

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	I ² C, 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 ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	14-DIP (0.300", 7.62mm)
Supplier Device Package	14-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18326-e-p

4.0 MEMORY ORGANIZATION

These devices contain the following types of memory:

- Program Memory
 - Configuration Words
 - Device ID
 - Revision ID
 - User ID
 - Program Flash Memory
- Data Memory
 - Core Registers
 - Special Function Registers
 - General Purpose RAM
 - Common RAM
- Data EEPROM

The following features are associated with access and control of program memory and data memory:

- PCL and PCLATH
- Stack
- Indirect Addressing
- NVMREG Access

4.1 Program Memory Organization

The enhanced mid-range core has a 15-bit program counter capable of addressing 32K x 14 program memory space. [Table 4-1](#) shows the memory sizes implemented. Accessing a location above these boundaries will cause a wrap-around within the implemented memory space. The Reset vector is at 0000h and the interrupt vector is at 0004h (see [Figure 4-1](#)).

TABLE 4-1: DEVICE SIZES AND ADDRESSES

Device	Program Memory Size (Words)	Last Program Memory Address
PIC16(L)F18326/18346	16384	3FFFh

PIC16(L)F18326/18346

5.3 Code Protection

Code protection allows the device to be protected from unauthorized access. Program memory protection and data memory are controlled independently. Internal access to the program memory is unaffected by any code protection setting.

5.3.1 PROGRAM MEMORY PROTECTION

The entire program memory space is protected from external reads and writes by the CP bit in Configuration Words. When $\overline{CP} = 0$, external reads and writes of program memory are inhibited and a read will return all '0's. The CPU can continue to read program memory, regardless of the protection bit settings. Self-write writing the program memory is dependent upon the write protection setting. See [Section 5.4 "Write Protection"](#) for more information.

5.3.2 DATA MEMORY PROTECTION

The entire data EEPROM is protected from external reads and writes by the CPD bit in the Configuration Words. When $\overline{CPD} = 0$, external reads and writes of EEPROM memory are inhibited and a read will return all '0's. The CPU can continue to read and write EEPROM memory, regardless of the protection bit settings.

5.4 Write Protection

Write protection allows the device to be protected from unintended self-writes. Applications, such as boot loader software, can be protected while allowing other regions of the program memory to be modified.

The WRT<1:0> bits in Configuration Words define the size of the program memory block that is protected.

5.5 User ID

Four memory locations (8000h-8003h) are designated as ID locations where the user can store checksum or other code identification numbers. These locations are readable and writable during normal execution. See [Section 11.4.7 "NVMREG EEPROM, User ID, Device ID and Configuration Word Access"](#) for more information on accessing these memory locations. For more information on checksum calculation, see the *"PIC16(L)F183XX Memory Programming Specification"* (DS40001738).

5.6 Device ID and Revision ID

The 14-bit Device ID word is located at 8006h and the 14-bit Revision ID is located at 8005h. These locations are read-only and cannot be erased or modified. See [Section 11.4 "NVMREG Access"](#) for more information on accessing these memory locations.

Development tools, such as device programmers and debuggers, may be used to read the Device ID and Revision ID.

PIC16(L)F18326/18346

7.2.1.3 Oscillator Start-up Timer (OST)

If the oscillator module is configured for LP, XT or HS modes, the Oscillator Start-up Timer (OST) counts 1024 oscillations from OSC1. This occurs following a Power-on Reset (POR), Brown-out Reset (BOR), or a wake-up from Sleep. The OST ensures that the oscillator circuit, using a quartz crystal resonator or ceramic resonator, has started and is providing a stable system clock to the oscillator module.

7.2.1.4 4x PLL

The oscillator module contains a PLL that can be used with external clock sources to provide a system clock source. The input frequency for the PLL must fall within specifications. See the PLL Clock Timing Specifications in [Table 35-9](#).

The PLL may be enabled for use by one of two methods:

1. Program the RSTOSC bits in the Configuration Word 1 to enable the EXTOSC with 4x PLL.
2. Write the NOSC<2:0> bits in the OSCCON1 register to enable the EXTOSC with 4x PLL.

7.2.1.5 Secondary Oscillator

The secondary oscillator is a separate oscillator block that can be used as an alternate system clock source. The secondary oscillator is optimized for 32.768 kHz, and can be used with an external crystal oscillator connected to the SOSCI and SOSCO device pins, or an external clock source connected to the SOSCIN pin. The secondary oscillator can be selected during run-time using clock switching. Refer to [Section 7.3 “Clock Switching”](#) for more information.

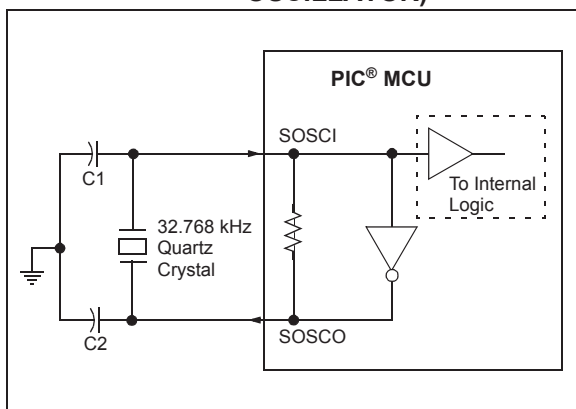
Note 1: Quartz crystal characteristics vary according to type, package and manufacturer. The user should consult the manufacturer data sheets for specifications and recommended application.

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

3: For oscillator design assistance, reference the following Microchip Application Notes:

- AN826, *Crystal Oscillator Basics and Crystal Selection for rfPIC® and PIC® Devices* (DS00826)
- AN849, *Basic PICmicro® Oscillator Design* (DS00849)
- AN943, *Practical PICmicro® Oscillator Analysis and Design* (DS00943)
- AN949, *Making Your Oscillator Work* (DS00949)
- TB097, *Interfacing a Micro Crystal MS1V-T1K 32.768 kHz Tuning Fork Crystal to a PIC16F690/SS* (DS91097)
- AN1288, *Design Practices for Low-Power External Oscillators* (DS01288)

FIGURE 7-5: QUARTZ CRYSTAL OPERATION (SECONDARY OSCILLATOR)



PIC16(L)F18326/18346

7.2.2.3 Internal Oscillator Frequency Adjustment

The HFINTOSC and LFINTOSC internal oscillators are both factory-calibrated. The HFINTOSC oscillator can be adjusted in software by writing to the OSCTUNE register ([Register 7-3](#)). OSCTUNE does not affect the LFINTOSC frequency.

The default value of the OSCTUNE register is 00h. The value is a 6-bit two's complement number. A value of 1Fh will provide an adjustment to the maximum frequency. A value of 20h will provide an adjustment to the minimum frequency.

When the OSCTUNE register is modified, the HFINTOSC oscillator frequency will begin shifting to the new frequency. Code execution continues during this shift. There is no indication that the shift has occurred.

7.2.2.4 LFINTOSC

The Low-Frequency Internal Oscillator (LFINTOSC) is a factory-calibrated 31 kHz internal clock source.

The LFINTOSC is the clock source for the Power-up Timer (PWRT), Watchdog Timer (WDT) and Fail-Safe Clock Monitor (FSCM). The LFINTOSC can also be used as the system clock, or as a clock or input source to certain peripherals.

The LFINTOSC is selected as the clock source through one of the following methods:

- Programming the RSTOSC<2:0> bits of Configuration Word 1 to enable LFINTOSC.
- Write to the NOSC<2:0> bits of the OSCCON1 register.

7.2.2.5 Oscillator Status and Manual Enable

The 'ready' status of each oscillator is displayed in the OSCSTAT1 register ([Register 7-4](#)). The oscillators can also be manually enabled through the OSCEN register ([Register 7-5](#)). Manual enables make it possible to verify the operation of the EXTOSC or SOSC crystal oscillators. This can be achieved by enabling the selected oscillator, then watching the corresponding 'ready' state of the oscillator in the OSCSTAT1 register.

PIC16(L)F18326/18346

11.4.7 NVMREG EEPROM, USER ID, DEVICE ID AND CONFIGURATION WORD ACCESS

Instead of accessing Program Flash Memory, the EEPROM, the User ID's, Device ID/Revision ID and Configuration Words can be accessed when NVMREGS = 1 in the NVMCON1 register. This is the region that would be pointed to by PC<15> = 1, but not all addresses are accessible. Different access may exist for reads and writes. Refer to [Table 11-3](#).

When read access is initiated on an address outside the parameters listed in [Table 11-3](#), the NVMDATH: NVMDATL register pair is cleared, reading back '0's.

**TABLE 11-3: EEPROM, USER ID, DEV/REV ID AND CONFIGURATION WORD ACCESS
(NVMREGS = 1)**

Address	Function	Read Access	Write Access
8000h-8003h	User IDs	Yes	Yes
8005h-8006h	Device ID/Revision ID	Yes	No
8007h-800Ah	Configuration Words 1-4	Yes	No
F000h-F0FFh	EEPROM	Yes	Yes

PIC16(L)F18326/18346

REGISTER 14-2: PMD1: PMD CONTROL REGISTER 1

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
NCOMD	TMR6MD	TMR5MD	TMR4MD	TMR3MD	TMR2MD	TMR1MD	TMR0MD
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	q = Value depends on condition

bit 7	NCOMD: Disable Numerically Control Oscillator bit 1 = NCO1 module disabled 0 = NCO1 module enabled
bit 6	TMR6MD: Disable Timer TMR6 bit 1 = TMR6 module disabled 0 = TMR6 module enabled
bit 5	TMR5MD: Disable Timer TMR5 bit 1 = TMR5 module disabled 0 = TMR5 module enabled
bit 4	TMR4MD: Disable Timer TMR4 bit 1 = TMR4 module disabled 0 = TMR4 module enabled
bit 3	TMR3MD: Disable Timer TMR3 bit 1 = TMR3 module disabled 0 = TMR3 module enabled
bit 2	TMR2MD: Disable Timer TMR2 bit 1 = TMR2 module disabled 0 = TMR2 module enabled
bit 1	TMR1MD: Disable Timer TMR1 bit 1 = TMR1 module disabled 0 = TMR1 module enabled
bit 0	TMR0MD: Disable Timer TMR0 bit 1 = TMR0 module disabled 0 = TMR0 module enabled

REGISTER 18-2: CMxCON1: COMPARATOR Cx CONTROL REGISTER 1

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	
CxINTP	CxINTN	CxPCH<2:0>			CxNCH<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	

- bit 7 **CxINTP:** Comparator Interrupt on Positive Going Edge Enable bits
 1 = The CxIF interrupt flag will be set upon a positive going edge of the CxOUT bit
 0 = No interrupt flag will be set on a positive going edge of the CxOUT bit
- bit 6 **CxINTN:** Comparator Interrupt on Negative Going Edge Enable bits
 1 = The CxIF interrupt flag will be set upon a negative going edge of the CxOUT bit
 0 = No interrupt flag will be set on a negative going edge of the CxOUT bit
- bit 5-3 **CxPCH<2:0>:** Comparator Positive Input Channel Select bits
 111 = CxVP connects to Vss
 110 = CxVP connects to FVR Buffer 2
 101 = CxVP connects to DAC output
 100 = CxVP unconnected
 011 = CxVP unconnected
 010 = CxVP unconnected
 001 = CxVN unconnected
 000 = CxVP connects to CxIN0+ pin
- bit 2-0 **CxNCH<2:0>:** Comparator Negative Input Channel Select bits
 111 = CxVN connects to Vss
 110 = CxVN connects to FVR Buffer 2
 101 = CxVN unconnected
 100 = CxVN unconnected
 011 = CxVN connects to CxIN3- pin
 010 = CxVN connects to CxIN2- pin
 001 = CxVN connects to CxIN1- pin
 000 = CxVN connects to CxIN0- pin

PIC16(L)F18326/18346

REGISTER 20-2: CWGxCON1: CWGx CONTROL REGISTER 1

U-0	U-0	R-x	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	IN	—	POLD	POLC	POLB	POLA
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	q = Value depends on condition

- bit 7-6 **Unimplemented:** Read as '0'
- bit 5 **IN:** CWGx Data Input Signal (read-only)
- bit 4 **Unimplemented:** Read as '0'
- bit 3 **POLD:** WGxD Output Polarity bit
1 = Signal output is inverted polarity
0 = Signal output is normal polarity
- bit 2 **POLC:** WGxC Output Polarity bit
1 = Signal output is inverted polarity
0 = Signal output is normal polarity
- bit 1 **POLB:** WGxB Output Polarity bit
1 = Signal output is inverted polarity
0 = Signal output is normal polarity
- bit 0 **POLA:** WGxA Output Polarity bit
1 = Signal output is inverted polarity
0 = Signal output is normal polarity

REGISTER 20-3: CWGxCLKCON: CWGx CLOCK INPUT SELECTION REGISTER

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0
—	—	—	—	—	—	—	CS
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	q = Value depends on condition

- bit 7-1 **Unimplemented:** Read as '0'
- bit 0 **CS:** CWG Clock Source Selection Select bits

WGCLK	Clock Source
0	FOSC
1	HFINTOSC (remains operating during Sleep)

PIC16(L)F18326/18346

REGISTER 22-3: ADOACT: A/D AUTO-CONVERSION TRIGGER

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—	ADACT<4:0>				
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 7-5 **Unimplemented:** Read as '0'

bit 4-0 **ADACT<4:0>:** Auto-Conversion Trigger Selection bits⁽¹⁾

1111	=	CCP4
1110	=	CCP3
1101	=	CCP2
1100	=	CCP1
1011	=	CLC4
1010	=	CLC3
1001	=	CLC2
1000	=	CLC1
0111	=	Comparator C2
0110	=	Comparator C1
0101	=	Timer2-PR2 match
0100	=	Timer1 overflow ⁽²⁾
0011	=	Timer0 overflow ⁽²⁾
0010	=	Timer6-PR6 match
0001	=	Timer4-PR4 match
0000	=	No auto-conversion trigger selected
10000	=	Timer3 overflow ⁽²⁾
10001	=	Timer5 overflow ⁽²⁾

Note 1: This is a rising edge sensitive input for all sources.

Note 2: Trigger corresponds to when the peripheral's interrupt flag is set.

PIC16(L)F18326/18346

REGISTER 22-4: ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 0

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
ADRES<9:2>							
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 **ADRES<9:2>**: ADC Result Register bits
Upper eight bits of 10-bit conversion result

REGISTER 22-5: ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 0

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
ADRES<1:0>		—	—	—	—	—	—
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 7-6 **ADRES<1:0>**: ADC Result Register bits
Lower two bits of 10-bit conversion result

bit 5-0 **Reserved**: Do not use.

PIC16(L)F18326/18346

FIGURE 25-1: SIMPLIFIED BLOCK DIAGRAM OF THE DATA SIGNAL MODULATOR

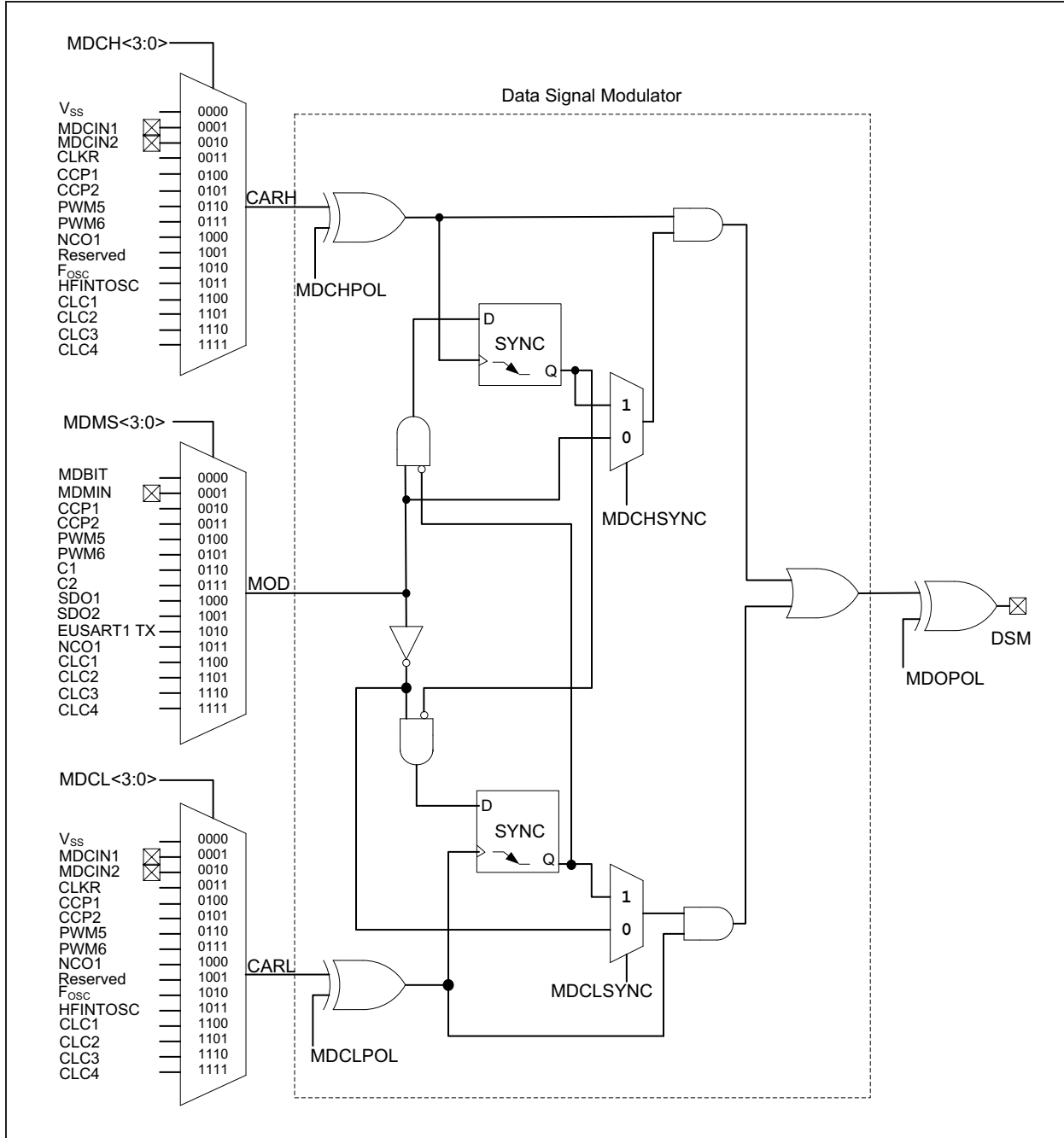


FIGURE 27-2: TIMER1 INCREMENTING EDGE

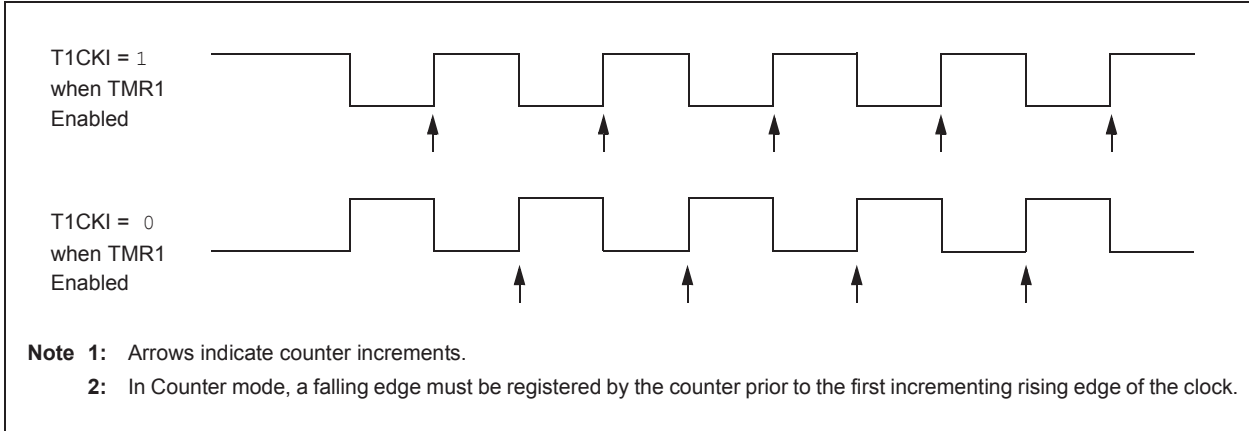
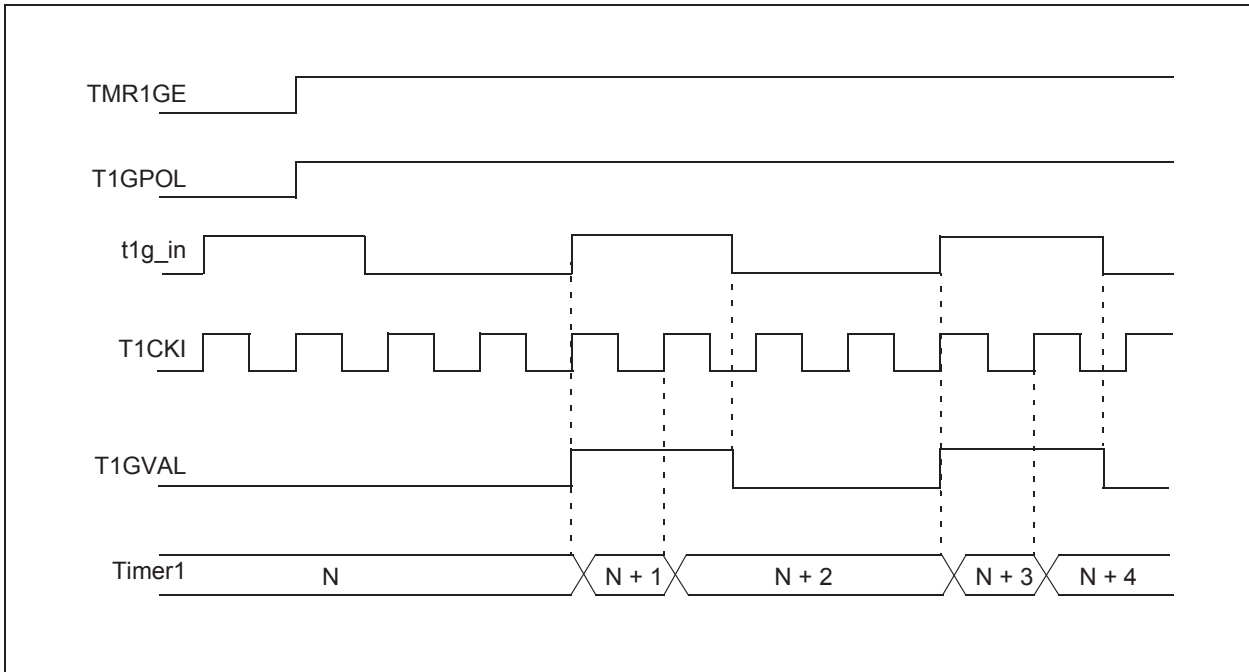


FIGURE 27-3: TIMER1 GATE ENABLE MODE



30.6.7 I²C MASTER MODE RECEPTION

Master mode reception (Figure 30-29) is enabled by programming the Receive Enable bit, RCEN bit of the SSPxCON2 register.

Note: The MSSPx module must be in an Idle state before the RCEN bit is set or the RCEN bit will be disregarded.

The Baud Rate Generator begins counting and on each rollover, the state of the SCL pin changes (high-to-low/low-to-high) and data is shifted into the SSPxSR. After the falling edge of the eighth clock, the receive enable flag is automatically cleared, the contents of the SSPxSR are loaded into the SSPxBUF, the BF flag bit is set, the SSPxIF flag bit is set and the Baud Rate Generator is suspended from counting, holding SCL low. The MSSPx is now in Idle state awaiting the next command. When the buffer is read by the CPU, the BF flag bit is automatically cleared. The user can then send an Acknowledge bit at the end of reception by setting the Acknowledge Sequence Enable, ACKEN bit of the SSPxCON2 register.

30.6.7.1 BF Status Flag

In receive operation, the BF bit is set when an address or data byte is loaded into SSPxBUF from SSPxSR. It is cleared when the SSPxBUF register is read.

30.6.7.2 SSPOV Status Flag

In receive operation, the SSPOV bit is set when eight bits are received into the SSPxSR and the BF flag bit is already set from a previous reception.

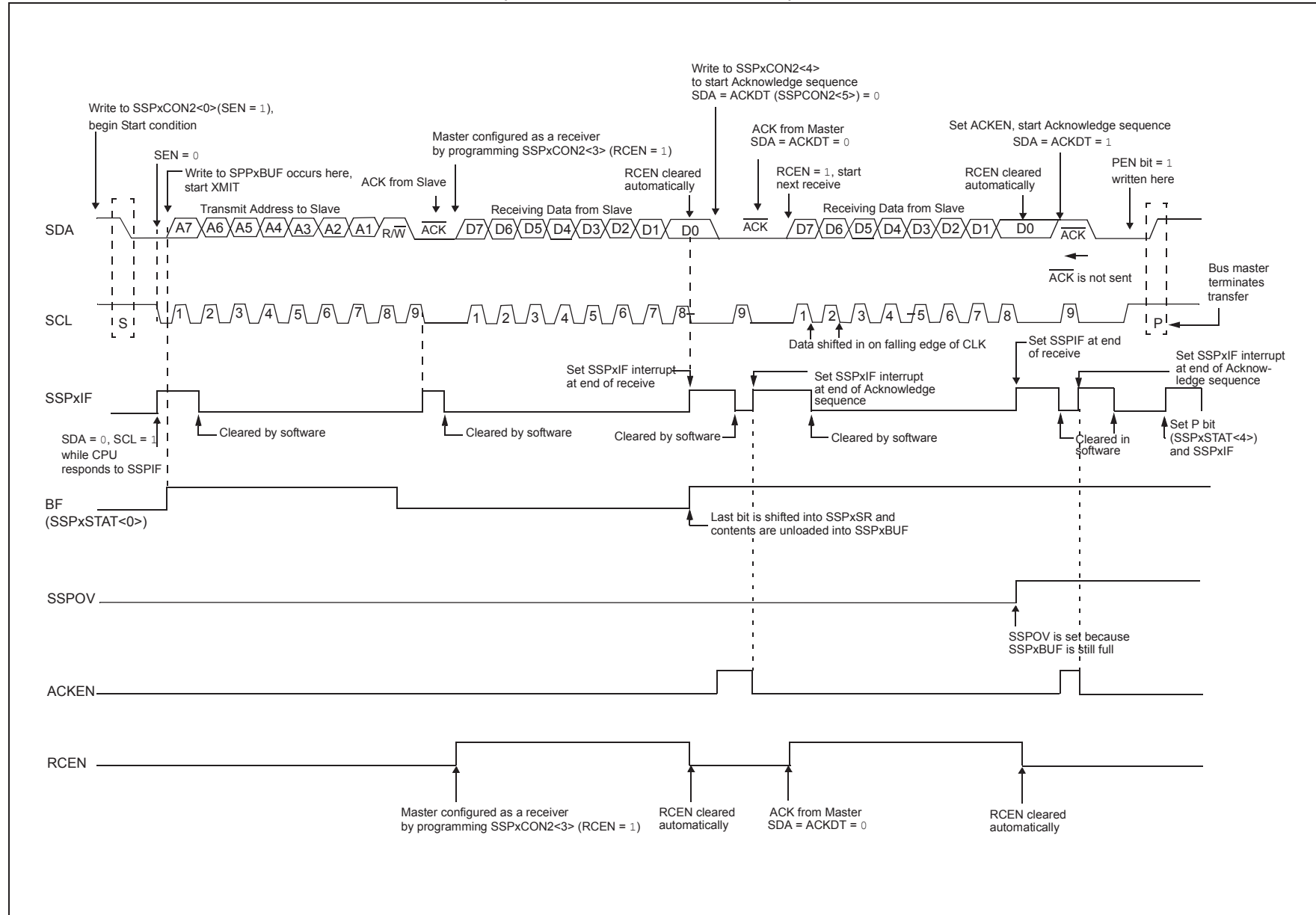
30.6.7.3 WCOL Status Flag

If the user writes the SSPxBUF when a receive is already in progress (i.e., SSPxSR is still shifting in a data byte), the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur).

30.6.7.4 Typical Receive Sequence:

1. The user generates a Start condition by setting the SEN bit of the SSPxCON2 register.
2. SSPxIF is set by hardware on completion of the Start.
3. SSPxIF is cleared by software.
4. User writes SSPxBUF with the slave address to transmit and the R/W bit set.
5. Address is shifted out the SDA pin until all eight bits are transmitted. Transmission begins as soon as SSPxBUF is written to.
6. The MSSPx module shifts in the $\overline{\text{ACK}}$ bit from the slave device and writes its value into the ACKSTAT bit of the SSPxCON2 register.
7. The MSSPx module generates an interrupt at the end of the ninth clock cycle by setting the SSPxIF bit.
8. User sets the RCEN bit of the SSPxCON2 register and the master clocks in a byte from the slave.
9. After the eighth falling edge of SCL, SSPxIF and BF are set.
10. Master clears SSPxIF and reads the received byte from SSPxBUF, clears BF.
11. Master sets $\overline{\text{ACK}}$ value sent to slave in ACKDT bit of the SSPxCON2 register and initiates the $\overline{\text{ACK}}$ by setting the ACKEN bit.
12. Master's $\overline{\text{ACK}}$ is clocked out to the slave and SSPxIF is set.
13. User clears SSPIF.
14. Steps 8-13 are repeated for each received byte from the slave.
15. Master sends a not $\overline{\text{ACK}}$ or Stop to end communication.

FIGURE 30-29: I²C MASTER MODE WAVEFORM (RECEPTION, 7-BIT ADDRESS)



PIC16(L)F18326/18346

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.

31.3.2 AUTO-BAUD OVERFLOW

During the course of automatic-baud detection, the ABDOVF bit of the BAUDxCON register will be set if the baud rate counter overflows before the fifth rising edge is detected on the RX pin. The ABDOVF bit indicates that the counter has exceeded the maximum count that can fit in the 16 bits of the SPxBRGH:SPxBRGL register pair. The Overflow condition will set the RCIF flag. The counter continues to count until the fifth rising edge is detected on the RX pin. The RCIDL bit will remain false ('0') until the fifth rising edge at which time the RCIDL bit will be set. If the RCREG is read after the overflow occurs but before the fifth rising edge, then the fifth rising edge will set the RCIF again.

Terminating the auto-baud process early to clear an Overflow condition will prevent proper detection of the sync character fifth rising edge. If any falling edges of the sync character have not yet occurred when the ABDEN bit is cleared, then those will be falsely detected as Start bits. The following steps are recommended to clear the Overflow condition:

1. Read RCREG to clear RCIF
2. If RCIDL is zero, then wait for RCIF and repeat step 1
3. Clear the ABDOVF bit

31.3.3 AUTO-WAKE-UP ON BREAK

During Sleep mode, all clocks to the EUSART1 are suspended. Because of this, the Baud Rate Generator is inactive and a proper character reception cannot be performed. The Auto-Wake-up feature allows the controller to wake-up due to activity on the RX/DT line. This feature is available only in Asynchronous mode.

The Auto-Wake-up feature is enabled by setting the WUE bit of the BAUD1CON register. Once set, the normal receive sequence on RX/DT is disabled, and the EUSART1 remains in an Idle state, monitoring for a wake-up event independent of the CPU mode. A wake-up event consists of a high-to-low transition on the RX/DT line. (This coincides with the start of a Sync Break or a wake-up signal character for the LIN protocol.)

The EUSART1 module generates an RCIF interrupt coincident with the wake-up event. The interrupt is generated synchronously to the Q clocks in normal CPU operating modes (Figure 31-7), and asynchronously if the device is in Sleep mode (Figure 31-8). The Interrupt condition is cleared by reading the RC1REG register.

The WUE bit is automatically cleared by the low-to-high transition on the RX line at the end of the Break. This signals to the user that the Break event is over. At this point, the EUSART1 module is in Idle mode waiting to receive the next character.

31.3.3.1 Special Considerations

Break Character

To avoid character errors or character fragments during a wake-up event, the wake-up character must be all zeros.

When the wake-up is enabled the function works independent of the low time on the data stream. If the WUE bit is set and a valid non-zero character is received, the low time from the Start bit to the first rising edge will be interpreted as the wake-up event. The remaining bits in the character will be received as a fragmented character and subsequent characters can result in framing or overrun errors.

Therefore, the initial character in the transmission must be all '0's. This must be ten or more bit times, 13-bit times recommended for LIN bus, or any number of bit times for standard RS-232 devices.

Oscillator Start-up Time

Oscillator start-up time must be considered, especially in applications using oscillators with longer start-up intervals (i.e., LP, XT or HS/PLL mode). The Sync Break (or wake-up signal) character must be of sufficient length, and be followed by a sufficient interval, to allow enough time for the selected oscillator to start and provide proper initialization of the EUSART1.

WUE Bit

The wake-up event causes a receive interrupt by setting the RCIF bit. The WUE bit is cleared in hardware by a rising edge on RX/DT. The Interrupt condition is then cleared in software by reading the RC1REG register and discarding its contents.

To ensure that no actual data is lost, check the RCIDL bit to verify that a receive operation is not in process before setting the WUE bit. If a receive operation is not occurring, the WUE bit may then be set just prior to entering the Sleep mode.

PIC16(L)F18326/18346

31.4.1.6 Slave Clock

Synchronous data transfers use a separate clock line, which is synchronous with the data. A device configured as a slave receives the clock on the TX/CK line. The TX/CK pin output driver is automatically disabled when the device is configured for synchronous slave transmit or receive operation. Serial data bits change on the leading edge to ensure they are valid at the trailing edge of each clock. One data bit is transferred for each clock cycle. Only as many clock cycles should be received as there are data bits.

Note: If the device is configured as a slave and the TX/CK function is on an analog pin, the corresponding ANSEL bit must be cleared.

31.4.1.7 Receive Overrun Error

The receive FIFO buffer can hold two characters. An overrun error will be generated if a third character, in its entirety, is received before RC1REG is read to access the FIFO. When this happens the OERR bit of the RC1STA register is set. Previous data in the FIFO will not be overwritten. The two characters in the FIFO buffer can be read, however, no additional characters will be received until the error is cleared. The OERR bit can only be cleared by clearing the Overrun condition. If the overrun error occurred when the SREN bit is set and CREN is clear then the error is cleared by reading RC1REG. If the overrun occurred when the CREN bit is set then the Error condition is cleared by either clearing the CREN bit of the RC1STA register or by clearing the SPEN bit which resets the EUSART1.

31.4.1.8 Receiving 9-bit Characters

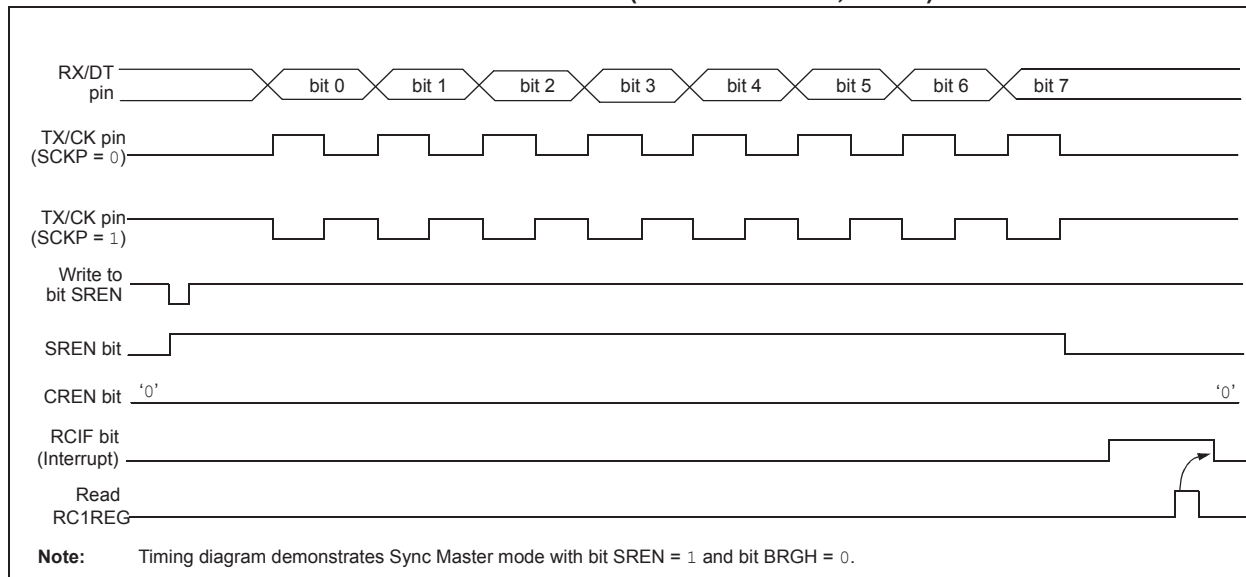
The EUSART1 supports 9-bit character reception. When the RX9 bit of the RC1STA register is set the EUSART1 will shift nine bits into the RSR for each

character received. The RX9D bit of the RC1STA register is the ninth, and Most Significant, data bit of the top unread character in the receive FIFO. When reading 9-bit data from the receive FIFO buffer, the RX9D data bit must be read before reading the eight Least Significant bits from the RC1REG.

31.4.1.9 Synchronous Master Reception Setup

1. Initialize the SP1BRGH, SP1BRGL register pair for the appropriate baud rate. Set or clear the BRGH and BRG16 bits, as required, to achieve the desired baud rate.
2. Clear the ANSEL bit for the RX pin (if applicable).
3. Enable the synchronous master serial port by setting bits SYNC, SPEN and CSRC.
4. Ensure bits CREN and SREN are clear.
5. If interrupts are desired, set the RCIE bit of the PIE1 register and the GIE and PEIE bits of the INTCON register.
6. If 9-bit reception is desired, set bit RX9.
7. Start reception by setting the SREN bit or for continuous reception, set the CREN bit.
8. Interrupt flag bit RCIF will be set when reception of a character is complete. An interrupt will be generated if the enable bit RCIE was set.
9. Read the RC1STA register to get the ninth bit (if enabled) and determine if any error occurred during reception.
10. Read the 8-bit received data by reading the RC1REG register.
11. If an overrun error occurs, clear the error by either clearing the CREN bit of the RC1STA register or by clearing the SPEN bit which resets the EUSART1.

FIGURE 31-12: SYNCHRONOUS RECEPTION (MASTER MODE, SREN)



PIC16(L)F18326/18346

REGISTER 31-7: SP1BRGH^(1, 2): BAUD RATE GENERATOR HIGH REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
SP1BRG<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 **SP1BRG<15:8>**: Upper eight bits of the Baud Rate Generator

Note 1: SP1BRGH value is ignored for all modes unless BAUD1CON<BRG16> is active.

2: Writing to SP1BRGH resets the BRG counter.

TABLE 31-2: SUMMARY OF REGISTERS ASSOCIATED WITH EUSART1

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
TRISA	—	—	TRISA5	TRISA4	— ⁽²⁾	TRISA2	TRISA1	TRISA0	143
ANSELA	—	—	ANSA5	ANSA4	—	ANSA2	ANSA1	ANSA0	144
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	149
ANSELB ⁽¹⁾	ANSB7	ANSB6	ANSB5	ANSB4	—	—	—	—	150
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	155
ANSELC	ANSC7 ⁽¹⁾	ANSC6 ⁽¹⁾	ANSC5	ANSC4	ANSC3	ANSC2	ANSC1	ANSC0	157
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
RC1STA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	384
TX1STA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	383
BAUD1CON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	385
RC1REG	RC1REG<7:0>								386
TX1REG	TX1REG<7:0>								386
SP1BRGL	SP1BRG<7:0>								386
SP1BRGH	SP1BRG<15:8>								387
RXPPS	—	—	—	RXPPS<4:0>				162	
CLCxSELY	—	—	—	LCxDyS<4:0>				229	
MDSRC	—	—	—	—	MDMS<3:0>				272

Legend: — = unimplemented location, read as '0'. Shaded cells are not used for the EUSART1 module.

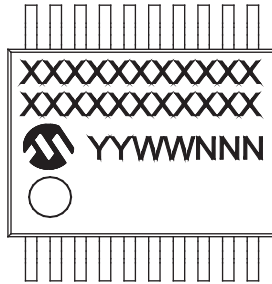
Note 1: PIC16(L)F18346 only.

2: Unimplemented, read as '1'.

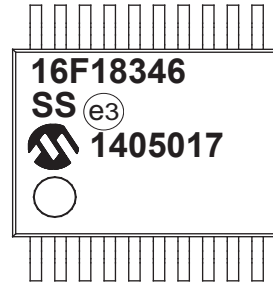
PIC16(L)F18326/18346

Package Marking Information (Continued)

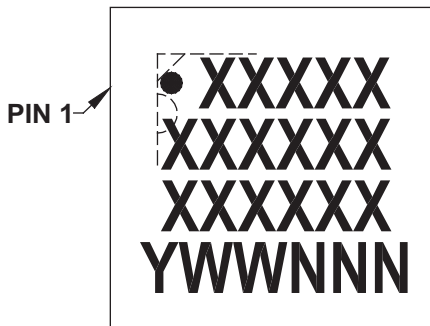
20-Lead SSOP (5.30 mm)



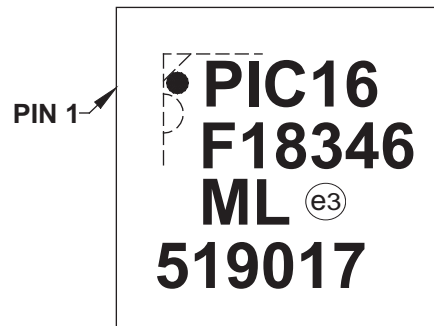
Example



20-Lead UQFN (4x4x0.5 mm)



Example



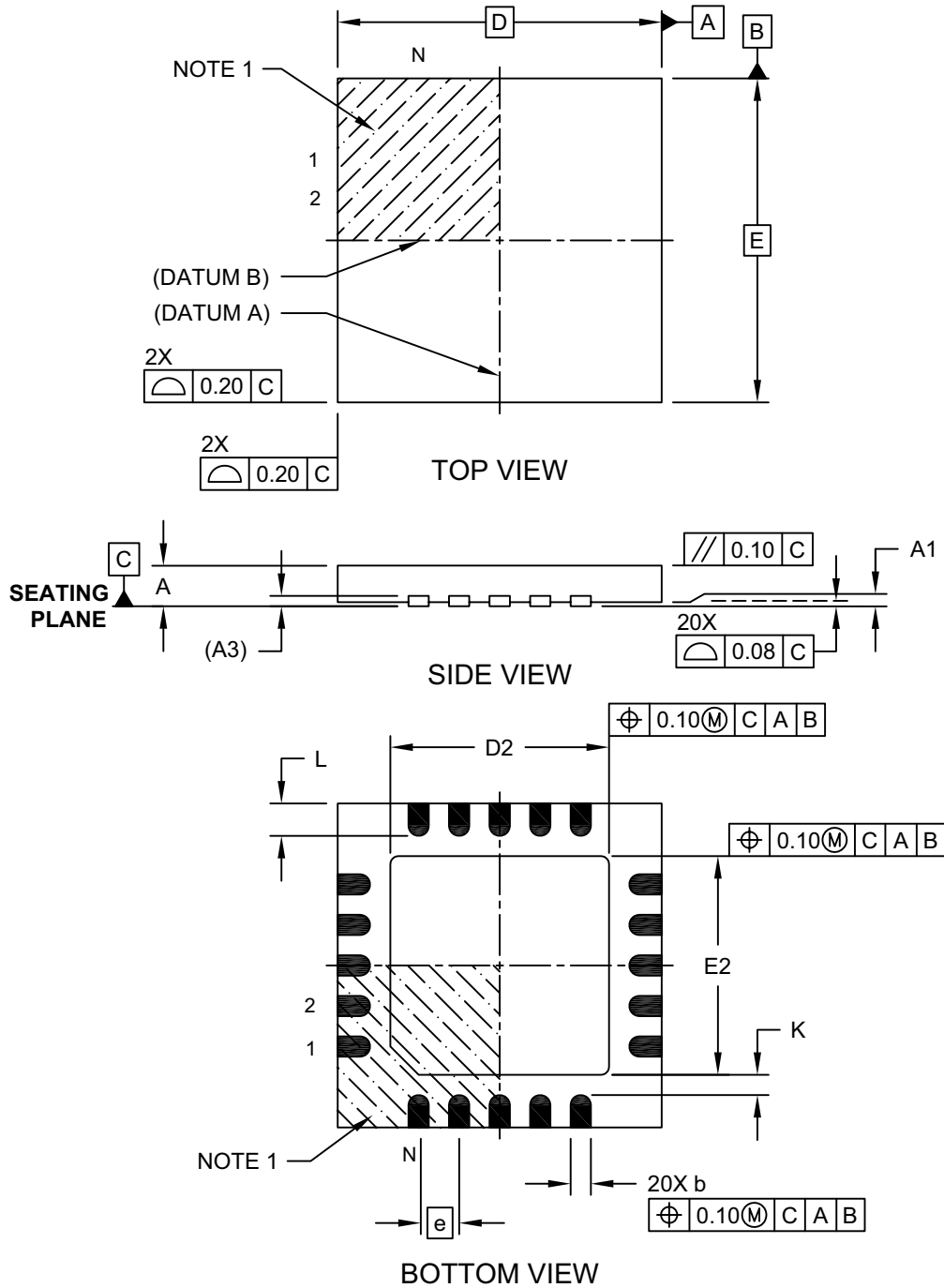
Legend:	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC® designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator (e3) can be found on the outer packaging for this package.

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

PIC16(L)F18326/18346

20-Lead Ultra Thin Plastic Quad Flat, No Lead Package (GZ) - 4x4x0.5 mm Body [UQFN]

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



Microchip Technology Drawing C04-255A Sheet 1 of 2