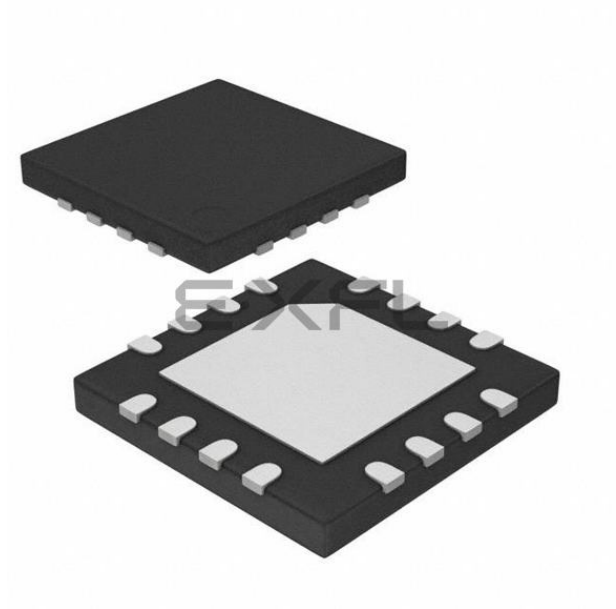


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	12
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 11x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	16-UQFN Exposed Pad
Supplier Device Package	16-UQFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18325-e-jq

TABLE 1-2: PIC16(L)F18325 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RC4/ANC4/T3G(1)/ SCK2(1)/ SCL2(1,3)/ CLCIN1(1)	RC4	TTL/ST	CMOS	General purpose I/O.
	ANC4	AN	—	ADC Channel C4 input.
	T3G	TTL/ST	—	TMR3 gate input.
	SCK2	TTL/ST	CMOS	SPI Clock 2.
	SCL2	I^2C	OD	I^2C Clock 2.
	CLCIN1	TTL/ST	—	Configurable Logic Cell 1 input.
RC5/ANC5/MDCIN2(1)/ T3CKI(1)/CCP1(1)/SDI2(1)/ SDA2(1,3)/RX(1)	RC5	TTL/ST	CMOS	General purpose I/O.
	ANC5	AN	—	ADC Channel C5 input.
	MDCIN2	TTL/ST	—	Modular Carrier input 2.
	T3CKI	TTL/ST	—	TMR3 Clock input.
	CCP1	TTL/ST	CMOS	Capture/Compare/PWM 1 input.
	SDI2	TTL/ST	CMOS	SPI Data 2.
	SDA2	I^2C	OD	I^2C Data 2.
	RX	TTL/ST	CMOS	EUSART asynchronous input.
VDD	VDD	Power	—	Positive supply.
VSS	VSS	Power	—	Ground reference.

Legend: AN = Analog input or output CMOS = CMOS compatible input or output OD = Open-Drain
TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I^2C = Schmitt Trigger input with I^2C
HV = High Voltage XTAL = Crystal levels

Note 1: Default peripheral input. Input can be moved to any other pin with the PPS input selection registers. See Register 13-1.

2: All pin outputs default to PORT latch data. Any pin can be selected as a digital peripheral output with the PPS output selection registers. See Register 13-2.

3: These I^2C functions are bidirectional. The output pin selections must be the same as the input pin selections.

TABLE 1-3: PIC16(L)F18345 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RA0/ANA0/C1IN0+/DAC1OUT/ ICDDAT/ICSPDAT	RA0	TTL/ST	CMOS	General purpose I/O.
	ANA0	AN	—	ADC Channel A0 input.
	C1IN0+	AN	—	Comparator C1 positive input.
	DAC1OUT	—	AN	Digital-to-Analog Converter output.
	ICDDAT	TTL/ST	CMOS	In-Circuit Debug Data I/O.
	ICSPDAT	TTL/ST	CMOS	ICSP™ Data I/O.
RA1/ANA1/VREF+/C1IN0-/ C2IN0-/ DAC1REF+/$\overline{SS2}$(1)/ ICDCLK/ ICSPCLK	RA1	TTL/ST	CMOS	General purpose I/O.
	ANA1	AN	—	ADC Channel A1 input.
	VREF+	AN	—	ADC positive voltage reference input.
	C1IN0-	AN	—	Comparator C1 negative input.
	C2IN0-	AN	—	Comparator C2 negative input.
	DAC1REF+	AN	—	Digital-to-Analog Converter positive reference input.
	$\overline{SS2}$	TTL/ST	—	Slave Select 2 input.
	ICDCLK	TTL/ST	CMOS	In-Circuit Debug Clock I/O.
	ICSPCLK	TTL/ST	CMOS	ICSPTM Clock I/O.
RA2/ANA2/VREF-/ DAC1REF-/ T0CKI(1)/ CCP3(1)/CWG1IN(1)/ CWG2IN(1)/CLCIN0(1)/ INT(1)	RA2	TTL/ST	CMOS	General purpose I/O.
	ANA2	AN	—	ADC Channel A2 input.
	VREF-	AN	—	ADC negative voltage reference input.
	DAC1REF-	AN	—	Digital-to-Analog Converter negative reference input.
	T0CKI	TTL/ST	—	TMR0 Clock input.
	CCP3	TTL/ST	CMOS	Capture/Compare/PWM 3 input.
	CWG1IN	TTL/ST	—	Complementary Waveform Generator 1 input.
	CWG2IN	TTL/ST	—	Complementary Waveform Generator 2 input.
	CLCIN0	TTL/ST	—	Configurable Logic Cell 0 input.
	INT	TTL/ST	—	External interrupt input.
RA3/$\overline{MCLR}$/VPP	RA3	TTL/ST	CMOS	General purpose I/O.
	$\overline{MCLR}$	TTL/ST	—	Master Clear with internal pull-up.
	VPP	HV	—	Programming voltage.
RA4/ANA4/T1G(1)/T3G(1)/T5G(1)/ SOSCO/CCP4(1)/CLKOUT/ OSC2	RA4	TTL/ST	CMOS	General purpose I/O.
	ANA4	AN	—	ADC Channel A4 input.
	T1G	TTL/ST	—	TMR1 gate input.
	T3G	TTL/ST	—	TMR3 gate input.
	T5G	TTL/ST	—	TMR5 gate input.
	SOSCO	—	XTAL	Secondary Oscillator connection.
	CCP4	TTL/ST	CMOS	Capture/Compare/PWM 4 input.
	CLKOUT	—	CMOS	FOSC/4 output.
	OSC2	—	XTAL	Crystal/Resonator (LP, XT, HS modes).

Legend: AN = Analog input or output CMOS= CMOS compatible input or output OD = Open-Drain
TTL = TTL compatible input ST = Schmitt Trigger input with CMOS levels I^2C = Schmitt Trigger input with I^2C
HV = High Voltage XTAL = Crystal levels

Note 1: Default peripheral input. Input can be moved to any other pin with the PPS input selection registers. See Register 13-2.

2: All pin outputs default to PORT latch data. Any pin can be selected as a digital peripheral output with the PPS output selection registers. See Register 13-2.

3: These I^2C functions are bidirectional. The output pin selections must be the same as the input pin selections.

9.0 POWER-SAVING OPERATION MODES

The purpose of the Power-Down modes is to reduce power consumption. There are three Power-Down modes: DOZE mode, IDLE mode, and Sleep mode.

9.1 DOZE Mode

Doze mode allows for power savings by reducing CPU operation and program memory access, without affecting peripheral operation. Doze mode differs from Sleep mode because the system oscillators continue to operate, while only the CPU and program memory are affected. The reduced execution saves power by eliminating unnecessary operations within the CPU and memory.

When the Doze Enable (DOZEN) bit is set (DOZEN = 1), the CPU executes only one instruction cycle out of every N cycles as defined by the DOZE<2:0> bits of the CPUDOZE register. For example, if DOZE<2:0> = 100, the instruction cycle ratio is 1:32. The CPU and memory execute for one instruction cycle and then lay idle for 31 instruction cycles. During the unused cycles, the peripherals continue to operate at the system clock speed.

9.1.1 DOZE OPERATION

The Doze operation is illustrated in Figure 9-1. For this example:

- Doze enable (DOZEN) bit set (DOZEN = 1)
- DOZE<2:0> = 001 (1:4) ratio
- Recover-on-Interrupt (ROI) bit set (ROI = 1)

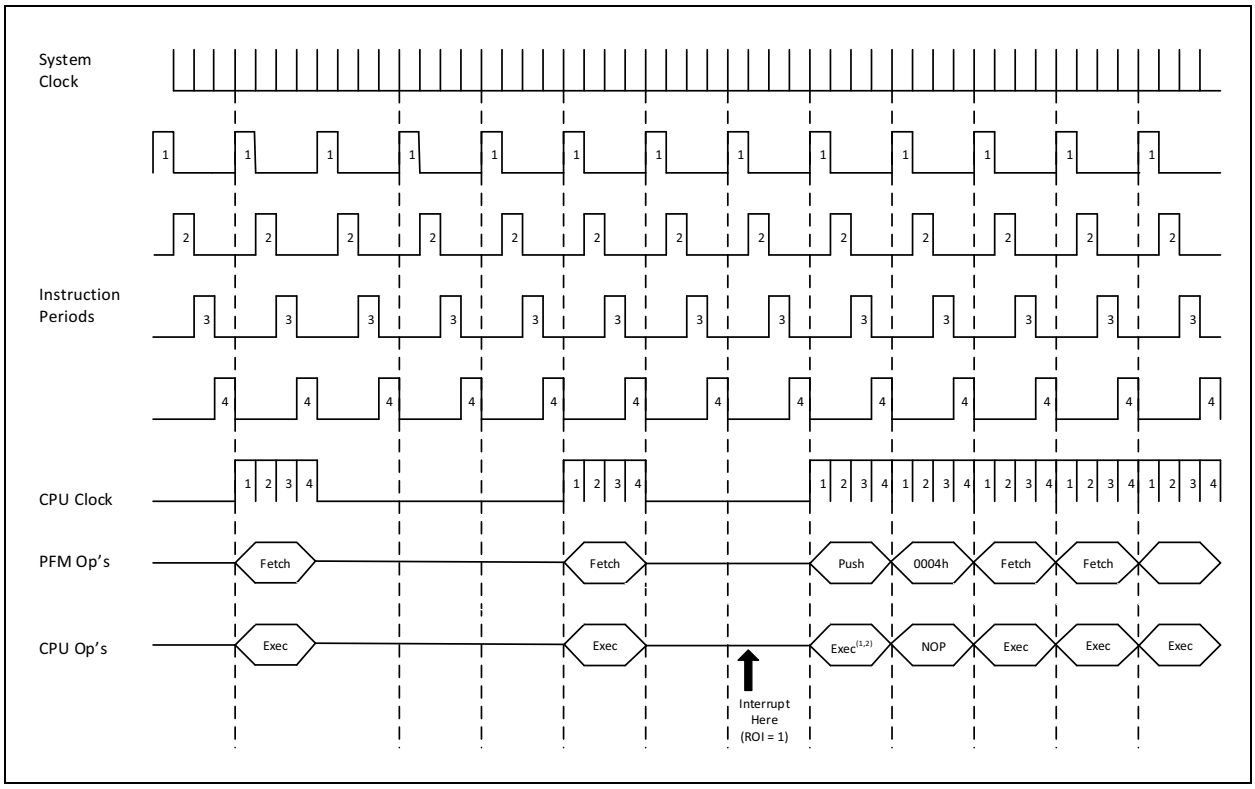
As with normal operation, the program memory fetches for the next instruction cycle. The instruction clocks to the peripherals continue throughout.

9.1.2 INTERRUPTS DURING DOZE

If an interrupt occurs and the Recover-on-Interrupt (ROI) bit is clear (ROI = 0) at the time of the interrupt, the Interrupt Service Routine (ISR) continues to execute at the rate selected by DOZE<2:0>. Interrupt latency is extended by the DOZE<2:0> ratio.

If an interrupt occurs and the ROI bit is set (ROI = 1) at the time of the interrupt, the DOZEN bit is cleared and the CPU executes at full speed. The prefetched instruction is executed and then the interrupt vector sequence is executed. In Figure 9-1, the interrupt occurs during the 2nd instruction cycle of the Doze period, and immediately brings the CPU out of Doze. If the Doze-on-Exit (DOE) bit is set (DOE = 1) when the RETFIE operation is executed, DOZEN is set, and the CPU executes at the reduced rate based on the DOZE<2:0> ratio.

FIGURE 9-1: DOZE MODE OPERATION EXAMPLE



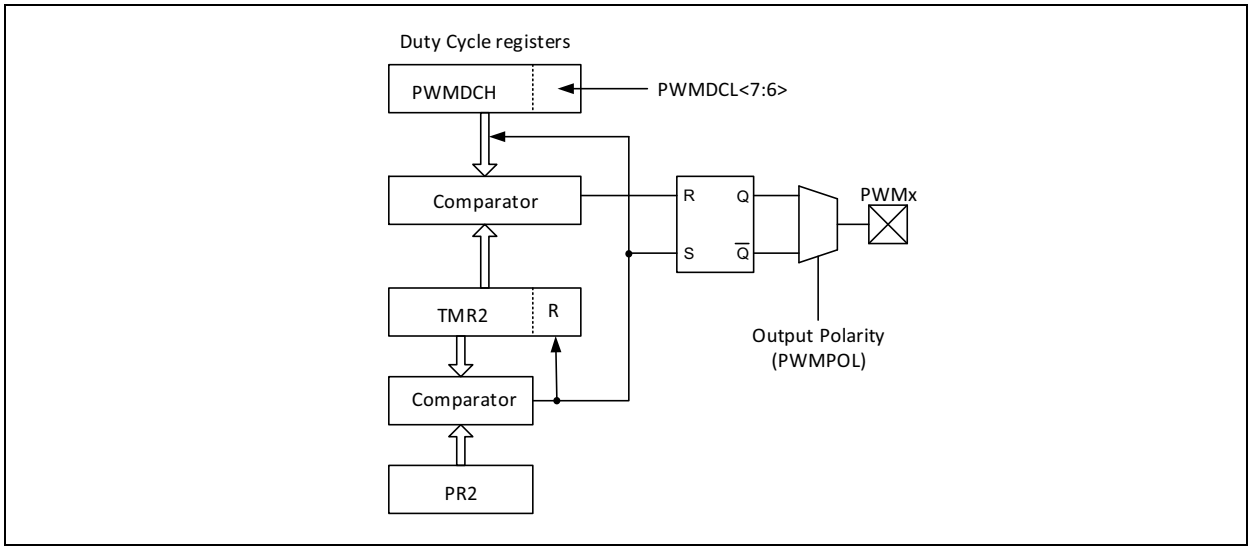
EXAMPLE 11-3: ERASING ONE ROW OF PROGRAM FLASH MEMORY

```
; This sample row erase routine assumes the following:
; 1.A valid address within the erase row is loaded in variables ADDRH:ADDRL
; 2.ADDRH and ADDRL are located in common RAM (locations 0x70 - 0x7F)

BANKSEL       NVMADRL
MOVF          ADDRL,W
MOVWF         NVMADRL              ; Load lower 8 bits of erase address boundary
MOVF          ADDRH,W
MOVWF         NVMADRH              ; Load upper 6 bits of erase address boundary
BCF           NVMCON1,NVMREGS      ; Choose Program Flash Memory area
BSF           NVMCON1,FREE         ; Specify an erase operation
BSF           NVMCON1,WREN         ; Enable writes
BCF           INTCON,GIE           ; Disable interrupts during unlock sequence

; ------------------------------REQUIRED UNLOCK SEQUENCE:------------------------------

MOVLW         55h                  ; Load 55h to get ready for unlock sequence
MOVWF         NVMCON2              ; First step is to load 55h into NVMCON2
MOVLW         AAh                  ; Second step is to load AAh into W
MOVWF         NVMCON2              ; Third step is to load AAh into NVMCON2
BSF           NVMCON1,WR           ; Final step is to set WR bit

; ------------------------------------------------------------------------------------

BSF           INTCON,GIE           ; Re-enable interrupts, erase is complete
BCF           NVMCON1,WREN         ; Disable writes
```

TABLE 11-2: NVM ORGANIZATION AND ACCESS INFORMATION

Master Values			NVMREG Access			FSR Access	
Memory Function	**Program Counter (PC), ICSP™ Address**	**Memory Type**	**NVMREGS bit (NVMCON1)**	**NVMADR <14:0>**	**Allowed Operations**	**FSR Address**	**FSR Programming Address**
Reset Vector	0000h	Program Flash Memory	0	0000h	READ WRITE	8000h	READ-ONLY
User Memory	0001h		0	0001h		8001h	
	0003h			0003h		8003h	
INT Vector	0004h		0	0004h		8004h	
User Memory	0005h		0	0005h		8005h	
	17FFh			17FFh		FFFFh	
User ID	No PC Address	Program Flash Memory	1	0000h	READ	No Access	
				0003h			
Reserved		—	—	0004h	—		
Rev ID		Program Flash Memory	1	0005h	READ		
Device ID			1	0006h			
CONFIG1			1	0007h			
CONFIG2			1	0008h			
CONFIG3			1	0009h			
CONFIG4				000Ah			
User Memory		EEPROM	1	7000h	READ	7000h	READ-ONLY
				70FFh	WRITE	70FFh	

FIGURE 19-2: SIMPLIFIED PWM BLOCK DIAGRAM



19.1.1 PWM PERIOD

Referring to Figure 19-1, the PWM output has a period and a pulse width. The frequency of the PWM is the inverse of the period (1/period).

The PWM period is specified by writing to the PR2 register. The PWM period can be calculated using the following formula:

EQUATION 19-1: PWM PERIOD

$$PWM\ Period = [(PR2) + 1] \bullet 4 \bullet T_{OSC} \bullet (TMR2\ Prescale\ Value)$$

Note: TOSC = 1/FOSC

When TMR2 is equal to PR2, the following three events occur on the next increment cycle:

- TMR2 is cleared
- The PWMx pin is set (Exception: If the PWM duty cycle = 0%, the pin will not be set.)
- The PWM pulse width is latched from PWMxDC.

Note: If the pulse width value is greater than the period the assigned PWM pin(s) will remain unchanged.

19.1.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing a 10-bit value to the PWMxDC register. The PWMxDCH contains the eight MSbs and bits <7:6> of the PWMxDCL register contain the two LSbs.

The PWMDC register is double-buffered and can be updated at any time. This double buffering is essential for glitch-free PWM operation. New values take effect when TMR2 = PR2. Note that PWMDC is left-justified.

The 8-bit timer TMR2 register is concatenated with either the 2-bit internal system clock (FOSC), or two bits of the prescaler, to create the 10-bit time base. The system clock is used if the Timer2 prescaler is set to 1:1.

Equation 19-2 is used to calculate the PWM pulse width.

Equation 19-3 is used to calculate the PWM duty cycle ratio.

EQUATION 19-2: PULSE WIDTH

$$Pulse\ Width = (PWMxDC) \bullet T_{OSC} \bullet \bullet (TMR2\ Prescale\ Value)$$

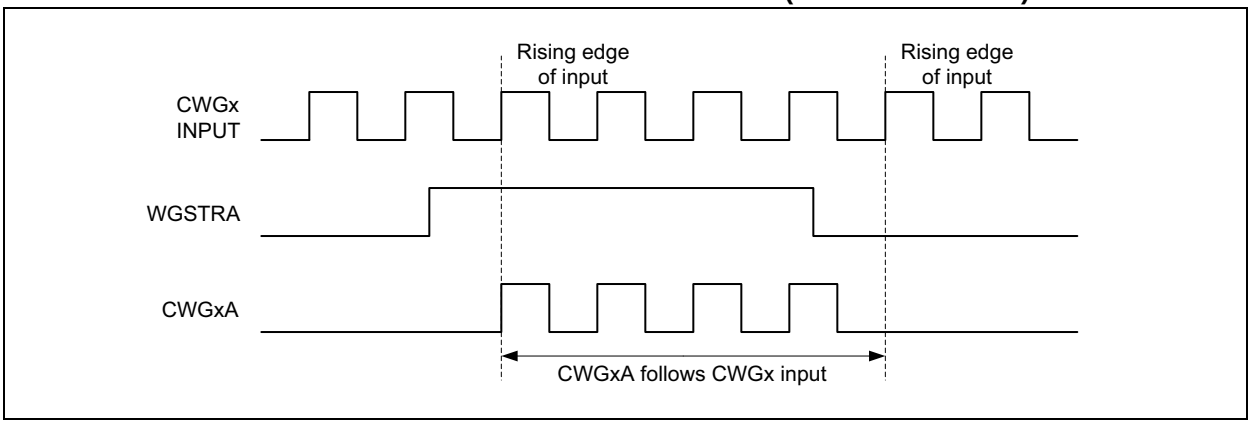
EQUATION 19-3: DUTY CYCLE RATIO

$$Duty\ Cycle\ Ratio = \frac{(PWMxDC)}{4(PR2 + 1)}$$

20.2.3.1 Synchronous Steering Mode

In Synchronous Steering mode (MODE<2:0> bits = 001, Register 20-1), changes to steering selection registers take effect on the next rising edge of the modulated data input (Figure 20-3). In Synchronous Steering mode, the output will always produce a complete waveform.

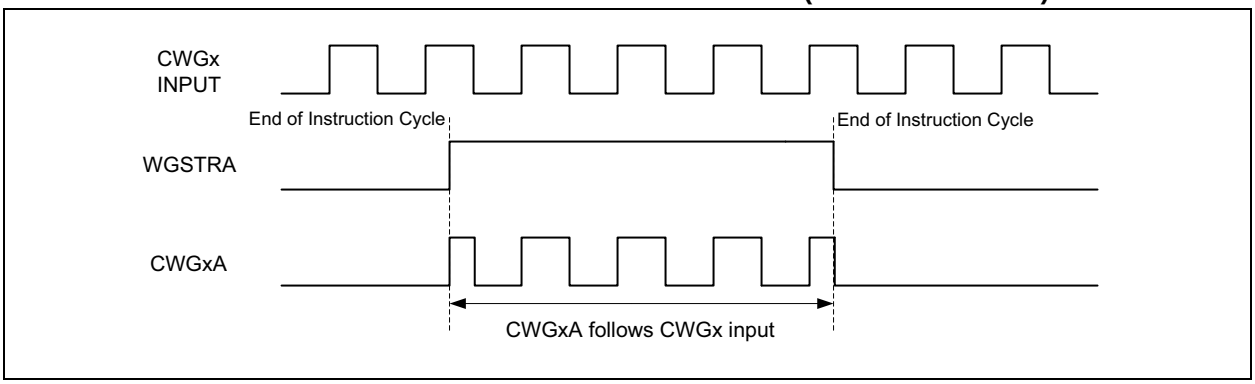
FIGURE 20-3: EXAMPLE OF SYNCHRONOUS STEERING (MODE<2:0> = 001)



20.2.3.2 Asynchronous Steering Mode

In Asynchronous mode (MODE<2:0> bits = 000, Register 20-1), steering takes effect at the end of the instruction cycle that writes to WGxSTR. In Asynchronous Steering mode, the output signal may be an incomplete waveform (Register 20-4). This operation may be useful when the user firmware needs to immediately remove a signal from the output pin.

FIGURE 20-4: EXAMPLE OF ASYNCHRONOUS STEERING (MODE<2:0> = 000)



20.2.3.3 Start-up Considerations

The application hardware must use the proper external pull-up and/or pull-down resistors on the CWG output pins. This is required because all I/O pins are forced to high-impedance at Reset.

The POLy bits (Register 20-2) allow the user to choose whether the output signals are active-high or active-low.

21.2 CLCx Interrupts

An interrupt will be generated upon a change in the output value of the CLCx when the appropriate interrupt enables are set. A rising edge detector and a falling edge detector are present in each CLC for this purpose.

The CLCxIF bit of the associated PIR3 register will be set when either edge detector is triggered and its associated enable bit is set. The LCxINTP bit enables rising edge interrupts and the LCxINTN bit enables falling edge interrupts. Both are located in the CLCxCON register.

To fully enable the interrupt, set the following bits:

- CLCxIE bit of the PIE3 register
- LCxINTP bit of the CLCxCON register (for a rising edge detection)
- LCxINTN bit of the CLCxCON register (for a falling edge detection)
- PEIE and GIE bits of the INTCON register

The CLCxIF bit of the PIR3 register, must be cleared in software as part of the interrupt service. If another edge is detected while this flag is being cleared, the flag will still be set at the end of the sequence.

21.3 Output Mirror Copies

Mirror copies of all LCxCON output bits are contained in the CLCDATA register. Reading this register samples the outputs of all CLCs simultaneously. This prevents any timing skew introduced by testing or reading the LCxOUT bits in the individual CLCxCON registers.

21.4 Effects of a Reset

The CLCxCON register is cleared to zero as the result of a Reset. All other selection and gating values remain unchanged.

21.5 Operation During Sleep

The CLC module operates independently from the system clock and will continue to run during Sleep, provided that the input sources selected remain active.

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

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

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

21.6 CLCx Setup Steps

The following steps should be followed when setting up the CLCx:

- Disable CLCx by clearing the LCxEN bit.
- Select desired inputs using CLCxSEL0 through CLCxSEL3 registers (See Table 21-1).
- Clear any associated ANSEL bits.
- Set all TRIS bits associated with external CLC inputs.
- Enable the chosen inputs through the four gates using CLCxGLS0, CLCxGLS1, CLCxGLS2, and CLCxGLS3 registers.
- Select the gate output polarities with the LCxGyPOL bits of the CLCxPOL register.
- Select the desired logic function with the LCxMODE<2:0> bits of the CLCxCON register.
- Select the desired polarity of the logic output with the LCxPOL bit of the CLCxPOL register. (This step may be combined with the previous gate output polarity step).
- If driving a device pin, set the desired pin PPS control register and also clear the TRIS bit corresponding to that output.
- If interrupts are desired, configure the following bits:
  - Set the LCxINTP bit in the CLCxCON register for rising event.
  - Set the LCxINTN bit in the CLCxCON register for falling event.
  - Set the CLCxIE bit of the PIE3 register.
  - Set the GIE and PEIE bits of the INTCON register.
- Enable the CLCx by setting the LCxEN bit of the CLCxCON register.

23.2 Fixed Duty Cycle (FDC) Mode

In Fixed Duty Cycle (FDC) mode, every time the accumulator overflows (NCO_overflow), the output is toggled. This provides a 50% duty cycle with a constant frequency, provided that the increment value remains constant. The FDC frequency can be calculated using Equation 23-2. The FDC frequency is half of the overflow frequency since it takes two overflow events to generate one FDC clock period. For more information, see Figure 23-2.

EQUATION 23-2: FDC FREQUENCY

$$F_{fdc} = F_{overflow} / 2$$

The FDC mode is selected by clearing the N1PFM bit in the NCO1CON register.

23.3 Pulse Frequency (PF) Mode

In Pulse Frequency (PF) mode, every time the accumulator overflows (NCO_overflow), the output becomes active for one or more clock periods. Once the clock period expires, the output returns to an inactive state. This provides a pulsed output. The output becomes active on the rising clock edge immediately following the overflow event. For more information, see Figure 23-2.

The value of the active and inactive states depends on the polarity bit, N1POL, in the NCO1CON register.

The PF mode is selected by setting the N1PFM bit in the NCO1CON register.

23.3.1 OUTPUT PULSE WIDTH CONTROL

When operating in PF mode, the active state of the output can vary in width by multiple clock periods. Various pulse widths are selected with the N1PWS<2:0> bits in the NCO1CLK register.

When the selected pulse width is greater than the accumulator overflow time frame, the output of the NCO1 does not toggle.

23.4 Output Polarity Control

The last stage in the NCO1 module is the output polarity. The N1POL bit in the NCO1CON register selects the output polarity. Changing the polarity while the interrupts are enabled will cause an interrupt for the resulting output transition.

The NCO1 output can be used internally by source code or other peripherals. Accomplish this by reading the N1OUT (read-only) bit of the NCO1CON register.

The NCO1 output signal is available to the following peripherals:

- CWG

REGISTER 23-4: NCO1ACCH: NCO1 ACCUMULATOR REGISTER – HIGH BYTE

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
NCO1ACC<15:8>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **NCO1ACC<15:8>:** NCO1 Accumulator, high byte

REGISTER 23-5: NCO1ACCU: NCO1 ACCUMULATOR REGISTER – UPPER BYTE(1)

U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	—	NCO1ACC<19:16>			
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-4 **Unimplemented:** Read as ‘0’

bit 3-0 **NCO1ACC<19:16>:** NCO1 Accumulator, upper byte

Note 1: The accumulator spans registers NCO1ACCU:NCO1ACCH:NCO1ACCL. The 24 bits are reserved but not all are used.This register updates in real-time, asynchronously to the CPU; there is no provision to guarantee atomic access to this 24-bit space using an 8-bit bus. Writing to this register while the module is operating will produce undefined results.

REGISTER 23-6: NCO1INCL(1,2): NCO1 INCREMENT REGISTER – LOW BYTE

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-1/1
NCO1INC<7:0>							
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as ‘0’
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
‘1’ = Bit is set	‘0’ = Bit is cleared	

bit 7-0 **NCO1INC<7:0>:** NCO1 Increment, low byte

Note 1: The logical increment spans NCO1INCU:NCO1INCH:NCO1INCL.

2: NCO1INC is double-buffered as INCBUF; INCBUF is updated on the next falling edge of NCOCLK after writing to NCO1INCL;NCO1INCU and NCO1INCH should be written prior to writing NCO1INCL.

30.6 I²C Master Mode

Master mode is enabled by setting and clearing the appropriate SSPM<3:0> bits in the SSPxCON1 register and by setting the SSPEN bit. In Master mode, the SDA and SCK pins must be configured as inputs. The MSSP peripheral hardware will override the output driver TRIS controls when necessary to drive the pins low.

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 MSSPx module is disabled. Control of the I²C bus may be taken when the P bit is set, or the bus is Idle.

In Firmware Controlled Master mode, user code conducts all I²C bus operations based on Start and Stop bit condition detection. Start and Stop condition detection is the only active circuitry in this mode. All other communication is done by the user software directly manipulating the SDA and SCL lines.

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

- Start condition generation
- Stop condition generation
- Data transfer byte transmitted/received
- Acknowledge transmitted/received
- Repeated Start generated

Note 1: The MSSPx module, when configured in I²C Master mode, does not allow queuing of events. For instance, the user is not allowed to initiate a Start condition and immediately write the SSPxBUF register to initiate transmission before the Start condition is complete. In this case, the SSPxBUF will not be written to and the WCOL bit will be set, indicating that a write to the SSPxBUF did not occur

2: When in Master mode, Start/Stop detection is masked and an interrupt is generated when the SEN/PEN bit is cleared and the generation is complete.

30.6.1 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 SDA, while SCL 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 eight 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 SDA, while SCL outputs the serial clock. Serial data is received eight 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.

A Baud Rate Generator is used to set the clock frequency output on SCL. See **Section 30.7 "Baud Rate Generator"** for more detail.

30.6.4 I²C MASTER MODE START CONDITION TIMING

To initiate a Start condition (Figure 30-26), the user sets the Start Enable bit, SEN bit of the SSPxCON2 register. If the SDA and SCL pins are sampled high, the Baud Rate Generator is reloaded with the contents of SSPxADD<7:0> and starts its count. If SCL and SDA are both sampled high when the Baud Rate Generator times out (TBRG), the SDA pin is driven low. The action of the SDA being driven low while SCL is high is the Start condition and causes the S bit of the SSPxSTAT register to be set. Following this, the Baud Rate Generator is reloaded with the contents of SSPxADD<7:0> and resumes its count. When the Baud Rate Generator times out (TBRG), the SEN bit of the SSPxCON2 register will be automatically cleared by hardware; the Baud Rate Generator is suspended, leaving the SDA line held low and the Start condition is complete.

Note 1: If at the beginning of the Start condition, the SDA and SCL pins are already sampled low, or if during the Start condition, the SCL line is sampled low before the SDA line is driven low, a bus collision occurs, the Bus Collision Interrupt Flag, BCLIF, is set, the Start condition is aborted and the I²C module is reset into its idle state.

2: The Philips I²C specification states that a bus collision cannot occur on a Start.

FIGURE 30-26: FIRST START BIT TIMING

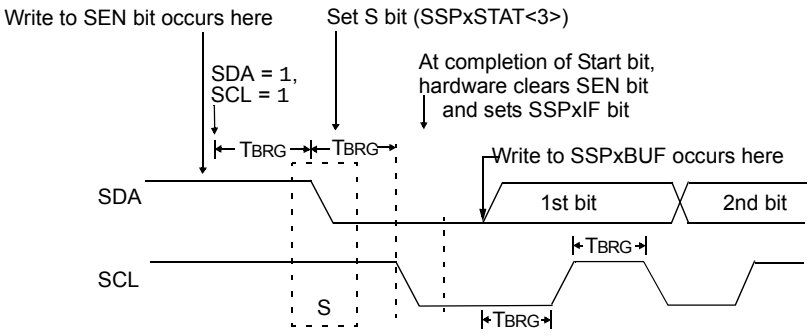
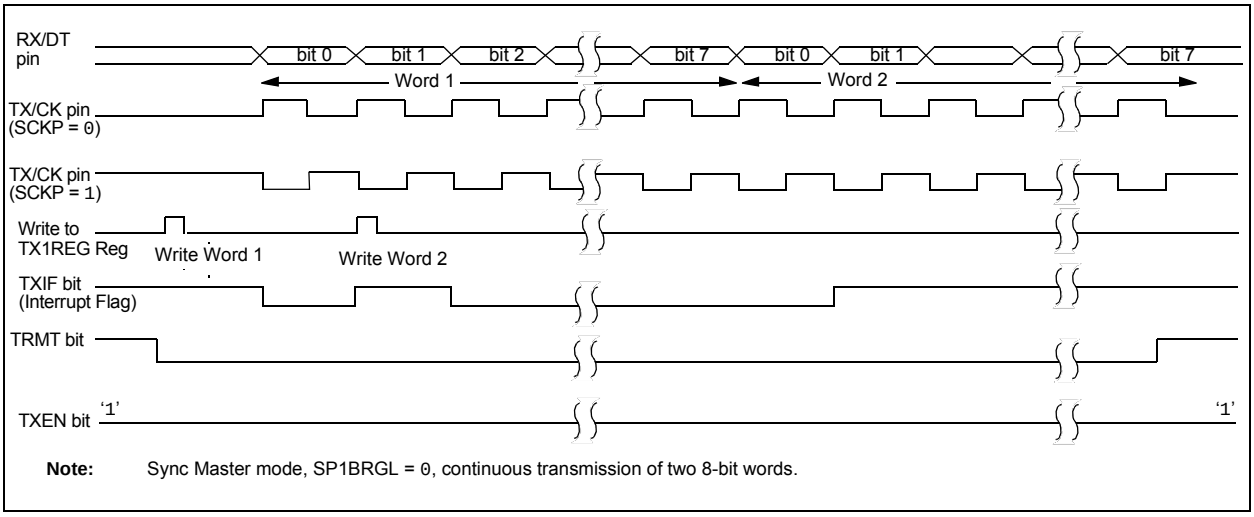
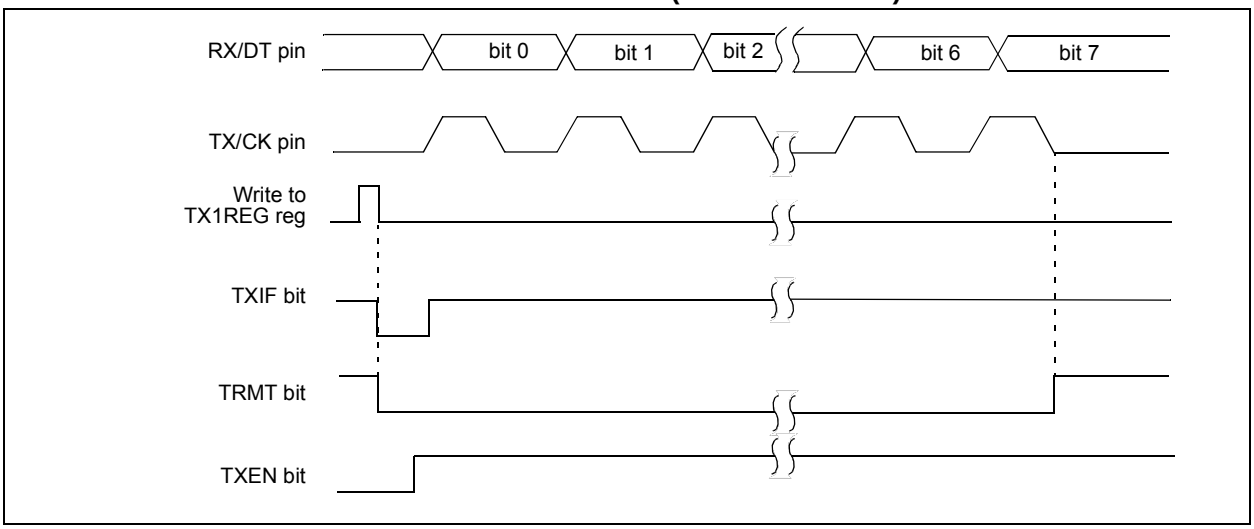


FIGURE 31-10: SYNCHRONOUS TRANSMISSION



Note: Sync Master mode, SP1BRGL = 0, continuous transmission of two 8-bit words.

FIGURE 31-11: SYNCHRONOUS TRANSMISSION (THROUGH TXEN)



31.4.1.5 Synchronous Master Reception

Data is received at the RX/DT pin. The RX/DT pin output driver is automatically disabled when the EUSART1 is configured for synchronous master receive operation.

In Synchronous mode, reception is enabled by setting either the Single Receive Enable bit (SREN of the RC1STA register) or the Continuous Receive Enable bit (CREN of the RC1STA register).

When SREN is set and CREN is clear, only as many clock cycles are generated as there are data bits in a single character. The SREN bit is automatically cleared at the completion of one character. When CREN is set, clocks are continuously generated until CREN is cleared. If CREN is cleared in the middle of a character the CK clock stops immediately and the partial character is discarded. If SREN and CREN are both set, then SREN is cleared at the completion of the first character and CREN takes precedence.

To initiate reception, set either SREN or CREN. Data is sampled at the RX/DT pin on the trailing edge of the TX/CK clock pin and is shifted into the Receive Shift Register (RSR). When a complete character is received into the RSR, the RCIF bit is set and the character is automatically transferred to the two character receive FIFO. The Least Significant eight bits of the top character in the receive FIFO are available in RC1REG. The RCIF bit remains set as long as there are unread characters in the receive FIFO.

Note:	If the RX/DT function is on an analog pin, the corresponding ANSEL bit must be cleared for the receiver to function.

31.6 Register Definitions: EUSART1 Control

REGISTER 31-1: TX1STA: TRANSMIT STATUS AND CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-1/1	R/W-0/0
CSRC	TX9	TXEN(1)	SYNC	SENDB	BRGH	TRMT	TX9D
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **CSRC:** Clock Source Select bit
Asynchronous mode:
Unused in this mode – value ignored
Synchronous mode:
1 = Master mode (clock generated internally from BRG)
0 = Slave mode (clock from external source)

bit 6 **TX9:** 9-bit Transmit Enable bit
1 = Selects 9-bit transmission
0 = Selects 8-bit transmission

bit 5 **TXEN:** Transmit Enable bit(1)
1 = Transmit enabled
0 = Transmit disabled

bit 4 **SYNC:** EUSART1 Mode Select bit
1 = Synchronous mode
0 = Asynchronous mode

bit 3 **SENDB:** Send Break Character bit
Asynchronous mode:
1 = Send SYNCH BREAK on next transmission – Start bit, followed by 12 '0' bits, followed by Stop bit; cleared by hardware upon completion
0 = SYNCH BREAK transmission disabled or completed
Synchronous mode:
Unused in this mode – value ignored

bit 2 **BRGH:** High Baud Rate Select bit
Asynchronous mode:
1 = High speed
0 = Low speed
Synchronous mode:
Unused in this mode – value ignored

bit 1 **TRMT:** Transmit Shift Register Status bit
1 = TSR empty
0 = TSR full

bit 0 **TX9D:** Ninth bit of Transmit Data
Can be address/data bit or a parity bit.

Note 1: SREN/CREN overrides TXEN in Sync mode.

REGISTER 31-2: RC1STA: RECEIVE STATUS AND CONTROL REGISTER

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-0/0	R-0/0	R-x/x
SPEN(1)	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
bit 7							bit 0

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **SPEN:** Serial Port Enable bit(1)
- 1 = Serial port enabled
- 0 = Serial port disabled (held in Reset)

bit 6 **RX9:** 9-bit Receive Enable bit
- 1 = Selects 9-bit reception
- 0 = Selects 8-bit reception

bit 5 **SREN:** Single Receive Enable bit

Asynchronous mode:

Unused in this mode – value ignored

Synchronous mode – Master:
- 1 = Enables single receive
- 0 = Disables single receive

This bit is cleared after reception is complete.

Synchronous mode – Slave

Unused in this mode – value ignored

bit 4 **CREN:** Continuous Receive Enable bit

Asynchronous mode:
- 1 = Enables continuous receive until enable bit CREN is cleared
- 0 = Disables continuous receive

Synchronous mode:
- 1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN)
- 0 = Disables continuous receive

bit 3 **ADDEN:** Address Detect Enable bit

Asynchronous mode 9-bit (RX9 = 1):
- 1 = Enables address detection – enable interrupt and load of the receive buffer when the ninth bit in the receive buffer is set
- 0 = Disables address detection, all bytes are received and ninth bit can be used as parity bit

Asynchronous mode 8-bit (RX9 = 0):

Unused in this mode – value ignored

bit 2 **FERR:** Framing Error bit
- 1 = Framing error (can be updated by reading RC1REG register and receive next valid byte)
- 0 = No framing error

bit 1 **OERR:** Overrun Error bit
- 1 = Overrun error (can be cleared by clearing bit CREN)
- 0 = No overrun error

bit 0 **RX9D:** Ninth bit of Received Data

This can be address/data bit or a parity bit and must be calculated by user firmware.

Note 1: The EUSART1 module automatically changes the pin from tri-state to drive as needed. Configure the associated TRIS bits for TX/CK and RX/DT to 1.

TABLE 35-7: EXTERNAL CLOCK/OSCILLATOR TIMING REQUIREMENTS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	**Sym.**	**Characteristic**	**Min.**	**Typ.†**	**Max.**	**Units**	**Conditions**
ECL Oscillator							
OS1	FECL	Clock Frequency	—	—	500	kHz	
OS2	TECL_DC	Clock Duty Cycle	40	—	60	%	
ECM Oscillator							
OS3	FECM	Clock Frequency	—	—	4	MHz	**Note 4**
OS4	TECM_DC	Clock Duty Cycle	40	—	60	%	
ECH Oscillator							
OS5	FECH	Clock Frequency	—	—	32	MHz	
OS6	TECH_DC	Clock Duty Cycle	40	—	60	%	
LP Oscillator							
OS7	FLP	Clock Frequency	—	—	100	kHz	**Note 4**
XT Oscillator							
OS8	FXT	Clock Frequency	—	—	4	MHz	**Note 4**
HS Oscillator							
OS9	FHS	Clock Frequency	—	—	20	MHz	**Note 4**
System Clock							
OS20	FOSC	System Clock Frequency	—	—	32	MHz	**Note 2, Note 3**
OS21	FCY	Instruction Frequency	—	FOSC/4	—	MHz	
OS22	TCY	Instruction Period	125	1/FCY	—	ns	

* These parameters are characterized but not tested.

† Data in "Typ." column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Instruction cycle period (TCY) equals four times the input oscillator time base period. 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 OSC1 pin. When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

2: The system clock frequency (FOSC) is selected by the "main clock switch controls" as described in **Section 7.3 "Clock Switching"**.

3: The system clock frequency (FOSC) must meet the voltage requirements defined in the **Section 35.2 "Standard Operating Conditions"**. LP, XT and HS oscillator modes require an appropriate crystal or resonator to be connected to the device.

4: For clocking the device with an external square wave, one of the EC mode selections must be used.

TABLE 35-10: CLKOUT AND I/O TIMING SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	**Sym.**	**Characteristic**	**Min.**	**Typ.†**	**Max.**	**Units**	**Conditions**
IO1	TCLKOUTH	CLKOUT rising edge delay (rising edge FOSC (Q1 cycle) to falling edge CLKOUT	—	—	—	ns	
IO2	TCLKOUTL	CLKOUT falling edge delay (rising edge FOSC (Q3 cycle) to rising edge CLKOUT	—	—	—	ns	
IO3	TIO_VALID	Port output valid time (rising edge FOSC (Q1 cycle) to port valid)	—	—	—	ns	
IO4	TIO_SETUP	Port input setup time (Setup time before rising edge FOSC - Q2 cycle)	—	—	—	ns	
IO5	TIO_HOLD	Port input hold time (Hold time after rising edge FOSC - Q2 cycle)	—	—	—	ns	
IO6	TIOR_SLREN	Port I/O rise time, slew rate enabled	—	—	—	ns	VDD = 3.0V, Load conditions
IO7	TIOR_SLRDIS	Port I/O rise time, slew rate disabled	—	—	—	ns	VDD = 3.0V, Load conditions
IO8	TIOF_SLREN	Port I/O fall time, slew rate enabled	—	—	—	ns	VDD = 3.0V, Load conditions
IO9	TIOF_SLRDIS	Port I/O fall time, slew rate disabled	—	—	—	ns	VDD = 3.0V, Load conditions
IO10	TINT	INT pin high or low time to trigger an interrupt	—	—	—	ns	
IO11	TIOC	Interrupt-on-Change minimum high or low time to trigger interrupt	—	—	—	ns	

* These parameters are characterized but not tested.

† Data in “Typ.” column is at 3.0V, 25°C unless otherwise stated.

FIGURE 35-19: SPI SLAVE MODE TIMING (CKE = 0)

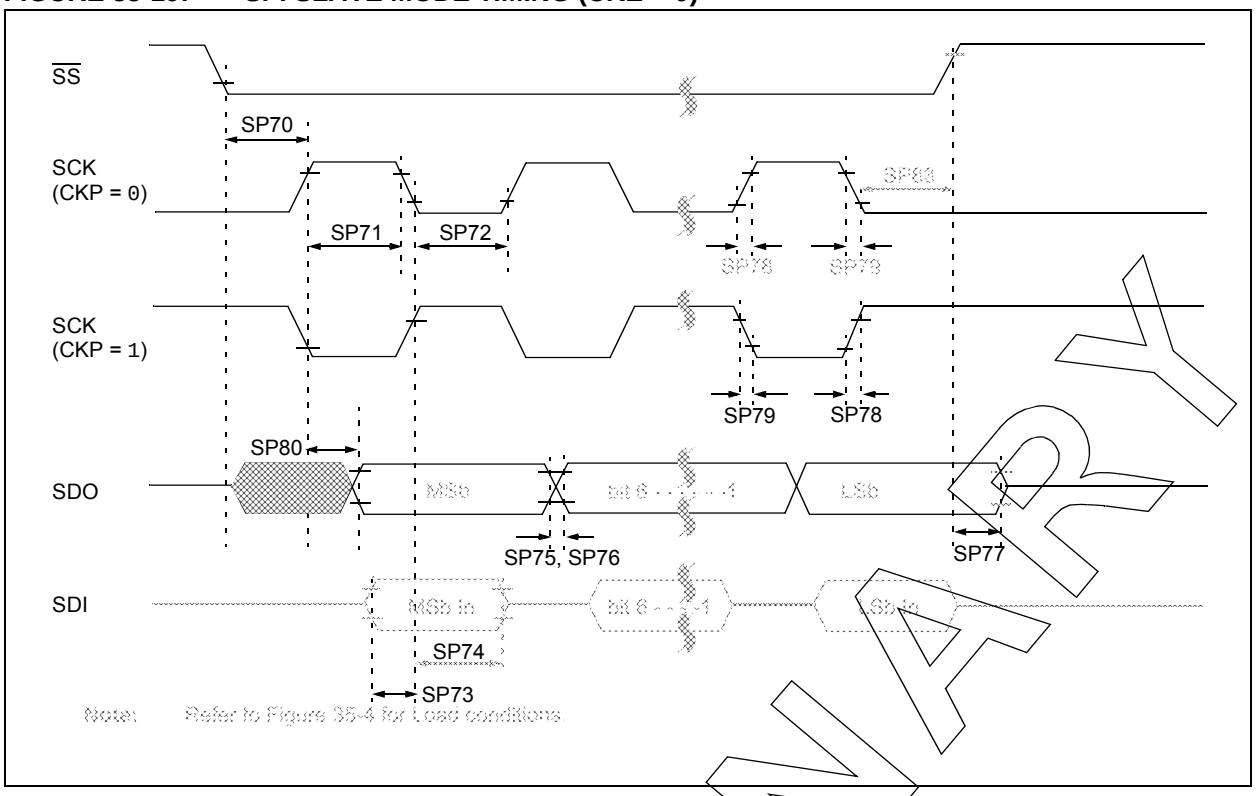
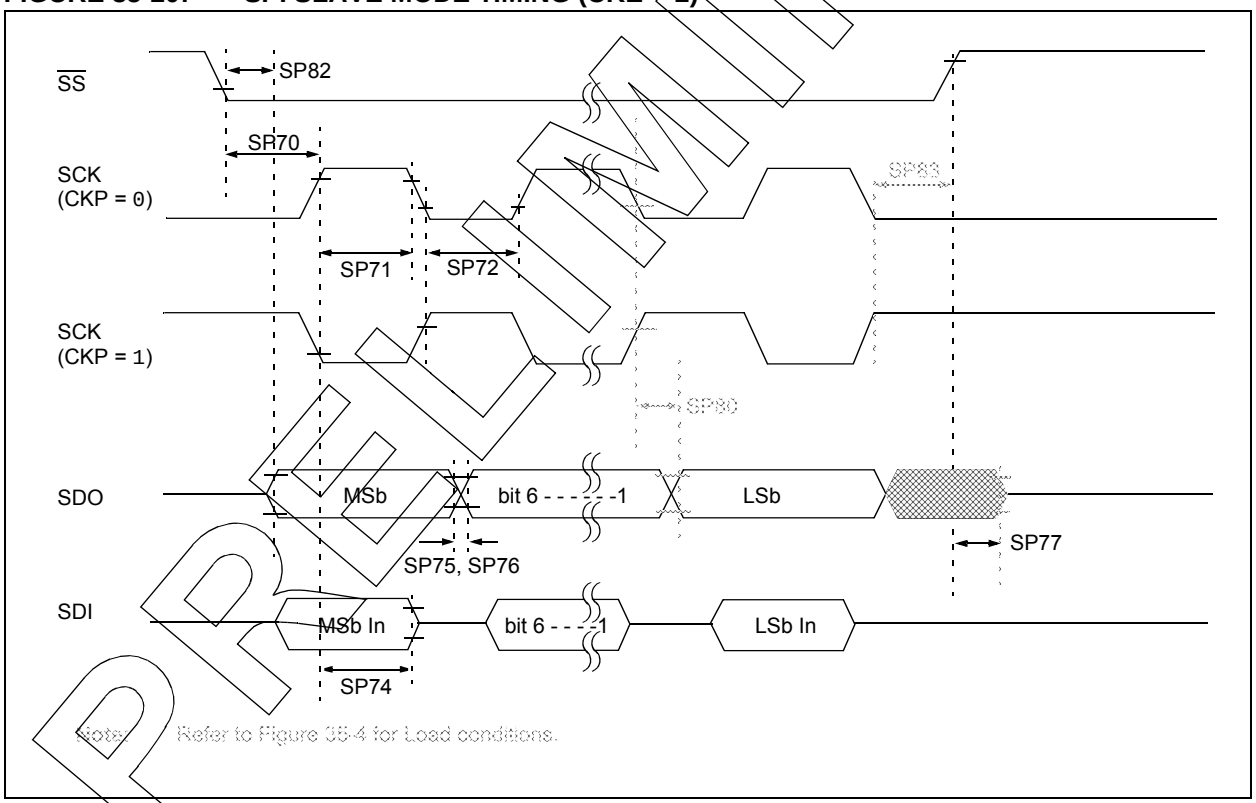


FIGURE 35-20: SPI SLAVE MODE TIMING (CKE = 1)



37.0 DEVELOPMENT SUPPORT

The PIC® microcontrollers (MCU) and dsPIC® digital signal controllers (DSC) are supported with a full range of software and hardware development tools:

- Integrated Development Environment
  - MPLAB® X IDE Software
  - MPLAB Xpress IDE Software
  - Microchip Code Configurator (MCC)
- Compilers/Assemblers/Linkers
  - MPLAB XC Compiler
  - MPASM™ Assembler
  - MPLINK™ Object Linker/ MPLIB™ Object Librarian
  - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
  - MPLAB X SIM Software Simulator
- Emulators
  - MPLAB REAL ICE™ In-Circuit Emulator
- In-Circuit Debuggers/Programmers
  - MPLAB ICD 3
  - PICkit™ 3
- Device Programmers
  - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits and Starter Kits
- Third-party development tools

37.1 MPLAB X Integrated Development Environment Software

The MPLAB X IDE is a single, unified graphical user interface for Microchip and third-party software, and hardware development tool that runs on Windows®, Linux and Mac OS® X. Based on the NetBeans IDE, MPLAB X IDE is an entirely new IDE with a host of free software components and plug-ins for high-performance application development and debugging. Moving between tools and upgrading from software simulators to hardware debugging and programming tools is simple with the seamless user interface.

With complete project management, visual call graphs, a configurable watch window and a feature-rich editor that includes code completion and context menus, MPLAB X IDE is flexible and friendly enough for new users. With the ability to support multiple tools on multiple projects with simultaneous debugging, MPLAB X IDE is also suitable for the needs of experienced users.

Feature-Rich Editor:

- Color syntax highlighting
- Smart code completion makes suggestions and provides hints as you type
- Automatic code formatting based on user-defined rules
- Live parsing

User-Friendly, Customizable Interface:

- Fully customizable interface: toolbars, toolbar buttons, windows, window placement, etc.
- Call graph window

Project-Based Workspaces:

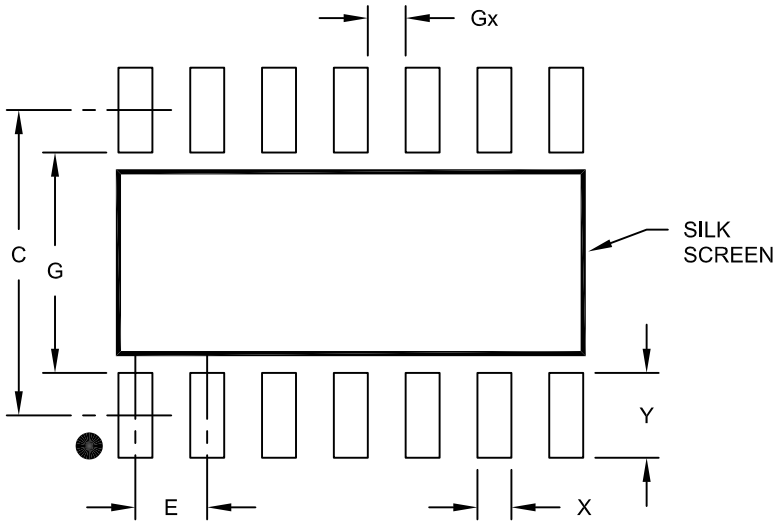
- Multiple projects
- Multiple tools
- Multiple configurations
- Simultaneous debugging sessions

File History and Bug Tracking:

- Local file history feature
- Built-in support for Bugzilla issue tracker

14-Lead Plastic Small Outline (SL) - Narrow, 3.90 mm Body [SOIC]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	1.27 BSC		
Contact Pad Spacing	C		5.40	
Contact Pad Width	X			0.60
Contact Pad Length	Y			1.50
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	3.90		

Notes:

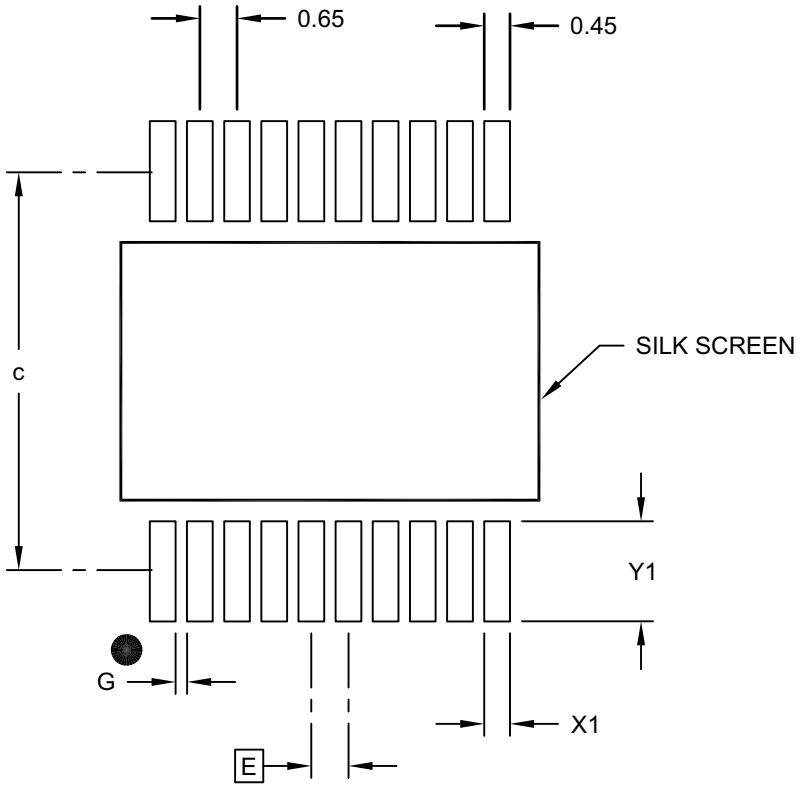
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2065A

20-Lead Plastic Shrink Small Outline (SS) - 5.30 mm Body [SSOP]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Contact Pad Spacing	C		7.20	
Contact Pad Width (X20)	X1			0.45
Contact Pad Length (X20)	Y1			1.75
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2072B