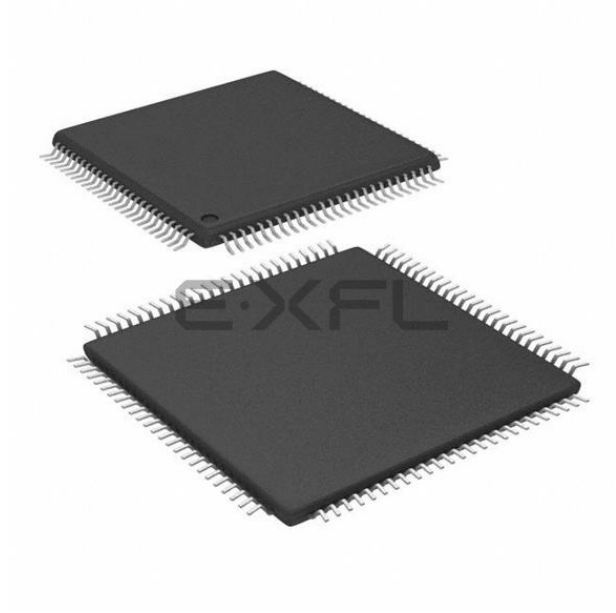


Welcome to E-XFL.COM

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	41.667MHz
Connectivity	EBI/EMI, Ethernet, I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	70
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	3808 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 16x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-TQFP
Supplier Device Package	100-TQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f96j60-i-pt

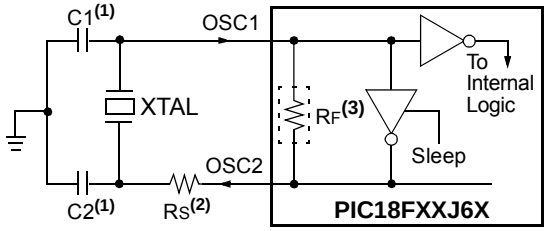
3.3 Crystal Oscillator/Ceramic Resonators (HS Modes)

In HS or HSPLL Oscillator modes, a crystal is connected to the OSC1 and OSC2 pins to establish oscillation. Figure 3-2 shows the pin connections.

The oscillator design requires the use of a crystal that is rated for parallel resonant operation.

Note:	Use of a crystal rated for series resonant operation may give a frequency out of the crystal manufacturer's specifications.

FIGURE 3-2: CRYSTAL OSCILLATOR OPERATION (HS OR HSPLL CONFIGURATION)



Note 1: See Table 3-1 for initial values of C1 and C2.

2: A series resistor (RS) may be required for crystals with a low drive specification.

3: RF varies with the oscillator mode chosen.

TABLE 3-1: CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR

Osc Type	Crystal Freq.	Typical Capacitor Values Tested:	
		C1	C2
HS	25 MHz	33 pF	33 pF

Capacitor values are for design guidance only.

Different capacitor values may be required to produce acceptable oscillator operation. The user should test the performance of the oscillator over the expected VDD and temperature range for the application. Refer to the following application notes for oscillator specific information:

- *AN588, "PIC® Microcontroller Oscillator Design Guide"*
- *AN826, "Crystal Oscillator Basics and Crystal Selection for rfPIC® and PIC® Devices"*
- *AN849, "Basic PIC® Oscillator Design"*
- *AN943, "Practical PIC® Oscillator Analysis and Design"*
- *AN949, "Making Your Oscillator Work"*

See the notes following this table for additional information.

Note 1: Higher capacitance increases the stability of the oscillator but also increases the start-up time.

2: Since each crystal has its own characteristics, the user should consult the crystal manufacturer for appropriate values of external components.

3: Rs may be required to avoid overdriving crystals with low drive level specifications.

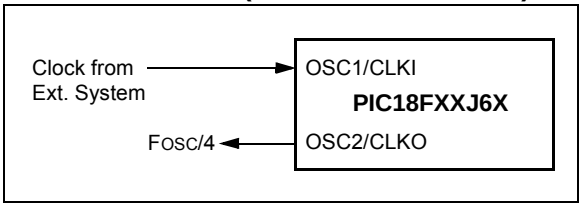
4: Always verify oscillator performance over the VDD and temperature range that is expected for the application.

3.4 External Clock Input (EC Modes)

The EC and ECPLL Oscillator modes require an external clock source to be connected to the OSC1 pin. There is no oscillator start-up time required after a Power-on Reset or after an exit from Sleep mode.

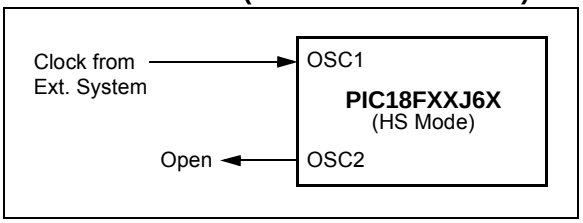
In the EC Oscillator mode, the oscillator frequency, divided by 4, is available on the OSC2 pin. This signal may be used for test purposes or to synchronize other logic. Figure 3-3 shows the pin connections for the EC Oscillator mode.

FIGURE 3-3: EXTERNAL CLOCK INPUT OPERATION (EC CONFIGURATION)



An external clock source may also be connected to the OSC1 pin in the HS mode, as shown in Figure 3-4. In this configuration, the OSC2 pin is left open. Current consumption in this configuration will be somewhat higher than EC mode, as the internal oscillator's feedback circuitry will be enabled (in EC mode, the feedback circuit is disabled).

FIGURE 3-4: EXTERNAL CLOCK INPUT OPERATION (HS CONFIGURATION)



5.2 Master Clear ($\overline{MCLR}$)

The $\overline{MCLR}$ pin provides a method for triggering a hard external Reset of the device. A Reset is generated by holding the pin low. PIC18 extended microcontroller devices have a noise filter in the $\overline{MCLR}$ Reset path which detects and ignores small pulses.

The $\overline{MCLR}$ pin is not driven low by any internal Resets, including the WDT.

5.3 Power-on Reset (POR)

A Power-on Reset condition is generated on-chip whenever VDD rises above a certain threshold. This allows the device to start in the initialized state when VDD is adequate for operation.

To take advantage of the POR circuitry, tie the $\overline{MCLR}$ pin through a resistor (1 kΩ to 10 kΩ) to VDD. This will eliminate external RC components usually needed to create a Power-on Reset delay. A minimum rise rate for VDD is specified (Parameter D004). For a slow rise time, see Figure 5-2.

When the device starts normal operation (i.e., exits the Reset condition), device operating parameters (voltage, frequency, temperature, etc.) must be met to ensure operation. If these conditions are not met, the device must be held in Reset until the operating conditions are met.

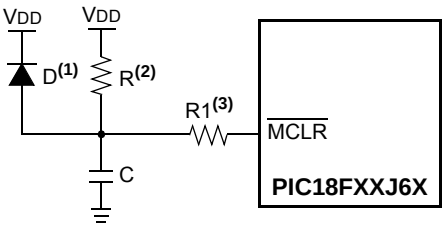
POR events are captured by the $\overline{POR}$ bit (RCON<1>). The state of the bit is set to '0' whenever a Power-on Reset occurs; it does not change for any other Reset event. $\overline{POR}$ is not reset to '1' by any hardware event. To capture multiple events, the user manually resets the bit to '1' in software following any Power-on Reset.

5.4 Brown-out Reset (BOR)

The PIC18F97J60 family of devices incorporates a simple BOR function when the internal regulator is enabled (ENVREG pin is tied to VDD). Any drop of VDD below VBOR (Parameter D005), for greater than time, TBOR (Parameter 35), will reset the device. A Reset may or may not occur if VDD falls below VBOR for less than TBOR. The chip will remain in Brown-out Reset until VDD rises above VBOR.

Once a BOR has occurred, the Power-up Timer will keep the chip in Reset for TPWRT (Parameter 33). If VDD drops below VBOR while the Power-up Timer is running, the chip will go back into a Brown-out Reset and the Power-up Timer will be initialized. Once VDD rises above VBOR, the Power-up Timer will execute the additional time delay.

FIGURE 5-2: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW VDD POWER-UP)



Note 1: External Power-on Reset circuit is required only if the VDD power-up slope is too slow. The diode, D, helps discharge the capacitor quickly when VDD powers down.

2: R < 40 kΩ is recommended to make sure that the voltage drop across R does not violate the device's electrical specification.

3: R1 ≥ 1 kΩ will limit any current flowing into $\overline{MCLR}$ from external capacitor, C, in the event of $\overline{MCLR}$/VPP pin breakdown, due to Electrostatic Discharge (ESD), or Electrical Overstress (EOS).

5.4.1 DETECTING BOR

The $\overline{BOR}$ bit always resets to '0' on any Brown-out Reset or Power-on Reset event. This makes it difficult to determine if a Brown-out Reset event has occurred just by reading the state of $\overline{BOR}$ alone. A more reliable method is to simultaneously check the state of both $\overline{POR}$ and $\overline{BOR}$. This assumes that the $\overline{POR}$ bit is reset to '1' in software immediately after any Power-on Reset event. If $\overline{BOR}$ is '0' while $\overline{POR}$ is '1', it can be reliably assumed that a Brown-out Reset event has occurred.

If the voltage regulator is disabled, Brown-out Reset functionality is disabled. In this case, the $\overline{BOR}$ bit cannot be used to determine a Brown-out Reset event. The $\overline{BOR}$ bit is still cleared by a Power-on Reset event.

5.5 Configuration Mismatch (CM)

The Configuration Mismatch (CM) Reset is designed to detect and attempt to recover from random, memory corrupting events. These include Electrostatic Discharge (ESD) events which can cause widespread single-bit changes throughout the device and result in catastrophic failure.

In PIC18FXXJ Flash devices, the device Configuration registers (located in the configuration memory space) are continuously monitored during operation by comparing their values to complimentary shadow registers. If a mismatch is detected between the two sets of registers, a CM Reset automatically occurs. These events are captured by the $\overline{CM}$ bit (RCON<5>). The state of the bit is set to '0' whenever a CM event occurs; it does not change for any other Reset event.

TABLE 6-5: REGISTER FILE SUMMARY (PIC18F97J60 FAMILY) (CONTINUED)

File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Values on POR, BOR	Details on Page:
PORTF	RF7	RF6	RF5	RF4	RF3	RF2	RF1	RF0(5)	0000 0000	72, 161
PORTE	RE7(6)	RE6(6)	RE5	RE4	RE3	RE2	RE1	RE0	xxxx xxxx	72, 159
PORTD	RD7(5)	RD6(5)	RD5(5)	RD4(5)	RD3(5)	RD2	RD1	RD0	xxxx xxxx	72, 156
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	72, 153
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	72, 150
PORTA	RJPU(6)	—	RA5	RA4	RA3	RA2	RA1	RA0	0-0x 0000	72, 147
SPBRGH1	EUSART1 Baud Rate Generator Register High Byte								0000 0000	72, 320
BAUDCON1	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN	0100 0-00	72, 318
SPBRGH2	EUSART2 Baud Rate Generator Register High Byte								0000 0000	72, 320
BAUDCON2	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN	0100 0-00	72, 318
ERDPTH	—	—	—	Buffer Read Pointer High Byte					---0 0101	72, 223
ERDPTL	Buffer Read Pointer Low Byte								1111 1010	72, 223
ECCP1DEL	P1RSEN	P1DC6	P1DC5	P1DC4	P1DC3	P1DC2	P1DC1	P1DC0	0000 0000	72, 211
TMR4	Timer4 Register								0000 0000	72, 187
PR4	Timer4 Period Register								1111 1111	72, 187
T4CON	—	T4OUTPS3	T4OUTPS2	T4OUTPS1	T4OUTPS0	TMR4ON	T4CKPS1	T4CKPS0	-000 0000	72, 187
CCPR4H	Capture/Compare/PWM Register 4 High Byte								xxxx xxxx	72, 193
CCPR4L	Capture/Compare/PWM Register 4 Low Byte								xxxx xxxx	72, 193
CCP4CON	—	—	DC4B1	DC4B0	CCP4M3	CCP4M2	CCP4M1	CCP4M0	--00 0000	73, 189
CCPR5H	Capture/Compare/PWM Register 5 High Byte								xxxx xxxx	73, 193
CCPR5L	Capture/Compare/PWM Register 5 Low Byte								xxxx xxxx	73, 193
CCP5CON	—	—	DC5B1	DC5B0	CCP5M3	CCP5M2	CCP5M1	CCP5M0	--00 0000	73, 189
SPBRG2	EUSART2 Baud Rate Generator Register Low Byte								0000 0000	73, 320
RCREG2	EUSART2 Receive Register								0000 0000	73, 327
TXREG2	EUSART2 Transmit Register								0000 0000	73, 329
TXSTA2	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	0000 0010	73, 316
RCSTA2	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	73, 317
ECCP3AS	ECCP3ASE	ECCP3AS2	ECCP3AS1	ECCP3AS0	PSS3AC1	PSS3AC0	PSS3BD1	PSS3BD0	0000 0000	73, 212
ECCP3DEL	P3RSEN	P3DC6	P3DC5	P3DC4	P3DC3	P3DC2	P3DC1	P3DC0	0000 0000	73, 211
ECCP2AS	ECCP2ASE	ECCP2AS2	ECCP2AS1	ECCP2AS0	PSS2AC1	PSS2AC0	PSS2BD1	PSS2BD0	0000 0000	73, 212
ECCP2DEL	P2RSEN	P2DC6	P2DC5	P2DC4	P2DC3	P2DC2	P2DC1	P2DC0	0000 0000	73, 211
SSP2BUF	MSSP2 Receive Buffer/Transmit Register								xxxx xxxx	73, 279
SSP2ADD	MSSP2 Address Register (I^2C™ Slave mode), MSSP2 Baud Rate Reload Register (I^2C Master mode)								0000 0000	73, 279
SSP2STAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	0000 0000	73, 270
SSP2CON1	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	73, 271, 281
SSP2CON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	73, 282
	GCEN	ACKSTAT	ADMSK5(4)	ADMSK4(4)	ADMSK3(4)	ADMSK2(4)	ADMSK1(4)	SEN		
EDATA	Ethernet Transmit/Receive Buffer Register (EDATA<7:0>)								xxxx xxxx	73, 223
EIR	—	PKTIF	DMAIF	LINKIF	TXIF	—	TXERIF	RXERIF	-000 0-00	73, 241
ECON2	AUTOINC	PKTDEC	ETHEN	—	—	—	—	—	100- ----	73, 228

Legend: x = unknown; u = unchanged; - = unimplemented, read as '0'; q = value depends on condition; r = reserved bit, do not modify. Shaded cells are unimplemented, read as '0'.

Note 1: Bit 7 and bit 6 are cleared by user software or by a POR.

2: Bit 21 of the PC is only available in Serial Programming modes.

3: Reset value is '0' when Two-Speed Start-up is enabled and '1' if disabled.

4: Alternate names and definitions for these bits when the MSSP module is operating in I^2C™ Slave mode.

5: These bits and/or registers are only available in 100-pin devices; otherwise, they are unimplemented and read as '0'. Reset values shown apply only to 100-pin devices.

6: These bits and/or registers are only available in 80-pin and 100-pin devices. In 64-pin devices, they are unimplemented and read as '0'. Reset values are shown for 100-pin devices.

7: In Microcontroller mode, the bits in this register are unwritable and read as '0'.

8: PLLEN is only available when either ECPLL or HSPLL Oscillator mode is selected; otherwise, read as '0'.

9: Implemented in 100-pin devices in Microcontroller mode only.

TABLE 6-5: REGISTER FILE SUMMARY (PIC18F97J60 FAMILY) (CONTINUED)

File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Values on POR, BOR	Details on Page:
ESTAT	—	BUFER	—	r	—	RXBUSY	TXABRT	PHYRDY	-0-0 -000	73, 228
EIE	—	PKTIE	DMAIE	LINKIE	TXIE	—	TXERIE	RXERIE	-000 0-00	73, 240
EDMACSH	DMA Checksum Register High Byte								0000 0000	73, 265
EDMACSL	DMA Checksum Register Low Byte								0000 0000	73, 265
EDMADSTH	—	—	—	DMA Destination Register High Byte					---0 0000	73, 265
EDMADSTL	DMA Destination Register Low Byte								0000 0000	73, 265
EDMANDH	—	—	—	DMA End Register High Byte					---0 0000	73, 265
EDMANDL	DMA End Register Low Byte								0000 0000	73, 265
EDMASTH	—	—	—	DMA Start Register High Byte					---0 0000	73, 265
EDMASTL	DMA Start Register Low Byte								0000 0000	73, 265
ERXWRPTH	—	—	—	Receive Buffer Write Pointer High Byte					---0 0000	73, 225
ERXWRPTL	Receive Buffer Write Pointer Low Byte								0000 0000	73, 225
ERXRDPTH	—	—	—	Receive Buffer Read Pointer High Byte					---0 0101	73, 225
ERXRDPTL	Receive Buffer Read Pointer Low Byte								1111 1010	73, 225
ERXNDH	—	—	—	Receive End Register High Byte					---1 1111	73, 225
ERXNDL	Receive End Register Low Byte								1111 1111	73, 225
ERXSTH	—	—	—	Receive Start Register High Byte					---0 0101	73, 225
ERXSTL	Receive Start Register Low Byte								1111 1010	73, 225
ETXNDH	—	—	—	Transmit End Register High Byte					---0 0000	74, 226
ETXNDL	Transmit End Register Low Byte								0000 0000	74, 226
ETXSTH	—	—	—	Transmit Start Register High Byte					---0 0000	74, 226
ETXSTL	Transmit Start Register Low Byte								0000 0000	74, 226
EWRPTH	—	—	—	Buffer Write Pointer High Byte					---0 0000	74, 223
EWRPTL	Buffer Write Pointer Low Byte								0000 0000	74, 223
EPKTCNT	Ethernet Packet Count Register								0000 0000	74, 252
ERXFCON	UCEN	ANDOR	CRCEN	PMEN	MPEN	HTEN	MCEN	BCEN	1010 0001	74, 260
EPMOH	—	—	—	Pattern Match Offset Register High Byte					---0 0000	74, 263
EPMOL	Pattern Match Offset Register Low Byte								0000 0000	74, 263
EPMCSH	Pattern Match Checksum Register High Byte								0000 0000	74, 263
EPMCSL	Pattern Match Checksum Register Low Byte								0000 0000	74, 263
EPMM7	Pattern Match Mask Register Byte 7								0000 0000	74, 263
EPMM6	Pattern Match Mask Register Byte 6								0000 0000	74, 263
EPMM5	Pattern Match Mask Register Byte 5								0000 0000	74, 263
EPMM4	Pattern Match Mask Register Byte 4								0000 0000	74, 263
EPMM3	Pattern Match Mask Register Byte 3								0000 0000	74, 263
EPMM2	Pattern Match Mask Register Byte 2								0000 0000	74, 263
EPMM1	Pattern Match Mask Register Byte 1								0000 0000	74, 263
EPMM0	Pattern Match Mask Register Byte 0								0000 0000	74, 263

Legend: x = unknown; u = unchanged; - = unimplemented, read as '0'; q = value depends on condition; r = reserved bit, do not modify. Shaded cells are unimplemented, read as '0'.

Note 1: Bit 7 and bit 6 are cleared by user software or by a POR.

2: Bit 21 of the PC is only available in Serial Programming modes.

3: Reset value is '0' when Two-Speed Start-up is enabled and '1' if disabled.

4: Alternate names and definitions for these bits when the MSSP module is operating in I^2C™ Slave mode.

5: These bits and/or registers are only available in 100-pin devices; otherwise, they are unimplemented and read as '0'. Reset values shown apply only to 100-pin devices.

6: These bits and/or registers are only available in 80-pin and 100-pin devices. In 64-pin devices, they are unimplemented and read as '0'. Reset values are shown for 100-pin devices.

7: In Microcontroller mode, the bits in this register are unwritable and read as '0'.

8: PLLEN is only available when either ECPLL or HSPLL Oscillator mode is selected; otherwise, read as '0'.

9: Implemented in 100-pin devices in Microcontroller mode only.

NOTES:

8.8 Operation in Power-Managed Modes

In alternate power-managed Run modes, the external bus continues to operate normally. If a clock source with a lower speed is selected, bus operations will run at that speed. In these cases, excessive access times for the external memory may result if Wait states have been enabled and added to external memory operations. If operations in a lower power Run mode are anticipated, user applications should provide memory access time adjustments at the lower clock speeds.

In Sleep and Idle modes, the microcontroller core does not need to access data; bus operations are suspended. The state of the external bus is frozen, with the address/data pins and most of the control pins holding at the same state they were in when the mode was invoked. The only potential changes are the $\overline{CE}$, $\overline{LB}$ and $\overline{UB}$ pins, which are held at logic high.

11.9 PORTH, LATH and TRISH Registers

Note: PORTH is available only on 80-pin and 100-pin devices.

PORTH is an 8-bit wide, bidirectional I/O port; it is fully implemented on 80-pin and 100-pin devices. The corresponding Data Direction register is TRISH. Setting a TRISH bit (= 1) will make the corresponding PORTH pin an input (i.e., put the corresponding output driver in a High-Impedance mode). Clearing a TRISH bit (= 0) will make the corresponding PORTH pin an output (i.e., put the contents of the output latch on the selected pin). PORTH<3:0> pins are digital only and tolerate voltages up to 5.5V.

The Output Latch register (LATH) is also memory mapped. Read-modify-write operations on the LATH register, read and write the latched output value for PORTH.

All pins on PORTH are implemented with Schmitt Trigger input buffers. Each pin is individually configurable as an input or output.

When the external memory interface is enabled, four of the PORTH pins function as the high-order address lines for the interface. The address output from the interface takes priority over other digital I/O. The corresponding TRISH bits are also overridden.

PORTH pins, RH4 through RH7, are multiplexed with analog converter inputs. The operation of these pins as analog inputs is selected by clearing or setting the PCFG<3:0> control bits in the ADCON1 register.

PORTH can also be configured as the alternate Enhanced PWM Output Channels B and C for the ECCP1 and ECCP3 modules. This is done by clearing the ECCPMX Configuration bit.

EXAMPLE 11-8: INITIALIZING PORTH

```
CLRF    PORTH       ; Initialize PORTH by
                    ; clearing output
                    ; data latches
CLRF    LATH        ; Alternate method
                    ; to clear output
                    ; data latches
MOVLW   0Fh         ; Configure PORTH as
MOVWF   ADCON1      ; digital I/O
MOVLW   0CFh        ; Value used to
                    ; initialize data
                    ; direction
MOVWF   TRISH       ; Set RH3:RH0 as inputs
                    ; RH5:RH4 as outputs
                    ; RH7:RH6 as inputs
```

17.0 CAPTURE/COMPARE/PWM (CCP) MODULES

Members of the PIC18F97J60 family of devices all have a total of five CCP (Capture/Compare/PWM) modules. Two of these (CCP4 and CCP5) implement standard Capture, Compare and Pulse-Width Modulation (PWM) modes and are discussed in this section. The other three modules (ECCP1, ECCP2, ECCP3) implement standard Capture and Compare modes, as well as Enhanced PWM modes. These are discussed in **Section 18.0 "Enhanced Capture/Compare/PWM (ECCP) Modules"**.

Each CCPx/ECCPx module contains a 16-bit register which can operate as a 16-Bit Capture register, a 16-Bit Compare register or a PWM Master/Slave Duty Cycle register. For the sake of clarity, all CCPx module operation in the following sections is described with respect to CCP4, but is equally applicable to CCP5.

Capture and Compare operations described in this chapter apply to all standard and Enhanced CCPx modules. The operations of PWM mode, described in **Section 17.4 "PWM Mode"**, apply to CCP4 and CCP5 only.

Note: Throughout this section and **Section 18.0 "Enhanced Capture/Compare/PWM (ECCP) Modules"**, references to register and bit names that may be associated with a specific CCP module are referred to generically by the use of 'x' or 'y' in place of the specific module number. Thus, "CCPxCON" might refer to the control register for ECCP1, ECCP2, ECCP3, CCP4 or CCP5.

REGISTER 17-1: CCPxCON: CCPx CONTROL REGISTER (CCP4 AND CCP5)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	DCxB1	DCxB0	CCPxM3	CCPxM2	CCPxM1	CCPxM0
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

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

bit 5-4 **DCxB<1:0>**: CCPx Module PWM Duty Cycle Bit 1 and Bit 0

Capture mode:
Unused.

Compare mode:
Unused.

PWM mode:
These bits are the two Least Significant bits (bit 1 and bit 0) of the 10-bit PWM duty cycle. The eight Most Significant bits (DCxB<9:2>) of the duty cycle are found in CCPRxL.

bit 3-0 **CCPxM<3:0>**: CCPx Module Mode Select bits

0000 = Capture/Compare/PWM disabled (resets CCPx module)
0001 = Reserved
0010 = Compare mode; toggle output on match (CCPxIF bit is set)
0011 = Reserved
0100 = Capture mode; every falling edge
0101 = Capture mode; every rising edge
0110 = Capture mode; every 4th rising edge
0111 = Capture mode; every 16th rising edge
1000 = Compare mode; initialize CCPx pin low; on compare match, force CCPx pin high (CCPxIF bit is set)
1001 = Compare mode; initialize CCPx pin high; on compare match, force CCPx pin low (CCPxIF bit is set)
1010 = Compare mode; generate software interrupt on compare match (CCPxIF bit is set, CCPx pin reflects I/O state)
1011 = Reserved
11xx = PWM mode

The CCPRxH register and a 2-bit internal latch are used to double-buffer the PWM duty cycle. This double-buffering is essential for glitchless PWM operation.

When the CCPRxH and 2-bit latch match TMR2 (TMR4), concatenated with an internal 2-bit Q clock or 2 bits of the TMR2 (TMR4) prescaler, the CCPx pin is cleared.

The maximum PWM resolution (bits) for a given PWM frequency is given by Equation 17-3:

EQUATION 17-3:

$$\text{PWM Resolution (max)} = \frac{\log\left(\frac{FOSC}{FPWM}\right)}{\log(2)} \text{ bits}$$

Note: If the PWM duty cycle value is longer than the PWM period, the CCPx pin will not be cleared.

17.4.3 SETUP FOR PWM OPERATION

The following steps should be taken when configuring the CCPx module for PWM operation:

1. Set the PWM period by writing to the PR2 (PR4) register.
2. Set the PWM duty cycle by writing to the CCPRxL register and CCPxCON<5:4> bits.
3. Make the CCPx pin an output by clearing the appropriate TRIS bit.
4. Set the TMR2 (TMR4) prescale value, then enable Timer2 (Timer4) by writing to T2CON (T4CON).
5. Configure the CCPx module for PWM operation.

TABLE 17-3: EXAMPLE PWM FREQUENCIES AND RESOLUTIONS AT 40 MHz

PWM Frequency	2.44 kHz	9.77 kHz	39.06 kHz	156.25 kHz	312.50 kHz	416.67 kHz
Timer Prescaler (1, 4, 16)	16	4	1	1	1	1
PR2 Value	FFh	FFh	FFh	3Fh	1Fh	17h
Maximum Resolution (bits)	10	10	10	8	7	6.58

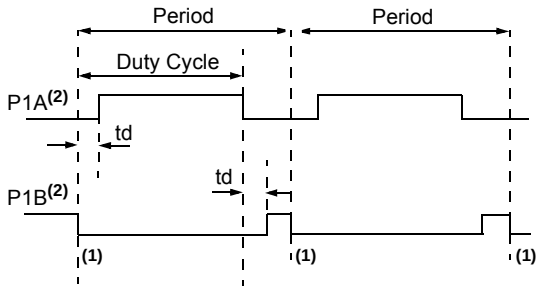
18.4.4 HALF-BRIDGE MODE

In the Half-Bridge Output mode, two pins are used as outputs to drive push-pull loads. The PWM output signal is output on the P1A pin, while the complementary PWM output signal is output on the P1B pin (Figure 18-4). This mode can be used for half-bridge applications, as shown in Figure 18-5, or for full-bridge applications, where four power switches are being modulated with two PWM signals.

In Half-Bridge Output mode, the programmable dead-band delay can be used to prevent shoot-through current in half-bridge power devices. The value of bits, P1DC<6:0>, sets the number of instruction cycles before the output is driven active. If the value is greater than the duty cycle, the corresponding output remains inactive during the entire cycle. See **Section 18.4.6 "Programmable Dead-Band Delay"** for more details on dead-band delay operations.

Since the P1A and P1B outputs are multiplexed with the PORTC<2> and PORTE<6> data latches, the TRISC<2> and TRISE<6> bits must be cleared to configure P1A and P1B as outputs.

FIGURE 18-4: HALF-BRIDGE PWM OUTPUT



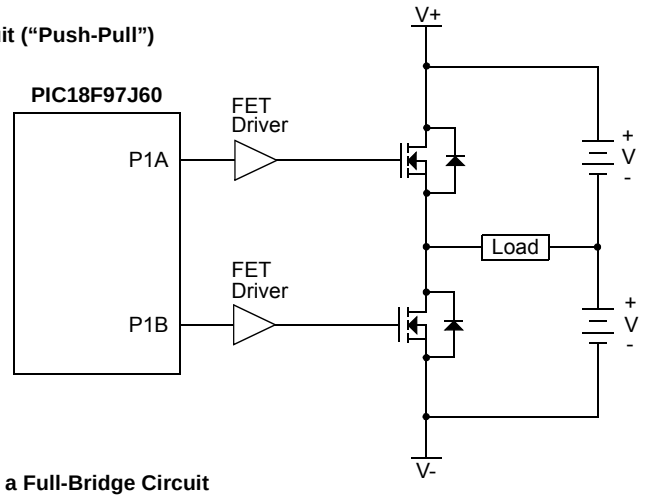
td = Dead Band Delay

Note 1: At this time, the TMR2 register is equal to the PR2 register.

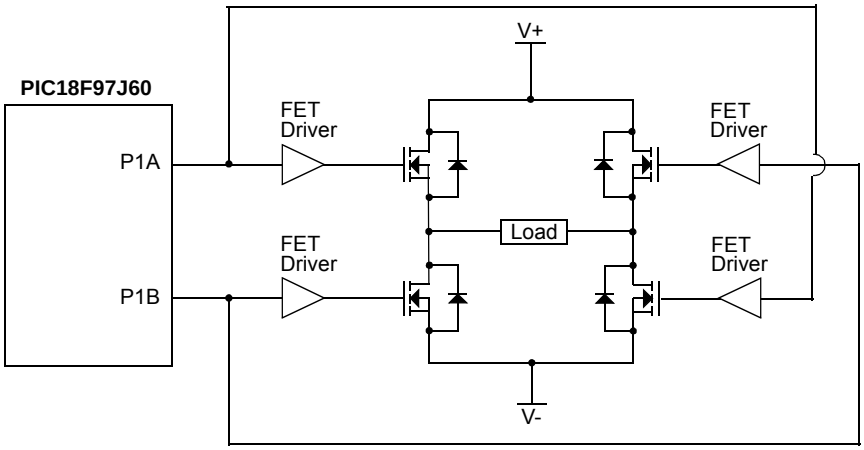
2: Output signals are shown as active-high.

FIGURE 18-5: EXAMPLES OF HALF-BRIDGE OUTPUT MODE APPLICATIONS

Standard Half-Bridge Circuit ("Push-Pull")



Half-Bridge Output Driving a Full-Bridge Circuit



20.3.2 OPERATION

When initializing the SPI, several options need to be specified. This is done by programming the appropriate control bits (SSPxCON1<5:0> and SSPxSTAT<7:6>). These control bits allow the following to be specified:

- Master mode (SCKx is the clock output)
- Slave mode (SCKx is the clock input)
- Clock Polarity (Idle state of SCKx)
- Data Input Sample Phase (middle or end of data output time)
- Clock Edge (output data on rising/falling edge of SCKx)
- Clock Rate (Master mode only)
- Slave Select mode (Slave mode only)

Each MSSP module consists of a transmit/receive shift register (SSPxSR) and a buffer register (SSPxBUF). The SSPxSR shifts the data in and out of the device, MSb first. The SSPxBUF holds the data that was written to the SSPxSR until the received data is ready. Once the 8 bits of data have been received, that byte is moved to the SSPxBUF register. Then, the Buffer Full detect bit, BF (SSPxSTAT<0>), and the interrupt flag bit, SSPxIF, are set. This double-buffering of the received data (SSPxBUF) allows the next byte to start reception before reading the data that was just received. Any write to the SSPxBUF register during transmission/reception of data will be ignored and the Write Collision detect bit, WCOL (SSPxCON1<7>), will be set. User software must clear the WCOL bit so that it can be determined if the following write(s) to the SSPxBUF register completed successfully.

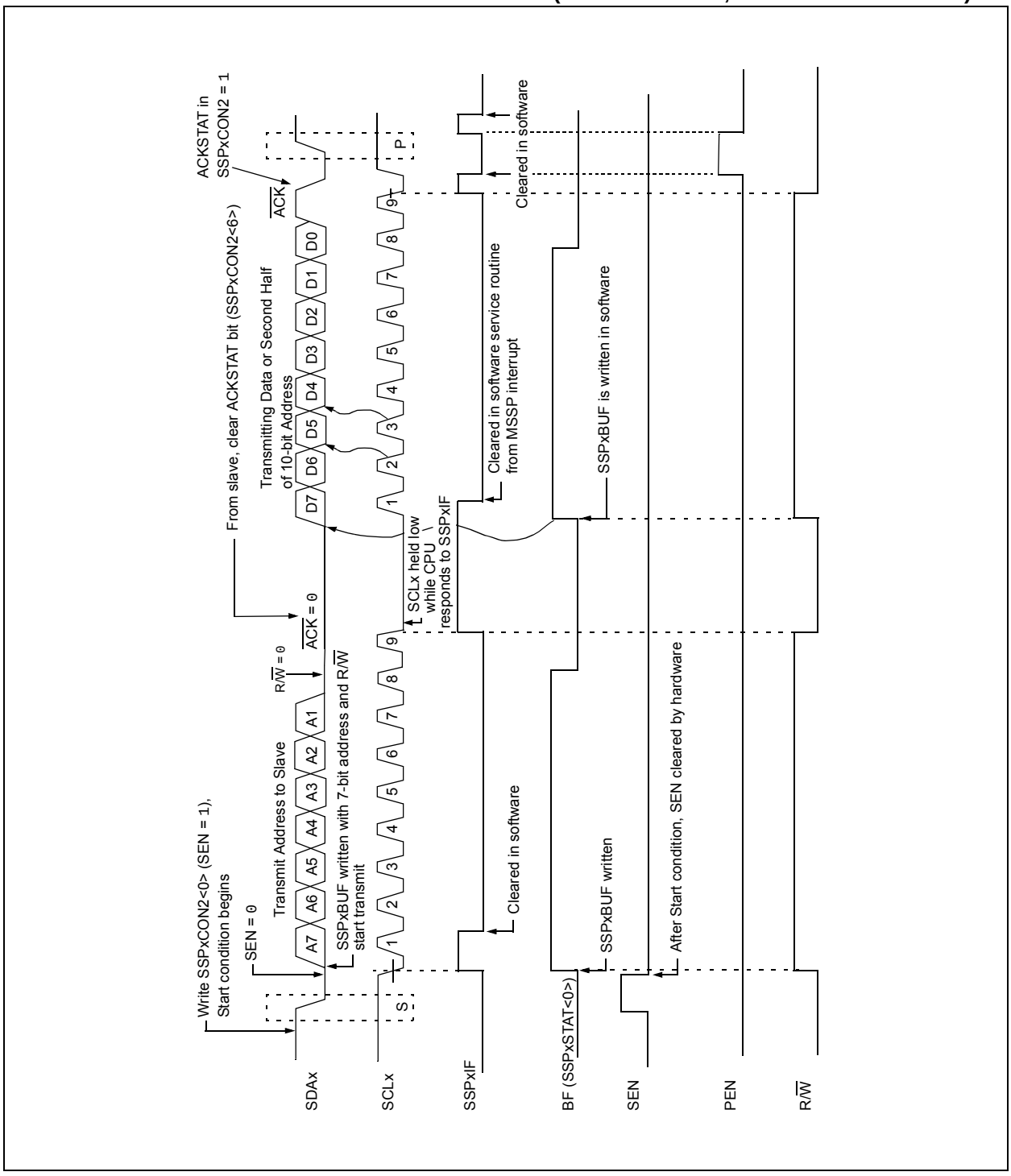
When the application software is expecting to receive valid data, the SSPxBUF should be read before the next byte of data to transfer is written to the SSPxBUF. The Buffer Full bit, BF (SSPxSTAT<0>), indicates when SSPxBUF has been loaded with the received data (transmission is complete). When the SSPxBUF is read, the BF bit is cleared. This data may be irrelevant if the SPI is only a transmitter. Generally, the MSSP interrupt is used to determine when the transmission/reception has completed. The SSPxBUF must be read and/or written. If the interrupt method is not going to be used, then software polling can be done to ensure that a write collision does not occur. Example 20-1 shows the loading of the SSP1BUF (SSP1SR) for data transmission.

The SSPxSR is not directly readable or writable and can only be accessed by addressing the SSPxBUF register. Additionally, the SSPxSTAT register indicates the various status conditions.

EXAMPLE 20-1: LOADING THE SSP1BUF (SSP1SR) REGISTER

```
LOOP    BTFSS   SSP1STAT, BF    ;Has data been received (transmit complete)?
        BRA     LOOP            ;No
        MOVF    SSP1BUF, W      ;WREG reg = contents of SSP1BUF
        MOVWF   RXDATA          ;Save in user RAM, if data is meaningful
        MOVF    TXDATA, W       ;W reg = contents of TXDATA
        MOVWF   SSP1BUF         ;New data to xmit
```

FIGURE 20-23: I²C™ MASTER MODE WAVEFORM (TRANSMISSION, 7 OR 10-BIT ADDRESS)



21.2.2 EUSARTx ASYNCHRONOUS RECEIVER

The receiver block diagram is shown in Figure 21-6. The data is received on the RXx pin and drives the data recovery block. The data recovery block is actually a high-speed shifter operating at x16 times the baud rate, whereas the main receive serial shifter operates at the bit rate or at FOSC. This mode would typically be used in RS-232 systems.

The RXDTP bit (BAUDCON<5>) allows the RXx signal to be inverted (polarity reversed). Devices that buffer signals from RS-232 to TTL levels also perform an inversion of the signal (when RS-232 = positive, TTL = 0). Inverting the polarity of the RXx pin data by setting the RXDTP bit allows for the use of circuits that provide buffering without inverting the signal.

To set up an Asynchronous Reception:

1. Initialize the SPBRGHx:SPBRGx registers for the appropriate baud rate. Set or clear the BRGH and BRG16 bits, as required, to achieve the desired baud rate.
2. Enable the asynchronous serial port by clearing the SYNC bit and setting bit, SPEN.
3. If the signal at the RXx pin is to be inverted, set the RXDTP bit.
4. If interrupts are desired, set enable bit, RCxIE.
5. If 9-bit reception is desired, set bit, RX9.
6. Enable the reception by setting bit, CREN.
7. Flag bit, RCxIF, will be set when reception is complete and an interrupt will be generated if enable bit, RCxIE, was set.
8. Read the RCSTAx register to get the 9th bit (if enabled) and determine if any error occurred during reception.
9. Read the 8-bit received data by reading the RCREGx register.
10. If any error occurred, clear the error by clearing enable bit, CREN.
11. If using interrupts, ensure that the GIE and PEIE bits in the INTCON register (INTCON<7:6>) are set.

21.2.3 SETTING UP 9-BIT MODE WITH ADDRESS DETECT

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

1. Initialize the SPBRGHx:SPBRGx registers for the appropriate baud rate. Set or clear the BRGH and BRG16 bits, as required, to achieve the desired baud rate.
2. Enable the asynchronous serial port by clearing the SYNC bit and setting the SPEN bit.
3. If the signal at the RXx pin is to be inverted, set the RXDTP bit. If the signal from the TXx pin is to be inverted, set the TXCKP bit.
4. If interrupts are required, set the RCEN bit and select the desired priority level with the RCxIP bit.
5. Set the RX9 bit to enable 9-bit reception.
6. Set the ADDEN bit to enable address detect.
7. Enable reception by setting the CREN bit.
8. The RCxIF bit will be set when reception is complete. The interrupt will be Acknowledged if the RCxIE and GIE bits are set.
9. Read the RCSTAx register to determine if any error occurred during reception, as well as read Bit 9 of data (if applicable).
10. Read RCREGx to determine if the device is being addressed.
11. If any error occurred, clear the CREN bit.
12. If the device has been addressed, clear the ADDEN bit to allow all received data into the receive buffer and interrupt the CPU.

21.4 EUSARTx Synchronous Slave Mode

Synchronous Slave mode is entered by clearing bit, CSRC (TXSTAx<7>). This mode differs from the Synchronous Master mode in that the shift clock is supplied externally at the CKx pin (instead of being supplied internally in Master mode). This allows the device to transfer or receive data while in any low-power mode.

21.4.1 EUSARTx SYNCHRONOUS SLAVE TRANSMISSION

The operation of the Synchronous Master and Slave modes is identical, except in the case of Sleep mode.

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

a) The first word will immediately transfer to the TSR register and transmit.
b) The second word will remain in the TXREGx register.
c) Flag bit, TXxIF, will not be set.
d) When the first word has been shifted out of TSR, the TXREGx register will transfer the second word to the TSR and flag bit, TXxIF, will now be set.
e) If enable bit, TXxIE, is set, the interrupt will wake the chip from Sleep. If the global interrupt is enabled, the program will branch to the interrupt vector.

To set up a Synchronous Slave Transmission:

1. Enable the synchronous slave serial port by setting bits, SYNC and SPEN, and clearing bit, CSRC.
2. Clear bits, CREN and SREN.
3. If the signal from the CKx pin is to be inverted, set the TXCKP bit. If the signal from the DTx pin is to be inverted, set the RXDTP bit.
4. If interrupts are desired, set enable bit, TXxIE.
5. If 9-bit transmission is desired, set bit, TX9.
6. Enable the transmission by setting enable bit, TXEN.
7. If 9-bit transmission is selected, the ninth bit should be loaded in bit, TX9D.
8. Start transmission by loading data to the TXREGx register.
9. If using interrupts, ensure that the GIE and PEIE bits in the INTCON register (INTCON<7:6>) are set.

TABLE 21-9: REGISTERS ASSOCIATED WITH SYNCHRONOUS SLAVE TRANSMISSION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on Page:
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	69
PIR1	PSPIF	ADIF	RC1IF	TX1IF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	71
PIE1	PSPIE	ADIE	RC1IE	TX1IE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	71
IPR1	PSPIP	ADIP	RC1IP	TX1IP	SSP1IP	CCP1IP	TMR2IP	TMR1IP	71
PIR3	SSP2IF	BCL2IF	RC2IF	TX2IF(1)	TMR4IF	CCP5IF	CCP4IF	CCP3IF	71
PIE3	SSP2IE	BCL2IE	RC2IE	TX2IE(1)	TMR4IE	CCP5IE	CCP4IE	CCP3IE	71
IPR3	SSP2IP	BCL2IP	RC2IP	TX2IP(1)	TMR4IP	CCP5IP	CCP4IP	CCP3IP	71
RCSTAx	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	71
TXREGx	EUSARTx Transmit Register								71
TXSTAx	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	71
BAUDCONx	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN	72
SPBRGHx	EUSARTx Baud Rate Generator Register High Byte								72
SPBRGx	EUSARTx Baud Rate Generator Register Low Byte								72

Legend: — = unimplemented, read as '0'. Shaded cells are not used for synchronous slave transmission.

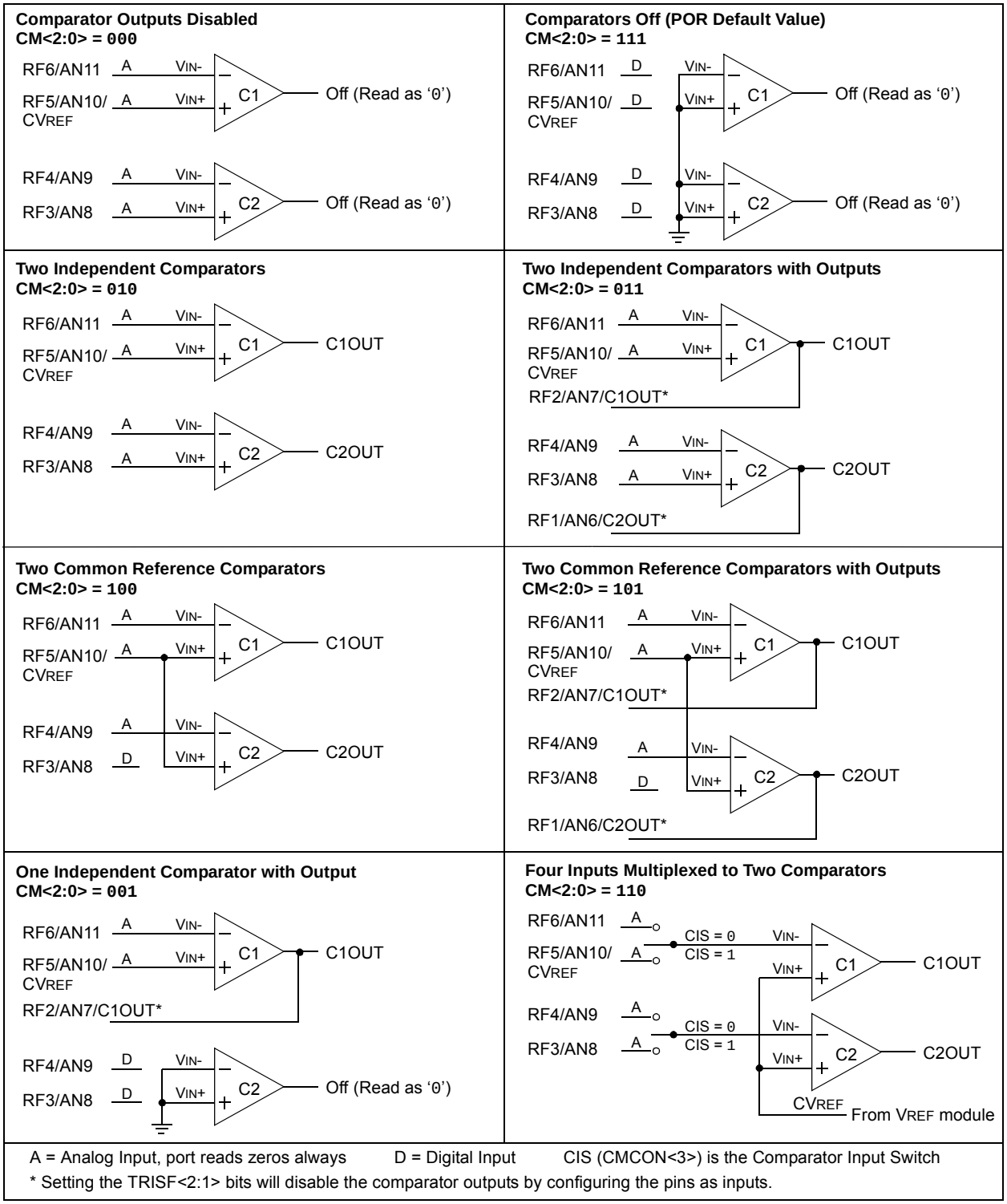
Note 1: These bits are only available in 80-pin and 100-pin devices; otherwise, they are unimplemented and read as '0'.

23.1 Comparator Configuration

There are eight modes of operation for the comparators, shown in Figure 23-1. Bits CM<2:0> of the CMCON register are used to select these modes. The TRISF register controls the data direction of the comparator pins for each mode. If the Comparator mode is changed, the comparator output level may not be valid for the specified mode change delay shown in **Section 28.0 "Electrical Characteristics"**.

Note:	Comparator interrupts should be disabled during a Comparator mode change; otherwise, a false interrupt may occur.

FIGURE 23-1: COMPARATOR I/O OPERATING MODES



A = Analog Input, port reads zeros always D = Digital Input CIS (CMCON<3>) is the Comparator Input Switch

* Setting the TRISF<2:1> bits will disable the comparator outputs by configuring the pins as inputs.

REGISTER 25-3: CONFIG2L: CONFIGURATION REGISTER 2 LOW (BYTE ADDRESS 300002h)

R/WO-1	R/WO-1	U-0	U-0	U-0	R/WO-1	R/WO-1	R/WO-1
IESO	FCMEN	—	—	—	FOSC2	FOSC1	FOSC0
bit 7							bit 0

Legend:		
R = Readable bit	WO = Write-Once bit	U = Unimplemented bit, read as ‘0’
-n = Value when device is unprogrammed	‘1’ = Bit is set	‘0’ = Bit is cleared

bit 7 **IESO:** Two-Speed Start-up (Internal/External Oscillator Switchover) Control bit
1 = Two-Speed Start-up is enabled
0 = Two-Speed Start-up is disabled

bit 6 **FCMEN:** Fail-Safe Clock Monitor Enable bit
1 = Fail-Safe Clock Monitor is enabled
0 = Fail-Safe Clock Monitor is disabled

bit 5-3 **Unimplemented:** Read as ‘0’

bit 2 **FOSC2:** Default/Reset System Clock Select bit
1 = Clock selected by FOSC<1:0> as system clock is enabled when OSCCON<1:0> = 00
0 = INTRC enabled as system clock when OSCCON<1:0> = 00

bit 1-0 **FOSC<1:0>:** Oscillator Selection bits
11 = EC oscillator, PLL is enabled and under software control, CLKO function on OSC2
10 = EC oscillator, CLKO function on OSC2
01 = HS oscillator, PLL is enabled and under software control
00 = HS oscillator

TBLWT **Table Write**

Syntax: TBLWT (*; *+; *-; +*)

Operands: None

Operation: if TBLWT*,
(TABLAT) → Holding Register;
TBLPTR – No Change
if TBLWT*+,
(TABLAT) → Holding Register;
(TBLPTR) + 1 → TBLPTR
if TBLWT*-,
(TABLAT) → Holding Register;
(TBLPTR) – 1 → TBLPTR
if TBLWT+*,
(TBLPTR) + 1 → TBLPTR;
(TABLAT) → Holding Register

Status Affected: None

Encoding:

0000	0000	0000	11nn nn=0 * =1 *+ =2 *- =3 +*

Description: This instruction uses the 3 LSBs of TBLPTR to determine which of the 8 holding registers the TABLAT is written to. The holding registers are used to program the contents of Program Memory (P.M.). (Refer to **Section 6.0 "Memory Organization"** for additional details on programming Flash memory.)

The TBLPTR (a 21-bit pointer) points to each byte in the program memory. TBLPTR has a 2-Mbyte address range. The LSb of the TBLPTR selects which byte of the program memory location to access.

TBLPTR[0] = 0: Least Significant Byte of Program Memory Word

TBLPTR[0] = 1: Most Significant Byte of Program Memory Word

The TBLWT instruction can modify the value of TBLPTR as follows:

- no change
- post-increment
- post-decrement
- pre-increment

Words: 1

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	No operation	No operation
No operation	No operation (Read TABLAT)	No operation	No operation (Write to Holding Register)

TBLWT **Table Write (Continued)**

Example 1: TBLWT*+;

Before Instruction
TABLAT = 55h
TBLPTR = 00A356h
HOLDING REGISTER (00A356h) = FFh
After Instructions (table write completion)
TABLAT = 55h
TBLPTR = 00A357h
HOLDING REGISTER (00A356h) = 55h

Example 2: TBLWT +*;

Before Instruction
TABLAT = 34h
TBLPTR = 01389Ah
HOLDING REGISTER (01389Ah) = FFh
HOLDING REGISTER (01389Bh) = FFh
After Instruction (table write completion)
TABLAT = 34h
TBLPTR = 01389Bh
HOLDING REGISTER (01389Ah) = FFh
HOLDING REGISTER (01389Bh) = 34h

26.2.2 EXTENDED INSTRUCTION SET

ADDFSR **Add Literal to FSR**

Syntax: ADDFSR f, k

Operands: $0 \le k \le 63$
$f \in [0, 1, 2]$

Operation: FSR(f) + k → FSR(f)

Status Affected: None

Encoding:

1110	1000	ffkk	kkkk

Description: The 6-bit literal 'k' is added to the contents of the FSR specified by 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process Data	Write to FSR

Example: ADDFSR 2, 23h

Before Instruction
FSR2 = 03FFh
After Instruction
FSR2 = 0422h

ADDULNK **Add Literal to FSR2 and Return**

Syntax: ADDULNK k

Operands: $0 \le k \le 63$

Operation: FSR2 + k → FSR2,
(TOS) → PC

Status Affected: None

Encoding:

1110	1000	11kk	kkkk

Description: The 6-bit literal 'k' is added to the contents of FSR2. A RETURN is then executed by loading the PC with the TOS.

The instruction takes two cycles to execute; a NOP is performed during the second cycle.

This may be thought of as a special case of the ADDFSR instruction, where f = 3 (binary '11'); it operates only on FSR2.

Words: 1

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process Data	Write to FSR
No Operation	No Operation	No Operation	No Operation

Example: ADDULNK 23h

Before Instruction
FSR2 = 03FFh
PC = 0100h
After Instruction
FSR2 = 0422h
PC = (TOS)

Note: All PIC18 instructions may take an optional label argument preceding the instruction mnemonic for use in symbolic addressing. If a label is used, the instruction format then becomes: {label} instruction argument(s).

NOTES:

NOTES: