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, LCD, POR, PWM, WDT
Number of I/O	25
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 11x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SSOP (0.209", 5.30mm Width)
Supplier Device Package	28-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f1938-i-ss

TABLE 3-10: SPECIAL FUNCTION REGISTER SUMMARY (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 all other Resets
Bank 2											
100h(2)	INDF0	Addressing this location uses contents of FSR0H/FSR0L to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx
101h(2)	INDF1	Addressing this location uses contents of FSR1H/FSR1L to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx
102h(2)	PCL	Program Counter (PC) Least Significant Byte								0000 0000	0000 0000
103h(2)	STATUS	—	—	—	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu
104h(2)	FSR0L	Indirect Data Memory Address 0 Low Pointer								0000 0000	uuuu uuuu
105h(2)	FSR0H	Indirect Data Memory Address 0 High Pointer								0000 0000	0000 0000
106h(2)	FSR1L	Indirect Data Memory Address 1 Low Pointer								0000 0000	uuuu uuuu
107h(2)	FSR1H	Indirect Data Memory Address 1 High Pointer								0000 0000	0000 0000
108h(2)	BSR	—	—	—	BSR<4:0>					---0 0000	---0 0000
109h(2)	WREG	Working Register								0000 0000	uuuu uuuu
10Ah(1, 2)	PCLATH	—	Write Buffer for the upper 7 bits of the Program Counter							-000 0000	-000 0000
10Bh(2)	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000
10Ch	LATA	PORTA Data Latch								xxxx xxxx	uuuu uuuu
10Dh	LATB	PORTB Data Latch								xxxx xxxx	uuuu uuuu
10Eh	LATC	PORTC Data Latch								xxxx xxxx	uuuu uuuu
10Fh(3)	LATD	PORTD Data Latch								xxxx xxxx	uuuu uuuu
110h	LATE	—	—	—	—	—	LATE2(3)	LATE1(3)	LATE0(3)	---- -xxx	---- -uuu
111h	CM1CON0	C1ON	C1OUT	C1OE	C1POL	—	C1SP	C1HYS	C1SYNC	0000 -100	0000 -100
112h	CM1CON1	C1INTP	C1INTN	C1PCH1	C1PCH0	—	—	C1NCH<1:0>		0000 --00	0000 --00
113h	CM2CON0	C2ON	C2OUT	C2OE	C2POL	—	C2SP	C2HYS	C2SYNC	0000 -100	0000 -100
114h	CM2CON1	C2INTP	C2INTN	C2PCH1	C2PCH0	—	—	C2NCH<1:0>		0000 --00	0000 --00
115h	CMOUT	—	—	—	—	—	—	MC2OUT	MC1OUT	---- --00	---- --00
116h	BORCON	SBOREN	—	—	—	—	—	—	BORRDY	1--- ---q	u--- ---u
117h	FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR1	CDAFVR0	ADFVR<1:0>		0q00 0000	0q00 0000
118h	DACCON0	DACEN	DACLPS	DACOE	---	DACPSS<1:0>		---	DACNSS	000- 00-0	000- 00-0
119h	DACCON1	---	---	---	DACR<4:0>					---0 0000	---0 0000
11Ah	SRCON0	SRLEN	SRCLK2	SRCLK1	SRCLK0	SRQEN	SRNQEN	SRPS	SRPR	0000 0000	0000 0000
11Bh	SRCON1	SRSPE	SRSCKE	SRSC2E	SRSC1E	SRRPE	SRRCKE	SRRC2E	SRRC1E	0000 0000	0000 0000
11Ch	—	Unimplemented								—	—
11Dh	APFCON	—	CCP3SEL	T1GSEL	P2BSEL	SRNQSEL	C2OUTSEL	SSSEL	CCP2SEL	-000 0000	-000 0000
11Eh	—	Unimplemented								—	—
11Fh	—	Unimplemented								—	—

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

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<14:8>, whose contents are transferred to the upper byte of the program counter.

2: These registers can be addressed from any bank.

3: These registers/bits are not implemented on PIC16(L)F1938 devices, read as '0'.

4: Unimplemented, read as '1'.

TABLE 3-10: SPECIAL FUNCTION REGISTER SUMMARY (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 all other Resets
Bank 31											
F80h(2)	INDF0	Addressing this location uses contents of FSR0H/FSR0L to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx
F81h(2)	INDF1	Addressing this location uses contents of FSR1H/FSR1L to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx
F82h(2)	PCL	Program Counter (PC) Least Significant Byte								0000 0000	0000 0000
F83h(2)	STATUS	—	—	—	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu
F84h(2)	FSR0L	Indirect Data Memory Address 0 Low Pointer								0000 0000	uuuu uuuu
F85h(2)	FSR0H	Indirect Data Memory Address 0 High Pointer								0000 0000	0000 0000
F86h(2)	FSR1L	Indirect Data Memory Address 1 Low Pointer								0000 0000	uuuu uuuu
F87h(2)	FSR1H	Indirect Data Memory Address 1 High Pointer								0000 0000	0000 0000
F88h(2)	BSR	—	—	—	BSR<4:0>					---0 0000	---0 0000
F89h(2)	WREG	Working Register								0000 0000	uuuu uuuu
F8Ah(1),(2)	PCLATH	—	Write Buffer for the upper 7 bits of the Program Counter							-000 0000	-000 0000
F8Bh(2)	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000
F8Ch — FE3h	—	Unimplemented								—	—
FE4h	STATUS_ SHAD						Z_SHAD	DC_SHAD	C_SHAD	---- -xxx	---- -uuu
FE5h	WREG_ SHAD	Working Register Normal (Non-ICD) Shadow								xxxx xxxx	uuuu uuuu
FE6h	BSR_ SHAD				Bank Select Register Normal (Non-ICD) Shadow					---x xxxx	---u uuuu
FE7h	PCLATH_ SHAD		Program Counter Latch High Register Normal (Non-ICD) Shadow							-xxx xxxx	uuuu uuuu
FE8h	FSR0L_ SHAD	Indirect Data Memory Address 0 Low Pointer Normal (Non-ICD) Shadow								xxxx xxxx	uuuu uuuu
FE9h	FSR0H_ SHAD	Indirect Data Memory Address 0 High Pointer Normal (Non-ICD) Shadow								xxxx xxxx	uuuu uuuu
FEAh	FSR1L_ SHAD	Indirect Data Memory Address 1 Low Pointer Normal (Non-ICD) Shadow								xxxx xxxx	uuuu uuuu
FEBh	FSR1H_ SHAD	Indirect Data Memory Address 1 High Pointer Normal (Non-ICD) Shadow								xxxx xxxx	uuuu uuuu
FECh	—	Unimplemented								—	—
FEDh	STKPTR	—	—	—	Current Stack pointer					---1 1111	---1 1111
FEEh	TOSL	Top of Stack Low byte								xxxx xxxx	uuuu uuuu
FEFh	TOSH	—	Top of Stack High byte							-xxx xxxx	-uuu uuuu

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

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<14:8>, whose contents are transferred to the upper byte of the program counter.

2: These registers can be addressed from any bank.

3: These registers/bits are not implemented on PIC16(L)F1938 devices, read as '0'.

4: Unimplemented, read as '1'.

10.6 Register Definitions: Watchdog Control

REGISTER 10-1: WDTCON: WATCHDOG TIMER CONTROL REGISTER

U-0	U-0	R/W-0/0	R/W-1/1	R/W-0/0	R/W-1/1	R/W-1/1	R/W-0/0
—	—	WDTPS<4:0>					SWDTEN
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	-m/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-6 **Unimplemented:** Read as '0'

bit 5-1 **WDTPS<4:0>:** Watchdog Timer Period Select bits

Bit Value = Prescale Rate

00000 = 1:32 (Interval 1 ms typ)
00001 = 1:64 (Interval 2 ms typ)
00010 = 1:128 (Interval 4 ms typ)
00011 = 1:256 (Interval 8 ms typ)
00100 = 1:512 (Interval 16 ms typ)
00101 = 1:1024 (Interval 32 ms typ)
00110 = 1:2048 (Interval 64 ms typ)
00111 = 1:4096 (Interval 128 ms typ)
01000 = 1:8192 (Interval 256 ms typ)
01001 = 1:16384 (Interval 512 ms typ)
01010 = 1:32768 (Interval 1s typ)
01011 = 1:65536 (Interval 2s typ) (Reset value)
01100 = 1:131072 (2^{17}) (Interval 4s typ)
01101 = 1:262144 (2^{18}) (Interval 8s typ)
01110 = 1:524288 (2^{19}) (Interval 16s typ)
01111 = 1:1048576 (2^{20}) (Interval 32s typ)
10000 = 1:2097152 (2^{21}) (Interval 64s typ)
10001 = 1:4194304 (2^{22}) (Interval 128s typ)
10010 = 1:8388608 (2^{23}) (Interval 256s typ)

10011 = Reserved. Results in minimum interval (1:32)
•
•
•
11111 = Reserved. Results in minimum interval (1:32)

bit 0 **SWDTEN:** Software Enable/Disable for Watchdog Timer bit

If WDTE<1:0> = 00:
This bit is ignored.
If WDTE<1:0> = 01:
1 = WDT is turned on
0 = WDT is turned off
If WDTE<1:0> = 1x:
This bit is ignored.

12.11 PORTE Registers

PORTE is a 4-bit wide, bidirectional port. The corresponding data direction register is TRISE. Setting a TRISE bit (= 1) will make the corresponding PORTE pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISE bit (= 0) will make the corresponding PORTE pin an output (i.e., enable the output driver and put the contents of the output latch on the selected pin). The exception is RE3, which is input only and its TRIS bit will always read as '1'. Example 12-1 shows how to initialize an I/O port.

Reading the PORTE register (Register 12-18) reads the status of the pins, whereas writing to it will write to the PORT latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified and then written to the PORT data latch (LATE). RE3 reads '0' when MCLRE = 1.

Note:	RE<2:0> and TRISE<2:0> pins are available on PIC16(L)F1939 only.

12.11.1 ANSELE REGISTER

The ANSELE register (Register 12-21) is used to configure the Input mode of an I/O pin to analog. Setting the appropriate ANSELE bit high will cause all digital reads on the pin to be read as '0' and allow analog functions on the pin to operate correctly.

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

The TRISE register (Register 12-19) controls the PORTE pin output drivers, even when they are being used as analog inputs. The user should ensure the bits in the TRISE register are maintained set when using them as analog inputs. I/O pins configured as analog input always read '0'.

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

12.11.2 PORTE FUNCTIONS AND OUTPUT PRIORITIES

Each PORTE pin is multiplexed with other functions. The pins, their combined functions and their output priorities are shown in Table 12-12.

When multiple outputs are enabled, the actual pin control goes to the peripheral with the lowest number in the following lists.

Analog input and some digital input functions are not included in the list below. These input functions can remain active when the pin is configured as an output. Certain digital input functions override other port functions and are included in priority list.

TABLE 12-12: PORTE OUTPUT PRIORITY

Pin Name	Function Priority(1)
RE0	SEG21 (LCD) CCP3/P3A (CCP) RE0
RE1	SEG22 (LCD) P3B (CCP) RE1
RE2	SEG23 (LCD) CCP5 (CCP) RE2

Note 1: Priority listed from highest to lowest.

REGISTER 12-20: LATE: PORTE DATA LATCH REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u
—	—	—	—	—	LATE2(2)	LATE1(2)	LATE0(2)
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 7-4 **Unimplemented**: Read as '0'

bit 3-0 **LATE<3:0>**: PORTE Output Latch Value bits(1)

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

2: LATE register is not implemented on the PIC16(L)F1938. Read as '0'

REGISTER 12-21: ANSELE: PORTE ANALOG SELECT REGISTER

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1
—	—	—	—	—	ANSE2(2)	ANSE1(2)	ANSE0(2)
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 7-3 **Unimplemented:** Read as '0'

bit 2-0 **ANSE<2:0>**: Analog Select between Analog or Digital Function on Pins RE<2:0>, respectively

0 = Digital I/O. Pin is assigned to port or digital special function.

1 = Analog input. Pin is assigned as analog input(1). Digital input buffer disabled.

Note 1: When setting a pin to an analog input, the corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

2: ANSELE register is not implemented on the PIC16(L)F1938. Read as '0'

13.6 Register Definitions: Interrupt-On-Change

REGISTER 13-1: IOCBP: INTERRUPT-ON-CHANGE POSITIVE EDGE 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
IOCBP7	IOCBP6	IOCBP5	IOCBP4	IOCBP3	IOCBP2	IOCBP1	IOCBP0
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 **IOCBP<7:0>:** Interrupt-on-Change Positive Edge Enable bits

1 = Interrupt-on-Change enabled on the pin for a positive going edge. IOCBFx bit and IOCIF flag will be set upon detecting an edge.

0 = Interrupt-on-Change disabled for the associated pin.

REGISTER 13-2: IOCBN: INTERRUPT-ON-CHANGE NEGATIVE EDGE 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
IOCBN7	IOCBN6	IOCBN5	IOCBN4	IOCBN3	IOCBN2	IOCBN1	IOCBN0
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 **IOCBN<7:0>:** Interrupt-on-Change Negative Edge Enable bits

1 = Interrupt-on-Change enabled on the pin for a negative going edge. IOCBFx bit and IOCIF will be set upon detecting an edge.

0 = Interrupt-on-Change disabled for the associated pin.

REGISTER 13-3: IOCBF: INTERRUPT-ON-CHANGE FLAG REGISTER

R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0
IOCBF7	IOCBF6	IOCBF5	IOCBF4	IOCBF3	IOCBF2	IOCBF1	IOCBF0
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 - Bit is set in hardware

bit 7-0 **IOCBF<7:0>:** Interrupt-on-Change Flag bits

1 = An enabled change was detected on the associated pin.
Set when IOCBPx = 1 and a rising edge was detected on RBx, or when IOCBNx = 1 and a falling edge was detected on RBx.

0 = No change was detected, or the user cleared the detected change.

15.3 Register Definitions: ADC Control

REGISTER 15-1: ADCON0: A/D CONTROL REGISTER 0

U-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
—	CHS<4:0>					GO/$\overline{\text{DONE}}$	ADON
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 **Unimplemented:** Read as '0'

bit 6-2 **CHS<4:0>:** Analog Channel Select bits

11111 = FVR (Fixed Voltage Reference) Buffer 1 Output(2)
11110 = DAC output(1)
11101 = Temperature Indicator(3)
11100 = Reserved. No channel connected.
•
•
•
01110 = Reserved. No channel connected.
01101 = AN13
01100 = AN12
01011 = AN11
01010 = AN10
01001 = AN9
01000 = AN8
00111 = AN7(4)
00110 = AN6(4)
00101 = AN5(4)
00100 = AN4
00011 = AN3
00010 = AN2
00001 = AN1
00000 = AN0

bit 1 **GO/$\overline{\text{DONE}}$:** A/D Conversion Status bit

1 = A/D conversion cycle in progress. Setting this bit starts an A/D conversion cycle.
This bit is automatically cleared by hardware when the A/D conversion has completed.
0 = A/D conversion completed/not in progress

bit 0 **ADON:** ADC Enable bit

1 = ADC is enabled
0 = ADC is disabled and consumes no operating current

Note 1: See **Section 17.0 "Digital-to-Analog Converter (DAC) Module"** for more information.
2: See **Section 14.0 "Fixed Voltage Reference (FVR)"** for more information.
3: See **Section 16.0 "Temperature Indicator Module"** for more information.
4: Not available on the PIC16(L)F1933/1936/1938.

REGISTER 19-2: SRCON1: SR LATCH CONTROL 1 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
SRSPE	SRSCKE	SRSC2E	SRSC1E	SRRPE	SRRCKE	SRRC2E	SRRC1E
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 **SRSPE:** SR Latch Peripheral Set Enable bit
1 = SR Latch is set when the SRI pin is high.
0 = SRI pin has no effect on the set input of the SR Latch

bit 6 **SRSCKE:** SR Latch Set Clock Enable bit
1 = Set input of SR Latch is pulsed with SRCLK
0 = SRCLK has no effect on the set input of the SR Latch

bit 5 **SRSC2E:** SR Latch C2 Set Enable bit
1 = SR Latch is set when the C2 Comparator output is high
0 = C2 Comparator output has no effect on the set input of the SR Latch

bit 4 **SRSC1E:** SR Latch C1 Set Enable bit
1 = SR Latch is set when the C1 Comparator output is high
0 = C1 Comparator output has no effect on the set input of the SR Latch

bit 3 **SRRPE:** SR Latch Peripheral Reset Enable bit
1 = SR Latch is reset when the SRI pin is high.
0 = SRI pin has no effect on the reset input of the SR Latch

bit 2 **SRRCKE:** SR Latch Reset Clock Enable bit
1 = Reset input of SR Latch is pulsed with SRCLK
0 = SRCLK has no effect on the reset input of the SR Latch

bit 1 **SRRC2E:** SR Latch C2 Reset Enable bit
1 = SR Latch is reset when the C2 Comparator output is high
0 = C2 Comparator output has no effect on the reset input of the SR Latch

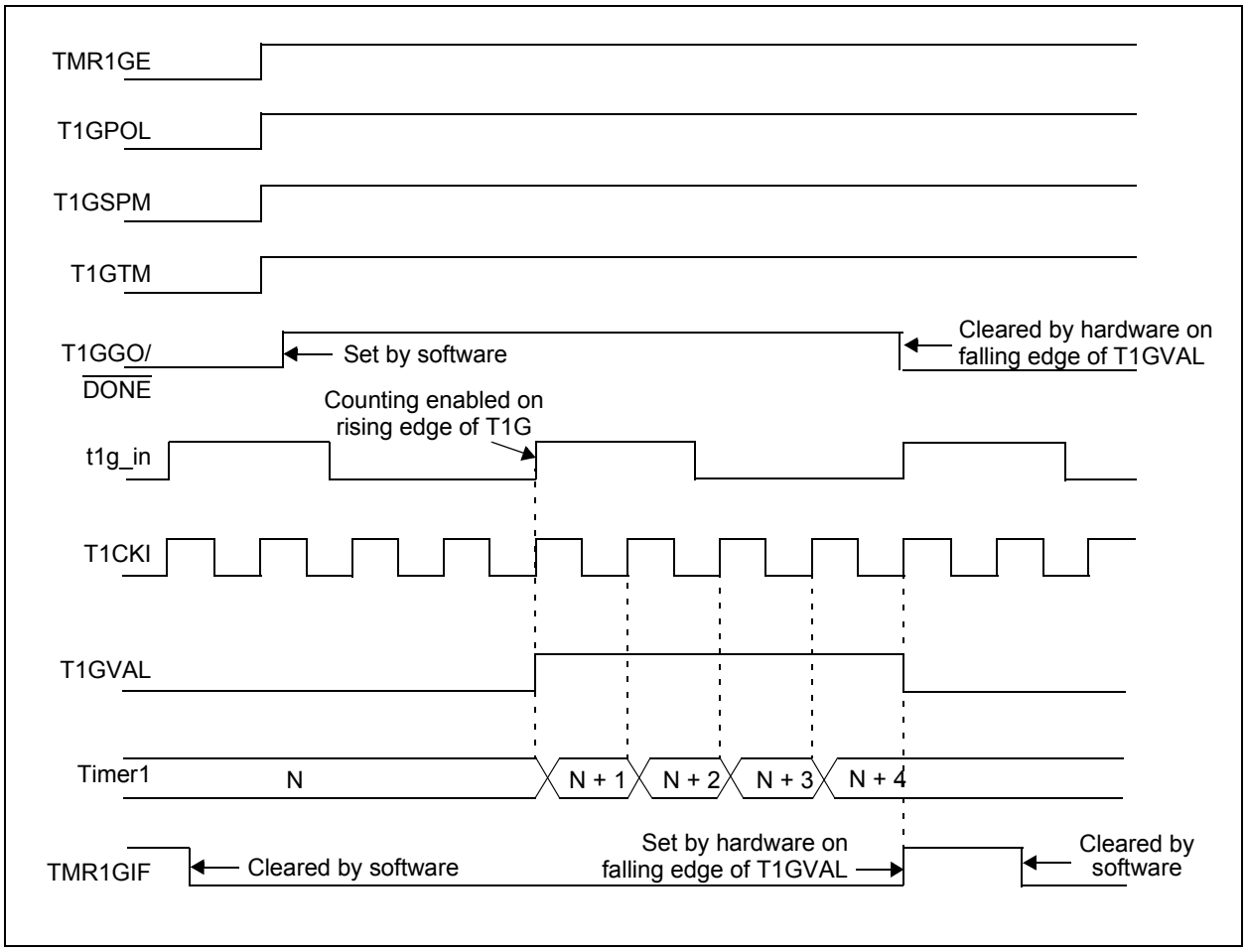
bit 0 **SRRC1E:** SR Latch C1 Reset Enable bit
1 = SR Latch is reset when the C1 Comparator output is high
0 = C1 Comparator output has no effect on the reset input of the SR Latch

TABLE 19-2: SUMMARY OF REGISTERS ASSOCIATED WITH SR LATCH MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	—	—	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	126
SRCON0	SRLEN	SRCLK<2:0>			SRQEN	SRNQEN	SRPS	SRPR	182
SRCON1	SRSPE	SRSCKE	SRSC2E	SRSC1E	SRRPE	SRRCKE	SRRC2E	SRRC1E	183
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	125

Legend: — = unimplemented location, read as '0'. Shaded cells are unused by the SR Latch module.

FIGURE 21-6: TIMER1 GATE SINGLE-PULSE AND TOGGLE COMBINED MODE



22.0 TIMER2/4/6 MODULES

There are up to three identical Timer2-type modules available. To maintain pre-existing naming conventions, the Timers are called Timer2, Timer4 and Timer6 (also Timer2/4/6).

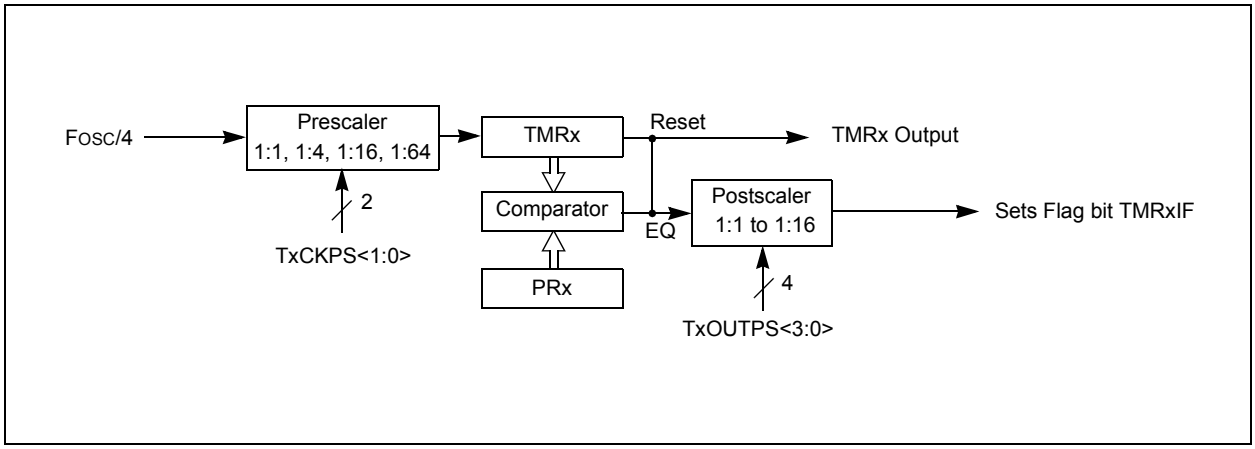
Note:	The 'x' variable used in this section is used to designate Timer2, Timer4, or Timer6. For example, TxCON references T2CON, T4CON, or T6CON. PRx references PR2, PR4, or PR6.

The Timer2/4/6 modules incorporate the following features:

- 8-bit Timer and Period registers (TMRx and PRx, respectively)
- Readable and writable (both registers)
- Software programmable prescaler (1:1, 1:4, 1:16, and 1:64)
- Software programmable postscaler (1:1 to 1:16)
- Interrupt on TMRx match with PRx, respectively
- Optional use as the shift clock for the MSSP module (Timer2 only)

See Figure 22-1 for a block diagram of Timer2/4/6.

FIGURE 22-1: TIMER2/4/6 BLOCK DIAGRAM



24.5.6 CLOCK STRETCHING

Clock stretching occurs when a device on the bus holds the SCL line low effectively pausing communication. The slave may stretch the clock to allow more time to handle data or prepare a response for the master device. A master device is not concerned with stretching as anytime it is active on the bus and not transferring data it is stretching. Any stretching done by a slave is invisible to the master software and handled by the hardware that generates SCL.

The CKP bit of the SSPCON1 register is used to control stretching in software. Any time the CKP bit is cleared, the module will wait for the SCL line to go low and then hold it. Setting CKP will release SCL and allow more communication.

24.5.6.1 Normal Clock Stretching

Following an $\overline{ACK}$ if the R/$\overline{W}$ bit of SSPSTAT is set, a read request, the slave hardware will clear CKP. This allows the slave time to update SSPBUF with data to transfer to the master. If the SEN bit of SSPCON2 is set, the slave hardware will always stretch the clock after the $\overline{ACK}$ sequence. Once the slave is ready; CKP is set by software and communication resumes.

Note 1: The BF bit has no effect on if the clock will be stretched or not. This is different than previous versions of the module that would not stretch the clock, clear CKP, if SSPBUF was read before the ninth falling edge of SCL.

2: Previous versions of the module did not stretch the clock for a transmission if SSPBUF was loaded before the ninth falling edge of SCL. It is now always cleared for read requests.

24.5.6.2 10-bit Addressing Mode

In 10-bit Addressing mode, when the UA bit is set, the clock is always stretched. This is the only time the SCL is stretched without CKP being cleared. SCL is released immediately after a write to SSPADD.

Note: Previous versions of the module did not stretch the clock if the second address byte did not match.

24.5.6.3 Byte NACKing

When AHEN bit of SSPCON3 is set; CKP is cleared by hardware after the eighth falling edge of SCL for a received matching address byte. When DHEN bit of SSPCON3 is set; CKP is cleared after the eighth falling edge of SCL for received data.

Stretching after the eighth falling edge of SCL allows the slave to look at the received address or data and decide if it wants to ACK the received data.

24.5.7 CLOCK SYNCHRONIZATION AND THE CKP BIT

Any time the CKP bit is cleared, the module will wait for the SCL line to go low and then hold it. However, clearing the CKP bit will not assert the SCL output low until the SCL output is already sampled low. Therefore, the CKP bit will not assert the SCL line until an external I^2C master device has already asserted the SCL line. The SCL output will remain low until the CKP bit is set and all other devices on the I^2C bus have released SCL. This ensures that a write to the CKP bit will not violate the minimum high time requirement for SCL (see Figure 24-23).

FIGURE 24-23: CLOCK SYNCHRONIZATION TIMING

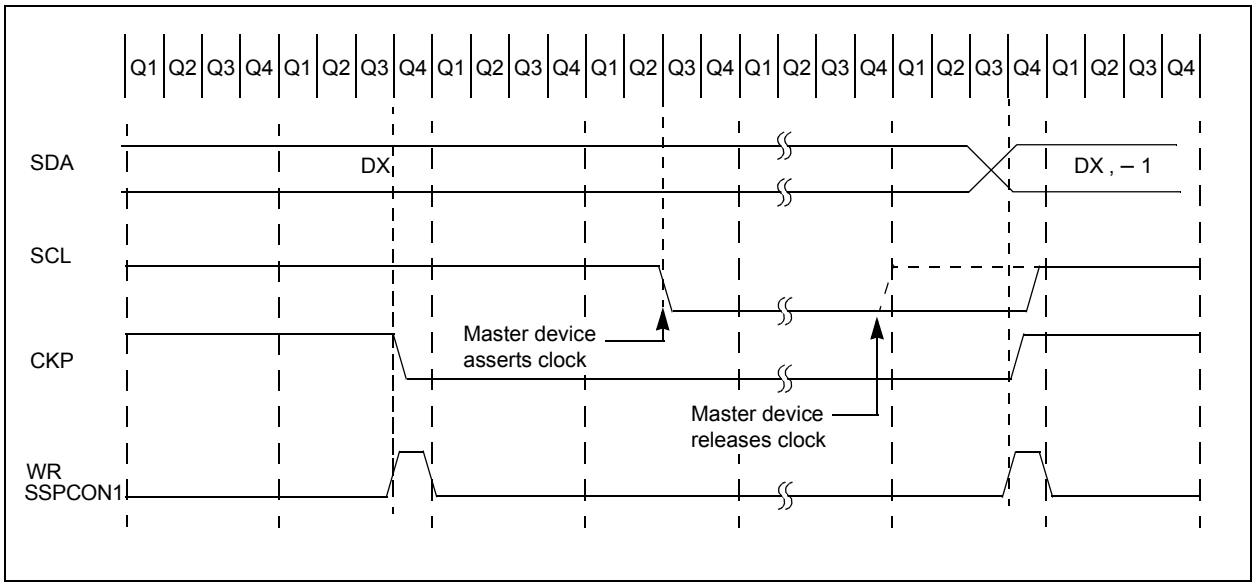
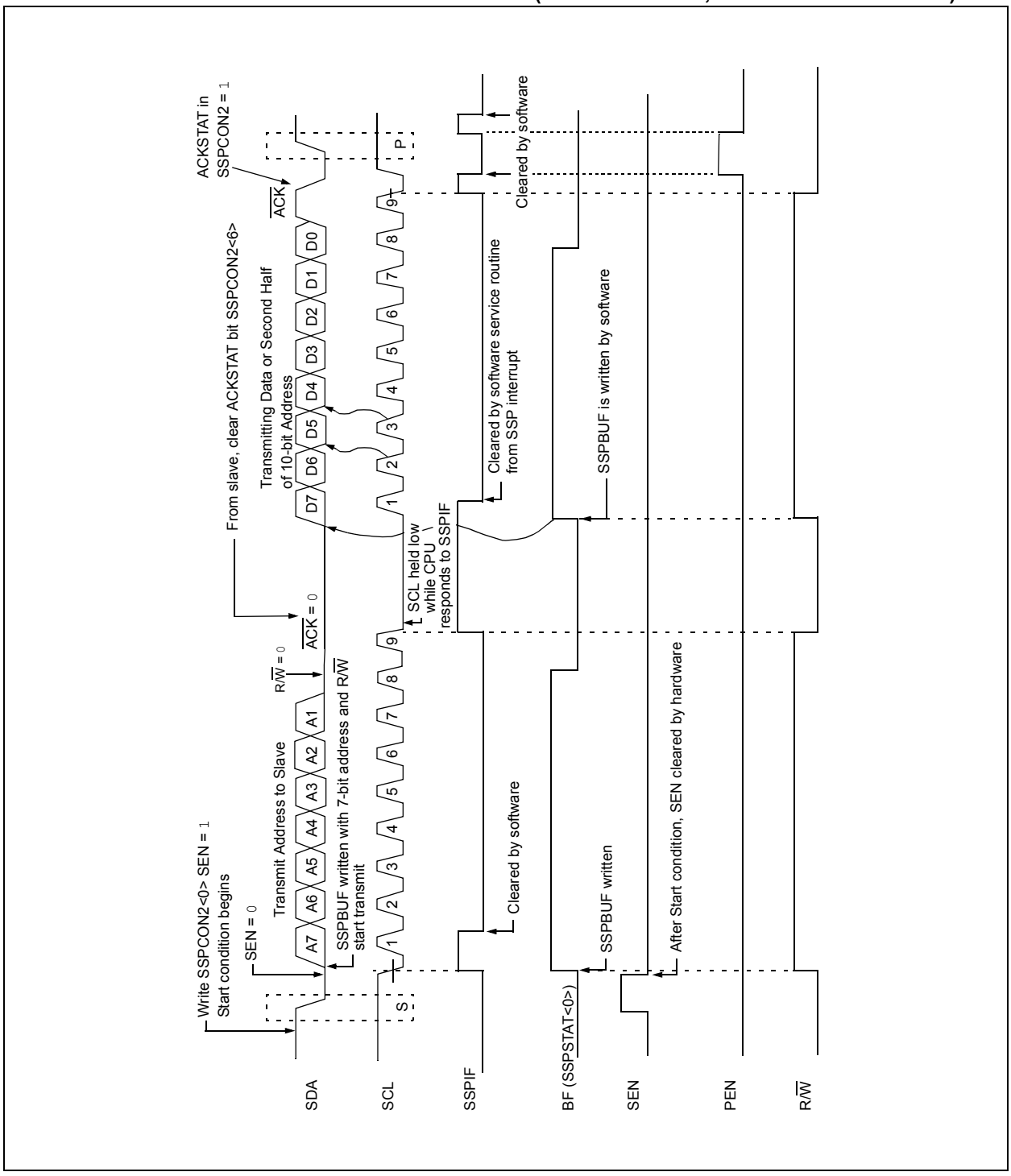


FIGURE 24-28: I²C MASTER MODE WAVEFORM (TRANSMISSION, 7 OR 10-BIT ADDRESS)



24.6.10 SLEEP OPERATION

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

24.6.11 EFFECTS OF A RESET

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

24.6.12 MULTI-MASTER MODE

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

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

The states where arbitration can be lost are:

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

24.6.13 MULTI -MASTER COMMUNICATION, BUS COLLISION AND BUS ARBITRATION

Multi-Master mode support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA, by letting SDA float high and another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin is '0', then a bus collision has taken place. The master will set the Bus Collision Interrupt Flag, BCLIF, and reset the I²C port to its Idle state (Figure 24-32).

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

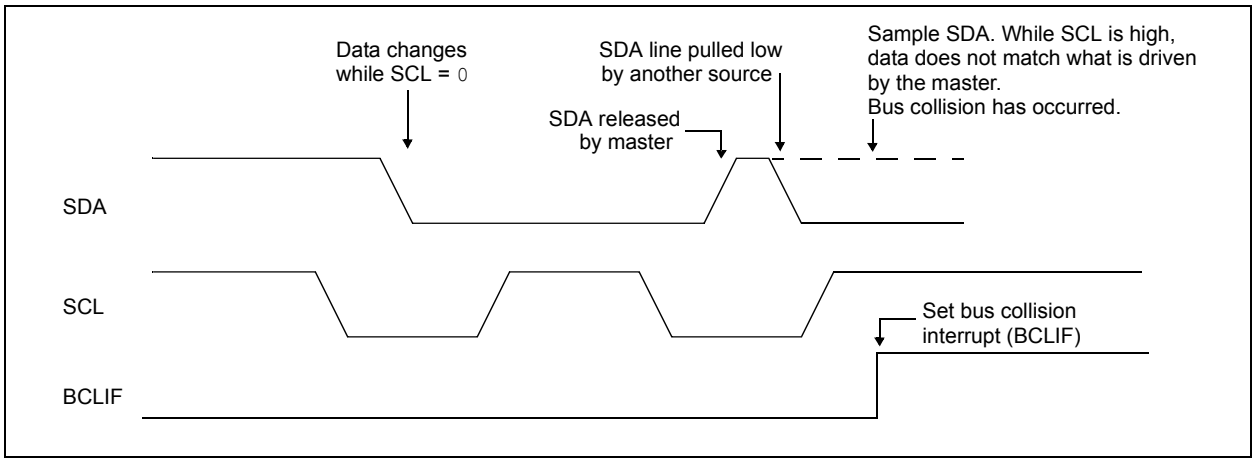
If a Start, Repeated Start, Stop or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are deasserted and the respective control bits in the SSPCON2 register are cleared. When the user services the bus collision Interrupt Service Routine and if the I²C bus is free, the user can resume communication by asserting a Start condition.

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

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

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

FIGURE 24-32: BUS COLLISION TIMING FOR TRANSMIT AND ACKNOWLEDGE



24.6.13.3 Bus Collision During a Stop Condition

Bus collision occurs during a Stop condition if:

a) After the SDA pin has been deasserted and allowed to float high, SDA is sampled low after the BRG has timed out.

b) After the SCL pin is deasserted, SCL is sampled low before SDA goes high.

The Stop condition begins with SDA asserted low. When SDA is sampled low, the SCL pin is allowed to float. When the pin is sampled high (clock arbitration), the Baud Rate Generator is loaded with SSPADD and counts down to zero. After the BRG times out, SDA is sampled. If SDA is sampled low, a bus collision has occurred. This is due to another master attempting to drive a data '0' (Figure 24-38). If the SCL pin is sampled low before SDA is allowed to float high, a bus collision occurs. This is another case of another master attempting to drive a data '0' (Figure 24-39).

FIGURE 24-38: BUS COLLISION DURING A STOP CONDITION (CASE 1)

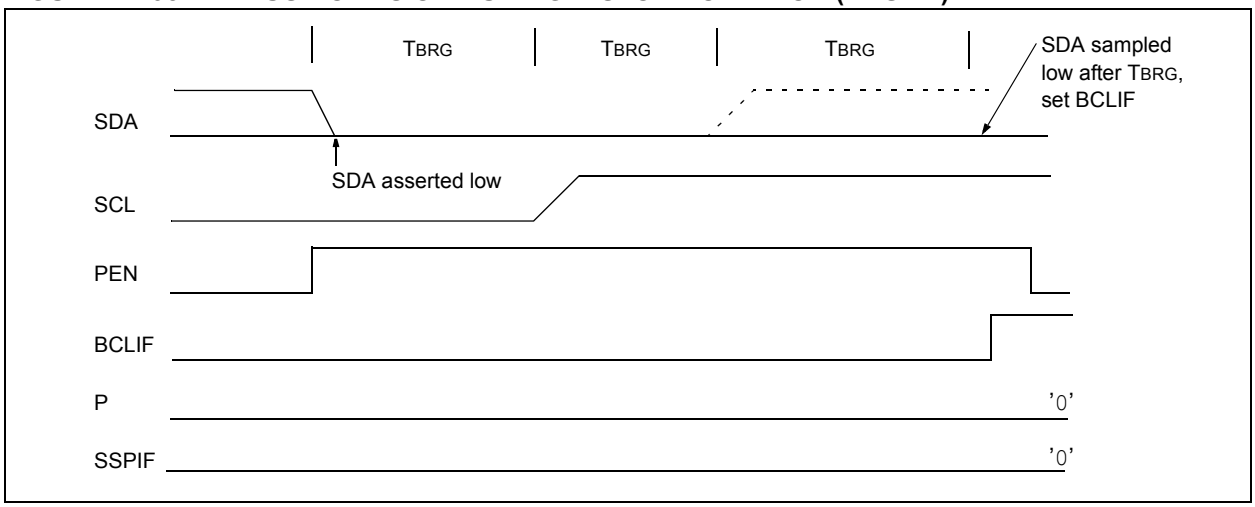
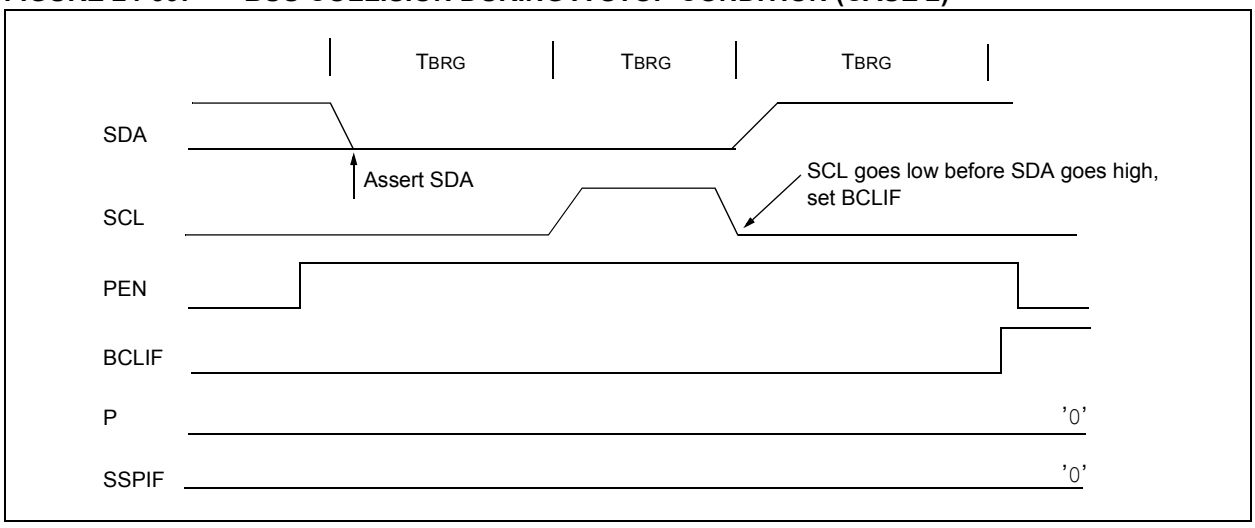


FIGURE 24-39: BUS COLLISION DURING A STOP CONDITION (CASE 2)



25.1.2.8 Asynchronous Reception Set-up:

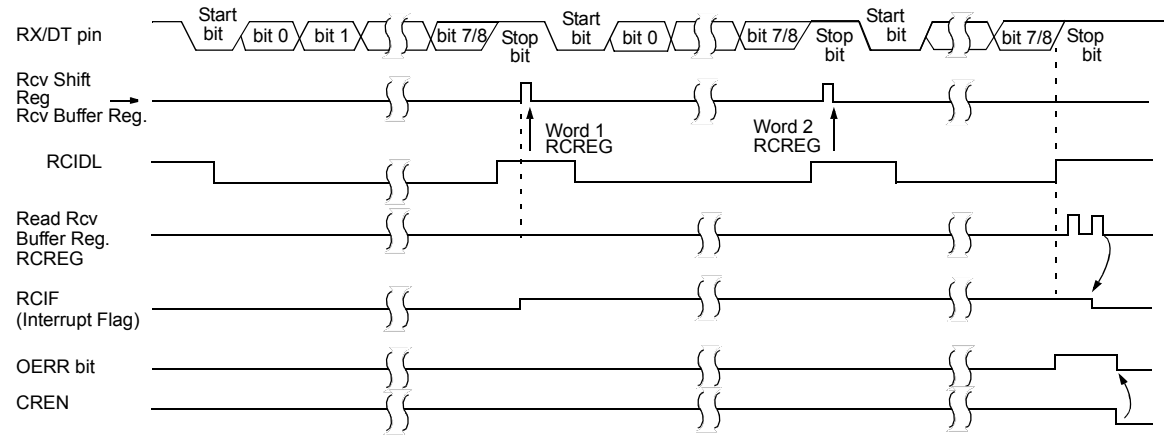
1. Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 25.4 "EUSART Baud Rate Generator (BRG)"**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RCIE bit of the PIE1 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 RCIF 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 RCIE interrupt enable bit was also set.
8. Read the RCSTA 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 RCREG register.
10. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.

25.1.2.9 9-bit Address Detection Mode Set-up

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

1. Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 25.4 "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 RCIE bit of the PIE1 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 RCIF 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 RCIE interrupt enable bit was also set.
9. Read the RCSTA 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 RCREG 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 25-5: ASYNCHRONOUS RECEPTION



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

25.5.2.3 EUSART Synchronous Slave Reception

The operation of the Synchronous Master and Slave modes is identical (**Section 25.5.1.5 "Synchronous Master Reception"**), with the following exceptions:

- Sleep
- CREN bit is always set, therefore the receiver is never Idle
- SREN bit, which is a "don't care" in Slave mode

A character may be received while in Sleep mode by setting the CREN bit prior to entering Sleep. Once the word is received, the RSR register will transfer the data to the RCREG register. If the RCIE enable bit is set, the interrupt generated will wake the device from Sleep and execute the next instruction. If the GIE bit is also set, the program will branch to the interrupt vector.

25.5.2.4 Synchronous Slave Reception Set-up:

1. Set the SYNC and SPEN bits and clear the CSRC bit.
2. Clear the ANSEL bit for both the CK and DT pins (if applicable).
3. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
4. If 9-bit reception is desired, set the RX9 bit.
5. Set the CREN bit to enable reception.
6. The RCIF bit will be set when reception is complete. An interrupt will be generated if the RCIE bit was set.
7. If 9-bit mode is enabled, retrieve the Most Significant bit from the RX9D bit of the RCSTA register.
8. Retrieve the eight Least Significant bits from the receive FIFO by reading the RCREG register.
9. If an overrun error occurs, clear the error by either clearing the CREN bit of the RCSTA register or by clearing the SPEN bit which resets the EUSART.

TABLE 25-10: SUMMARY OF REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE RECEPTION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	298
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	90
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	91
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	94
RCREG	EUSART Receive Data Register								292*
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	297
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	134
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	296

Legend: — = unimplemented read as '0'. Shaded cells are not used for Synchronous Slave Reception.

* Page provides register information.

TABLE 29-3: PIC16(L)F193X ENHANCED INSTRUCTION SET (CONTINUED)

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb	LSb	Status Affected	Notes
CONTROL OPERATIONS							
BRA	k	Relative Branch	2	11 001k kkkk kkkk			
BRW	–	Relative Branch with W	2	00 0000 0000 1011			
CALL	k	Call Subroutine	2	10 0kkk kkkk kkkk			
CALLW	–	Call Subroutine with W	2	00 0000 0000 1010			
GOTO	k	Go to address	2	10 1kkk kkkk kkkk			
RETFIE	k	Return from interrupt	2	00 0000 0000 1001			
RETLW	k	Return with literal in W	2	11 0100 kkkk kkkk			
RETURN	–	Return from Subroutine	2	00 0000 0000 1000			
INHERENT OPERATIONS							
CLRWDT	–	Clear Watchdog Timer	1	00 0000 0110 0100		$\overline{TO}$, $\overline{PD}$	
NOP	–	No Operation	1	00 0000 0000 0000			
OPTION	–	Load OPTION_REG register with W	1	00 0000 0110 0010			
RESET	–	Software device Reset	1	00 0000 0000 0001			
SLEEP	–	Go into Standby mode	1	00 0000 0110 0011		$\overline{TO}$, $\overline{PD}$	
TRIS	f	Load TRIS register with W	1	00 0000 0110 0fff			
C-COMPILER OPTIMIZED							
ADDFSR	n, k	Add Literal k to FSRn	1	11 0001 0nkk kkkk			
MOVIW	n mm	Move Indirect FSRn to W with pre/post inc/dec modifier, mm	1	00 0000 0001 0nmm		Z	**2, 3**
	k[n]	Move INDFn to W, Indexed Indirect.	1	11 1111 0nkk kkkk		Z	**2**
MOVWI	n mm	Move W to Indirect FSRn with pre/post inc/dec modifier, mm	1	00 0000 0001 1nmm			**2, 3**
	k[n]	Move W to INDFn, Indexed Indirect.	1	11 1111 1nkk kkkk			**2**

Note 1: If the Program Counter (PC) is modified, or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

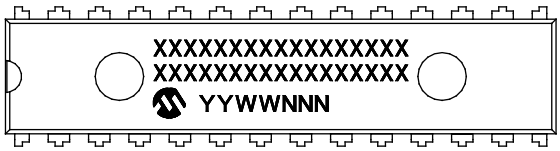
2: If this instruction addresses an INDF register and the MSb of the corresponding FSR is set, this instruction will require one additional instruction cycle.

3: See Table in the MOVIW and MOVWI instruction descriptions.

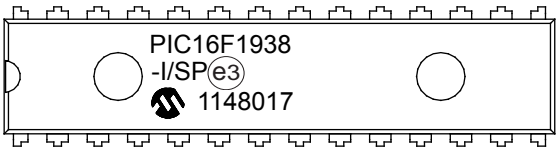
33.0 PACKAGING INFORMATION

33.1 Package Marking Information

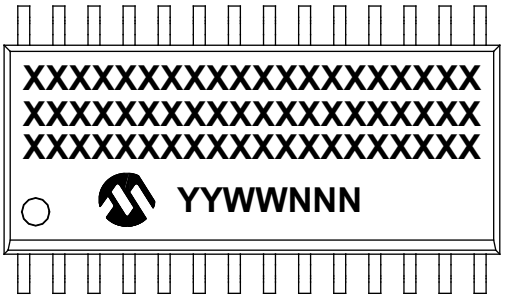
28-Lead SPDIP (.300")



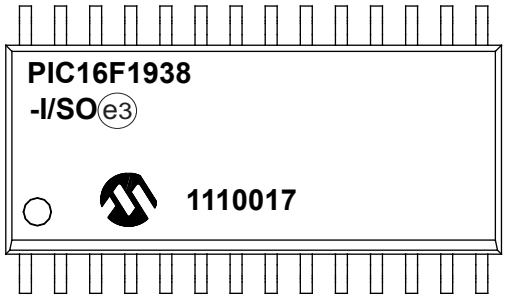
Example



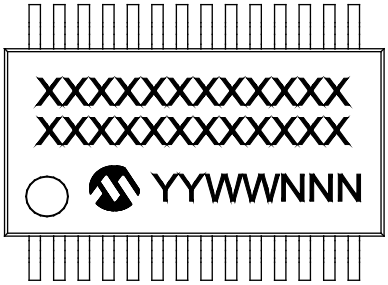
28-Lead SOIC (7.50 mm)



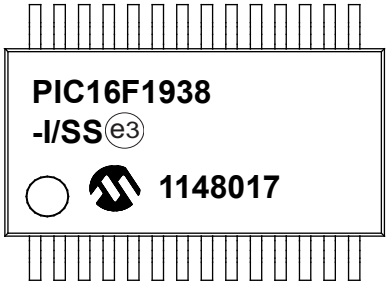
Example



28-Lead SSOP (5.30 mm)



Example



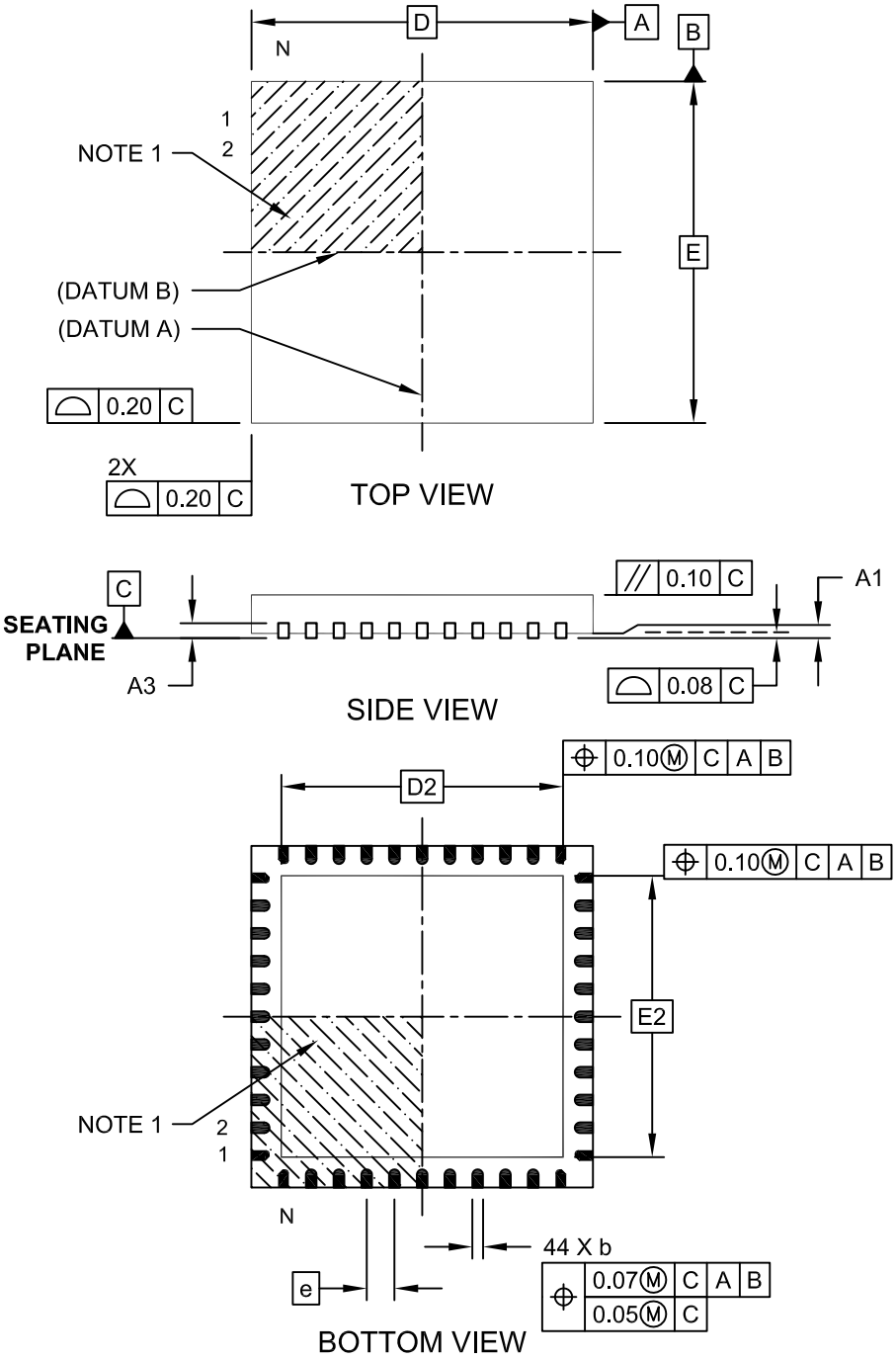
Legend:	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free 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.

* Standard PICmicro® device marking consists of Microchip part number, year code, week code and traceability code. For PICmicro device marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

44-Lead Plastic Quad Flat, No Lead Package (ML) - 8x8 mm Body [QFN]

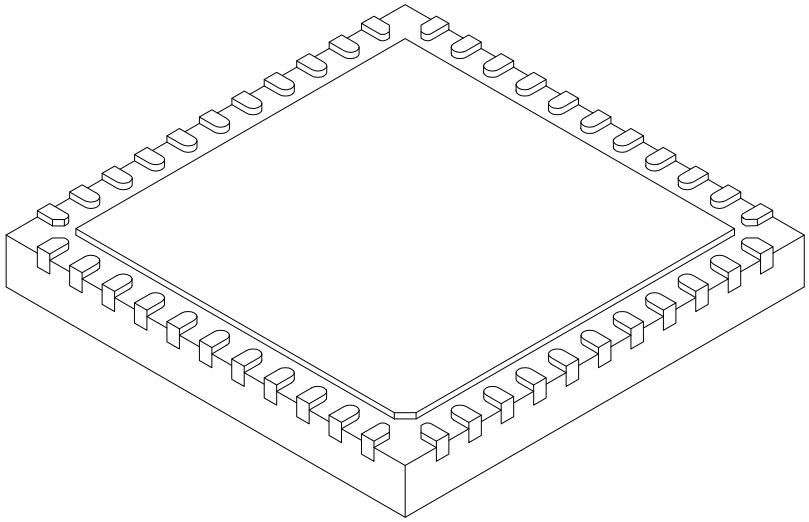
Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Microchip Technology Drawing C04-103C Sheet 1 of 2

44-Lead Plastic Quad Flat, No Lead Package (ML) - 8x8 mm Body [QFN]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		44	
Pitch	e		0.65 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Terminal Thickness	A3		0.20 REF	
Overall Width	E		8.00 BSC	
Exposed Pad Width	E2	6.25	6.45	6.60
Overall Length	D		8.00 BSC	
Exposed Pad Length	D2	6.25	6.45	6.60
Terminal Width	b	0.20	0.30	0.35
Terminal Length	L	0.30	0.40	0.50
Terminal-to-Exposed-Pad	K	0.20	-	-

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

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

   REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-103C Sheet 2 of 2