

Welcome to [E-XFL.COM](https://www.e-xfl.com)

### What is "[Embedded - Microcontrollers](#)"?

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

### Applications of "[Embedded - Microcontrollers](#)"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I <sup>2</sup> C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	18
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	224 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 17x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Through Hole
Package / Case	20-DIP (0.300", 7.62mm)
Supplier Device Package	20-PDIP
Purchase URL	<a href="https://www.e-xfl.com/product-detail/microchip-technology/pic16lf15345-i-p">https://www.e-xfl.com/product-detail/microchip-technology/pic16lf15345-i-p</a>

# PIC16(L)F15325/45

**TABLE 1-2: PIC16(L)F15325 PINOUT DESCRIPTION (CONTINUED)**

Name	Function	Input Type	Output Type	Description
RC3/ANC3/C1IN3-/C2IN3-/CCP2 <sup>(1)</sup> / SS1 <sup>(1)</sup> /CLCIN0 <sup>(1)</sup> /IOCC3	RC3	TTL/ST	CMOS/OD	General purpose I/O.
	ANC3	AN	—	ADC Channel C3 input.
	C1IN3-	AN	—	Comparator 1 positive input.
	C2IN3-	AN	—	Comparator 2 positive input.
	CCP2 <sup>(1)</sup>	TTL/ST	CMOS/OD	Capture/compare/PWM2 (default input location for capture function).
	SS1 <sup>(1)</sup>	TTL/ST	—	MSSP1 SPI slave select input.
	CLCIN0 <sup>(1)</sup>	TTL/ST	—	Configurable Logic Cell source input.
RC4/ANC4/TX1 <sup>(1)</sup> /CK1 <sup>(1)</sup> /CLCIN1 <sup>(1)</sup> / IOCC4	RC4	TTL/ST	CMOS/OD	General purpose I/O.
	ANC4	AN	—	ADC Channel C4 input.
	TX1	—	CMOS	EUSART1 asynchronous transmit.
	CLCIN1 <sup>(1)</sup>	TTL/ST	—	Configurable Logic Cell source input.
	CK1 <sup>(1)</sup>	TTL/ST	CMOS/OD	EUSART1 synchronous mode clock input/output.
	IOCC4	TTL/ST	—	Interrupt-on-change input.
RC5/ANC5/CCP1 <sup>(1)</sup> /RX1 <sup>(1)</sup> /DT1 <sup>(1)</sup> / IOCC5	RC5	TTL/ST	CMOS/OD	General purpose I/O.
	ANC5	AN	—	ADC Channel C5 input.
	CCP1 <sup>(1)</sup>	TTL/ST	CMOS/OD	Capture/compare/PWM1 (default input location for capture function).
	RX1 <sup>(1)</sup>	TTL/ST	—	EUSART1 Asynchronous mode receiver data input.
	DT1 <sup>(1)</sup>	TTL/ST	CMOS/OD	EUSART1 Synchronous mode data input/output.
	IOCC5	TTL/ST	—	Interrupt-on-change input.
VDD	Vbd	Power	—	Positive supply voltage input.
VSS	VSS	Power	—	Ground reference.

**Legend:** AN = Analog input or output    CMOS = CMOS compatible input or output    OD = Open-Drain  
TTL = TTL compatible input    ST = Schmitt Trigger input with CMOS levels    I<sup>2</sup>C = Schmitt Trigger input with I<sup>2</sup>C  
HV = High Voltage    XTAL = Crystal levels

- Note**
- 1: This is a PPS remappable input signal. The input function may be moved from the default location shown to one of several other PORTx pins. Refer to Table 15-3 for details on which PORT pins may be used for this signal.
  - 2: All output signals shown in this row are PPS remappable. These signals may be mapped to output onto one of several PORTx pin options as described in Table 15-3.
  - 3: This is a bidirectional signal. For normal module operation, the firmware should map this signal to the same pin in both the PPS input and PPS output registers.
  - 4: These pins are configured for I<sup>2</sup>C logic levels. The SCLx/SDAx signals may be assigned to any of the RB1/RB2/RC3/RC4 pins. PPS assignments to the other pins (e.g., RA5) will operate, but input logic levels will be standard TTL/ST, as selected by the INLVL register, instead of the I<sup>2</sup>C specific or SMBus input buffer thresholds.

**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
<b>Bank 21-59</b>											
CPU CORE REGISTERS; see Table 4-3 for specifics											
x0Ch/ x8Ch — x1Fh/ x9Fh	—	Unimplemented								—	—

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

## 9.2.2.2 Internal Oscillator Frequency Adjustment

The internal oscillator is factory-calibrated. This internal oscillator can be adjusted in software by writing to the OSCTUNE register (Register 9-7).

The default value of the OSCTUNE register is 00h. The value is a 6-bit two's complement number. A value of 1Fh will provide an adjustment to the maximum frequency. A value of 20h will provide an adjustment to the minimum frequency.

When the OSCTUNE register is modified, the oscillator frequency will begin shifting to the new frequency. Code execution continues during this shift. There is no indication that the shift has occurred.

OSCTUNE does not affect the LFINTOSC frequency. Operation of features that depend on the LFINTOSC clock source frequency, such as the Power-up Timer (PWRT), Watchdog Timer (WDT), Fail-Safe Clock Monitor (FSCM) and peripherals, are *not* affected by the change in frequency.

## 9.2.2.3 LFINTOSC

The Low-Frequency Internal Oscillator (LFINTOSC) is a factory calibrated 31 kHz internal clock source.

The LFINTOSC is the frequency for the Power-up Timer (PWRT), Windowed Watchdog Timer (WWDT) and Fail-Safe Clock Monitor (FSCM).

The LFINTOSC is enabled through one of the following methods:

- Programming the RSTOSC<2:0> bits of Configuration Word 1 to enable LFINTOSC.
- Write to the NOSC<2:0> bits of the OSCCON1 register.

Peripherals that use the LFINTOSC are:

- Power-up Timer (PWRT)
- Windowed Watchdog Timer (WWDT)
- Timer1
- Timer0
- Timer2
- Fail-Safe Clock Monitor (FSCM)
- CLKR
- CLC

## 9.2.2.4 Oscillator Status and Manual Enable

The 'ready' status of each oscillator is displayed in the OSCSTAT register (Register 9-4). The oscillators can also be manually enabled through the OSCEN register (Register 9-7). Manual enabling makes it possible to verify the operation of the EXTOSC or SOSC crystal oscillators. This can be achieved by enabling the selected oscillator, then watching the corresponding 'ready' state of the oscillator in the OSCSTAT register.

## 9.3 Clock Switching

The system clock source can be switched between external and internal clock sources via software using the New Oscillator Source (NOSC) and New Divider selection request (NDIV) bits of the OSCCON1 register.

### 9.3.1 NEW OSCILLATOR SOURCE (NOSC) AND NEW DIVIDER SELECTION REQUEST (NDIV) BITS

The New Oscillator Source (NOSC) and New Divider selection request (NDIV) bits of the OSCCON1 register select the system clock source and the frequency that are used for the CPU and peripherals.

When new values of NOSC and NDIV are written to OSCCON1, the current oscillator selection will continue to operate while waiting for the new clock source to indicate that it is stable and ready. In some cases, the newly requested source may already be in use, and is ready immediately. In the case of a divider-only change, the new and old sources are the same, and will be immediately ready. The device may enter Sleep while waiting for the switch as described in **Section 9.3.3 "Clock Switch and Sleep"**.

When the new oscillator is ready, the New Oscillator is Ready (NOSCR) bit of OSCCON3 and the Clock Switch Interrupt Flag (CSWIF) bit of PIR1 become set (CSWIF = 1). If Clock Switch Interrupts are enabled (CSWIE = 1), an interrupt will be generated at that time. The Oscillator Ready (ORDY) bit of OSCCON3 can also be polled to determine when the oscillator is ready in lieu of an interrupt.

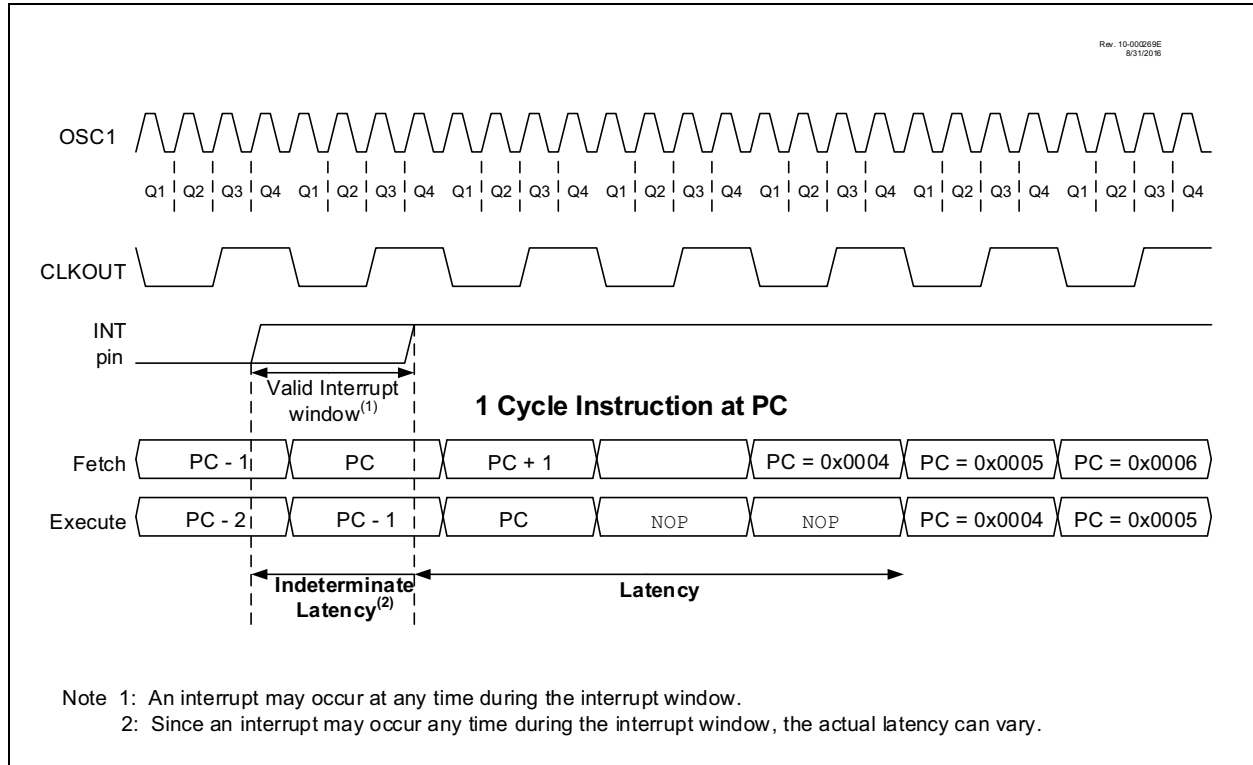
If the Clock Switch Hold (CSWHOLD) bit of OSCCON3 is clear, the oscillator switch will occur when the new Oscillator's READY bit (NOSCR) is set, and the interrupt (if enabled) will be serviced at the new oscillator setting.

If CSWHOLD is set, the oscillator switch is suspended, while execution continues using the current (old) clock source. When the NOSCR bit is set, software should:

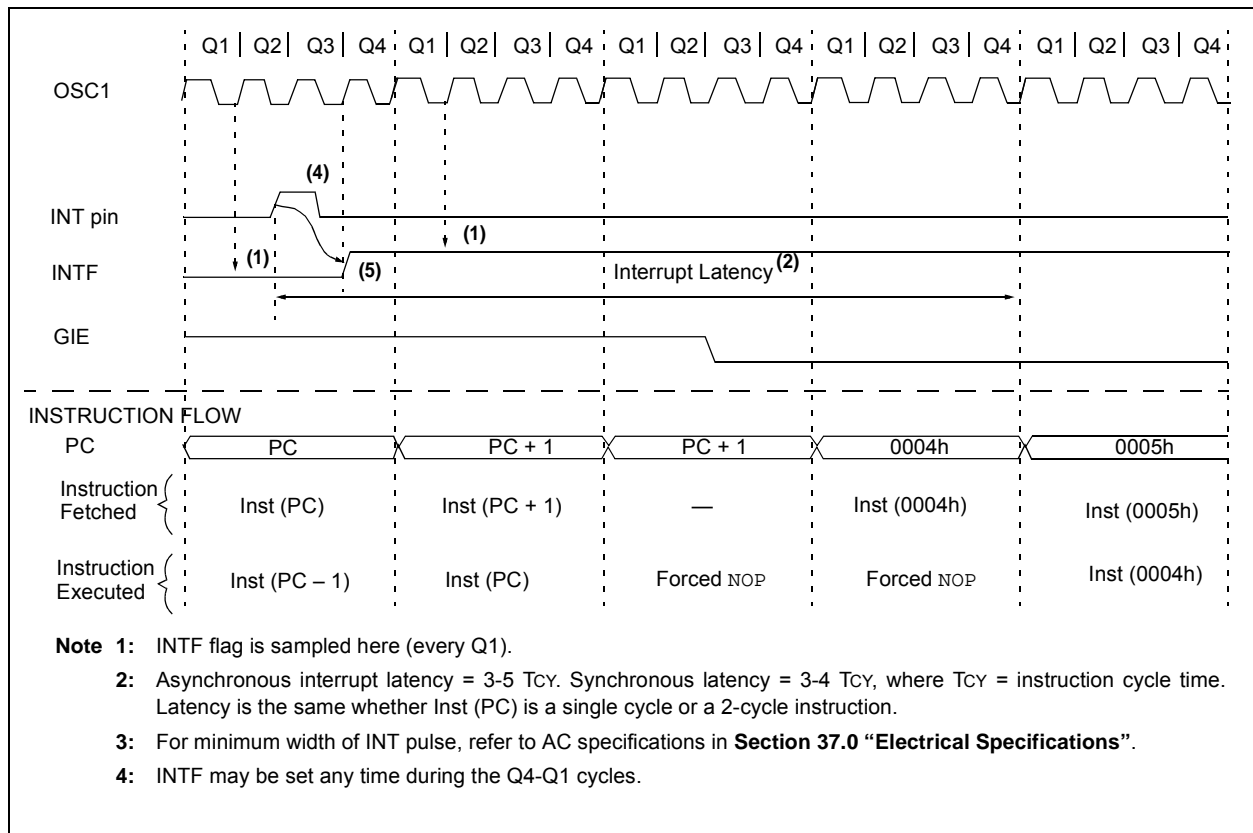
- set CSWHOLD = 0 so the switch can complete, or
- copy COSC into NOSC to abandon the switch.

If DOZE is in effect, the switch occurs on the next clock cycle, whether or not the CPU is operating during that cycle.

**FIGURE 10-2: INTERRUPT LATENCY**



**FIGURE 10-3: INT PIN INTERRUPT TIMING**



## REGISTER 10-17: PIR7: PERIPHERAL INTERRUPT REQUEST REGISTER 7

U-0	U-0	R/W/HS-0/0	R/W/HS-0/0	U-0	U-0	U-0	R/W/HS-0/0
—	—	NVMIF	NCO1IF	—	—	—	CWG1IF
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 7-6      **Unimplemented:** Read as '0'

bit 5      **NVMIF:** Nonvolatile Memory (NVM) Interrupt Flag bit  
 1 = The requested NVM operation has completed  
 0 = NVM interrupt not asserted

bit 4      **NCO1IF:** Numerically Controlled Oscillator (NCO) Interrupt Flag bit  
 1 = The NCO has rolled over  
 0 = No NCO interrupt event has occurred

bit 3-1      **Unimplemented:** Read as '0'

bit 0      **CWG1IF:** CWG1 Interrupt Flag bit  
 1 = CWG1 has gone into shutdown  
 0 = CWG1 is operating normally, or interrupt cleared

**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.

## 23.2 Comparator Control

Each comparator has two control registers: CMxCON0 and CMxCON1.

The CMxCON0 register (see Register 23-1) contains Control and Status bits for the following:

- Enable
- Output
- Output polarity
- Hysteresis enable
- Timer1 output synchronization

The CMxCON1 register (see Register 23-2) contains Control bits for the following:

- Interrupt on positive/negative edge enables
- The CMxNSEL and CMxPSEL (Register 23-3 and Register 23-4) contain control bits for the following:
  - Positive input channel selection
  - Negative input channel selection

### 23.2.1 COMPARATOR ENABLE

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

### 23.2.2 COMPARATOR OUTPUT

The output of the comparator can be monitored by reading either the CxOUT bit of the CMxCON0 register or the MCxOUT bit of the CMOUT register.

The comparator output can also be routed to an external pin through the RxyPPS register (Register 15-2). The corresponding TRIS bit must be clear to enable the pin as an output.

**Note 1:** The internal output of the comparator is latched with each instruction cycle. Unless otherwise specified, external outputs are not latched.

### 23.2.3 COMPARATOR OUTPUT POLARITY

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

Table 23-2 shows the output state versus input conditions, including polarity control.

**TABLE 23-2: COMPARATOR OUTPUT STATE VS. INPUT CONDITIONS**

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

## 23.10 CWG1 Auto-shutdown Source

The output of the comparator module can be used as an auto-shutdown source for the CWG1 module. When the output of the comparator is active and the corresponding ASxE is enabled, the CWG operation will be suspended immediately (see **Section 30.10 “Auto-Shutdown”**).

## 23.11 Operation in Sleep Mode

The comparator module can operate during Sleep. The comparator clock source is based on the Timer1 clock source. If the Timer1 clock source is either the system clock (Fosc) or the instruction clock (Fosc/4), Timer1 will not operate during Sleep, and synchronized comparator outputs will not operate.

A comparator interrupt will wake the device from Sleep. The CxIE bits of the PIE2 register must be set to enable comparator interrupts.



## 26.1 Timer1 Operation

The Timer1 modules are 16-bit incrementing counters which are accessed through the TMR1H:TMR1L register pairs. Writes to TMR1H or TMR1L directly update the counter.

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

The timer is enabled by configuring the TMR1ON and GE bits in the T1CON and T1GCON registers, respectively. Table 26-1 displays the Timer1 enable selections.

**TABLE 26-1: TIMER1 ENABLE SELECTIONS**

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

## 26.2 Clock Source Selection

The T1CLK register is used to select the clock source for the timer. Register 26-3 shows the possible clock sources that may be selected to make the timer increment.

### 26.2.1 INTERNAL CLOCK SOURCE

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

When the FOSC internal clock source is selected, the timer register value will increment by four counts every instruction clock cycle. Due to this condition, a 2 LSB error in resolution will occur when reading the TMR1H:TMR1L value. To utilize the full resolution of the timer in this mode, an asynchronous input signal must be used to gate the timer clock input.

Out of the total timer gate signal sources, the following subset of sources can be asynchronous and may be useful for this purpose:

- CLC4 output
- CLC3 output
- CLC2 output
- CLC1 output
- Zero-Cross Detect output
- Comparator2 output
- Comparator1 output
- TxG PPS remappable input pin

### 26.2.2 EXTERNAL CLOCK SOURCE

When the timer is enabled and the external clock input source (ex: T1CKI PPS remappable input) is selected as the clock source, the timer will increment on the rising edge of the external clock input.

When using an external clock source, the timer can be configured to run synchronously or asynchronously, as described in **Section 26.5 “Timer Operation in Asynchronous Counter Mode”**.

When used as a timer with a clock oscillator, an external 32.768 kHz crystal can be used connected to the SOSCI/SOSCO pins.

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

- The timer is first enabled after POR
- Firmware writes to TMR1H or TMR1L
- The timer is disabled
- The timer is re-enabled (e.g., TMR1ON-->1) when the T1CKI signal is currently logic low.

## 26.6.2 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.

### 26.6.2.1 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.

### 26.6.2.2 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.

### 26.6.2.3 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”**.

### 26.6.2.4 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”**.

## 26.6.3 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.

## 26.6.4 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/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/DONE bit will automatically be cleared. No other gate events will be allowed to increment the timer until the GGO/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/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.

## 26.6.5 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).

## 26.6.6 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).

**TABLE 28-2: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS (Fosc = 20 MHz)**

PWM Frequency	1.22 kHz	4.88 kHz	19.53 kHz	78.12 kHz	156.3 kHz	208.3 kHz
Timer Prescale	16	4	1	1	1	1
PR2 Value	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
Maximum Resolution (bits)	10	10	10	8	7	6.6

**TABLE 28-3: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS (Fosc = 8 MHz)**

PWM Frequency	1.22 kHz	4.90 kHz	19.61 kHz	76.92 kHz	153.85 kHz	200.0 kHz
Timer Prescale	16	4	1	1	1	1
PR2 Value	0x65	0x65	0x65	0x19	0x0C	0x09
Maximum Resolution (bits)	8	8	8	6	5	5

### 28.3.8 OPERATION IN SLEEP MODE

In Sleep mode, the TMR2 register will not increment and the state of the module will not change. If the CCPx pin is driving a value, it will continue to drive that value. When the device wakes up, TMR2 will continue from its previous state.

### 28.3.9 CHANGES IN SYSTEM CLOCK FREQUENCY

The PWM frequency is derived from the system clock frequency. Any changes in the system clock frequency will result in changes to the PWM frequency. See **Section 9.0 “Oscillator Module (with Fail-Safe Clock Monitor)”** for additional details.

### 28.3.10 EFFECTS OF RESET

Any Reset will force all ports to Input mode and the CCP registers to their Reset states.

## 29.1 Standard PWM Mode

The standard PWM mode generates a Pulse-Width Modulation (PWM) signal on the PWMx pin with up to ten bits of resolution. The period, duty cycle, and resolution are controlled by the following registers:

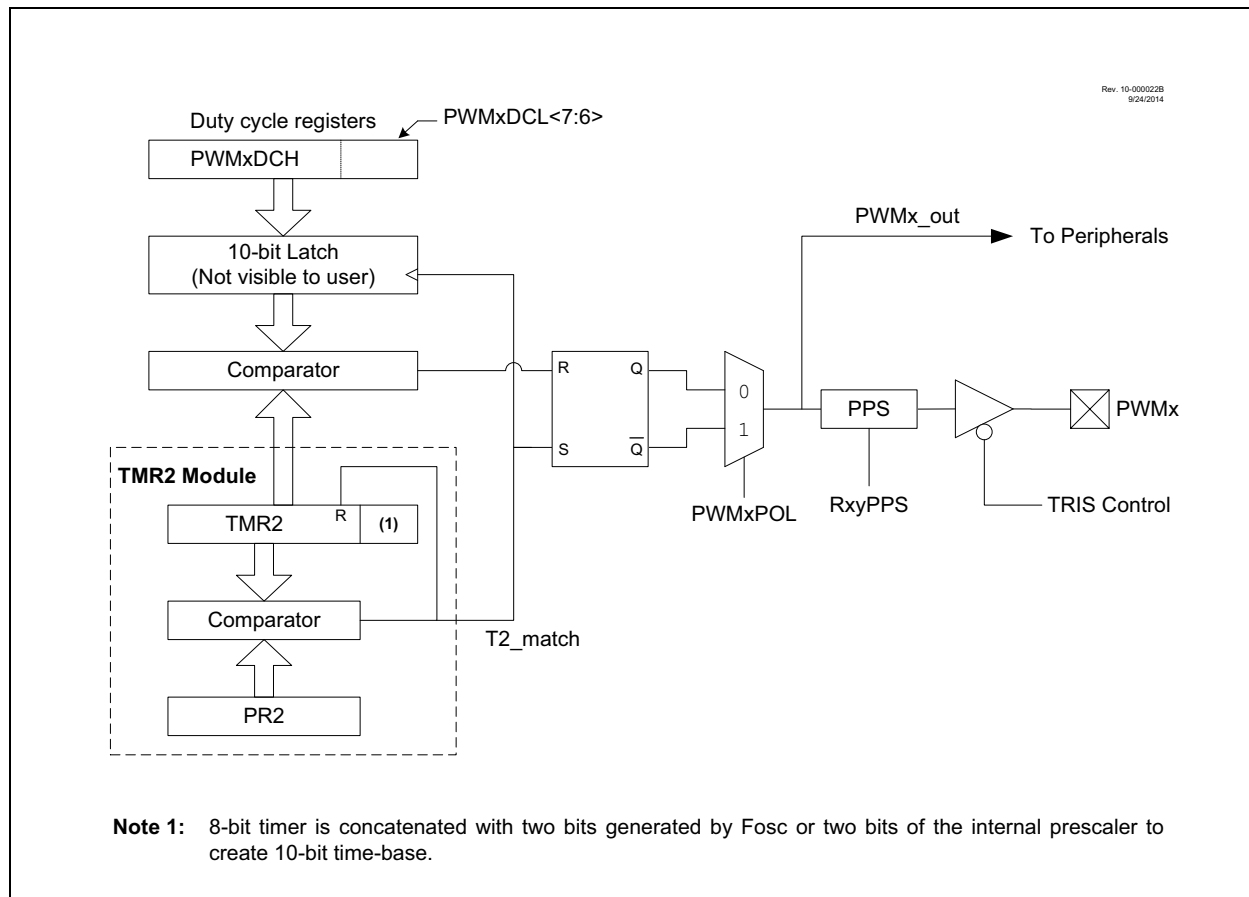
- TMR2 register
- PR2 register
- PWMxCON registers
- PWMxDCH registers
- PWMxDCL registers

Figure 29-2 shows a simplified block diagram of PWM operation.

If PWMPOL = 0, the default state of the output is '0'. If PWMPOL = 1, the default state is '1'. If PWMEN = 0, the output will be the default state.

**Note:** The corresponding TRIS bit must be cleared to enable the PWM output on the PWMx pin

**FIGURE 29-2: SIMPLIFIED PWM BLOCK DIAGRAM**



## REGISTER 30-7: CWG1STR: CWG1 STEERING CONTROL REGISTER<sup>(1)</sup>

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
OVRD	OVRC	OVRB	OVRA	STRD <sup>(2)</sup>	STRC <sup>(2)</sup>	STRB <sup>(2)</sup>	STRA <sup>(2)</sup>
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	<b>OVRD:</b> Steering Data D bit
bit 6	<b>OVRC:</b> Steering Data C bit
bit 5	<b>OVRB:</b> Steering Data B bit
bit 4	<b>OVRA:</b> Steering Data A bit
bit 3	<b>STRD:</b> Steering Enable D bit <sup>(2)</sup> 1 = CWG1D output has the CWG1_data waveform with polarity control from POLD bit 0 = CWG1D output is assigned the value of OVRD bit
bit 2	<b>STRC:</b> Steering Enable C bit <sup>(2)</sup> 1 = CWG1C output has the CWG1_data waveform with polarity control from POLC bit 0 = CWG1C output is assigned the value of OVRC bit
bit 1	<b>STRB:</b> Steering Enable B bit <sup>(2)</sup> 1 = CWG1B output has the CWG1_data waveform with polarity control from POLB bit 0 = CWG1B output is assigned the value of OVRB bit
bit 0	<b>STRA:</b> Steering Enable A bit <sup>(2)</sup> 1 = CWG1A output has the CWG1_data waveform with polarity control from POLA bit 0 = CWG1A output is assigned the value of OVRA bit

**Note 1:** The bits in this register apply only when MODE<2:0> = 00x.

**2:** This bit is effectively double-buffered when MODE<2:0> = 001.

## REGISTER 31-8: CLCxGLS1: GATE 1 LOGIC SELECT 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
LCxG2D4T	LCxG2D4N	LCxG2D3T	LCxG2D3N	LCxG2D2T	LCxG2D2N	LCxG2D1T	LCxG2D1N
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

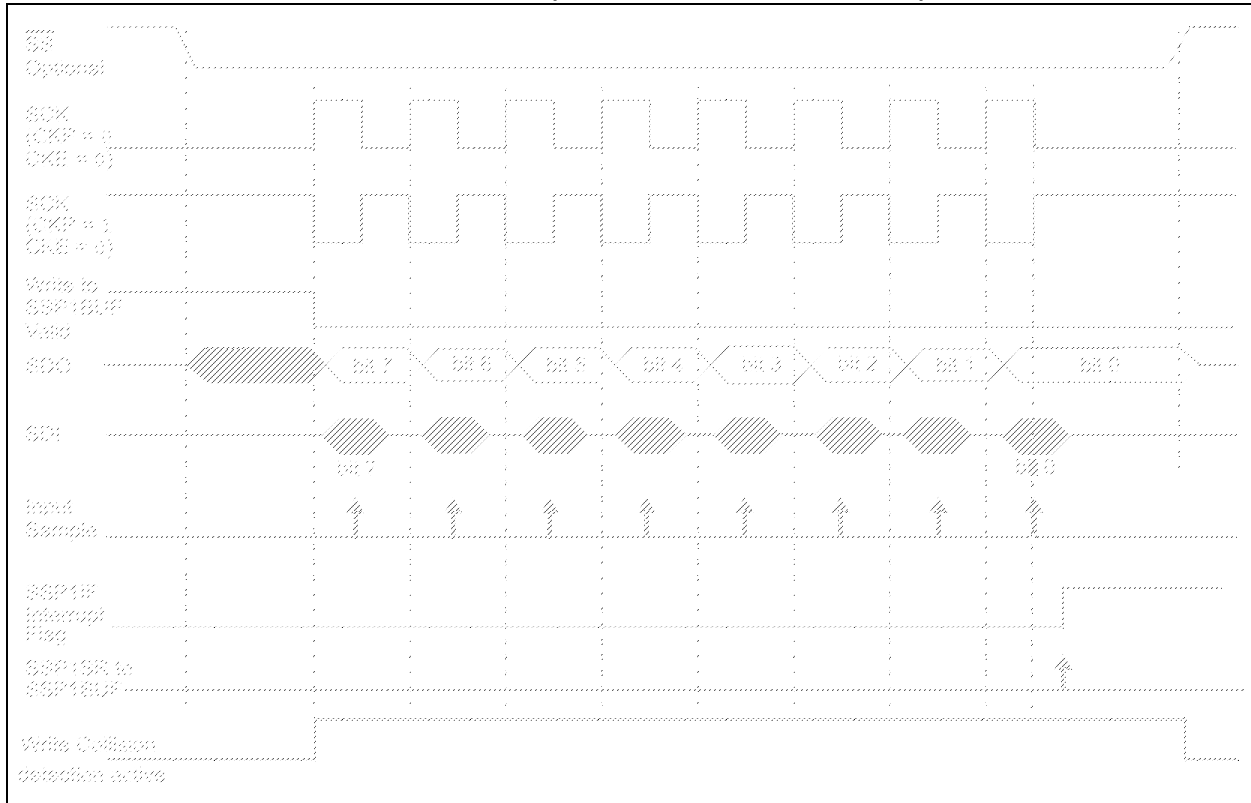
-n/n = Value at POR and BOR/Value at all other Resets

'1' = Bit is set

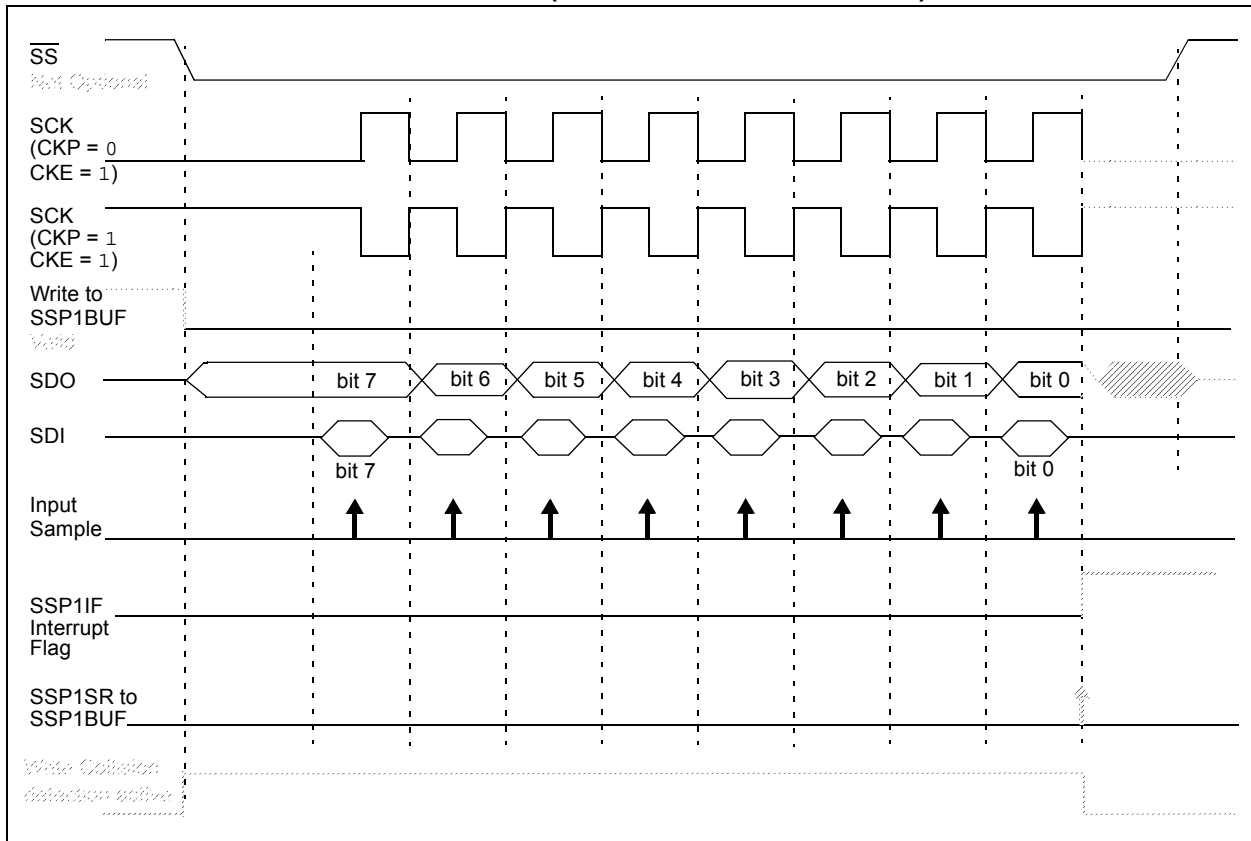
'0' = Bit is cleared

- bit 7      **LCxG2D4T:** Gate 1 Data 4 True (non-inverted) bit  
 1 = CLCIN3 (true) is gated into CLCx Gate 1  
 0 = CLCIN3 (true) is not gated into CLCx Gate 1
- bit 6      **LCxG2D4N:** Gate 1 Data 4 Negated (inverted) bit  
 1 = CLCIN3 (inverted) is gated into CLCx Gate 1  
 0 = CLCIN3 (inverted) is not gated into CLCx Gate 1
- bit 5      **LCxG2D3T:** Gate 1 Data 3 True (non-inverted) bit  
 1 = CLCIN2 (true) is gated into CLCx Gate 1  
 0 = CLCIN2 (true) is not gated into CLCx Gate 1
- bit 4      **LCxG2D3N:** Gate 1 Data 3 Negated (inverted) bit  
 1 = CLCIN2 (inverted) is gated into CLCx Gate 1  
 0 = CLCIN2 (inverted) is not gated into CLCx Gate 1
- bit 3      **LCxG2D2T:** Gate 1 Data 2 True (non-inverted) bit  
 1 = CLCIN1 (true) is gated into CLCx Gate 1  
 0 = CLCIN1 (true) is not gated into CLCx Gate 1
- bit 2      **LCxG2D2N:** Gate 1 Data 2 Negated (inverted) bit  
 1 = CLCIN1 (inverted) is gated into CLCx Gate 1  
 0 = CLCIN1 (inverted) is not gated into CLCx Gate 1
- bit 1      **LCxG2D1T:** Gate 1 Data 1 True (non-inverted) bit  
 1 = CLCIN0 (true) is gated into CLCx Gate 1  
 0 = CLCIN0 (true) is not gated into CLCx Gate 1
- bit 0      **LCxG2D1N:** Gate 1 Data 1 Negated (inverted) bit  
 1 = CLCIN0 (inverted) is gated into CLCx Gate 1  
 0 = CLCIN0 (inverted) is not gated into CLCx Gate 1

**FIGURE 32-9: SPI MODE WAVEFORM (SLAVE MODE WITH CKE = 0)**



**FIGURE 32-10: SPI MODE WAVEFORM (SLAVE MODE WITH CKE = 1)**



## 32.6.13.1 Bus Collision During a Start Condition

During a Start condition, a bus collision occurs if:

- SDA or SCL are sampled low at the beginning of the Start condition (Figure 32-33).
- SCL is sampled low before SDA is asserted low (Figure 32-34).

During a Start condition, both the SDA and the SCL pins are monitored.

If the SDA pin is already low, or the SCL pin is already low, then all of the following occur:

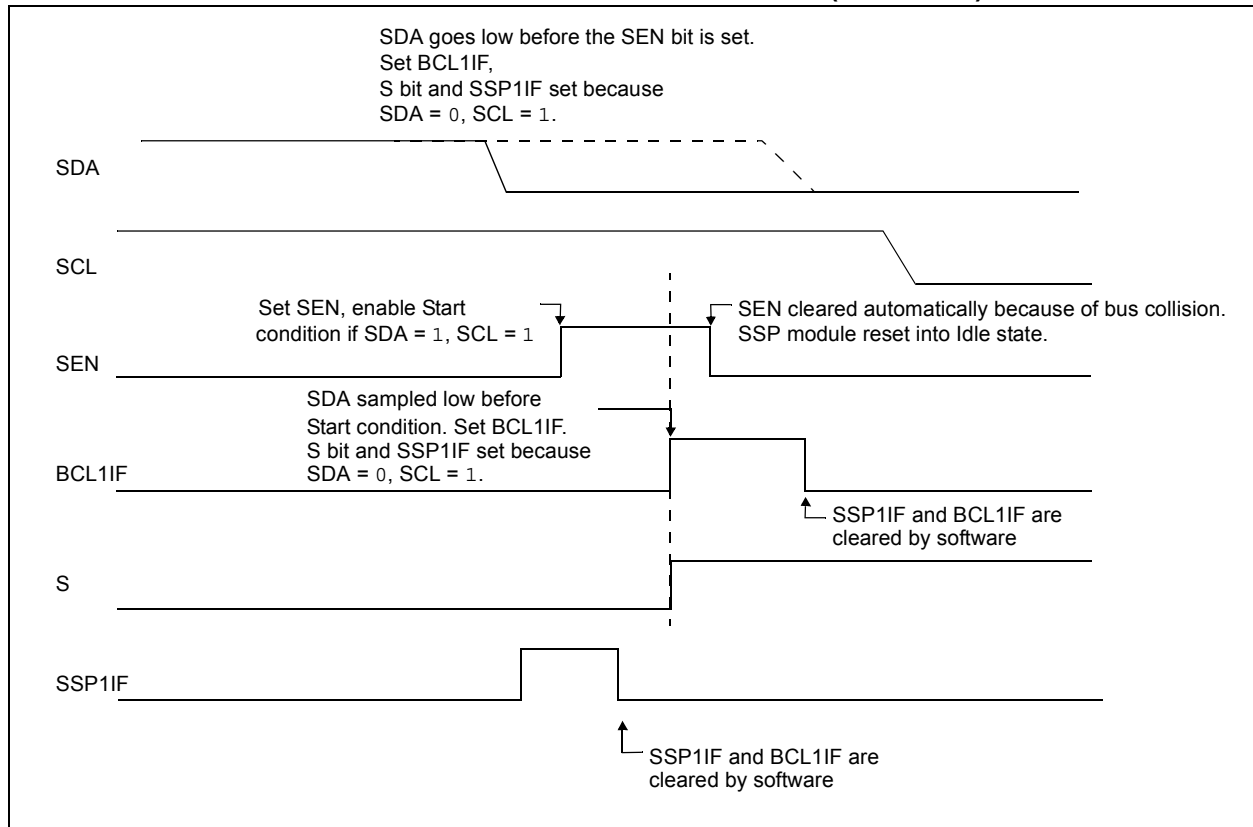
- the Start condition is aborted,
- the BCL1IF flag is set and
- the MSSP module is reset to its Idle state (Figure 32-33).

The Start condition begins with the SDA and SCL pins deasserted. When the SDA pin is sampled high, the Baud Rate Generator is loaded and counts down. If the SCL pin is sampled low while SDA is high, a bus collision occurs because it is assumed that another master is attempting to drive a data '1' during the Start condition.

If the SDA pin is sampled low during this count, the BRG is reset and the SDA line is asserted early (Figure 32-35). If, however, a '1' is sampled on the SDA pin, the SDA pin is asserted low at the end of the BRG count. The Baud Rate Generator is then reloaded and counts down to zero; if the SCL pin is sampled as '0' during this time, a bus collision does not occur. At the end of the BRG count, the SCL pin is asserted low.

**Note:** The reason that bus collision is not a factor during a Start condition is that no two bus masters can assert a Start condition at the exact same time. Therefore, one master will always assert SDA before the other. This condition does not cause a bus collision because the two masters must be allowed to arbitrate the first address following the Start condition. If the address is the same, arbitration must be allowed to continue into the data portion, Repeated Start or Stop conditions.

**FIGURE 32-33: BUS COLLISION DURING START CONDITION (SDA ONLY)**





## 33.1.1.5 TSR Status

The TRMT bit of the TXxSTA register indicates the status of the TSR register. This is a read-only bit. The TRMT bit is set when the TSR register is empty and is cleared when a character is transferred to the TSR register from the TXxREG. The TRMT bit remains clear until all bits have been shifted out of the TSR register. No interrupt logic is tied to this bit, so the user has to poll this bit to determine the TSR status.

**Note:** The TSR register is not mapped in data memory, so it is not available to the user.

## 33.1.1.6 Transmitting 9-Bit Characters

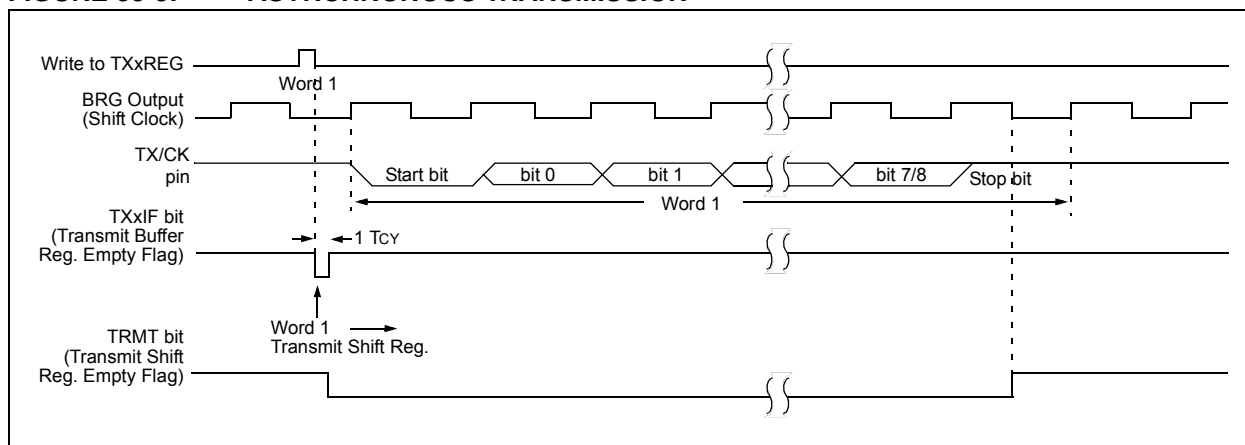
The EUSART supports 9-bit character transmissions. When the TX9 bit of the TXxSTA register is set, the EUSART will shift nine bits out for each character transmitted. The TX9D bit of the TXxSTA register is the ninth, and Most Significant data bit. When transmitting 9-bit data, the TX9D data bit must be written before writing the eight Least Significant bits into the TXxREG. All nine bits of data will be transferred to the TSR shift register immediately after the TXxREG is written.

A special 9-bit Address mode is available for use with multiple receivers. See **Section 33.1.2.7 “Address Detection”** for more information on the Address mode.

## 33.1.1.7 Asynchronous Transmission Set-up:

1. Initialize the SPxBRGH, SPxBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 33.3 “EUSART Baud Rate Generator (BRG)”**).
2. Enable the asynchronous serial port by clearing the SYNC bit and setting the SPEN bit.
3. If 9-bit transmission is desired, set the TX9 control bit. A set ninth data bit will indicate that the eight Least Significant data bits are an address when the receiver is set for address detection.
4. Set SCKP bit if inverted transmit is desired.
5. Enable the transmission by setting the TXEN control bit. This will cause the TXxIF interrupt bit to be set.
6. If interrupts are desired, set the TXxIE interrupt enable bit of the PIE3 register. An interrupt will occur immediately provided that the GIE and PEIE bits of the INTCON register are also set.
7. If 9-bit transmission is selected, the ninth bit should be loaded into the TX9D data bit.
8. Load 8-bit data into the TXxREG register. This will start the transmission.

**FIGURE 33-3: ASYNCHRONOUS TRANSMISSION**



## REGISTER 33-3: BAUDxCON: BAUD RATE CONTROL REGISTER

R/W-0/0	R-1/1	U-0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0
ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN
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 **ABDOVF:** Auto-Baud Detect Overflow bit

Asynchronous mode:

- 1 = Auto-baud timer overflowed
- 0 = Auto-baud timer did not overflow

Synchronous mode:

Don't care

bit 6 **RCIDL:** Receive Idle Flag bit

Asynchronous mode:

- 1 = Receiver is Idle
- 0 = Start bit has been received and the receiver is receiving

Synchronous mode:

Don't care

bit 5 **Unimplemented:** Read as '0'

bit 4 **SCKP:** Clock/Transmit Polarity Select bit

Asynchronous mode:

- 1 = Idle state for transmit (TX) is a low level
- 0 = Idle state for transmit (TX) is a high level

Synchronous mode:

- 1 = Idle state for clock (CK) is a high level
- 0 = Idle state for clock (CK) is a low level

bit 3 **BRG16:** 16-bit Baud Rate Generator bit

- 1 = 16-bit Baud Rate Generator is used
- 0 = 8-bit Baud Rate Generator is used

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

bit 1 **WUE:** Wake-up Enable bit

Asynchronous mode:

- 1 = USART will continue to sample the Rx pin – interrupt generated on falling edge; bit cleared in hardware on following rising edge.
- 0 = RX pin not monitored nor rising edge detected

Synchronous mode:

Unused in this mode – value ignored

bit 0 **ABDEN:** Auto-Baud Detect Enable bit

Asynchronous mode:

- 1 = Enable baud rate measurement on the next character – requires reception of a SYNCH field (55h); cleared in hardware upon completion
- 0 = Baud rate measurement disabled or completed

Synchronous mode:

Unused in this mode – value ignored

**REGISTER 33-7: SPxBRGH<sup>(1, 2)</sup>: BAUD RATE GENERATOR HIGH REGISTER**

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

**Legend:**

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

bit 7            **SPxBRG<15:8>**: Upper eight bits of the Baud Rate Generator

- Note 1:** SPxBRGH value is ignored for all modes unless BAUDxCON<BRG16> is active.  
**2:** Writing to SPxBRGH resets the BRG counter.

<b>RETLW</b>	<b>Return with literal in W</b>
Syntax:	<code>[ label ] RETLW k</code>
Operands:	$0 \leq k \leq 255$
Operation:	$k \rightarrow (W)$ ; $TOS \rightarrow PC$
Status Affected:	None
Description:	The W register is loaded with the 8-bit literal 'k'. The program counter is loaded from the top of the stack (the return address). This is a 2-cycle instruction.
Words:	1
Cycles:	2
<u>Example:</u>	<pre>CALL TABLE;W contains table ;offset value • ;W now has table value • • ADDWF PC ;W = offset RETLW k1 ;Begin table RETLW k2 ; • • • RETLW kn ; End of table</pre>
TABLE	<pre>Before Instruction W = 0x07 After Instruction W = value of k8</pre>

<b>RETURN</b>	<b>Return from Subroutine</b>
Syntax:	<code>[ label ] RETURN</code>
Operands:	None
Operation:	$TOS \rightarrow PC$
Status Affected:	None
Description:	Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a 2-cycle instruction.

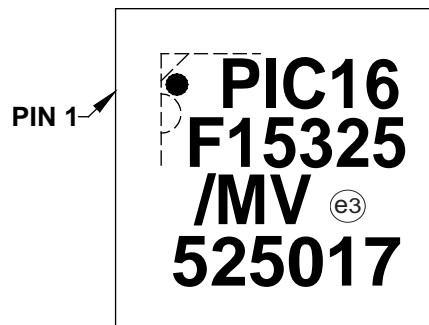
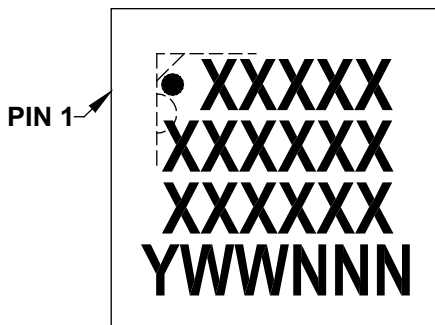
<b>RLF</b>	<b>Rotate Left f through Carry</b>
Syntax:	<code>[ label ] RLF f,d</code>
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	See description below
Status Affected:	C
Description:	The contents of register 'f' are rotated one bit to the left through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is stored back in register 'f'.
Words:	1
Cycles:	1
<u>Example:</u>	<pre>RLF REG1,0</pre> <p>Before Instruction</p> <pre>REG1 = 1110 0110 C = 0</pre> <p>After Instruction</p> <pre>REG1 = 1110 0110 W = 1100 1100 C = 1</pre>

<b>RRF</b>	<b>Rotate Right f through Carry</b>
Syntax:	<code>[ label ] RRF f,d</code>
Operands:	$0 \leq f \leq 127$ $d \in [0,1]$
Operation:	See description below
Status Affected:	C
Description:	The contents of register 'f' are rotated one bit to the right through the Carry flag. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed back in register 'f'.

## 40.1 Package Marking Information (Continued)

16-Lead UQFN (4x4x0.5 mm)

Example



<b>Legend:</b>	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code
		Pb-free JEDEC® designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator (e3) can be found on the outer packaging for this package.

**Note:** In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.