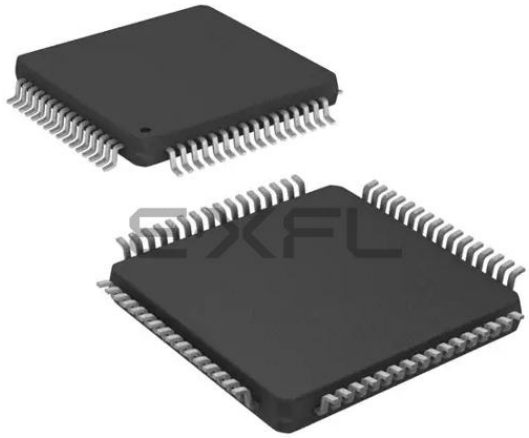


Welcome to E-XFL.COM

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"

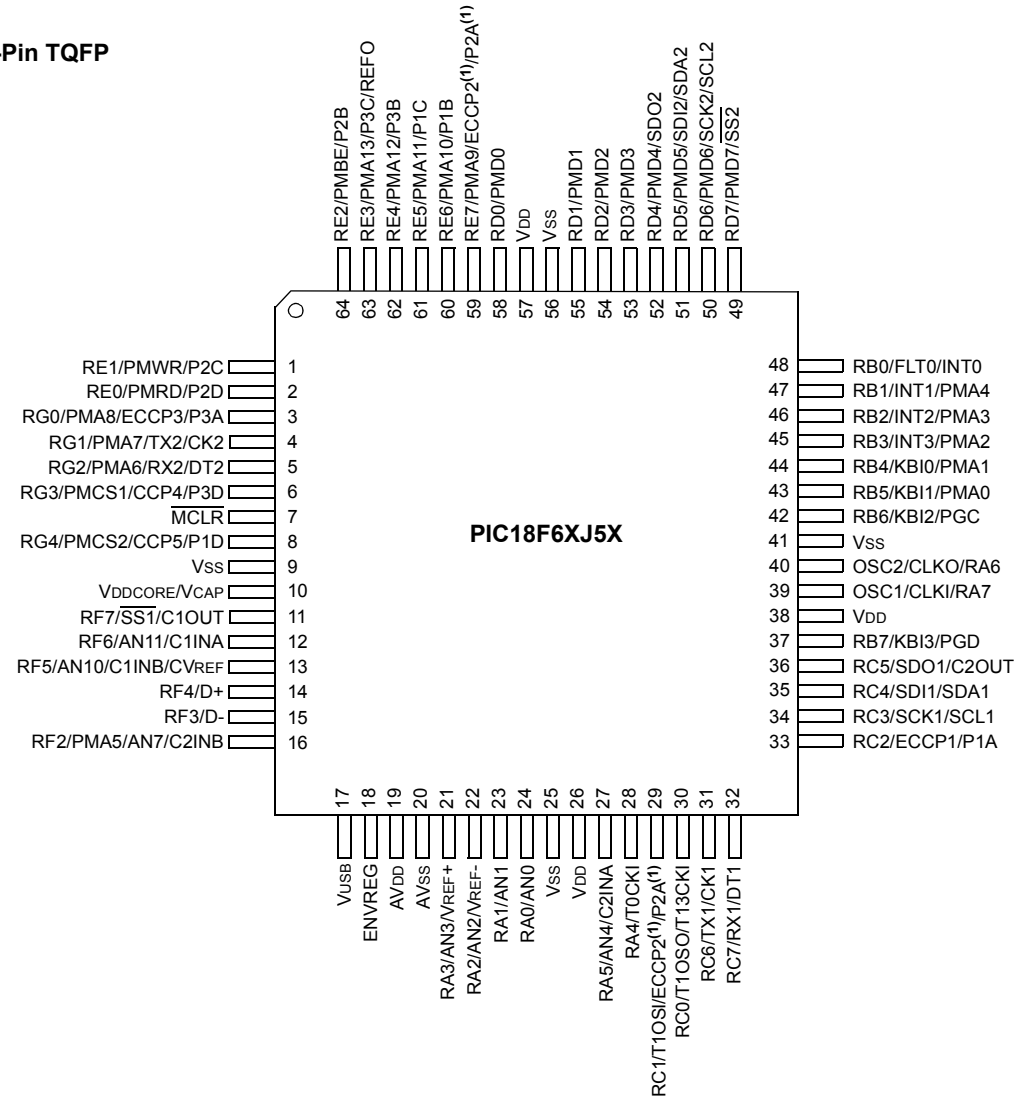
Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	48MHz
Connectivity	I²C, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	49
Program Memory Size	96KB (48K x 16)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	3.8K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f66j55t-i-pt

Device	Flash Program Memory (bytes)	SRAM Data Memory (bytes)	I/O	10-Bit A/D (ch)	CCP/ ECCP (PWM)	MSSP			EUSART	Comparators	Timers 8/16-Bit	External Bus	PMP/PSP
							SPI	Master I^2C™					
PIC18F65J50	32K	3904*	49	8	2/3	2	Y	Y	2	2	2/3	N	Y
PIC18F66J50	64K	3904*	49	8	2/3	2	Y	Y	2	2	2/3	N	Y
PIC18F66J55	96K	3904*	49	8	2/3	2	Y	Y	2	2	2/3	N	Y
PIC18F67J50	128K	3904*	49	8	2/3	2	Y	Y	2	2	2/3	N	Y
PIC18F85J50	32K	3904*	65	12	2/3	2	Y	Y	2	2	2/3	Y	Y
PIC18F86J50	64K	3904*	65	12	2/3	2	Y	Y	2	2	2/3	Y	Y
PIC18F86J55	96K	3904*	65	12	2/3	2	Y	Y	2	2	2/3	Y	Y
PIC18F87J50	128K	3904*	65	12	2/3	2	Y	Y	2	2	2/3	Y	Y

* Includes the dual access RAM used by the USB module which is shared with data memory.

Pin Diagrams

64-Pin TQFP



Note 1: The ECCP2/P2A pin placement depends on the setting of the CCP2MX Configuration bit.

TABLE 1-3: PIC18F6XJ5X PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number	Pin Type	Buffer Type	Description
	64-TQFP			
				PORTA is a bidirectional I/O port.
RA0/AN0	24			
RA0		I/O	TTL	Digital I/O.
AN0		I	Analog	Analog input 0.
RA1/AN1	23			
RA1		I/O	TTL	Digital I/O.
AN1		I	Analog	Analog input 1.
RA2/AN2/VREF-	22			
RA2		I/O	TTL	Digital I/O.
AN2		I	Analog	Analog input 2.
VREF-		I	Analog	A/D reference voltage (low) input.
RA3/AN3/VREF+	21			
RA3		I/O	TTL	Digital I/O.
AN3		I	Analog	Analog input 3.
VREF+		I	Analog	A/D reference voltage (high) input.
RA4/T0CKI	28			
RA4		I/O	ST	Digital I/O.
T0CKI		I	ST	Timer0 external clock input.
RA5/AN4/C2INA	27			
RA5		I/O	TTL	Digital I/O.
AN4		I	Analog	Analog input 4.
C2INA		—	Analog	Comparator 2 input A
RA6	—	—	—	See the OSC2/CLKO/RA6 pin.
RA7	—	—	—	See the OSC1/CLKI/RA7 pin.

Legend: TTL = TTL compatible input; CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; Analog = Analog input; I = Input; O = Output; P = Power; OD = Open-Drain (no P diode to VDD)

Note 1: Default assignment for ECCP2/P2A when CCP2MX Configuration bit is set.

2: Alternate assignment for ECCP2/P2A when CCP2MX Configuration bit is cleared.

3: RA7 and RA6 will be disabled if OSC1 and OSC2 are used for the clock function.

TABLE 5-5: REGISTER FILE SUMMARY (PIC18F87J50 FAMILY)

File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Details on Page:
TOSU	—	—	—	Top-of-Stack Upper Byte (TOS<20:16>)					---0 0000	61, 73
TOSH	Top-of-Stack High Byte (TOS<15:8>)								0000 0000	61, 73
TOSL	Top-of-Stack Low Byte (TOS<7:0>)								0000 0000	61, 73
STKPTR	STKFUL	STKUNF	—	SP4	SP3	SP2	SP1	SP0	00-0 0000	61, 74
PCLATU	—	—	bit 21(1)	Holding Register for PC<20:16>					---0 0000	61, 73
PCLATH	Holding Register for PC<15:8>								0000 0000	61, 73
PCL	PC Low Byte (PC<7:0>)								0000 0000	61, 73
TBLPTRU	—	—	bit 21	Program Memory Table Pointer Upper Byte (TBLPTR<20:16>)					--00 0000	61, 106
TBLPTRH	Program Memory Table Pointer High Byte (TBLPTR<15:8>)								0000 0000	61, 106
TBLPTRL	Program Memory Table Pointer Low Byte (TBLPTR<7:0>)								0000 0000	61, 106
TABLAT	Program Memory Table Latch								0000 0000	61, 106
PRODH	Product Register High Byte								xxxx xxxx	61, 119
PRODL	Product Register Low Byte								xxxx xxxx	61, 119
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 000x	61, 123
INTCON2	$\overline{\text{RBPU}}$	INTEDG0	INTEDG1	INTEDG2	INTEDG3	TMR0IP	INT3IP	RBIP	1111 1111	61, 123
INTCON3	INT2IP	INT1IP	INT3IE	INT2IE	INT1IE	INT3IF	INT2IF	INT1IF	1100 0000	61, 123
INDF0	Uses contents of FSR0 to address data memory – value of FSR0 not changed (not a physical register)								N/A	61, 91
POSTINC0	Uses contents of FSR0 to address data memory – value of FSR0 post-incremented (not a physical register)								N/A	61, 92
POSTDEC0	Uses contents of FSR0 to address data memory – value of FSR0 post-decremented (not a physical register)								N/A	61, 92
PREINC0	Uses contents of FSR0 to address data memory – value of FSR0 pre-incremented (not a physical register)								N/A	61, 92
PLUSW0	Uses contents of FSR0 to address data memory – value of FSR0 pre-incremented (not a physical register) – value of FSR0 offset by W								N/A	61, 92
FSR0H	—	—	—	—	Indirect Data Memory Address Pointer 0 High Byte				---- xxxx	61, 91
FSR0L	Indirect Data Memory Address Pointer 0 Low Byte								xxxx xxxx	61, 91
WREG	Working Register								xxxx xxxx	61, 75
INDF1	Uses contents of FSR1 to address data memory – value of FSR1 not changed (not a physical register)								N/A	61, 91
POSTINC1	Uses contents of FSR1 to address data memory – value of FSR1 post-incremented (not a physical register)								N/A	61, 92
POSTDEC1	Uses contents of FSR1 to address data memory – value of FSR1 post-decremented (not a physical register)								N/A	61, 92
PREINC1	Uses contents of FSR1 to address data memory – value of FSR1 pre-incremented (not a physical register)								N/A	61, 92
PLUSW1	Uses contents of FSR1 to address data memory – value of FSR1 pre-incremented (not a physical register) – value of FSR1 offset by W								N/A	61, 92
FSR1H	—	—	—	—	Indirect Data Memory Address Pointer 1 High Byte				---- xxxx	61, 91
FSR1L	Indirect Data Memory Address Pointer 1 Low Byte								xxxx xxxx	61, 91
BSR	—	—	—	—	Bank Select Register				---- 0000	61, 78
INDF2	Uses contents of FSR2 to address data memory – value of FSR2 not changed (not a physical register)								N/A	62, 91
POSTINC2	Uses contents of FSR2 to address data memory – value of FSR2 post-incremented (not a physical register)								N/A	62, 92
POSTDEC2	Uses contents of FSR2 to address data memory – value of FSR2 post-decremented (not a physical register)								N/A	62, 92
PREINC2	Uses contents of FSR2 to address data memory – value of FSR2 pre-incremented (not a physical register)								N/A	62, 92
PLUSW2	Uses contents of FSR2 to address data memory – value of FSR2 pre-incremented (not a physical register) – value of FSR2 offset by W								N/A	62, 92
FSR2H	—	—	—	—	Indirect Data Memory Address Pointer 2 High Byte				---- xxxx	62, 91

Legend: x = unknown, u = unchanged, - = unimplemented, q = value depends on condition. **Bold** indicates shared-access SFRs.

Note 1: Bit 21 of the PC is only available in Serial Programming modes.

2: Default (legacy) SFR at this address, available when WDTCON<4> = 0.

3: Configuration SFR, overlaps with default SFR at this address; available only when WDTCON<4> = 1.

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

5: The SSPxMSK registers are only accessible when SSPxCON2<3:0> = 1001.

6: Alternate names and definitions for these bits when the MSSP module is operating in I^2C™ Slave mode. See **Section 19.4.3.2 "Address Masking Modes"** for details

7: These bits and/or registers are only available in 80-pin devices; otherwise, they are unimplemented and read as '0'. Reset values are shown for 80-pin devices.

8: The PMADDRH/PMDOUT1H and PMADDRL/PMDOUT1L register pairs share the physical registers and addresses, but have different functions determined by the module's operating mode. See **Section 11.1.2 "Data Registers"** for more information.

TABLE 5-5: REGISTER FILE SUMMARY (PIC18F87J50 FAMILY) (CONTINUED)

File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Details on Page:
ECCP1AS	ECCP1ASE	ECCP1AS2	ECCP1AS1	ECCP1AS0	PSS1AC1	PSS1AC0	PSS1BD1	PSS1BD0	0000 0000	63, 232
ECCP1DEL	P1RSEN	P1DC6	P1DC5	P1DC4	P1DC3	P1DC2	P1DC1	P1DC0	0000 0000	63, 232
CCPR1H	Capture/Compare/PWM Register 1 HIgh Byte								xxxx xxxx	63, 232
CCPR1L	Capture/Compare/PWM Register 1 Low Byte								xxxx xxxx	63, 232
CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	63, 232
ECCP2AS	ECCP2ASE	ECCP2AS2	ECCP2AS1	ECCP2AS0	PSS2AC1	PSS2AC0	PSS2BD1	PSS2BD0	0000 0000	63, 232
ECCP2DEL	P2RSEN	P2DC6	P2DC5	P2DC4	P2DC3	P2DC2	P2DC1	P2DC0	0000 0000	63, 232
CCPR2H	Capture/Compare/PWM Register 2 High Byte								xxxx xxxx	63, 232
CCPR2L	Capture/Compare/PWM Register 2 Low Byte								xxxx xxxx	63, 232
CCP2CON	P2M1	P2M0	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	0000 0000	63, 232
ECCP3AS	ECCP3ASE	ECCP3AS2	ECCP3AS1	ECCP3AS0	PSS3AC1	PSS3AC0	PSS3BD1	PSS3BD0	0000 0000	63, 232
ECCP3DEL	P3RSEN	P3DC6	P3DC5	P3DC4	P3DC3	P3DC2	P3DC1	P3DC0	0000 0000	63, 232
CCPR3H	Capture/Compare/PWM Register 3 High Byte								xxxx xxxx	63, 232
CCPR3L	Capture/Compare/PWM Register 3 Low Byte								xxxx xxxx	63, 232
CCP3CON	P3M1	P3M0	DC3B1	DC3B0	CCP3M3	CCP3M2	CCP3M1	CCP3M0	0000 0000	63, 232
SPBRG1	EUSART1 Baud Rate Generator Register Low Byte								0000 0000	63, 283
RCREG1	EUSART1 Receive Register								0000 0000	63, 291, 292
TXREG1	EUSART1 Transmit Register								xxxx xxxx	63, 289, 290
TXSTA1	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	0000 0010	63, 289
RCSTA1	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	63, 291
SPBRG2	EUSART2 Baud Rate Generator Register Low Byte								0000 0000	63, 283
RCREG2	EUSART2 Receive Register								0000 0000	63, 291, 292
TXREG2	EUSART2 Transmit Register								0000 0000	63, 289, 290
TXSTA2	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	0000 0010	63, 289
EECON2	Program Memory Control Register 2 (not a physical register)								---- ----	63, 98
EECON1	—	—	WPROG	FREE	WRERR	WREN	WR	—	--00 x00-	63, 98
IPR3	SSP2IP	BCL2IP	RC2IP	TX2IP	TMR4IP	CCP5IP	CCP4IP	CCP3IP	1111 1111	64, 132
PIR3	SSP2IF	BCL2IF	RC2IF	TX2IF	TMR4IF	CCP5IF	CCP4IF	CCP3IF	0000 0000	64, 126
PIE3	SSP2IE	BCL2IE	RC2IE	TX2IE	TMR4IE	CCP5IE	CCP4IE	CCP3IE	0000 0000	64, 129
IPR2	OSCFIP	CM2IP	CM1IP	USBIP	BCL1IP	LVDIP	TMR3IP	CCP2IP	1111 1111	64, 132
PIR2	OSCFIF	CM2IF	CM1IF	USBIF	BCL1IF	LVDIF	TMR3IF	CCP2IF	0000 0000	64, 126
PIE2	OSCFIE	CM2IE	CM1IE	USBIE	BCL1IE	LVDIE	TMR3IE	CCP2IE	0000 0000	64, 129
IPR1	PMPIP	ADIP	RC1IP	TX1IP	SSP1IP	CCP1IP	TMR2IP	TMR1IP	1111 1111	64, 132
PIR1	PMPIF	ADIF	RC1IF	TX1IF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	0000 0000	64, 126
PIE1	PMPIE	ADIE	RC1IE	TX1IE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	0000 0000	64, 129
RCSTA2	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	64, 291
OSCTUNE	INTSRC	PLLEN	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0	0000 0000	64, 39

Legend: x = unknown, u = unchanged, - = unimplemented, q = value depends on condition. **Bold** indicates shared-access SFRs.

Note 1: Bit 21 of the PC is only available in Serial Programming modes.

2: Default (legacy) SFR at this address, available when WDTCON<4> = 0.

3: Configuration SFR, overlaps with default SFR at this address; available only when WDTCON<4> = 1.

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

5: The SSPxMSK registers are only accessible when SSPxCON2<3:0> = 1001.

6: Alternate names and definitions for these bits when the MSSP module is operating in I^2C™ Slave mode. See **Section 19.4.3.2 "Address Masking Modes"** for details

7: These bits and/or registers are only available in 80-pin devices; otherwise, they are unimplemented and read as '0'. Reset values are shown for 80-pin devices.

8: The PMADDRH/PMDOUT1H and PMADDRL/PMDOUT1L register pairs share the physical registers and addresses, but have different functions determined by the module's operating mode. See **Section 11.1.2 "Data Registers"** for more information.

5.3.6 STATUS REGISTER

The STATUS register, shown in Register 5-4, contains the arithmetic status of the ALU. The STATUS register can be the operand for any instruction, as with any other register. If the STATUS register is the destination for an instruction that affects the Z, DC, C, OV or N bits, then the write to these five bits is disabled.

These bits are set or cleared according to the device logic. Therefore, the result of an instruction with the STATUS register as destination may be different than intended. For example, `CLRF STATUS` will set the Z bit but leave the other bits unchanged. The STATUS register then reads back as '`000u u1uu`'. It is recommended, therefore, that only `BCF`, `BSF`, `SWAPF`, `MOVFF` and `MOVWF` instructions are used to alter the STATUS register because these instructions do not affect the Z, C, DC, OV or N bits in the STATUS register.

For other instructions not affecting any Status bits, see the instruction set summaries in Table 26-2 and Table 26-3.

Note:	The C and DC bits operate as a $\overline{\text{borrow}}$ and $\overline{\text{digit borrow}}$ bit respectively, in subtraction.

REGISTER 5-4: STATUS REGISTER

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	N	OV	Z	DC(1)	C(2)
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 7-5 **Unimplemented:** Read as '0'

bit 4 **N:** Negative bit

This bit is used for signed arithmetic (2's complement). It indicates whether the result was negative (ALU MSB = 1).

1 = Result was negative
0 = Result was positive

bit 3 **OV:** Overflow bit

This bit is used for signed arithmetic (2's complement). It indicates an overflow of the 7-bit magnitude which causes the sign bit (bit 7) to change state.

1 = Overflow occurred for signed arithmetic (in this arithmetic operation)
0 = No overflow occurred

bit 2 **Z:** Zero bit

1 = The result of an arithmetic or logic operation is zero
0 = The result of an arithmetic or logic operation is not zero

bit 1 **DC:** Digit carry/$\overline{\text{borrow}}$ bit(1)

For `ADDWF`, `ADDLW`, `SUBLW` and `SUBWF` instructions:

1 = A carry-out from the 4th low-order bit of the result occurred
0 = No carry-out from the 4th low-order bit of the result

bit 0 **C:** Carry/$\overline{\text{borrow}}$ bit(2)

For `ADDWF`, `ADDLW`, `SUBLW` and `SUBWF` instructions:

1 = A carry-out from the Most Significant bit of the result occurred
0 = No carry-out from the Most Significant bit of the result occurred

Note 1: For $\overline{\text{borrow}}$, the polarity is reversed. A subtraction is executed by adding the 2's complement of the second operand. For rotate (`RRF`, `RLF`) instructions, this bit is loaded with either bit 4 or bit 3 of the source register.

2: For $\overline{\text{borrow}}$, the polarity is reversed. A subtraction is executed by adding the 2's complement of the second operand. For rotate (`RRF`, `RLF`) instructions, this bit is loaded with either the high or low-order bit of the source register.

REGISTER 6-1: EECON1: EEPROM CONTROL REGISTER 1

U-0	U-0	R/W-0	R/W-0	R/W-x	R/W-0	R/S-0	U-0
—	—	WPROG	FREE	WRERR(1)	WREN	WR	—
bit 7							bit 0

Legend:	S = Settable only bit (cannot be cleared in software)		
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 **WPROG:** One Word-Wide Program bit
1 = Program 2 bytes on the next WR command
0 = Program 64 bytes on the next WR command

bit 4 **FREE:** Flash Row Erase Enable bit
1 = Erase the program memory row addressed by TBLPTR on the next WR command (cleared by completion of erase operation)
0 = Perform write only

bit 3 **WRERR:** Flash Program Error Flag bit(1)
1 = A write operation is prematurely terminated (any Reset during self-timed programming in normal operation, or an improper write attempt)
0 = The write operation completed

bit 2 **WREN:** Flash Program Write Enable bit
1 = Allows write cycles to Flash program memory
0 = Inhibits write cycles to Flash program memory

bit 1 **WR:** Write Control bit
1 = Initiates a program memory erase cycle or write cycle (The operation is self-timed and the bit is cleared by hardware once write is complete. The WR bit can only be set (not cleared) in software.)
0 = Write cycle is complete

bit 0 **Unimplemented:** Read as '0'

Note 1: When a WRERR occurs, the EEPGD and CFGS bits are not cleared. This allows tracing of the error condition.

7.7.1 8-BIT MODE TIMING

The presentation of control signals on the External Memory Bus is different for the various operating modes. Typical signal timing diagrams are shown in Figure 7-7 and Figure 7-8.

FIGURE 7-7: EXTERNAL MEMORY BUS TIMING FOR TBLRD (EXTENDED MICROCONTROLLER MODE)

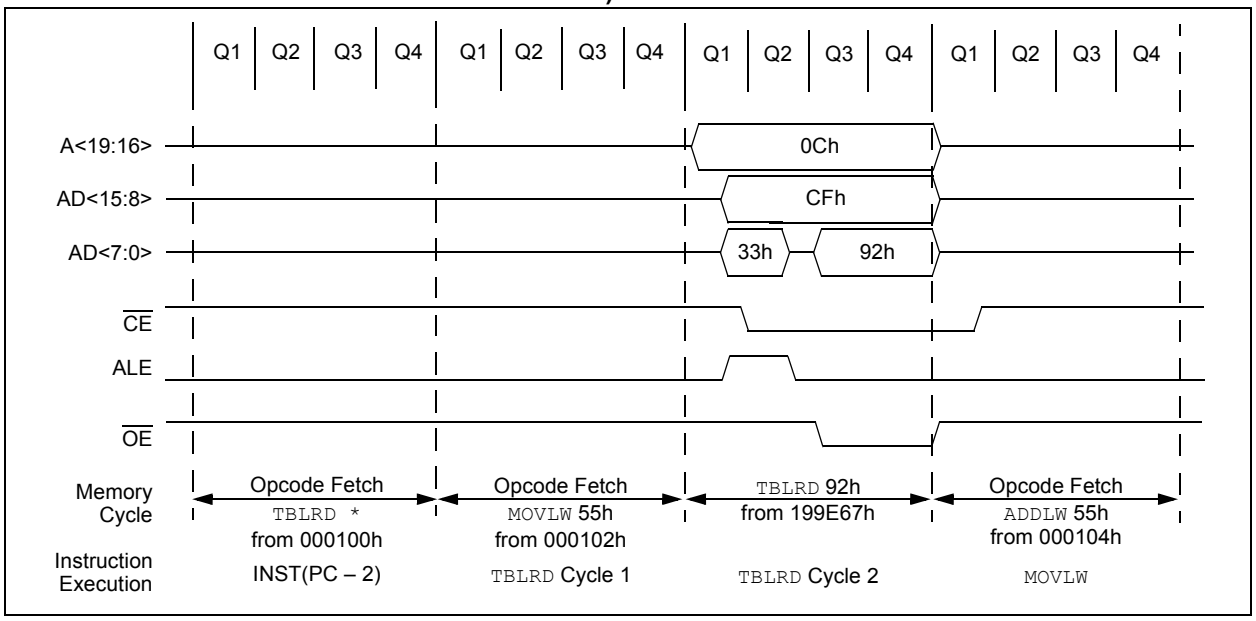


FIGURE 7-8: EXTERNAL MEMORY BUS TIMING FOR SLEEP (EXTENDED MICROCONTROLLER MODE)

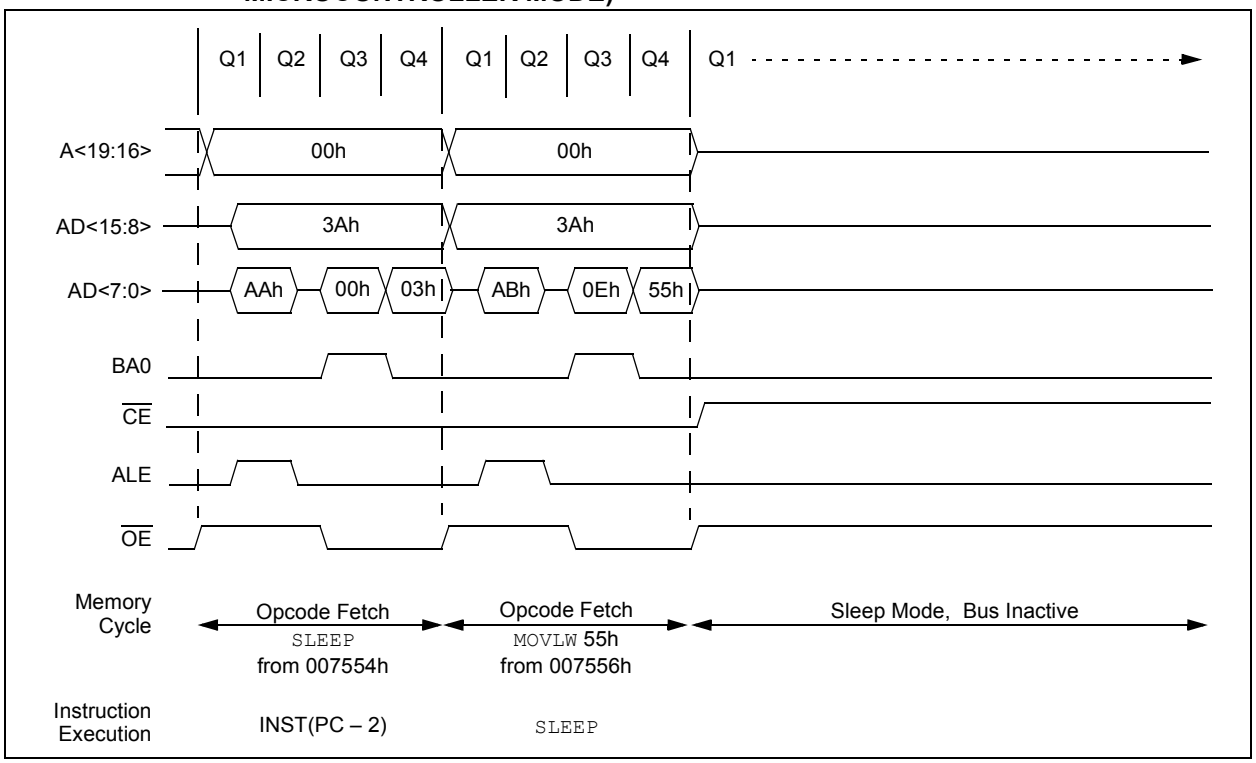
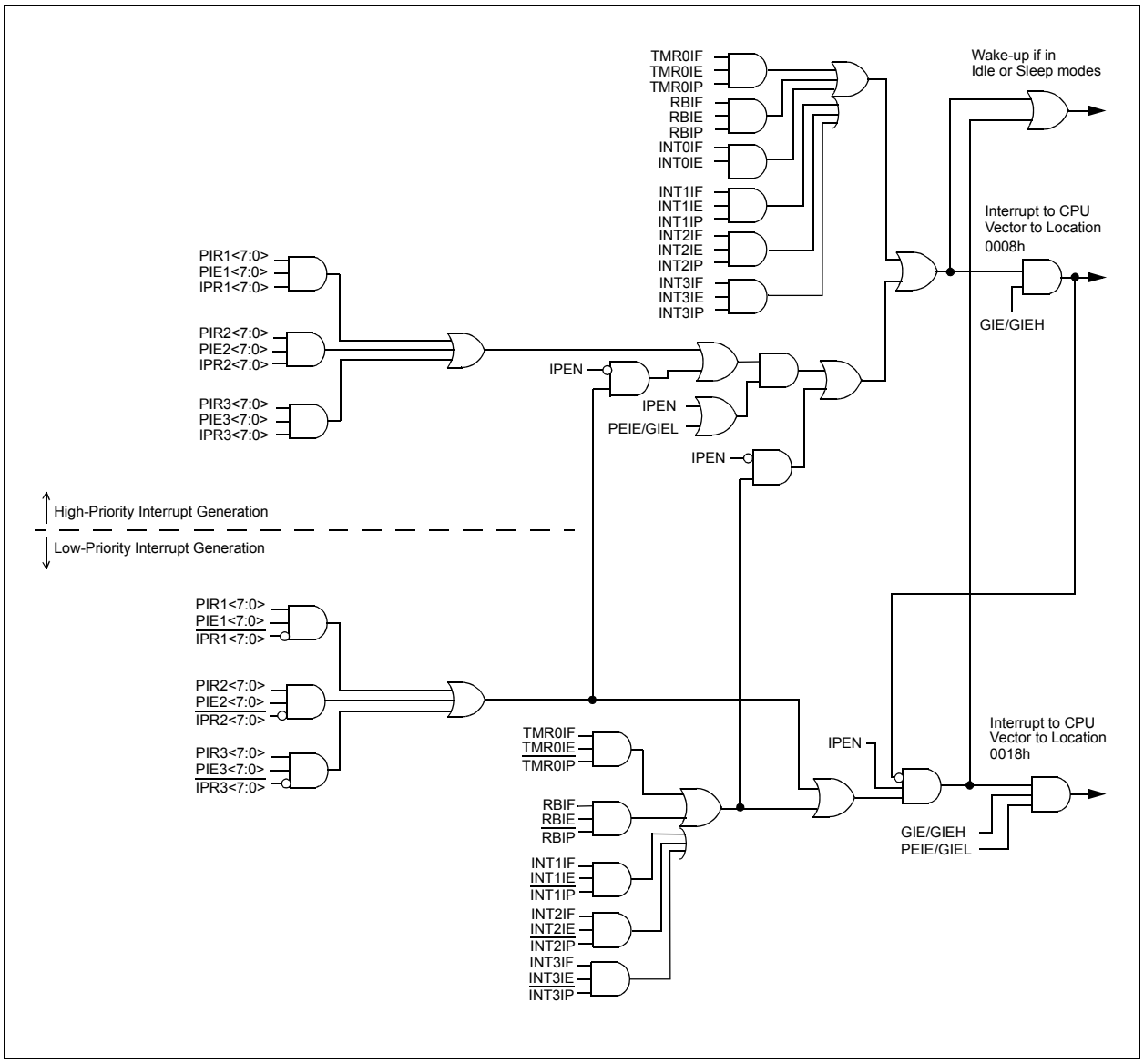


FIGURE 9-1: PIC18F87J50 FAMILY INTERRUPT LOGIC



REGISTER 11-4: PMMODEL: PARALLEL PORT MODE REGISTER LOW BYTE

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WAITB1(1)	WAITB0(1)	WAITM3	WAITM2	WAITM1	WAITM0	WAITE1(1)	WAITE0(1)
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 **WAITB1:WAITB0:** Data Setup to Read/Write Wait State Configuration bits(1)
11 = Data wait of 4 TCY; multiplexed address phase of 4 TCY
10 = Data wait of 3 TCY; multiplexed address phase of 3 TCY
01 = Data wait of 2 TCY; multiplexed address phase of 2 TCY
00 = Data wait of 1 TCY; multiplexed address phase of 1 TCY

bit 5-2 **WAITM3:WAITM0:** Read to Byte Enable Strobe Wait State Configuration bits
1111 = Wait of additional 15 TCY
...
0001 = Wait of additional 1 TCY
0000 = No additional wait cycles (operation forced into one TCY)

bit 1-0 **WAITE1:WAITE0:** Data Hold After Strobe Wait State Configuration bits(1)
11 = Wait of 4 TCY
10 = Wait of 3 TCY
01 = Wait of 2 TCY
00 = Wait of 1 TCY

Note 1: WAITB and WAITE bits are ignored whenever WAITM3:WAITM0 = 0000.

REGISTER 11-5: PMEH: PARALLEL PORT ENABLE REGISTER HIGH BYTE

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PTEN15	PTEN14	PTEN13	PTEN12	PTEN11	PTEN10	PTEN9	PTEN8
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 **PTEN15:PTEN14:** PMCSx Strobe Enable bits
1 = PMA15 and PMA14 function as either PMA<15:14> or PMCS2 and PMCS1
0 = PMA15 and PMA14 function as port I/O

bit 5-0 **PTEN13:PTEN8:** PMP Address Port Enable bits
1 = PMA<13:8> function as PMP address lines
0 = PMA<13:8> function as port I/O

18.0 ENHANCED CAPTURE/ COMPARE/PWM (ECCP) MODULE

In the PIC18F87J10 family of devices, three of the CCP modules are implemented as standard CCP modules with Enhanced PWM capabilities. These include the provision for 2 or 4 output channels, user-selectable polarity, dead-band control and automatic shutdown and restart. The Enhanced features are discussed in detail in **Section 18.4 "Enhanced PWM Mode"**. Capture, Compare and single-output PWM functions of the ECCP module are the same as described for the standard CCP module.

The control register for the Enhanced CCP module is shown in Register 18-1. It differs from the CCP4CON/ CCP5CON registers in that the two Most Significant bits are implemented to control PWM functionality.

In addition to the expanded range of modes available through the Enhanced CCPxCON register, the ECCP modules each have two additional registers associated with Enhanced PWM operation and auto-shutdown features. They are:

- ECCPxDEL (ECCPx PWM Delay)
- ECCPxAS (ECCPx Auto-Shutdown Control)

REGISTER 18-1: CCPxCON: ECCPx CONTROL REGISTER (ECCP1/ECCP2/ECCP3)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PxM1	PxM0	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 **PxM1:PxM0:** Enhanced PWM Output Configuration bits

If CCPxM3:CCPxM2 = 00, 01, 10:

xx = PxA assigned as Capture/Compare input/output; PxB, PxC, PxD assigned as port pins

If CCPxM3:CCPxM2 = 11:

00 = Single output: PxA modulated; PxB, PxC, PxD assigned as port pins

01 = Full-bridge output forward: P1D modulated; P1A active; P1B, P1C inactive

10 = Half-bridge output: P1A, P1B modulated with dead-band control; P1C, P1D assigned as port pins

11 = Full-bridge output reverse: P1B modulated; P1C active; P1A, P1D inactive

bit 5-4 **DCxB1:DCxB0**: PWM Duty Cycle bit 1 and bit 0

Capture mode:

Unused.

Compare mode:

Unused.

PWM mode:

These bits are the two LSbs of the 10-bit PWM duty cycle. The eight MSbs of the duty cycle are found in CCPRxL.

bit 3-0 **CCPxM3:CCPxM0**: ECCPx Module Mode Select bits

0000 = Capture/Compare/PWM off (resets ECCPx module)
0001 = Reserved
0010 = Compare mode: toggle output on match
0011 = Capture mode
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 ECCPx pin low; set output on compare match (set CCPxIF)
1001 = Compare mode: initialize ECCPx pin high; clear output on compare match (set CCPxIF)
1010 = Compare mode: generate software interrupt only; ECCPx pin reverts to I/O state
1011 = Compare mode: trigger special event (ECCPx resets TMR1 or TMR3, sets CCPxIF bit, ECCP2 trigger also starts A/D conversion if A/D module is enabled)(1)
1100 = PWM mode: PxA, PxC active-high; PxB, PxD active-high
1101 = PWM mode: PxA, PxC active-high; PxB, PxD active-low
1110 = PWM mode: PxA, PxC active-low; PxB, PxD active-high
1111 = PWM mode: PxA, PxC active-low; PxB, PxD active-low

Note 1: Implemented only for ECCP1 and ECCP2; same as '1010' for ECCP3.

19.3.7 SLAVE MODE

In Slave mode, the data is transmitted and received as the external clock pulses appear on SCKx. When the last bit is latched, the SSPxIF interrupt flag bit is set.

While in Slave mode, the external clock is supplied by the external clock source on the SCKx pin. This external clock must meet the minimum high and low times as specified in the electrical specifications.

While in Sleep mode, the slave can transmit/receive data. When a byte is received, the device can be configured to wake-up from Sleep.

19.3.8 SLAVE SELECT SYNCHRONIZATION

The $\overline{SSx}$ pin allows a Synchronous Slave mode. The SPI must be in Slave mode with the $\overline{SSx}$ pin control enabled (SSPxCON1<3:0> = 04h). When the $\overline{SSx}$ pin is low, transmission and reception are enabled and the SDOx pin is driven. When the $\overline{SSx}$ pin goes high, the SDOx pin is no longer driven, even if in the middle of a transmitted byte and becomes a floating output. External pull-up/pull-down resistors may be desirable depending on the application.

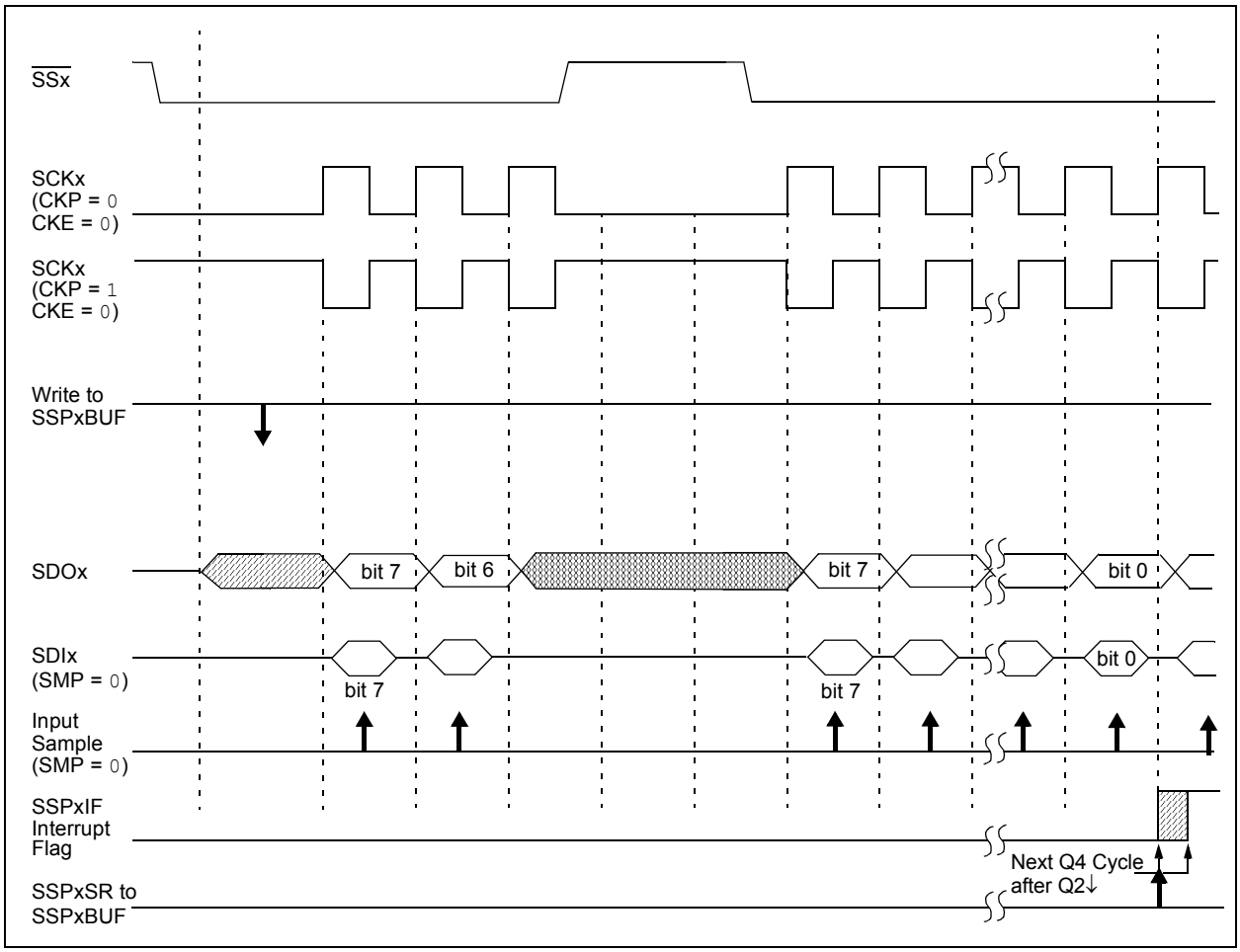
Note 1: When the SPI is in Slave mode with $\overline{SSx}$ pin control enabled (SSPxCON1<3:0> = 0100), the SPI module will reset if the $\overline{SSx}$ pin is set to VDD.

2: If the SPI is used in Slave mode with CKE set, then the $\overline{SSx}$ pin control must be enabled.

When the SPI module resets, the bit counter is forced to '0'. This can be done by either forcing the $\overline{SSx}$ pin to a high level or clearing the SSPEN bit.

To emulate two-wire communication, the SDOx pin can be connected to the SDIx pin. When the SPI needs to operate as a receiver, the SDOx pin can be configured as an input. This disables transmissions from the SDOx. The SDIx can always be left as an input (SDIx function) since it cannot create a bus conflict.

FIGURE 19-4: SLAVE SYNCHRONIZATION WAVEFORM



19.4.3.5 Reception

When the R/$\overline{W}$ bit of the address byte is clear and an address match occurs, the R/$\overline{W}$ bit of the SSPxSTAT register is cleared. The received address is loaded into the SSPxBUF register and the SDAx line is held low ($\overline{ACK}$).

When the address byte overflow condition exists, then the no Acknowledge ($\overline{ACK}$) pulse is given. An overflow condition is defined as either bit, BF (SSPxSTAT<0>), is set or bit, SSPOV (SSPxCON1<6>), is set.

An MSSP interrupt is generated for each data transfer byte. The interrupt flag bit, SSPxIF, must be cleared in software. The SSPxSTAT register is used to determine the status of the byte.

If SEN is enabled (SSPxCON2<0> = 1), SCLx will be held low (clock stretch) following each data transfer. The clock must be released by setting bit, CKP (SSPxCON1<4>). See **Section 19.4.4 "Clock Stretching"** for more details.

19.4.3.6 Transmission

When the R/$\overline{W}$ bit of the incoming address byte is set and an address match occurs, the R/$\overline{W}$ bit of the SSPxSTAT register is set. The received address is loaded into the SSPxBUF register. The $\overline{ACK}$ pulse will be sent on the ninth bit and pin SCLx is held low regardless of SEN (see **Section 19.4.4 "Clock Stretching"** for more details). By stretching the clock, the master will be unable to assert another clock pulse until the slave is done preparing the transmit data. The transmit data must be loaded into the SSPxBUF register which also loads the SSPxSR register. Then, pin SCLx should be enabled by setting bit, CKP (SSPxCON1<4>). The eight data bits are shifted out on the falling edge of the SCLx input. This ensures that the SDAx signal is valid during the SCLx high time (Figure 19-10).

The $\overline{ACK}$ pulse from the master-receiver is latched on the rising edge of the ninth SCLx input pulse. If the SDAx line is high (not $\overline{ACK}$), then the data transfer is complete. In this case, when the $\overline{ACK}$ is latched by the slave, the slave logic is reset (resets the SSPxSTAT register) and the slave monitors for another occurrence of the Start bit. If the SDAx line was low ($\overline{ACK}$), the next transmit data must be loaded into the SSPxBUF register. Again, pin SCLx must be enabled by setting bit, CKP.

An MSSP interrupt is generated for each data transfer byte. The SSPxIF bit must be cleared in software and the SSPxSTAT register is used to determine the status of the byte. The SSPxIF bit is set on the falling edge of the ninth clock pulse.

19.4.6.1 I²C Master Mode Operation

The master device generates all of the serial clock pulses and the Start and Stop conditions. A transfer is ended with a Stop condition or with a Repeated Start condition. Since the Repeated Start condition is also the beginning of the next serial transfer, the I²C bus will not be released.

In Master Transmitter mode, serial data is output through SDAx while SCLx outputs the serial clock. The first byte transmitted contains the slave address of the receiving device (7 bits) and the Read/Write (R/$\overline{W}$) bit. In this case, the R/$\overline{W}$ bit will be logic '0'. Serial data is transmitted 8 bits at a time. After each byte is transmitted, an Acknowledge bit is received. Start and Stop conditions are output to indicate the beginning and the end of a serial transfer.

In Master Receive mode, the first byte transmitted contains the slave address of the transmitting device (7 bits) and the R/$\overline{W}$ bit. In this case, the R/$\overline{W}$ bit will be logic '1'. Thus, the first byte transmitted is a 7-bit slave address, followed by a '1' to indicate the receive bit. Serial data is received via SDAx, while SCLx outputs the serial clock. Serial data is received 8 bits at a time. After each byte is received, an Acknowledge bit is transmitted. Start and Stop conditions indicate the beginning and end of transmission.

The Baud Rate Generator, used for the SPI mode operation, is used to set the SCLx clock frequency for either 100 kHz, 400 kHz or 1 MHz I²C operation. See **Section 19.4.7 "Baud Rate"** for more details.

A typical transmit sequence would go as follows:

1. The user generates a Start condition by setting the Start Enable bit, SEN (SSPxCON2<0>).
2. SSPxIF is set. The MSSP module will wait the required start time before any other operation takes place.
3. The user loads the SSPxBUF with the slave address to transmit.
4. Address is shifted out the SDAx pin until all 8 bits are transmitted.
5. The MSSP module shifts in the ACK bit from the slave device and writes its value into the SSPxCON2 register (SSPxCON2<6>).
6. The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSPxIF bit.
7. The user loads the SSPxBUF with eight bits of data.
8. Data is shifted out the SDAx pin until all 8 bits are transmitted.
9. The MSSP module shifts in the ACK bit from the slave device and writes its value into the SSPxCON2 register (SSPxCON2<6>).
10. The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSPxIF bit.
11. The user generates a Stop condition by setting the Stop Enable bit, PEN (SSPxCON2<2>).
12. Interrupt is generated once the Stop condition is complete.

22.5.2 USB INTERRUPT ENABLE REGISTER (UIE)

The USB Interrupt Enable register (Register 22-8) contains the enable bits for the USB status interrupt sources. Setting any of these bits will enable the respective interrupt source in the UIR register.

The values in this register only affect the propagation of an interrupt condition to the microcontroller's interrupt logic. The flag bits are still set by their interrupt conditions, allowing them to be polled and serviced without actually generating an interrupt.

REGISTER 22-8: UIE: USB INTERRUPT ENABLE REGISTER

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	SOFIE	STALLIE	IDLEIE	TRNIE	ACTVIE	UERRIE	URSTIE
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 **Unimplemented:** Read as '0'

bit 6 **SOFIE:** Start-of-Frame Token Interrupt Enable bit
1 = Start-of-Frame token interrupt enabled
0 = Start-of-Frame token interrupt disabled

bit 5 **STALLIE:** STALL Handshake Interrupt Enable bit
1 = STALL interrupt enabled
0 = STALL interrupt disabled

bit 4 **IDLEIE:** Idle Detect Interrupt Enable bit
1 = Idle detect interrupt enabled
0 = Idle detect interrupt disabled

bit 3 **TRNIE:** Transaction Complete Interrupt Enable bit
1 = Transaction interrupt enabled
0 = Transaction interrupt disabled

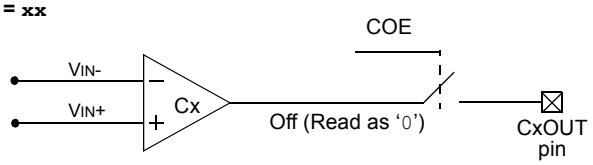
bit 2 **ACTVIE:** Bus Activity Detect Interrupt Enable bit
1 = Bus activity detect interrupt enabled
0 = Bus activity detect interrupt disabled

bit 1 **UERRIE:** USB Error Interrupt Enable bit
1 = USB error interrupt enabled
0 = USB error interrupt disabled

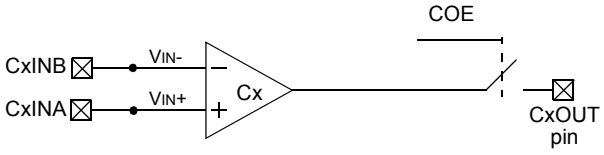
bit 0 **URSTIE:** USB Reset Interrupt Enable bit
1 = USB Reset interrupt enabled
0 = USB Reset interrupt disabled

FIGURE 23-4: COMPARATOR CONFIGURATIONS

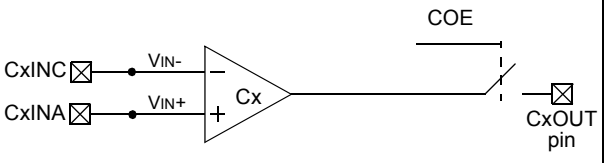
Comparator Off
CON = 0, CREF = x, CCH<1:0> = xx



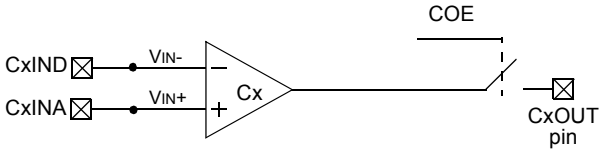
Comparator CxINB > CxINA Compare
CON = 1, CREF = 0, CCH<1:0> = 00



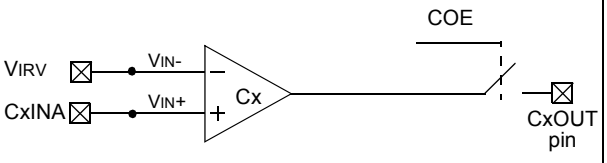
Comparator CxINC > CxINA Compare
CON = 1, CREF = 0, CCH<1:0> = 01



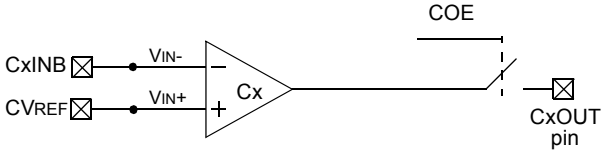
Comparator CxIND > CxINA Compare
CON = 1, CREF = 0, CCH<1:0> = 10



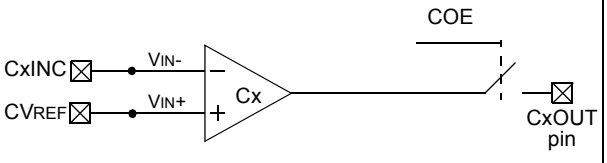
Comparator VIRV > CxINA Compare
CON = 1, CREF = 0, CCH<1:0> = 11



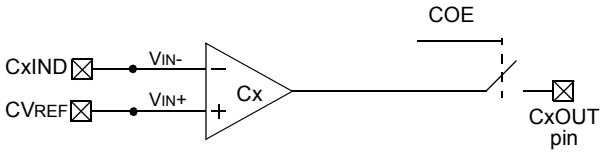
Comparator CxINB > CVREF Compare
CON = 1, CREF = 1, CCH<1:0> = 00



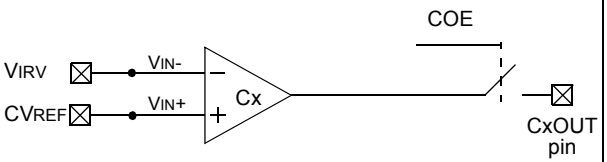
Comparator CxINC > CVREF Compare
CON = 1, CREF = 1, CCH<1:0> = 01



Comparator CxIND > CVREF Compare
CON = 1, CREF = 1, CCH<1:0> = 10



Comparator VIRV > CVREF Compare
CON = 1, CREF = 1, CCH<1:0> = 11



Note: VIRV is the Internal Reference Voltage (see Table 28-2).

23.7 Comparator Operation During Sleep

When a comparator is active and the device is placed in Sleep mode, the comparator remains active and the interrupt is functional, if enabled. This interrupt will wake-up the device from Sleep mode, when enabled. Each operational comparator will consume additional current. To minimize power consumption while in Sleep mode, turn off the comparators (CON = 0) before entering Sleep. If the device wakes up from Sleep, the contents of the CMxCON register are not affected.

23.8 Effects of a Reset

A device Reset forces the CMxCON registers to their Reset state. This forces both comparators and the voltage reference to the OFF state.

TABLE 23-3: REGISTERS ASSOCIATED WITH COMPARATOR MODULE

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	61
PIR2	OSCFIF	CM2IF	CM1IF	USBIF	BCL1IF	LVDIF	TMR3IF	CCP2IF	64
PIE2	OSCFIE	CM2IE	CM1IE	USBIE	BCL1IE	LVDIE	TMR3IE	CCP2IE	64
IPR2	OSCFIP	CM2IP	CM1IP	USBIP	BCL1IP	LVDIP	TMR3IP	CCP2IP	64
CMxCON	CON	COE	CPOL	EVPOL1	EVPOL0	CREF	CCH1	CCH0	62
CVRCON(1)	CVREN	CVROE	CVRR	CVRSS	CVR3	CVR2	CVR1	CVR0	65
CMSTAT	—	—	—	—	—	—	COUT2	COUT1	65
ANCON1(1)	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	—	—	63
ANCON0(1)	PCFG7	—	—	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	63
PORTA	—	—	RA5	RA4	RA3	RA2	RA1	—	65
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	—	64
LATA	—	—	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0	64
PORTC	RC7	RC6	RC5	RC4	RC3	RFC2	RFC1	RFC0	65
LATC	LATC7	LATC6	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	64
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	64
PORTF	RF7	RF6	RF5	RF4	RF3	RF2	—	—	65
LATF	LATF7	LATF6	LATF5	LATF4	LATF3	LATF2	—	—	64
TRISF	TRISF7	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2	—	—	64
PORTH(2)	RH7	RH6	RH5	RH4	RH3	RH2	RH1	RH0	65
TRISH(2)	TRISH7	TRISH6	TRISH5	TRISH4	TRISH3	TRISH2	TRISH1	TRISH0	64

Legend: — = unimplemented, read as '0'. Shaded cells are not used for A/D conversion.

Note 1: Configuration SFR, overlaps with default SFR at this address; available only when WDTCON<4> = 1.

2: This register is not implemented on 64-pin devices.

REGISTER 25-1: CONFIG1L: CONFIGURATION REGISTER 1 LOW (BYTE ADDRESS 300000h)

R/WO-1	R/WO-1	R/WO-1	U-0	R/WO-1	R/WO-1	R/WO-1	R/WO-1
$\overline{\text{DEBUG}}$	XINST	STVREN	—	PLLDIV2	PLLDIV1	PLLDIV0	WDTEN
bit 7							bit 0

Legend:			
R = Readable bit	WO = Write-Once 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 **$\overline{\text{DEBUG}}$:** Background Debugger Enable bit
1 = Background debugger disabled; RB6 and RB7 configured as general purpose I/O pins
0 = Background debugger enabled; RB6 and RB7 are dedicated to In-Circuit Debug

bit 6 **XINST:** Extended Instruction Set Enable bit
1 = Instruction set extension and Indexed Addressing mode enabled
0 = Instruction set extension and Indexed Addressing mode disabled (Legacy mode)

bit 5 **STVREN**: Stack Overflow/Underflow Reset Enable bit
1 = Reset on stack overflow/underflow enabled
0 = Reset on stack overflow/underflow disabled

bit 4 **Unimplemented:** Read as '0'

bit 3-1 **PLLDIV2:PLLDIV0:** Oscillator Selection bits
Divider must be selected to provide a 4 MHz input into the 96 MHz PLL
111 = No divide - oscillator used directly (4 MHz input)
110 = Oscillator divided by 2 (8 MHz input)
101 = Oscillator divided by 3 (12 MHz input)
100 = Oscillator divided by 4 (16 MHz input)
011 = Oscillator divided by 5 (20 MHz input)
010 = Oscillator divided by 6 (24 MHz input)
001 = Oscillator divided by 10 (40 MHz input)
000 = Oscillator divided by 12 (48 MHz input)

bit 0 **WDTEN:** Watchdog Timer Enable bit
1 = WDT enabled
0 = WDT disabled (control is placed on SWDTEN bit)

BTG **Bit Toggle f**

Syntax: BTG f, b {,a}

Operands: $0 \le f \le 255$
$0 \le b < 7$
$a \in [0,1]$

Operation: $(\overline{f<b>}) \rightarrow f<b>$

Status Affected: None

Encoding:

0111	bbba	ffff	ffff

Description: Bit 'b' in data memory location 'f' is inverted.

If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank (default).

If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever $f \le 95$ (5Fh). See **Section 26.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write register 'f'

Example: `BTG PORTC, 4, 0`

Before Instruction:
PORTC = 0111 0101 [75h]
After Instruction:
PORTC = 0110 0101 [65h]

BOV **Branch if Overflow**

Syntax: BOV n

Operands: $-128 \le n \le 127$

Operation: if Overflow bit is '1',
$(PC) + 2 + 2n \rightarrow PC$

Status Affected: None

Encoding:

1110	0100	nnnn	nnnn

Description: If the Overflow bit is '1', then the program will branch.

The 2's complement number '2n' is added to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is then a two-cycle instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:
If Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	Write to PC
No operation	No operation	No operation	No operation

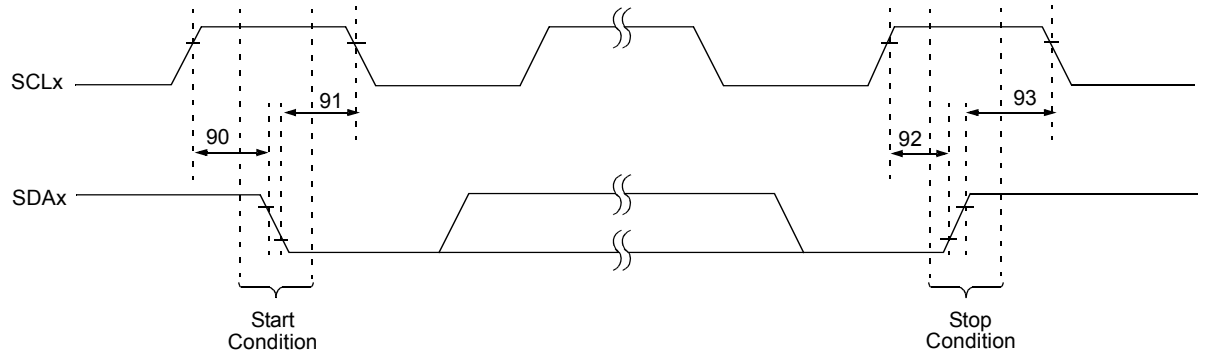
If No Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	No operation

Example: `HERE BOV Jump`

Before Instruction
PC = address (HERE)
After Instruction
If Overflow = 1;
PC = address (Jump)
If Overflow = 0;
PC = address (HERE + 2)

FIGURE 28-19: MSSPx I²C™ BUS START/STOP BITS TIMING WAVEFORMS



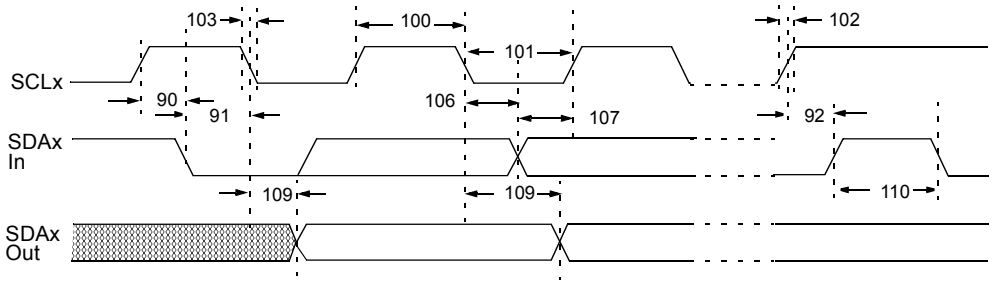
Note: Refer to Figure 28-3 for load conditions.

TABLE 28-24: MSSPx I²C™ BUS START/STOP BITS REQUIREMENTS

Param. No.	Symbol	Characteristic		Min	Max	Units	Conditions
90	TSU:STA	Start Condition Setup Time	100 kHz mode	2(TOSC)(BRG + 1)	—	ns	Only relevant for Repeated Start condition
			400 kHz mode	2(TOSC)(BRG + 1)	—		
			1 MHz mode(1)	2(TOSC)(BRG + 1)	—		
91	THD:STA	Start Condition Hold Time	100 kHz mode	2(TOSC)(BRG + 1)	—	ns	After this period, the first clock pulse is generated
			400 kHz mode	2(TOSC)(BRG + 1)	—		
			1 MHz mode(1)	2(TOSC)(BRG + 1)	—		
92	TSU:STO	Stop Condition Setup Time	100 kHz mode	2(TOSC)(BRG + 1)	—	ns	
			400 kHz mode	2(TOSC)(BRG + 1)	—		
			1 MHz mode(1)	2(TOSC)(BRG + 1)	—		
93	THD:STO	Stop Condition Hold Time	100 kHz mode	2(TOSC)(BRG + 1)	—	ns	
			400 kHz mode	2(TOSC)(BRG + 1)	—		
			1 MHz mode(1)	2(TOSC)(BRG + 1)	—		

Note 1: Maximum pin capacitance = 10 pF for all I²C™ pins.

FIGURE 28-20: MSSPx I²C™ BUS DATA TIMING



Note: Refer to Figure 28-3 for load conditions.

INDEX

A

B