



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	18
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 17x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SSOP (0.209", 5.30mm Width)
Supplier Device Package	20-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf18345-i-ss

PIC16(L)F18325/18345

- Up to 18 I/O Pins:
 - Individually programmable pull-ups
 - Slew rate control
 - Interrupt-on-change with edge-select
 - Input level selection control (ST or TTL)
 - Digital open-drain enable
- Peripheral Pin Select (PPS):
 - I/O pin remapping of digital peripherals
- Timer modules:
 - Timer0:
 - 8/16-bit timer/counter
 - Synchronous or asynchronous operation
 - Programmable prescaler/postscaler
 - Time base for capture/compare function
 - Timer1/3/5 with gate control:
 - 16-bit timer/counter
 - Programmable internal or external clock sources
 - Multiple gate sources
 - Multiple gate modes
 - Time base for capture/compare function
 - Timer2/4/6:
 - 8-bit timers
 - Programmable prescaler/postscaler
 - Time base for PWM function

Analog Peripherals

- 10-bit Analog-to-Digital Converter (ADC):
 - 17 external channels
 - Conversion available during Sleep
- Comparator:
 - Two comparators
 - Fixed Voltage Reference at non-inverting input(s)
 - Comparator outputs externally accessible
- 5-bit Digital-to-Analog Converter (DAC):
 - 5-bit resolution, rail-to-rail
 - Positive Reference Selection
 - Unbuffered I/O pin output
 - Internal connections to ADCs and comparators
- Voltage Reference:
 - Fixed Voltage Reference with 1.024V, 2.048V and 4.096V output levels

Flexible Oscillator Structure

- High-Precision Internal Oscillator:
 - Software-selectable frequency range up to 32 MHz
 - $\pm 2\%$ at nominal 4 MHz calibration point
- 4x PLL with External Sources
- Low-Power Internal 31 kHz Oscillator (LFINTOSC)
- External Low-Power 32 kHz Crystal Oscillator (SOSC)
- External Oscillator Block with:
 - Three Crystal/Resonator modes up to 20 MHz
 - Three External Clock modes up to 20 MHz
 - Fail-Safe Clock Monitor
 - Detects clock source failure
 - Oscillator Start-up Timer (OST)
 - Ensures stability of crystal oscillator sources

PIC16(L)F18325/18345

2.5 External Oscillator Pins

Many microcontrollers have options for at least two oscillators: a high-frequency primary oscillator and a low-frequency secondary oscillator (refer to **Section 7.0 “Oscillator Module”** for details).

The oscillator circuit should be placed on the same side of the board as the device. Place the oscillator circuit close to the respective oscillator pins with no more than 0.5 inch (12 mm) between the circuit components and the pins. The load capacitors should be placed next to the oscillator itself, on the same side of the board.

Use a grounded copper pour around the oscillator circuit to isolate it from surrounding circuits. The grounded copper pour should be routed directly to the MCU ground. Do not run any signal traces or power traces inside the ground pour. Also, if using a two-sided board, avoid any traces on the other side of the board where the crystal is placed.

Layout suggestions are shown in Figure 2-2. In-line packages may be handled with a single-sided layout that completely encompasses the oscillator pins. With fine-pitch packages, it is not always possible to completely surround the pins and components. A suitable solution is to tie the broken guard sections to a mirrored ground layer. In all cases, the guard trace(s) must be returned to ground.

In planning the application's routing and I/O assignments, ensure that adjacent port pins, and other signals in close proximity to the oscillator, are benign (i.e., free of high frequencies, short rise and fall times, and other similar noise).

For additional information and design guidance on oscillator circuits, please refer to these Microchip Application Notes, available at the corporate website (www.microchip.com):

- AN826, “Crystal Oscillator Basics and Crystal Selection for rPIC™ and PICmicro® Devices”
- AN849, “Basic PICmicro® Oscillator Design”
- AN943, “Practical PICmicro® Oscillator Analysis and Design”
- AN949, “Making Your Oscillator Work”

2.6 Unused I/Os

Unused I/O pins should be configured as outputs and driven to a logic low state. Alternatively, connect a 1 kΩ to 10 kΩ resistor to V_{SS} on unused pins and drive the output logic low.

FIGURE 2-2: SUGGESTED PLACEMENT OF THE OSCILLATOR CIRCUIT

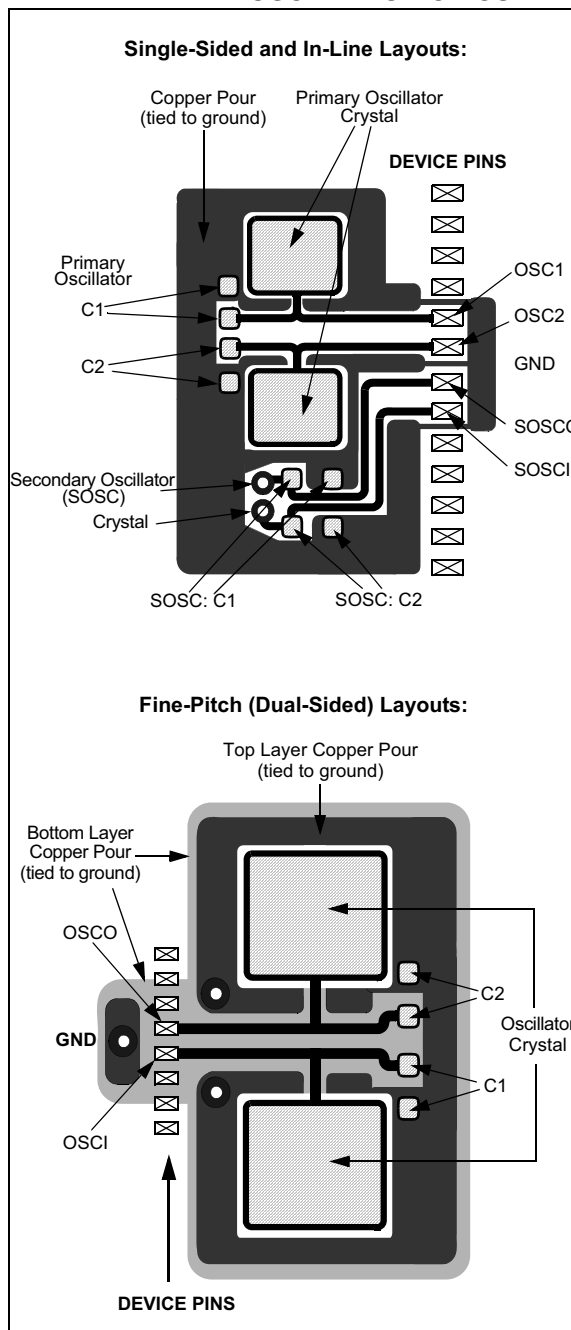


TABLE 4-4: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-31 (CONTINUED)

Address	Name	PIC16(L)F18325 PIC16(L)F18345	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets
Bank 10-11												
CPU CORE REGISTERS; see Table 4-2 for specifics												
50Ch to 51Fh	—	—	Unimplemented								—	—
58Ch to 59Fh	—	—	Unimplemented								—	—
Bank 12												
60Ch	—	—	Unimplemented								—	—
60Dh	—	—	Unimplemented								—	—
60Eh	—	—	Unimplemented								—	—
60Fh	—	—	Unimplemented								—	—
610h	—	—	Unimplemented								—	—
611h	—	—	Unimplemented								—	—
612h	—	—	Unimplemented								—	—
613h	—	—	Unimplemented								—	—
614h	—	—	Unimplemented								—	—
615h	—	—	Unimplemented								—	—
616h	—	—	Unimplemented								—	—
617h	PWM5DCL		PWM5DC<1:0>		—	—	—	—	—	—	xx-- ----	uu-- ----
618h	PWM5DCH		PWM5DC<9:2>								xxxx xxxx	uuuu uuuu
619h	PWM5CON		PWM5EN	—	PWM5OUT	PWM5POL	—	—	—	—	0-00 ----	0-00 ----
61Ah	PWM6DCL		PWM6DC<1:0>		—	—	—	—	—	—	xx-- ----	uu-- ----
61Bh	PWM6DCH		PWM6DC<9:2>								xxxx xxxx	uuuu uuuu
61Ch	PWM6CON		PWM6EN	—	PWM6OUT	PWM6POL	—	—	—	—	0-00 ----	0-00 ----
61Dh to 61Eh	—	—	Unimplemented								—	—
61Fh	PWMTMRS		—	—	—	—	P6TSEL<1:0>		P5TSEL<1:0>		---- 0101	---- 0101

Legend: x = unknown, u = unchanged, q = depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations unimplemented, read as '0'.

Note 1: Only on PIC16F18325/18345.

Note 2: Register accessible from both User and ICD Debugger.

TABLE 4-4: SPECIAL FUNCTION REGISTER SUMMARY BANKS 0-31 (CONTINUED)

Address	Name	PIC16(L)F18325 PIC16(L)F18345	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other Resets	
Bank 13													
CPU CORE REGISTERS; see Table 4-2 for specifics													
68Ch	—	—	Unimplemented									—	—
68Dh	—	—	Unimplemented									—	—
68Eh	—	—	Unimplemented									—	—
68Fh	—	—	Unimplemented									—	—
690h	—	—	Unimplemented									—	—
691h	CWG1CLKCON		—	—	—	—	—	—	—	CS	---- --0	---- --0	
692h	CWG1DAT		—	—	—	—	DAT<3:0>				---- 0000	---- 0000	
693h	CWG1DBR		—	—	DBR<5:0>						--00 0000	--00 0000	
694h	CWG1DBF		—	—	DBF<5:0>						--00 0000	--00 0000	
695h	CWG1CON0		EN	LD	—	—	—	MODE<2:0>			00-- -000	00-- -000	
696h	CWG1CON1		—	—	IN	—	POLD	POLC	POLB	POLA	--x- 0000	--x- 0000	
697h	CWG1AS0		SHUTDOWN	REN	LSBD<1:0>		LSAC<1:0>		—	—	0001 01--	0001 01--	
698h	CWG1AS1		—	—	—	AS4E	AS3E	AS2E	AS1E	AS0E	---0 0000	---0 0000	
699h	CWG1STR		OVRD	OVRC	OVRB	OVRA	STRD	STRC	STRB	STRA	0000 0000	0000 0000	
69Ah to 69Fh	—	—	Unimplemented									—	—

Legend: x = unknown, u = unchanged, q = depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations unimplemented, read as '0'.

Note 1: Only on PIC16F18325/18345.

Note 2: Register accessible from both User and ICD Debugger.

PIC16(L)F18325/18345

4.5 Indirect Addressing

The INDFn registers are not physical registers. Any instruction that accesses an INDFn register actually accesses the register at the address specified by the File Select Registers (FSR). If the FSRn address specifies one of the two INDFn registers, the read will return '0' and the write will not occur (though Status bits may be affected). The FSRn register value is created by the pair FSRnH and FSRnL.

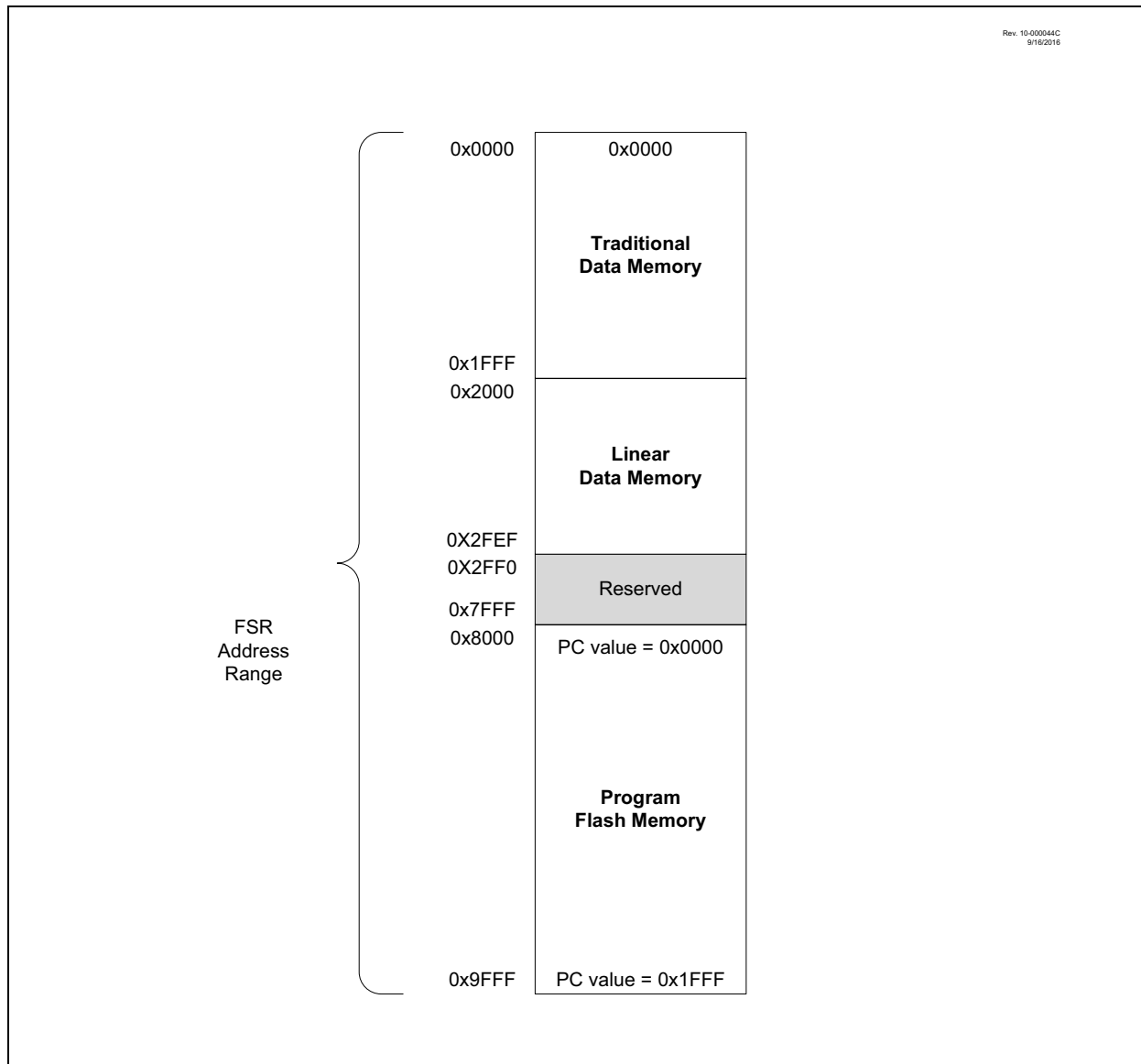
The FSR registers form a 16-bit address that allows an addressing space with 65536 locations. These locations are divided into four memory regions:

- Traditional/Banked Data Memory
- Linear Data Memory
- Program Flash Memory
- EEPROM

4.5.1 TRADITIONAL/BANKED DATA MEMORY

The banked data memory is a region from FSR address 0x000 to FSR address 0xFFFF. The addresses correspond to the absolute addresses of all SFR, GPR and common registers.

FIGURE 4-8: INDIRECT ADDRESSING



6.1 Power-on Reset (POR)

The POR circuit holds the device in Reset until VDD has reached an acceptable level for minimum operation. Slow rising VDD, fast operating speeds or analog performance may require greater than minimum VDD. The PWRT, BOR or MCLR features can be used to extend the start-up period until all device operation conditions have been met.

6.2 Brown-out Reset (BOR)

The BOR circuit holds the device in Reset while VDD is below a selectable minimum level. Between the POR and BOR, complete voltage range coverage for execution protection can be implemented.

The Brown-out Reset module has four operating modes controlled by the BOREN<1:0> bits in Configuration Words. The four operating modes are:

- BOR is always on
- BOR is off when in Sleep
- BOR is controlled by software
- BOR is always off

Refer to Table 6-1 for more information.

The Brown-out Reset voltage level is selectable by configuring the BORV bit in Configuration Words.

A VDD noise rejection filter prevents the BOR from triggering on small events. If VDD falls below VBOR for a duration greater than parameter TBORDC, the device will reset, and the BOR bit of the PCON0 register will be cleared, indicating that a Brown-out Reset condition occurred. See Figure 6-2 for more information.

TABLE 6-1: BOR OPERATING MODES

BOREN<1:0>	SBOREN	Device Mode	BOR Mode	Instruction Execution upon: Release of POR or Wake-up from Sleep
11	X	X	Active	In these specific cases, "Release of POR" and "Wake-up from Sleep", there is no delay in start-up. The BOR ready flag, (BORRDY = 1), will be set before the CPU is ready to execute instructions because the BOR circuit is forced on by the BOREN<1:0> bits.
10	X	Awake	Active	Waits for release of BOR (BORRDY = 1)
		Sleep	Disabled	BOR ignored when asleep
01	1	X	Active	In these specific cases, "Release of POR" and "Wake-up from Sleep", there is no delay in start-up. The BOR ready flag, (BORRDY = 1), will be set before the CPU is ready to execute instructions because the BOR circuit is forced on by the BOREN<1:0> bits.
	0	X	Disabled	Begins immediately (BORRDY = x)
00	X	X	Disabled	

PIC16(L)F18325/18345

REGISTER 8-5: PIE3: PERIPHERAL INTERRUPT ENABLE REGISTER 3

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
OSFIE	CSWIE	TMR3GIE	TMR3IE	CLC4IE	CLC3IE	CLC2IE	CLC1IE
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 **OSFIE:** Oscillator Fail Interrupt Enable bit.
1 = Enables the Oscillator Fail interrupt
0 = Disables the Oscillator Fail interrupt
- bit 6 **CSWIE:** Clock Switch Complete Interrupt Enable bit
1 = The clock switch module interrupt is enabled
0 = The clock switch module interrupt is not enabled
- bit 5 **TMR3GIE:** Timer3 Gate Interrupt Enable bit
1 = Timer3 Gate interrupt is enabled
0 = Timer3 Gate interrupt is not enabled
- bit 4 **TMR3IE:** TMR3 Overflow Interrupt Enable bit
1 = TMR3 overflow interrupt is enabled
0 = TMR3 overflow interrupt is not enabled
- bit 3 **CLC4IE:** CLC4 Interrupt Flag bit
1 = CLC4 interrupt is enabled
0 = CLC4 interrupt is not enabled
- bit 2 **CLC3IE:** CLC3 Interrupt Flag bit
1 = CLC3 interrupt is enabled
0 = CLC3 interrupt is not enabled
- bit 1 **CLC2IE:** CLC2 Interrupt Enable bit
1 = CLC2 interrupt enabled
0 = CLC2 interrupt disabled
- bit 0 **CLC1IE:** CLC1 Interrupt Enable bit
1 = CLC1 interrupt enabled
0 = CLC1 interrupt disabled

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

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)

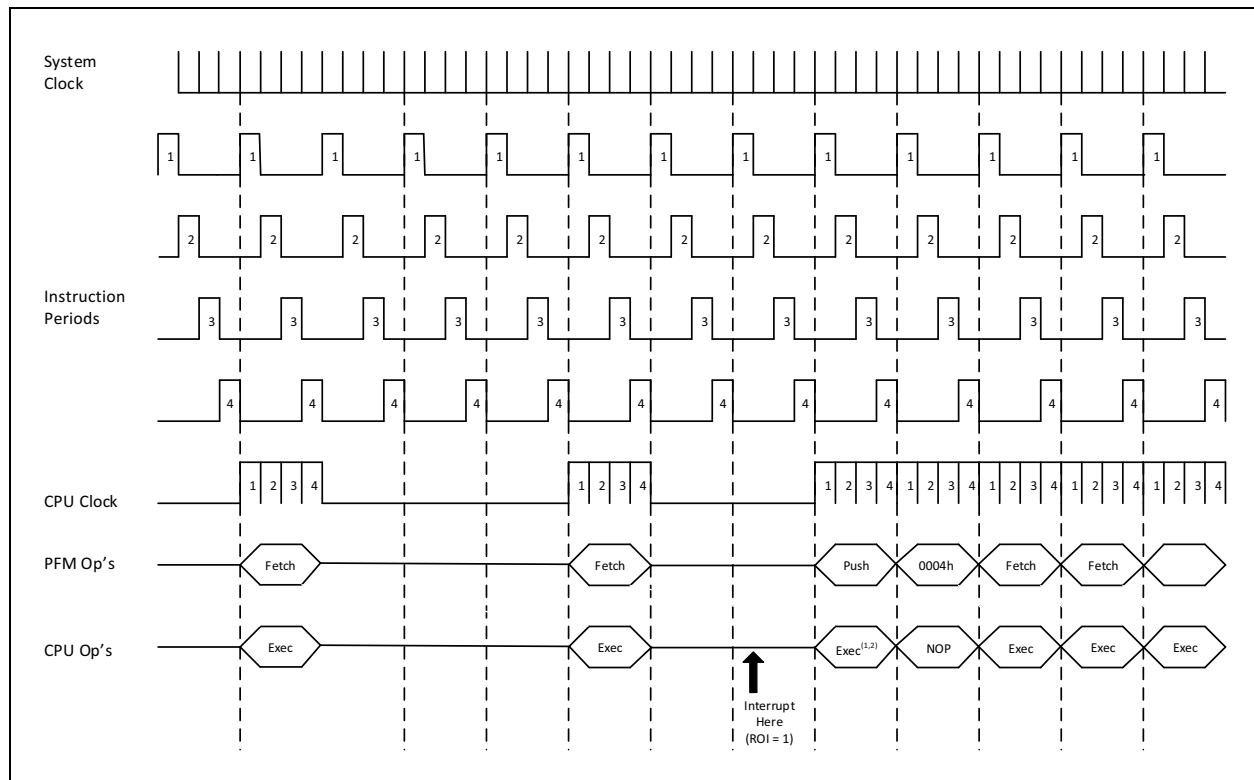
As with normal operation, the program memory fetches for the next instruction cycle. The instruction clocks to the peripherals continue throughout.

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



PIC16(L)F18325/18345

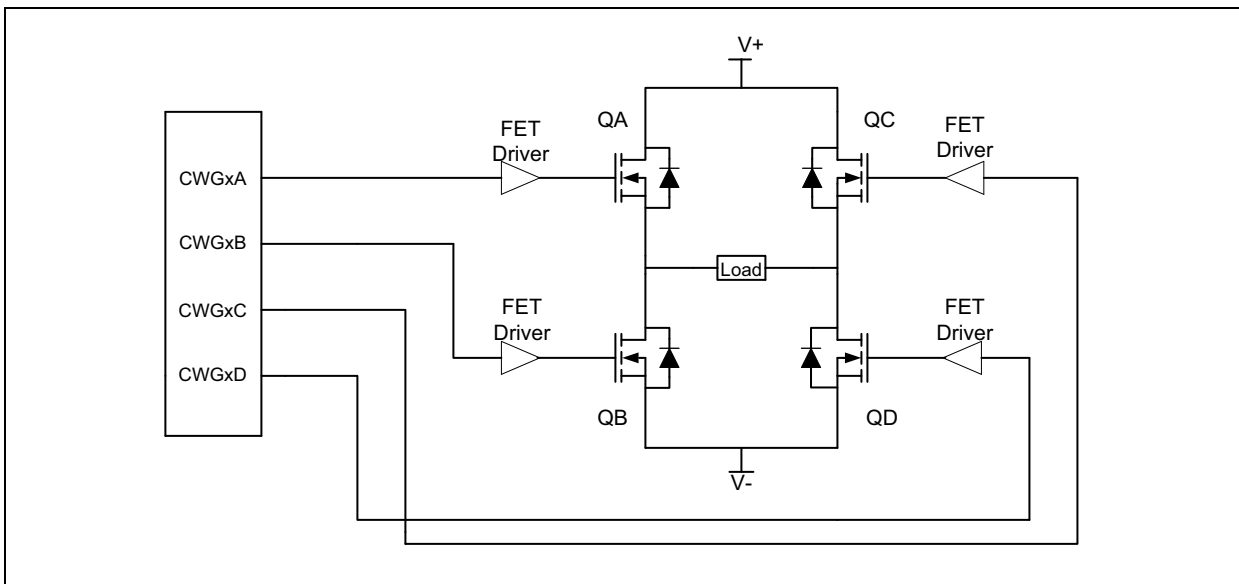
20.2.4 FULL-BRIDGE MODES

In Forward and Reverse Full-Bridge modes, three outputs drive static values while the fourth is modulated by the data input. Dead-band control is described in **Section 20.2.3 “Steering Modes”** and **Section 20.6 “Dead-Band Control”**. Steering modes are not used with either of the Full-Bridge modes.

The mode selection may be toggled between forward and reverse (changing MODE<2:0>) without clearing EN.

When connected as shown in Figure 20-5, the outputs are appropriate for a full-bridge motor driver. Each CWG output signal has independent polarity control, so the circuit can be adapted to high-active and low-active drivers.

FIGURE 20-5: EXAMPLE OF FULL-BRIDGE APPLICATION



PIC16(L)F18325/18345

TABLE 22-1: ADC CLOCK PERIOD (TAD) Vs. DEVICE OPERATING FREQUENCIES

ADC Clock Period (TAD)		Device Frequency (Fosc)					
ADC Clock Source	ADCS<2:0>	32 MHz	20 MHz	16 MHz	8 MHz	4 MHz	1 MHz
Fosc/2	000	62.5ns ⁽²⁾	100 ns ⁽²⁾	125 ns ⁽²⁾	250 ns ⁽²⁾	500 ns ⁽²⁾	2.0 μs
Fosc/4	100	125 ns ⁽²⁾	200 ns ⁽²⁾	250 ns ⁽²⁾	500 ns ⁽²⁾	1.0 μs	4.0 μs
Fosc/8	001	0.5 μs ⁽²⁾	400 ns ⁽²⁾	0.5 μs ⁽²⁾	1.0 μs	2.0 μs	8.0 μs ⁽³⁾
Fosc/16	101	800 ns	800 ns	1.0 μs	2.0 μs	4.0 μs	16.0 μs ⁽³⁾
Fosc/32	010	1.0 μs	1.6 μs	2.0 μs	4.0 μs	8.0 μs ⁽³⁾	32.0 μs ⁽²⁾
Fosc/64	110	2.0 μs	3.2 μs	4.0 μs	8.0 μs ⁽³⁾	16.0 μs ⁽²⁾	64.0 μs ⁽²⁾
ADCRC	x11	1.0-6.0 μs ^(1,4)	1.0-6.0 μs ^(1,4)	1.0-6.0 μs ^(1,4)	1.0-6.0 μs ^(1,4)	1.0-6.0 μs ^(1,4)	1.0-6.0 μs ^(1,4)

Legend: Shaded cells are outside of recommended range.

Note 1: See TAD parameter for ADCRC source typical TAD value.

2: These values violate the required TAD time.

3: Outside the recommended TAD time.

4: The ADC clock period (TAD) and total ADC conversion time can be minimized when the ADC clock is derived from the system clock Fosc. However, the ADCRC oscillator source must be used when conversions are to be performed with the device in Sleep mode.

FIGURE 22-2: ANALOG-TO-DIGITAL CONVERSION TAD CYCLES

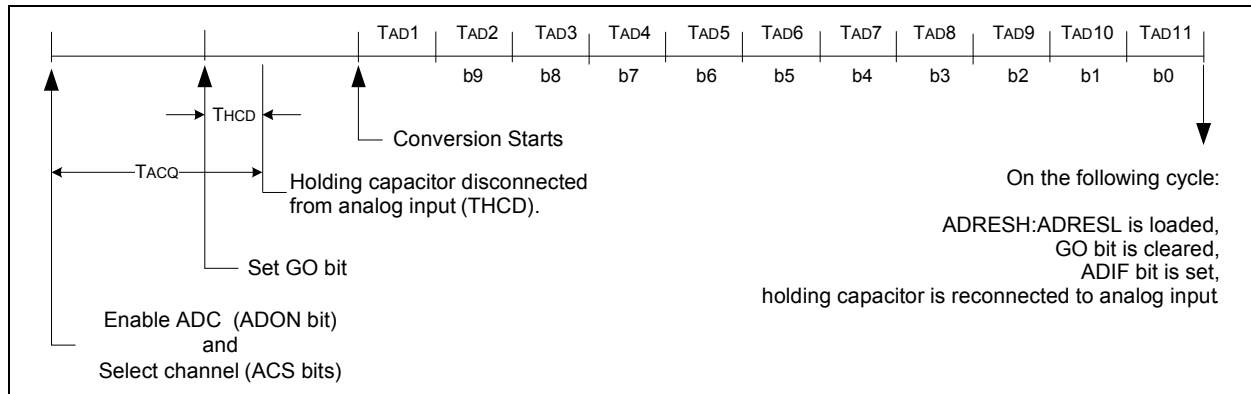


FIGURE 22-4: ANALOG INPUT MODEL

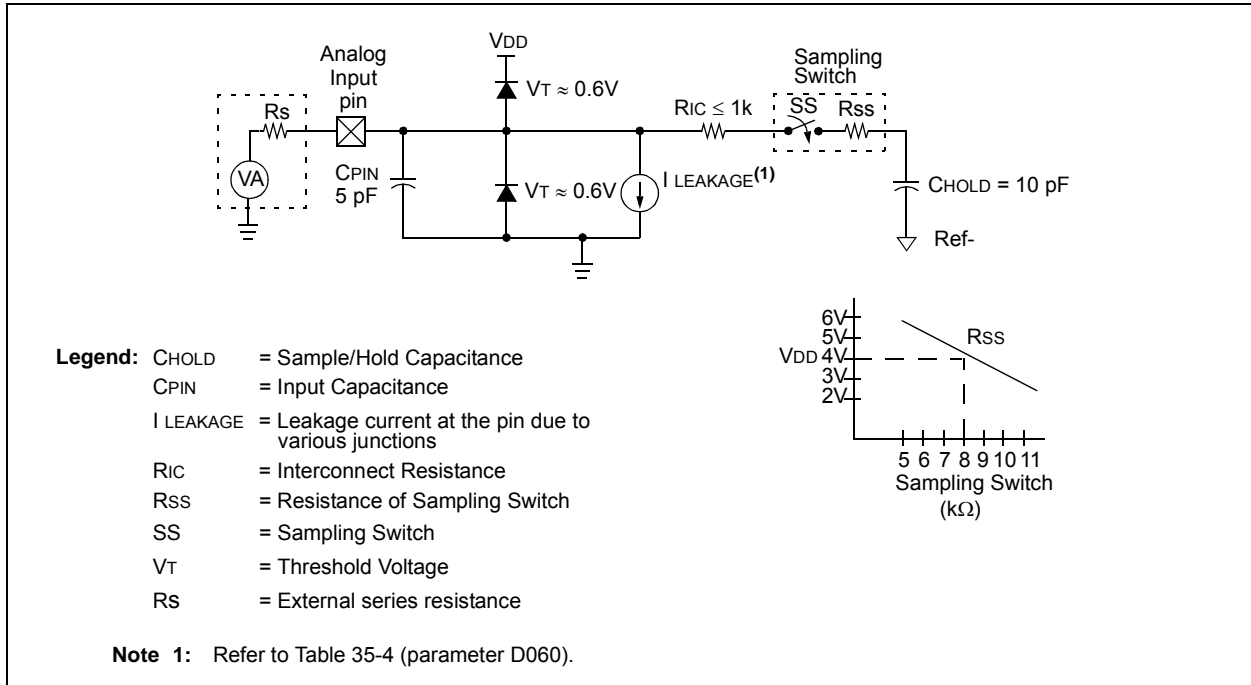
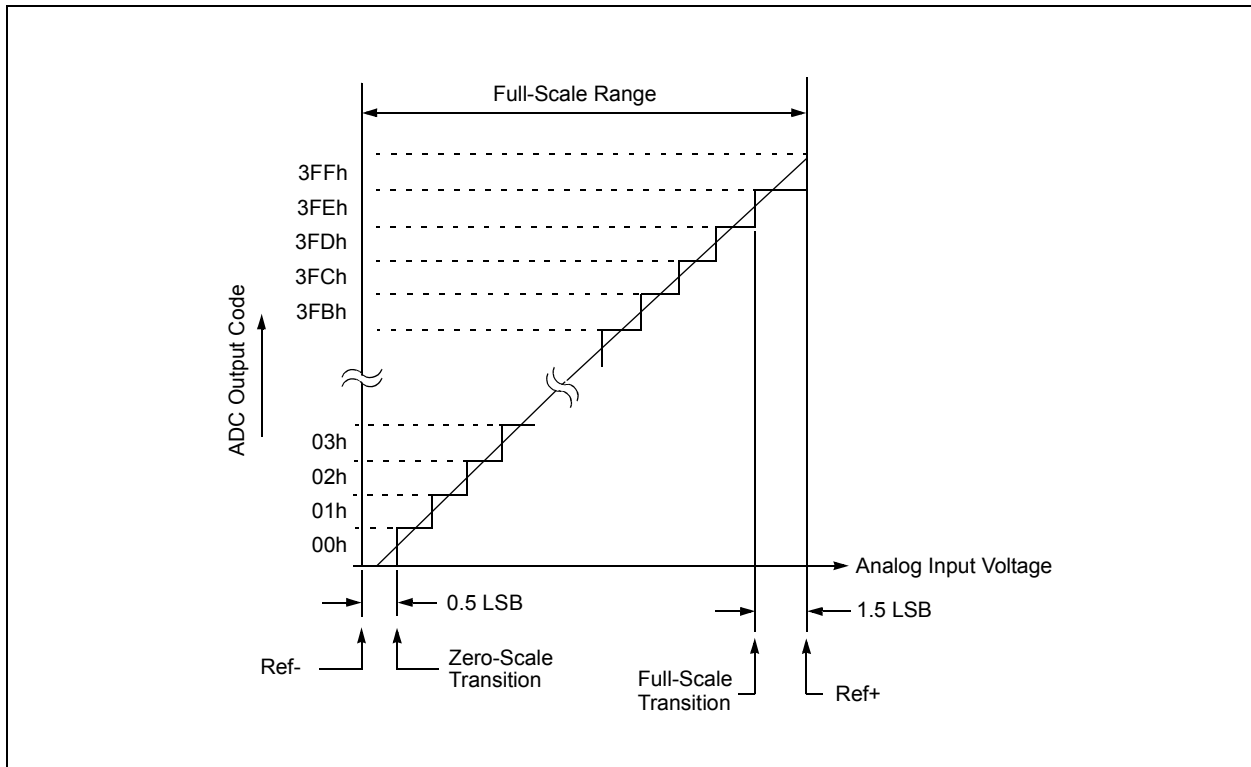


FIGURE 22-5: ADC TRANSFER FUNCTION



PIC16(L)F18325/18345

FIGURE 25-2: ON OFF KEYING (OOK) SYNCHRONIZATION

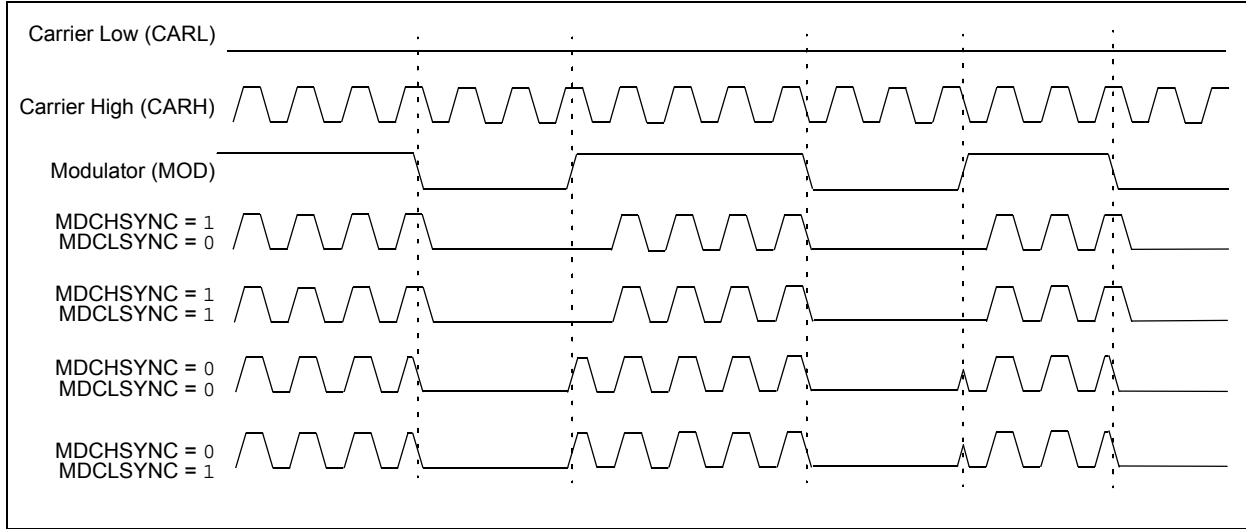


FIGURE 25-3: NO SYNCHRONIZATION (MDSHSYNC = 0, MDCLSYNC = 0)

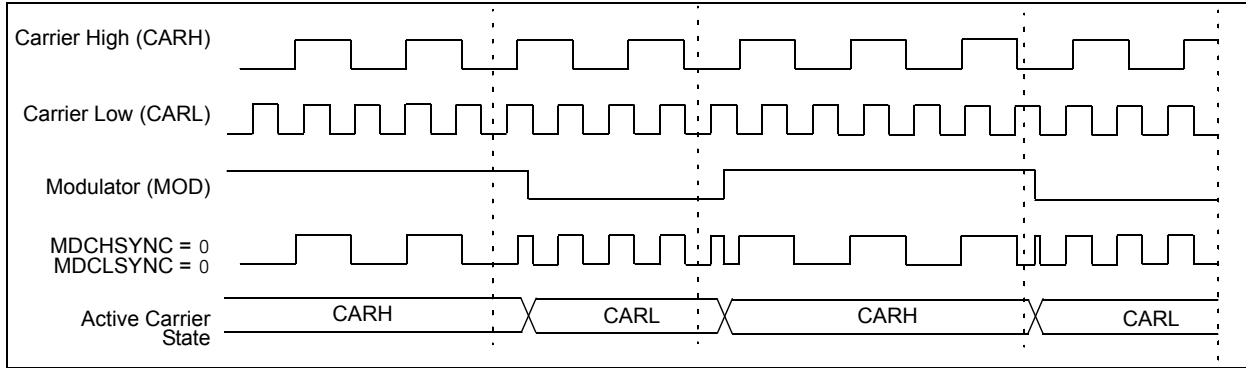
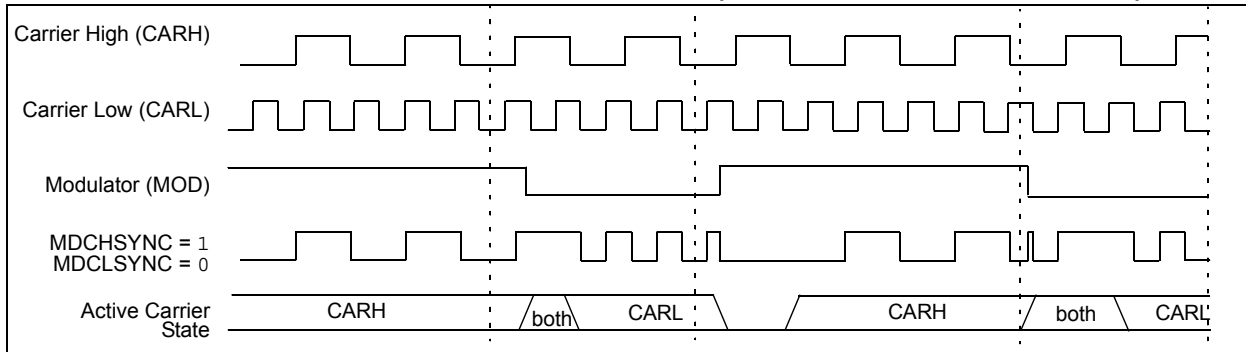


FIGURE 25-4: CARRIER HIGH SYNCHRONIZATION (MDSHSYNC = 1, MDCLSYNC = 0)



29.4 PWM Overview

Pulse-Width Modulation (PWM) is a scheme that provides power to a load by switching quickly between fully on and fully off states. The PWM signal resembles a square wave where the high portion of the signal is considered the on state and the low portion of the signal is considered the off state. The high portion, also known as the pulse width, can vary in time and is defined in steps. A larger number of steps applied, which lengthens the pulse width, also supplies more power to the load. Lowering the number of steps applied, which shortens the pulse width, supplies less power. The PWM period is defined as the duration of one complete cycle or the total amount of on and off time combined.

PWM resolution defines the maximum number of steps that can be present in a single PWM period. A higher resolution allows for more precise control of the pulse width time and in turn the power that is applied to the load.

The term duty cycle describes the proportion of the on time to the off time and is expressed in percentages, where 0% is fully off and 100% is fully on. A lower duty cycle corresponds to less power applied and a higher duty cycle corresponds to more power applied.

Figure 29-3 shows a typical waveform of the PWM signal.

29.4.1 STANDARD PWM OPERATION

The standard PWM mode generates a Pulse-Width Modulation (PWM) signal on the CCPx pin with up to ten bits of resolution. The period, duty cycle, and resolution are controlled by the following registers:

- PR2/4/6 registers
- T2/4/6CON registers
- CCPRxL registers
- CCPxCON registers

Figure 29-4 shows a simplified block diagram of PWM operation.

Note: The corresponding TRIS bit must be cleared to enable the PWM output on the CCPx pin.

FIGURE 29-3: CCP PWM OUTPUT SIGNAL

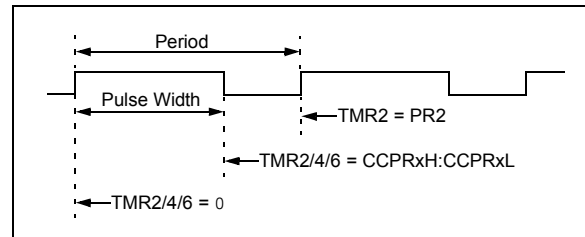
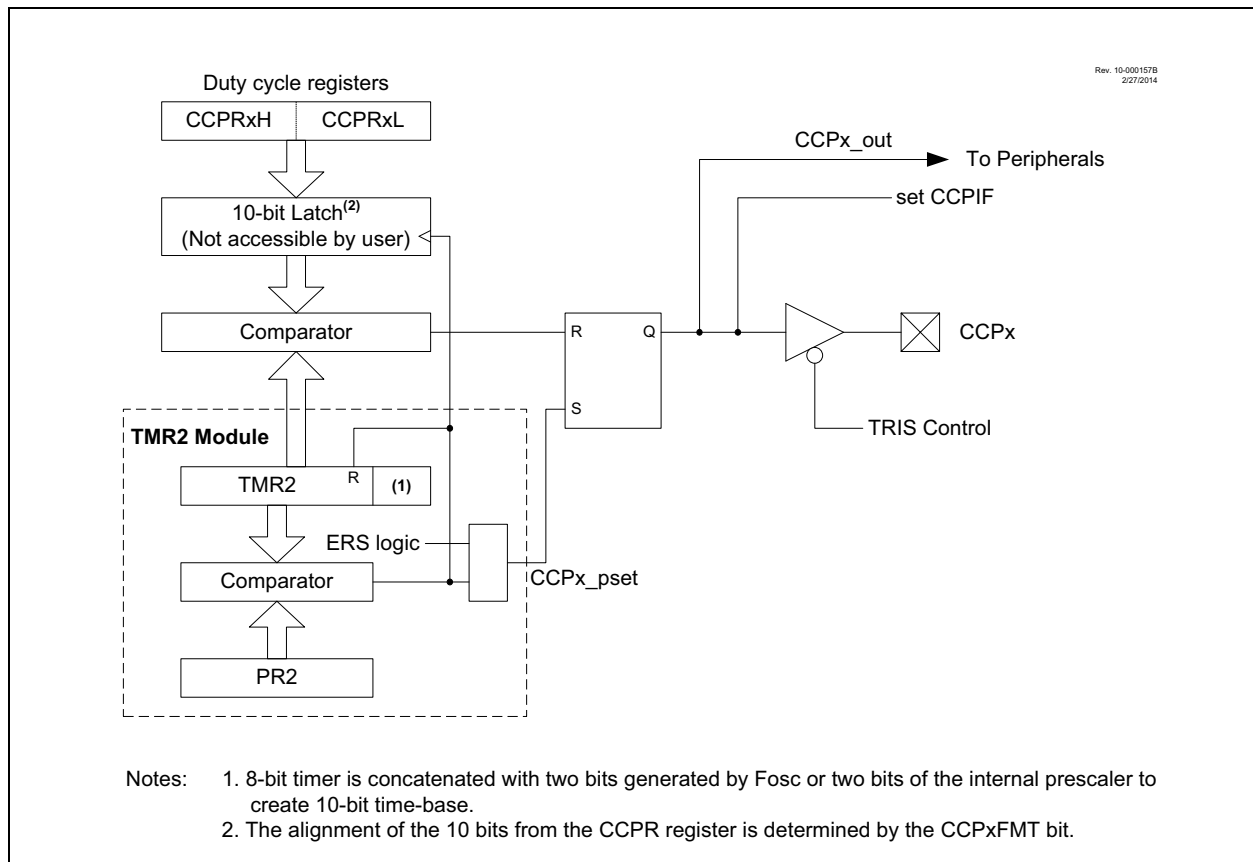


FIGURE 29-4: SIMPLIFIED PWM BLOCK DIAGRAM



PIC16(L)F18325/18345

31.0 ENHANCED UNIVERSAL SYNCHRONOUS ASYNCHRONOUS RECEIVER TRANSMITTER (EUSART1)

The Enhanced Universal Synchronous Asynchronous Receiver Transmitter (EUSART1) module is a serial I/O communications peripheral. It contains all the clock generators, shift registers and data buffers necessary to perform an input or output serial data transfer independent of device program execution. The EUSART1, also known as a Serial Communications Interface (SCI), can be configured as a full-duplex asynchronous system or half-duplex synchronous system. Full-Duplex mode is useful for communications with peripheral systems, such as CRT terminals and personal computers. Half-Duplex Synchronous mode is intended for communications with peripheral devices, such as A/D or D/A integrated circuits, serial EEPROMs or other microcontrollers. These devices typically do not have internal clocks for baud rate generation and require the external clock signal provided by a master synchronous device.

The EUSART1 module includes the following capabilities:

- Full-duplex asynchronous transmit and receive
- Two-character input buffer
- One-character output buffer
- Programmable 8-bit or 9-bit character length
- Address detection in 9-bit mode
- Input buffer overrun error detection
- Received character framing error detection
- Half-duplex synchronous master
- Half-duplex synchronous slave
- Programmable clock polarity in synchronous modes
- Sleep operation

The EUSART1 module implements the following additional features, making it ideally suited for use in Local Interconnect Network (LIN) bus systems:

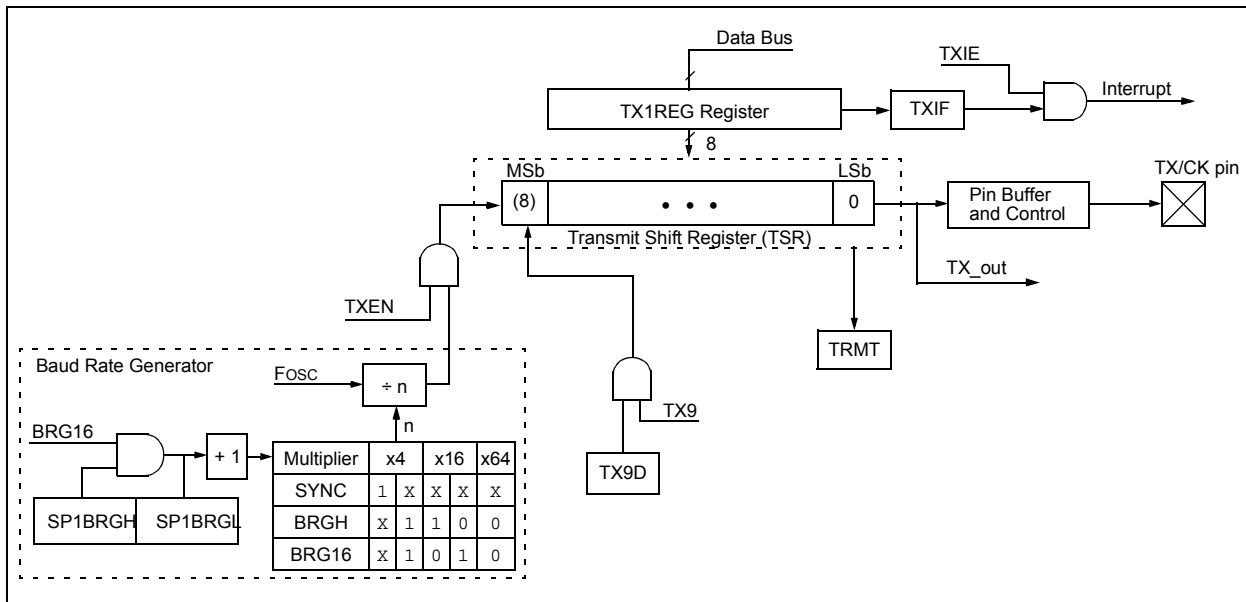
- Automatic detection and calibration of the baud rate
- Wake-up on Break reception
- 13-bit Break character transmit

Block diagrams of the EUSART1 transmitter and receiver are shown in Figure 31-1 and Figure 31-2.

The EUSART1 transmit output (TX_out) is available to the TX/CK pin and internally to the following peripherals:

- Configurable Logic Cell (CLC)

FIGURE 31-1: EUSART1 TRANSMIT BLOCK DIAGRAM



PIC16(L)F18325/18345

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

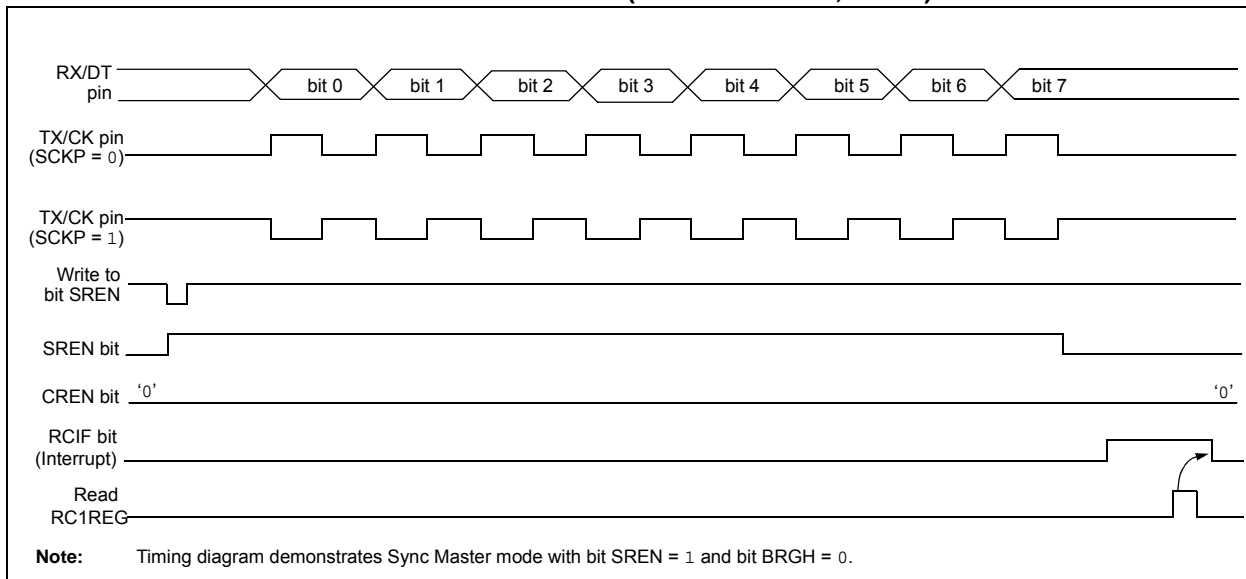
The EUSART1 supports 9-bit character reception. When the RX9 bit of the RC1STA register is set the EUSART1 will shift nine bits into the RSR for each

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 Set-up

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)F18325/18345

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	156
ANSELC	ANSC7 ⁽¹⁾	ANSC6 ⁽¹⁾	ANSC5	ANSC4	ANSC3	ANSC2	ANSC1	ANSC0	157
INTCON	GIE	PEIE	—	—	—	—	—	INTEDG	101
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	BCL1IF	TMR2IF	TMR1IF	108
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	BCL1IE	TMR2IE	TMR1IE	103
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)F18345 only.

2: Unimplemented, read as '1'.

PIC16(L)F18325/18345

34.2 Instruction Descriptions

ADDFSR	Add Literal to FSRn
Syntax:	[<i>label</i>] ADDFSR FSRn, k
Operands:	-32 ≤ k ≤ 31 n ∈ [0, 1]
Operation:	FSR(n) + k → FSR(n)
Status Affected:	None
Description:	The signed 6-bit literal 'k' is added to the contents of the FSRnH:FSRnL register pair. FSRn is limited to the range 0000h-FFFFh. Moving beyond these bounds will cause the FSR to wrap-around.

ADDLW	Add literal and W
Syntax:	[<i>label</i>] ADDLW k
Operands:	0 ≤ k ≤ 255
Operation:	(W) + k → (W)
Status Affected:	C, DC, Z
Description:	The contents of the W register are added to the 8-bit literal 'k' and the result is placed in the W register.

ADDWF	Add W and f
Syntax:	[<i>label</i>] ADDWF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(W) + (f) → (destination)
Status Affected:	C, DC, Z
Description:	Add the contents of the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

ADDWFC	ADD W and CARRY bit to f
Syntax:	[<i>label</i>] ADDWFC f {,d}
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(W) + (f) + (C) → dest
Status Affected:	C, DC, Z
Description:	Add W, the Carry flag and data memory location 'f'. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed in data memory location 'f'.

ANDLW	AND literal with W
Syntax:	[<i>label</i>] ANDLW k
Operands:	0 ≤ k ≤ 255
Operation:	(W) .AND. (k) → (W)
Status Affected:	Z
Description:	The contents of W register are AND'ed with the 8-bit literal 'k'. The result is placed in the W register.

ANDWF	AND W with f
Syntax:	[<i>label</i>] ANDWF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(W) .AND. (f) → (destination)
Status Affected:	Z
Description:	AND the W register with register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

ASRF	Arithmetic Right Shift
Syntax:	[<i>label</i>] ASRF f {,d}
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f<7>) → dest<7> (f<7:1>) → dest<6:0>, (f<0>) → C,
Status Affected:	C, Z
Description:	The contents of register 'f' are shifted one bit to the right through the Carry flag. The MSb remains unchanged. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is stored back in register 'f'.

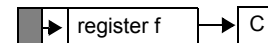
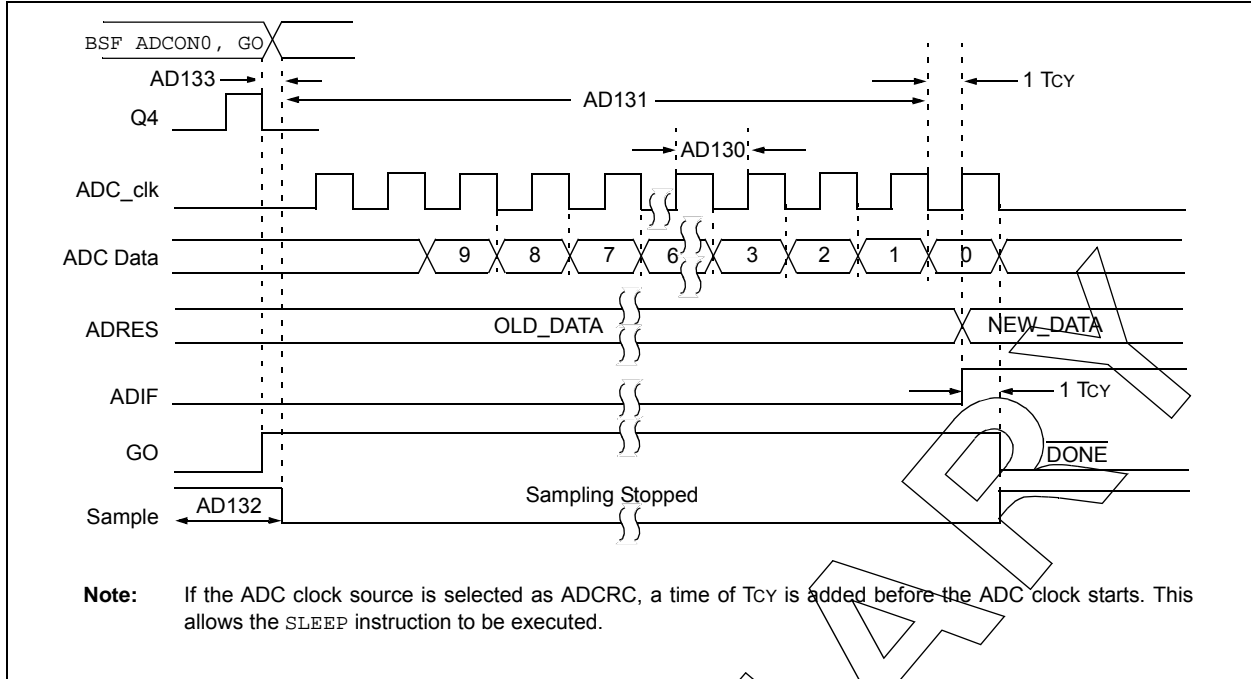


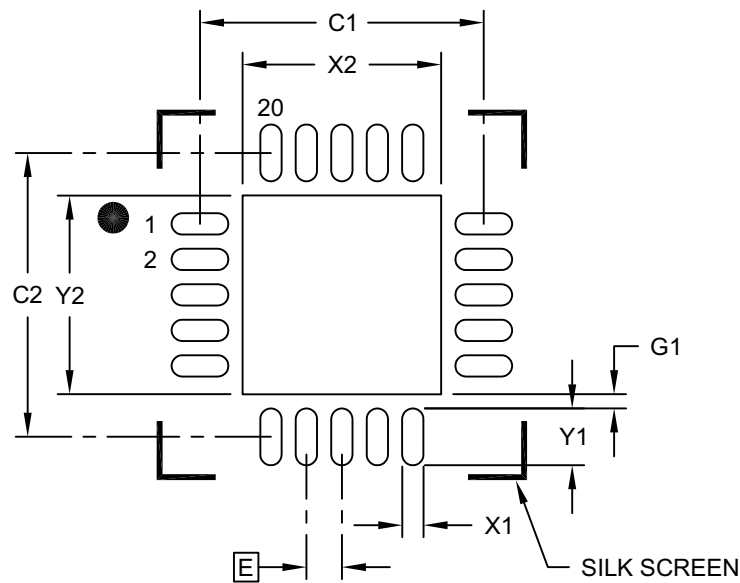
FIGURE 35-11: ADC CONVERSION TIMING (ADC CLOCK FROM ADCRC)



PIC16(L)F18325/18345

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>



RECOMMENDED LAND PATTERN

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.50 BSC		
Optional Center Pad Width	X2			2.80
Optional Center Pad Length	Y2			2.80
Contact Pad Spacing	C1		4.00	
Contact Pad Spacing	C2		4.00	
Contact Pad Width (X20)	X1			0.30
Contact Pad Length (X20)	Y1			0.80
Contact Pad to Center Pad (X20)	G1	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing C04-2255A

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as “unbreakable.”

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC[®] MCUs and dsPIC[®] DSCs, KEELoQ[®] code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

**QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
= ISO/TS 16949 =**

Trademarks

The Microchip name and logo, the Microchip logo, AnyRate, dsPIC, FlashFlex, flexPWR, Heldo, JukeBlox, KeeLoq, KeeLoq logo, Klear, LANCheck, LINK MD, MediaLB, MOST, MOST logo, MPLAB, OptoLyzer, PIC, PICSTART, PIC32 logo, RightTouch, SpyNIC, SST, SST Logo, SuperFlash and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

ClockWorks, The Embedded Control Solutions Company, ETHERSYNCH, Hyper Speed Control, HyperLight Load, IntelliMOS, mTouch, Precision Edge, and QUIET-WIRE are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Any Capacitor, AnyIn, AnyOut, BodyCom, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, Dynamic Average Matching, DAM, ECAN, EtherGREEN, In-Circuit Serial Programming, ICSP, Inter-Chip Connectivity, JitterBlocker, KlearNet, KlearNet logo, MiWi, motorBench, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, MultiTRAK, NetDetach, Omniscient Code Generation, PICDEM, PICDEM.net, PICkit, PICtail, PureSilicon, RightTouch logo, REAL ICE, Ripple Blocker, Serial Quad I/O, SQL, SuperSwitcher, SuperSwitcher II, Total Endurance, TSHARC, USBCheck, VariSense, ViewSpan, WiperLock, Wireless DNA, and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

GettIC is a registered trademarks of Microchip Technology Germany II GmbH & Co. KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

© 2015-2016, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

ISBN: 978-1-5224-1001-0