1 = The output of gate 4 is inverted when applied to the logic cell
0 = The output of gate 4 is not inverted

bit 2 **LCxG3POL:** Gate 3 Output Polarity Control bit
1 = The output of gate 3 is inverted when applied to the logic cell
0 = The output of gate 3 is not inverted

bit 1 **LCxG2POL:** Gate 2 Output Polarity Control bit
1 = The output of gate 2 is inverted when applied to the logic cell
0 = The output of gate 2 is not inverted

bit 0 **LCxG1POL:** Gate 1 Output Polarity Control bit
1 = The output of gate 1 is inverted when applied to the logic cell
0 = The output of gate 1 is not inverted

21.1 Fundamental Operation

The CWG generates a two output complementary waveform from one of four selectable input sources.

The off-to-on transition of each output can be delayed from the on-to-off transition of the other output, thereby, creating a time delay immediately where neither output is driven. This is referred to as dead time and is covered in **Section 21.5 "Dead-Band Control"**. A typical operating waveform, with dead band, generated from a single input signal is shown in Figure 21-2.

It may be necessary to guard against the possibility of circuit faults or a feedback event arriving too late or not at all. In this case, the active drive must be terminated before the Fault condition causes damage. This is referred to as auto-shutdown and is covered in **Section 21.9 "Auto-shutdown Control"**.

21.2 Clock Source

The CWG module allows the following clock sources to be selected:

- Fosc (system clock)
- HFINTOSC (16 MHz only)

The clock sources are selected using the G1CS0 bit of the CWGxCON0 register (Register 21-1).

21.3 Selectable Input Sources

The CWG can generate the complementary waveform for the following input sources:

- PWM1
- PWM2
- N1OUT
- LC1OUT

The input sources are selected using the GxIS<1:0> bits in the CWGxCON1 register (Register 21-2).

21.4 Output Control

Immediately after the CWG module is enabled, the complementary drive is configured with both CWGxA and CWGxB drives cleared.

21.4.1 OUTPUT ENABLES

Each CWG output pin has individual output enable control. Output enables are selected with the GxOEA and GxOEB bits of the CWGxCON0 register. When an output enable control is cleared, the module asserts no control over the pin. When an output enable is set, the override value or active PWM waveform is applied to the pin per the port priority selection. The output pin enables are dependent on the module enable bit, GxEN. When GxEN is cleared, CWG output enables and CWG drive levels have no effect.

21.4.2 POLARITY CONTROL

The polarity of each CWG output can be selected independently. When the output polarity bit is set, the corresponding output is active-high. Clearing the output polarity bit configures the corresponding output as active-low. However, polarity does not affect the override levels. Output polarity is selected with the GxPOLA and GxPOLB bits of the CWGxCON0 register.

21.5 Dead-Band Control

Dead-band control provides for non-overlapping output signals to prevent shoot-through current in power switches. The CWG contains two 6-bit dead-band counters. One dead-band counter is used for the rising edge of the input source control. The other is used for the falling edge of the input source control.

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 (Register 21-4 and Register 21-5, respectively).

21.6 Rising Edge Dead Band

The rising edge dead band delays the turn-on of the CWGxA output from when the CWGxB output is turned off. The rising edge dead-band time starts when the rising edge of the input source signal goes true. When this happens, the CWGxB output is immediately turned off and the rising edge dead-band delay time starts. When the rising edge dead-band delay time is reached, the CWGxA output is turned on.

The CWGxDBR register sets the duration of the dead-band interval on the rising edge of the input source signal. This duration is from 0 to 64 counts of dead band.

Dead band is always counted off the edge on the input source signal. A count of 0 (zero), indicates that no dead band is present.

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.

FIGURE 21-3: DEAD-BAND OPERATION, CWGxDBR = 01H, CWGxDBF = 02H

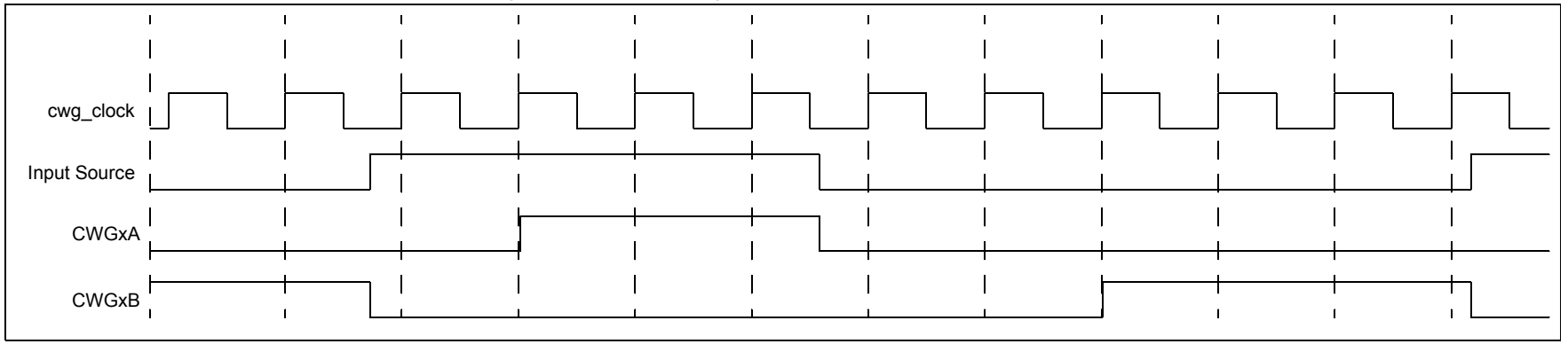
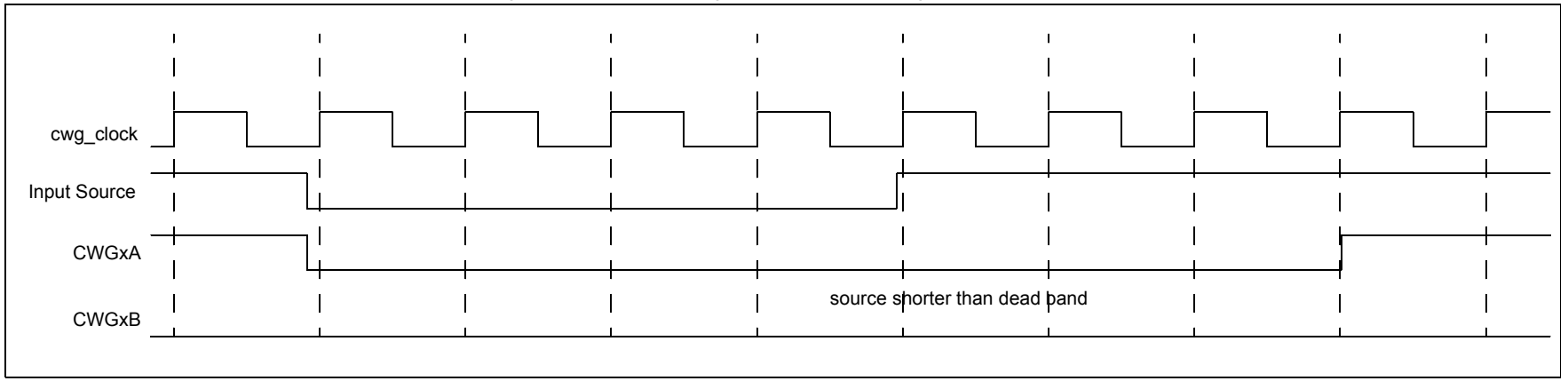


FIGURE 21-4: DEAD-BAND OPERATION, CWGxDBR = 03H, CWGxDBF = 04H, SOURCE SHORTER THAN DEAD BAND



22.0 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
- $\overline{MCLR}$/VPP
- VDD
- VSS

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 **"***PIC10(L)F320/322 Flash Memory Programming Specification*" (DS41572).

22.1 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 $\overline{MCLR}$/VPP to VIHH.

22.2 Low-Voltage Programming Entry Mode

The Low-Voltage Programming Entry mode allows the PIC® Flash MCUs to be programmed using VDD only, without high voltage. When the LVP bit of Configuration Word is set to '1', the low-voltage ICSP programming entry 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 VIL.
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 VIL for as long as Program/Verify mode is to be maintained.

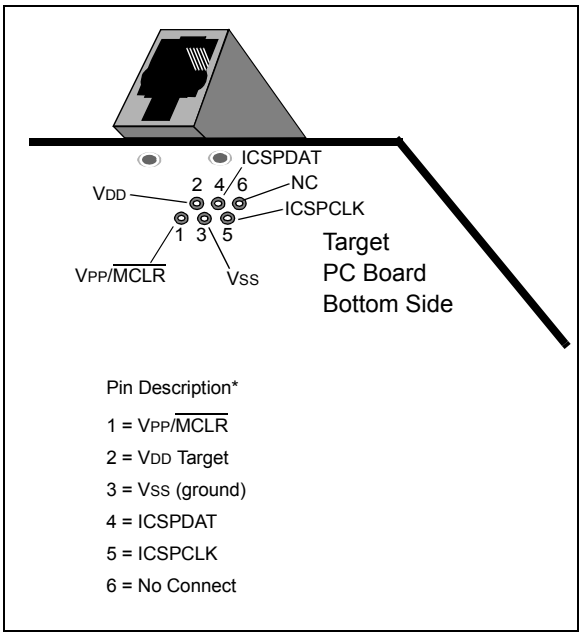
If low-voltage programming is enabled (LVP = 1), the $\overline{MCLR}$ Reset function is automatically enabled and cannot be disabled. See **Section 5.4 "Low-Power Brown-out Reset (LPBOR)"** for more information.

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

22.3 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 22-1.

FIGURE 22-1: 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 22-2.

26.0 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

26.1 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

TABLE 27-1: 8-LEAD 2x3 DFN (MC) TOP MARKING

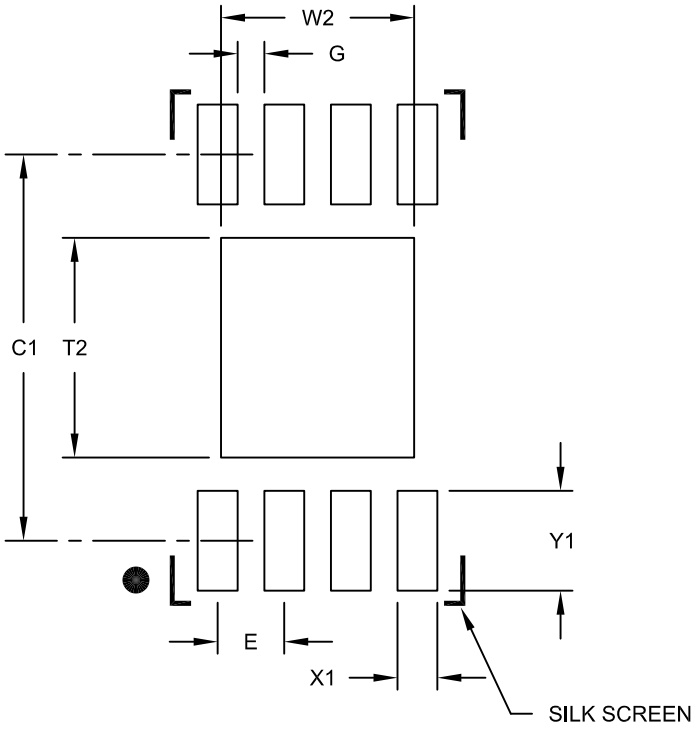
Part Number	Marking
PIC10F322(T)-I/MC	BAA
PIC10F322(T)-E/MC	BAB
PIC10F320(T)-I/MC	BAC
PIC10F320(T)-E/MC	BAD
PIC10LF322(T)-I/MC	BAF
PIC10LF322(T)-E/MC	BAG
PIC10LF320(T)-I/MC	BAH
PIC10LF320(T)-E/MC	BAJ

TABLE 27-2: 6-LEAD SOT-23 (OT) PACKAGE TOP MARKING

Part Number	Marking
PIC10F322(T)-I/OT	LA/LJ
PIC10F322(T)-E/OT	LB/LK
PIC10F320(T)-I/OT	LC
PIC10F320(T)-E/OT	LD
PIC10LF322(T)-I/OT	LE
PIC10LF322(T)-E/OT	LF
PIC10LF320(T)-I/OT	LG
PIC10LF320(T)-E/OT	LH

8-Lead Plastic Dual Flat, No Lead Package (MC) - 2x3x0.9mm Body [DFN]

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.50 BSC		
Optional Center Pad Width	W2			1.45
Optional Center Pad Length	T2			1.75
Contact Pad Spacing	C1		2.90	
Contact Pad Width (X8)	X1			0.30
Contact Pad Length (X8)	Y1			0.75
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2123B