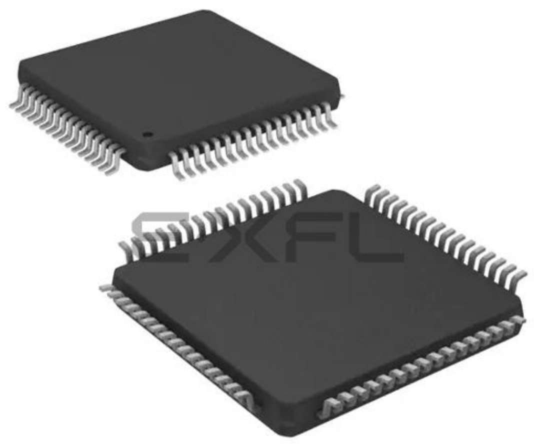
Microchip Technology - PIC18F66J90T-I/PT Datasheet



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	Active
Core Processor	PIC
Core Size	8-Bit
Speed	48MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LCD, LVD, POR, PWM, WDT
Number of I/O	51
Program Memory Size	64KB (32K 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 12x10b
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/pic18f66j90t-i-pt

3.4 External Oscillator Modes

3.4.1 CRYSTAL OSCILLATOR/CERAMIC RESONATORS (HS MODES)

In HS or HSPLL Oscillator modes, a crystal or ceramic resonator 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 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.

TABLE 3-1: CAPACITOR SELECTION FOR CERAMIC RESONATORS

Typical Capacitor Values Used:			
Mode	**Freq.**	**OSC1**	**OSC2**
HS	8.0 MHz 16.0 MHz	27 pF 22 pF	27 pF 22 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 Table 3-2 for additional information.

TABLE 3-2: CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR

Osc Type	Crystal Freq.	Typical Capacitor Values Tested:	
		C1	**C2**
HS	4 MHz	27 pF	27 pF
	8 MHz	22 pF	22 pF
	20 MHz	15 pF	15 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 Microchip application notes cited in Table 3-1 for oscillator specific information. Also see the notes following this table for additional information.

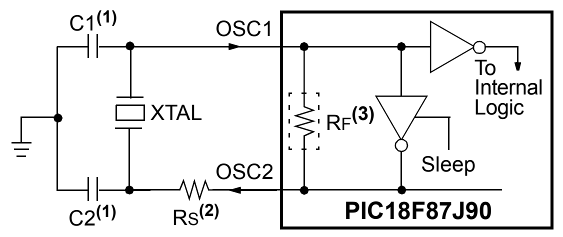
Note 1: Higher capacitance increases the stability of oscillator but also increases the start-up time.

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

3: RS may be required to avoid overdriving crystals with low drive level specification.

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

FIGURE 3-2: CRYSTAL/CERAMIC RESONATOR OPERATION (HS OR HSPLL CONFIGURATION)



Note 1: See Table 3-1 and Table 3-2 for initial values of C1 and C2.

2: A series resistor (RS) may be required for AT strip cut crystals.

3: RF varies with the oscillator mode chosen.

10.9 PORTH, LATH and TRISH Registers

Note:	PORTH is available only on PIC18F8XJ90 devices.

PORTH is an 8-bit wide, bidirectional I/O port. The corresponding Data Direction and Output Latch registers are TRISH and LATH. All pins are digital only and tolerate voltages up to 5.5V.

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

All PORTH pins are multiplexed with LCD segment drives controlled by the LCDSE5 register. I/O port functions are only available when the segments are disabled.

EXAMPLE 10-8: 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
```

16.2 Capture Mode

In Capture mode, the CCPR2H:CCPR2L register pair captures the 16-bit value of the TMR1 or TMR3 register when an event occurs on the CCP2 pin (RC1 or RE7, depending on device configuration). An event is defined as one of the following:

- Every falling edge
- Every rising edge
- Every 4th rising edge
- Every 16th rising edge

The event is selected by the mode select bits, CCP2M<3:0> (CCP2CON<3:0>). When a capture is made, the interrupt request flag bit, CCP2IF (PIR3<2>), is set; it must be cleared in software. If another capture occurs before the value in register, CCPR2, is read, the old captured value is overwritten by the new captured value.

16.2.1 CCP PIN CONFIGURATION

In Capture mode, the appropriate CCPx pin should be configured as an input by setting the corresponding TRIS direction bit.

Note:	If RC1/CCP2 or RE7/CCP2 is configured as an output, a write to the port can cause a capture condition.

16.2.2 TIMER1/TIMER3 MODE SELECTION

The timers that are to be used with the capture feature (Timer1 and/or Timer3) must be running in Timer mode or Synchronized Counter mode. In Asynchronous Counter mode, the capture operation may not work. The timer to be used with each CCP module is selected in the T3CON register (see **Section 16.1.1 “CCP Modules and Timer Resources”**).

16.2.3 SOFTWARE INTERRUPT

When the Capture mode is changed, a false capture interrupt may be generated. The user should keep the CCP2IE bit (PIE3<2>) clear to avoid false interrupts and should clear the flag bit, CCP2IF, following any such change in operating mode.

16.2.4 CCP PRESCALER

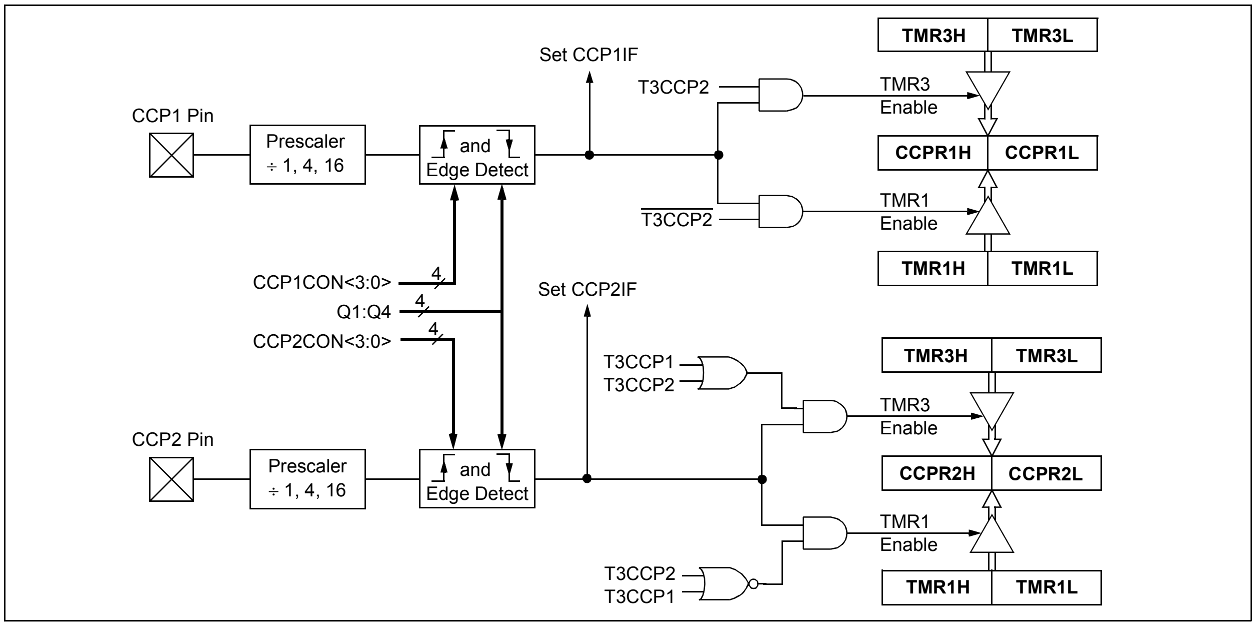
There are four prescaler settings in Capture mode. They are specified as part of the operating mode selected by the mode select bits (CCP2M<3:0>). Whenever the CCP module is turned off, or the CCP module is not in Capture mode, the prescaler counter is cleared. This means that any Reset will clear the prescaler counter.

Switching from one capture prescaler to another may generate an interrupt. Also, the prescaler counter will not be cleared; therefore, the first capture may be from a non-zero prescaler. Example 16-1 shows the recommended method for switching between capture prescalers. This example also clears the prescaler counter and will not generate the “false” interrupt.

EXAMPLE 16-1: CHANGING BETWEEN CAPTURE PRESCALERS

```
CLRF   CCP2CON      ; Turn CCP module off
MOVLW  NEW_CAPT_PS  ; Load WREG with the
                    ; new prescaler mode
                    ; value and CCP ON
MOVWF  CCP2CON      ; Load CCP2CON with
                    ; this value
```

FIGURE 16-2: CAPTURE MODE OPERATION BLOCK DIAGRAM



© 2010 Microchip Technology Inc.

REGISTER 17-2: LCDPS: LCD PHASE REGISTER

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
WFT	BIASMD	LCDA	WA	LP3	LP2	LP1	LP0
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 **WFT:** Waveform Type Select bit
`1` = Type-B waveform (phase changes on each frame boundary)
`0` = Type-A waveform (phase changes within each common type)

bit 6 **BIASMD:** Bias Mode Select bit
When LMUX<1:0> = `00`:
`0` = Static Bias mode (do not set this bit to ‘`1`’)
When LMUX<1:0> = `01` or `10`:
`1` = 1/2 Bias mode
`0` = 1/3 Bias mode
When LMUX<1:0> = `11`:
`0` = 1/3 Bias mode (do not set this bit to ‘`1`’)

bit 5 **LCDA:** LCD Active Status bit
`1` = LCD driver module is active
`0` = LCD driver module is inactive

bit 4 **WA:** LCD Write Allow Status bit
`1` = Write into the LCDDATAx registers is allowed
`0` = Write into the LCDDATAx registers is not allowed

bit 3-0 **LP<3:0>:** LCD Prescaler Select bits
`1111` = 1:16
`1110` = 1:15
`1101` = 1:14
`1100` = 1:13
`1011` = 1:12
`1010` = 1:11
`1001` = 1:10
`1000` = 1:9
`0111` = 1:8
`0110` = 1:7
`0101` = 1:6
`0100` = 1:5
`0011` = 1:4
`0010` = 1:3
`0001` = 1:2
`0000` = 1:1

17.3 LCD Bias Generation

The LCD driver module is capable of generating the required bias voltages for LCD operation with a minimum of external components. This includes the ability to generate the different voltage levels required by the different bias types that are required by the LCD. The driver module can also provide bias voltages, both above and below microcontroller VDD, through the use of an on-chip LCD voltage regulator.

17.3.1 LCD BIAS TYPES

PIC18F87J90 family devices support three bias types based on the waveforms generated to control segments and commons:

- Static (two discrete levels)
- 1/2 Bias (three discrete levels
- 1/3 Bias (four discrete levels)

The use of different waveforms in driving the LCD is discussed in more detail in **Section 17.8 "LCD Waveform Generation"**.

17.3.2 LCD VOLTAGE REGULATOR

The purpose of the LCD regulator is to provide proper bias voltage and good contrast for the LCD, regardless of VDD levels. This module contains a charge pump and internal voltage reference. The regulator can be configured by using external components to boost bias voltage above VDD. It can also operate a display at a constant voltage below VDD. The regulator can also be selectively disabled to allow bias voltages to be generated by an external resistor network.

The LCD regulator is controlled through the LCDREG register (Register 17-5). It is enabled or disabled using the CKSEL<1:0> bits, while the charge pump can be selectively enabled using the CPEN bit. When the regulator is enabled, the MODE13 bit is used to select the bias type. The peak LCD bias voltage, measured as a difference between the potentials of LCDBIAS3 and LCDBIAS0, is configured with the BIAS bits.

REGISTER 17-5: LCDREG: VOLTAGE REGULATOR CONTROL REGISTER

U-0	RW-0	RW-1	RW-1	RW-1	RW-1	RW-0	RW-0
—	CPEN	BIAS2	BIAS1	BIAS0	MODE13	CKSEL1	CKSEL0
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'

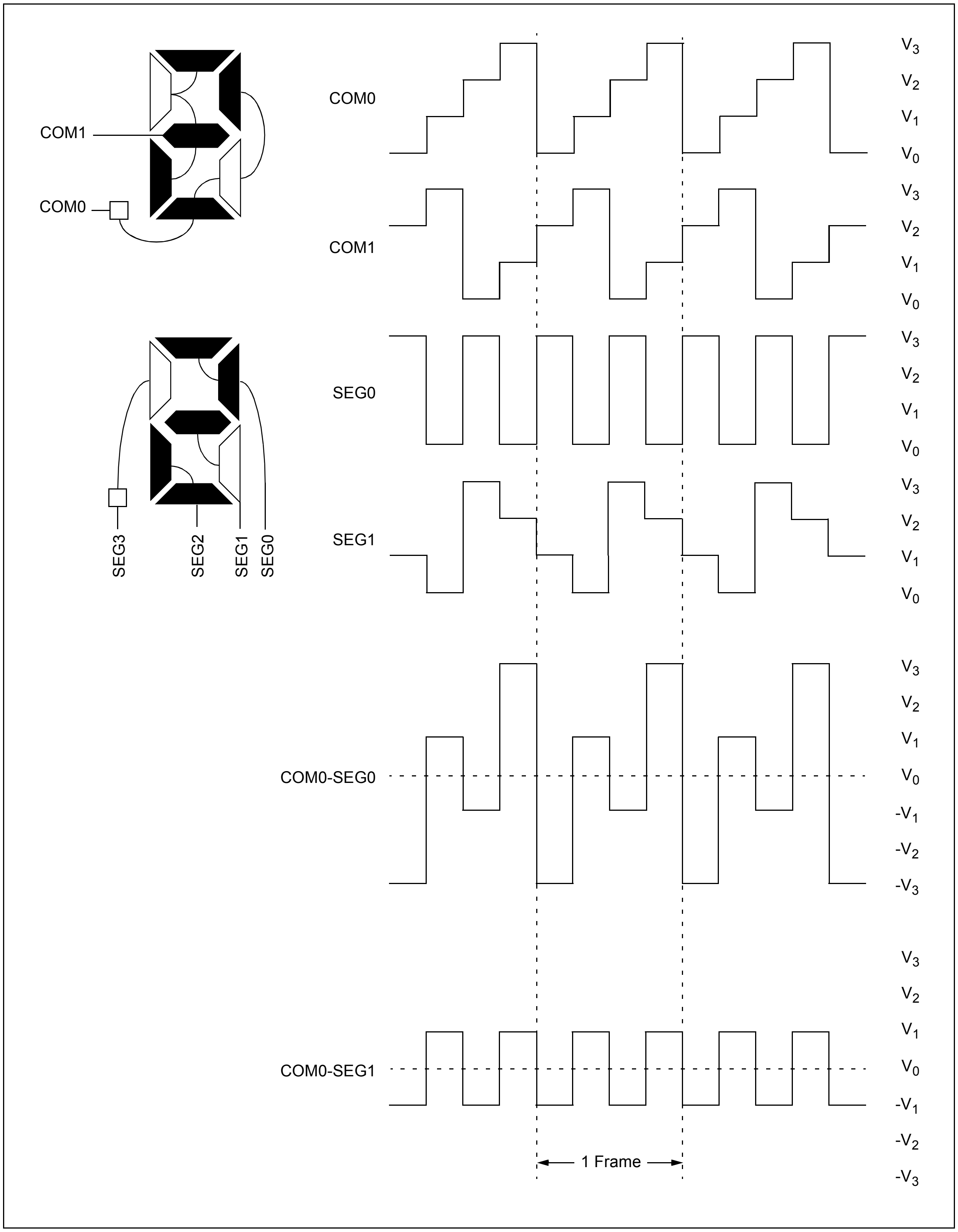
bit 6 **CPEN:** LCD Charge Pump Enable bit
1 = Charge pump enabled; highest LCD bias voltage is 3.6V
0 = Charge pump disabled; highest LCD bias voltage is AVDD

bit 5-3 **BIAS<2:0>:** Regulator Voltage Output Control bits
111 = 3.60V peak (offset on LCDBIAS0 of 0V)
110 = 3.47V peak (offset on LCDBIAS0 of 0.13V)
101 = 3.34V peak (offset on LCDBIAS0 of 0.26V)
100 = 3.21V peak (offset on LCDBIAS0 of 0.39V)
011 = 3.08V peak (offset on LCDBIAS0 of 0.52V)
010 = 2.95V peak (offset on LCDBIAS0 of 0.65V)
001 = 2.82V peak (offset on LCDBIAS0 of 0.78V)
000 = 2.69V peak (offset on LCDBIAS0 of 0.91V)

bit 2 **MODE13:** 1/3 LCD Bias Enable bit
1 = Regulator output supports 1/3 LCD Bias mode
0 = Regulator output supports static LCD Bias mode

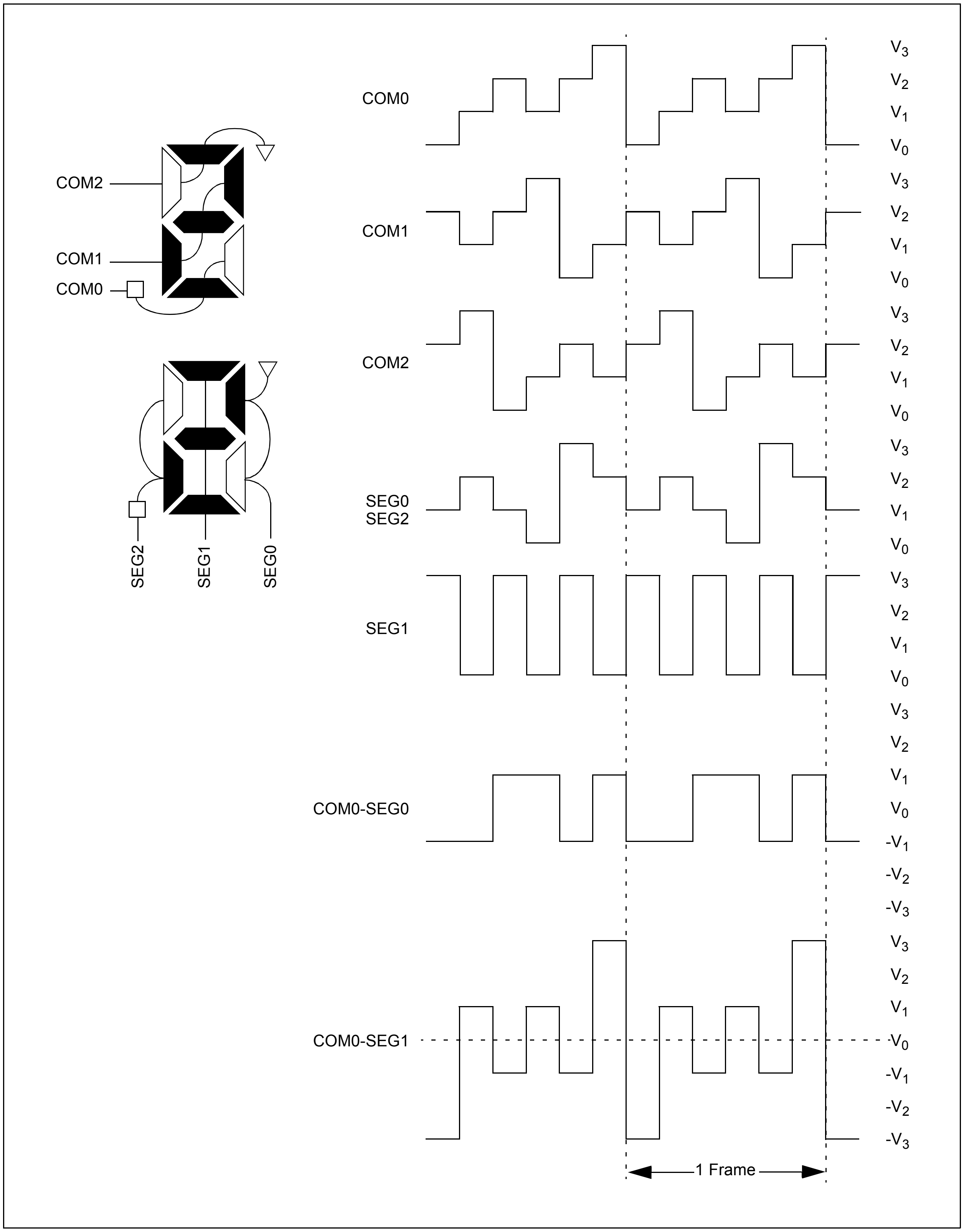
bit 1-0 **CKSEL<1:0>:** Regulator Clock Source Select bits
11 = INTRC
10 = INTOSC 8 MHz source
01 = Timer1 oscillator
00 = LCD regulator disabled

FIGURE 17-9: TYPE-A WAVEFORMS IN 1/2 MUX, 1/3 BIAS DRIVE



© 2010 Microchip Technology Inc.

FIGURE 17-13: TYPE-A WAVEFORMS IN 1/3 MUX, 1/3 BIAS DRIVE



© 2010 Microchip Technology Inc.

17.9 LCD Interrupts

The LCD timing generation provides an interrupt that defines the LCD frame timing. This interrupt can be used to coordinate the writing of the pixel data with the start of a new frame. Writing pixel data at the frame boundary allows a visually crisp transition of the image. This interrupt can also be used to synchronize external events to the LCD. For example, the interface to an external segment driver can be synchronized for a segment data update to the LCD frame.

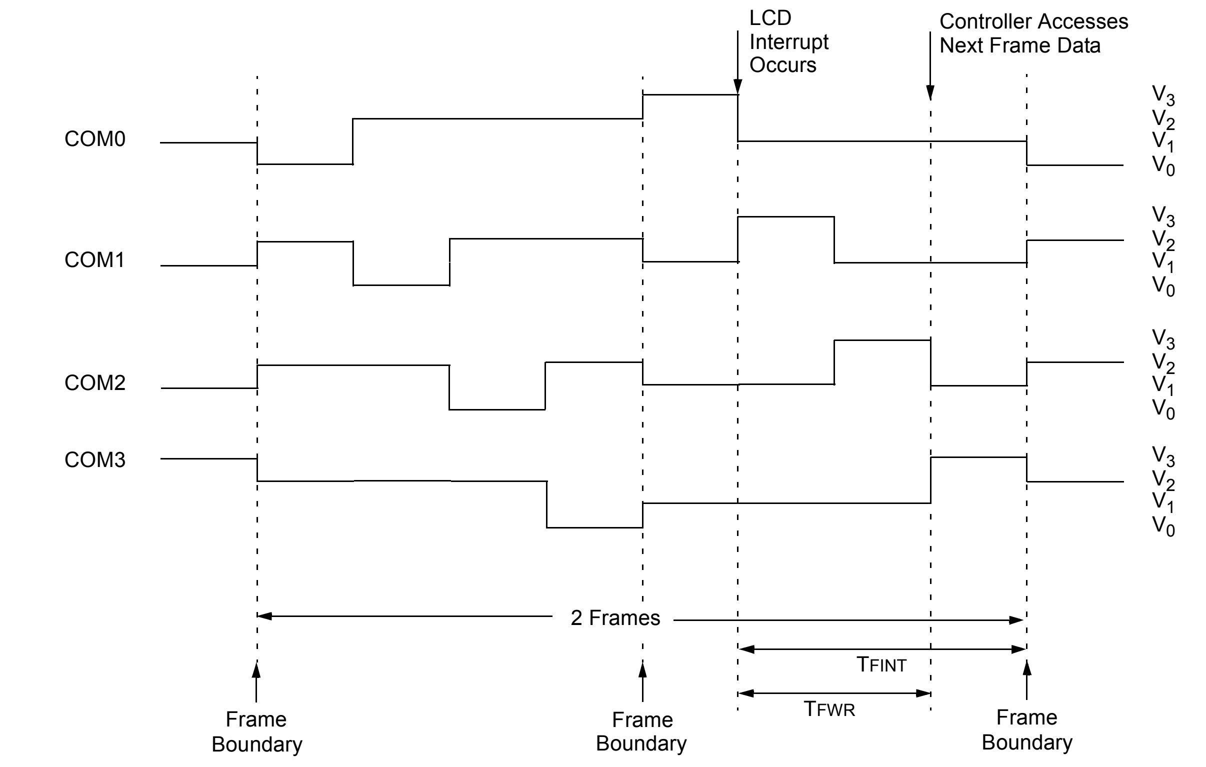
A new frame is defined to begin at the leading edge of the COM0 common signal. The interrupt will be set immediately after the LCD controller completes accessing all pixel data required for a frame. This will occur at a fixed interval before the frame boundary (TFINT), as shown in Figure 17-17. The LCD controller will begin to access data for the next frame within the interval from the interrupt to when the controller begins to access data after the interrupt (TFWR). New data must be written within TFWR, as this is when the LCD controller will begin to access the data for the next frame.

When the LCD driver is running with Type-B waveforms, and the LMUX<1:0> bits are not equal to '00', there are some additional issues that must be addressed. Since the DC voltage on the pixel takes two frames to maintain zero volts, the pixel data must not change between subsequent frames. If the pixel data was allowed to change, the waveform for the odd frames would not necessarily be the complement of the waveform generated in the even frames and a DC component would be introduced into the panel. Therefore, when using Type-B waveforms, the user must synchronize the LCD pixel updates to occur within a subframe after the frame interrupt.

To correctly sequence writing while in Type-B, the interrupt will only occur on complete phase intervals. If the user attempts to write when the write is disabled, the WERR (LCDCON<5>) bit is set.

Note: The interrupt is not generated when the Type-A waveform is selected and when the Type-B with no multiplex (static) is selected.

FIGURE 17-17: EXAMPLE WAVEFORMS AND INTERRUPT TIMING IN QUARTER DUTY CYCLE DRIVE



TFWR = TFRAME/2 * (LMUX<1:0> + 1) + TCY/2

TFINT = (TFWR/2 – (2 TCY + 40 ns)) → Minimum = 1.5(TFRAME/4) – (2 TCY + 40 ns)
(TFWR/2 – (1 TCY + 40 ns)) → Maximum = 1.5(TFRAME/4) – (1 TCY + 40 ns)

18.3.1 REGISTERS

Each MSSP module has four registers for SPI mode operation. These are:

- MSSP Control Register 1 (SSPCON1)
- MSSP Status Register (SSPSTAT)
- Serial Receive/Transmit Buffer Register (SSPBUF)
- MSSP Shift Register (SSPSR) – Not directly accessible

SSPCON1 and SSPSTAT are the control and status registers in SPI mode operation. The SSPCON1 register is readable and writable. The lower 6 bits of the SSPSTAT are read-only. The upper two bits of the SSPSTAT are read/write.

SSPSR is the shift register used for shifting data in or out. SSPBUF is the buffer register to which data bytes are written to or read from.

In receive operations, SSPSR and SSPBUF together, create a double-buffered receiver. When SSPSR receives a complete byte, it is transferred to SSPBUF and the SSPIF interrupt is set.

During transmission, the SSPBUF is not double-buffered. A write to SSPBUF will write to both, SSPBUF and SSPSR.

REGISTER 18-1: SSPSTAT: MSSP STATUS REGISTER (SPI MODE)

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R0	R-0
SMP	CKE[1]	D/$\overline{\text{A}}$	P	S	R/$\overline{\text{W}}$	UA	BF
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 **SMP:** Sample bit
SPI Master mode:
1 = Input data sampled at the end of data output time
0 = Input data sampled at the middle of data output time
SPI Slave mode:
SMP must be cleared when SPI is used in Slave mode.

bit 6 **CKE:** SPI Clock Select bit[1]
1 = Transmit occurs on transition from active to Idle clock state
0 = Transmit occurs on transition from Idle to active clock state

bit 5 **D/$\overline{\text{A}}$:** Data/$\overline{\text{Address}}$ bit
Used in I^{2}C™ mode only.

bit 4 **P:** Stop bit
Used in I^{2}C mode only. This bit is cleared when the MSSP module is disabled; SSPEN is cleared.

bit 3 **S:** Start bit
Used in I^{2}C mode only.

bit 2 **R/$\overline{\text{W}}$:** Read/$\overline{\text{Write}}$ Information bit
Used in I^{2}C mode only.

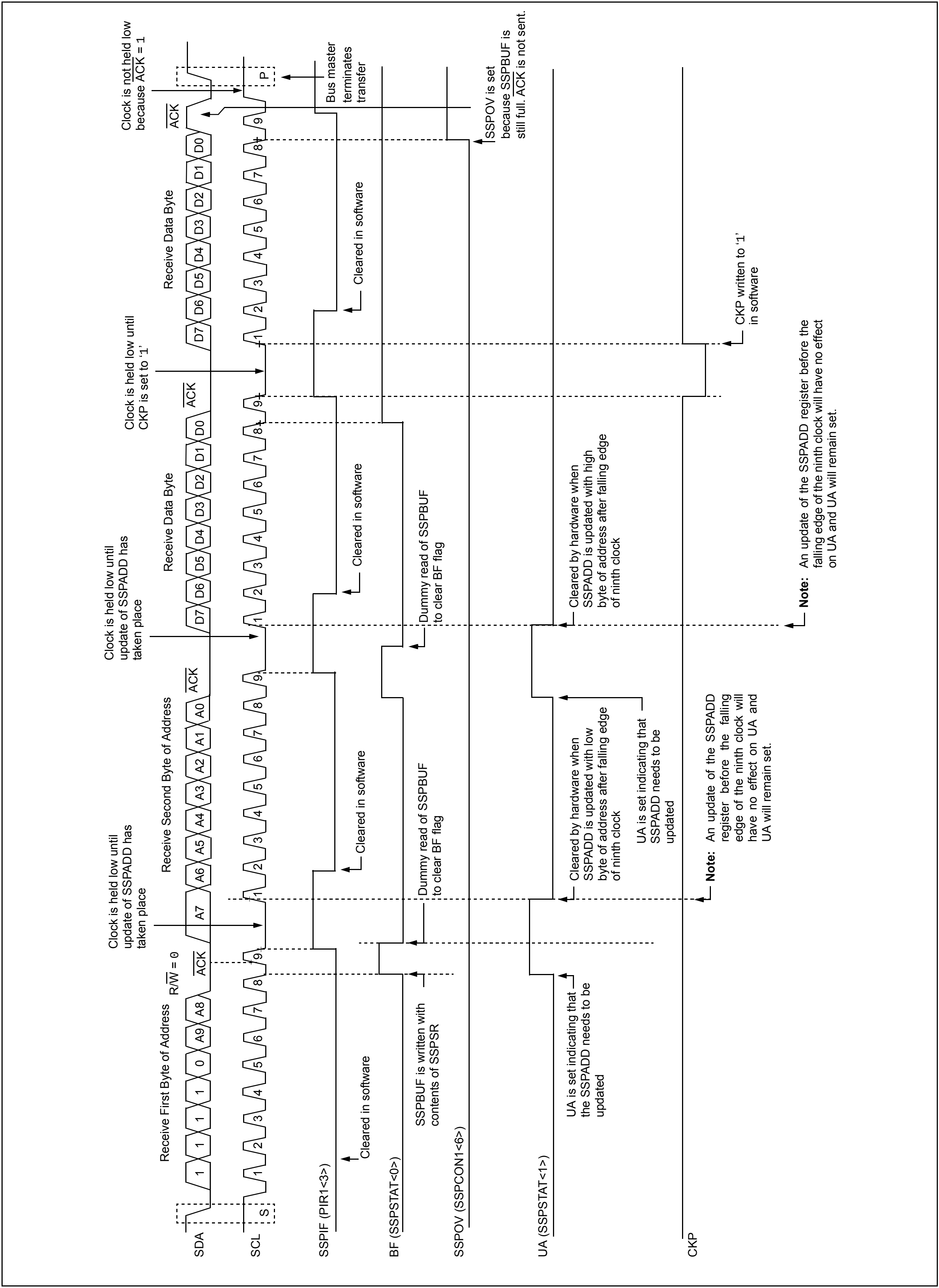
bit 1 **UA:** Update Address bit
Used in I^{2}C mode only.

bit 0 **BF:** Buffer Full Status bit (Receive mode only)
1 = Receive complete; SSPBUF is full
0 = Receive not complete; SSPBUF is empty

Note 1: Polarity of the clock state is set by the CKP bit (SSPCON1<4>).

© 2010 Microchip Technology Inc.

FIGURE 18-16: I²C™ SLAVE MODE TIMING WITH SEN = 1 (RECEPTION, 10-BIT ADDRESSING)


SDA
SCL
S
P
SSPIF (PIR1<3>)
BF (SSPSTAT<0>)
SSPOV (SSPCON1<6>)
UA (SSPSTAT<1>)
CKP
Receive First Byte of Address
1 1 1 1 0 A9 A8
R/W = 0
ACK
Receive Second Byte of Address
A7 A6 A5 A4 A3 A2 A1 A0
Receive Data Byte
D7 D6 D5 D4 D3 D2 D1 D0
1 2 3 4 5 6 7 8 9
Clock is held low until update of SSPADD has taken place
Clock is held low until CKP is set to '1'
Clock is not held low because ACK = 1
Bus master terminates transfer
Cleared in software
SSPBUF is written with contents of SSPSR
Dummy read of SSPBUF to clear BF flag
SSPOV is set because SSPBUF is still full. ACK is not sent.
UA is set indicating that the SSPADD needs to be updated
Cleared by hardware when SSPADD is updated with low byte of address after falling edge of ninth clock
Cleared by hardware when SSPADD is updated with high byte of address after falling edge of ninth clock
CKP written to '1' in software
Note: An update of the SSPADD register before the falling edge of the ninth clock will have no effect on UA and UA will remain set.

19.0 ENHANCED UNIVERSAL SYNCHRONOUS ASYNCHRONOUS RECEIVER TRANSMITTER (EUSART)

PIC18F87J90 family devices have three serial I/O modules: the MSSP module, discussed in the previous chapter and two Universal Synchronous Asynchronous Receiver Transmitter (USART) modules. (Generically, the USART is also known as a Serial Communications Interface or SCI.) The USART can be configured as a full-duplex, asynchronous system that can communicate with peripheral devices, such as CRT terminals and personal computers. It can also be configured as a half-duplex synchronous system that can communicate with peripheral devices, such as A/D or D/A integrated circuits, serial EEPROMs, etc.

There are two distinct implementations of the USART module in these devices: the Enhanced USART (EUSART) discussed here and the Addressable USART discussed in the next chapter. For this device family, USART1 always refers to the EUSART, while USART2 is always the AUSART.

The EUSART and AUSART modules implement the same core features for serial communications; their basic operation is essentially the same. The EUSART module provides additional features, including Automatic Baud Rate Detection and calibration, automatic wake-up on Sync Break reception, and 12-bit Break character transmit. These features make it ideally suited for use in Local Interconnect Network bus (LIN/J2602 bus) systems.

The EUSART can be configured in the following modes:

- Asynchronous (full-duplex) with:
  - Auto-wake-up on character reception
  - Auto-baud calibration
  - 12-bit Break character transmission
- Synchronous – Master (half-duplex) with selectable clock polarity
- Synchronous – Slave (half-duplex) with selectable clock polarity

The pins of the EUSART are multiplexed with the functions of PORTC (RC6/TX1/CK1/SEG27 and RC7/RX1/DT1/SEG28). In order to configure these pins as an EUSART:

- bit, SPEN (RCSTA1<7>), must be set (= `1`)
- bit, TRISC<7>, must be set (= `1`)
- bit, TRISC<6>, must be set (= `1`)

Note:	The EUSART control will automatically reconfigure the pin from input to output as needed.

The driver for the TX1 output pin can also be optionally configured as an open-drain output. This feature allows the voltage level on the pin to be pulled to a higher level through an external pull-up resistor, and allows the output to communicate with external circuits without the need for additional level shifters.

The open-drain output option is controlled by the U1OD bit (LATG<6>). Setting the bit configures the pin for open-drain operation.

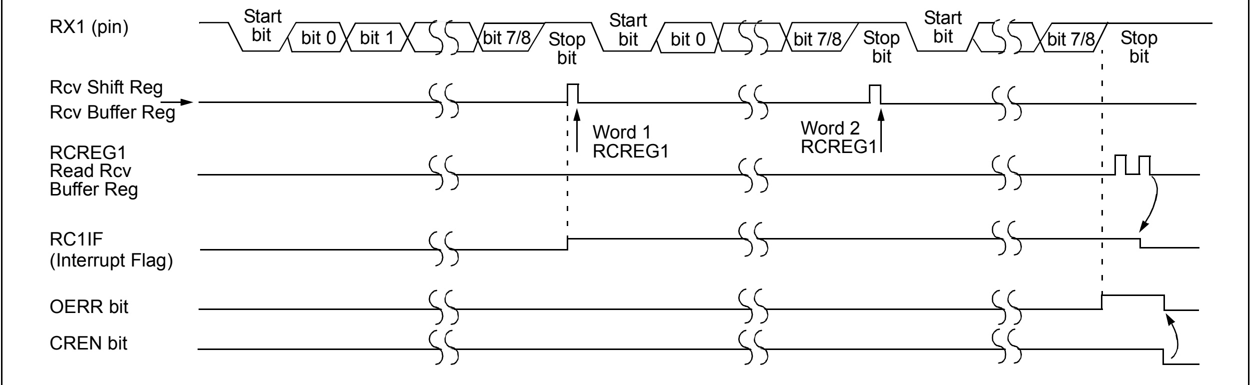
19.1 Control Registers

The operation of the Enhanced USART module is controlled through three registers:

- Transmit Status and Control Register 1 (TXSTA1)
- Receive Status and Control Register 1 (RCSTA1)
- Baud Rate Control Register 1 (BAUDCON1)

The registers are described in Register 19-1, Register 19-2 and Register 19-3.

FIGURE 19-7: ASYNCHRONOUS RECEPTION



Note: This timing diagram shows three words appearing on the RX1 input. The RCREG1 (Receive Buffer register) is read after the third word causing the OERR (Overrun) bit to be set.

TABLE 19-6: REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION

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	59
PIR1	—	ADIF	RC1IF	TX1IF	SSPIF	—	TMR2IF	TMR1IF	62
PIE1	—	ADIE	RC1IE	TX1IE	SSPIE	—	TMR2IE	TMR1IE	62
IPR1	—	ADIP	RC1IP	TX1IP	SSPIP	—	TMR2IP	TMR1IP	62
RCSTA1	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	61
RCREG1	EUSART Receive Register								61
TXSTA1	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	61
BAUDCON1	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN	63
SPBRGH1	EUSART Baud Rate Generator Register High Byte								61
SPBRG1	EUSART Baud Rate Generator Register Low Byte								61

Legend: — = unimplemented locations read as '0'. Shaded cells are not used for asynchronous reception.

24.1 CTMU Operation

The CTMU works by using a fixed current source to charge a circuit. The type of circuit depends on the type of measurement being made. In the case of charge measurement, the current is fixed and the amount of time the current is applied to the circuit is fixed. The amount of voltage read by the A/D is then a measurement of the capacitance of the circuit. In the case of time measurement, the current, as well as the capacitance of the circuit, are fixed. In this case, the voltage read by the A/D is then representative of the amount of time elapsed from the time the current source starts and stops charging the circuit.

If the CTMU is being used as a time delay, both capacitance and current source are fixed, as well as the voltage supplied to the comparator circuit. The delay of a signal is determined by the amount of time it takes the voltage to charge to the comparator threshold voltage.

24.1.1 THEORY OF OPERATION

The operation of the CTMU is based on the equation for charge:

$$C = I \cdot \frac{dV}{dT}$$

More simply, the amount of charge measured in coulombs in a circuit is defined as current in amperes (I) multiplied by the amount of time in seconds that the current flows (t). Charge is also defined as the capacitance in farads (C) multiplied by the voltage of the circuit (V). It follows that:

$$I \cdot t = C \cdot V$$

The CTMU module provides a constant, known current source. The A/D Converter is used to measure (V) in the equation, leaving two unknowns: capacitance (C) and time (t). The above equation can be used to calculate capacitance or time, by either relationship using the known fixed capacitance of the circuit:

$$t = (C \cdot V)/I$$

or by:

$$C = (I \cdot t)/V$$

using a fixed time that the current source is applied to the circuit.

24.1.2 CURRENT SOURCE

At the heart of the CTMU is a precision current source, designed to provide a constant reference for measurements. The level of current is user-selectable across three ranges or a total of two orders of magnitude, with the ability to trim the output in ±2% increments (nominal). The current range is selected by the IRNG<1:0> bits (CTMUICON<1:0>), with a value of '`00`' representing the lowest range.

Current trim is provided by the ITRIM<5:0> bits (CTMUICON<7:2>). These six bits allow trimming of the current source in steps of approximately 2% per step. Note that half of the range adjusts the current source positively and the other half reduces the current source. A value of '`000000`' is the neutral position (no change). A value of '`100000`' is the maximum negative adjustment (approximately -62%) and '`011111`' is the maximum positive adjustment (approximately +62%).

24.1.3 EDGE SELECTION AND CONTROL

CTMU measurements are controlled by edge events occurring on the module's two input channels. Each channel, referred to as Edge 1 and Edge 2, can be configured to receive input pulses from one of the edge input pins (CTEDG1 and CTEDG2) or the CCPx Special Event Triggers. The input channels are level-sensitive, responding to the instantaneous level on the channel rather than a transition between levels. The inputs are selected using the EDG1SEL and EDG2SEL bit pairs (CTMUCONL<3:2, 6:5>).

In addition to source, each channel can be configured for event polarity using the EDGE2POL and EDGE1POL bits (CTMUCONL<7,4>). The input channels can also be filtered for an edge event sequence (Edge 1 occurring before Edge 2) by setting the EDGSEQEN bit (CTMUCONH<2>).

24.1.4 EDGE STATUS

The CTMUCON register also contains two status bits, EDG2STAT and EDG1STAT (CTMUCONL<1:0>). Their primary function is to show if an edge response has occurred on the corresponding channel. The CTMU automatically sets a particular bit when an edge response is detected on its channel. The level-sensitive nature of the input channels also means that the status bits become set immediately if the channel's configuration is changed and is the same as the channel's current state.

The module uses the edge status bits to control the current source output to external analog modules (such as the A/D Converter). Current is only supplied to external modules when only one (but not both) of the status bits is set, and shuts current off when both bits are either set or cleared. This allows the CTMU to measure current only during the interval between edges. After both status bits are set, it is necessary to clear them before another measurement is taken. Both bits should be cleared simultaneously, if possible, to avoid re-enabling the CTMU current source.

In addition to being set by the CTMU hardware, the edge status bits can also be set by software. This is also the user's application to manually enable or disable the current source. Setting either one (but not both) of the bits enables the current source. Setting or clearing both bits at once disables the source.

© 2010 Microchip Technology Inc.

IORLW	**Inclusive OR Literal with W**
Syntax:	IORLW k
Operands:	$0 \le k \le 255$
Operation:	(W) .OR. k → W
Status Affected:	N, Z
Encoding:	`0000` `1001` `kkkk` `kkkk`
Description:	The contents of W are ORed with the eight-bit literal 'k'. The result is placed in W.
Words:	1
Cycles:	1

Q Cycle Activity:

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

Example: `IORLW 35h`

Before Instruction
W = 9Ah
After Instruction
W = BFh

IORWF	**Inclusive OR W with f**
Syntax:	IORWF f {,d {,a}}
Operands:	$0 \le f \le 255$ $d \in [0,1]$ $a \in [0,1]$
Operation:	(W) .OR. (f) → dest
Status Affected:	N, Z
Encoding:	`0001` `00da` `ffff` `ffff`
Description:	Inclusive OR W with register 'f'. If 'd' is '`0`', the result is placed in W. If 'd' is '`1`', the result is placed back in register 'f'. If 'a' is '`0`', the Access Bank is selected. If 'a' is '`1`', the BSR is used to select the GPR bank. 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 to destination

Example: `IORWF RESULT, 0, 1`

Before Instruction
RESULT = 13h
W = 91h
After Instruction
RESULT = 13h
W = 93h

© 2010 Microchip Technology Inc.

POP	**Pop Top of Return Stack**
Syntax:	POP
Operands:	None
Operation:	(TOS) → bit bucket
Status Affected:	None
Encoding:	`0000` `0000` `0000` `0110`
Description:	The TOS value is pulled off the return stack and is discarded. The TOS value then becomes the previous value that was pushed onto the return stack. This instruction is provided to enable the user to properly manage the return stack to incorporate a software stack.
Words:	1
Cycles:	1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	POP TOS value	No operation

Example:
```
POP
GOTO      NEW
```

Before Instruction

TOS	=	0031A2h
Stack (1 level down)	=	014332h

After Instruction

TOS	=	014332h
PC	=	NEW

PUSH	**Push Top of Return Stack**
Syntax:	PUSH
Operands:	None
Operation:	(PC + 2) → TOS
Status Affected:	None
Encoding:	`0000` `0000` `0000` `0101`
Description:	The PC + 2 is pushed onto the top of the return stack. The previous TOS value is pushed down on the stack. This instruction allows implementing a software stack by modifying TOS and then pushing it onto the return stack.
Words:	1
Cycles:	1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	PUSH PC + 2 onto return stack	No operation	No operation

Example:
```
PUSH
```

Before Instruction

TOS	=	345Ah
PC	=	0124h

After Instruction

PC	=	0126h
TOS	=	0126h
Stack (1 level down)	=	345Ah

© 2010 Microchip Technology Inc.

RLNCF **Rotate Left f (No Carry)**

Syntax:	RLNCF f {,d {,a}}
Operands:	0 ≤ f ≤ 255 d ∈ [0,1] a ∈ [0,1]
Operation:	(f<n>) → dest<n + 1>, (f<7>) → dest<0>
Status Affected:	N, Z

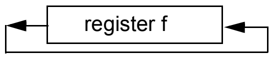
Encoding:

`0100`	`01da`	`ffff`	`ffff`

Description: The contents of register ‘f’ are rotated one bit to the left. If ‘d’ is ‘`0`’, the result is placed in W. If ‘d’ is ‘`1`’, the result is stored back in register ‘f’.

If ‘a’ is ‘`0`’, the Access Bank is selected. If ‘a’ is ‘`1`’, the BSR is used to select the GPR bank.

If ‘a’ is ‘`0`’ and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever f ≤ 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 to destination

Example: `RLNCF REG, 1, 0`

```
Before Instruction
    REG    =   1010 1011
After Instruction
    REG    =   0101 0111
```

RRCF **Rotate Right f through Carry**

Syntax:	RRCF f {,d {,a}}
Operands:	0 ≤ f ≤ 255 d ∈ [0,1] a ∈ [0,1]
Operation:	(f<n>) → dest<n – 1>, (f<0>) → C, (C) → dest<7>
Status Affected:	C, N, Z

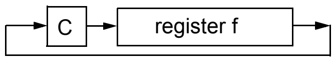
Encoding:

`0011`	`00da`	`ffff`	`ffff`

Description: The contents of register ‘f’ are rotated one bit to the right through the Carry flag. If ‘d’ is ‘`0`’, the result is placed in W. If ‘d’ is ‘`1`’, the result is placed back in register ‘f’.

If ‘a’ is ‘`0`’, the Access Bank is selected. If ‘a’ is ‘`1`’, the BSR is used to select the GPR bank.

If ‘a’ is ‘`0`’ and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever f ≤ 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 to destination

Example: `RRCF REG, 0, 0`

```
Before Instruction
    REG    =   1110 0110
    C      =   0
After Instruction
    REG    =   1110 0110
    W      =   0111 0011
    C      =   0
```

© 2010 Microchip Technology Inc.

SUBWFB — Subtract W from f with Borrow

Syntax:	SUBWFB f {,d {,a}}
Operands:	$0 \le f \le 255$ $d \in [0,1]$ $a \in [0,1]$
Operation:	$(f) - (W) - (\overline{C}) \rightarrow$ dest
Status Affected:	N, OV, C, DC, Z
Encoding:	`0101` `10da` `ffff` `ffff`
Description:	Subtract W and the Carry flag (borrow) from register 'f' (2's complement method). If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f'. If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank. 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 to destination

Example 1: `SUBWFB REG, 1, 0`

```
Before Instruction
    REG   =   19h    (0001 1001)
    W     =   0Dh    (0000 1101)
    C     =   1
After Instruction
    REG   =   0Ch    (0000 1011)
    W     =   0Dh    (0000 1101)
    C     =   1
    Z     =   0
    N     =   0      ; result is positive
```

Example 2: `SUBWFB REG, 0, 0`

```
Before Instruction
    REG   =   1Bh    (0001 1011)
    W     =   1Ah    (0001 1010)
    C     =   0
After Instruction
    REG   =   1Bh    (0001 1011)
    W     =   00h
    C     =   1
    Z     =   1      ; result is zero
    N     =   0
```

Example 3: `SUBWFB REG, 1, 0`

```
Before Instruction
    REG   =   03h    (0000 0011)
    W     =   0Eh    (0000 1101)
    C     =   1
After Instruction
    REG   =   F5h    (1111 0100)
                     ; [2's comp]
    W     =   0Eh    (0000 1101)
    C     =   0
    Z     =   0
    N     =   1      ; result is negative
```

SWAPF — Swap f

Syntax:	SWAPF f {,d {,a}}
Operands:	$0 \le f \le 255$ $d \in [0,1]$ $a \in [0,1]$
Operation:	$(f<3:0>) \rightarrow dest<7:4>$, $(f<7:4>) \rightarrow dest<3:0>$
Status Affected:	None
Encoding:	`0011` `10da` `ffff` `ffff`
Description:	The upper and lower nibbles of register 'f' are exchanged. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed in register 'f'. If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank. 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 to destination

Example: `SWAPF REG, 1, 0`

```
Before Instruction
    REG   =   53h
After Instruction
    REG   =   35h
```

© 2010 Microchip Technology Inc.

28.3 DC Characteristics: PIC18F87J90 Family (Industrial)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial			
Param No.	**Symbol**	**Characteristic**	**Min**	**Max**	**Units**	**Conditions**
	VIL	**Input Low Voltage**				
		All I/O Ports:				
D030		with TTL Buffer	VSS	0.15 VDD	V	VDD < 3.3V
D030A			—	0.8	V	3.3V ≤ VDD ≤ 3.6V
D031		with Schmitt Trigger Buffer	VSS	0.2 VDD	V	
D031A		with RC3 and RC4	VSS	0.3 VDD	V	I^{2}C™ enabled
D031B			VSS	0.8	V	SMBus enabled
D032		$\overline{\text{MCLR}}$	VSS	0.2 VDD	V	
D033		OSC1	VSS	0.3 VDD	V	HS, HSPLL modes
D033A		OSC1	VSS	0.2 VDD	V	EC, ECPLL modes
D034		T13CKI	VSS	0.3	V	
	VIH	**Input High Voltage**				
		I/O Ports with non 5.5V Tolerance:				
D040		with TTL Buffer	0.25 VDD + 0.8V	VDD	V	VDD < 3.3V
D040A			2.0	VDD	V	3.3V ≤ VDD ≤ 3.6V
D041		with Schmitt Trigger Buffer	0.8 VDD	VDD	V	
D041A		RC3 and RC4	0.7 VDD	VDD	V	I^{2}C enabled
D041B			2.1	VDD	V	SMBus enabled
		I/O Ports with 5.5V Tolerance:				
		with TTL Buffer	0.25 VDD + 0.8V	5.5	V	VDD < 3.3V
			2.0	5.5	V	3.3V ≤ VDD ≤ 3.6V
		with Schmitt Trigger Buffer	0.8 VDD	5.5	V	
D042		$\overline{\text{MCLR}}$	0.8 VDD	VDD	V	
D043		OSC1	0.7 VDD	VDD	V	HS, HSPLL modes
D043A		OSC1	0.8 VDD	VDD	V	EC, ECPLL modes
D044		T13CKI	1.6	VDD	V	
	IIL	**Input Leakage Current**(1)				
D060		I/O Ports with Analog Functions	—	200	nA	VSS ≤ VPIN ≤ VDD, Pin at high-impedance
		Digital Only I/O Ports	—	200	nA	VSS ≤ VPIN ≤ 5.5V
D061		$\overline{\text{MCLR}}$	—	±1	μA	Vss ≤ VPIN ≤ VDD
D063		OSC1	—	±1	μA	Vss ≤ VPIN ≤ VDD
	IPU	**Weak Pull-up Current**				
D070	IPURB	PORTB Weak Pull-up Current	30	400	μA	VDD = 3.3V, VPIN = VSS

Note 1: Negative current is defined as current sourced by the pin.

© 2010 Microchip Technology Inc.

28.5.3 TIMING DIAGRAMS AND SPECIFICATIONS

FIGURE 28-4: EXTERNAL CLOCK TIMING

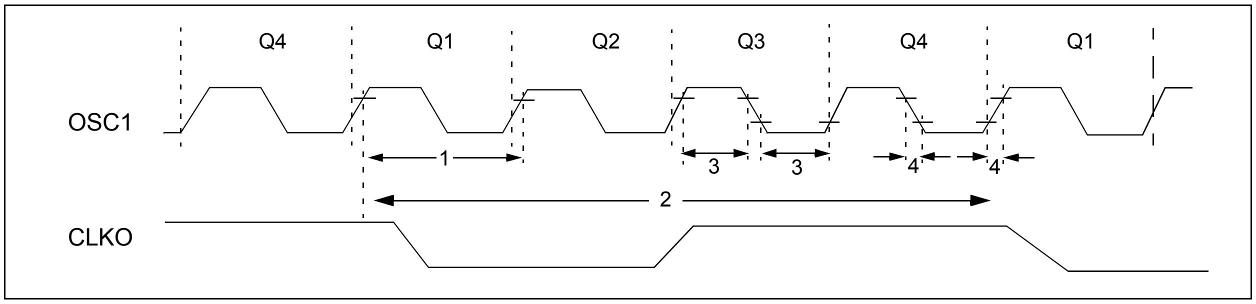


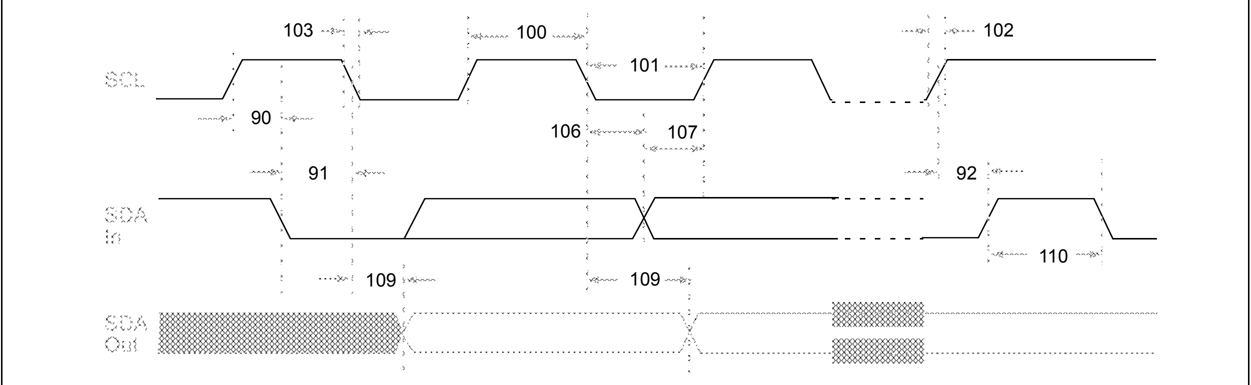
TABLE 28-7: EXTERNAL CLOCK TIMING REQUIREMENTS

Param. No.	Symbol	Characteristic	Min	Max	Units	Conditions
1A	FOSC	External CLKI Frequency(1)	DC	48	MHz	EC Oscillator mode
			DC	10		ECPLL Oscillator mode
		Oscillator Frequency(1)	4	25	MHz	HS Oscillator mode
			4	10		HSPLL Oscillator mode
1	TOSC	External CLKI Period(1)	20.8	—	ns	EC Oscillator mode
			100	—		ECPLL Oscillator mode
		Oscillator Period(1)	40.0	250	ns	HS Oscillator mode
			100	250		HSPLL Oscillator mode
2	TCY	Instruction Cycle Time(1)	83.3	—	ns	TCY = 4/FOSC, Industrial
3	TOSL, TOSH	External Clock in (OSC1) High or Low Time	10	—	ns	HS Oscillator mode
4	TOSR, TOSF	External Clock in (OSC1) Rise or Fall Time	—	7.5	ns	HS Oscillator mode

Note 1: Instruction cycle period (TCY) equals four times the input oscillator time base period for all configurations except PLL. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at “min.” values with an external clock applied to the OSC1/CLKI pin. When an external clock input is used, the “max.” cycle time limit is “DC” (no clock) for all devices.

© 2010 Microchip Technology Inc.

FIGURE 28-14: I²C™ BUS DATA TIMING



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

TABLE 28-19: I²C™ BUS DATA REQUIREMENTS (SLAVE MODE)

Param. No.	Symbol	Characteristic		Min	Max	Units	Conditions
100	THIGH	Clock High Time	100 kHz mode	4.0	—	μs	
			400 kHz mode	0.6	—	μs	
			MSSP Module	1.5 TCY	—		
101	TLOW	Clock Low Time	100 kHz mode	4.7	—	μs	
			400 kHz mode	1.3	—	μs	
			MSSP Module	1.5 TCY	—		
102	TR	SDA and SCL Rise Time	100 kHz mode	—	1000	ns	
			400 kHz mode	20 + 0.1 CB	300	ns	CB is specified to be from 10 to 400 pF
103	TF	SDA and SCL Fall Time	100 kHz mode	—	300	ns	
			400 kHz mode	20 + 0.1 CB	300	ns	CB is specified to be from 10 to 400 pF
90	TSU:STA	Start Condition Setup Time	100 kHz mode	4.7	—	μs	Only relevant for Repeated Start condition
			400 kHz mode	0.6	—	μs	
91	THD:STA	Start Condition Hold Time	100 kHz mode	4.0	—	μs	After this period, the first clock pulse is generated
			400 kHz mode	0.6	—	μs	
106	THD:DAT	Data Input Hold Time	100 kHz mode	0	—	ns	
			400 kHz mode	0	0.9	μs	
107	TSU:DAT	Data Input Setup Time	100 kHz mode	250	—	ns	**(Note 2)**
			400 kHz mode	100	—	ns	
92	TSU:STO	Stop Condition Setup Time	100 kHz mode	4.7	—	μs	
			400 kHz mode	0.6	—	μs	
109	TAA	Output Valid from Clock	100 kHz mode	—	3500	ns	**(Note 1)**
			400 kHz mode	—	—	ns	
110	TBUF	Bus Free Time	100 kHz mode	4.7	—	μs	Time the bus must be free before a new transmission can start
			400 kHz mode	1.3	—	μs	
D102	CB	Bus Capacitive Loading		—	400	pF	

Note 1: As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCL to avoid unintended generation of Start or Stop conditions.

2: A Fast mode I²C™ bus device can be used in a Standard mode I²C bus system, but the requirement, TSU:DAT $\geq$ 250 ns, must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line, TR max. + TSU:DAT = 1000 + 250 = 1250 ns (according to the Standard mode I²C bus specification), before the SCL line is released.