



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 ² C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	12
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 11x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	14-TSSOP (0.173", 4.40mm Width)
Supplier Device Package	14-TSSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f15324t-i-st

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at docerrors@microchip.com. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Website at:

<http://www.microchip.com>

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000000A is version A of document DS30000000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Website; <http://www.microchip.com>
- Your local Microchip sales office (see last page)

When contacting a sales office, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Customer Notification System

Register on our website at www.microchip.com to receive the most current information on all of our products.

2.3 Master Clear (MCLR) Pin

The $\overline{\text{MCLR}}$ pin provides two specific device functions: Device Reset, and Device Programming and Debugging. If programming and debugging are not required in the end application, a direct connection to V_{DD} may be all that is required. The addition of other components, to help increase the application's resistance to spurious Resets from voltage sags, may be beneficial. A typical configuration is shown in Figure 2-1. Other circuit designs may be implemented, depending on the application's requirements.

During programming and debugging, the resistance and capacitance that can be added to the pin must be considered. Device programmers and debuggers drive the MCLR pin. Consequently, specific voltage levels (V_{IH} and V_{IL}) and fast signal transitions must not be adversely affected. Therefore, specific values of $R1$ and $C1$ will need to be adjusted based on the application and PCB requirements. For example, it is recommended that the capacitor, $C1$, be isolated from the MCLR pin during programming and debugging operations by using a jumper (Figure 2-2). The jumper is replaced for normal run-time operations.

Any components associated with the $\overline{\text{MCLR}}$ pin should be placed within 0.25 inch (6 mm) of the pin.

2.4 ICSP™ Pins

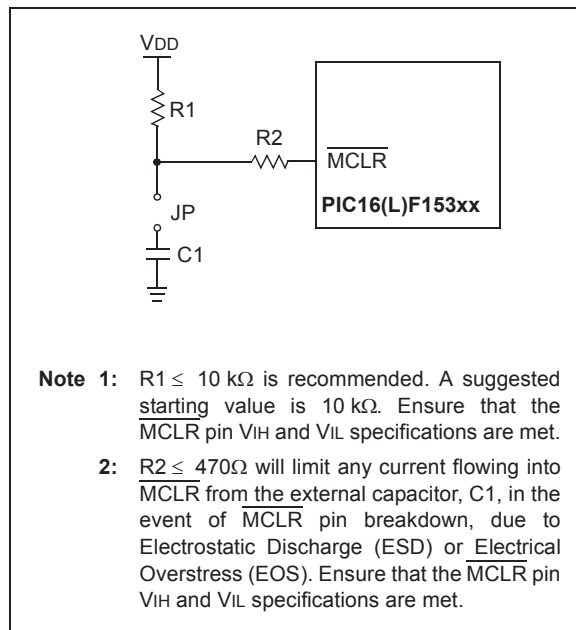
The ICSPCLK and ICSPDAT pins are used for In-Circuit Serial Programming™ (ICSP™) and debugging purposes. It is recommended to keep the trace length between the ICSP connector and the ICSP pins on the device as short as possible. If the ICSP connector is expected to experience an ESD event, a series resistor is recommended, with the value in the range of a few tens of ohms, not to exceed 100Ω.

Pull-up resistors, series diodes and capacitors on the ICSPCLK and ICSPDAT pins are not recommended as they will interfere with the programmer/debugger communications to the device. If such discrete components are an application requirement, they should be removed from the circuit during programming and debugging. Alternatively, refer to the AC/DC characteristics and timing requirements information in the respective device Flash programming specification for information on capacitive loading limits, and pin input voltage high (V_{IH}) and input low (V_{IL}) requirements.

For device emulation, ensure that the “Communication Channel Select” (i.e., ICSPCLK/ICSPDAT pins), programmed into the device, matches the physical connections for the ICSP to the Microchip debugger/emulator tool.

For more information on available Microchip development tools connection requirements, refer to [Section 39.0 “Development Support”](#).

FIGURE 2-2: EXAMPLE OF MCLR PIN CONNECTIONS



2.5 External Oscillator Pins

Many microcontrollers have options for at least two oscillators: a high-frequency primary oscillator and a low-frequency secondary oscillator (refer to [Section 9.0 “Oscillator Module \(with Fail-Safe Clock Monitor\)”](#) for details). The PIC16(L)F15324/44 devices do not have a secondary oscillator.

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

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

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

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

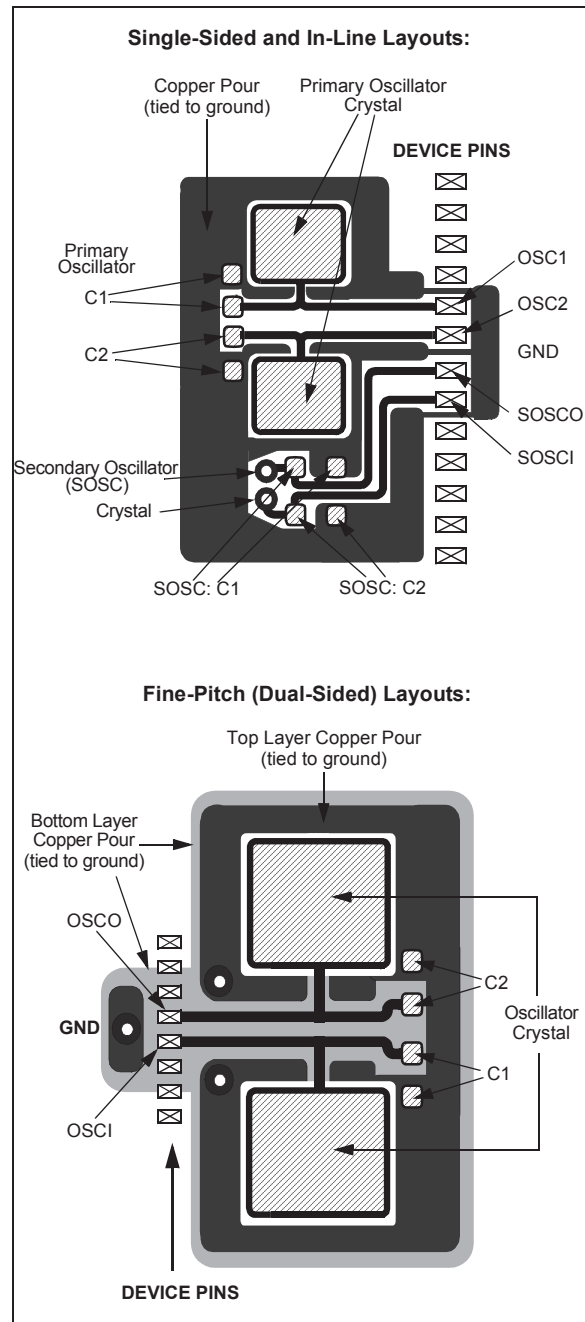
For additional information and design guidance on oscillator circuits, refer to these Microchip Application Notes, available at the corporate website (www.microchip.com):

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

2.6 Unused I/Os

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

FIGURE 2-3: SUGGESTED PLACEMENT OF THE OSCILLATOR CIRCUIT



REGISTER 5-4: CONFIGURATION WORD 4: MEMORY

R/W-1	U-1	R/W-1	U-1	R/W-1	R/W-1
LVP	—	WRTSAF ⁽¹⁾	—	WRTC ⁽¹⁾	WRTB ⁽¹⁾
bit 13	12	11	10	9	bit 8

R/W-1	U-1	U-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
WRTAPP ⁽¹⁾	—	—	SAFEN ⁽¹⁾	BBEN ⁽¹⁾	BBSIZE2	BBSIZE1	BBSIZE0
bit 7	6	5	4	3	2	1	bit 0

Legend:

R = Readable bit

P = Programmable bit

x = Bit is unknown

U = Unimplemented bit, read as '1'

'0' = Bit is cleared

'1' = Bit is set

W = Writable bit

n = Value when blank or after Bulk Erase

bit 13

LVP: Low Voltage Programming Enable bit

1 = Low voltage programming enabled. MCLR/VPP pin function is MCLR. MCLRE Configuration bit is ignored.

0 = HV on MCLR/VPP must be used for programming.

The LVP bit cannot be written (to zero) while operating from the LVP programming interface. The purpose of this rule is to prevent the user from dropping out of LVP mode while programming from LVP mode, or accidentally eliminating LVP mode from the configuration state.

The preconditioned (erased) state for this bit is critical.

bit 12

Unimplemented: Read as '1'

bit 11

WRTSAF: Storage Area Flash Write Protection bit

1 = SAF NOT write-protected

0 = SAF write-protected

Unimplemented, if SAF is not supported in the device family and only applicable if SAFEN = 0.

bit 10

Unimplemented: Read as '1'

bit 9

WRTC: Configuration Register Write Protection bit

1 = Configuration Register NOT write-protected

0 = Configuration Register write-protected

bit 8

WRTB: Boot Block Write Protection bit

1 = Boot Block NOT write-protected

0 = Boot Block write-protected

Only applicable if BBEN = 0.

bit 7

WRTAPP: Application Block Write Protection bit

1 = Application Block NOT write-protected

0 = Application Block write-protected

bit 6-5

Unimplemented: Read as '1'

bit 4

SAFEN: SAF Enable bit

1 = SAF disabled

0 = SAF enabled

bit 3

BBEN: Boot Block Enable bit

1 = Boot Block disabled

0 = Boot Block enabled

bit 2-0

BBSIZE<2:0>: Boot Block Size Selection bits (See Table 5-1)

BBSIZE is used only when BBEN = 0

BBSIZ bits can only be written while BBEN = 1; after BBEN = 0, BBSIZ is write-protected.

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

FIGURE 9-5: CLOCK SWITCH (CSWHOLD = 0)

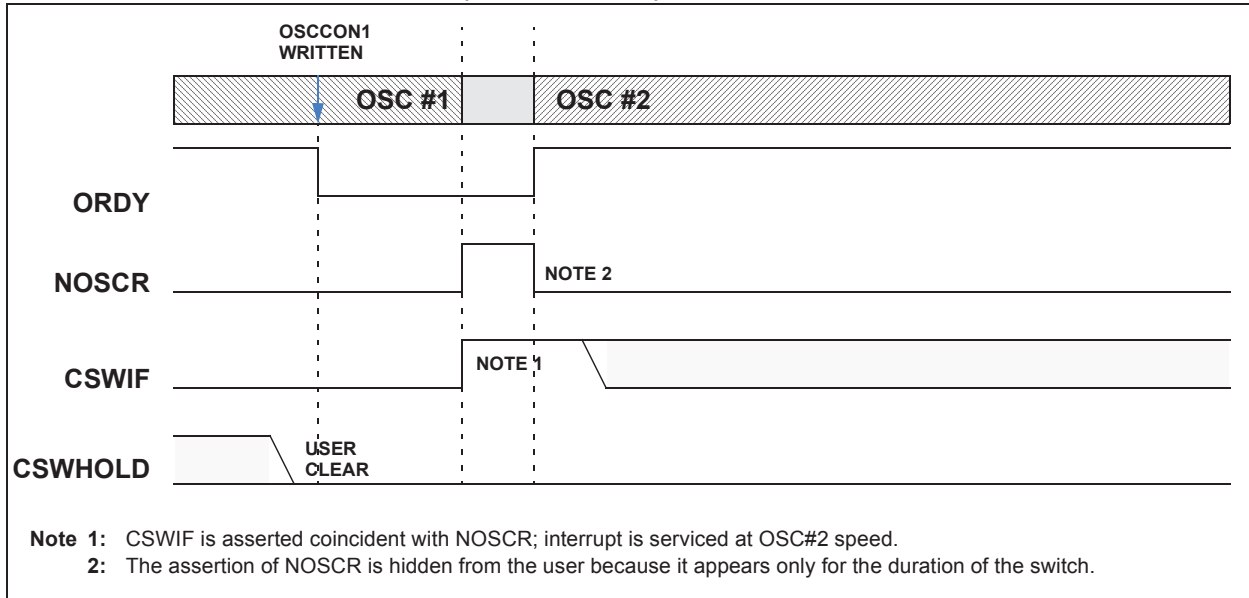
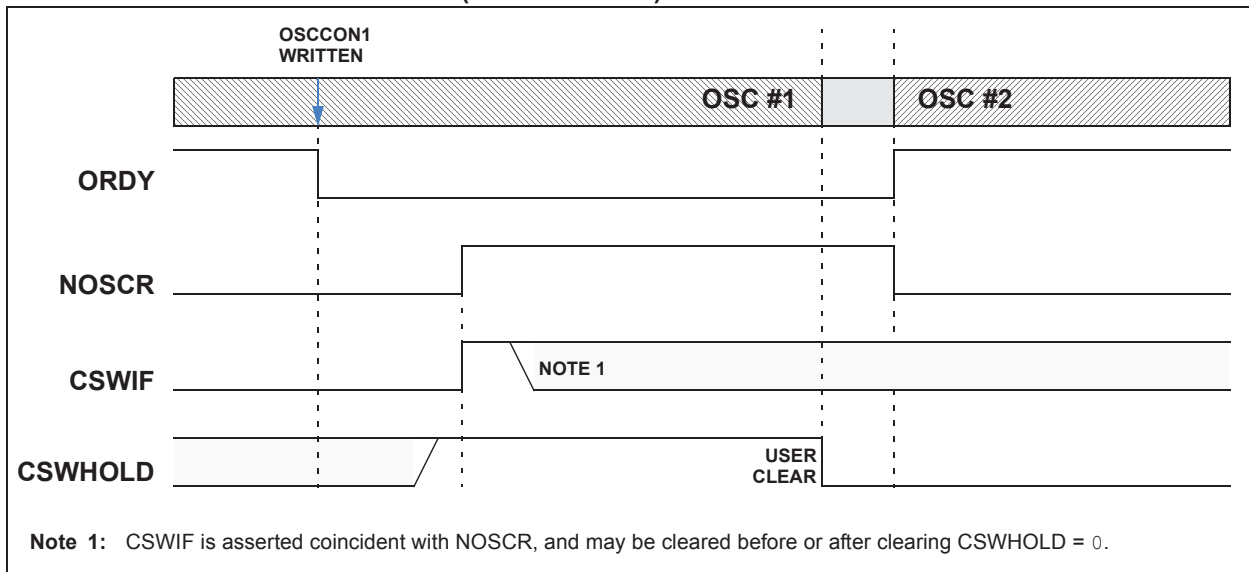


FIGURE 9-6: CLOCK SWITCH (CSWHOLD = 1)



REGISTER 10-2: PIE0: PERIPHERAL INTERRUPT ENABLE REGISTER 0

U-0	U-0	R/W-0/0	R/W-0/0	U-0	U-0	U-0	R/W-0/0
—	—	TMR0IE	IOCIE	—	—	—	INTE
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 **TMR0IE:** Timer0 Overflow Interrupt Enable bit
 1 = Enables the Timer0 interrupt
 0 = Disables the Timer0 interrupt
- bit 4 **IOCIE:** Interrupt-on-Change Interrupt Enable bit
 1 = Enables the IOC change interrupt
 0 = Disables the IOC change interrupt
- bit 3-1 **Unimplemented:** Read as '0'
- bit 0 **INTE:** INT External Interrupt Flag bit⁽¹⁾
 1 = Enables the INT external interrupt
 0 = Disables the INT external interrupt

Note 1: The External Interrupt GPIO pin is selected by INTPPS ([Register 15-1](#)).

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt controlled by PIE1-PIE7. Interrupt sources controlled by the PIE0 register do not require PEIE to be set in order to allow interrupt vectoring (when GIE is set).

13.3.2 NVM UNLOCK SEQUENCE

The unlock sequence is a mechanism that protects the NVM from unintended self-write programming or erasing. The sequence must be executed and completed without interruption to successfully complete any of the following operations:

- PFM Row Erase
- Load of PFM write latches
- Write of PFM write latches to PFM memory
- Write of PFM write latches to User IDs

The unlock sequence consists of the following steps and must be completed in order:

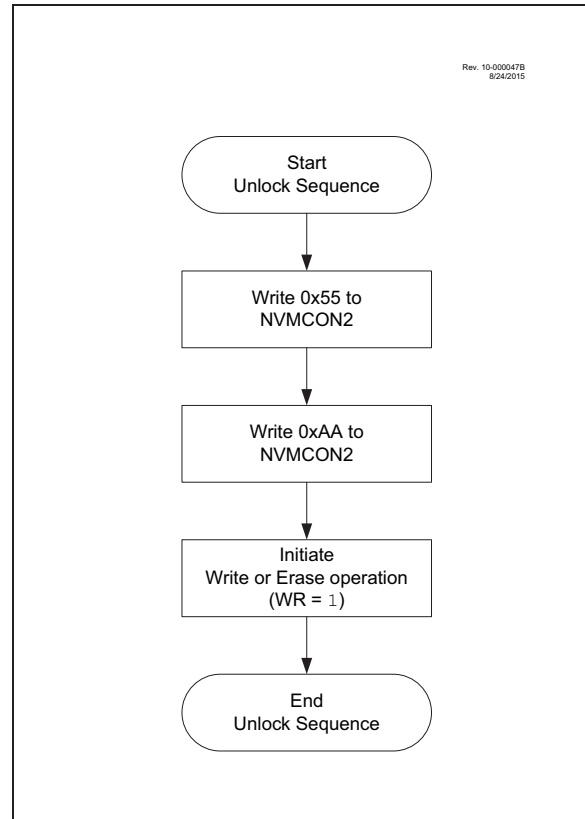
- Write 55h to NVMCON2
- Write AAh to NVMCON2
- Set the WR bit of NVMCON1

Once the WR bit is set, the processor will stall internal operations until the operation is complete and then resume with the next instruction.

Note: The two NOP instructions after setting the WR bit that were required in previous devices are not required for PIC16(L)F15324/44 devices. See Figure 13-2.

Since the unlock sequence must not be interrupted, global interrupts should be disabled prior to the unlock sequence and re-enabled after the unlock sequence is completed.

FIGURE 13-2: NVM UNLOCK SEQUENCE FLOWCHART



EXAMPLE 13-2: NVM UNLOCK SEQUENCE

```

BCF      INTCON, GIE      ; Recommended so sequence is not interrupted
BANKSEL  NVMCON1          ;
BSF      NVMCON1, WREN    ; Enable write/erase
MOVLW    55h              ; Load 55h
MOVWF    NVMCON2          ; Step 1: Load 55h into NVMCON2
MOVLW    AAh              ; Step 2: Load W with AAh
MOVWF    NVMCON2          ; Step 3: Load AAH into NVMCON2
BSF      NVMCON1, WR      ; Step 4: Set WR bit to begin write/erase
BSF      INTCON, GIE      ; Re-enable interrupts
    
```

- Note 1:** Sequence begins when NVMCON2 is written; steps 1-4 must occur in the cycle-accurate order shown.
2: Opcodes shown are illustrative; any instruction that has the indicated effect may be used.

26.6 Timer1 Interrupts

The timer register pair (TMR1H:TMR1L) increments to FFFFh and rolls over to 0000h. When the timer rolls over, the respective timer interrupt flag bit of the PIR5 register is set. To enable the interrupt on rollover, you must set these bits:

- ON bit of the T1CON register
- TMR1IE bit of the PIE4 register
- PEIE bit of the INTCON register
- GIE bit of the INTCON register

The interrupt is cleared by clearing the TMR1IF bit in the Interrupt Service Routine.

Note: To avoid immediate interrupt vectoring, the TMR1H:TMR1L register pair should be preloaded with a value that is not imminently about to rollover, and the TMR1IF flag should be cleared prior to enabling the timer interrupts.

26.7 Timer1 Operation During Sleep

Timer1 can only operate during Sleep when setup in Asynchronous Counter mode. In this mode, an external crystal or clock source can be used to increment the counter. To set up the timer to wake the device:

- ON bit of the T1CON register must be set
- TMR1IE bit of the PIE4 register must be set
- PEIE bit of the INTCON register must be set
- SYNC bit of the T1CON register must be set
- CS bits of the T1CLK register must be configured
- The timer clock source must be enabled and continue operation during sleep.

The device will wake-up on an overflow and execute the next instructions. If the GIE bit of the INTCON register is set, the device will call the Interrupt Service Routine.

Secondary oscillator will continue to operate in Sleep regardless of the SYNC bit setting.

26.8 CCP Capture/Compare Time Base

The CCP modules use the TMR1H:TMR1L register pair as the time base when operating in Capture or Compare mode.

In Capture mode, the value in the TMR1H:TMR1L register pair is copied into the CCPRxH:CCPRxL register pair on a configured event.

In Compare mode, an event is triggered when the value CCPRxH:CCPRxL register pair matches the value in the TMR1H:TMR1L register pair. This event can be an Auto-conversion Trigger.

For more information, see [Section 28.0 “Capture/Compare/PWM Modules”](#).

26.9 CCP Auto-Conversion Trigger

When any of the CCP's are configured to trigger an auto-conversion, the trigger will clear the TMR1H:TMR1L register pair. This auto-conversion does not cause a timer interrupt. The CCP module may still be configured to generate a CCP interrupt.

In this mode of operation, the CCPRxH:CCPRxL register pair becomes the period register for Timer1.

The timer should be synchronized and FOSC/4 should be selected as the clock source in order to utilize the Auto-conversion Trigger. Asynchronous operation of the timer can cause an Auto-conversion Trigger to be missed.

In the event that a write to TMR1H or TMR1L coincides with an Auto-conversion Trigger from the CCP, the write will take precedence.

For more information, see [Section 28.2.4 “Compare During Sleep”](#).

REGISTER 26-4: T1GATE TIMER1 GATE SELECT REGISTER

U-0	U-0	U-0	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u
—	—	—	GSS<4:0>				
bit 7			bit 0				

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

u = Bit is unchanged

x = Bit is unknown

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

'1' = Bit is set

'0' = Bit is cleared

HC = Bit is cleared by hardware

bit 7-5

Unimplemented: Read as '0'

bit 4-0

GSS<4:0>: Timer1 Gate Select bits

11111-10001 = Reserved

10000 = LC4_out

01111 = LC3_out

01110 = LC2_out

01101 = LC1_out

00100 = ZCD1_output

01011 = C2OUT_sync

01010 = C1OUT_sync

01001 = NCO1_out

01000 = PWM6_out

00111 = PWM5_out

00110 = PWM4_out

00101 = PWM3_out

00100 = CCP2_out

00011 = CCP1_out

00010 = TMR2_postscaled

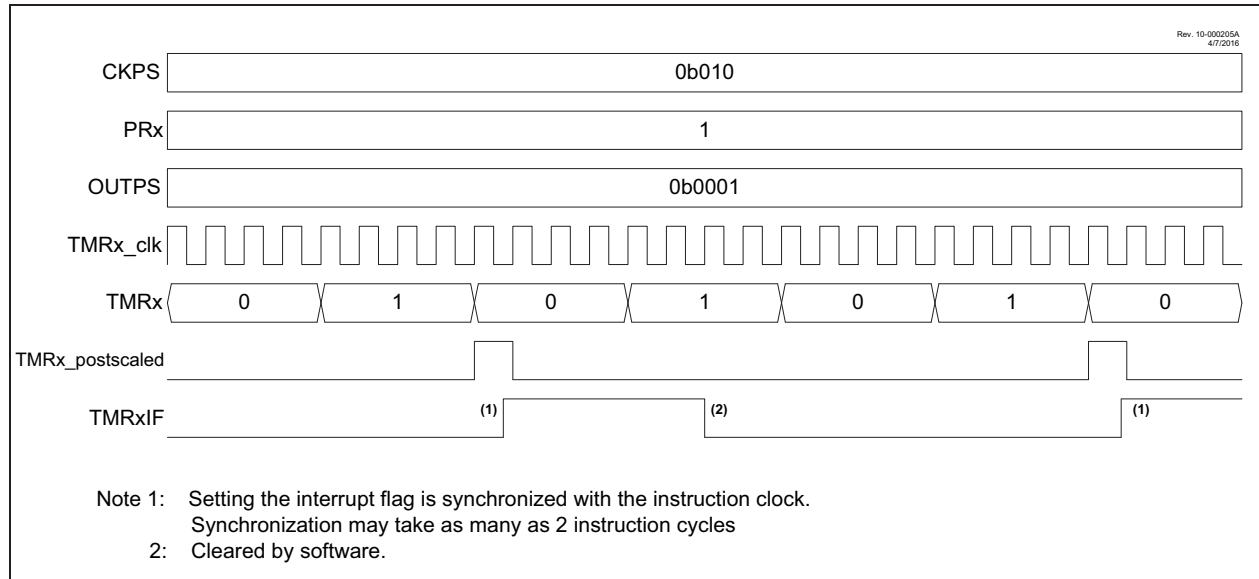
00001 = Timer0 overflow output

00000 = T1GPPS

27.4 Timer2 Interrupt

Timer2 can also generate a device interrupt. The interrupt is generated when the postscaler counter matches one of 16 postscale options (from 1:1 through 1:16), which are selected with the postscaler control bits, OUTPS<3:0> of the T2CON register. The interrupt is enabled by setting the TMR2IE interrupt enable bit of the PIE4 register. Interrupt timing is illustrated in Figure 27-3.

FIGURE 27-3: TIMER2 PRESCALER, POSTSCALER, AND INTERRUPT TIMING DIAGRAM



27.5 Operation Examples

Unless otherwise specified, the following notes apply to the following timing diagrams:

- Both the prescaler and postscaler are set to 1:1 (both the CKPS and OUTPS bits in the T2CON register are cleared).
- The diagrams illustrate any clock except $F_{osc}/4$ and show clock-sync delays of at least two full cycles for both ON and Timer2_ers. When using $F_{osc}/4$, the clock-sync delay is at least one instruction period for Timer2_ers; ON applies in the next instruction period.
- The PWM Duty Cycle and PWM output are illustrated assuming that the timer is used for the PWM function of the CCP module as described in **Section 28.0 "Capture/Compare/PWM Modules"**. The signals are not a part of the Timer2 module.

27.5.1 SOFTWARE GATE MODE

This mode corresponds to legacy Timer2 operation. The timer increments with each clock input when $ON = 1$ and does not increment when $ON = 0$. When the TMRx count equals the PRx period count the timer resets on the next clock and continues counting from 0. Operation with the ON bit software controlled is illustrated in Figure 27-4. With $PRx = 5$, the counter advances until $TMRx = 5$, and goes to zero with the next clock.

REGISTER 27-3: T2HLT: TIMERx HARDWARE LIMIT CONTROL REGISTER

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
PSYNC ^(1, 2)	CKPOL ⁽³⁾	CKSYNC ^(4, 5)	MODE<4:0> ^(6, 7)				
bit 7							bit 0

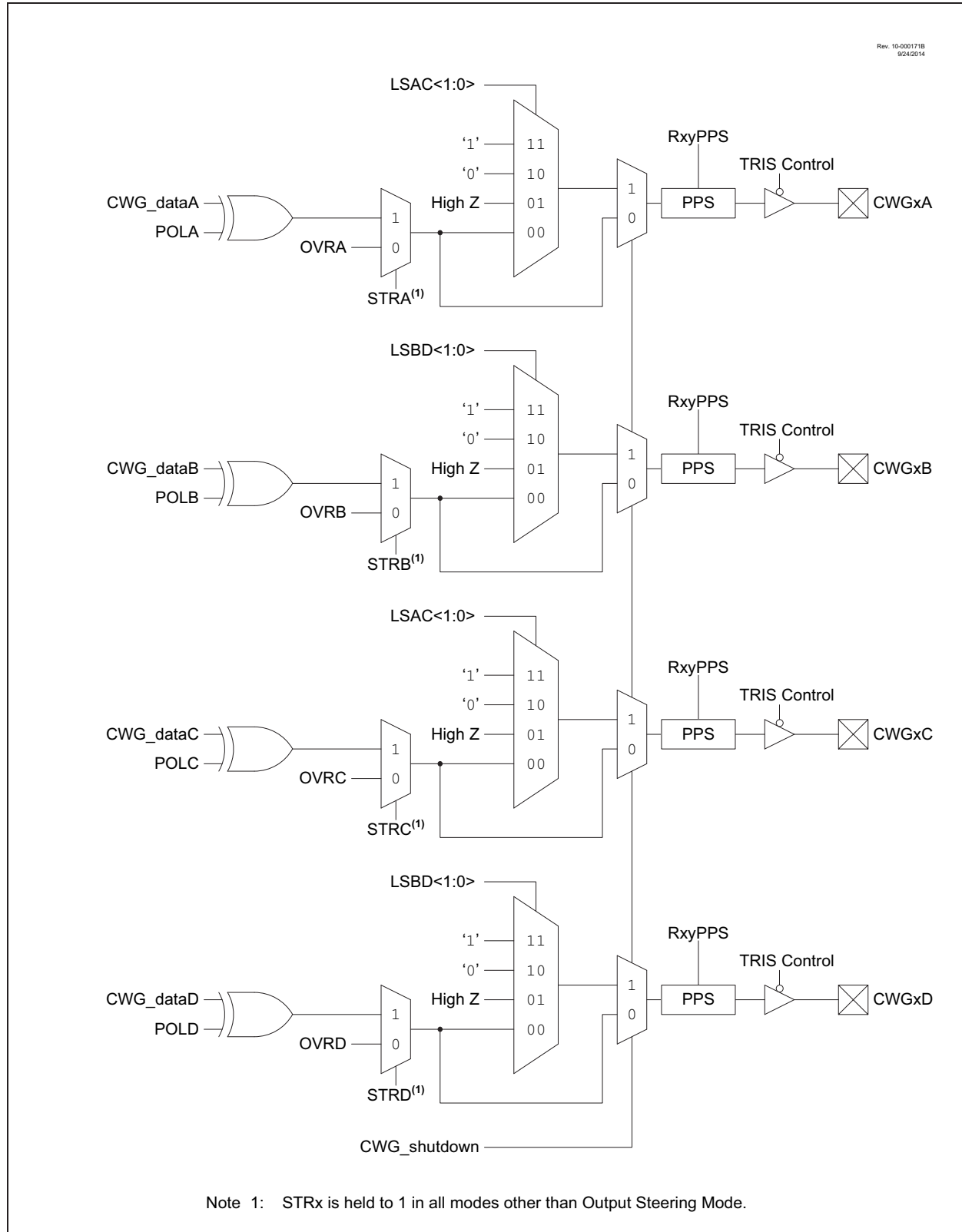
Legend:

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 **PSYNC:** Timerx Prescaler Synchronization Enable bit^(1, 2)
1 = TMRx Prescaler Output is synchronized to Fosc/4
0 = TMRx Prescaler Output is not synchronized to Fosc/4
- bit 6 **CKPOL:** Timerx Clock Polarity Selection bit⁽³⁾
1 = Falling edge of input clock clocks timer/prescaler
0 = Rising edge of input clock clocks timer/prescaler
- bit 5 **CKSYNC:** Timerx Clock Synchronization Enable bit^(4, 5)
1 = ON register bit is synchronized to TMR2_clk input
0 = ON register bit is not synchronized to TMR2_clk input
- bit 4-0 **MODE<4:0>:** Timerx Control Mode Selection bits^(6, 7)
See [Table 27-1](#).

- Note 1:** Setting this bit ensures that reading TMRx will return a valid value.
- 2:** When this bit is '1', Timer2 cannot operate in Sleep mode.
- 3:** CKPOL should not be changed while ON = 1.
- 4:** Setting this bit ensures glitch-free operation when the ON is enabled or disabled.
- 5:** When this bit is set then the timer operation will be delayed by two TMRx input clocks after the ON bit is set.
- 6:** Unless otherwise indicated, all modes start upon ON = 1 and stop upon ON = 0 (stops occur without affecting the value of TMRx).
- 7:** When TMRx = PRx, the next clock clears TMRx, regardless of the operating mode.

FIGURE 30-5: CWG OUTPUT BLOCK DIAGRAM



REGISTER 32-5: SSP1MSK: SSP1 MASK REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
SSP1MSK<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 7-1 **SSP1MSK<7:1>**: Mask bits
 1 = The received address bit n is compared to SSP1ADD<n> to detect I²C address match
 0 = The received address bit n is not used to detect I²C address match
- bit 0 **SSP1MSK<0>**: Mask bit for I²C Slave mode, 10-bit Address
I²C Slave mode, 10-bit address (SSPM<3:0> = 0111 or 1111):
 1 = The received address bit 0 is compared to SSP1ADD<0> to detect I²C address match
 0 = The received address bit 0 is not used to detect I²C address match
I²C Slave mode, 7-bit address:
 MSK0 bit is ignored.

REGISTER 32-6: SSP1ADD: MSSP1 ADDRESS AND BAUD RATE REGISTER (I²C MODE)

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
SSP1ADD<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	

Master mode:

- bit 7-0 **SSP1ADD<7:0>**: Baud Rate Clock Divider bits

$$\text{SCL pin clock period} = ((\text{ADD}<7:0> + 1) * 4) / \text{Fosc}$$

10-Bit Slave mode – Most Significant Address Byte:

- bit 7-3 **Not used**: Unused for Most Significant Address Byte. Bit state of this register is a “don't care”. Bit pattern sent by master is fixed by I²C specification and must be equal to '11110'. However, those bits are compared by hardware and are not affected by the value in this register.
- bit 2-1 **SSP1ADD<2:1>**: Two Most Significant bits of 10-bit address
- bit 0 **Not used**: Unused in this mode. Bit state is a “don't care”.

10-Bit Slave mode – Least Significant Address Byte:

- bit 7-0 **SSP1ADD<7:0>**: Eight Least Significant bits of 10-bit address

7-Bit Slave mode:

- bit 7-1 **SSP1ADD<7:1>**: 7-bit address
- bit 0 **Not used**: Unused in this mode. Bit state is a “don't care”.

FIGURE 33-10: SYNCHRONOUS TRANSMISSION

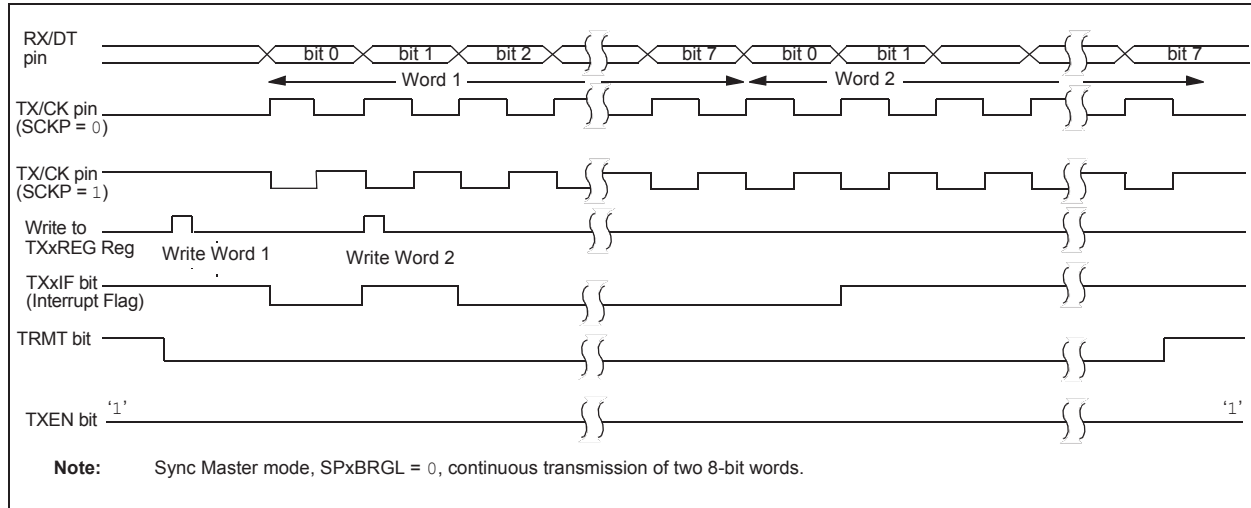
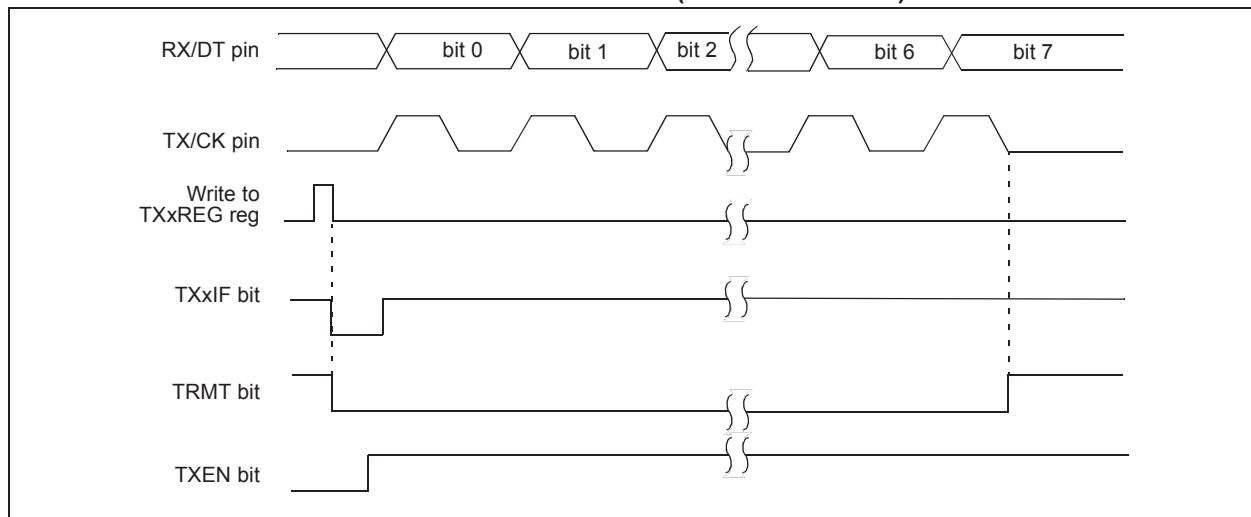


FIGURE 33-11: SYNCHRONOUS TRANSMISSION (THROUGH TXEN)



33.4.1.5 Synchronous Master Reception

Data is received at the RX/DT pin. The RX/DT pin output driver is automatically disabled when the EUSART is configured for synchronous master receive operation.

In Synchronous mode, reception is enabled by setting either the Single Receive Enable bit (SREN of the RCxSTA register) or the Continuous Receive Enable bit (CREN of the RCxSTA register).

When SREN is set and CREN is clear, only as many clock cycles are generated as there are data bits in a single character. The SREN bit is automatically cleared at the completion of one character. When CREN is set, clocks are continuously generated until CREN is cleared. If CREN is cleared in the middle of a character the CK clock stops immediately and the partial character is discarded. If SREN and CREN are both set, then SREN is cleared at the completion of the first character and CREN takes precedence.

To initiate reception, set either SREN or CREN. Data is sampled at the RX/DT pin on the trailing edge of the TX/CK clock pin and is shifted into the Receive Shift Register (RSR). When a complete character is received into the RSR, the RXxIF bit is set and the character is automatically transferred to the two character receive FIFO. The Least Significant eight bits of the top character in the receive FIFO are available in RCxREG. The RXxIF bit remains set as long as there are unread characters in the receive FIFO.

Note: If the RX/DT function is on an analog pin, the corresponding ANSEL bit must be cleared for the receiver to function.

33.6 Register Definitions: EUSART Control

REGISTER 33-1: TXxSTA: TRANSMIT STATUS AND CONTROL REGISTER

R/W-/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-1/1	R/W-0/0
CSRC	TX9	TXEN ⁽¹⁾	SYNC	SENDB	BRGH	TRMT	TX9D
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 **CSRC:** Clock Source Select bit
Asynchronous mode:
 Unused in this mode – value ignored
Synchronous mode:
 1 = Master mode (clock generated internally from BRG)
 0 = Slave mode (clock from external source)
- bit 6 **TX9:** 9-bit Transmit Enable bit
 1 = Selects 9-bit transmission
 0 = Selects 8-bit transmission
- bit 5 **TXEN:** Transmit Enable bit⁽¹⁾
 1 = Transmit enabled
 0 = Transmit disabled
- bit 4 **SYNC:** EUSART Mode Select bit
 1 = Synchronous mode
 0 = Asynchronous mode
- bit 3 **SENDB:** Send Break Character bit
Asynchronous mode:
 1 = Send SYNCH BREAK on next transmission – Start bit, followed by 12 '0' bits, followed by Stop bit; cleared by hardware upon completion
 0 = SYNCH BREAK transmission disabled or completed
Synchronous mode:
 Unused in this mode – value ignored
- bit 2 **BRGH:** High Baud Rate Select bit
Asynchronous mode:
 1 = High speed
 0 = Low speed
Synchronous mode:
 Unused in this mode – value ignored
- bit 1 **TRMT:** Transmit Shift Register Status bit
 1 = TSR empty
 0 = TSR full
- bit 0 **TX9D:** Ninth bit of Transmit Data
 Can be address/data bit or a parity bit.

Note 1: SREN/CREN overrides TXEN in Sync mode.

REGISTER 33-7: SPxBRGH^(1, 2): 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.

35.0 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
- 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 “PIC16(L)F153XX Memory Programming Specification” (DS40001838).

35.1 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 MCLR/VPP to VIH.

35.2 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 Words 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’. The LVP bit can only be reprogrammed to ‘0’ by using the High-Voltage Programming mode.

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

1. MCLR is brought to VIL.
2. A 32-bit key sequence is presented on ICSPDAT, while clocking ICSPCLK.

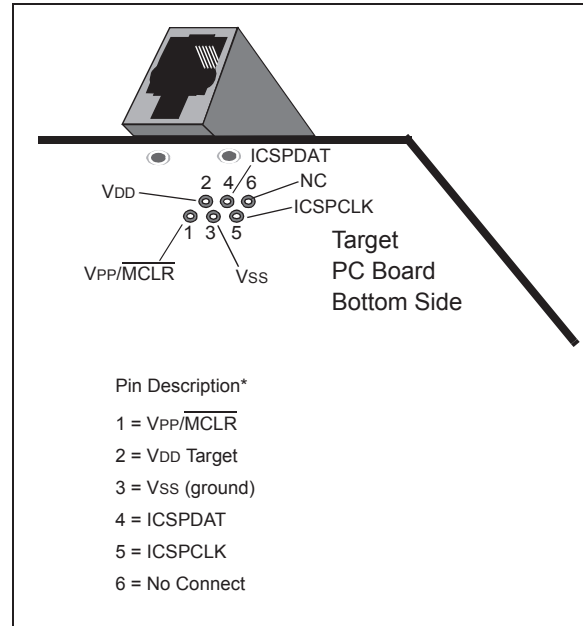
Once the key sequence is complete, MCLR must be held at VIL for as long as Program/Verify mode is to be maintained.

If low-voltage programming is enabled (LVP = 1), the MCLR Reset function is automatically enabled and cannot be disabled. See [Section 8.5 “MCLR”](#) for more information.

35.3 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-conductor) configuration. See [Figure 35-1](#).

FIGURE 35-1: 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 35-2](#).

For additional interface recommendations, refer to your specific device programmer manual prior to PCB design.

It is recommended that isolation devices be used to separate the programming pins from other circuitry. The type of isolation is highly dependent on the specific application and may include devices such as resistors, diodes, or even jumpers. See [Figure 35-3](#) for more information.

36.0 INSTRUCTION SET SUMMARY

Each instruction is a 14-bit word containing the operation code (opcode) and all required operands. The opcodes are broken into three broad categories.

- Byte Oriented
- Bit Oriented
- Literal and Control

The literal and control category contains the most varied instruction word format.

Table 36-3 lists the instructions recognized by the MPASM™ assembler.

All instructions are executed within a single instruction cycle, with the following exceptions, which may take two or three cycles:

- Subroutine entry takes two cycles (CALL, CALLW)
- Returns from interrupts or subroutines take two cycles (RETURN, RETLW, RETFIE)
- Program branching takes two cycles (GOTO, BRA, BRW, BTFSS, BTFSC, DECFSZ, INCSFZ)
- One additional instruction cycle will be used when any instruction references an indirect file register and the file select register is pointing to program memory.

One instruction cycle consists of 4 oscillator cycles; for an oscillator frequency of 4 MHz, this gives a nominal instruction execution rate of 1 MHz.

All instruction examples use the format '0xhh' to represent a hexadecimal number, where 'h' signifies a hexadecimal digit.

36.1 Read-Modify-Write Operations

Any instruction that specifies a file register as part of the instruction performs a Read-Modify-Write (R-M-W) operation. The register is read, the data is modified, and the result is stored according to either the instruction, or the destination designator 'd'. A read operation is performed on a register even if the instruction writes to that register.

TABLE 36-1: OPCODE FIELD DESCRIPTIONS

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1). The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1.
n	FSR or INDF number. (0-1)
mm	Prepost increment-decrement mode selection

TABLE 36-2: ABBREVIATION DESCRIPTIONS

Field	Description
PC	Program Counter
$\overline{\text{TO}}$	Time-Out bit
C	Carry bit
DC	Digit Carry bit
Z	Zero bit
$\overline{\text{PD}}$	Power-Down bit

39.11 Demonstration/Development Boards, Evaluation Kits, and Starter Kits

A wide variety of demonstration, development and evaluation boards for various PIC MCUs and dsPIC DSCs allows quick application development on fully functional systems. Most boards include prototyping areas for adding custom circuitry and provide application firmware and source code for examination and modification.

The boards support a variety of features, including LEDs, temperature sensors, switches, speakers, RS-232 interfaces, LCD displays, potentiometers and additional EEPROM memory.

The demonstration and development boards can be used in teaching environments, for prototyping custom circuits and for learning about various microcontroller applications.

In addition to the PICDEM™ and dsPICDEM™ demonstration/development board series of circuits, Microchip has a line of evaluation kits and demonstration software for analog filter design, KEELOQ® security ICs, CAN, IrDA®, PowerSmart battery management, SEEVAL® evaluation system, Sigma-Delta ADC, flow rate sensing, plus many more.

Also available are starter kits that contain everything needed to experience the specified device. This usually includes a single application and debug capability, all on one board.

Check the Microchip web page (www.microchip.com) for the complete list of demonstration, development and evaluation kits.

39.12 Third-Party Development Tools

Microchip also offers a great collection of tools from third-party vendors. These tools are carefully selected to offer good value and unique functionality.

- Device Programmers and Gang Programmers from companies, such as SoftLog and CCS
- Software Tools from companies, such as Gimpel and Trace Systems
- Protocol Analyzers from companies, such as Saleae and Total Phase
- Demonstration Boards from companies, such as MikroElektronika, Digilent® and Olimex
- Embedded Ethernet Solutions from companies, such as EZ Web Lynx, WIZnet and IPLogika®

Worldwide Sales and Service

AMERICAS

Corporate Office
 2355 West Chandler Blvd.
 Chandler, AZ 85224-6199
 Tel: 480-792-7200
 Fax: 480-792-7277
 Technical Support:
<http://www.microchip.com/support>
 Web Address:
www.microchip.com

Atlanta
 Duluth, GA
 Tel: 678-957-9614
 Fax: 678-957-1455

Austin, TX
 Tel: 512-257-3370

Boston
 Westborough, MA
 Tel: 774-760-0087
 Fax: 774-760-0088

Chicago
 Itasca, IL
 Tel: 630-285-0071
 Fax: 630-285-0075

Dallas
 Addison, TX
 Tel: 972-818-7423
 Fax: 972-818-2924

Detroit
 Novi, MI
 Tel: 248-848-4000

Houston, TX
 Tel: 281-894-5983

Indianapolis
 Noblesville, IN
 Tel: 317-773-8323
 Fax: 317-773-5453
 Tel: 317-536-2380

Los Angeles
 Mission Viejo, CA
 Tel: 949-462-9523
 Fax: 949-462-9608
 Tel: 951-273-7800

Raleigh, NC
 Tel: 919-844-7510

New York, NY
 Tel: 631-435-6000

San Jose, CA
 Tel: 408-735-9110
 Tel: 408-436-4270

Canada - Toronto
 Tel: 905-695-1980
 Fax: 905-695-2078

ASIA/PACIFIC

Asia Pacific Office
 Suites 3707-14, 37th Floor
 Tower 6, The Gateway
 Harbour City, Kowloon

Hong Kong
 Tel: 852-2943-5100
 Fax: 852-2401-3431

Australia - Sydney
 Tel: 61-2-9868-6733
 Fax: 61-2-9868-6755

China - Beijing
 Tel: 86-10-8569-7000
 Fax: 86-10-8528-2104

China - Chengdu
 Tel: 86-28-8665-5511
 Fax: 86-28-8665-7889

China - Chongqing
 Tel: 86-23-8980-9588
 Fax: 86-23-8980-9500

China - Dongguan
 Tel: 86-769-8702-9880

China - Guangzhou
 Tel: 86-20-8755-8029

China - Hangzhou
 Tel: 86-571-8792-8115
 Fax: 86-571-8792-8116

China - Hong Kong SAR
 Tel: 852-2943-5100
 Fax: 852-2401-3431

China - Nanjing
 Tel: 86-25-8473-2460
 Fax: 86-25-8473-2470

China - Qingdao
 Tel: 86-532-8502-7355
 Fax: 86-532-8502-7205

China - Shanghai
 Tel: 86-21-3326-8000
 Fax: 86-21-3326-8021

China - Shenyang
 Tel: 86-24-2334-2829
 Fax: 86-24-2334-2393

China - Shenzhen
 Tel: 86-755-8864-2200
 Fax: 86-755-8203-1760

China - Wuhan
 Tel: 86-27-5980-5300
 Fax: 86-27-5980-5118

China - Xian
 Tel: 86-29-8833-7252
 Fax: 86-29-8833-7256

ASIA/PACIFIC

China - Xiamen
 Tel: 86-592-2388138
 Fax: 86-592-2388130

China - Zhuhai
 Tel: 86-756-3210040
 Fax: 86-756-3210049

India - Bangalore
 Tel: 91-80-3090-4444
 Fax: 91-80-3090-4123

India - New Delhi
 Tel: 91-11-4160-8631
 Fax: 91-11-4160-8632

India - Pune
 Tel: 91-20-3019-1500

Japan - Osaka
 Tel: 81-6-6152-7160
 Fax: 81-6-6152-9310

Japan - Tokyo
 Tel: 81-3-6880-3770
 Fax: 81-3-6880-3771

Korea - Daegu
 Tel: 82-53-744-4301
 Fax: 82-53-744-4302

Korea - Seoul
 Tel: 82-2-554-7200
 Fax: 82-2-558-5932 or
 82-2-558-5934

Malaysia - Kuala Lumpur
 Tel: 60-3-6201-9857
 Fax: 60-3-6201-9859

Malaysia - Penang
 Tel: 60-4-227-8870
 Fax: 60-4-227-4068

Philippines - Manila
 Tel: 63-2-634-9065
 Fax: 63-2-634-9069

Singapore
 Tel: 65-6334-8870
 Fax: 65-6334-8850

Taiwan - Hsin Chu
 Tel: 886-3-5778-366
 Fax: 886-3-5770-955

Taiwan - Kaohsiung
 Tel: 886-7-213-7830

Taiwan - Taipei
 Tel: 886-2-2508-8600
 Fax: 886-2-2508-0102

Thailand - Bangkok
 Tel: 66-2-694-1351
 Fax: 66-2-694-1350

EUROPE

Austria - Wels
 Tel: 43-7242-2244-39
 Fax: 43-7242-2244-393

Denmark - Copenhagen
 Tel: 45-4450-2828
 Fax: 45-4485-2829

Finland - Espoo
 Tel: 358-9-4520-820

France - Paris
 Tel: 33-1-69-53-63-20
 Fax: 33-1-69-30-90-79

France - Saint Cloud
 Tel: 33-1-30-60-70-00

Germany - Garching
 Tel: 49-8931-9700

Germany - Haan
 Tel: 49-2129-3766400

Germany - Heilbronn
 Tel: 49-7131-67-3636

Germany - Karlsruhe
 Tel: 49-721-625370

Germany - Munich
 Tel: 49-89-627-144-0
 Fax: 49-89-627-144-44

Germany - Rosenheim
 Tel: 49-8031-354-560

Israel - Ra'anana
 Tel: 972-9-744-7705

Italy - Milan
 Tel: 39-0331-742611
 Fax: 39-0331-466781

Italy - Padova
 Tel: 39-049-7625286

Netherlands - Drunen
 Tel: 31-416-690399
 Fax: 31-416-690340

Norway - Trondheim
 Tel: 47-7289-7561

Poland - Warsaw
 Tel: 48-22-3325737

Romania - Bucharest
 Tel: 40-21-407-87-50

Spain - Madrid
 Tel: 34-91-708-08-90
 Fax: 34-91-708-08-91

Sweden - Gothenberg
 Tel: 46-31-704-60-40

Sweden - Stockholm
 Tel: 46-8-5090-4654

UK - Wokingham
 Tel: 44-118-921-5800
 Fax: 44-118-921-5820