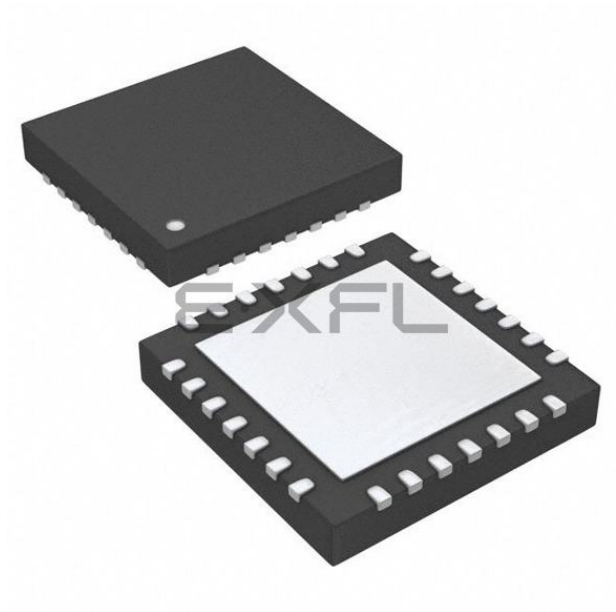


Welcome to E-XFL.COM

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	25
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 17x10b; D/A 1x5b, 1x8b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-UFQFN Exposed Pad
Supplier Device Package	28-UQFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f1713-i-mv

Clocking Structure:

- 16 MHz Internal Oscillator Block:
  - ±1% at calibration
  - Selectable frequency range from 0 to 32 MHz
- 31 kHz Low-Power Internal Oscillator
- External Oscillator Block with:
  - Three crystal/resonator modes up to 20 MHz
  - Two external clock modes up to 20 MHz
- Fail-Safe Clock Monitor
- Two-Speed Oscillator Start-up
- Oscillator Start-up Timer (OST)

Programming/Debug Features:

- In-Circuit Debug Integrated On-Chip
- Emulation Header for Advanced Debug:
  - Provides trace, background debug and up to 32 hardware break points
- In-Circuit Serial Programming™ (ICSP™) via Two Pins

PIC16(L)F1713/6 Family Types

Device	Data Sheet Index	Program Memory Flash (words)	Data SRAM (bytes)	High-Endurance Flash (bytes)	I/Os(2)	10-bit ADC (ch)	5/8-bit DAC	High-Speed/ Comparators	Op Amp	Zero Cross	Timers (8/16-bit)	CCP	PWM	COG	EUSART	MSSP (I^2C/SPI)	CLC	NCO	PPS	Debug(1)	XLP
PIC16(L)F1713	(1)	4096	512	128	25	17	1/1	2	2	1	4/1	2	2	1	1	1	4	1	Y	I/E	Y
PIC16(L)F1716	(1)	8192	1024	128	25	17	1/1	2	2	1	4/1	2	2	1	1	1	4	1	Y	I/E	Y
PIC16(L)F1717	(2)	8192	1024	128	36	28	1/1	2	2	1	4/1	2	2	1	1	1	4	1	Y	I/E	Y
PIC16(L)F1718	(2)	16384	2048	128	25	17	1/1	2	2	1	4/1	2	2	1	1	1	4	1	Y	I/E	Y
PIC16(L)F1719	(2)	16384	2048	128	36	28	1/1	2	2	1	4/1	2	2	1	1	1	4	1	Y	I/E	Y

Note 1: Debugging Methods: (I) – Integrated on Chip; (H) – using Debug Header; E – using Emulation Header.
2: One pin is input-only.

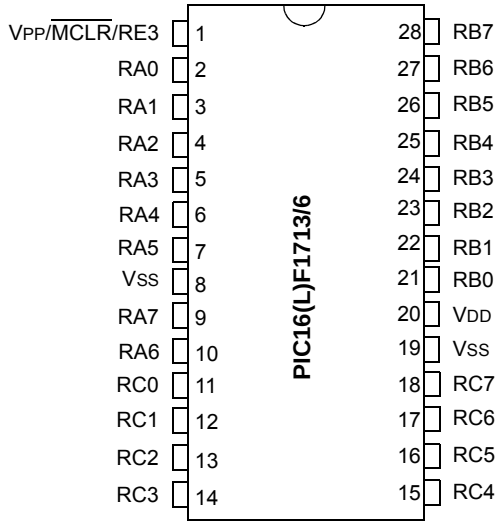
Data Sheet Index: (Unshaded devices are described in this document.)

1: DS40001726 PIC16(L)F1713/6 Data Sheet, 28-Pin Flash, 8-bit Microcontrollers.
2: DS40001740 PIC16(L)F1717/8/9 Data Sheet, 28/40-Pin Flash, 8-bit Microcontrollers.

Note: For other small form-factor package availability and marking information, please visit **http://www.microchip.com/packaging** or contact your local sales office.

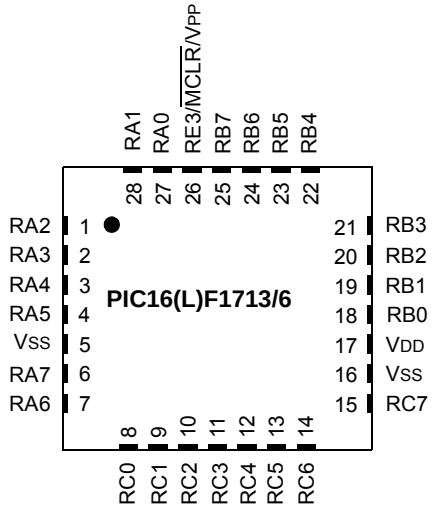
Pin Diagrams

FIGURE 1: 28-PIN PDIP, SOIC, SSOP



Note: See Table 1 for the pin allocation table.

FIGURE 2: 28-PIN (U)QFN



Note: See Table 1 for the pin allocation table.

					Comparator					Timers			CCP			PWM		COG				MSSP			EUSART		CLC						
RC4	15	12	AN16																			SDI(1) SDA(1)								IOC	Y	
RC5	16	13	AN17																											IOC	Y	
RC6	17	14	AN18																						CK(3)					IOC	Y	
RC7	18	15	AN19																						RX(3)					IOC	Y	
RE3	1	26																												IOC	Y	MCLR Vpp
Vdd	20	17																														Vdd
Vss	8	5																														Vss
	19	16																														
OUT(4)					C1OUT	C2OUT							CCP1	CCP2	NCO1OUT	PWM3OUT	PWM4OUT	COG1A	COG1B	COG1C	COG1D	SDA(3)	SCK/SCL(3)	SDO	TX/CK	DT(3)	CLC4OUT	CLC3OUT	CLC2OUT	CLC1OUT			
IN(5)										T1G	T1CKI	T0CKI	CCP1	CCP2				COG1IN				SDI	SCK/SCL(3)	SS	RX(3)	CK	CLCIN0	CLCIN1	CLCIN2	CLCIN3	INT		

Note 1: Default peripheral input. Alternate pins can be selected as the peripheral input with the PPS input selection registers.

2: All pin digital outputs default to PORT latch data. Alternate outputs can be selected as the peripheral digital output with the PPS output selection registers.

3: These peripheral functions are bidirectional. The output pin selections must be the same as the input pin selections.

4: Alternate outputs are excluded from solid shaded areas.

5: Alternate inputs are excluded from dot shaded areas.

3.4.1 SPECIAL FUNCTION REGISTER

The Special Function Registers are registers used by the application to control the desired operation of peripheral functions in the device. The Special Function Registers occupy the 20 bytes after the core registers of every data memory bank (addresses x0Ch/x8Ch through x1Fh/x9Fh). The registers associated with the operation of the peripherals are described in the appropriate peripheral chapter of this data sheet.

3.4.2 GENERAL PURPOSE RAM

There are up to 80 bytes of GPR in each data memory bank. The Special Function Registers occupy the 20 bytes after the core registers of every data memory bank (addresses x0Ch/x8Ch through x1Fh/x9Fh).

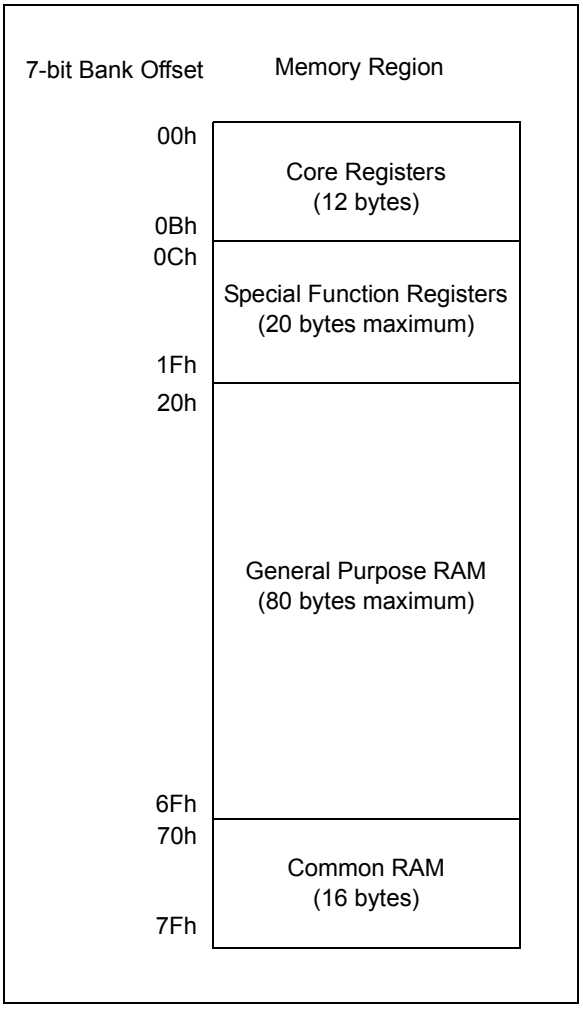
3.4.2.1 Linear Access to GPR

The general purpose RAM can be accessed in a non-banked method via the FSRs. This can simplify access to large memory structures. See **Section 3.7.2 "Linear Data Memory"** for more information.

3.4.3 COMMON RAM

There are 16 bytes of common RAM accessible from all banks.

FIGURE 3-3: BANKED MEMORY PARTITIONING



3.4.4 DEVICE MEMORY MAPS

The memory maps for the device family are as shown in Tables 3-3 through 3-9.

6.2 Clock Source Types

Clock sources can be classified as external or internal.

External clock sources rely on external circuitry for the clock source to function. Examples are: oscillator modules (ECH, ECM, ECL mode), quartz crystal resonators or ceramic resonators (LP, XT and HS modes) and Resistor-Capacitor (EXTRC) mode circuits.

Internal clock sources are contained within the oscillator module. The internal oscillator block has two internal oscillators and a dedicated Phase-Lock Loop (HFPLL) that are used to generate three internal system clock sources: the 16 MHz High-Frequency Internal Oscillator (HFINTOSC), 500 kHz (MFINTOSC) and the 31 kHz Low-Frequency Internal Oscillator (LFINTOSC).

The system clock can be selected between external or internal clock sources via the System Clock Select (SCS) bits in the OSCCON register. See **Section 6.3 "Clock Switching"** for additional information.

6.2.1 EXTERNAL CLOCK SOURCES

An external clock source can be used as the device system clock by performing one of the following actions:

- Program the FOSC<2:0> bits in the Configuration Words to select an external clock source that will be used as the default system clock upon a device Reset.
- Write the SCS<1:0> bits in the OSCCON register to switch the system clock source to:
  - Secondary oscillator during run-time, or
  - An external clock source determined by the value of the FOSC bits.

See **Section 6.3 "Clock Switching"**for more information.

6.2.1.1 EC Mode

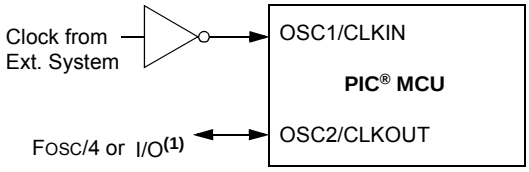
The External Clock (EC) mode allows an externally generated logic level signal to be the system clock source. When operating in this mode, an external clock source is connected to the OSC1 input. OSC2/CLKOUT is available for general purpose I/O or CLKOUT. Figure 6-2 shows the pin connections for EC mode.

EC mode has three power modes to select from through Configuration Words:

- ECH – High power, 4-32 MHz
- ECM – Medium power, 0.5-4 MHz
- ECL – Low power, 0-0.5 MHz

The Oscillator Start-up Timer (OST) is disabled when EC mode is selected. Therefore, there is no delay in operation after a Power-on Reset (POR) or wake-up from Sleep. Because the PIC® MCU design is fully static, stopping the external clock input will have the effect of halting the device while leaving all data intact. Upon restarting the external clock, the device will resume operation as if no time had elapsed.

FIGURE 6-2: EXTERNAL CLOCK (EC) MODE OPERATION



Note 1: Output depends upon $\overline{\text{CLKOUTEN}}$ bit of the Configuration Words.

6.2.1.2 LP, XT, HS Modes

The LP, XT and HS modes support the use of quartz crystal resonators or ceramic resonators connected to OSC1 and OSC2 (Figure 6-3). The three modes select a low, medium or high gain setting of the internal inverter-amplifier to support various resonator types and speed.

LP Oscillator mode selects the lowest gain setting of the internal inverter-amplifier. LP mode current consumption is the least of the three modes. This mode is designed to drive only 32.768 kHz tuning-fork type crystals (watch crystals).

XT Oscillator mode selects the intermediate gain setting of the internal inverter-amplifier. XT mode current consumption is the medium of the three modes. This mode is best suited to drive resonators with a medium drive level specification.

HS Oscillator mode selects the highest gain setting of the internal inverter-amplifier. HS mode current consumption is the highest of the three modes. This mode is best suited for resonators that require a high drive setting.

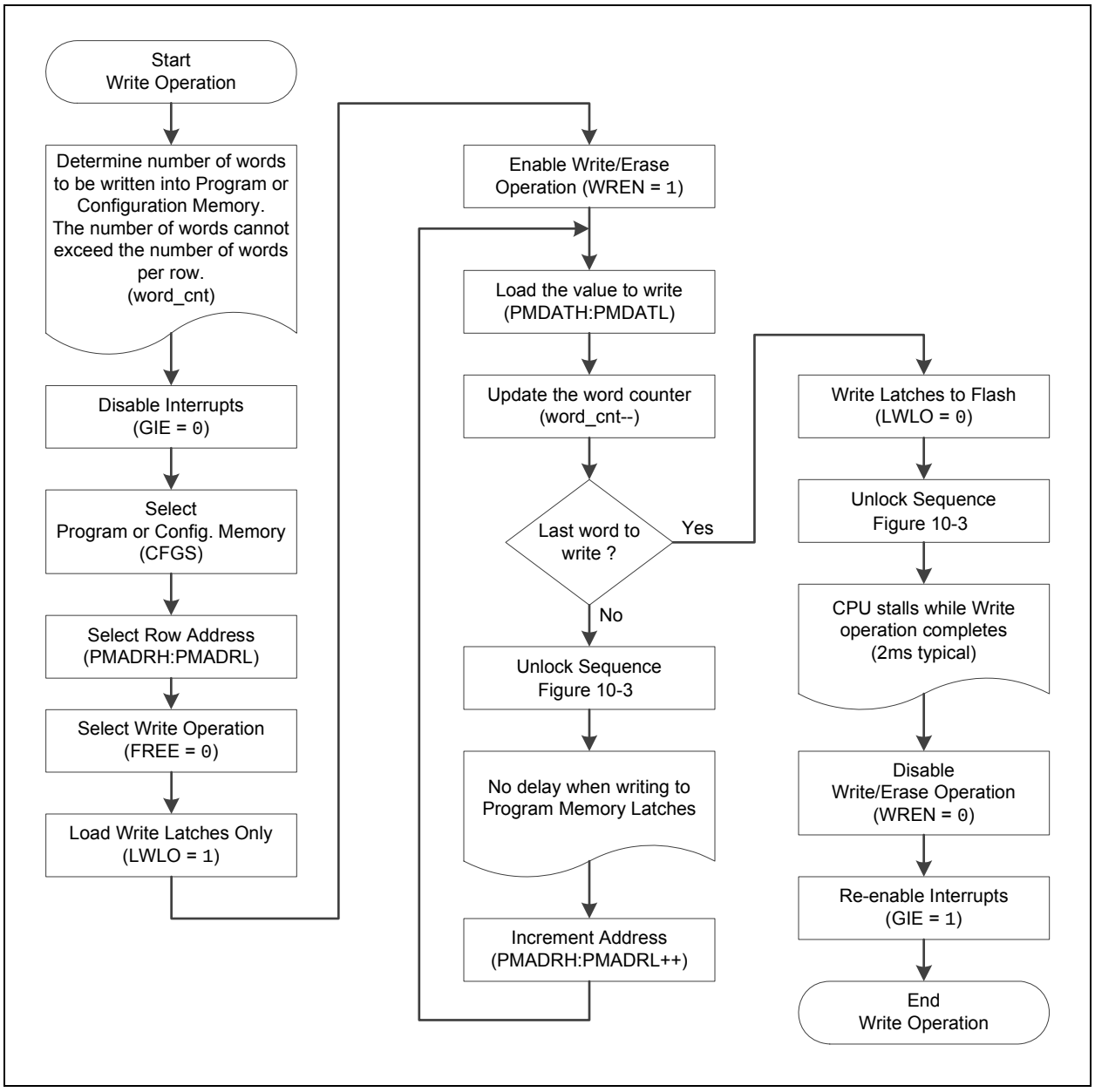
Figure 6-3 and Figure 6-4 show typical circuits for quartz crystal and ceramic resonators, respectively.

TABLE 7-1: SUMMARY OF REGISTERS ASSOCIATED WITH INTERRUPTS

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	83
OPTION_REG	$\overline{\text{WPUEN}}$	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			256
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	84
PIE2	OSFIE	C2IE	C1IE	—	BCL1IE	TMR6IE	TMR4IE	CCP2IE	85
PIE3	—	NCOIE	COGIE	ZCDIE	CLC4IE	CLC3IE	CLC2IE	CLC1IE	86
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	87
PIR2	OSFIF	C2IF	C1IF	—	BCL1IF	TMR6IF	TMR4IF	CCP2IF	88
PIR3	—	NCOIF	COGIF	ZCDIF	CLC4IF	CLC3IF	CLC2IF	CLC1IF	89

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by interrupts.

FIGURE 10-6: FLASH PROGRAM MEMORY WRITE FLOWCHART



10.5 Write Verify

It is considered good programming practice to verify that program memory writes agree with the intended value. Since program memory is stored as a full page then the stored program memory contents are compared with the intended data stored in RAM after the last write is complete.

FIGURE 10-8: FLASH PROGRAM MEMORY VERIFY FLOWCHART

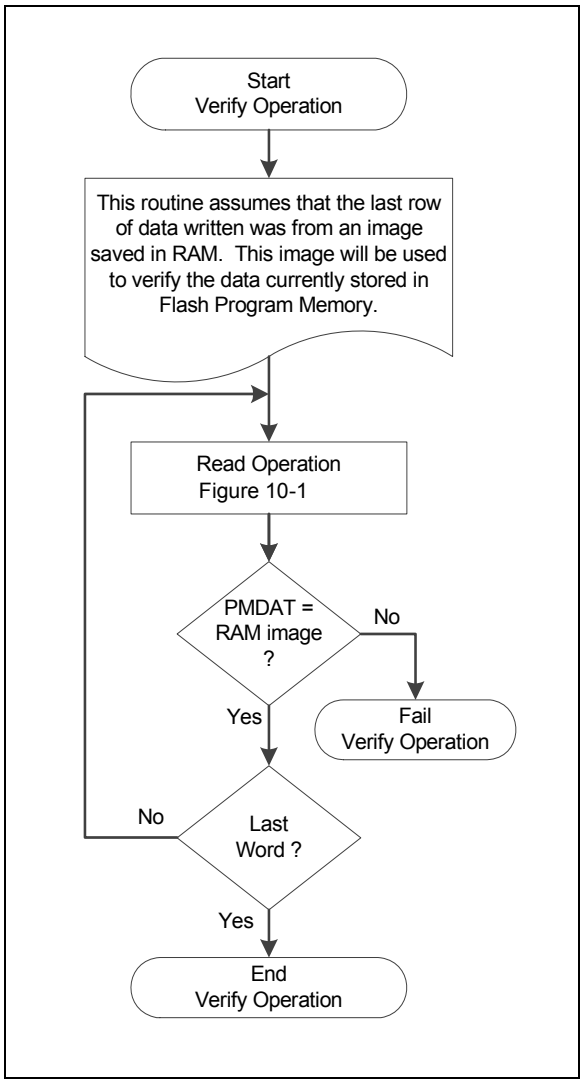
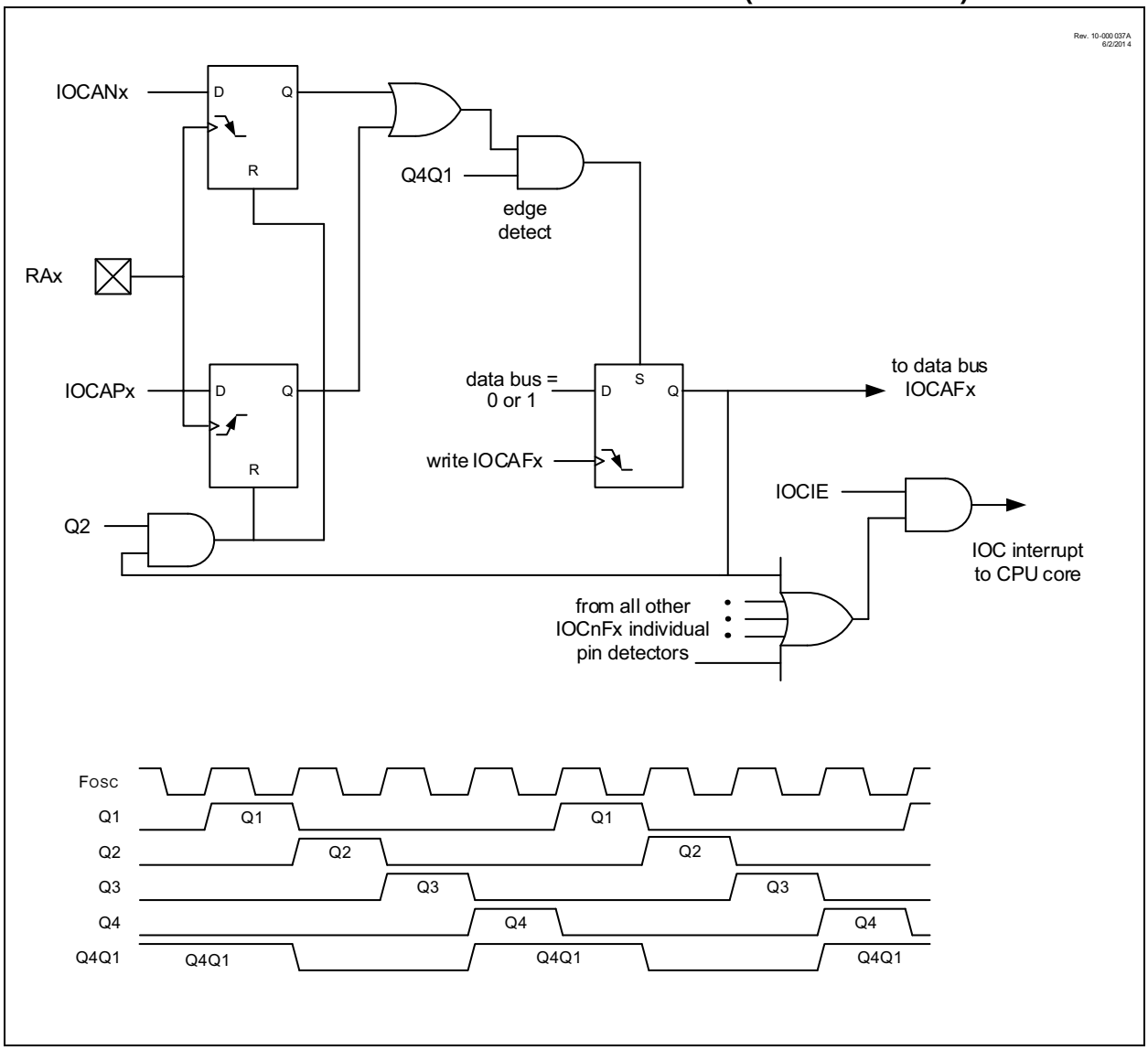


FIGURE 13-1: INTERRUPT-ON-CHANGE BLOCK DIAGRAM (PORTA EXAMPLE)



REGISTER 13-9: IOCCF: INTERRUPT-ON-CHANGE PORTC 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
IOCCF7	IOCCF6	IOCCF5	IOCCF4	IOCCF3	IOCCF2	IOCCF1	IOCCF0
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 **IOCCF<7:0>:** Interrupt-on-Change PORTC Flag bits
1 = An enabled change was detected on the associated pin.
Set when IOCCPx = 1 and a rising edge was detected on RCx, or when IOCCNx = 1 and a falling edge was detected on RCx.
0 = No change was detected, or the user cleared the detected change.

REGISTER 13-10: IOCEP: INTERRUPT-ON-CHANGE PORTE POSITIVE EDGE REGISTER

U-0	U-0	U-0	U-0	R/W-0/0	U-0	U-0	U-0
—	—	—	—	IOCEP3	—	—	—
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 **IOCEP:** Interrupt-on-Change PORTE Positive Edge Enable bits
1 = Interrupt-on-Change enabled on the pin for a positive going edge. IOCEFx bit and IOCIF flag will be set upon detecting an edge.
0 = Interrupt-on-Change disabled for the associated pin.

bit 2-0 **Unimplemented:** Read as ‘0’

REGISTER 13-11: IOCEN: INTERRUPT-ON-CHANGE PORTE NEGATIVE EDGE REGISTER

U-0	U-0	U-0	U-0	R/W-0/0	U-0	U-0	U-0
—	—	—	—	IOCEN3	—	—	—
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 **IOCEN:** Interrupt-on-Change PORTE Negative Edge Enable bits
1 = Interrupt-on-Change enabled on the pin for a negative going edge. IOCEFx bit and IOCIF flag will be set upon detecting an edge.
0 = Interrupt-on-Change disabled for the associated pin.

bit 2-0 **Unimplemented:** Read as ‘0’

REGISTER 13-12: IOCEF: INTERRUPT-ON-CHANGE PORTE FLAG REGISTER

U-0	U-0	U-0	U-0	R/W/HS-0/0	U-0	U-0	U-0
—	—	—	—	IOCEF3	—	—	—
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 7-4 **Unimplemented:** Read as ‘0’

bit 3 **IOCEF:** Interrupt-on-Change PORTE Flag bits
1 = An enabled change was detected on the associated pin.
Set when IOCEPx = 1 and a rising edge was detected on REx, or when IOCENx = 1 and a falling edge was detected on REx.
0 = No change was detected, or the user cleared the detected change.

bit 2-0 **Unimplemented:** Read as ‘0’

16.0 COMPARATOR MODULE

Comparators are used to interface analog circuits to a digital circuit by comparing two analog voltages and providing a digital indication of their relative magnitudes. Comparators are very useful mixed signal building blocks because they provide analog functionality independent of program execution. The analog comparator module includes the following features:

- Independent comparator control
- Programmable input selection
- Comparator output is available internally/externally
- Programmable output polarity
- Interrupt-on-change
- Wake-up from Sleep
- Programmable Speed/Power optimization
- PWM shutdown
- Programmable and fixed voltage reference

16.1 Comparator Overview

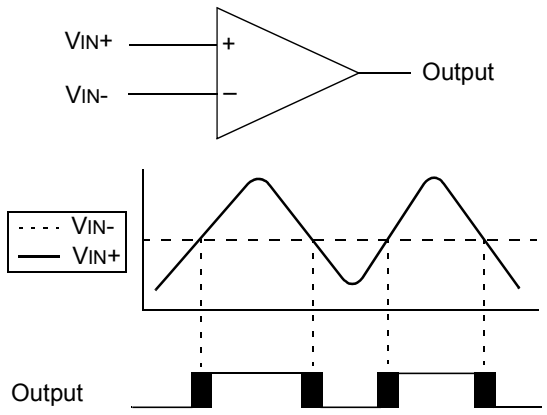
A single comparator is shown in Figure 16-1 along with the relationship between the analog input levels and the digital output. When the analog voltage at VIN+ is less than the analog voltage at VIN-, the output of the comparator is a digital low level. When the analog voltage at VIN+ is greater than the analog voltage at VIN-, the output of the comparator is a digital high level.

The comparators available for this device are located in Table 16-1.

TABLE 16-1: AVAILABLE COMPARATORS

Device	C1	C2
PIC16(L)F1713/6	•	•

FIGURE 16-1: SINGLE COMPARATOR



Note: The black areas of the output of the comparator represents the uncertainty due to input offsets and response time.

16.2 Comparator Control

Each comparator has two control registers: CMxCON0 and CMxCON1.

The CMxCON0 register (see Register 16-1) contains Control and Status bits for the following:

- Enable
- Output
- Output polarity
- Zero latency filter
- Speed/Power selection
- Hysteresis enable
- Output synchronization

The CMxCON1 register (see Register 16-2) contains Control bits for the following:

- Interrupt enable
- Interrupt edge polarity
- Positive input channel selection
- Negative input channel selection

16.2.1 COMPARATOR ENABLE

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

16.2.2 COMPARATOR OUTPUT SELECTION

The output of the comparator can be monitored by reading either the CxOUT bit of the CMxCON0 register or the MCxOUT bit of the CMOUT register. In order to make the output available for an external connection, the following conditions must be true:

- Desired pin PPS control
- Corresponding TRIS bit must be cleared
- CxON bit of the CMxCON0 register must be set

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

16.2.3 COMPARATOR OUTPUT POLARITY

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

Table 16-2 shows the output state versus input conditions, including polarity control.

TABLE 16-2: COMPARATOR OUTPUT STATE VS. INPUT CONDITIONS

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

16.2.4 COMPARATOR SPEED/POWER SELECTION

The trade-off between speed or power can be optimized during program execution with the CxSP control bit. The default state for this bit is '1', which selects the Normal-Speed mode. Device power consumption can be optimized at the cost of slower comparator propagation delay by clearing the CxSP bit to '0'.

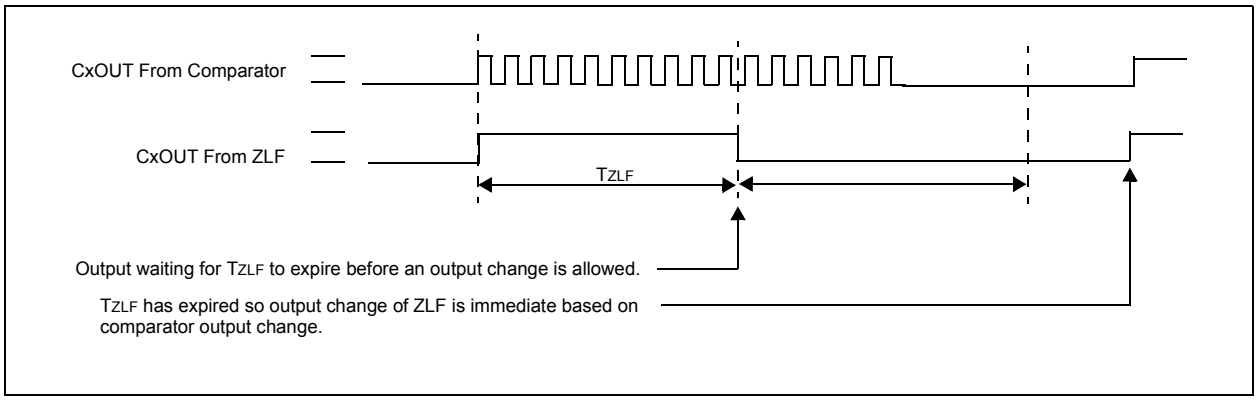
16.8 Comparator Response Time

The comparator output is indeterminate for a period of time after the change of an input source or the selection of a new reference voltage. This period is referred to as the response time. The response time of the comparator differs from the settling time of the voltage reference. Therefore, both of these times must be considered when determining the total response time to a comparator input change. See the Comparator and Voltage Reference Specifications in Table 34-18: Comparator Specifications for more details.

16.9 Zero Latency Filter

In high-speed operation, and under proper circuit conditions, it is possible for the comparator output to oscillate. This oscillation can have adverse effects on the hardware and software relying on this signal. Therefore, a digital filter has been added to the comparator output to suppress the comparator output oscillation. Once the comparator output changes, the output is prevented from reversing the change for a nominal time of 20 ns. This allows the comparator output to stabilize without affecting other dependent devices. Refer to Figure 16-3.

FIGURE 16-3: COMPARATOR ZERO LATENCY FILTER OPERATION



20.2 Fixed Duty Cycle (FDC) Mode

In Fixed Duty Cycle (FDC) mode, every time the accumulator overflows (NCO_overflow), the output is toggled. This provides a 50% duty cycle, provided that the increment value remains constant. For more information, see Figure 20-2.

The FDC mode is selected by clearing the NxPFM bit in the NCOxCON register.

20.3 Pulse Frequency (PF) Mode

In Pulse Frequency (PF) mode, every time the accumulator overflows (NCO_overflow), the output becomes active for one or more clock periods. Once the clock period expires, the output returns to an inactive state. This provides a pulsed output.

The output becomes active on the rising clock edge immediately following the overflow event. For more information, see Figure 20-2.

The value of the active and inactive states depends on the polarity bit, NxPOL in the NCOxCON register.

The PF mode is selected by setting the NxPFM bit in the NCOxCON register.

20.3.1 OUTPUT PULSE WIDTH CONTROL

When operating in PF mode, the active state of the output can vary in width by multiple clock periods. Various pulse widths are selected with the NxPWS<2:0> bits in the NCOxCLK register.

When the selected pulse width is greater than the accumulator overflow time frame, the output of the NCOx operation is indeterminate.

20.4 Output Polarity Control

The last stage in the NCOx module is the output polarity. The NxPOL bit in the NCOxCON register selects the output polarity. Changing the polarity while the interrupts are enabled will cause an interrupt for the resulting output transition.

The NCOx output can be used internally by source code or other peripherals. Accomplish this by reading the NxOUT (read-only) bit of the NCOxCON register.

The NCOx output signal is available to the following peripherals:

- CLC
- CWG

20.5 Interrupts

When the accumulator overflows (NCO_overflow), the NCOx Interrupt Flag bit, NCOxIF, of the PIRx register is set. To enable the interrupt event (NCO_interrupt), the following bits must be set:

- NxEN bit of the NCOxCON register
- NCOxIE bit of the PIEx register
- PEIE bit of the INTCON register
- GIE bit of the INTCON register

The interrupt must be cleared by software by clearing the NCOxIF bit in the Interrupt Service Routine.

20.6 Effects of a Reset

All of the NCOx registers are cleared to zero as the result of a Reset.

20.7 Operation In Sleep

The NCO module operates independently from the system clock and will continue to run during Sleep, provided that the clock source selected remains active.

The HFINTOSC remains active during Sleep when the NCO module is enabled and the HFINTOSC is selected as the clock source, regardless of the system clock source selected.

In other words, if the HFINTOSC is simultaneously selected as the system clock and the NCO clock source, when the NCO is enabled, the CPU will go idle during Sleep, but the NCO will continue to operate and the HFINTOSC will remain active.

This will have a direct effect on the Sleep mode current.

FIGURE 26-6: TIMER1 GATE SINGLE-PULSE AND TOGGLE COMBINED MODE

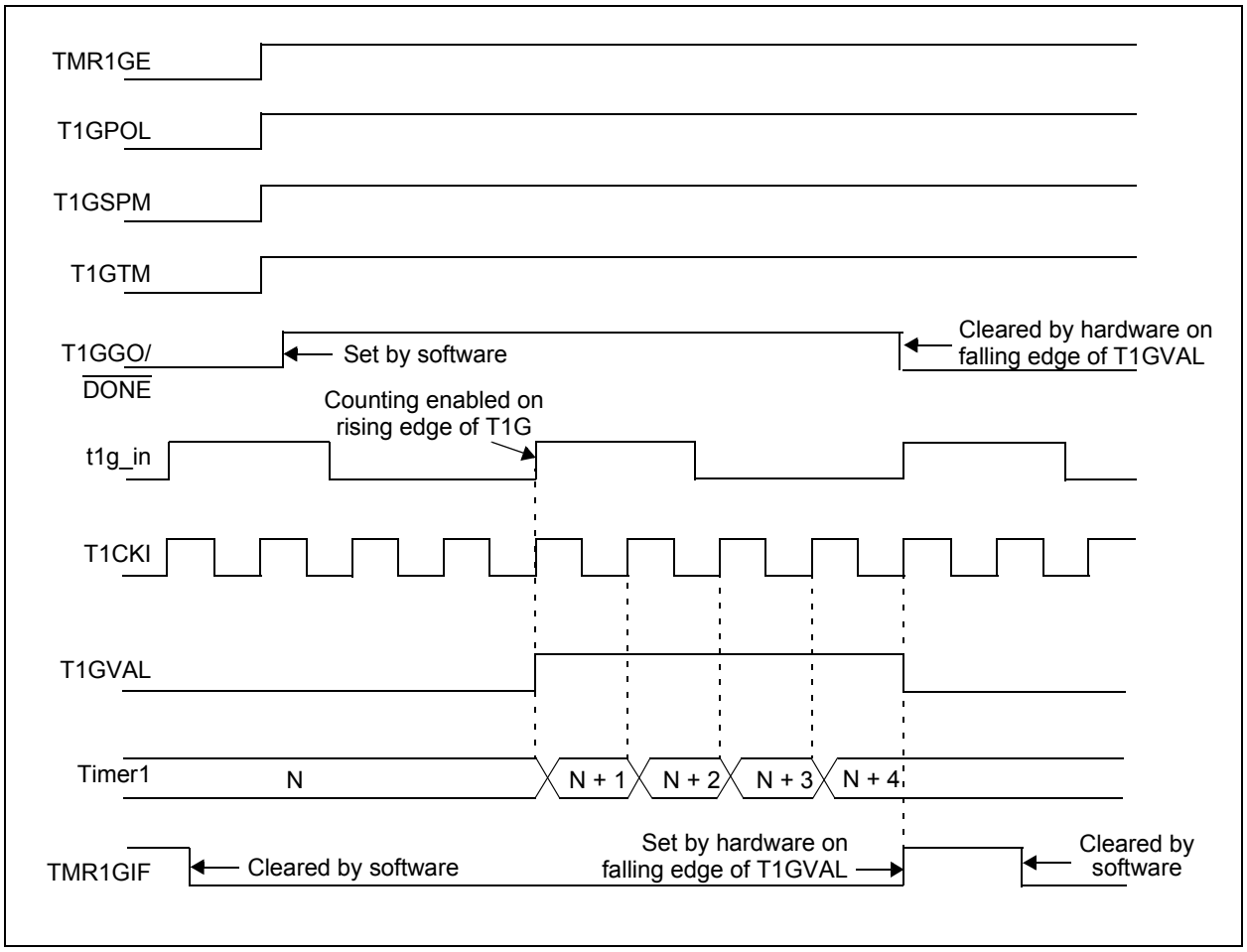


FIGURE 30-9: SPI MODE WAVEFORM (SLAVE MODE WITH CKE = 0)

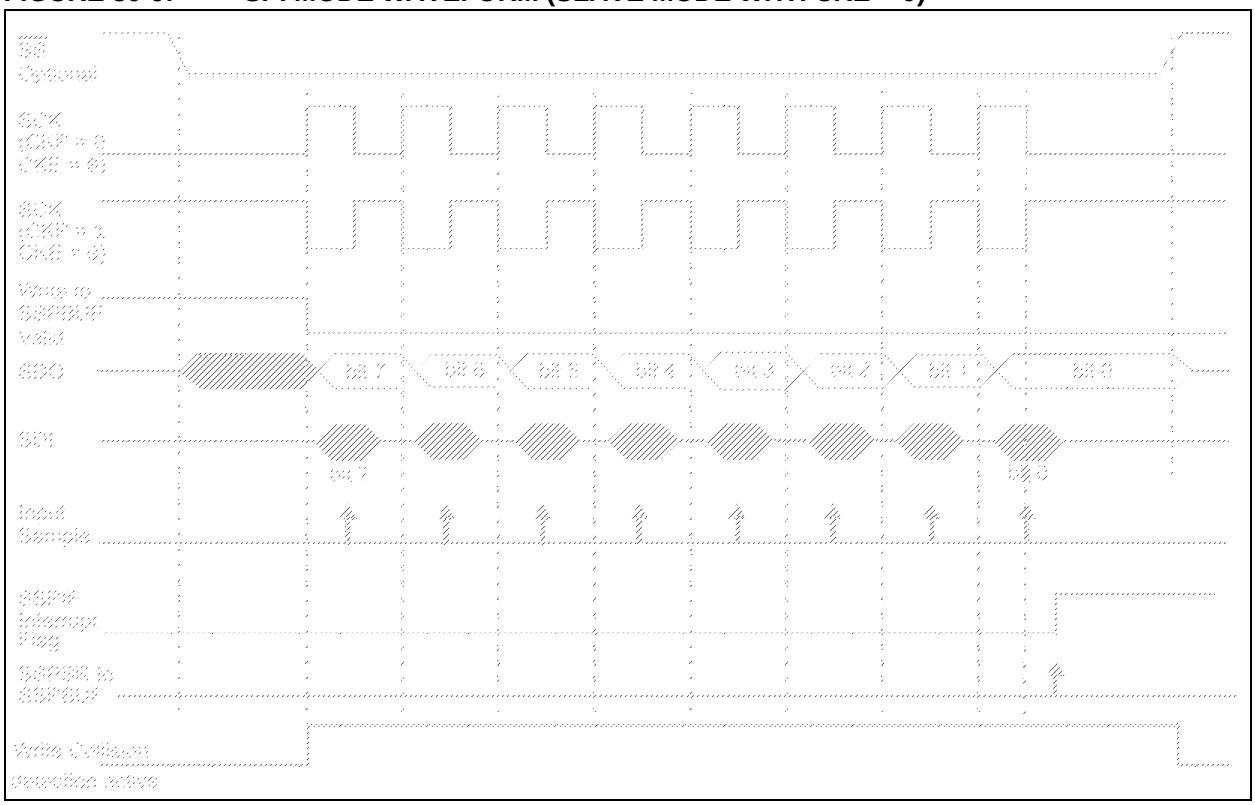
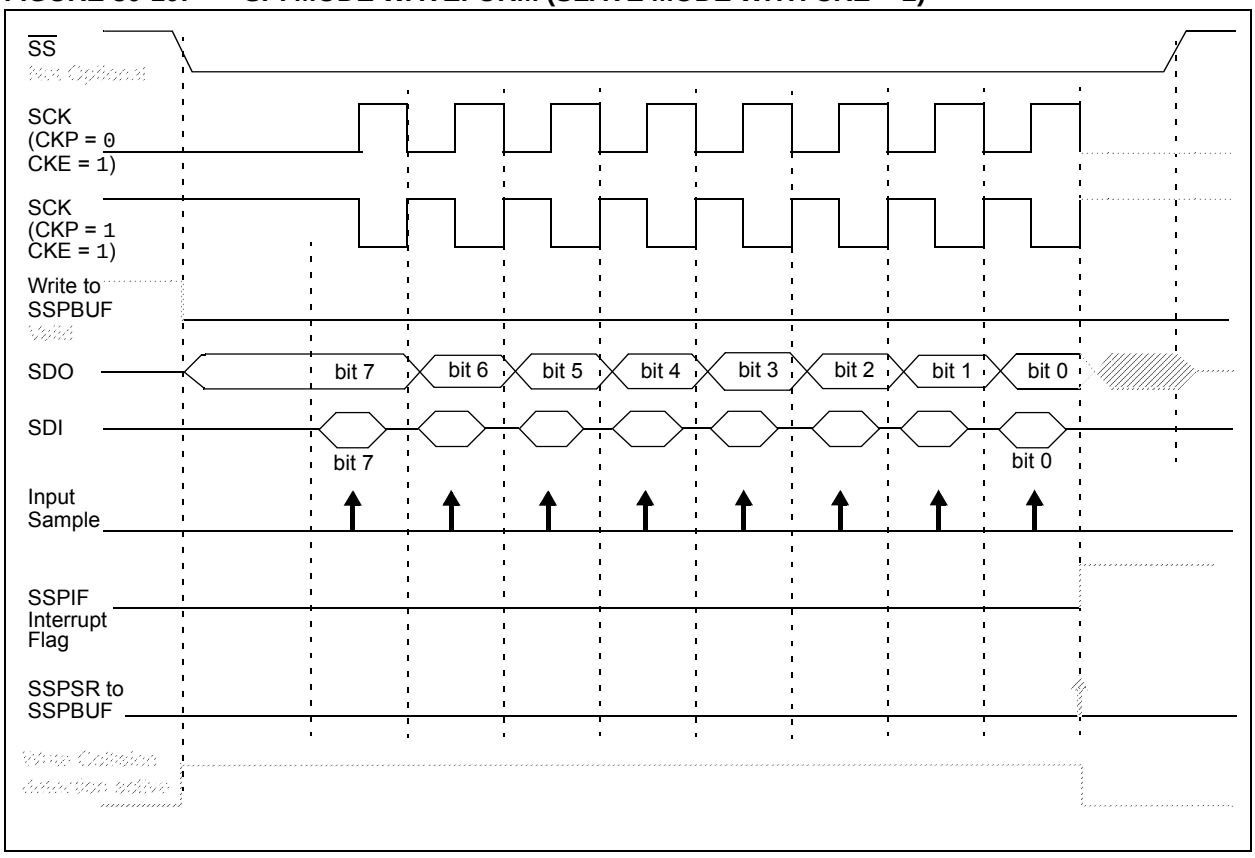


FIGURE 30-10: SPI MODE WAVEFORM (SLAVE MODE WITH CKE = 1)



30.2.6 SPI OPERATION IN SLEEP MODE

In SPI Master mode, module clocks may be operating at a different speed than when in Full-Power mode; in the case of the Sleep mode, all clocks are halted.

Special care must be taken by the user when the MSSP clock is much faster than the system clock.

In Slave mode, when MSSP interrupts are enabled, after the master completes sending data, an MSSP interrupt will wake the controller from Sleep.

If an exit from Sleep mode is not desired, MSSP interrupts should be disabled.

In SPI Master mode, when the Sleep mode is selected, all module clocks are halted and the transmission/reception will remain in that state until the device wakes. After the device returns to Run mode, the module will resume transmitting and receiving data.

In SPI Slave mode, the SPI Transmit/Receive Shift register operates asynchronously to the device. This allows the device to be placed in Sleep mode and data to be shifted into the SPI Transmit/Receive Shift register. When all eight bits have been received, the MSSP interrupt flag bit will be set and if enabled, will wake the device.

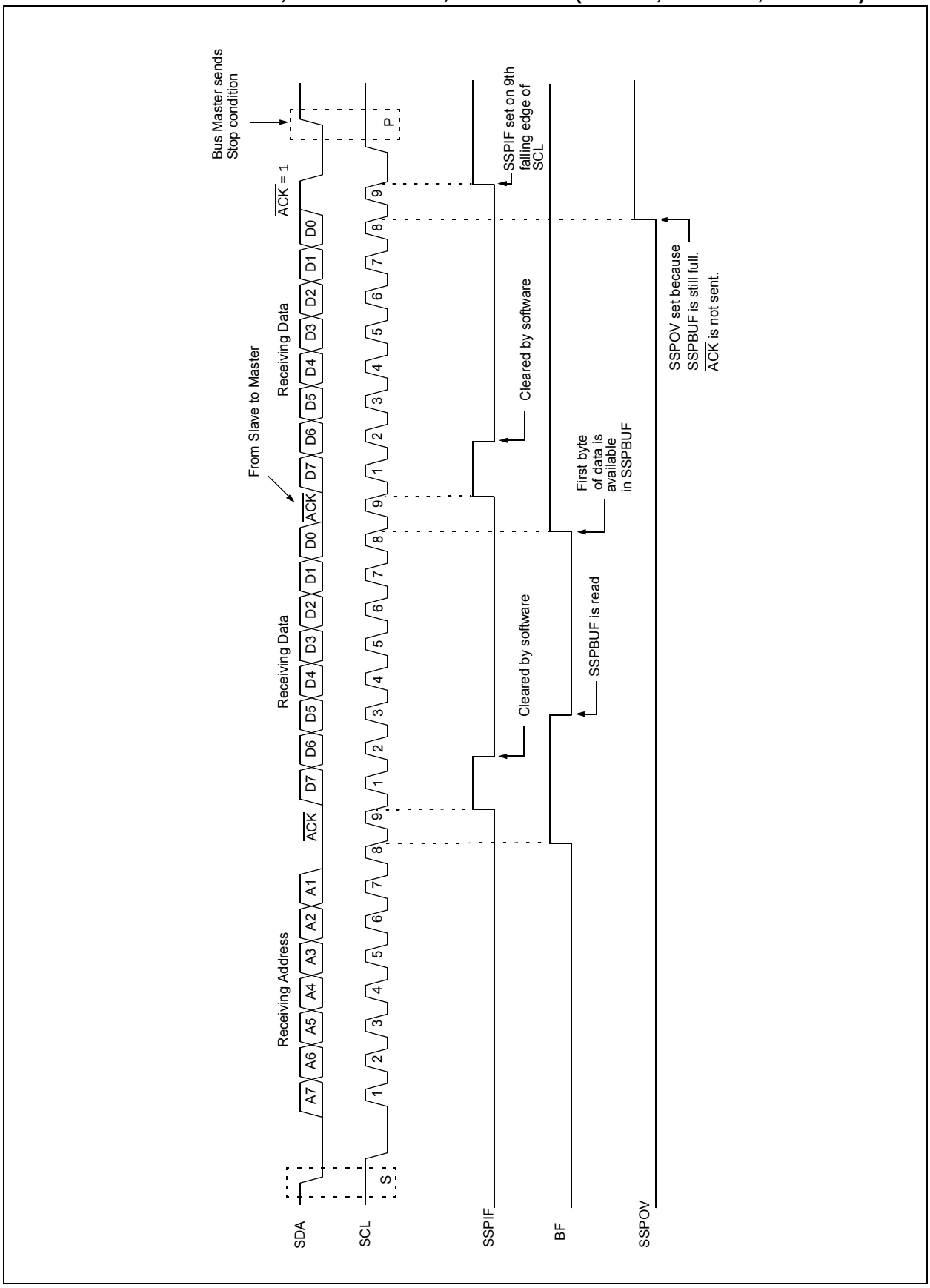
TABLE 30-1: SUMMARY OF REGISTERS ASSOCIATED WITH SPI OPERATION

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	120
ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	126
ANSELC	ANSC7	ANSC6	ANSC5	ANSC4	ANSC3	ANSC2	—	—	131
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	83
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	84
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	87
RxyPPS	—	—	—	RxyPPS<4:0>					137
SSPCLKPPS	—	—	—	SSPCLKPPS<4:0>					136
SSPDATPPS	—	—	—	SSPDATPPS<4:0>					136
SSPSSPPS	—	—	—	SSPSSPPS<4:0>					136
SSP1BUF	Synchronous Serial Port Receive Buffer/Transmit Register								289*
SSP1CON1	WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				333
SSP1CON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	332
SSP1STAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	332
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	119
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	125
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130

Legend: — = Unimplemented location, read as '0'. Shaded cells are not used by the MSSP in SPI mode.

* Page provides register information.

FIGURE 30-14: I²C SLAVE, 7-BIT ADDRESS, RECEPTION (SEN = 0, AHEN = 0, DHEN = 0)



31.1.2.8 Asynchronous Reception Setup:

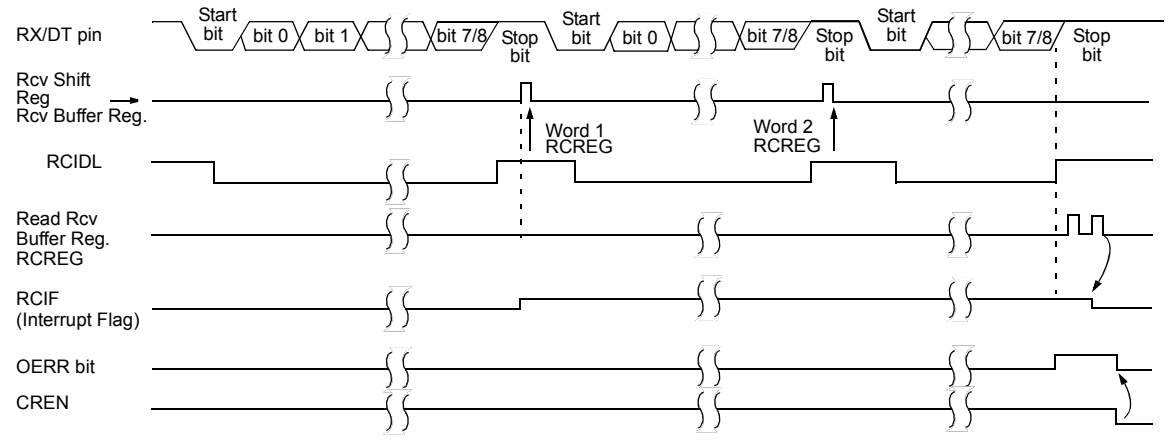
1. Initialize the SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 31.4 "EUSART Baud Rate Generator (BRG)"**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for 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.

31.1.2.9 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 SPBRGH, SPBRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 31.4 "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 31-5: 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.