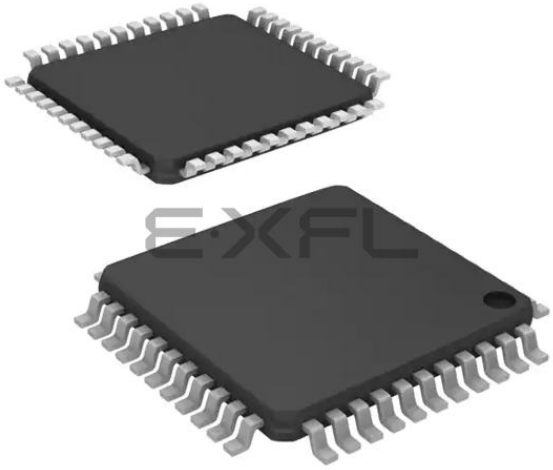


Welcome to E-XFL.COM

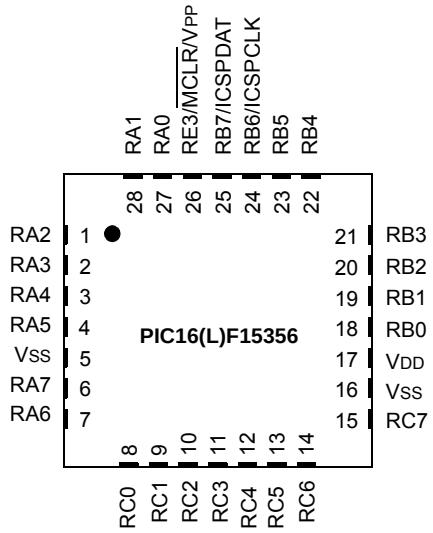
What is "Embedded - Microcontrollers"?

"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	36
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 35x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f15375-i-pt

28-PIN UQFN (4x4), UQFN (6x6)

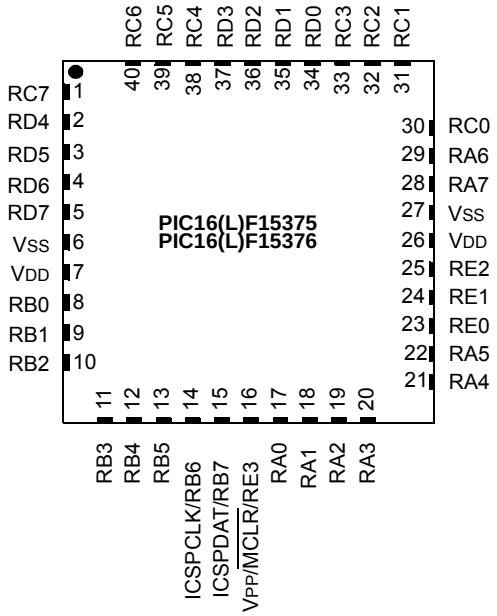


Note 1: See Table 3 for location of all peripheral functions.

2: All VDD and all VSS pins must be connected at the circuit board level. Allowing one or more VSS or VDD pins to float may result in degraded electrical performance or non-functionality.

3: The bottom pad of the QFN/UQFN package should be connected to VSS at the circuit board level.

40-PIN UQFN (5x5)



Note 1: See Table 4 for the pin allocation tables.

2: The bottom pad of the QFN/UQFN package should be connected to VSS at the circuit board level.

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 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

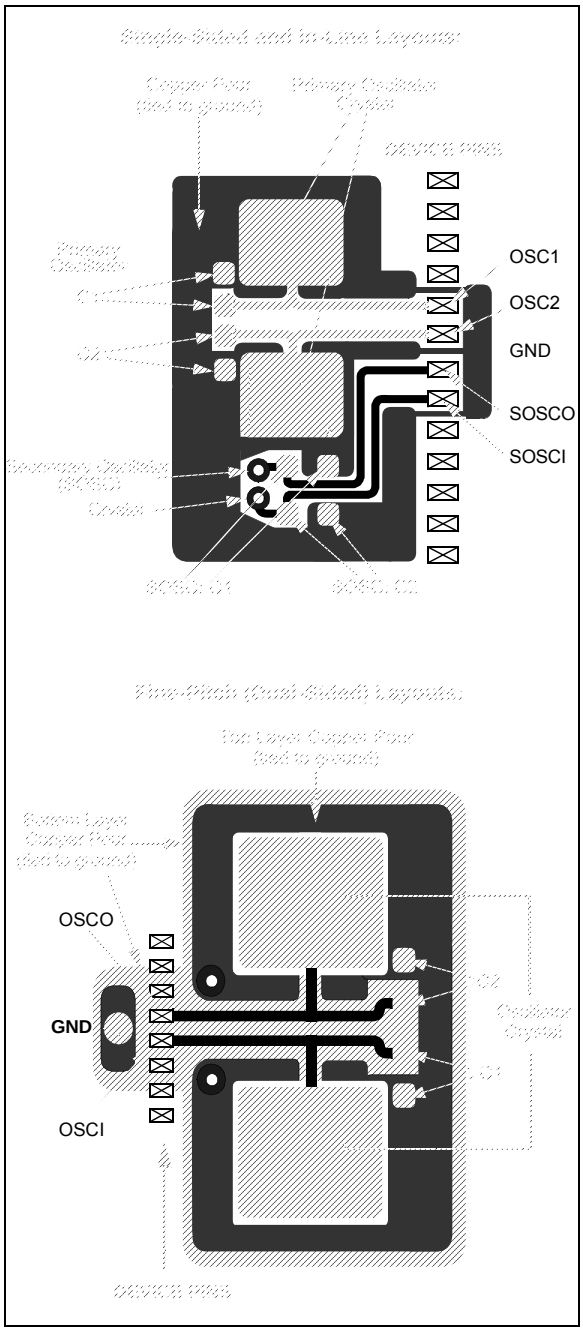


TABLE 4-5: PIC16(L)F15356/75/76/85/86 MEMORY MAP, BANKS 8-15

	BANK 8	BANK 9	BANK 10	BANK 11	BANK 12	BANK 13	BANK 14	BANK 15
400h	Core Register (Table 4-3)							
40Bh								
40Ch	—	—	—	NCO1ACCL	CWG1CLK	—	PIR0	—
40Dh	—	—	—	NCO1ACCH	CWG1DAT	—	PIR1	—
40Eh	—	—	—	NCO1ACCU	CWG1DBR	—	PIR2	—
40Fh	—	—	—	NCO1INCL	CWG1DBF	—	PIR3	—
410h	—	—	—	NCO1INCH	CWG1CON0	—	PIR4	—
411h	—	—	—	NCO1INCU	CWG1CON1	—	PIR5	—
412h	—	—	—	NCO1CON	CWG1AS0	—	PIR6	—
413h	—	—	—	NCO1CLK	CWG1AS1	—	PIR7	—
414h	—	—	—	—	CWG1STR	—	—	—
415h	—	—	—	—	—	—	—	—
416h	—	—	—	—	—	—	PIE0	PMD0
417h	—	—	—	—	—	—	PIE1	PMD1
418h	—	—	—	—	—	—	PIE2	PMD2
419h	—	—	—	—	—	—	PIE3	PMD3
41Ah	—	—	—	—	—	—	PIE4	PMD4
41Bh	—	—	—	—	—	—	PIE5	PMD5
41Ch	—	—	—	TMR0	—	—	PIE6	—
41Dh	—	—	—	PR0	—	—	PIE7	—
41Eh	—	—	—	TMR0CON0	—	—	—	—
41Fh	—	—	—	TMR0CON1	—	—	—	—
420h	General Purpose Register 80 Bytes				General Purpose Register 48 Bytes			
46Fh								
470h	Common RAM Accesses 70h-7Fh							
47Fh								
480h		Core Register (Table 4-3)						
48Bh								
48Ch		—	—	—	—	—	—	—
48Dh		—	—	—	—	—	—	—
48Eh		—	—	—	—	—	—	—
48Fh		—	—	—	—	—	—	—
490h		—	—	—	—	—	—	—
491h		—	—	—	—	—	—	—
492h		—	—	—	—	—	—	—
493h		—	—	—	—	—	—	—
494h		—	—	—	—	—	—	—
495h		—	—	—	—	—	—	—
496h		—	—	—	—	—	—	—
497h		—	—	—	—	—	—	—
498h		—	—	—	—	—	—	—
499h		—	—	—	—	—	—	—
49Ah		—	—	—	—	—	—	—
49Bh		—	—	—	—	—	—	—
49Ch		—	—	—	—	—	—	—
49Dh		—	—	—	—	—	—	—
49Eh		—	—	—	—	—	—	—
49Fh		—	—	—	—	—	—	—
4A0h		General Purpose Register 80 Bytes						
4EFh								
4F0h		Common RAM Accesses 70h-7Fh						
4FFh								
500h			Core Register (Table 4-3)					
50Bh								
50Ch			—	—	—	—	—	—
50Dh			—	—	—	—	—	—
50Eh			—	—	—	—	—	—
50Fh			—	—	—	—	—	—
510h			—	—	—	—	—	—
511h			—	—	—	—	—	—
512h			—	—	—	—	—	—
513h			—	—	—	—	—	—
514h			—	—	—	—	—	—
515h			—	—	—	—	—	—
516h			—	—	—	—	—	—
517h			—	—	—	—	—	—
518h			—	—	—	—	—	—
519h			—	—	—	—	—	—
51Ah			—	—	—	—	—	—
51Bh			—	—	—	—	—	—
51Ch			—	—	—	—	—	—
51Dh			—	—	—	—	—	—
51Eh			—	—	—	—	—	—
51Fh			—	—	—	—	—	—
520h			General Purpose Register 80 Bytes					
56Fh								
570h			Common RAM Accesses 70h-7Fh					
57Fh								
580h				Core Register (Table 4-3)				
58Bh								
58Ch				—	—	—	—	—
58Dh				—	—	—	—	—
58Eh				—	—	—	—	—
58Fh				—	—	—	—	—
590h				—	—	—	—	—
591h				—	—	—	—	—
592h				—	—	—	—	—
593h				—	—	—	—	—
594h				—	—	—	—	—
595h				—	—	—	—	—
596h				—	—	—	—	—
597h				—	—	—	—	—
598h				—	—	—	—	—
599h				—	—	—	—	—
59Ah				—	—	—	—	—
59Bh				—	—	—	—	—
59Ch				TMR0	—	—	—	—
59Dh				PR0	—	—	—	—
59Eh				TMR0CON0	—	—	—	—
59Fh				TMR0CON1	—	—	—	—
5A0h				General Purpose Register 80 Bytes				
5EFh								
5F0h				Common RAM Accesses 70h-7Fh				
5FFh								
600h					Core Register (Table 4-3)			
60Bh								
60Ch					CWG1CLK	—	PIR0	—
60Dh					CWG1DAT	—	PIR1	—
60Eh					CWG1DBR	—	PIR2	—
60Fh					CWG1DBF	—	PIR3	—
610h					CWG1CON0	—	PIR4	—
611h					CWG1CON1	—	PIR5	—
612h					CWG1AS0	—	PIR6	—
613h					CWG1AS1	—	PIR7	—
614h					CWG1STR	—	—	—
615h					—	—	—	—
616h					—	—	PIE0	PMD0
617h					—	—	PIE1	PMD1
618h					—	—	PIE2	PMD2
619h					—	—	PIE3	PMD3
61Ah					—	—	PIE4	PMD4
61Bh					—	—	PIE5	PMD5
61Ch					—	—	PIE6	—
61Dh					—	—	PIE7	—
61Eh					—	—	—	—
61Fh					—	—	—	—
620h					General Purpose Register 48 Bytes			
64Fh								
650h					General Purpose Register 32 Bytes(2)			
66Fh								
670h					Common RAM Accesses 70h-7Fh			
67Fh								
680h						Core Register (Table 4-3)		
68Bh								
68Ch						—	PIR0	—
68Dh						—	PIR1	—
68Eh						—	PIR2	—
68Fh						—	PIR3	—
690h						—	PIR4	—
691h						—	PIR5	—
692h						—	PIR6	—
693h						—	PIR7	—
694h						—	—	—
695h						—	—	—
696h						—	PIE0	PMD0
697h						—	PIE1	PMD1
698h						—	PIE2	PMD2
699h						—	PIE3	PMD3
69Ah						—	PIE4	PMD4
69Bh						—	PIE5	PMD5
69Ch						—	PIE6	—
69Dh						—	PIE7	—
69Eh						—	—	—
69Fh						—	—	—
6A0h						General Purpose Register 80 Bytes(2)		
6EFh								
6F0h						Common RAM Accesses 70h-7Fh		
6FFh								
700h							Core Register (Table 4-3)	
70Bh								
70Ch							PIR0	—
70Dh							PIR1	—
70Eh							PIR2	—
70Fh							PIR3	—
710h							PIR4	—
711h							PIR5	—
712h							PIR6	—
713h							PIR7	—
714h							—	—
715h							—	—
716h							PIE0	PMD0
717h							PIE1	PMD1
718h							PIE2	PMD2
719h							PIE3	PMD3
71Ah							PIE4	PMD4
71Bh							PIE5	PMD5
71Ch							PIE6	—
71Dh							PIE7	—
71Eh							—	—
71Fh							—	—
720h							General Purpose Register 80 Bytes(2)	
76Fh								
770h							Common RAM Accesses 70h-7Fh	
77Fh								
780h								Core Register (Table 4-3)
78Bh								
78Ch								—
78Dh								—
78Eh								—
78Fh								—
790h								—
791h								—
792h								—
793h								—
794h								—
795h								—
796h								PMD0
797h								PMD1
798h								PMD2
799h								PMD3
79Ah								PMD4
79Bh								PMD5
79Ch								—
79Dh								—
79Eh								—
79Fh								—
7A0h								General Purpose Register 80 Bytes(2)
7EFh								
7F0h								Common RAM Accesses 70h-7Fh
7FFh								

Note 1: Unimplemented locations read as '0'.

2: Present only on PIC16(L)F15356/76/86.

TABLE 4-11: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-63 (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on: MCLR
Bank 15											
CPU CORE REGISTERS; see Table 4-3 for specifics											
78Ch — 795h	—	Unimplemented								—	—
796h	PMD0	SYSCMD	FVRMD	—	—	—	NVMMD	CLKRMD	IOCMD	00-- -000	00-- -000
797h	PMD1	NCO1MD	—	—	—	—	TMR2MD	TMR1MD	TMR0MD	0--- -000	0--- -000
798h	PMD2	—	DAC1MD	ADCMD	—	—	CMP2MD	CMP1MD	ZCDMD	-00- -000	-00- -000
799h	PMD3	—	—	PWM6MD	PWM5MD	PWM4MD	PWM3MD	CCP2MD	CCP1MD	--00 0000	--00 0000
79Ah	PMD4	UART2MD	UART1MD	MSSP2MD	MSSP1MD	—	—	—	CWG1MD	0000 ---0	0000 ---0
79Bh	PMD5	—	—	—	CLC4MD	CLC3MD	CLC2MD	CLC1MD	—	---0 000-	---0 000-
79Ch	—	Unimplemented								—	—
79Dh	—	Unimplemented								—	—
79Eh	—	Unimplemented								—	—
79Fh	—	Unimplemented								—	—

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

TABLE 4-11: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-63 (CONTINUED)

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on: MCLR
Bank 18											
CPU CORE REGISTERS; see Table 4-3 for specifics											
90Ch	FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>		ADFVR<1:0>		0x00 xxxx	0q00 uuuu
90Dh	—	Unimplemented								—	—
90Eh	DAC1CON0	EN	—	OE1	OE2	PSS<1:0>		—	NSS	0-00 00-0	0-00 00-0
90Fh	DAC1CON1	—	—	—	DAC1R<4:0>					---0 0000	---0 0000
910h — 91Eh	—	Unimplemented								—	—
91Fh	ZCDCON	ZCDSEN	—	ZCDOUT	ZCDPOL	—	—	ZCDINTP	ZCDINTN	0-x0 --00	0-x0 --00

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

REGISTER 10-3: PIE1: PERIPHERAL INTERRUPT ENABLE REGISTER 1

R/W-0/0	R/W-0/0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0
OSFIE	CSWIE	—	—	—	—	—	ADIE
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 **OSFIE:** Oscillator Fail Interrupt Enable bit
1 = Enables the Oscillator Fail Interrupt
0 = Disables the Oscillator Fail Interrupt

bit 6 **CSWIE:** Clock Switch Complete Interrupt Enable bit
1 = The clock switch module interrupt is enabled
0 = The clock switch module interrupt is disabled

bit 5-1 **Unimplemented:** Read as ‘0’

bit 0 **ADIE:** Analog-to-Digital Converter (ADC) Interrupt Enable bit
1 = Enables the ADC interrupt
0 = Disables the ADC interrupt

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt controlled by registers PIE1-PIE7

REGISTER 10-12: PIR2: PERIPHERAL INTERRUPT REQUEST REGISTER 2

U-0	R/W/HS-0/0	U-0	U-0	U-0	U-0	R/W/HS-0/0	R/W/HS-0/0
—	ZCDIF	—	—	—	—	C2IF	C1IF
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 **Unimplemented**: Read as '0'

bit 6 **ZCDIF:** Zero-Cross Detect (ZCD1) Interrupt Flag bit
1 = An enabled rising and/or falling ZCD1 event has been detected (must be cleared in software)
0 = No ZCD1 event has occurred

bit 5-2 **Unimplemented**: Read as '0'

bit 1 **C2IF**: Comparator C2 Interrupt Flag bit
1 = Comparator 2 interrupt asserted (must be cleared in software)
0 = Comparator 2 interrupt not asserted

bit 0 **C1IF:** Comparator C1 Interrupt Flag bit
1 = Comparator 1 interrupt asserted (must be cleared in software)
0 = Comparator 1 interrupt not asserted

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.

REGISTER 10-16: PIR6: PERIPHERAL INTERRUPT REQUEST REGISTER 6

U-0	U-0	U-0	U-0	U-0	U-0	R/W/HS-0/0	R/W/HS-0/0
—	—	—	—	—	—	CCP2IF	CCP1IF
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 7-2 **Unimplemented:** Read as '0'

bit 1 **CCP2IF:** CCP2 Interrupt Flag bit

Value	CCPM Mode		
	Capture	Compare	PWM
1	Capture occurred (must be cleared in software)	Compare match occurred (must be cleared in software)	Output trailing edge occurred (must be cleared in software)
0	Capture did not occur	Compare match did not occur	Output trailing edge did not occur

bit 0 **CCP1IF:** CCP1 Interrupt Flag bit

Value	CCPM Mode		
	Capture	Compare	PWM
1	Capture occurred (must be cleared in software)	Compare match occurred (must be cleared in software)	Output trailing edge occurred (must be cleared in software)
0	Capture did not occur	Compare match did not occur	Output trailing edge did not occur

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.

14.9 Register Definitions: PORTD

REGISTER 14-25: PORTD: PORTD 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
RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0
bit 7							bit 0

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

bit 7-0 **RD<7:0>**: PORTD General Purpose I/O Pin bits(1)
1 = Port pin is ≥ VIH
0 = Port pin is ≤ VIL

Note 1: Writes to PORTD are actually written to corresponding LATD register. Reads from PORTD register is return of actual I/O pin values.

REGISTER 14-26: TRISD: PORTD TRI-STATE 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
TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0
bit 7							bit 0

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

bit 7-0 **TRISD<7:0>:** PORTD Tri-State Control bits
1 = PORTD pin configured as an input (tri-stated)
0 = PORTD pin configured as an output

REGISTER 14-27: LATD: PORTD DATA LATCH 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
LATD7	LATD6	LATD5	LATD4	LATD3	LATD2	LATD1	LATD0
bit 7							bit 0

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

bit 7-0 **LATD<7:0>**: PORTD Output Latch Value bits(1)

Note 1: Writes to PORTD are actually written to corresponding LATD register. Reads from PORTD register is return of actual I/O pin values.

14.13 Register Definitions: PORTF

REGISTER 14-41: PORTF: PORTF 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
RF7	RF6	RF5	RF4	RF3	RF2	RF1	RF0
bit 7							bit 0

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

bit 7-0 **RF<7:0>**: PORTF General Purpose I/O Pin bits(1)
1 = Port pin is ≥ VIH
0 = Port pin is ≤ VIL

Note 1: Writes to PORTF are actually written to corresponding LATF register. Reads from PORTF register is return of actual I/O pin values.

REGISTER 14-42: TRISF: PORTF TRI-STATE 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
TRISF7	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2	TRISF1	TRISF0
bit 7							bit 0

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

bit 7-0 **TRISF<7:0>:** PORTF Tri-State Control bits
1 = PORTF pin configured as an input (tri-stated)
0 = PORTF pin configured as an output

REGISTER 14-43: LATF: PORTF DATA LATCH 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
LATF7	LATF6	LATF5	LATF4	LATF3	LATF2	LATF1	LATF0
bit 7							bit 0

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

bit 7-0 **LATF<7:0>**: PORTF Output Latch Value bits(1)

Note 1: Writes to PORTF are actually written to corresponding LATF register. Reads from PORTF register is return of actual I/O pin values.

20.3 ADC Acquisition Requirements

For the ADC to meet its specified accuracy, the charge holding capacitor (CHOLD) must be allowed to fully charge to the input channel voltage level. The Analog Input model is shown in Figure 20-4. The source impedance (RS) and the internal sampling switch (RSS) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch (RSS) impedance varies over the device voltage (VDD), refer to Figure 20-4. **The maximum recommended impedance for analog sources is 10 kΩ.** As the source impedance is decreased, the acquisition time may be decreased. After the analog input channel is selected (or changed), an ADC acquisition must be done before the conversion can be started. To calculate the minimum acquisition time, Equation 20-1 may be used. This equation assumes that 1/2 LSb error is used (1,024 steps for the ADC). The 1/2 LSb error is the maximum error allowed for the ADC to meet its specified resolution.

EQUATION 20-1: ACQUISITION TIME EXAMPLE

Assumptions: $Temperature = 50°C$ *and external impedance of* $10k\Omega$ *5.0V* V_{DD}

$$T_{ACQ} = \text{Amplifier Settling Time} + \text{Hold Capacitor Charging Time} + \text{Temperature Coefficient}$$
$$= T_{AMP} + T_C + T_{COFF}$$
$$= 2\mu s + T_C + [(\text{Temperature} - 25°C)(0.05\mu s/°C)]$$

The value for T_C *can be approximated with the following equations:*

$$V_{APPLIED}\left(1 - \frac{1}{(2^{n+1}) - 1}\right) = V_{CHOLD} \quad ;[1]\ V_{CHOLD}\ \text{charged to within 1/2 lsb}$$

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{RC}}\right) = V_{CHOLD} \quad ;[2]\ V_{CHOLD}\ \text{charge response to}\ V_{APPLIED}$$

$$V_{APPLIED}\left(1 - e^{\frac{-T_C}{RC}}\right) = V_{APPLIED}\left(1 - \frac{1}{(2^{n+1}) - 1}\right) \quad ;\text{combining [1] and [2]}$$

Note: Where n = number of bits of the ADC.

Solving for T_C*:*

$$T_C = -C_{HOLD}(R_{IC} + R_{SS} + R_S)\ ln(1/2047)$$
$$= -10pF(1k\Omega + 7k\Omega + 10k\Omega)\ ln(0.0004885)$$
$$= 1.37\mu s$$

Therefore:

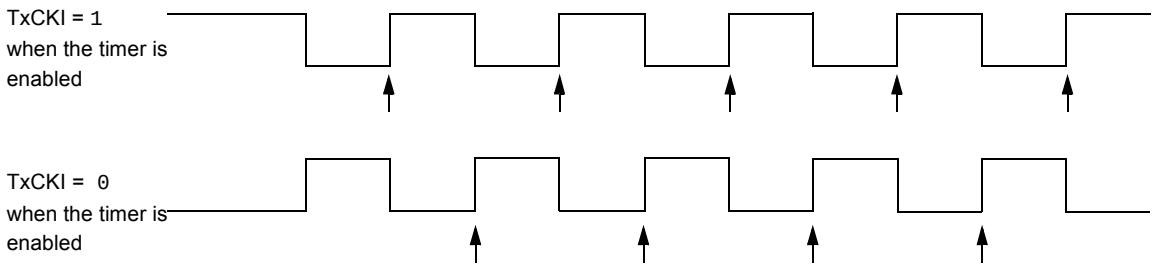
$$T_{ACQ} = 2\mu s + 1.37 + [(50°C - 25°C)(0.05\mu s/°C)]$$
$$= 4.62\mu s$$

Note 1: The VAPPLIED has no effect on the equation, since it cancels itself out.

2: The charge holding capacitor (CHOLD) is not discharged after each conversion.

3: The maximum recommended impedance for analog sources is 10 kΩ. This is required to meet the pin leakage specification.

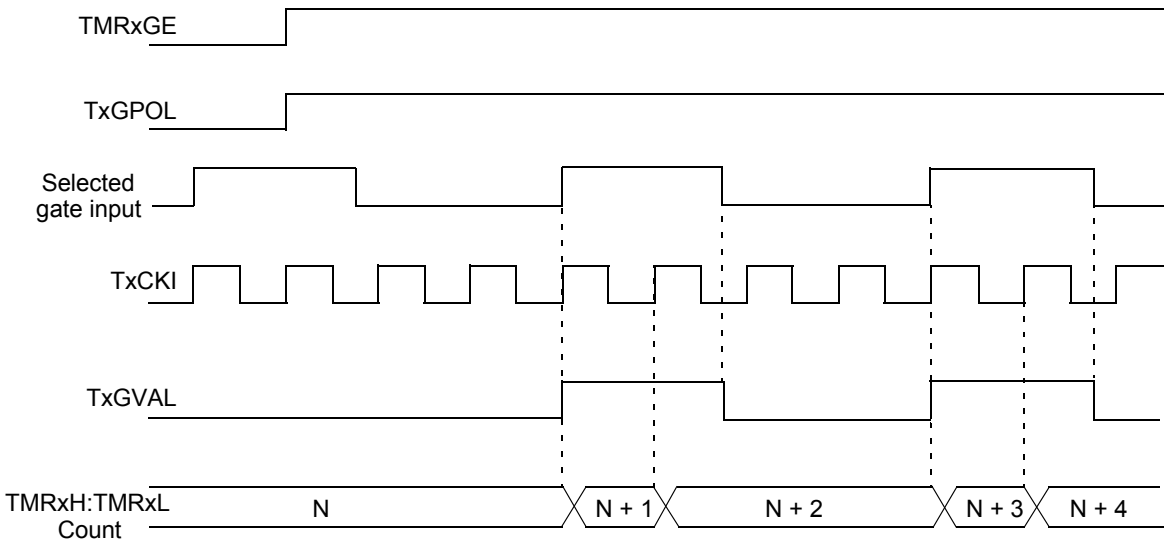
FIGURE 26-2: TIMER1 INCREMENTING EDGE



Note 1: Arrows indicate counter increments.

2: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge of the clock.

FIGURE 26-3: TIMER1 GATE ENABLE MODE



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 TxCON register are cleared).
- The diagrams illustrate any clock except Fosc/4 and show clock-sync delays of at least two full cycles for both ON and Timer2_ers. When using Fosc/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

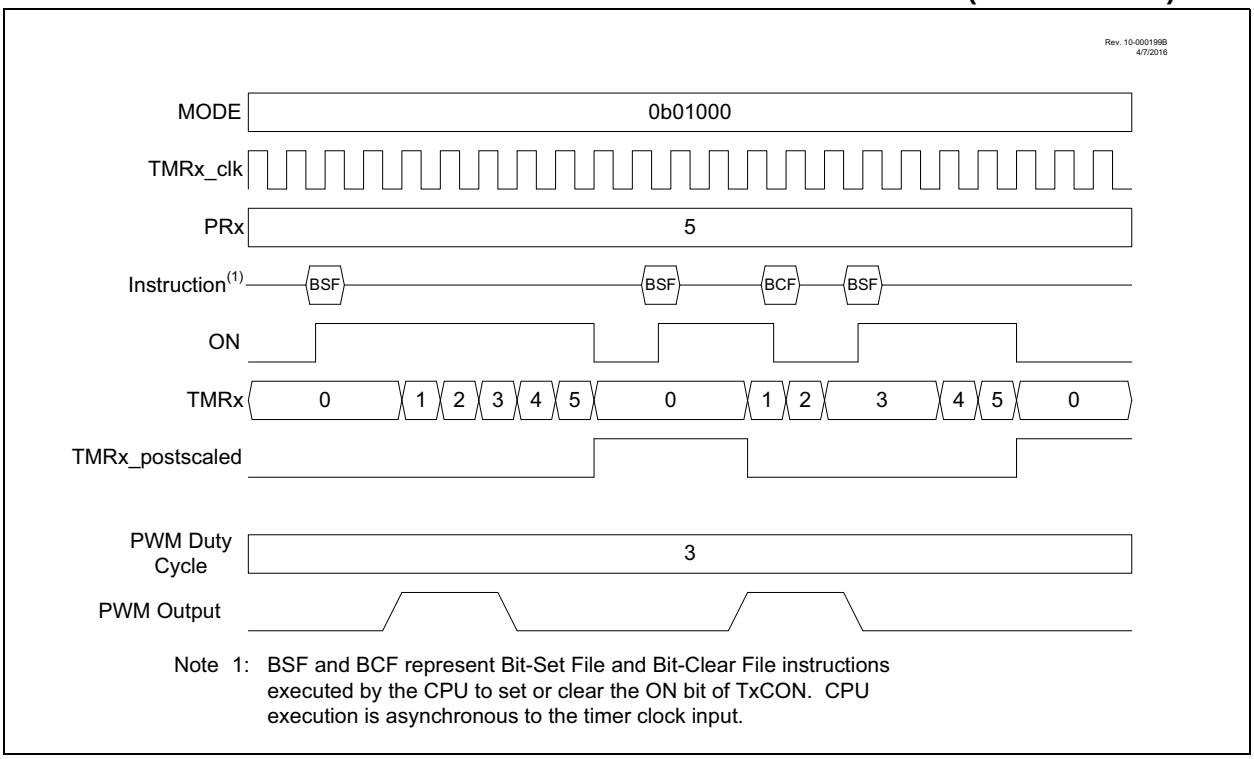
This mode corresponds to legacy Timer2 operation. The timer increments with each clock input when ON = 1 and does not increment when ON = 0. When the TMRx count equals the PRx period count the timer resets on the next clock and continues counting from 0. Operation with the ON bit software controlled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

27.5.5 SOFTWARE START ONE-SHOT MODE

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

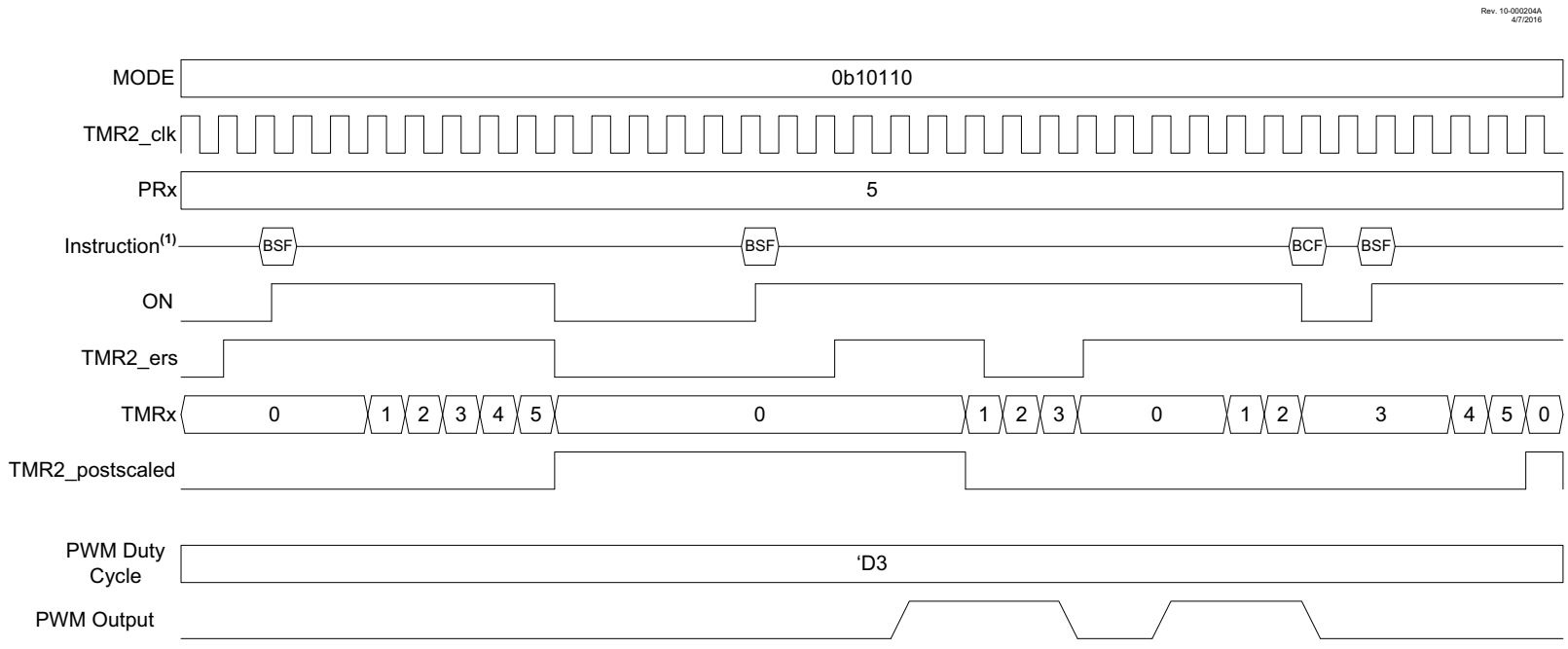
When One-Shot mode is used in conjunction with the CCP PWM operation the PWM pulse drive starts concurrent with setting the ON bit. Clearing the ON bit while the PWM drive is active will extend the PWM drive. The PWM drive will terminate when the timer value matches the CCPRx pulse width value. The PWM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

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



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

FIGURE 27-13: LEVEL-TRIGGERED HARDWARE LIMIT ONE-SHOT MODE TIMING DIAGRAM (MODE = 10110)



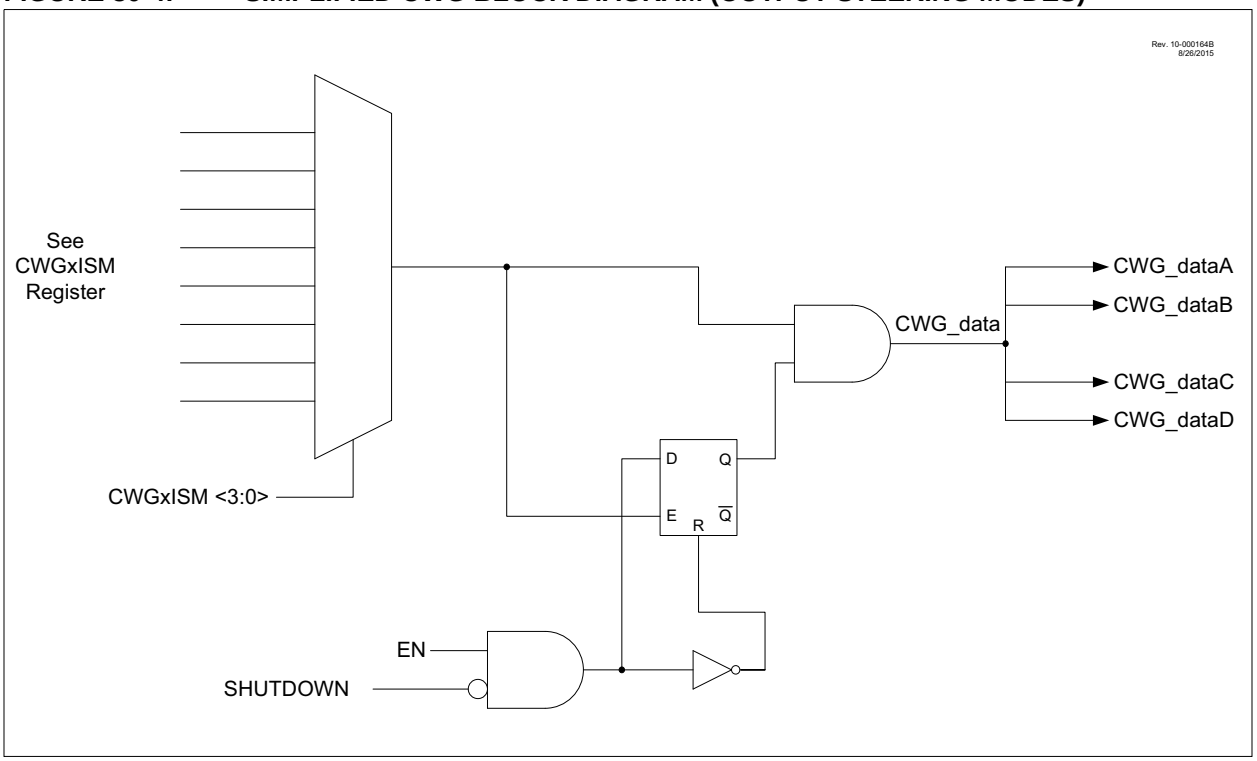
Note 1: BSF and BCF represent Bit-Set File and Bit-Clear File instructions executed by the CPU to set or clear the ON bit of TxCON. CPU execution is asynchronous to the timer clock input.

30.1.4 STEERING MODES

In Steering modes, the data input can be steered to any or all of the four CWG output pins. In Synchronous Steering mode, changes to steering selection registers take effect on the next rising input.

In Non-Synchronous mode, steering takes effect on the next instruction cycle. Additional details are provided in **Section 30.9 "CWG Steering Mode"**.

FIGURE 30-4: SIMPLIFIED CWG BLOCK DIAGRAM (OUTPUT STEERING MODES)



30.2 Clock Source

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

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

The clock sources are selected using the CS bit of the CWG1CLKCON register.

33.1.2.8 Asynchronous Reception Setup:

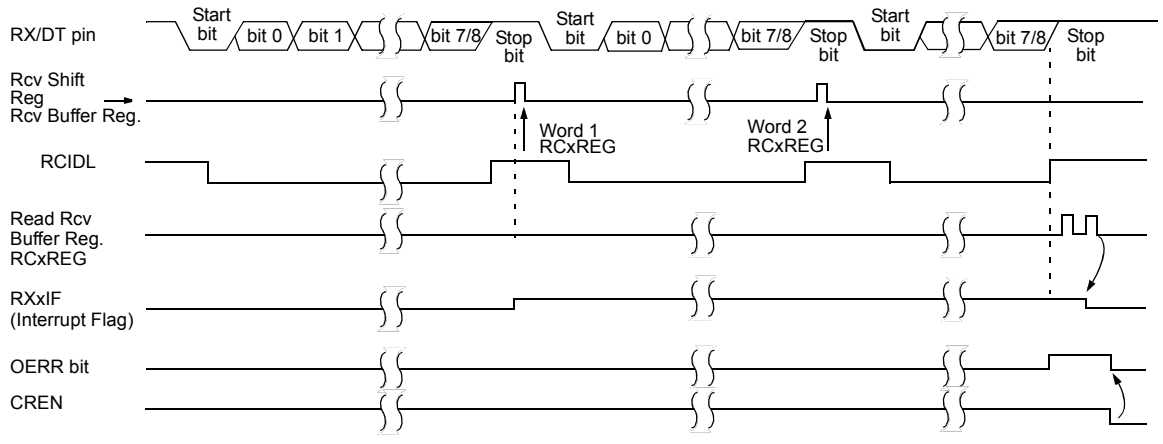
1. Initialize the SPxBRGH, SPxBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 33.3 "EUSART Baud Rate Generator (BRG)"**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RXxIE bit of the PIE3 register and the GIE and PEIE bits of the INTCON register.
5. If 9-bit reception is desired, set the RX9 bit.
6. Enable reception by setting the CREN bit.
7. The RXxIF interrupt flag bit will be set when a character is transferred from the RSR to the receive buffer. An interrupt will be generated if the RXxIE interrupt enable bit was also set.
8. Read the RCxSTA register to get the error flags and, if 9-bit data reception is enabled, the ninth data bit.
9. Get the received eight Least Significant data bits from the receive buffer by reading the RCxREG register.
10. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.

33.1.2.9 9-bit Address Detection Mode Setup

This mode would typically be used in RS-485 systems. To set up an Asynchronous Reception with Address Detect Enable:

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. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RXxIE bit of the PIE3 register and the GIE and PEIE bits of the INTCON register.
5. Enable 9-bit reception by setting the RX9 bit.
6. Enable address detection by setting the ADDEN bit.
7. Enable reception by setting the CREN bit.
8. The RXxIF interrupt flag bit will be set when a character with the ninth bit set is transferred from the RSR to the receive buffer. An interrupt will be generated if the RXxIE interrupt enable bit was also set.
9. Read the RCxSTA register to get the error flags. The ninth data bit will always be set.
10. Get the received eight Least Significant data bits from the receive buffer by reading the RCxREG register. Software determines if this is the device's address.
11. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.
12. If the device has been addressed, clear the ADDEN bit to allow all received data into the receive buffer and generate interrupts.

FIGURE 33-5: ASYNCHRONOUS RECEPTION



Note: This timing diagram shows three words appearing on the RX input. The RCxREG (receive buffer) is read after the third word, causing the OERR (overrun) bit to be set.

33.4.2 SYNCHRONOUS SLAVE MODE

The following bits are used to configure the EUSART for synchronous slave operation:

- SYNC = 1
- CSRC = 0
- SREN = 0 (for transmit); SREN = 1 (for receive)
- CREN = 0 (for transmit); CREN = 1 (for receive)
- SPEN = 1

Setting the SYNC bit of the TXxSTA register configures the device for synchronous operation. Clearing the CSRC bit of the TXxSTA register configures the device as a slave. Clearing the SREN and CREN bits of the RCxSTA register ensures that the device is in the Transmit mode, otherwise the device will be configured to receive. Setting the SPEN bit of the RCxSTA register enables the EUSART.

33.4.2.1 EUSART Synchronous Slave Transmit

The operation of the Synchronous Master and Slave modes are identical (see **Section 33.4.1.3 "Synchronous Master Transmission")**, except in the case of the Sleep mode.

If two words are written to the TXxREG and then the SLEEP instruction is executed, the following will occur:

1. The first character will immediately transfer to the TSR register and transmit.
2. The second word will remain in the TXxREG register.
3. The TXxIF bit will not be set.
4. After the first character has been shifted out of TSR, the TXxREG register will transfer the second character to the TSR and the TXxIF bit will now be set.
5. If the PEIE and TXxIE bits are set, the interrupt will wake the device from Sleep and execute the next instruction. If the GIE bit is also set, the program will call the Interrupt Service Routine.

33.4.2.2 Synchronous Slave Transmission Set-up:

1. Set the SYNC and SPEN bits and clear the CSRC bit.
2. Clear the ANSEL bit for the CK pin (if applicable).
3. Clear the CREN and SREN bits.
4. If interrupts are desired, set the TXxIE bit of the PIE3 register and the GIE and PEIE bits of the INTCON register.
5. If 9-bit transmission is desired, set the TX9 bit.
6. Enable transmission by setting the TXEN bit.
7. If 9-bit transmission is selected, insert the Most Significant bit into the TX9D bit.
8. Start transmission by writing the Least Significant eight bits to the TXxREG register.

37.2 Standard Operating Conditions

The standard operating conditions for any device are defined as:

Operating Voltage: $V_{DDMIN} \le V_{DD} \le V_{DDMAX}$
Operating Temperature: $T_{A\_MIN} \le T_A \le T_{A\_MAX}$

V_{DD} — Operating Supply Voltage(1)

PIC16LF15356/75/76/85/86

- V_{DDMIN} (Fosc ≤ 16 MHz) ..... +1.8V
- V_{DDMIN} (Fosc ≤ 32 MHz) ..... +2.5V
- V_{DDMAX} ..... +3.6V

PIC16F15356/75/76/85/86

- V_{DDMIN} (Fosc ≤ 16 MHz) ..... +2.3V
- V_{DDMIN} (Fosc ≤ 32 MHz) ..... +2.5V
- V_{DDMAX} ..... +5.5V

T_A — Operating Ambient Temperature Range

Industrial Temperature

- T_{A_MIN} ..... -40°C
- T_{A_MAX} ..... +85°C

Extended Temperature

- T_{A_MIN} ..... -40°C
- T_{A_MAX} ..... +125°C

Note 1: See Parameter Supply Voltage, DS Characteristics: Supply Voltage.

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®