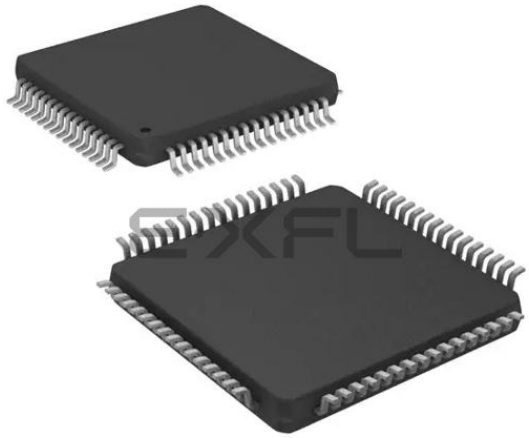
Microchip Technology - PIC18LC658T-I/PT Datasheet



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

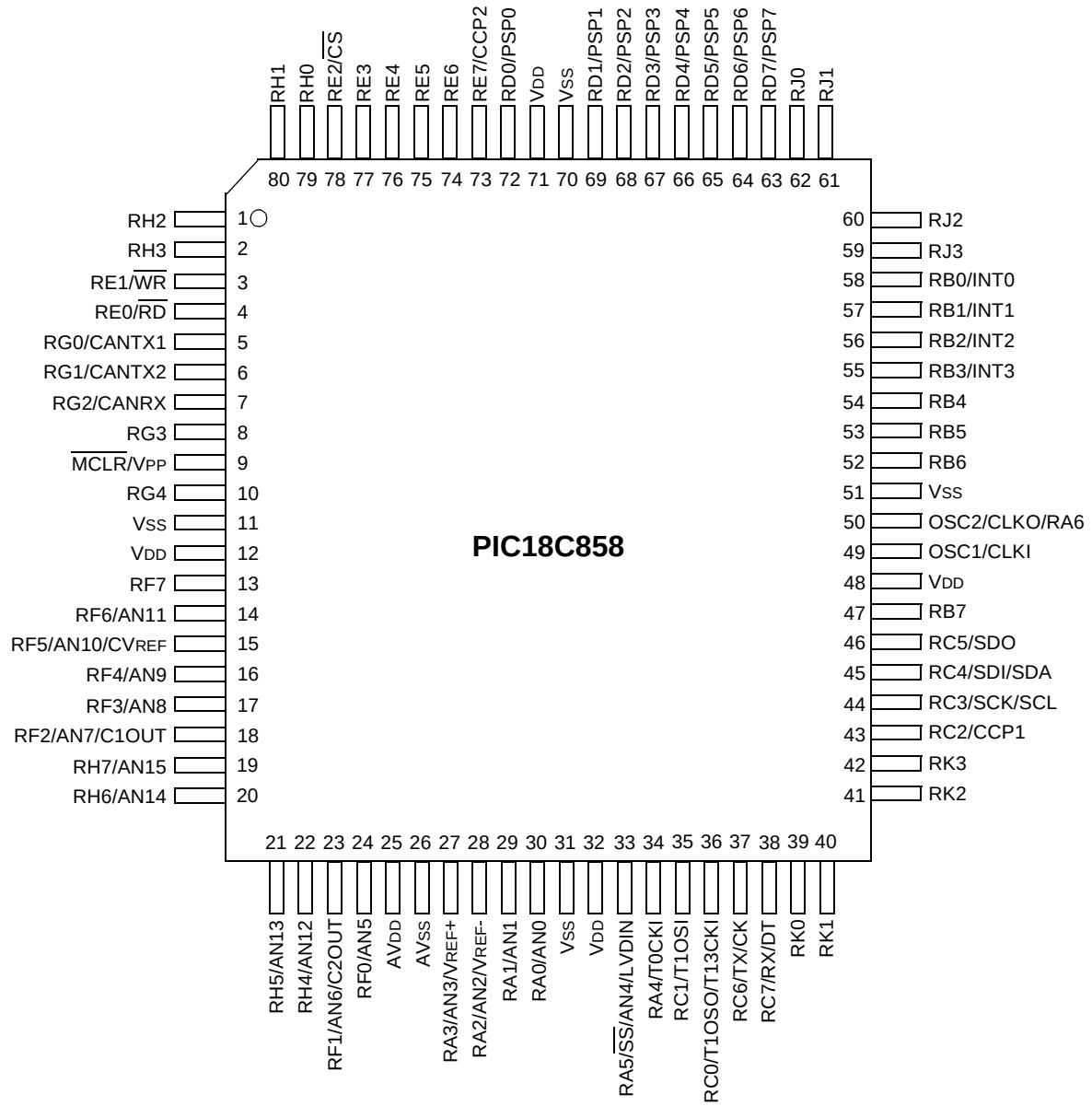
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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	Obsolete
Core Processor	PIC
Core Size	8-Bit
Speed	40MHz
Connectivity	CANbus, I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	52
Program Memory Size	32KB (16K x 16)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	1.5K x 8
Voltage - Supply (Vcc/Vdd)	2.5V ~ 5.5V
Data Converters	A/D 12x10b
Oscillator Type	External
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/pic18lc658t-i-pt

Pin Diagrams (Cont.'d)

80-Pin TQFP



 © 2000 Microchip Technology Inc.

3.0 RESET

The PIC18CXX8 differentiates between various kinds of RESET:

a) Power-on Reset (POR)
b) MCLR Reset during normal operation
c) MCLR Reset during SLEEP
d) Watchdog Timer (WDT) Reset (during normal operation)
e) Programmable Brown-out Reset (PBOR)
f) RESET Instruction
g) Stack Full Reset
h) Stack Underflow Reset

Most registers are unaffected by a RESET. Their status is unknown on POR and unchanged by all other RESETs. The other registers are forced to a "RESET" state on Power-on Reset, MCLR, WDT Reset, Brown-out Reset, MCLR Reset during SLEEP and by the RESET instruction.

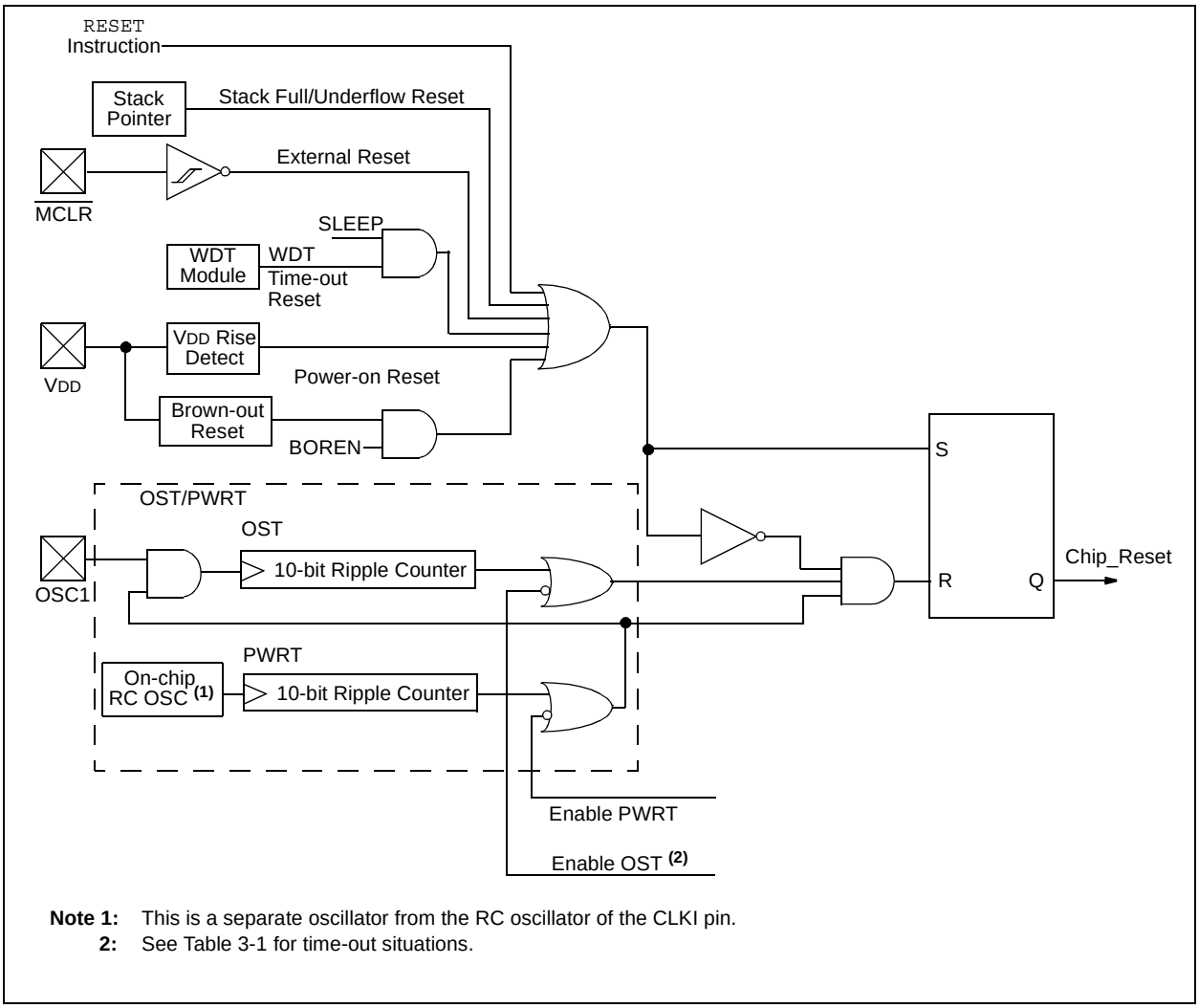
Most registers are not affected by a WDT wake-up, since this is viewed as the resumption of normal operation. Status bits from the RCON register, RI, TO, PD, POR and BOR are set or cleared differently in different RESET situations, as indicated in Table 3-2. These bits are used in software to determine the nature of the RESET. See Table 3-3 for a full description of the RESET states of all registers.

A simplified block diagram of the on-chip RESET circuit is shown in Figure 3-1.

The Enhanced MCU devices have a MCLR noise filter in the MCLR Reset path. The filter will detect and ignore small pulses.

A WDT Reset does not drive MCLR pin low.

FIGURE 3-1: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



Note 1: This is a separate oscillator from the RC oscillator of the CLKI pin.
2: See Table 3-1 for time-out situations.

4.9 Data Memory Organization

The data memory is implemented as static RAM. Each register in the data memory has a 12-bit address, allowing up to 4096 bytes of data memory. Figure 4-4 shows the data memory organization for the PIC18CXX8 devices.

The data memory map is divided into as many as 16 banks that contain 256 bytes each. The lower 4 bits of the Bank Select Register (BSR<3:0>) select which bank will be accessed. The upper 4 bits for the BSR are not implemented.

The data memory contains Special Function Registers (SFR) and General Purpose Registers (GPR). The SFR's are used for control and status of the controller and peripheral functions, while GPR's are used for data storage and scratch pad operations in the user's application. The SFR's start at the last location of Bank 15 (0xFFF) and grow downwards. GPR's start at the first location of Bank 0 and grow upwards. Any read of an unimplemented location will read as '0's.

The entire data memory may be accessed directly or indirectly. Direct addressing may require the use of the BSR register. Indirect addressing requires the use of the File Select Register (FSR). Each FSR holds a 12-bit address value that can be used to access any location in the Data Memory map without banking.

The instruction set and architecture allow operations across all banks. This may be accomplished by indirect addressing or by the use of the `MOVFF` instruction. The `MOVFF` instruction is a two word/two cycle instruction that moves a value from one register to another.

To ensure that commonly used registers (SFR's and select GPR's) can be accessed in a single cycle, regardless of the current BSR values, an Access Bank is implemented. A segment of Bank 0 and a segment of Bank 15 comprise the Access RAM. Section 4.10 provides a detailed description of the Access RAM.

4.9.1 GENERAL PURPOSE REGISTER FILE

The register file can be accessed either directly or indirectly. Indirect addressing operates through the File Select Registers (FSR). The operation of indirect addressing is shown in Section 4.12.

Enhanced MCU devices may have banked memory in the GPR area. GPR's are not initialized by a Power-on Reset and are unchanged on all other RESETS.

Data RAM is available for use as GPR registers by all instructions. Bank 15 (0xF00 to 0xFFF) contains SFR's. All other banks of data memory contain GPR registers starting with bank 0.

4.9.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers (SFR's) are registers used by the CPU and Peripheral Modules for controlling the desired operation of the device. These registers are implemented as static RAM. A list of these registers is given in Table 4-2.

The SFR's can be classified into two sets: those associated with the "core" function and those related to the peripheral functions. Those registers related to the "core" are described in this section, while those related to the operation of the peripheral features are described in the section of that peripheral feature.

The SFR's are typically distributed among the peripherals whose functions they control.

The unused SFR locations will be unimplemented and read as '0's. See Table 4-2 for addresses for the SFR's.

 © 2000 Microchip Technology Inc.

Filename	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other RESETS(3)
LATJ(4)	Read PORTJ Data Latch, Write PORTJ Data Latch								xxxx xxxx	uuuu uuuu
LATH(4)	Read PORTH Data Latch, Write PORTH Data Latch								xxxx xxxx	uuuu uuuu
LATG	—	—	—	Read PORTG Data Latch, Write PORTG Data Latch					---x xxxx	---u uuuu
LATF	Read PORTF Data Latch, Write PORTF Data Latch								xxxx xxxx	uuuu uuuu
LATE	Read PORTE Data Latch, Write PORTE Data Latch								xxxx xxxx	uuuu uuuu
LATD	Read PORTD Data Latch, Write PORTD Data Latch								xxxx xxxx	uuuu uuuu
LATC	Read PORTC Data Latch, Write PORTC Data Latch								xxxx xxxx	uuuu uuuu
LATB	Read PORTB Data Latch, Write PORTB Data Latch								xxxx xxxx	uuuu uuuu
LATA	—	Bit 6(1)	Read PORTA Data Latch, Write PORTA Data Latch						--xx xxxx	--uu uuuu
PORTJ(4)	Read PORTJ pins, Write PORTJ Data Latch								xxxx xxxx	uuuu uuuu
PORTH(4)	Read PORTH pins, Write PORTH Data Latch								xxxx xxxx	uuuu uuuu
PORTG	—	—	—	Read PORTG pins, Write PORTG Data Latch					---x xxxx	uuuu uuuu
PORTF	Read PORTF pins, Write PORTF Data Latch								0000 0000	0000 0000
PORTE	Read PORTE pins, Write PORTE Data Latch								xxxx xxxx	uuuu uuuu
PORTD	Read PORTD pins, Write PORTD Data Latch								xxxx xxxx	uuuu uuuu
PORTC	Read PORTC pins, Write PORTC Data Latch								xxxx xxxx	uuuu uuuu
PORTB	Read PORTB pins, Write PORTB Data Latch								xxxx xxxx	uuuu uuuu
PORTA	—	Bit 6(1)	Read PORTA pins, Write PORTA Data Latch						--0x 0000	--0u 0000
TRISK(4)	Data Direction Control Register for PORTK								1111 1111	1111 1111
LATK(4)	Read PORTK Data Latch, Write PORTK Data Latch								xxxx xxxx	uuuu uuuu
PORTK(4)	Read PORTK pins, Write PORTK Data Latch								xxxx xxxx	uuuu uuuu
TXERRCNT	TEC7	TEC6	TEC5	TEC4	TEC3	TEC2	TEC1	TEC0	0000 0000	0000 0000
RXERRCNT	REC7	REC6	REC5	REC4	REC3	REC2	REC1	REC0	0000 0000	0000 0000
COMSTAT	RXB0OVFL	RXB1OVFL	TXBO	TXBP	RXBP	TXWARN	RXWARN	EWARN	0000 0000	0000 0000
CIOCON	TX1SRC	TX1EN	ENDRHI	CANCAP	—	—	—	—	1000 ----	1000 ----
BRGCON3	—	WAKFIL	—	—	—	SEG2PH2	SEG2PH1	SEG2PH0	-0-- -000	-0-- -000
BRGCON2	SEG2PHTS	SAM	SEG1PH2	SEG1PH1	SEG1PH0	PRSEG2	PRSEG1	PRSEG0	0000 0000	0000 0000
BRGCON1	SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0	0000 0000	0000 0000
CANCON	REQOP2	REQOP1	REQOP0	ABAT	WIN2	WIN1	WIN0	—	xxxx xxx-	uuuu uuu-
CANSTAT	OPMODE2	OPMODE1	OPMODE0	—	ICODE2	ICODE1	ICOED0	—	xxx- xxx-	uuu- uuu-

Legend: x = unknown, u = unchanged, - = unimplemented, q = value depends on condition

Note 1: Bit 6 of PORTA, LATA and TRISA are enabled in ECIO and RCIO oscillator modes only. In all other oscillator modes, they are disabled and read '0'.

2: Bit 21 of the TBLPTRU allows access to the device configuration bits.

3: Other (non-power-up) RESETs include external RESET through $\overline{\text{MCLR}}$ and Watchdog Timer Reset.

4: These registers are reserved on PIC18C658.

REGISTER 7-2: INTCON2 REGISTER

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RBPU}}$	INTEDG0	INTEDG1	INTEDG2	INTEDG3	TMR0IP	INT3IP	RBIP
bit 7							bit 0

bit 7 **$\overline{\text{RBPU}}$**: PORTB Pull-up Enable bit
1 = All PORTB pull-ups are disabled
0 = PORTB pull-ups are enabled by individual port latch values

bit 6 **INTEDG0**: External Interrupt 0 Edge Select bit
1 = Interrupt on rising edge
0 = Interrupt on falling edge

bit 5 **INTEDG1**: External Interrupt 1 Edge Select bit
1 = Interrupt on rising edge
0 = Interrupt on falling edge

bit 4 **INTEDG2**: External Interrupt 2 Edge Select bit
1 = Interrupt on rising edge
0 = Interrupt on falling edge

bit 3 **INTEDG3:** External Interrupt 3 Edge Select bit
1 = Interrupt on rising edge
0 = Interrupt on falling edge

bit 2 **TMR0IP**: TMR0 Overflow Interrupt Priority bit
1 = High priority
0 = Low priority

bit 1 **INT3IP**: INT3 External Interrupt Priority bit
1 = High priority
0 = Low priority

bit 0 **RBIP**: RB Port Change Interrupt Priority bit
1 = High priority
0 = Low priority

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

Note: Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt. This feature allows software polling.

© 2000 Microchip Technology Inc.

REGISTER 9-1: PSPCON REGISTER

R-0	R-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
IBF	OBF	IBOV	PSPMODE	—	—	—	—
bit 7							bit 0

bit 7 **IBF:** Input Buffer Full Status bit
1 = A word has been received and waiting to be read by the CPU
0 = No word has been received

bit 6 **OBF**: Output Buffer Full Status bit
1 = The output buffer still holds a previously written word
0 = The output buffer has been read

bit 5 **IBOV**: Input Buffer Overflow Detect bit (in Microprocessor mode)
1 = A write occurred when a previously input word has not been read (must be cleared in software)
0 = No overflow occurred

bit 4 **PSPMODE**: Parallel Slave Port Mode Select bit
1 = Parallel Slave Port mode
0 = General purpose I/O mode

bit 3-0 **Unimplemented**: Read as ’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

© 2000 Microchip Technology Inc.

15.4.2 GENERAL CALL ADDRESS SUPPORT

The addressing procedure for the I^2C bus is such that the first byte after the START condition usually determines which device will be the slave addressed by the master. The exception is the general call address, which can address all devices. When this address is used, all devices should, in theory, respond with an acknowledge.

The general call address is one of eight addresses reserved for specific purposes by the I^2C protocol. It consists of all 0's with R/$\overline{W}$ = 0.

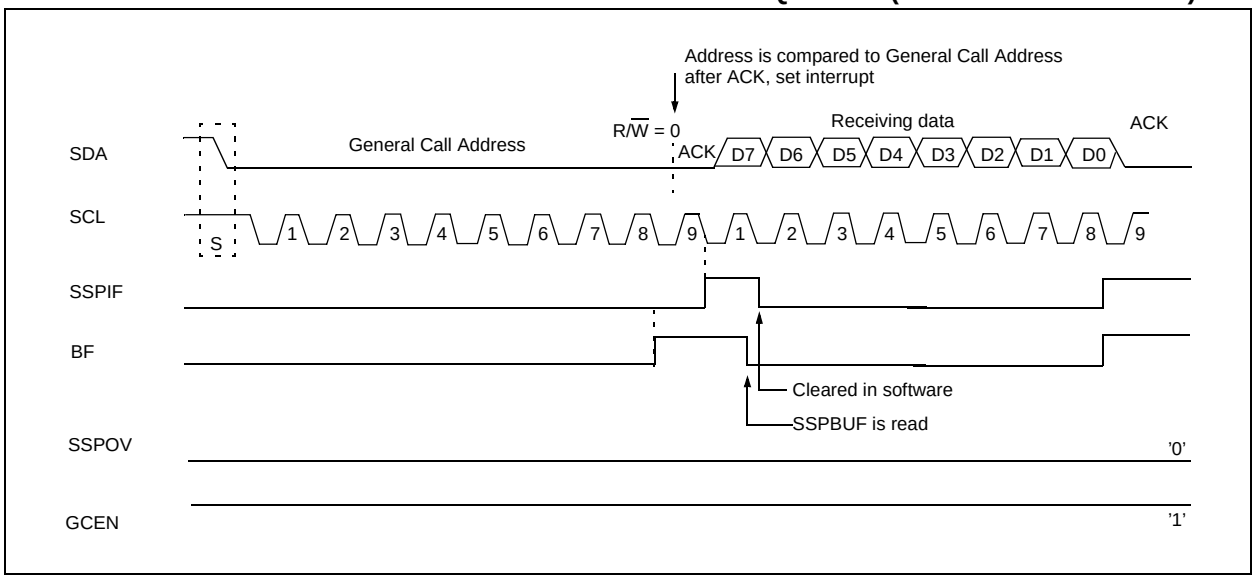
The general call address is recognized (enabled) when the General Call Enable (GCEN) bit is set (SSPCON2 register). Following a START bit detect, eight bits are shifted into the SSPSR and the address is compared against the SSPADD. It is also compared to the general call address and fixed in hardware.

If the general call address matches, the SSPSR is transferred to the SSPBUF, the BF bit is set (eighth bit), and on the falling edge of the ninth bit ($\overline{ACK}$ bit), the SSPIF interrupt flag bit is set.

When the interrupt is serviced, the source for the interrupt can be checked by reading the contents of the SSPBUF. The value can be used to determine if the address was device specific or a general call address.

In 10-bit mode, the SSPADD is required to be updated for the second half of the address to match, and the UA bit is set (SSPSTAT register). If the general call address is sampled when the GCEN bit is set, and while the slave is configured in 10-bit address mode; then, the second half of the address is not necessary. The UA bit will not be set, and the slave will begin receiving data after the Acknowledge (Figure 15-9).

FIGURE 15-9: SLAVE MODE GENERAL CALL ADDRESS SEQUENCE (7 OR 10-BIT ADDRESS)



 © 2000 Microchip Technology Inc.

15.4.3 MASTER MODE

Master mode of operation is supported by interrupt generation on the detection of the START and STOP conditions. The STOP (P) and START (S) bits are cleared from a RESET, or when the MSSP module is disabled. Control of the I^2C bus may be taken when the P bit is set, or the bus is idle, with both the S and P bits clear.

In Master mode, the SCL and SDA lines are manipulated by the MSSP hardware.

The following events will cause SSP Interrupt Flag bit, SSPIF, to be set (SSP Interrupt if enabled):

- START condition
- STOP condition
- Data transfer byte transmitted/received
- Acknowledge Transmit
- Repeated START condition

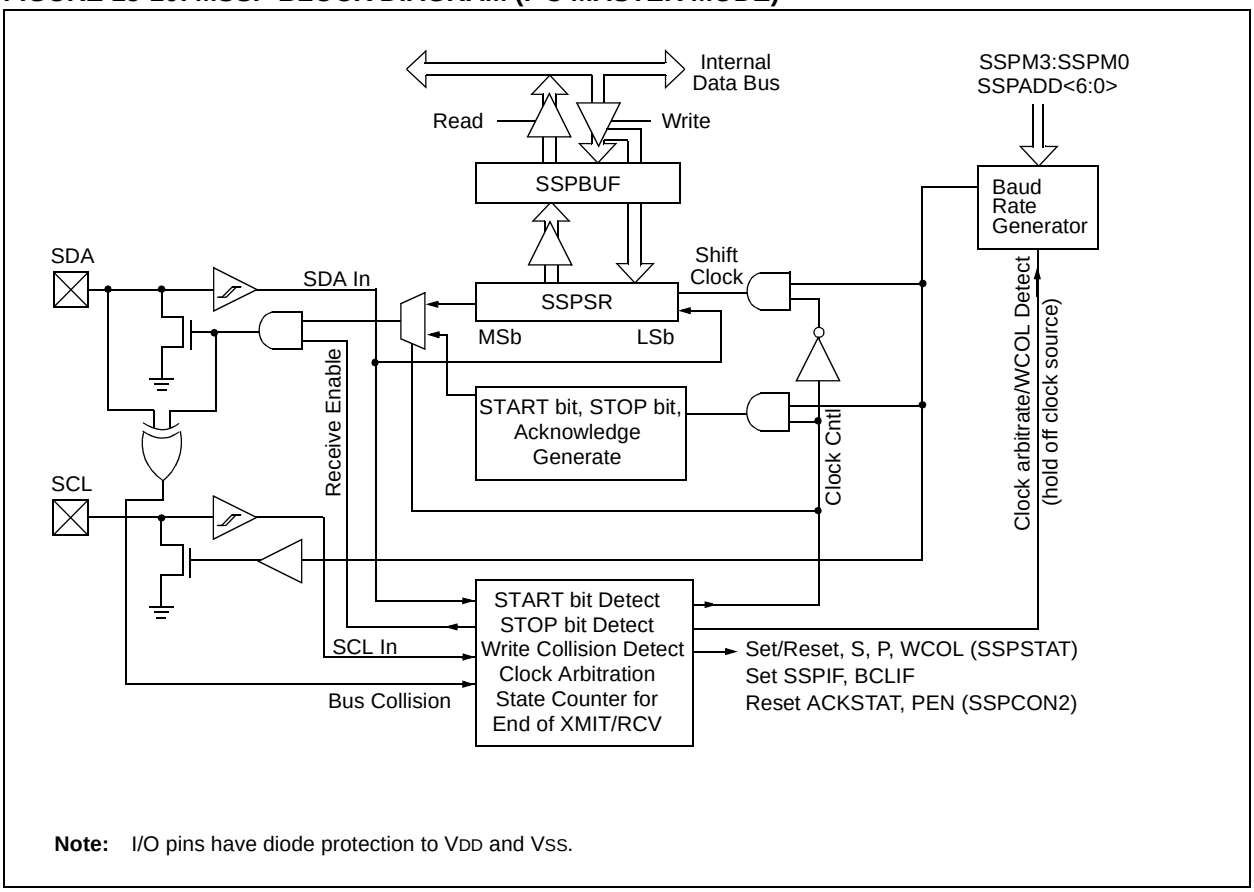
15.4.4 I^2C MASTER MODE SUPPORT

Master mode is enabled by setting and clearing the appropriate SSPM bits in SSPCON1 and by setting the SSPEN bit. Once Master mode is enabled, the user has the following six options:

1. Assert a START condition on SDA and SCL.
2. Assert a Repeated START condition on SDA and SCL.
3. Write to the SSPBUF register initiating transmission of data/address.
4. Generate a STOP condition on SDA and SCL.
5. Configure the I^2C port to receive data.
6. Generate an Acknowledge condition at the end of a received byte of data.

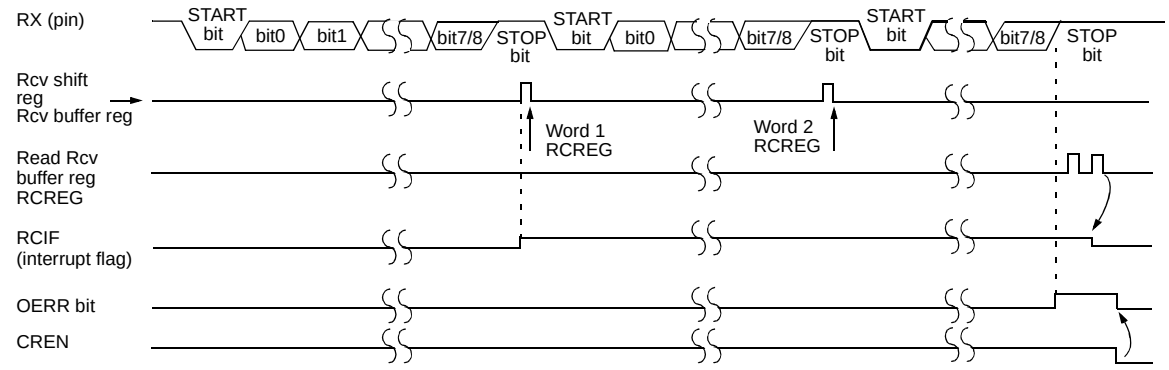
Note: The MSSP module, when configured in I^2C Master mode, does not allow queueing of events. For instance, the user is not allowed to initiate a START condition and immediately write the SSPBUF register to imitate transmission before the START condition is complete. In this case, the SSPBUF will not be written to and the WCOL bit will be set, indicating that a write to the SSPBUF did not occur.

FIGURE 15-10: MSSP BLOCK DIAGRAM (I^2C MASTER MODE)



Note: I/O pins have diode protection to VDD and VSS.

FIGURE 16-5: 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.

TABLE 16-7: REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other RESETS
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 000x	0000 000u
PIR1	PSPIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIE1	PSPIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
IPR1	PSPIP	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	—	FERR	OERR	RX9D	0000 -00x	0000 -00x
RCREG	USART Receive Register								0000 0000	0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC	ADDEN	BRGH	TRMT	TX9D	0000 0010	0000 0010
SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'. Shaded cells are not used for Asynchronous Reception.

 © 2000 Microchip Technology Inc.

```
RXB0Interrupt
    bcf     PIR3, RXB0IF                ; Clear the interrupt flag
    goto    AccessBuffer

AccessBuffer                            ; This is either TX or RX interrupt
    ; Copy CANCON.ICODE bits to CANSTAT.WIN bits
    movf    TempCANCON, W               ; Clear CANCON.WIN bits before copying
                                        ; new ones.
    andlw   b'11110001'                 ; Use previously saved CANCON value to
                                        ; make sure same value.
    movwf   TempCANCON                  ; Copy masked value back to TempCANCON

    movf    TempCANSTAT, W              ; Retrieve ICODE bits
    andlw   b'00001110'                 ; Use previously saved CANSTAT value
                                        ; to make sure same value.

    iorwf   TempCANCON                  ; Copy ICODE bits to WIN bits.
    movff   TempCANCON, CANCON          ; Copy the result to actual CANCON

    ; Access current buffer…
    ; Your code

    ; Restore CANCON.WIN bits
    movf    CANCON, W                   ; Preserve current non WIN bits
    andlw   b'11110001'
    iorwf   TempCANCON                  ; Restore original WIN bits

    ; Do not need to restore CANSTAT - it is read-only register.

    ; Return from interrupt or check for another module interrupt source
```

 © 2000 Microchip Technology Inc.

17.2.3 CAN RECEIVE BUFFER REGISTERS

This section shows the Receive Buffer registers with its associated control registers.

REGISTER 17-12: RXB0CON – RECEIVE BUFFER 0 CONTROL REGISTER

R/C-0	R/W-0	R/W-0	U-0	R-0	R/W-0	R-0	R/W-0
RXFUL	RXM1	RXM0	—	RXRTRRO	RXB0DBEN	JTOFF	FILHIT0
bit 7							bit 0

bit 7 **RXFUL:** Receive Full Status bit
1 = Receive buffer contains a received message
0 = Receive buffer is open to receive a new message

Note: This bit is set by the CAN module and should be cleared by software after the buffer is read.

bit 6-5 **RXM1:RXM0:** Receive Buffer Mode bits
11 = Receive all messages (including those with errors)
10 = Receive only valid messages with extended identifier
01 = Receive only valid messages with standard identifier
00 = Receive all valid messages

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

bit 3 **RXRTRRO:** Receive Remote Transfer Request Read Only bit
1 = Remote transfer request
0 = No remote transfer request

bit 2 **RXB0DBEN:** Receive Buffer 0 Double Buffer Enable bit
1 = Receive Buffer 0 overflow will write to Receive Buffer 1
0 = No Receive Buffer 0 overflow to Receive Buffer 1

bit 1 **JTOFF:** Jump Table Offset bit (read only copy of RX0DBEN)
1 = Allows Jump Table offset between 6 and 7
0 = Allows Jump Table offset between 1 and 0

Note: This bit allows same filter jump table for both RXB0CON and RXB1CON.

bit 0 **FILHIT0:** Filter Hit bit
This bit indicates which acceptance filter enabled the message reception into receive buffer 0
1 = Acceptance Filter 1 (RXF1)
0 = Acceptance Filter 0 (RXF0)

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

 © 2000 Microchip Technology Inc.

17.2.5 CAN BAUD RATE REGISTERS

This subsection describes the CAN Baud Rate registers.

REGISTER 17-29: BRGCON1 – BAUD RATE CONTROL REGISTER 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SJW1	SJW0	BRP5	BRP4	BRP3	BRP2	BRP1	BRP0
bit 7							bit 0

bit 7-6 **SJW1:SJW0:** Synchronized Jump Width bits
11 = Synchronization Jump Width Time = 4 x TQ
10 = Synchronization Jump Width Time = 3 x TQ
01 = Synchronization Jump Width Time = 2 x TQ
00 = Synchronization Jump Width Time = 1 x TQ

bit 5-0 **BRP5:BRP0:** Baud Rate Prescaler bits
111111 = TQ = (2 x 64)/FOSC
111110 = TQ = (2 x 63)/FOSC
:
:
000001 = TQ = (2 x 2)/FOSC
000000 = TQ = (2 x 1)/FOSC

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

Note: This register is only accessible in Configuration mode.

 © 2000 Microchip Technology Inc.

18.0 10-BIT ANALOG-TO-DIGITAL CONVERTER (A/D) MODULE

The analog-to-digital (A/D) converter module has twelve inputs for the PIC18C658 devices and sixteen for the PIC18C858 devices. This module has the ADCON0, ADCON1, and ADCON2 registers.

The A/D allows conversion of an analog input signal to a corresponding 10-bit digital number.

The A/D module has five registers:

- A/D Result High Register (ADRESH)
- A/D Result Low Register (ADRESL)
- A/D Control Register 0 (ADCON0)
- A/D Control Register 1 (ADCON1)
- A/D Control Register 2 (ADCON2)

The ADCON0 register, shown in Register 18-1, controls the operation of the A/D module. The ADCON1 register, shown in Register 18-2, configures the functions of the port pins. The ADCON2, shown in Register 16-3, configures the A/D clock source and justification.

REGISTER 18-1: ADCON0 REGISTER

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	CHS3	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	ADON
bit 7							bit 0

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

bit 5-2 **CHS3:CHS0:** Analog Channel Select bits
0000 = channel 00, (AN0)
0001 = channel 01, (AN1)
0010 = channel 02, (AN2)
0011 = channel 03, (AN3)
0100 = channel 04, (AN4)
0101 = channel 05, (AN5)
0110 = channel 06, (AN6)
0111 = channel 07, (AN7)
1000 = channel 08, (AN8)
1001 = channel 09, (AN9)
1010 = channel 10, (AN10)
1011 = channel 11, (AN11)
1100 = channel 12, (AN12)**(1)**
1101 = channel 13, (AN13)**(1)**
1110 = channel 14, (AN14)**(1)**
1111 = channel 15, (AN15)**(1)**

Note 1: These channels are not available on the PIC18C658 devices.

bit 1 **GO/$\overline{\text{DONE}}$:** A/D Conversion Status bit
When ADON = 1
1 = A/D conversion in progress. Setting this bit starts an A/D conversion cycle. This bit is automatically cleared by hardware when the A/D conversion is complete.
0 = A/D conversion not in progress

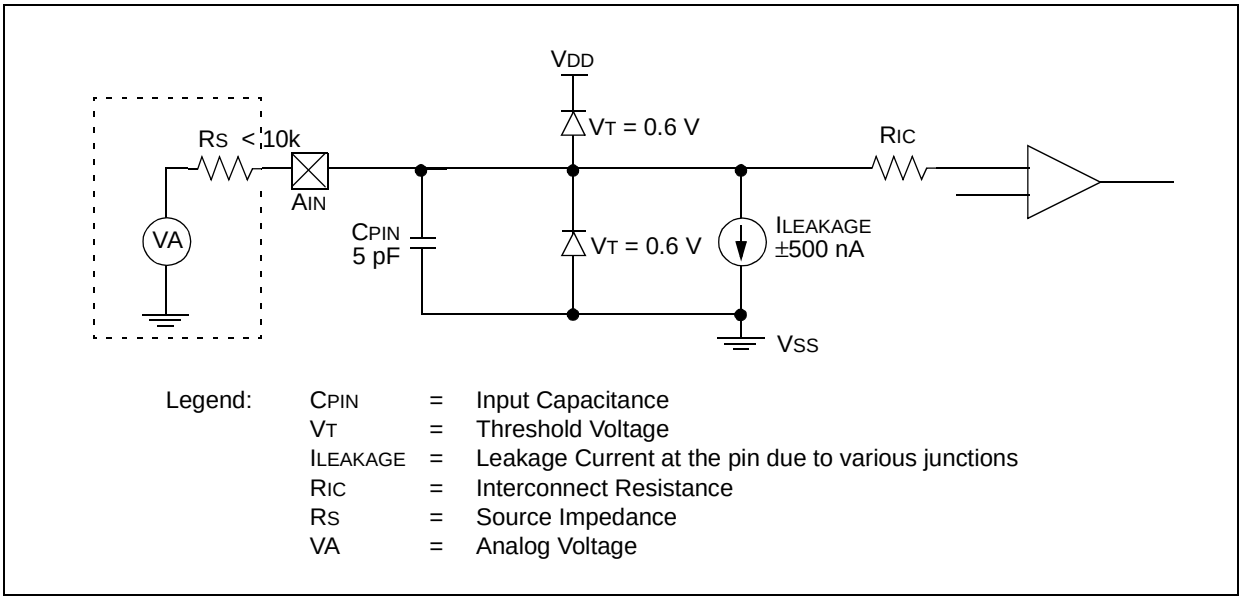
bit 0 **ADON:** A/D On bit
1 = A/D converter module is operating
0 = A/D converter module is shut off and consumes no operating current

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

19.9 Analog Input Connection Considerations

A simplified circuit for an analog input is shown in Figure 19-4. Since the analog pins are connected to a digital output, they have reverse biased diodes to VDD and VSS. The analog input, therefore, must be between VSS and VDD. If the input voltage deviates from this range by more than 0.6 V in either direction, one of the diodes is forward biased and a latch-up condition may occur. A maximum source impedance of 10 kΩ is recommended for the analog sources. Any external component connected to an analog input pin, such as a capacitor or a Zener diode, should have very little leakage current.

FIGURE 19-4: ANALOG INPUT MODEL



Legend:

CPIN	=	Input Capacitance
VT	=	Threshold Voltage
ILEAKAGE	=	Leakage Current at the pin due to various junctions
RIC	=	Interconnect Resistance
RS	=	Source Impedance
VA	=	Analog Voltage

TABLE 19-1: REGISTERS ASSOCIATED WITH COMPARATOR MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR	Value on All Other RESETS
CMCON	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0000	0000 0000
VRCON	VREN	VROE	VRR	VRSS	VR3	VR2	VR1	VR0	0000 0000	0000 0000
INTCON	GIE/ GIEH	PEIE/ GIEL	TMR0IE	INTIE	RBIE	TMR0IF	INTIF	RBIF	0000 000x	0000 000u
PIR2	—	CMIF	—	—	BCLIF	LVDIF	TMR3IF	CCP2IF	-0-- 0000	-0-- 0000
PIE2	—	CMIE	—	—	BCLIE	LVDIE	TMR3IE	CCP2IE	-0-- 0000	-0-- 0000
IPR2	—	CMIP	—	—	BCLIP	LVDIP	TMR3IP	CCP2IP	-1-- 1111	-1-- 1111
PORTF	RF7	RF6	RF5	RF4	RF3	RF2	RF1	RF0	x000 0000	u000 0000
LATF	LATF7	LATF6	LATF5	LATF4	LATF3	LATF2	LATF1	LATF0	xxxx xxxx	uuuu uuuu
TRISF	PORTF Data Direction Register								1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented, read as "0"

© 2000 Microchip Technology Inc.

TBLWT **Table Write**

Syntax: [*label*] TBLWT (*; *+; *-; +*)

Operands: None

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

Status Affected: None

Encoding:

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

Description: This instruction is used to program the contents of Program Memory (P.M.).

The TBLPTR (a 21-bit pointer) points to each byte in the program memory. TBLPTR has a 2 MBtye 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 (many if long write is to on-chip EPROM program memory)

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 or Memory)

TBLWT **Table Write (Continued)**

Example 1:

```
TBLWT  *+;
```

Before Instruction
TABLAT = 0x55
TBLPTR = 0x00A356
MEMORY(0x00A356) = 0xFF

After Instructions (table write completion)
TABLAT = 0x55
TBLPTR = 0x00A357
MEMORY(0x00A356) = 0x55

Example 2:

```
TBLWT  +*;
```

Before Instruction
TABLAT = 0x34
TBLPTR = 0x01389A
MEMORY(0x01389A) = 0xFF
MEMORY(0x01389B) = 0xFF

After Instruction (table write completion)
TABLAT = 0x34
TBLPTR = 0x01389B
MEMORY(0x01389A) = 0xFF
MEMORY(0x01389B) = 0x34

© 2000 Microchip Technology Inc.

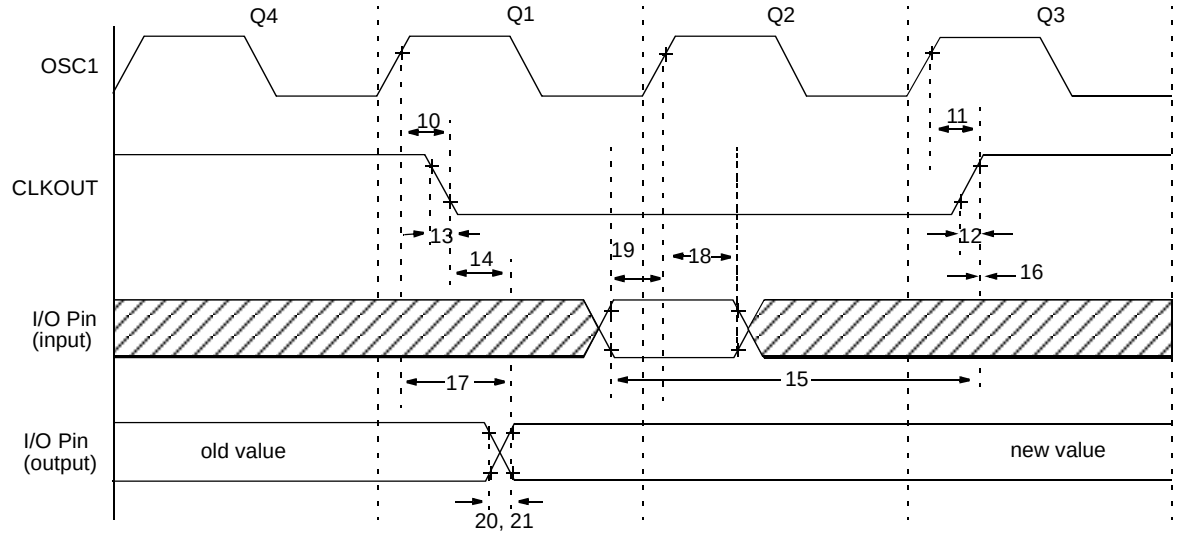
TABLE 24-1: DEVELOPMENT TOOLS FROM MICROCHIP

		PIC12CXXX	PIC14000	PIC16C5X	PIC16C6X	PIC16CXXX	PIC16F62X	PIC16C7X	PIC16C7XX	PIC16C8X	PIC16F8XX	PIC16C9XX	PIC17C4X	PIC17C7XX	PIC18CXX2	24CXX/ 25CXX/ 93CXX	HCSXXX	MCRFXXX	MCP2510
Software Tools	MPLAB® Integrated Development Environment	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
	MPLAB® C17 C Compiler												✓	✓					
	MPLAB® C18 C Compiler														✓				
	MPASM™ Assembler/ MPLINK™ Object Linker	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Emulators	MPLAB® ICE In-Circuit Emulator	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓				
	ICEPIC™ In-Circuit Emulator	✓		✓	✓	✓		✓	✓	✓		✓							
Debugger	MPLAB® ICD In-Circuit Debugger				✓*			✓*			✓								
Programmers	PICSTART® Plus Entry Level Development Programmer	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓				
	PRO MATE® II Universal Device Programmer	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓	✓			
Demo Boards and Eval Kits	PICDEM™ 1 Demonstration Board			✓		✓		✓†		✓			✓						
	PICDEM™ 2 Demonstration Board				✓†			✓†							✓				
	PICDEM™ 3 Demonstration Board											✓							
	PICDEM™ 14A Demonstration Board		✓																
	PICDEM™ 17 Demonstration Board													✓					
	KEELOQ® Evaluation Kit															✓	✓		
	KEELOQ® Transponder Kit															✓			
	microID™ Programmer's Kit																✓		
	125 kHz microID™ Developer's Kit																	✓	
	125 kHz Anticollision microID™ Developer's Kit																	✓	
	13.56 MHz Anticollision microID™ Developer's Kit																	✓	
	MCP2510 CAN Developer's Kit																		✓

* Contact the Microchip Technology Inc. web site at www.microchip.com for information on how to use the MPLAB® ICD In-Circuit Debugger (DV164001) with PIC16C62, 63, 64, 65, 72, 73, 74, 76, 77.

** Contact Microchip Technology Inc. for availability date.

FIGURE 25-6: CLKOUT AND I/O TIMING



Note: Refer to Figure 25-4 for load conditions.

TABLE 25-6: CLKOUT AND I/O TIMING REQUIREMENTS

Param. No.	Symbol	Characteristic		Min	Typ	Max	Units	Conditions
10	TosH2ckL	OSC1↑ to CLKOUT↓		—	75	200	ns	(1)
11	TosH2ckH	OSC1↑ to CLKOUT↑		—	75	200	ns	(1)
12	TckR	CLKOUT rise time		—	35	100	ns	(1)
13	TckF	CLKOUT fall time		—	35	100	ns	(1)
14	TckL2ioV	CLKOUT ↓ to Port out valid		—	—	0.5TCY + 20	ns	(1)
15	TioV2ckH	Port in valid before CLKOUT ↑		0.25TCY + 25	—	—	ns	(1)
16	TckH2ioI	Port in hold after CLKOUT ↑		0	—	—	ns	(1)
17	TosH2ioV	OSC1↑ (Q1 cycle) to Port out valid		—	50	150	ns	
18	TosH2ioI	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	PIC18**C**XX8	100	—	—	ns	
18A			PIC18**LC**XX8	200	—	—	ns	
19	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)		0	—	—	ns	
20	TioR	Port output rise time	PIC18**C**XX8	—	10	25	ns	
20A			PIC18**LC**XX8	—	—	60	ns	
21	TioF	Port output fall time	PIC18**C**XX8	—	10	25	ns	
21A			PIC18**LC**XX8	—	—	60	ns	
22††	TINP	INT pin high or low time		TCY	—	—	ns	
23††	TRBP	RB7:RB4 change INT high or low time		TCY	—	—	ns	
24††	TRCP	RC7:RC4 change INT high or low time		20	—	—	ns	

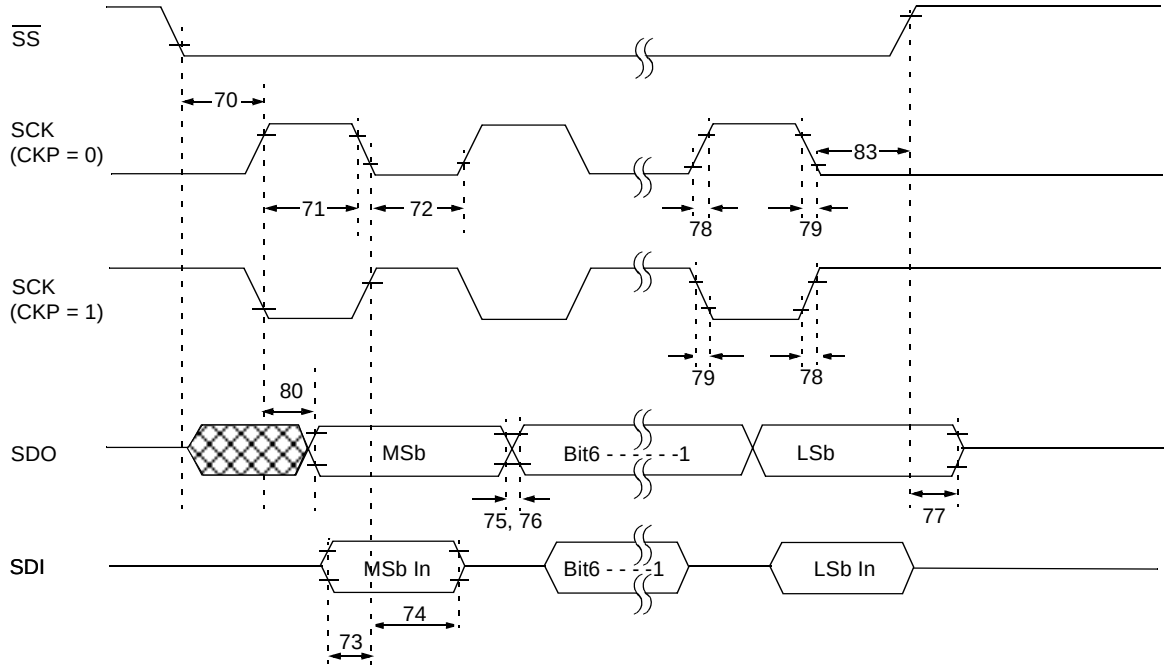
†† These parameters are asynchronous events, not related to any internal clock edges.

Note 1: Measurements are taken in RC mode where CLKO pin output is 4 x TOSC.

PRELIMINARY

 © 2000 Microchip Technology Inc.

FIGURE 25-14: EXAMPLE SPI SLAVE MODE TIMING (CKE = 0)



Note: Refer to Figure 25-4 for load conditions.

TABLE 25-13: EXAMPLE SPI MODE REQUIREMENTS (SLAVE MODE TIMING (CKE = 0))

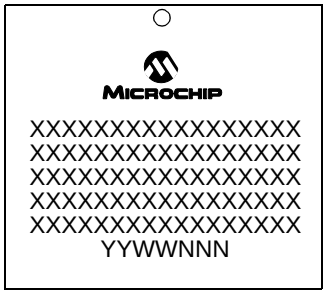
Parm. No.	Symbol	Characteristic		Min	Max	Units	Conditions
70	TssL2scH, TssL2scL	$\overline{SS}\downarrow$ to SCK$\downarrow$ or SCK$\uparrow$ input		TCY	—	ns	
71	TscH	SCK input high time (Slave mode)	Continuous	1.25TCY + 30	—	ns	
71A			Single Byte	40	—	ns	**(Note 1)**
72	TscL	SCK input low time (Slave mode)	Continuous	1.25TCY + 30	—	ns	
72A			Single Byte	40	—	ns	**(Note 1)**
73	TdiV2scH, TdiV2scL	Setup time of SDI data input to SCK edge		100	—	ns	
73A	TB2B	Last clock edge of Byte1 to the 1st clock edge of Byte2		1.5TCY + 40	—	ns	**(Note 2)**
74	TscH2diL, TscL2diL	Hold time of SDI data input to SCK edge		100	—	ns	
75	TdoR	SDO data output rise time	PIC18**C**XX8	—	25	ns	
			PIC18**LC**XX8		45	ns	
76	TdoF	SDO data output fall time		—	25	ns	
77	TssH2doZ	$\overline{SS}\uparrow$ to SDO output hi-impedance		10	50	ns	
78	TscR	SCK output rise time (Master mode)	PIC18**C**XX8	—	25	ns	
			PIC18**LC**XX8		45	ns	
79	TscF	SCK output fall time (Master mode)		—	25	ns	
80	TscH2doV, TscL2doV	SDO data output valid after SCK edge	PIC18**C**XX8	—	50	ns	
			PIC18**LC**XX8		100	ns	
83	TscH2ssH, TscL2ssH	$\overline{SS}\uparrow$ after SCK edge		1.5TCY + 40	—	ns	

Note 1: Requires the use of parameter # 73A.
2: Only if parameter #s 71A and 72A are used.

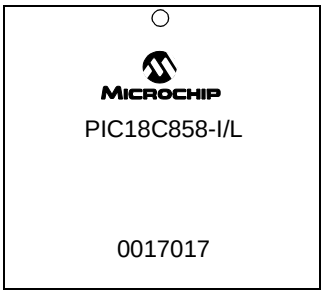
PRELIMINARY

Package Marking Information (Cont'd)

84-Lead PLCC



Example



 © 2000 Microchip Technology Inc.

APPENDIX E: DEVELOPMENT TOOL VERSION REQUIREMENTS

This lists the minimum requirements (software/firmware) of the specified development tool to support the devices listed in this data sheet.

MPLAB-IDE: version 5.11

MPLAB-SIM: version 7.10

MPLAB-ICE 2000:

PIC18CXX8 Processor Module:
Part Number - PCM 18XB0

PIC18CXX8 Device Adapter:

Socket	Part Number
64-pin TQFP	DVD18P2640
68-pin PLCC	DVD18XL680
80-pin TQFP	DVD18PQ800
84-pin PLCC	DVD18XL840

MPLAB-ICD: Not Available

PROMATE II: version 5.20

PICSTART Plus: version 2.20

MPASM: version 2.50

MPLAB-C18: version 1.00

CAN-TOOL: Not available at time of printing.

Note: Please read all associated README.TXT files that are supplied with the development tools. These "read me" files will discuss product support and any known limitations.