Welcome to E-XFL.COM

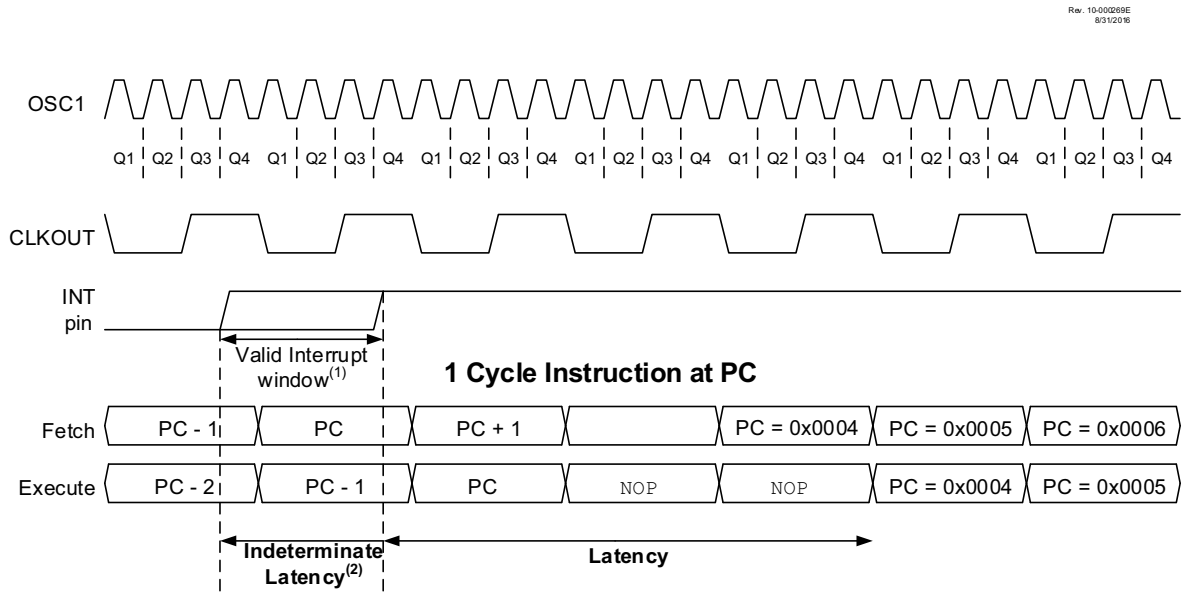
What is "Embedded - Microcontrollers"?

"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"

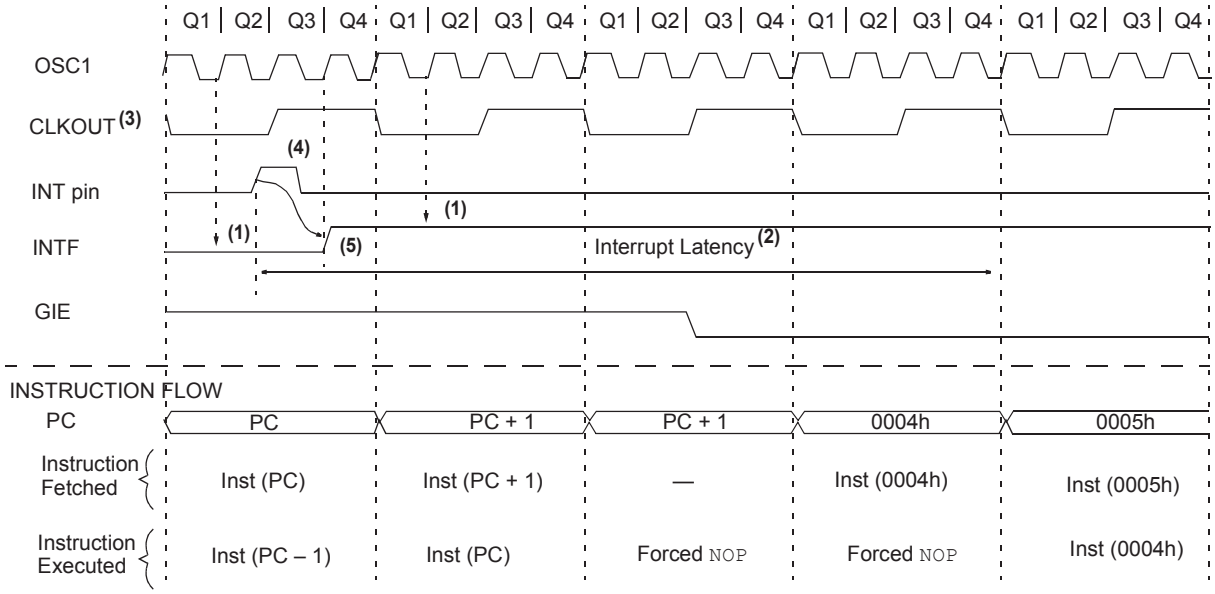
Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I²C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	12
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 11x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	14-SOIC (0.154", 3.90mm Width)
Supplier Device Package	14-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18326-i-sl

FIGURE 8-2: INTERRUPT LATENCY



Note 1: An interrupt may occur at any time during the interrupt window.
2: Since an interrupt may occur any time during the interrupt window, the actual latency can vary.

FIGURE 8-3: INT PIN INTERRUPT TIMING



Note 1: INTF flag is sampled here (every Q1).

2: Asynchronous interrupt latency = 3-5 TCY. Synchronous latency = 3-4 TCY, where TCY = instruction cycle time. Latency is the same whether Inst (PC) is a single cycle or a 2-cycle instruction.

3: CLKOUT not available in all oscillator modes.

4: For minimum width of INT pulse, refer to AC specifications in **Section 35.0 "Electrical Specifications"**.

5: INTF is enabled to be set any time during the Q4-Q1 cycles.

REGISTER 8-6: PIE4: PERIPHERAL INTERRUPT ENABLE REGISTER 4

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
CWG2IE	CWG1IE	TMR5GIE	TMR5IE	CCP4IE	CCP3IE	CCP2IE	CCP1IE
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	HS = Hardware set

bit 7 **CWG2IE:** CWG 2 Interrupt Enable bit
1 = CWG2 interrupt enabled
0 = CWG2 interrupt not enabled

bit 6 **CWG1IE:** CWG 1 Interrupt Enable bit
1 = CWG1 interrupt enabled
0 = CWG1 interrupt not enabled

bit 5 **TMR5GIE:** Timer5 Gate Interrupt Enable bit
1 = TMR5 Gate interrupt is enabled
0 = TMR5 Gate interrupt is not enabled

bit 4 **TMR5IE:** TMR5 Overflow Interrupt Enable bit
1 = TMR5 overflow interrupt is enabled
0 = TMR5 overflow interrupt is not enabled

bit 3 **CCP4IE:** CCP4 Interrupt Enable bit
1 = CCP4 interrupt is enabled
0 = CCP4 interrupt is not enabled

bit 2 **CCP3IE:** CCP3 Interrupt Enable bit
1 = CCP3 interrupt is enabled
0 = CCP3 interrupt is not enabled

bit 1 **CCP2IE:** CCP2 Interrupt Enable bit
1 = CCP2 interrupt is enabled
0 = CCP2 interrupt is not enabled

bit 0 **CCP1IE:** CCP1 Interrupt Enable bit
1 = CCP1 interrupt is enabled
0 = CCP1 interrupt is not enabled

Note: Bit PEIE of the INTCON register must be set to enable any peripheral interrupt.

11.0 NONVOLATILE MEMORY (NVM) CONTROL

NVM is separated into two types: Program Flash Memory and Data EEPROM.

NVM is accessible by using both the FSR and INDF registers, or through the NVMREG register interface.

The write time is controlled by an on-chip timer. The write/erase voltages are generated by an on-chip charge pump rated to operate over the operating voltage range of the device.

NVM can be protected in two ways; by either code protection or write protection.

Code protection ($\overline{CP}$ and $\overline{CPD}$ bits in Configuration Word 4) disables access, reading and writing, to both the Program Flash Memory and EEPROM via external device programmers. Code protection does not affect the self-write and erase functionality. Code protection can only be Reset by a device programmer performing a Bulk Erase to the device, clearing all nonvolatile memory, Configuration bits, and User IDs.

Write protection prohibits self-write and erase to a portion or all of the Program Flash Memory, as defined by the WRT<1:0> bits of Configuration Word 3. Write protection does not affect a device programmer's ability to read, write, or erase the device.

11.1 Program Flash Memory

Program Flash Memory consists of 16,384 14-bit words as user memory, with additional words for User ID information, Configuration Words, and interrupt vectors. Program Flash Memory provides storage locations for:

- User program instructions
- User defined data

Program Flash Memory data can be read and/or written to through:

- CPU instruction fetch (read-only)
- FSR/INDF indirect access (read-only) (**Section 11.3 "FSR and INDF Access"**)
- NVMREG access (**Section 11.4 "NVMREG Access"**
- In-Circuit Serial Programming™ (ICSP™)

Read operations return a single word of memory. When write and erase operations are done on a row basis, the row size is defined in Table 11-1. Program Flash Memory will erase to a logic '1' and program to a logic '0'.

TABLE 11-1: FLASH MEMORY ORGANIZATION BY DEVICE

Device	Row Erase (words)	Write Latches (words)
PIC16(L)F18326	32	32
PIC16(L)F18346		

It is important to understand the Program Flash Memory structure for erase and programming operations. Program Flash Memory is arranged in rows. A row consists of 32 14-bit program memory words. A row is the minimum size that can be erased by user software.

All or a portion of a row can be programmed. Data to be written into the program memory row is written to 14-bit wide data write latches. These latches are not directly accessible to the user, but may be loaded via sequential writes to the NVMDATH:NVMDATL register pair.

Note: To modify only a portion of a previously programmed row, then the contents of the entire row must be read and saved in RAM prior to the erase. Then, the new data and retained data can be written into the write latches to reprogram the row of Program Flash Memory. Any unprogrammed locations can be written without first erasing the row. In this case, it is not necessary to save and rewrite the other previously programmed locations

11.1.1 PROGRAM MEMORY VOLTAGES

The Program Flash Memory is readable and writable during normal operation over the full V_{DD} range.

11.1.1.1 Programming Externally

The program memory cell and control logic support write and Bulk Erase operations down to the minimum device operating voltage.

11.1.1.2 Self-Programming

The program memory cell and control logic will support write and row erase operations across the entire V_{DD} range. Bulk Erase is not supported when self-programming.

TABLE 12-4: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
PORTB	RB7	RB6	RB5	RB4	—	—	—	—	149
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	149
LATB	LATB7	LATB6	LATB5	LATB4	—	—	—	—	150
ANSELB	ANSB7	ANSB6	ANSB5	ANSB4	—	—	—	—	150
WPUB	WPUB7	WPU6	WPUB5	WPUB4	—	—	—	—	151
ODCONB	ODCB7	ODCB6	ODCB5	ODCB4	—	—	—	—	151
SLRCONB	SLRB7	SLRB6	SLRB5	SLRB4	—	—	—	—	152
INLVLB	INLVLB7	INLVLB6	INLVLB5	INLVLB4	—	—	—	—	152

Legend: x = unknown, u = unchanged, – = unimplemented locations read as '0'. Shaded cells are not used by PORTB.

REGISTER 12-22: ODCONC: PORTC OPEN-DRAIN 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/W-0/0	R/W-0/0
ODCC7(1)	ODCC6(1)	ODCC5	ODCC4	ODCC3	ODCC2	ODCC1	ODCC0
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 7-6 **ODCC<7:6>**: PORTC Open-Drain Enable bits(1)
For RC<7:6> pins, respectively
1 = Port pin operates as open-drain drive (sink current only)
0 = Port pin operates as standard push-pull drive (source and sink current)

bit 5-0 **ODCC<5:0>:** PORTC Open-Drain Enable bits
For RC<5:0> pins, respectively
1 = Port pin operates as open-drain drive (sink current only)
0 = Port pin operates as standard push-pull drive (source and sink current)

Note 1: PIC16(L)F18346 only; otherwise read as '0'.

REGISTER 12-23: SLRCONC: PORTC SLEW RATE CONTROL REGISTER

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
SLRC7(1)	SLRC6(1)	SLRC5	SLRC4	SLRC3	SLRC2	SLRC1	SLRC0
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 7-6 **SLRC<7:6>:** PORTC Slew Rate Enable bits(1)
For RC<7:6> pins, respectively
1 = Port pin slew rate is limited
0 = Port pin slews at maximum rate

bit 5-0 **SLRC<5:0>:** PORTC Slew Rate Enable bits
For RC<5:0> pins, respectively
1 = Port pin slew rate is limited
0 = Port pin slews at maximum rate

Note 1: PIC16(L)F18346 only; otherwise read as '0'.

20.11 Register Definitions: CWG Control

REGISTER 20-1: CWGxCON0: CWGx CONTROL REGISTER 0

R/W-0/0	R/W/HC-0/0	U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
EN	LD(1)	—	—	—	MODE<2:0>		
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	HS/HC = Bit is set/cleared by hardware

bit 7 **EN:** CWGx Enable bit
- 1 = CWGx is enabled
- 0 = CWGx is disabled

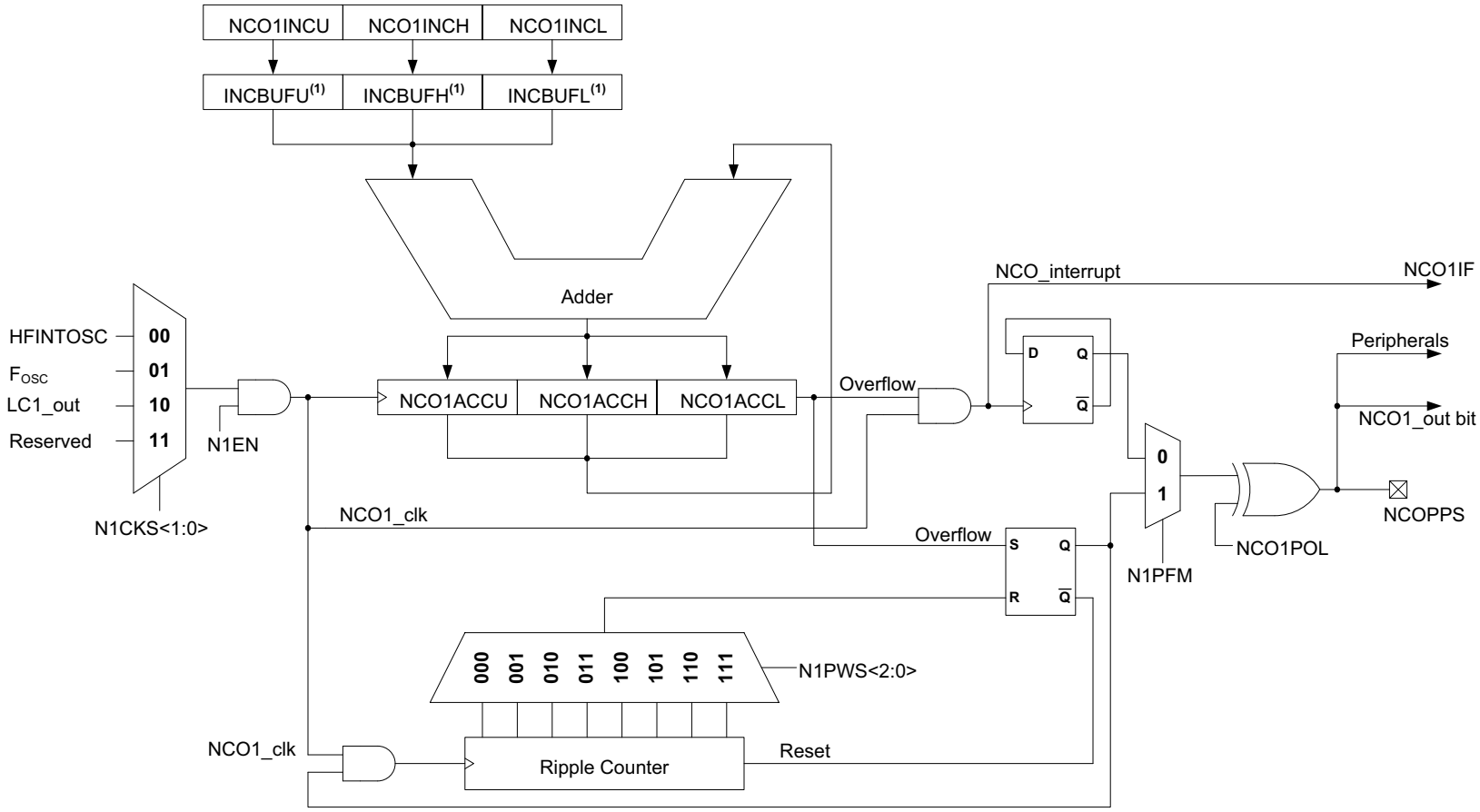
bit 6 **LD:** CWG Load Buffers bit(1)
- 1 = Dead-band count buffers to be loaded on CWG data rising edge following first falling edge after this bit is set.
- 0 = Buffers remain unchanged

bit 5-3 **Unimplemented**: Read as '0'

bit 2-0 **MODE<2:0>**: CWGx Mode bits
- 111 = Reserved
- 110 = Reserved
- 101 = CWG outputs operate in Push-Pull mode
- 100 = CWG outputs operate in Half-Bridge mode
- 011 = CWG outputs operate in Reverse Full-Bridge mode
- 010 = CWG outputs operate in Forward Full-Bridge mode
- 001 = CWG outputs operate in Synchronous Steering mode
- 000 = CWG outputs operate in Asynchronous Steering mode

Note 1: This bit can only be set after EN = 1; it cannot be set in the same cycle when EN is set.

FIGURE 23-1: NUMERICALLY CONTROLLED OSCILLATOR MODULE SIMPLIFIED BLOCK DIAGRAM



Note 1: The increment registers are double-buffered to allow for value changes to be made without first disabling the NCO1 module. They are shown for reference only and are not user accessible.

REGISTER 25-4: MDCARL: MODULATION LOW CARRIER CONTROL REGISTER

U-0	R/W-x/u	R/W-x/u	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
—	MDCLPOL	MDCLSYNC	—	MDCL<3:0>(1)			
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 **Unimplemented:** Read as '0'

bit 6 **MDCLPOL:** Modulator Low Carrier Polarity Select bit
- 1 = Selected low carrier signal is inverted
- 0 = Selected low carrier signal is not inverted

bit 5 **MDCLSYNC:** Modulator Low Carrier Synchronization Enable bit
- 1 = Modulator waits for a falling edge on the low time carrier signal before allowing a switch to the high time carrier
- 0 = Modulator output is not synchronized to the low time carrier signal(1)

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

bit 3-0 **MDCL<3:0>** Modulator Data High Carrier Selection bits (1)
- 1111 = CLC4 output
- 1110 = CLC3 output
- 1101 = CLC2 output
- 1100 = CLC1 output
- 1011 = HFINTOSC
- 1010 = FOSC
- 1001 = Reserved. No channel connected.
- 1000 = NCO1 output
- 0111 = PWM6 output
- 0110 = PWM5 output
- 0101 = CCP2 output (PWM Output mode only)
- 0100 = CCP1 output (PWM Output mode only)
- 0011 = Reference clock module signal (CLKR)
- 0010 = MDCIN2PPS
- 0001 = MDCIN1PPS
- 0000 = VSS

Note 1: Narrowed carrier pulse widths or spurs may occur in the signal stream if the carrier is not synchronized.

REGISTER 27-3: TMRxL(1): TIMERx LOW BYTE REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
TMRxL<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 **TMRxL<7:0>:** TMRx Low Byte bits

Note 1: 'x' refers to either '1', '3' or '5' for the respective Timer1/3/5 registers.

REGISTER 27-4: TMRxH(1): TIMERx HIGH BYTE REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
TMRxH<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 **TMRxH<7:0>:** TMRx High Byte bits

Note 1: 'x' refers to either '1', '3' or '5' for the respective Timer1/3/5 registers.

28.0 TIMER 2/4/6 MODULE

Timer2/4/6 modules are 8-bit timers that incorporate the following features:

- 8-bit Timer and Period registers (TMR2/4/6 and PR2/4/6, respectively)
- Readable and writable (both registers)
- Software programmable prescaler (1:1, 1:4, 1:16, and 1:64)
- Software programmable postscaler (1:1 to 1:16)
- Interrupt on TMR2/4/6 match with PR2/4/6
- Optional use as the shift clock for the MSSPx module

See Figure 28-1 for a block diagram of Timer2/4/6.

Note 1: In devices with more than one Timer module, it is very important to pay close attention to the register names used. A number placed after the module acronym is used to distinguish between separate modules. For example, the T2CON and T4CON control the same operational aspects of two completely different Timer modules.

2: Throughout this section, generic references to Timer2 module in any of its operating modes may be interpreted as being equally applicable to Timerx module. Register names, module signals, I/O pins and bit names may use the generic designator 'x' to indicate the use of a numeral to distinguish a particular module, when required.

FIGURE 28-1: TIMER2/4/6 BLOCK DIAGRAM

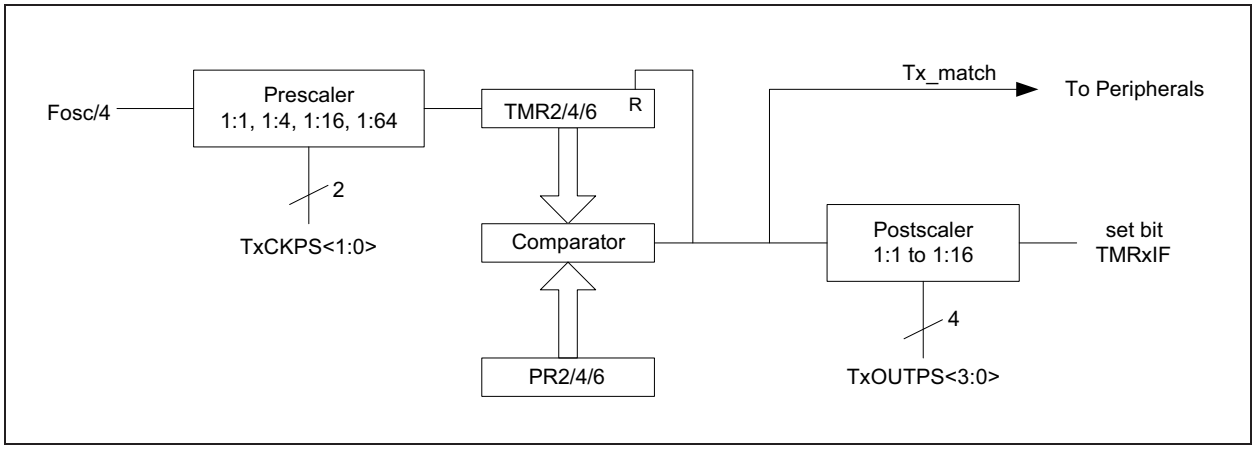


TABLE 30-2: MSSP CLOCK RATE W/BRG

FOSC	FCY	BRG Value	FCLOCK (2 Rollovers of BRG)
32 MHz	8 MHz	13h	400 kHz
32 MHz	8 MHz	19h	308 kHz
32 MHz	8 MHz	4Fh	100 kHz
16 MHz	4 MHz	09h	400 kHz
16 MHz	4 MHz	0Ch	308 kHz
16 MHz	4 MHz	27h	100 kHz
4 MHz	1 MHz	09h	100 kHz

Note: Refer to the I/O port electrical specifications in Table 35-4 to ensure the system is designed to support IOL requirements.

REGISTER 30-7: SSPxBUF: MSSP BUFFER REGISTER

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
SSPxBUF<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 **SSPxBUF<7:0>:** MSSP Buffer bits

TABLE 30-3: SUMMARY OF REGISTERS ASSOCIATED WITH MSSPx

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
TRISA	—	—	TRISA5	TRISA4	—(3)	TRISA2	TRISA1	TRISA0	143
ANSELA	—	—	ANSA5	ANSA4	—	ANSA2	ANSA1	ANSA0	144
INLVLA(1)	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	146
TRISB(2)	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	149
ANSELB(2)	ANSB7	ANSB6	ANSB5	ANSB4	—	—	—	—	150
INLVLB(2)	INLVLB7	INLVLB6	INLVLB5	INLVLB4	—	—	—	—	152
TRISC	TRISC7(2)	TRISC6(2)	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	155
ANSELC	ANSC7(2)	ANSC6(2)	ANSC5	ANSC4	ANSC3	ANSC2	ANSC1	ANSC0	157
INLVLC(1)	INLVLC7(2)	INLVLC6(2)	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	159
INTCON	GIE	PEIE	—	—	—	—	—	INTEDG	100
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	BCL1IF	TMR2IF	TMR1IF	107
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	BCL1IE	TMR2IE	TMR1IE	102
PIR2	TMR6IF	C2IF	C1IF	NVMIF	SSP2IF	BCL2IF	TMR4IF	NCO1IF	108
PIE2	TMR6IE	C2IE	C1IE	NVMIE	SSP2IE	BCL2IE	TMR4IE	NCO1IE	103
SSPxSTAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	359
SSPxCON1	WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				360
SSPxCON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	362
SSPxCON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	363
SSPxMSK	SSPxMSK<7:0>								364
SSPxADD	SSPxADD<7:0>								364
SSPxBUF	SSPxBUF<7:0>								365
SSPxCLKPPS	—	—	—	SSPxCLKPPS<4:0>					162
SSPxDATPPS	—	—	—	SSPxDATPPS<4:0>					162
SSPxSSPPS	—	—	—	SSPxSSPPS<4:0>					162

Legend: — = Unimplemented location, read as '0'. Shaded cells are not used by the MSSP module

Note 1: When using designated I^2C pins, the associated pin values in INLVLx will be ignored.

2: PIC16(L)F18346 only.

3: Unimplemented, read as '1'.

31.1.2.8 Asynchronous Reception Setup

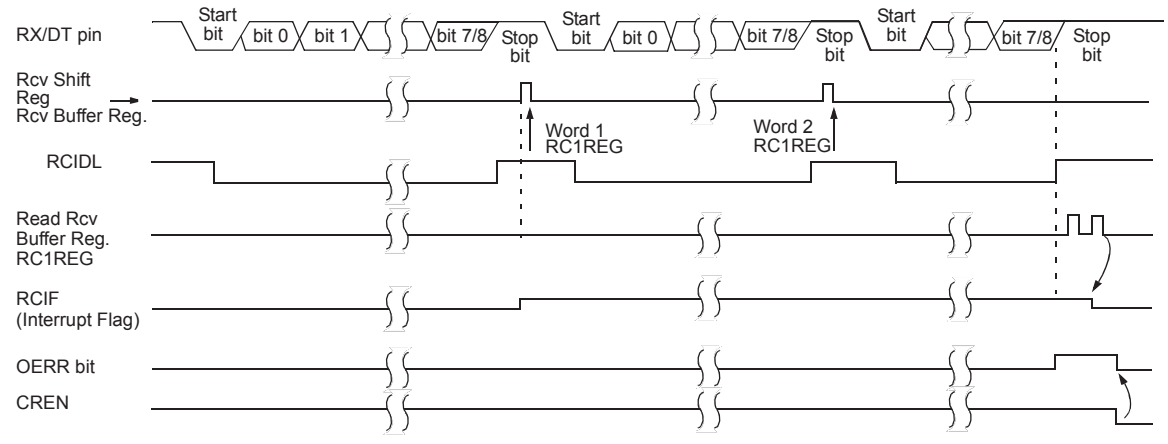
1. Initialize the SP1BRGH, SP1BRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 31.3 "EUSART1 Baud Rate Generator (BRG)"**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
5. If 9-bit reception is desired, set the RX9 bit.
6. Enable reception by setting the CREN bit.
7. The RCIF interrupt flag bit will be set when a character is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit was also set.
8. Read the RC1STA register to get the error flags and, if 9-bit data reception is enabled, the ninth data bit.
9. Get the received eight Least Significant data bits from the receive buffer by reading the RC1REG register.
10. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.

31.1.2.9 9-bit Address Detection Mode Setup

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

1. Initialize the SP1BRGH, SP1BRGL register pair and the BRGH and BRG16 bits to achieve the desired baud rate (see **Section 31.3 "EUSART1 Baud Rate Generator (BRG)"**).
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the serial port by setting the SPEN bit. The SYNC bit must be clear for asynchronous operation.
4. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
5. Enable 9-bit reception by setting the RX9 bit.
6. Enable address detection by setting the ADDEN bit.
7. Enable reception by setting the CREN bit.
8. The RCIF interrupt flag bit will be set when a character with the ninth bit set is transferred from the RSR to the receive buffer. An interrupt will be generated if the RCIE interrupt enable bit was also set.
9. Read the RC1STA register to get the error flags. The ninth data bit will always be set.
10. Get the received eight Least Significant data bits from the receive buffer by reading the RC1REG register. Software determines if this is the device's address.
11. If an overrun occurred, clear the OERR flag by clearing the CREN receiver enable bit.
12. If the device has been addressed, clear the ADDEN bit to allow all received data into the receive buffer and generate interrupts.

FIGURE 31-5: ASYNCHRONOUS RECEPTION



Note: This timing diagram shows three words appearing on the RX input. The RC1REG (receive buffer) is read after the third word, causing the OERR (overrun) bit to be set.

31.3.5 RECEIVING A BREAK CHARACTER

The Enhanced EUSART1 module can receive a Break character in two ways.

The first method to detect a Break character uses the FERR bit of the RC1STA register and the received data as indicated by RC1REG. The Baud Rate Generator is assumed to have been initialized to the expected baud rate.

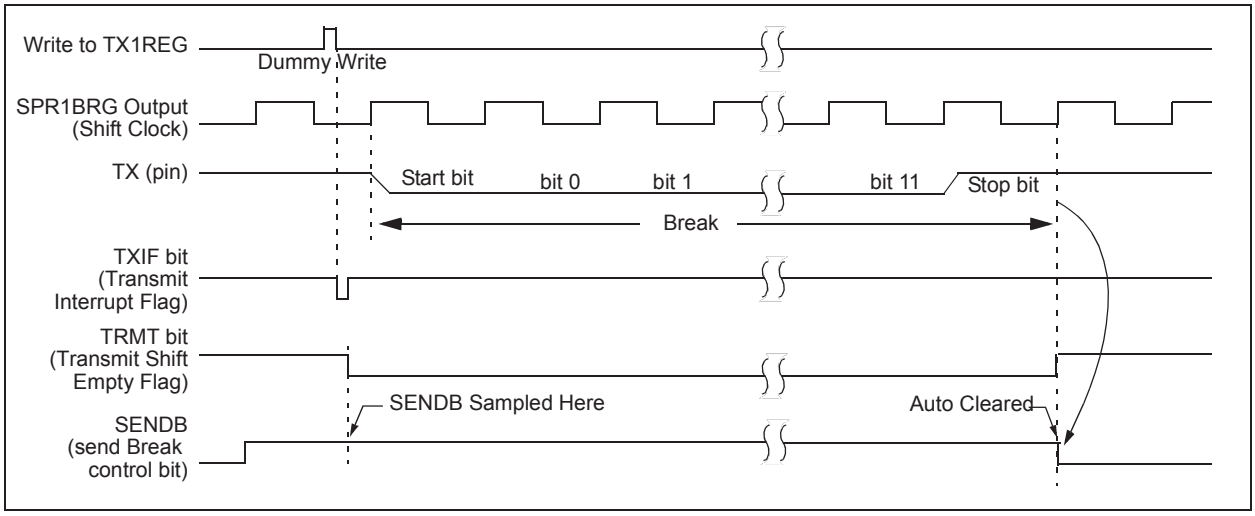
A Break character has been received when:

- RCIF bit is set
- FERR bit is set
- RC1REG = 00h

The second method uses the Auto-Wake-up feature described in **Section 31.3.3 "Auto-Wake-up on Break"**. By enabling this feature, the EUSART1 will sample the next two transitions on RX/DT, cause an RCIF interrupt, and receive the next data byte followed by another interrupt.

Note that following a Break character, the user will typically want to enable the Auto-Baud Detect feature. For both methods, the user can set the ABDEN bit of the BAUD1CON register before placing the EUSART1 in Sleep mode.

FIGURE 31-9: SEND BREAK CHARACTER SEQUENCE



BCF **Bit Clear f**

Syntax:	[*label*] BCF f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b ≤ 7
Operation:	0 → (f<b>)
Status Affected:	None
Description:	Bit 'b' in register 'f' is cleared.

BRA **Relative Branch**

Syntax:	[*label*] BRA label [*label*] BRA $+k
Operands:	-256 ≤ label - PC + 1 ≤ 255 -256 ≤ k ≤ 255
Operation:	(PC) + 1 + k → PC
Status Affected:	None
Description:	Add the signed 9-bit literal 'k' to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 1 + k. This instruction is a 2-cycle instruction. This branch has a limited range.

BRW **Relative Branch with W**

Syntax:	[*label*] BRW
Operands:	None
Operation:	(PC) + (W) → PC
Status Affected:	None
Description:	Add the contents of W (unsigned) to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 1 + (W). This instruction is a 2-cycle instruction.

BSF **Bit Set f**

Syntax:	[*label*] BSF f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b ≤ 7
Operation:	1 → (f<b>)
Status Affected:	None
Description:	Bit 'b' in register 'f' is set.

BTFSC **Bit Test f, Skip if Clear**

Syntax:	[*label*] BTFSC f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b ≤ 7
Operation:	skip if (f<b>) = 0
Status Affected:	None
Description:	If bit 'b' in register 'f' is '1', the next instruction is executed. If bit 'b', in register 'f', is '0', the next instruction is discarded, and a NOP is executed instead, making this a 2-cycle instruction.

BTFSS **Bit Test f, Skip if Set**

Syntax:	[*label*] BTFSS f,b
Operands:	0 ≤ f ≤ 127 0 ≤ b < 7
Operation:	skip if (f<b>) = 1
Status Affected:	None
Description:	If bit 'b' in register 'f' is '0', the next instruction is executed. If bit 'b' is '1', then the next instruction is discarded and a NOP is executed instead, making this a 2-cycle instruction.

TABLE 35-5: I/O AND CLOCK TIMING SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	**Sym.**	**Characteristic**	**Min.**	**Typ.†**	**Max.**	**Units**	**Conditions**
High Voltage Entry Programming Mode Specifications							
MEM01	VIHH	Voltage on $\overline{MCLR}$/VPP pin to enter Programming mode	8	—	9	V	**Note 2**
MEM02	IPPGM	Current on $\overline{MCLR}$/VPP pin during Programming mode	—	—	600	uA	**Note 2**
Programming Mode Specifications							
MEM10	VBE	VDD for Bulk Erase	—	2.7	—	V	
MEM11	IDDPGM	Supply Current during Programming Operation	—	—	3	mA	
Data EEPROM Memory Specifications							
MEM20	ED	DataEE Byte Endurance	100k	—	—	E/W	-40°C ≤ TA ≤ 85°C
MEM21	TD_RET	Characteristic Retention	—	40	—	Year	Provided no other specifications are violated
MEM22	ND_REF	Total Erase/Write Cycles before Refresh	—	—	100k	E/W	
MEM23	VD_RW	VDD for Read or Erase/Write Operation	VDDMIN	—	VDDMAX	V	
MEM24	TD_BEW	Byte Erase and Write Cycle Time	—	4.0	5.0	ms	
Program Flash Memory Specifications							
MEM30	EP	Flash Memory Cell Endurance	10k	—	—	E/W	-40°C ≤ TA ≤ 85°C **(Note 1)**
MEM31	EPHEF	High-Endurance Flash Memory Cell Endurance	100k	—	—	E/W	TBD
MEM32	TP_RET	Characteristic Retention	—	40	—	Year	Provided no other specifications are violated
MEM33	VP_RD	VDD for Read Operation	VDDMIN	—	VDDMAX	V	
MEM34	VP_REW	VDD for Row Erase or Write Operation	VDDMIN	—	VDDMAX	V	
MEM35	TP_REW	Self-Timed Row Erase or Self-Timed Write	—	2.0	2.5	ms	

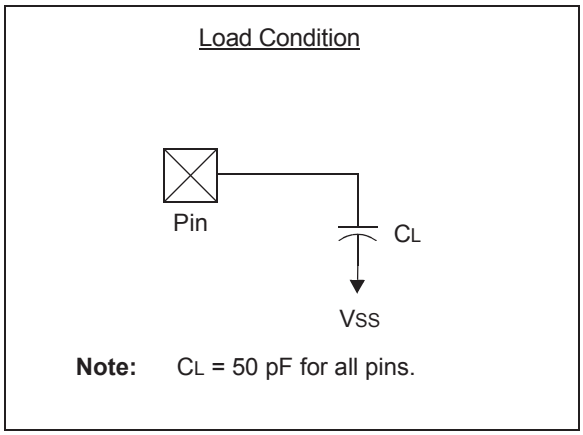
† Data in "Typ." column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Flash Memory Cell Endurance for the Flash memory is defined as: One Row Erase operation and one Self-Timed Write.

2: Required only if CONFIG3.LVP is disabled.

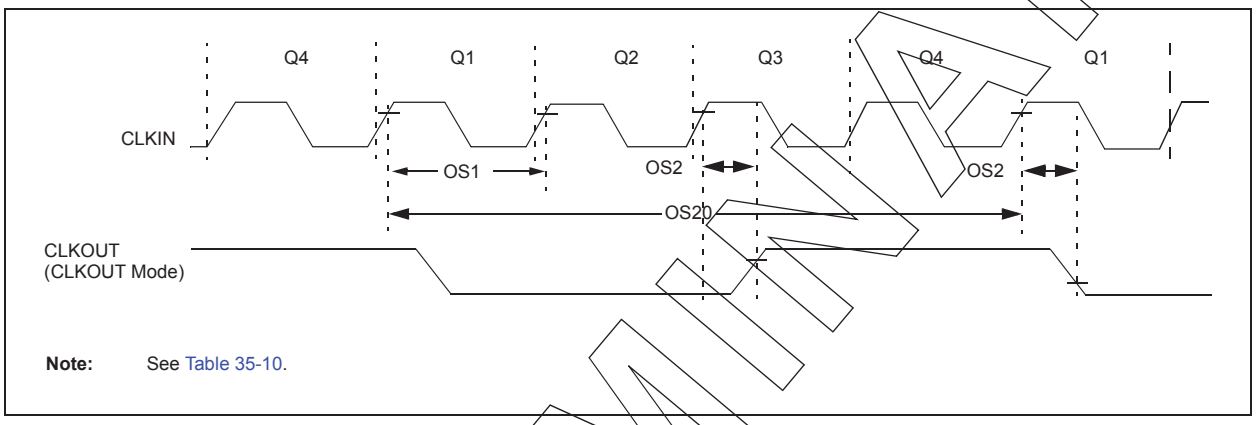
35.4 AC Characteristics

FIGURE 35-4: LOAD CONDITIONS



Note: CL = 50 pF for all pins.

FIGURE 35-5: CLOCK TIMING



Note: See Table 35-10.

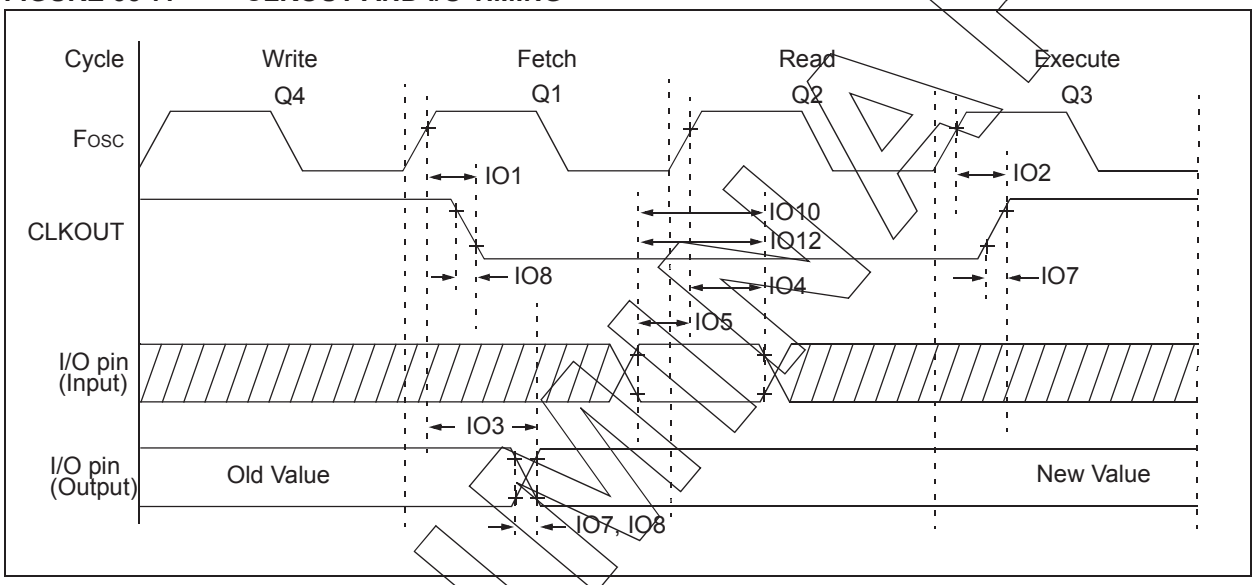
TABLE 35-9: PLL CLOCK TIMING SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	Sym.	Characteristic	Min.	Typ.†	Max.	Units	Conditions
PLL01	FPLLIN	PLL Input Frequency Range	4	—	8	MHz	
PLL02	FPLLOUT	PLL Output Frequency Range	16	—	32	MHz	
PLL03	TPLLST	PLL Lock Time from Start-up	—	200	—	µs	
PLL04	FPLLJIT	PLL Output Frequency Stability (Jitter)	-0.25	—	0.25	%	

* These parameters are characterized but not tested.

† Data in "Typ." column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 35-7: CLKOUT AND I/O TIMING



37.6 MPLAB X SIM Software Simulator

The MPLAB X SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB X SIM Software Simulator fully supports symbolic debugging using the MPLAB XC Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

37.7 MPLAB REAL ICE In-Circuit Emulator System

The MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs all 8, 16 and 32-bit MCU, and DSC devices with the easy-to-use, powerful graphical user interface of the MPLAB X IDE.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ-11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradeable through future firmware downloads in MPLAB X IDE. MPLAB REAL ICE offers significant advantages over competitive emulators including full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, logic probes, a ruggedized probe interface and long (up to three meters) interconnection cables.

37.8 MPLAB ICD 3 In-Circuit Debugger System

The MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost-effective, high-speed hardware debugger/programmer for Microchip Flash DSC and MCU devices. It debugs and programs PIC Flash microcontrollers and dsPIC DSCs with the powerful, yet easy-to-use graphical user interface of the MPLAB IDE.

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

37.9 PICkit 3 In-Circuit Debugger/ Programmer

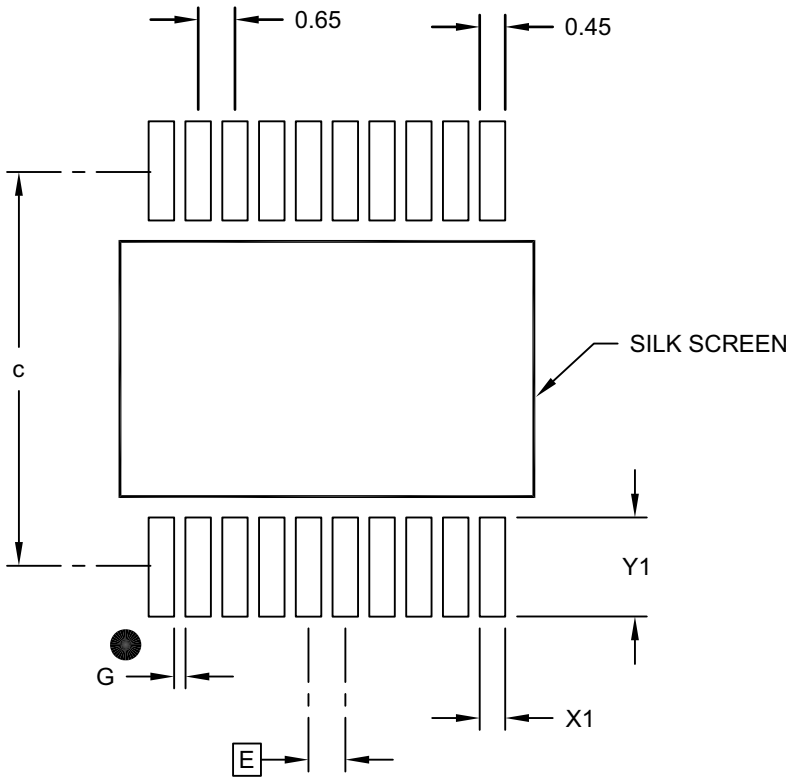
The MPLAB PICkit 3 allows debugging and programming of PIC and dsPIC Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB IDE. The MPLAB PICkit 3 is connected to the design engineer's PC using a full-speed USB interface and can be connected to the target via a Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the Reset line to implement in-circuit debugging and In-Circuit Serial Programming™ (ICSP™).

37.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at V_{DDMIN} and V_{DDMAX} for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages, and a modular, detachable socket assembly to support various package types. The ICSP cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices, and incorporates an MMC card for file storage and data applications.

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