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	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 17x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SSOP (0.209", 5.30mm Width)
Supplier Device Package	20-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f15344t-i-ss

1.1 Register and Bit Naming Conventions

1.1.1 REGISTER NAMES

When there are multiple instances of the same peripheral in a device, the peripheral control registers will be depicted as the concatenation of a peripheral identifier, peripheral instance, and control identifier. The control registers section will show just one instance of all the register names with an 'x' in the place of the peripheral instance number. This naming convention may also be applied to peripherals when there is only one instance of that peripheral in the device to maintain compatibility with other devices in the family that contain more than one.

1.1.2 BIT NAMES

There are two variants for bit names:

- Short name: Bit function abbreviation
- Long name: Peripheral abbreviation + short name

1.1.2.1 Short Bit Names

Short bit names are an abbreviation for the bit function. For example, some peripherals are enabled with the EN bit. The bit names shown in the registers are the short name variant.

Short bit names are useful when accessing bits in C programs. The general format for accessing bits by the short name is *RegisterName*bits.*ShortName*. For example, the enable bit, EN, in the COG1CON0 register can be set in C programs with the instruction `COG1CON0bits.EN = 1`.

Short names are generally not useful in assembly programs because the same name may be used by different peripherals in different bit positions. When this occurs, during the include file generation, all instances of that short bit name are appended with an underscore plus the name of the register in which the bit resides to avoid naming contentions.

1.1.2.2 Long Bit Names

Long bit names are constructed by adding a peripheral abbreviation prefix to the short name. The prefix is unique to the peripheral thereby making every long bit name unique. The long bit name for the COG1 enable bit is the COG1 prefix, G1, appended with the enable bit short name, EN, resulting in the unique bit name G1EN.

Long bit names are useful in both C and assembly programs. For example, in C the COG1CON0 enable bit can be set with the `G1EN = 1` instruction. In assembly, this bit can be set with the `BSF COG1CON0,G1EN` instruction.

1.1.2.3 Bit Fields

Bit fields are two or more adjacent bits in the same register. Bit fields adhere only to the short bit naming convention. For example, the three Least Significant bits of the COG1CON0 register contain the mode control bits. The short name for this field is MD. There is no long bit name variant. Bit field access is only possible in C programs. The following example demonstrates a C program instruction for setting the COG1 to the Push-Pull mode:

```
COG1CON0bits.MD = 0x5;
```

Individual bits in a bit field can also be accessed with long and short bit names. Each bit is the field name appended with the number of the bit position within the field. For example, the Most Significant mode bit has the short bit name MD2 and the long bit name is G1MD2. The following two examples demonstrate assembly program sequences for setting the COG1 to Push-Pull mode:

Example 1:

```
MOVLW  ~(1<<G1MD1)
ANDWF  COG1CON0,F
MOVLW  1<<G1MD2 | 1<<G1MD0
IORWF  COG1CON0,F
```

Example 2:

```
BSF    COG1CON0,G1MD2
BCF    COG1CON0,G1MD1
BSF    COG1CON0,G1MD0
```

1.1.3 REGISTER AND BIT NAMING EXCEPTIONS

1.1.3.1 Status, Interrupt, and Mirror Bits

Status, interrupt enables, interrupt flags, and mirror bits are contained in registers that span more than one peripheral. In these cases, the bit name shown is unique so there is no prefix or short name variant.

1.1.3.2 Legacy Peripherals

There are some peripherals that do not strictly adhere to these naming conventions. Peripherals that have existed for many years and are present in almost every device are the exceptions. These exceptions were necessary to limit the adverse impact of the new conventions on legacy code. Peripherals that do adhere to the new convention will include a table in the registers section indicating the long name prefix for each peripheral instance. Peripherals that fall into the exception category will not have this table. These peripherals include, but are not limited to, the following:

- EUSART
- MSSP

TABLE 4-6: PIC16(L)F15324/44 MEMORY MAP, BANKS 16-23

	BANK 16		BANK 17		BANK 18		BANK 19		BANK 20		BANK 21		BANK 22		BANK 23
800h–80Bh	Core Register (Table 4-3)	880h–88Bh	Core Register (Table 4-3)	900h–90Bh	Core Register (Table 4-3)	980h–98Bh	Core Register (Table 4-3)	A00h–A0Bh	Core Register (Table 4-3)	A80h–A8Bh	Core Register (Table 4-3)	B00h–B0Bh	Core Register (Table 4-3)	B80h–B8Bh	Core Register (Table 4-3)
80Ch	WDTCON0	88Ch	CPUDOZE	90Ch	FVRCON	98Ch	—	A0Ch	—	A8Ch	—	B0Ch	—	B8Ch	—
80Dh	WDTCON1	88Dh	OSCCON1	90Dh	—	98Dh	—	A0Dh	—	A8Dh	—	B0Dh	—	B8Dh	—
80Eh	WDTL	88Eh	OSCCON2	90Eh	DAC1CON0	98Eh	—	A0Eh	—	A8Eh	—	B0Eh	—	B8Eh	—
80Fh	WDTH	88Fh	OSCCON3	90Fh	DAC1CON1	98Fh	CMOUT	A0Fh	—	A8Fh	—	B0Fh	—	B8Fh	—
810h	WDTU	890h	OSCSTAT1	910h	—	990h	CM1CON0	A10h	—	A90h	—	B10h	—	B90h	—
811h	BORCON	891h	OSCEN	911h	—	991h	CM1CON1	A11h	—	A91h	—	B11h	—	B91h	—
812h	VREGCON(2)	892h	OSCTUNE	912h	—	992h	CM1NCH	A12h	—	A92h	—	B12h	—	B92h	—
813h	PCON0	893h	OSCFRQ	913h	—	993h	CM1PCH	A13h	—	A93h	—	B13h	—	B93h	—
814h	PCON1	894h	—	914h	—	994h	CM2CON0	A14h	—	A94h	—	B14h	—	B94h	—
815h	—	895h	CLKRCON	915h	—	995h	CM2CON1	A15h	—	A95h	—	B15h	—	B95h	—
816h	—	896h	CLKCLK	916h	—	996h	CM2NCH	A16h	—	A96h	—	B16h	—	B96h	—
817h	—	897h	—	917h	—	997h	CM2PCH	A17h	—	A97h	—	B17h	—	B97h	—
818h	—	898h	—	918h	—	998h	—	A18h	—	A98h	—	B18h	—	B98h	—
819h	—	899h	—	919h	—	999h	—	A19h	RC2REG	A99h	—	B19h	—	B99h	—
81Ah	NVMADRL	89Ah	—	91Ah	—	99Ah	—	A1Ah	TX2REG	A9Ah	—	B1Ah	—	B9Ah	—
81Bh	NVMADRH	89Bh	—	91Bh	—	99Bh	—	A1Bh	SP2BRGL	A9Bh	—	B1Bh	—	B9Bh	—
81Ch	NVMDATL	89Ch	—	91Ch	—	99Ch	—	A1Ch	SP2BRGH	A9Ch	—	B1Ch	—	B9Ch	—
81Dh	NVMDATH	89Dh	—	91Dh	—	99Dh	—	A1Dh	RC2STA	A9Dh	—	B1Dh	—	B9Dh	—
81Eh	NVMCON1	89Eh	—	91Eh	—	99Eh	—	A1Eh	TX2STA	A9Eh	—	B1Eh	—	B9Eh	—
81Fh	NVMCON2	89Fh	—	91Fh	ZCDCON	99Fh	—	A1Fh	BAUD2CON	A9Fh	—	B1Fh	—	B9Fh	—
820h–86Fh	Unimplemented Read as '0'	8A0h–8EFh	Unimplemented Read as '0'	920h–96Fh	Unimplemented Read as '0'	9A0h–9EFh	Unimplemented Read as '0'	A20h–A6Fh	Unimplemented Read as '0'	AA0h–AEFh	Unimplemented Read as '0'	B20h–B6Fh	Unimplemented Read as '0'	BA0h–BEFh	Unimplemented Read as '0'
870h–87Fh	Common RAM Accesses 70h-7Fh	8F0h–8FFh	Common RAM Accesses 70h-7Fh	970h–97Fh	Common RAM Accesses 70h-7Fh	9F0h–9FFh	Common RAM Accesses 70h-7Fh	A70h–A7Fh	Common RAM Accesses 70h-7Fh	AF0h–AFFh	Common RAM Accesses 70h-7Fh	B70h–B7Fh	Common RAM Accesses 70h-7Fh	BF0h–BFFh	Common RAM Accesses 70h-7Fh

Note 1: Unimplemented locations read as '0'.
2: Register not implemented on LF devices.

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

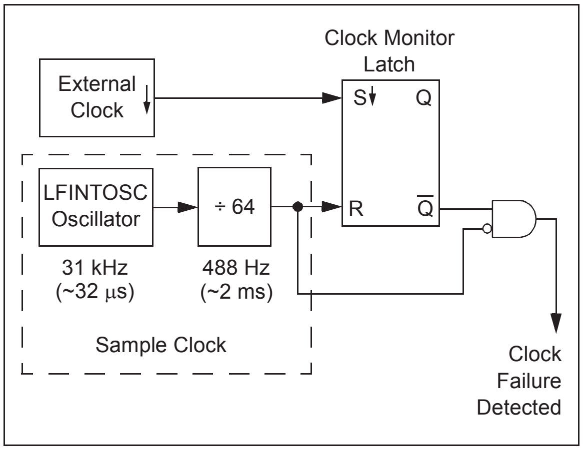
Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on: $\overline{\text{MCLR}}$
Bank 4											
CPU CORE REGISTERS; see Table 4-3 for specifics											
20Ch	TMR1L	Holding Register for the Least Significant Byte of the 16-bit TMR1 Register								0000 0000	uuuu uuuu
20Dh	TMR1H	Holding Register for the Most Significant Byte of the 16-bit TMR1 Register								0000 0000	uuuu uuuu
20Eh	T1CON	—	—	CKPS<1:0>		—	$\overline{\text{SYNC}}$	RD16	ON	--00 -000	--uu -u0u
20Fh	T1GCON	GE	GPOL	GTM	GSPM	GGO/$\overline{\text{DONE}}$	GVAL	—	—	0000 0x--	uuuu ux--
210h	T1GATE	—	—	—	GSS<4:0>					---0 0000	---u uuuu
211h	T1CLK	—	—	—	—	CS<3:0>				---- 0000	---- uuuu
212h — 21Fh	—	Unimplemented								—	—

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

9.4 Fail-Safe Clock Monitor

The Fail-Safe Clock Monitor (FSCM) allows the device to continue operating should the external oscillator fail. The FSCM is enabled by setting the FCMEN bit in the Configuration Words. The FSCM is applicable to all external Oscillator modes (LP, XT, HS, ECL, ECM, ECH and Secondary Oscillator).

FIGURE 9-8: FSCM BLOCK DIAGRAM



9.4.1 FAIL-SAFE DETECTION

The FSCM module detects a failed oscillator by comparing the external oscillator to the FSCM sample clock. The sample clock is generated by dividing the LFINTOSC by 64. See Figure 9-8. Inside the fail detector block is a latch. The external clock sets the latch on each falling edge of the external clock. The sample clock clears the latch on each rising edge of the sample clock. A failure is detected when an entire half-cycle of the sample clock elapses before the external clock goes low.

9.4.2 FAIL-SAFE OPERATION

When the external clock fails, the FSCM switches the device clock to the HFINTOSC at 1 MHz clock frequency and sets the bit flag OSFIF of the PIR1 register. Setting this flag will generate an interrupt if the OSFIE bit of the PIE1 register is also set. The device firmware can then take steps to mitigate the problems that may arise from a failed clock. The system clock will continue to be sourced from the internal clock source until the device firmware successfully restarts the external oscillator and switches back to external operation, by writing to the NOSC and NDIV bits of the OSCCON1 register.

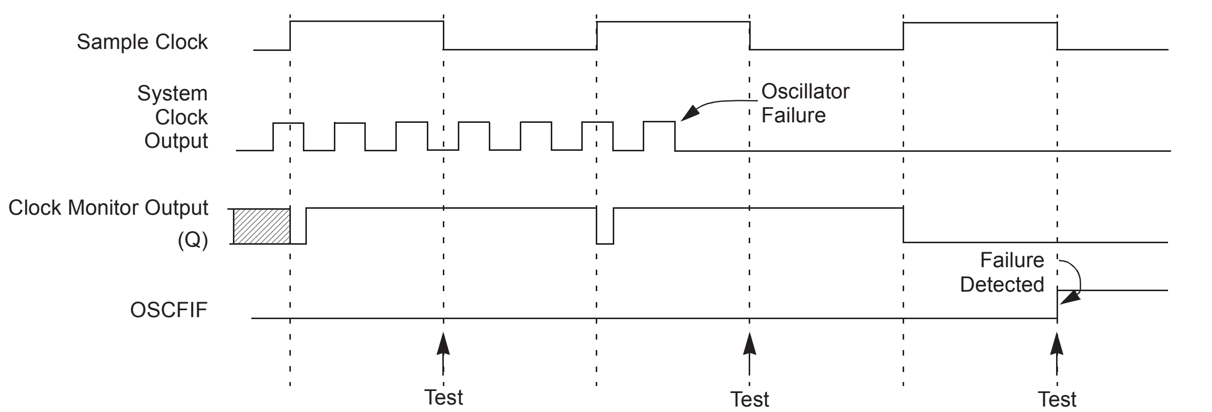
9.4.3 FAIL-SAFE CONDITION CLEARING

The Fail-Safe condition is cleared after a Reset, executing a `SLEEP` instruction or changing the NOSC and NDIV bits of the OSCCON1 register. When switching to the external oscillator, or external oscillator and PLL, the OST is restarted. While the OST is running, the device continues to operate from the INTOSC selected in OSCCON1. When the OST times out, the Fail-Safe condition is cleared after successfully switching to the external clock source. The OSFIF bit should be cleared prior to switching to the external clock source. If the Fail-Safe condition still exists, the OSFIF flag will again become set by hardware.

9.4.4 RESET OR WAKE-UP FROM SLEEP

The FSCM is designed to detect an oscillator failure after the Oscillator Start-up Timer (OST) has expired. The OST is used after waking up from Sleep and after any type of Reset. The OST is not used with the EC Clock modes so that the FSCM will be active as soon as the Reset or wake-up has completed. Therefore, the device will always be executing code while the OST is operating.

FIGURE 9-9: FSCM TIMING DIAGRAM



Note: The system clock is normally at a much higher frequency than the sample clock. The relative frequencies in this example have been chosen for clarity.

REGISTER 10-15: PIR5: PERIPHERAL INTERRUPT REQUEST REGISTER 5

R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	U-0	U-0	U-0	R/W/HS-0/0
CLC4IF	CLC3IF	CLC2IF	CLC1IF	—	—	—	TMR1GIF
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 set

bit 7 **CLC4IF:** CLC4 Interrupt Flag bit
1 = A CLC4OUT interrupt condition has occurred (must be cleared in software)
0 = No CLC4 interrupt event has occurred

bit 6 **CLC3IF:** CLC3 Interrupt Flag bit
1 = A CLC3OUT interrupt condition has occurred (must be cleared in software)
0 = No CLC3 interrupt event has occurred

bit 5 **CLC2IF:** CLC2 Interrupt Flag bit
1 = A CLC2OUT interrupt condition has occurred (must be cleared in software)
0 = No CLC2 interrupt event has occurred

bit 4 **CLC1IF:** CLC1 Interrupt Flag bit
1 = A CLC1OUT interrupt condition has occurred (must be cleared in software)
0 = No CLC1 interrupt event has occurred

bit 3-1 **Unimplemented:** Read as ‘0’

bit 0 **TMR1GIF:** Timer1 Gate Interrupt Flag bit
1 = The Timer1 Gate has gone inactive (the acquisition is complete)
0 = The Timer1 Gate has not gone inactive

> **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.

12.1 Independent Clock Source

The WDT can derive its time base from either the 31 kHz LFINTOSC or 31.25 kHz MFINTOSC internal oscillators, depending on the value of either the WDTCCS<2:0> Configuration bits or the WDTCS<2:0> bits of WDTCON1. Time intervals in this chapter are based on a minimum nominal interval of 1 ms. See **Section 37.0 "Electrical Specifications"** for LFINTOSC and MFINTOSC tolerances.

12.2 WDT Operating Modes

The Watchdog Timer module has four operating modes controlled by the WDTE<1:0> bits in Configuration Words. See Table 12-1.

12.2.1 WDT IS ALWAYS ON

When the WDTE bits of Configuration Words are set to '11', the WDT is always on.

WDT protection is active during Sleep.

12.2.2 WDT IS OFF IN SLEEP

When the WDTE bits of Configuration Words are set to '10', the WDT is on, except in Sleep.

WDT protection is not active during Sleep.

12.2.3 WDT CONTROLLED BY SOFTWARE

When the WDTE bits of Configuration Words are set to '01', the WDT is controlled by the SWDTEN bit of the WDTCON0 register.

12.2.4 WDT IS OFF

When the WDTE bits of the Configuration Word are set to '00', the WDT is always OFF.

WDT protection is unchanged by Sleep. See Table 12-1 for more details.

TABLE 12-1: WDT OPERATING MODES

WDTE<1:0>	SWDTEN	Device Mode	WDT Mode
11	x	X	Active
10	x	Awake	Active
		Sleep	Disabled
01	1	X	Active
	0	X	Disabled
00	x	X	Disabled

12.3 Time-Out Period

The WDTPS bits of the WDTCON0 register set the time-out period from 1 ms to 256 seconds (nominal). After a Reset, the default time-out period is two seconds.

12.4 Watchdog Window

The Watchdog Timer has an optional Windowed mode that is controlled by the WDTCWS<2:0> Configuration bits and WINDOW<2:0> bits of the WDTCON1 register. In the Windowed mode, the `CLRWDT` instruction must occur within the allowed window of the WDT period. Any `CLRWDT` instruction that occurs outside of this window will trigger a window violation and will cause a WDT Reset, similar to a WDT time out. See Figure 12-2 for an example.

The window size is controlled by the WDTCWS<2:0> Configuration bits, or the WINDOW<2:0> bits of WDTCON1, if WDTCWS<2:0> = 111.

In the event of a window violation, a Reset will be generated and the $\overline{\text{WDTWV}}$ bit of the PCON register will be cleared. This bit is set by a POR or can be set in firmware.

12.5 Clearing the WDT

The WDT is cleared when any of the following conditions occur:

- Any Reset
- Valid `CLRWDT` instruction is executed
- Device enters Sleep
- Device wakes up from Sleep
- WDT is disabled
- Oscillator Start-up Timer (OST) is running
- Any write to the WDTCON0 or WDTCON1 registers

12.5.1 `CLRWDT` CONSIDERATIONS (WINDOWED MODE)

When in Windowed mode, the WDT must be armed before a `CLRWDT` instruction will clear the timer. This is performed by reading the WDTCON0 register. Executing a `CLRWDT` instruction without performing such an arming action will trigger a window violation.

See Table 12-2 for more information.

12.6 Operation During Sleep

When the device enters Sleep, the WDT is cleared. If the WDT is enabled during Sleep, the WDT resumes counting. When the device exits Sleep, the WDT is cleared again.

The WDT remains clear until the OST, if enabled, completes. See **Section 9.0 "Oscillator Module (with Fail-Safe Clock Monitor)"** for more information on the OST.

When a WDT time-out occurs while the device is in Sleep, no Reset is generated. Instead, the device wakes up and resumes operation. The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits in the STATUS register are changed to indicate the event. The $\overline{\text{RWDT}}$ bit in the PCON register can also be used. See **Section 4.3.2.1 "STATUS Register"** for more information.

14.6 PORTC Registers

14.6.1 DATA REGISTER

PORTC is a 6 to 8-bit wide bidirectional port. The corresponding data direction register is TRISC (Register 14-18). Setting a TRISC bit (= 1) will make the corresponding PORTC pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISC bit (= 0) will make the corresponding PORTC pin an output (i.e., enable the output driver and put the contents of the output latch on the selected pin). Figure 14-1 shows how to initialize an I/O port.

Reading the PORTC register (Register 14-17) reads the status of the pins, whereas writing to it will write to the PORT latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified and then written to the PORT data latch (LATC).

The PORT data latch LATC (Register 14-19) holds the output port data, and contains the latest value of a LATC or PORTC write.

14.6.2 DIRECTION CONTROL

The TRISC register (Register 14-18) controls the PORTC pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISC register are maintained set when using them as analog inputs. I/O pins configured as analog inputs always read ‘0’.

14.6.3 OPEN-DRAIN CONTROL

The ODCONC register (Register 14-22) controls the open-drain feature of the port. Open-drain operation is independently selected for each pin. When an ODCONC bit is set, the corresponding port output becomes an open-drain driver capable of sinking current only. When an ODCONC bit is cleared, the corresponding port output pin is the standard push-pull drive capable of sourcing and sinking current.

Note:	It is not necessary to set open-drain control when using the pin for I^2C; the I^2C module controls the pin and makes the pin open-drain.

14.6.4 SLEW RATE CONTROL

The SLRCONC register (Register 14-23) controls the slew rate option for each port pin. Slew rate control is independently selectable for each port pin. When an SLRCONC bit is set, the corresponding port pin drive is slew rate limited. When an SLRCONC bit is cleared, The corresponding port pin drive slews at the maximum rate possible.

14.6.5 INPUT THRESHOLD CONTROL

The INLVLC register (Register 14-24) controls the input voltage threshold for each of the available PORTC input pins. A selection between the Schmitt Trigger CMOS or the TTL Compatible thresholds is available. The input threshold is important in determining the value of a read of the PORTC register and also the level at which an interrupt-on-change occurs, if that feature is enabled. See Table 37-4 for more information on threshold levels.

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

14.6.6 ANALOG CONTROL

The ANSELC register (Register 14-20) is used to configure the Input mode of an I/O pin to analog. Setting the appropriate ANSELC bit high will cause all digital reads on the pin to be read as ‘0’ and allow analog functions on the pin to operate correctly.

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

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

14.6.7 WEAK PULL-UP CONTROL

The WPUC register (Register 14-21) controls the individual weak pull-ups for each port pin.

14.6.8 PORTC FUNCTIONS AND OUTPUT PRIORITIES

Each pin defaults to the PORT latch data after Reset. Other output functions are selected with the peripheral pin select logic. See **Section 15.0 “Peripheral Pin Select (PPS) Module”** for more information.

Analog input functions, such as ADC and comparator inputs, are not shown in the peripheral pin select lists. Digital output functions may continue to control the pin when it is in Analog mode.

20.1.5 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 1:** The ADIF bit is set at the completion of every conversion, regardless of whether or not the ADC interrupt is enabled.
>
> **2:** The ADC operates during Sleep only when the ADCRC oscillator is selected.

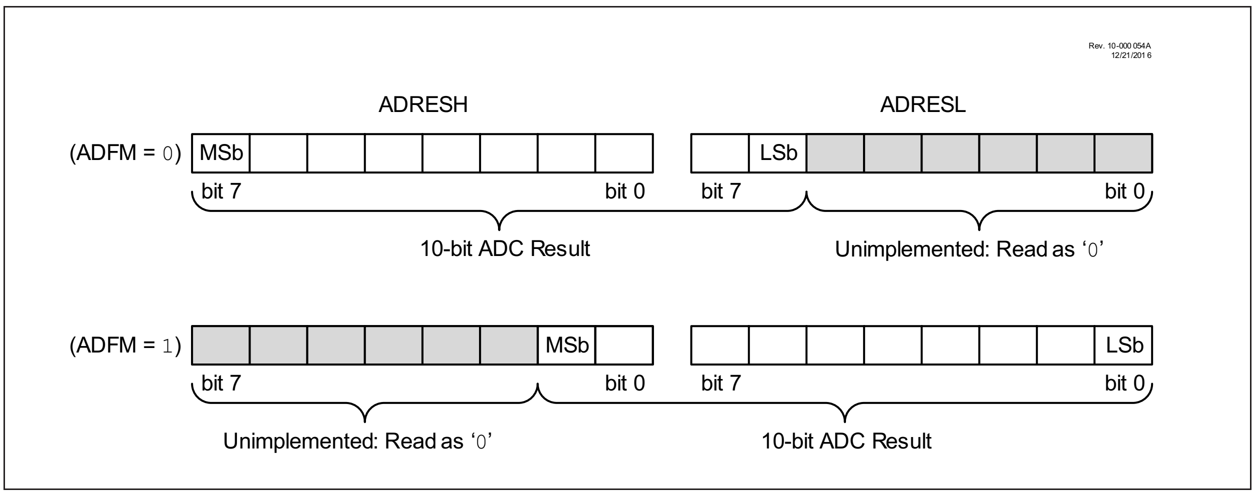
This interrupt can be generated while the device is operating or while in Sleep. If the device is in Sleep, the interrupt will wake-up the device. Upon waking from Sleep, the next instruction following the `SLEEP` instruction is always executed. If the user is attempting to wake-up from Sleep and resume in-line code execution, the ADIE bit of the PIE1 register and the PEIE bit of the INTCON register must both be set and the GIE bit of the INTCON register must be cleared. If all three of these bits are set, the execution will switch to the Interrupt Service Routine (ISR).

20.1.6 RESULT FORMATTING

The 10-bit ADC conversion result can be supplied in two formats, left justified or right justified. The ADFM bit of the ADCON1 register controls the output format.

Figure 20-3 shows the two output formats.

FIGURE 20-3: 10-BIT ADC CONVERSION RESULT FORMAT



26.0 TIMER1 MODULE WITH GATE CONTROL

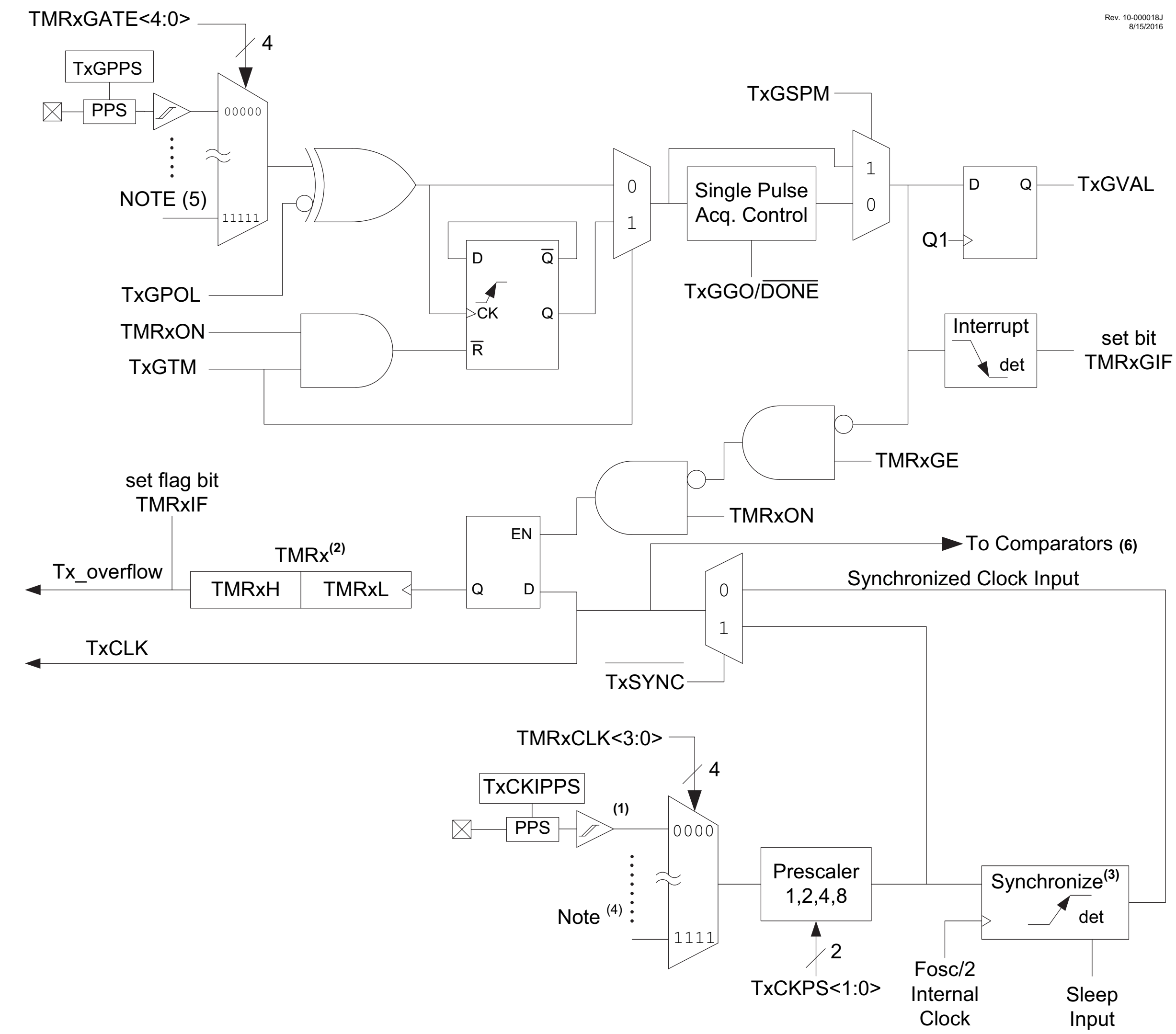
The Timer1 module is 16-bit timer/counters with the following features:

- 16-bit timer/counter register pair (TMR1H:TMR1L)
- Programmable internal or external clock source
- 2-bit prescaler
- Clock source for optional comparator synchronization
- Multiple Timer1 gate (count enable) sources
- Interrupt on overflow
- Wake-up on overflow (external clock, Asynchronous mode only)
- Time base for the Capture/Compare function
- Auto-conversion Trigger (with CCP)
- Selectable Gate Source Polarity
- Gate Toggle mode
- Gate Single-Pulse mode
- Gate Value Status
- Gate Event Interrupt

Figure 26-1 is a block diagram of the Timer1 module.

This device has one instance of Timer1 type modules.

FIGURE 26-1: TIMER1 BLOCK DIAGRAM



Note 1: ST Buffer is high speed type when using TxCKIPPS.

2: TMRx register increments on rising edge.

3: Synchronize does not operate while in Sleep.

4: See Register 26-3 for Clock source selections.

5: See Register 26-4 for GATE source selections.

6: Synchronized comparator output should not be used in conjunction with synchronized input clock.

26.5.2 TIMER GATE SOURCE SELECTION

One of the several different external or internal signal sources may be chosen to gate the timer and allow the timer to increment. The gate input signal source can be selected based on the T1GATE register setting. See the T1GATE register (Register 26-4) description for a complete list of the available gate sources. The polarity for each available source is also selectable. Polarity selection is controlled by the GPOL bit of the T1GCON register.

26.5.2.1 T1G Pin Gate Operation

The T1G pin is one source for the timer gate control. It can be used to supply an external source to the time gate circuitry.

26.5.2.2 Timer0 Overflow Gate Operation

When Timer0 overflows, or a period register match condition occurs (in 8-bit mode), a low-to-high pulse will automatically be generated and internally supplied to the Timer1 gate circuitry.

26.5.2.3 Comparator C1 Gate Operation

The output resulting from a Comparator 1 operation can be selected as a source for the timer gate control. The Comparator 1 output can be synchronized to the timer clock or left asynchronous. For more information see **Section 23.4.1 "Comparator Output Synchronization"**.

26.5.2.4 Comparator C2 Gate Operation

The output resulting from a Comparator 2 operation can be selected as a source for the timer gate control. The Comparator 2 output can be synchronized to the timer clock or left asynchronous. For more information see **Section 23.4.1 "Comparator Output Synchronization"**.

26.5.3 TIMER1 GATE TOGGLE MODE

When Timer1 Gate Toggle mode is enabled, it is possible to measure the full-cycle length of a timer gate signal, as opposed to the duration of a single level pulse.

The timer gate source is routed through a flip-flop that changes state on every incrementing edge of the signal. See Figure 26-4 for timing details.

Timer1 Gate Toggle mode is enabled by setting the GTM bit of the T1GCON register. When the GTM bit is cleared, the flip-flop is cleared and held clear. This is necessary in order to control which edge is measured.

Note:	Enabling Toggle mode at the same time as changing the gate polarity may result in indeterminate operation.

26.5.4 TIMER1 GATE SINGLE-PULSE MODE

When Timer1 Gate Single-Pulse mode is enabled, it is possible to capture a single-pulse gate event. Timer1 Gate Single-Pulse mode is first enabled by setting the GSPM bit in the T1GCON register. Next, the GGO/$\overline{\text{DONE}}$ bit in the T1GCON register must be set. The timer will be fully enabled on the next incrementing edge. On the next trailing edge of the pulse, the GGO/$\overline{\text{DONE}}$ bit will automatically be cleared. No other gate events will be allowed to increment the timer until the GGO/$\overline{\text{DONE}}$ bit is once again set in software. See Figure 26-5 for timing details.

If the Single-Pulse Gate mode is disabled by clearing the GSPM bit in the T1GCON register, the GGO/$\overline{\text{DONE}}$ bit should also be cleared.

Enabling the Toggle mode and the Single-Pulse mode simultaneously will permit both sections to work together. This allows the cycle times on the timer gate source to be measured. See Figure 26-6 for timing details.

26.5.5 TIMER1 GATE VALUE STATUS

When Timer1 Gate Value Status is utilized, it is possible to read the most current level of the gate control value. The value is stored in the GVAL bit in the T1GCON register. The GVAL bit is valid even when the timer gate is not enabled (GE bit is cleared).

26.5.6 TIMER1 GATE EVENT INTERRUPT

When Timer1 Gate Event Interrupt is enabled, it is possible to generate an interrupt upon the completion of a gate event. When the falling edge of GVAL occurs, the TMR1GIF flag bit in the PIR5 register will be set. If the TMR1GIE bit in the PIE5 register is set, then an interrupt will be recognized.

The TMR1GIF flag bit operates even when the timer gate is not enabled (TMR1GE bit is cleared).

REGISTER 26-2: T1GCON: TIMER1 GATE CONTROL REGISTER

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W/HC-0/u	R-x/x	U-0	U-0
GE	GPOL	GTM	GSPM	GGO/$\overline{\text{DONE}}$	GVAL	—	—
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	HC = Bit is cleared by hardware

bit 7 **GE:** Timer1 Gate Enable bit
If ON = 0:
This bit is ignored
If ON = 1:
1 = Timer1 counting is controlled by the Timer1 gate function
0 = Timer1 is always counting

bit 6 **GPOL:** Timer1 Gate Polarity bit
1 = Timer1 gate is active-high (Timer1 counts when gate is high)
0 = Timer1 gate is active-low (Timer1 counts when gate is low)

bit 5 **GTM:** Timer1 Gate Toggle Mode bit
1 = Timer1 Gate Toggle mode is enabled
0 = Timer1 Gate Toggle mode is disabled and toggle flip-flop is cleared
Timer1 gate flip-flop toggles on every rising edge.

bit 4 **GSPM:** Timer1 Gate Single-Pulse Mode bit
1 = Timer1 Gate Single-Pulse mode is enabled
0 = Timer1 Gate Single-Pulse mode is disabled

bit 3 **GGO/$\overline{\text{DONE}}$:** Timer1 Gate Single-Pulse Acquisition Status bit
1 = Timer1 gate single-pulse acquisition is ready, waiting for an edge
0 = Timer1 gate single-pulse acquisition has completed or has not been started
This bit is automatically cleared when GSPM is cleared

bit 2 **GVAL:** Timer1 Gate Value Status bit
Indicates the current state of the Timer1 gate that could be provided to TMR1H:TMR1L
Unaffected by Timer1 Gate Enable (GE)

bit 1-0 **Unimplemented:** Read as ‘0’

29.1.1 PWM CLOCK SELECTION

The PIC16(L)F15324/44 allows each individual CCP and PWM module to select the timer source that controls the module. Each module has an independent selection.

29.1.2 USING THE TMR2 WITH THE PWM MODULE

This device has a newer version of the TMR2 module that has many new modes, which allow for greater customization and control of the PWM signals than on older parts. Refer to **Section 27.5 "Operation Examples"** for examples of PWM signal generation using the different modes of Timer2.

Note:	PWM operation requires that the timer used as the PWM time base has the FOSC/4 clock source selected.

29.1.3 PWM PERIOD

Referring to Figure 29-1, the PWM output has a period and a pulse width. The frequency of the PWM is the inverse of the period (1/period).

The PWM period is specified by writing to the PR2 register. The PWM period can be calculated using the following formula:

EQUATION 29-1: PWM PERIOD

$$PWM\ Period = [(PR2) + 1] \cdot 4 \cdot TOSC \cdot (TMR2\ Prescale\ Value)$$

Note 1: Tosc = 1/Fosc

When TMR2 is equal to PR2, the following three events occur on the next increment cycle:

- TMR2 is cleared
- The PWMx pin is set (Exception: If the PWM duty cycle = 0%, the pin will not be set.)
- The PWM pulse width is latched from PWMxDC.

Note:	If the pulse width value is greater than the period the assigned PWM pin(s) will remain unchanged.

29.1.4 PWM DUTY CYCLE

The PWM duty cycle is specified by writing a 10-bit value to the PWMxDC register. The PWMxDCH contains the eight MSbs and the PWMxDCL<7:6> bits contain the two LSbs.

The PWMDC register is double-buffered and can be updated at any time. This double buffering is essential for glitch-free PWM operation. New values take effect when TMR2 = PR2. Note that PWMDC is left-justified.

The 8-bit timer TMR2 register is concatenated with either the 2-bit internal system clock (FOSC), or two bits of the prescaler, to create the 10-bit time base. The system clock is used if the Timer2 prescaler is set to 1:1.

Equation 29-2 is used to calculate the PWM pulse width.

Equation 29-3 is used to calculate the PWM duty cycle ratio.

EQUATION 29-2: PULSE WIDTH

$$Pulse\ Width = (PWMxDC) \cdot TOSC \cdot (TMR2\ Prescale\ Value)$$

EQUATION 29-3: DUTY CYCLE RATIO

$$Duty\ Cycle\ Ratio = \frac{(PWMxDC)}{4(PR2 + 1)}$$

29.1.5 PWM RESOLUTION

The resolution determines the number of available duty cycles for a given period. For example, a 10-bit resolution will result in 1024 discrete duty cycles, whereas an 8-bit resolution will result in 256 discrete duty cycles.

The maximum PWM resolution is ten bits when PR2 is 255. The resolution is a function of the PR2 register value as shown by Equation 29-4.

EQUATION 29-4: PWM RESOLUTION

$$Resolution = \frac{\log[4(PR2 + 1)]}{\log(2)}\ bits$$

TABLE 30-3: SUMMARY OF REGISTERS ASSOCIATED WITH CWG

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
CWG1CLKCON	—	—	—	—	—	—	—	CS	357
CWG1ISM	—	—	—	—	IS<3:0>				357
CWG1DBR	—	—	DBR<5:0>						353
CWG1DBF	—	—	DBF<5:0>						353
CWG1CON0	EN	LD	—	—	—	MODE<2:0>			356
CWG1CON1	—	—	IN	—	POLD	POLC	POLB	POLA	352
CWG1AS0	SHUTDOWN	REN	LSBD<1:0>		LSAC<1:0>		—	—	354
CWG1AS1	—	—	—	AS4E	AS3E	AS2E	AS1E	AS0E	355
CWG1STR	OVRD	OVRC	OVRB	OVRA	STRD	STRC	STRB	STRA	356

Legend: – = unimplemented locations read as ‘0’. Shaded cells are not used by CWG.

REGISTER 31-2: CLCxPOL: SIGNAL POLARITY CONTROL REGISTER

R/W-0/0	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 Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7 **LCxPOL:** CLCxOUT Output 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 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 2 **LCxG3POL:** 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 1 **LCxG2POL:** 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

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

32.6.10 SLEEP OPERATION

While in Sleep mode, the I²C slave module can receive addresses or data and when an address match or complete byte transfer occurs, wake the processor from Sleep (if the MSSP interrupt is enabled).

32.6.11 EFFECTS OF A RESET

A Reset disables the MSSP module and terminates the current transfer.

32.6.12 MULTI-MASTER MODE

In Multi-Master mode, the interrupt generation on the detection of the Start and Stop conditions allows the determination of when the bus is free. The Stop (P) and Start (S) bits are cleared from a Reset or when the MSSP module is disabled. Control of the I²C bus may be taken when the P bit of the SSP1STAT register is set, or the bus is Idle, with both the S and P bits clear. When the bus is busy, enabling the SSP interrupt will generate the interrupt when the Stop condition occurs.

In multi-master operation, the SDA line must be monitored for arbitration to see if the signal level is the expected output level. This check is performed by hardware with the result placed in the BCL1IF bit.

The states where arbitration can be lost are:

- Address Transfer
- Data Transfer
- A Start Condition
- A Repeated Start Condition
- An Acknowledge Condition

32.6.13 MULTI -MASTER COMMUNICATION, BUS COLLISION AND BUS ARBITRATION

Multi-Master mode support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA, by letting SDA float high and another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin is '0', then a bus collision has taken place. The master will set the Bus Collision Interrupt Flag, BCL1IF and reset the I²C port to its Idle state (Figure 32-32).

If a transmit was in progress when the bus collision occurred, the transmission is halted, the BF flag is cleared, the SDA and SCL lines are deasserted and the SSP1BUF can be written to. When the user services the bus collision Interrupt Service Routine and if the I²C bus is free, the user can resume communication by asserting a Start condition.

If a Start, Repeated Start, Stop or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are deasserted and the respective control bits in the SSP1CON2 register are cleared. When the user services the bus collision Interrupt Service Routine and if the I²C bus is free, the user can resume communication by asserting a Start condition.

The master will continue to monitor the SDA and SCL pins. If a Stop condition occurs, the SSP1IF bit will be set.

A write to the SSP1BUF will start the transmission of data at the first data bit, regardless of where the transmitter left off when the bus collision occurred.

In Multi-Master mode, the interrupt generation on the detection of Start and Stop conditions allows the determination of when the bus is free. Control of the I²C bus can be taken when the P bit is set in the SSP1STAT register, or the bus is Idle and the S and P bits are cleared.

FIGURE 32-32: BUS COLLISION TIMING FOR TRANSMIT AND ACKNOWLEDGE

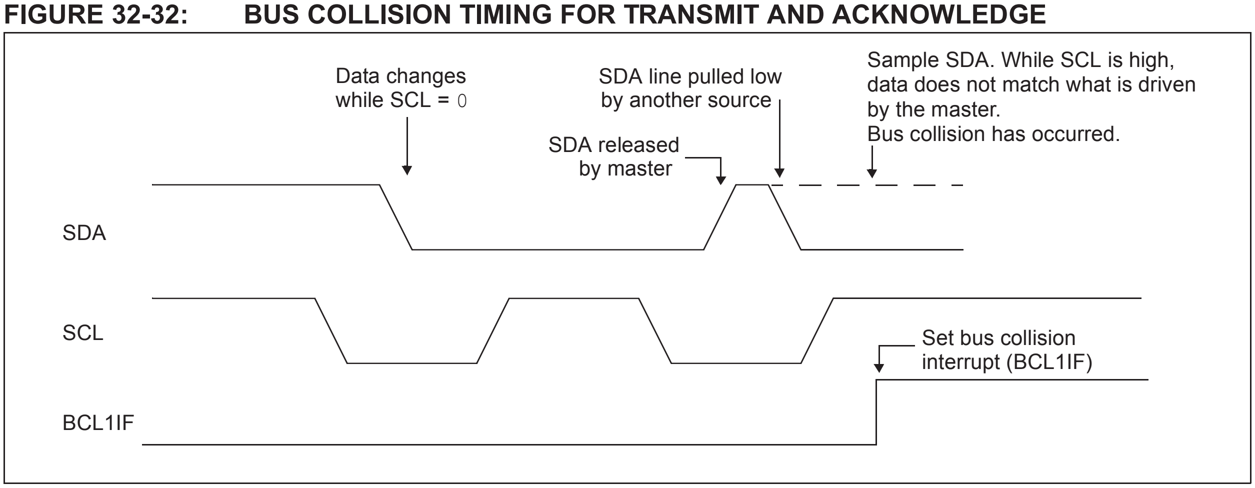


FIGURE 35-2: PICkit™ PROGRAMMER STYLE CONNECTOR INTERFACE

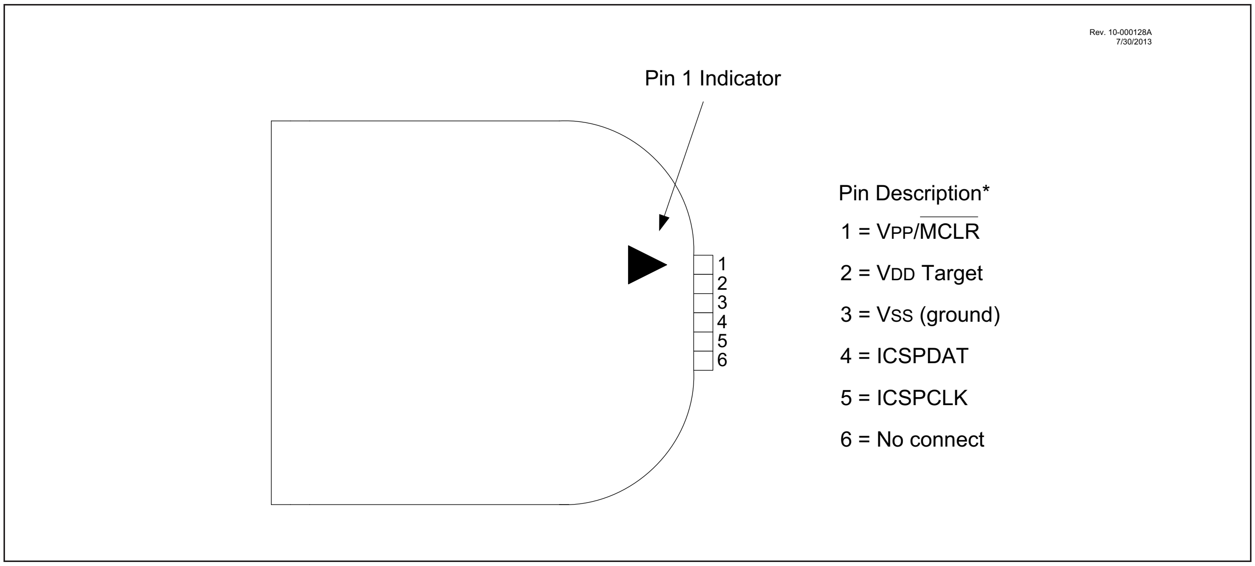
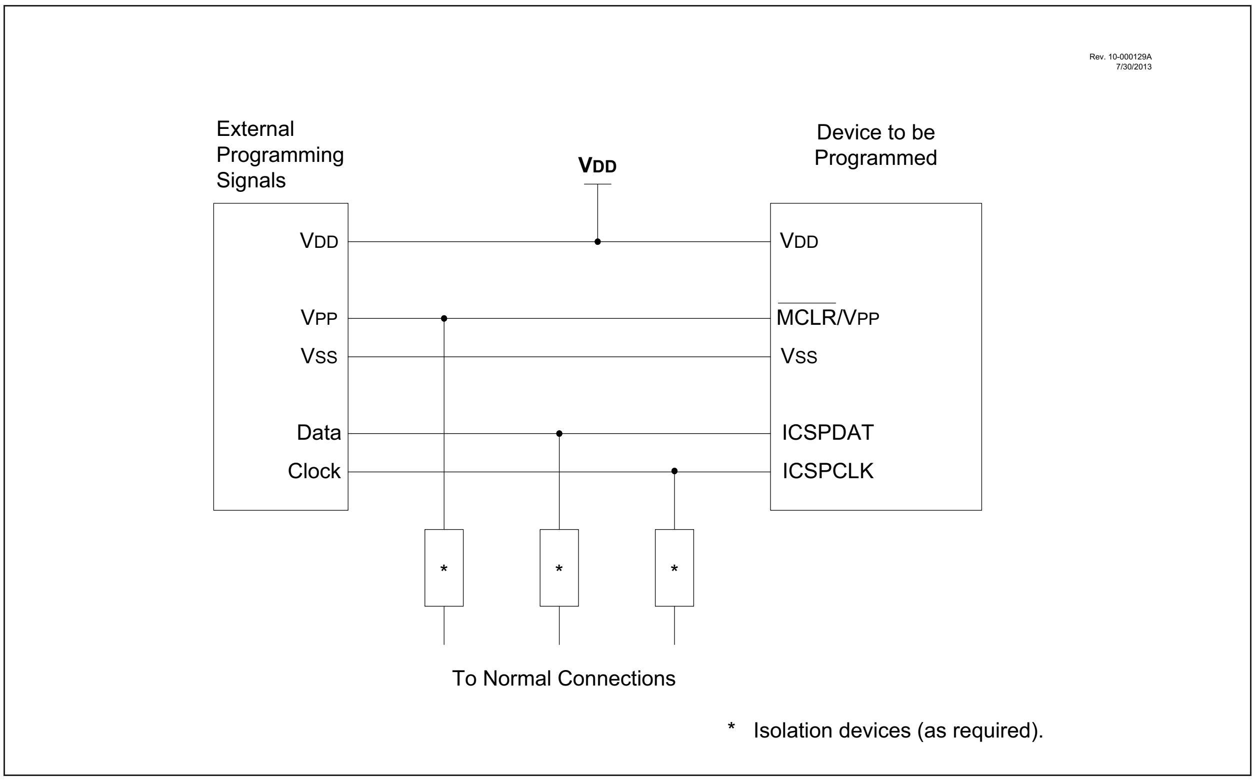


FIGURE 35-3: TYPICAL CONNECTION FOR ICSP™ PROGRAMMING



36.3 Instruction Descriptions

ADDFSR — Add Literal to FSRn

Syntax:	[*label*] ADDFSR FSRn, k
Operands:	-32 ≤ k ≤ 31 n ∈ [0, 1]
Operation:	FSR(n) + k → FSR(n)
Status Affected:	None
Description:	The signed 6-bit literal 'k' is added to the contents of the FSRnH:FSRnL register pair. FSRn is limited to the range 0000h-FFFFh. Moving beyond these bounds will cause the FSR to wrap-around.

ADDLW — Add literal and W

Syntax:	[*label*] ADDLW k
Operands:	0 ≤ k ≤ 255
Operation:	(W) + k → (W)
Status Affected:	C, DC, Z
Description:	The contents of the W register are added to the 8-bit literal 'k' and the result is placed in the W register.

ADDWF — Add W and f

Syntax:	[*label*] ADDWF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(W) + (f) → (destination)
Status Affected:	C, DC, Z
Description:	Add the contents of the W register with 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'.

ADDWFC — ADD W and CARRY bit to f

Syntax:	[*label*] ADDWFC f {,d}
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(W) + (f) + (C) → dest
Status Affected:	C, DC, Z
Description:	Add W, the Carry flag and data memory location 'f'. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed in data memory location 'f'.

ANDLW — AND literal with W

Syntax:	[*label*] ANDLW k
Operands:	0 ≤ k ≤ 255
Operation:	(W) .AND. (k) → (W)
Status Affected:	Z
Description:	The contents of W register are AND'ed with the 8-bit literal 'k'. The result is placed in the W register.

ANDWF — AND W with f

Syntax:	[*label*] ANDWF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(W) .AND. (f) → (destination)
Status Affected:	Z
Description:	AND the W register with 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'.

ASRF — Arithmetic Right Shift

Syntax:	[*label*] ASRF f {,d}
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f<7>)→ dest<7> (f<7:1>) → dest<6:0>, (f<0>) → C,
Status Affected:	C, Z
Description:	The contents of register 'f' are shifted one bit to the right through the Carry flag. The MSb remains unchanged. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is stored back in register 'f'.

BCF	Bit Clear f
Syntax:	[*label*] BCF f,b
Operands:	$0 \leq f \leq 127$ $0 \leq b \leq 7$
Operation:	`0` → (f<b>)
Status Affected:	None
Description:	Bit ‘b’ in register ‘f’ is cleared.

BRA	Relative Branch
Syntax:	[*label*] BRA label [*label*] BRA $+k
Operands:	$-256 \leq label - PC + 1 \leq 255$ $-256 \leq k \leq 255$
Operation:	(PC) + 1 + k → PC
Status Affected:	None
Description:	Add the signed 9-bit literal ‘k’ to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 1 + k. This instruction is a 2-cycle instruction. This branch has a limited range.

BRW	Relative Branch with W
Syntax:	[*label*] BRW
Operands:	None
Operation:	(PC) + (W) → PC
Status Affected:	None
Description:	Add the contents of W (unsigned) to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 1 + (W). This instruction is a 2-cycle instruction.

BSF	Bit Set f
Syntax:	[*label*] BSF f,b
Operands:	$0 \leq f \leq 127$ $0 \leq b \leq 7$
Operation:	`1` → (f<b>)
Status Affected:	None
Description:	Bit ‘b’ in register ‘f’ is set.

BTFSC	Bit Test f, Skip if Clear
Syntax:	[*label*] BTFSC f,b
Operands:	$0 \leq f \leq 127$ $0 \leq b \leq 7$
Operation:	skip if (f<b>) = `0`
Status Affected:	None
Description:	If bit ‘b’ in register ‘f’ is ‘`1`’, the next instruction is executed. If bit ‘b’, in register ‘f’, is ‘`0`’, the next instruction is discarded, and a `NOP` is executed instead, making this a 2-cycle instruction.

BTFSS	Bit Test f, Skip if Set
Syntax:	[*label*] BTFSS f,b
Operands:	$0 \leq f \leq 127$ $0 \leq b < 7$
Operation:	skip if (f<b>) = `1`
Status Affected:	None
Description:	If bit ‘b’ in register ‘f’ is ‘`0`’, the next instruction is executed. If bit ‘b’ is ‘`1`’, then the next instruction is discarded and a `NOP` is executed instead, making this a 2-cycle instruction.

TABLE 37-11: RESET, WDT, OSCILLATOR START-UP TIMER, POWER-UP TIMER, BROWN-OUT RESET AND LOW-POWER BROWN-OUT RESET SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
RST01*	TMCLR	$\overline{\text{MCLR}}$ Pulse Width Low to ensure Reset	2	—	—	µs	
RST02*	TIOZ	I/O high-impedance from Reset detection	—	—	2	µs	
RST03	TWDT	Watchdog Timer Time-out Period	—	16	—	ms	16 ms Nominal Reset Time
RST04*	TPWRT	Power-up Timer Period	—	65	—	ms	
RST05	TOST	Oscillator Start-up Timer Period(1,2)	—	1024	—	TOSC	
RST06	VBOR	Brown-out Reset Voltage(4)	2.55 2.30 1.80	2.70 2.45 1.90	2.85 2.60 2.10	V V V	BORV = 0 BORV = 1 (F devices) BORV = 1 (LF devices)
RST07	VBORHYS	Brown-out Reset Hysteresis	—	40	—	mV	
RST08	TBORDC	Brown-out Reset Response Time	—	3	—	µs	
RST09	VLPBOR	Low-Power Brown-out Reset Voltage	1.8	1.9	2.2	V	LF Devices Only

* 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.

Note 1: By design, the Oscillator Start-up Timer (OST) counts the first 1024 cycles, independent of frequency.

2: To ensure these voltage tolerances, VDD and VSS must be capacitively decoupled as close to the device as possible. 0.1 µF and 0.01 µF values in parallel are recommended.

TABLE 37-12: ANALOG-TO-DIGITAL CONVERTER (ADC) ACCURACY SPECIFICATIONS(1,2):

Standard Operating Conditions (unless otherwise stated) VDD = 3.0V, TA = 25°C							
Param. No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
AD01	NR	Resolution	—	—	10	bit	
AD02	EIL	Integral Error	—	±0.1	±1.0	LSb	ADCREF+ = 3.0V, ADCREF-= 0V
AD03	EDL	Differential Error	—	±0.1	±1.0	LSb	ADCREF+ = 3.0V, ADCREF-= 0V
AD04	EOFF	Offset Error	—	0.5	2.0	LSb	ADCREF+ = 3.0V, ADCREF-= 0V
AD05	EGN	Gain Error	—	±0.2	±1.0	LSb	ADCREF+ = 3.0V, ADCREF-= 0V
AD06	VADREF	ADC Reference Voltage (ADREF+ - ADREF-)	1.8	—	VDD	V	
AD07	VAIN	Full-Scale Range	ADREF-	—	ADREF+	V	
AD08	ZAIN	Recommended Impedance of Analog Voltage Source	—	10	—	kΩ	
AD09	RVREF	ADC Voltage Reference Ladder Impedance	—	50	—	kΩ	**Note 3**

* 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.

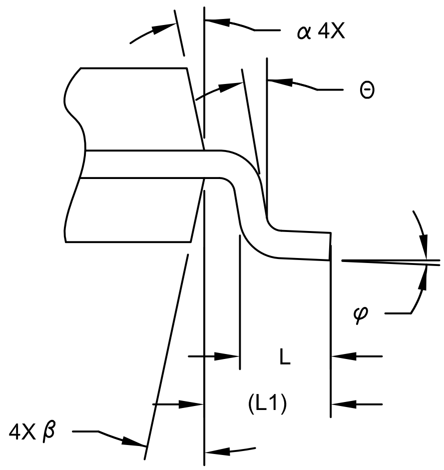
Note 1: Total Absolute Error is the sum of the offset, gain and integral non-linearity (INL) errors.

2: The ADC conversion result never decreases with an increase in the input and has no missing codes.

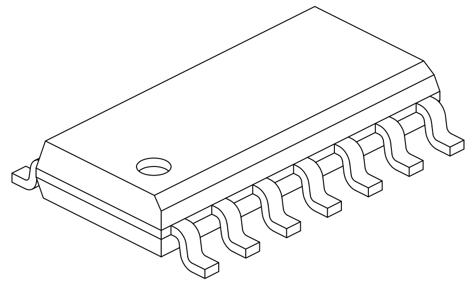
3: This is the impedance seen by the VREF pads when the external reference pads are selected.

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