Welcome to E-XFL.COM

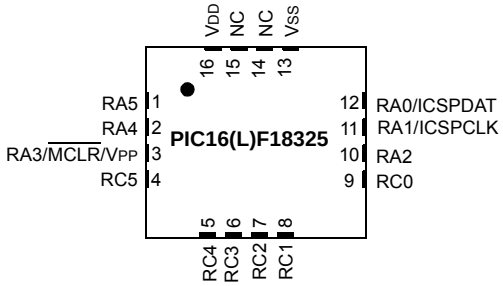
What is "Embedded - Microcontrollers"?

"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	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)	2.3V ~ 5.5V
Data Converters	A/D 17x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Through Hole
Package / Case	20-DIP (0.300", 7.62mm)
Supplier Device Package	20-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f18345-e-p

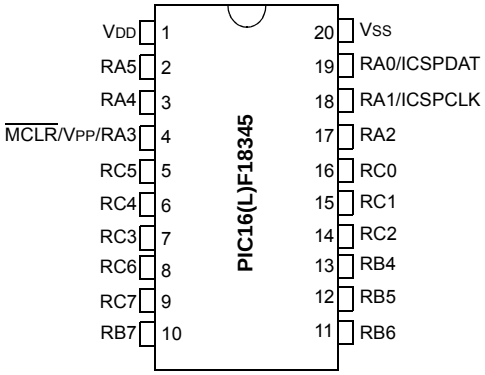
FIGURE 2: 16-PIN UQFN (4x4)



Note 1: See Table 1 for location of all peripheral functions.

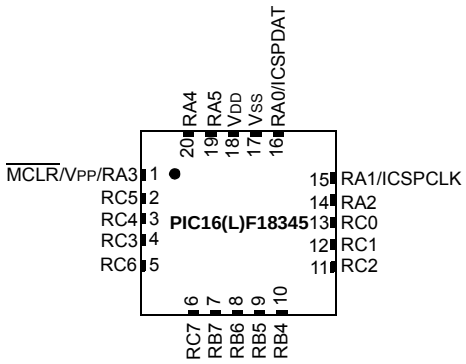
2: It is recommended that the exposed bottom pad be connected to VSS, but must not be the main VSS connection to the device.

FIGURE 3: 20-PIN PDIP, SOIC, SSOP



Note: See Table 2 for location of all peripheral functions.

FIGURE 4: 20-PIN UQFN (4x4)



Note 1: See Table 2 for location of all peripheral functions.

2: It is recommended that the exposed bottom pad be connected to VSS, but must not be the main VSS connection to the device.

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

Name	Function	Input Type	Output Type	Description
OUT(2)	C1	—	CMOS	Comparator C1 output.
	C2	—	CMOS	Comparator C2 output.
	NCO1	—	CMOS	Numerically Controlled Oscillator output.
	DSM	—	CMOS	Digital Signal Modulator output.
	TMR0	—	CMOS	TMR0 clock output.
	CCP1	—	CMOS	Capture/Compare/PWM 1 output.
	CCP2	—	CMOS	Capture/Compare/PWM 2 output.
	CCP3	—	CMOS	Capture/Compare/PWM 3 output.
	CCP4	—	CMOS	Capture/Compare/PWM 4 output.
	PWM5	—	CMOS	Pulse-Width Modulator 5 output.
	PWM6	—	CMOS	Pulse-Width Modulator 6 output.
	CWG1A	—	CMOS	Complementary Waveform Generator 1 output A.
	CWG2A	—	CMOS	Complementary Waveform Generator 2 output A.
	CWG1B	—	CMOS	Complementary Waveform Generator 1 output B.
	CWG2B	—	CMOS	Complementary Waveform Generator 2 output B.
	CWG1C	—	CMOS	Complementary Waveform Generator 1 output C.
	CWG2C	—	CMOS	Complementary Waveform Generator 2 output C.
	CWG1D	—	CMOS	Complementary Waveform Generator 1 output D.
	CWG2D	—	CMOS	Complementary Waveform Generator 2 output D.
	SDA1(3)	I^2C	OD	I^2C data output.
	SDA2(3)	I^2C	OD	I^2C data output.
	SCL1(3)	I^2C	OD	I^2C clock output.
	SCL2(3)	I^2C	OD	I^2C clock output.
	SDO1	—	CMOS	SPI1 data output.
	SD02	—	CMOS	SPI2 data output.
	SCK1	—	CMOS	SPI1 clock output.
	SCK2	—	CMOS	SPI2 clock output.
	TX/CK	—	CMOS	Asynchronous TX data/synchronous clock output.
	DT	—	CMOS	EUSART synchronous data output.
	CLC1OUT	—	CMOS	Configurable Logic Cell 1 source output.
	CLC2OUT	—	CMOS	Configurable Logic Cell 2 source output.
	CLC3OUT	—	CMOS	Configurable Logic Cell 3 source output.
	CLC4OUT	—	CMOS	Configurable Logic Cell 4 source output.
	CLKR	—	CMOS	Clock Reference output.

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

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

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

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

TABLE 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 5													
CPU CORE REGISTERS; see Table 4-2 for specifics													
28Ch	ODCONA			—	—	ODCA5	ODCA4	—	ODCA2	ODCA1	ODCA0	--00 -000	--00 -000
28Dh	ODCONB	X	—	Unimplemented								—	—
		—	X	ODCB7	ODCB6	ODCB5	ODCB4	—	—	—	—	0000 ----	0000 ----
28Eh	ODCONC	X	—	—	—	ODCC5	ODCC4	ODCC3	ODCC2	ODCC1	ODCC0	--00 0000	--00 0000
		—	X	ODCC7	ODCC6	ODCC5	ODCC4	ODCC3	ODCC2	ODCC1	ODCC0	0000 0000	0000 0000
28Fh	—	—		Unimplemented								—	—
290h	—	—		Unimplemented								—	—
291h	CCPR1L			CCPR1<7:0>								xxxx xxxx	xxxx xxxx
292h	CCPR1H			CCPR1<15:8>								xxxx xxxx	xxxx xxxx
293h	CCP1CON			CCP1EN	—	CCP1OUT	CCP1FMT	CCP1MODE<3:0>				0-x0 0000	0-x0 0000
294h	CCP1CAP			—		—	—	CCP1CTS<3:0>				---- 0000	---- xxxx
295h	CCPR2L			CCPR2<7:0>								xxxx xxxx	xxxx xxxx
296h	CCPR2H			CCPR2<15:8>								xxxx xxxx	xxxx xxxx
297h	CCP2CON			CCP2EN	—	CCP2OUT	CCP2FMT	CCP2MODE<3:0>				0-x0 0000	0-x0 0000
298h	CCP2CAP			—	—	—	—	CCP2CTS<3:0>				---- 0000	---- xxxx
299h	—	—		Unimplemented								—	—
29Ah	—	—		Unimplemented								—	—
29Bh	—	—		Unimplemented								—	—
29Ch	—	—		Unimplemented								—	—
29Dh	—	—		Unimplemented								—	—
29Eh	—	—		Unimplemented								—	—
29Fh	CCPTMRS			C4TSEL<1:0>		C3TSEL<1:0>		C2TSEL<1:0>		C1TSEL<1:0>		0101 0101	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.

2: Register accessible from both User and ICD Debugger.

4.4 Stack

All devices have a 16-level x 15-bit wide hardware stack (refer to Figure 4-4 through Figure 4-7). The stack space is not part of either program or data space. The PC is PUSHed onto the stack when CALL or CALLW instructions are executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not affected by a PUSH or POP operation.

The stack operates as a circular buffer and does not cause a Reset when either a Stack Overflow or Underflow occur if the STVREN bit is programmed to '0' (Configuration Words). This means that after the stack has been PUSHed sixteen times, the seventeenth PUSH overwrites the value that was stored from the first PUSH. The eighteenth PUSH overwrites the second PUSH (and so on). The STKOVF and STKUNF flag bits will be set on an Overflow/Underflow, regardless of whether the Reset is enabled.

If the STVREN bit in Configuration Words is programmed to '1', the device will be Reset if the stack is PUSHed beyond the sixteenth level or POPed beyond the first level, setting the appropriate bits (STKOVF or STKUNF, respectively) in the PCON register.

Note 1:	There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, CALLW, RETURN, RETLW and RETFIE instructions or the vectoring to an interrupt address.

4.4.1 ACCESSING THE STACK

The stack is accessible through the TOSH, TOSL and STKPTR registers. STKPTR is the current value of the Stack Pointer. TOSH:TOSL register pair points to the TOP of the stack. Both registers are read/writable. TOS is split into TOSH and TOSL due to the 15-bit size of the PC. To access the stack, adjust the value of STKPTR, which will position TOSH:TOSL, then read/write to TOSH:TOSL. STKPTR is five bits to allow detection of overflow and underflow.

Note:	Care should be taken when modifying the STKPTR while interrupts are enabled.

During normal program operation, CALL, CALLW and Interrupts will increment STKPTR while RETLW, RETURN, and RETFIE will decrement STKPTR. At any time, STKPTR can be read to see how many levels remain available on the stack. The STKPTR always points at the currently used place on the stack. Therefore, a CALL or CALLW will increment the STKPTR and then write the PC, and a return will write the PC and then decrement the STKPTR.

Reference Figure 4-4 through Figure 4-7 for examples of accessing the stack.

TABLE 7-1: NOSC/COSC BIT SETTINGS

NOSC<2:0> COSC<2:0>	Clock Source
111	EXTOSC(1)
110	HFINTOSC (1 MHz)
101	Reserved
100	LFINTOSC
011	SOSC
010	Reserved
001	EXTOSC with 4xPLL(1)
000	HFINTOSC with 2x PLL (32 MHz)

Note 1: EXTOSC configured by the FEXTOSC bits of Configuration Word 1 (Register 5-1).

TABLE 7-2: NDIV/CDIV BIT SETTINGS

NDIV<3:0> CDIV<3:0>	Clock Divider
1111–1010	Reserved
1001	512
1000	256
0111	128
0110	64
0101	32
0100	16
0011	8
0010	4
0001	2
0000	1

REGISTER 7-3: OSCCON3: OSCILLATOR CONTROL REGISTER 3

R/W/HC-0/0	R/W-0/0	R/W-0/0	R-0/0	R-0/0	U-0	U-0	U-0
CSWHOLD	SOSCPWR	SOSCBE	ORDY	NOSCR	—	—	—
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	f = determined by fuse setting
q = Reset value is determined by hardware		

bit 7 **CSWHOLD:** Clock Switch Hold bit
- 1 = Clock switch will hold (with interrupt) when the oscillator selected by NOSC is ready
- 0 = Clock switch may proceed when the oscillator selected by NOSC is ready; if this bit is set at the time that NOSCR becomes '1', the switch and interrupt will occur.

bit 6 **SOSCPWR:** Secondary Oscillator Power Mode Select bit

If SOSCBE = 0
- 1 = Secondary oscillator operating in High-Power mode
- 0 = Secondary oscillator operating in Low-Power mode

If SOSCBE = 1
- x = Bit is ignored

bit 5 **SOSCBE:** Secondary Oscillator Bypass Enable bit
- 1 = Secondary oscillator SOSCI is configured as an external clock input (ST-buffer); SOSCO is not used.
- 0 = Secondary oscillator is configured as a crystal oscillator using SOSCO and SOSCI pins.

bit 4 **ORDY:** Oscillator Ready bit (read-only)
- 1 = OSCCON1 = OSCCON2; the current system clock is the clock specified by NOSC
- 0 = A clock switch is in progress

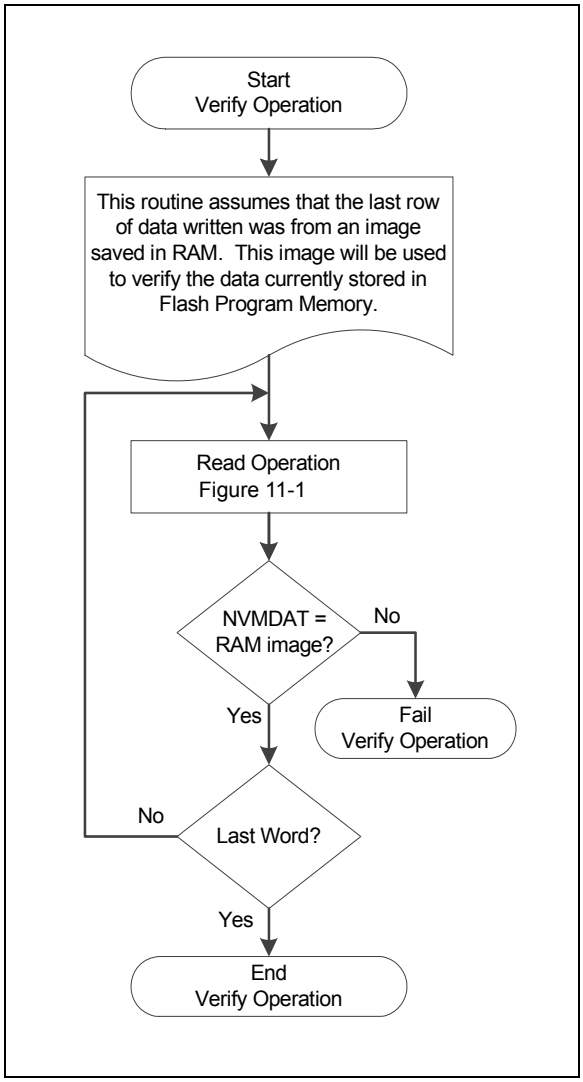
bit 3 **NOSCR:** New Oscillator is Ready bit (read-only)
- 1 = A clock switch is in progress and the oscillator selected by NOSC indicates a Ready condition
- 0 = A clock switch is not in progress, or the NOSC-selected oscillator is not yet ready

bit 2-0 **Unimplemented:** Read as '0'.

11.4.8 WRITE VERIFY

It is considered good programming practice to verify that program memory writes agree with the intended value. Since program memory is stored as a full row, then the stored program memory contents are compared with the intended data stored in RAM after the last write is complete.

FIGURE 11-7: PROGRAM FLASH MEMORY VERIFY FLOWCHART



REGISTER 12-20: ANSELC: PORTC ANALOG SELECT 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
ANSC7(1)	ANSC6(1)	ANSC5	ANSC4	ANSC3	ANSC2	ANSC1	ANSC0
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 **ANSC<7:6>**: Analog Select between Analog or Digital Function on pins RC<7:6>, respectively(1)
0 = Digital I/O. Pin is assigned to port or digital special function.
1 = Analog input. Pin is assigned as analog input(2). Digital input buffer disabled.

bit 5-0 **ANSC<5:0>**: Analog Select between Analog or Digital Function on pins RC<5:0>, respectively
0 = Digital I/O. Pin is assigned to port or digital special function.
1 = Analog input. Pin is assigned as analog input(2). Digital input buffer disabled.

Note 1: PIC16(L)F18345 only; otherwise read as '0'.
2: When setting a pin to an analog input, the corresponding TRIS bit must be set to Input mode in order to allow external control of the voltage on the pin.

REGISTER 12-21: WPUC: WEAK PULL-UP PORTC 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
WPUC7(1)	WPUC6(1)	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0
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 **WPUC<7:6>**(1): Weak Pull-up Register bits(2)
1 = Pull-up enabled
0 = Pull-up disabled

bit 5-0 **WPUC<5:0>**: Weak Pull-up Register bits(2)
1 = Pull-up enabled
0 = Pull-up disabled

Note 1: PIC16(L)F18345 only; otherwise read as '0'.
2: The weak pull-up is disabled if the pin is configured as an output except when the pin is also configured as open-drain. When configured as open-drain, the pull-up is enabled when the output value is high, and disabled when the output value is low.

REGISTER 15-5: IOCBN: INTERRUPT-ON-CHANGE PORTB NEGATIVE EDGE REGISTER(1)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	U-0	U-0
IOCBN7	IOCBN6	IOCBN5	IOCBN4	—	—	—	—
bit 7							bit 0

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

bit 7-4 **IOCBN<7:4>:** Interrupt-on-Change PORTB Negative Edge Enable bits
1 = Interrupt-on-Change enabled on the pin for a negative going edge. IOCAFx bit and IOCIF flag will be set upon detecting an edge.
0 = Interrupt-on-Change disabled for the associated pin

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

Note 1: PIC16(L)F18345 only.

REGISTER 15-6: IOCBF: INTERRUPT-ON-CHANGE PORTB FLAG REGISTER(1)

R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	U-0	U-0	U-0	U-0
IOCBF7	IOCBF6	IOCBF5	IOCBF4	—	—	—	—
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 - Bit is set in hardware

bit 7-4 **IOCBF<7:4>:** Interrupt-on-Change PORTB Flag bits
1 = An enabled change was detected on the associated pin
Set when IOCBPx = 1 and a rising edge was detected on RBx, or when IOCBNx = 1 and a falling edge was detected on RBx.
0 = No change was detected, or the user cleared the detected change.

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

Note 1: PIC16(L)F18345 only.

19.1.3 PWM RESOLUTION

PWM resolution, expressed in number of bits, defines the maximum number of discrete steps that can be present in a single PWM period. For example, a 10-bit resolution will result in 1024 discrete steps, whereas an 8-bit resolution will result in 256 discrete steps.

The maximum PWM resolution is ten bits when PR2 is 255. The resolution is a function of the PR2 register value as shown by Equation 19-4.

EQUATION 19-4:

$$Resolution = \frac{\log[4(PR2 + 1)]}{\log(2)} \ bits$$

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

19.1.4 OPERATION IN SLEEP MODE

In Sleep mode, the TMR2 register will not increment and the state of the module will not change. If the PWMx pin is driving a value, it will continue to drive that value. When the device wakes up, TMR2 will continue from its previous state.

19.1.5 CHANGES IN SYSTEM CLOCK FREQUENCY

The PWM frequency is derived from the system clock frequency. Any changes in the system clock frequency will result in changes to the PWM frequency. See **Section 7.0, Oscillator Module** for additional details.

19.1.6 EFFECTS OF RESET

Any Reset will force all ports to Input mode and the PWMx registers to their Reset states.

19.1.7 SETUP FOR PWM OPERATION

The following steps should be taken when configuring the module for using the PWMx outputs:

1. Disable the PWMx pin output driver(s) by setting the associated TRIS bit(s).
2. Configure the PWM output polarity by configuring the PWMxPOL bit of the PWMxCON register.
3. Load the PR2 register with the PWM period value, as determined by Equation 19-1.
4. Load the PWMxDCH register and bits <7:6> of the PWMxDCL register with the PWM duty cycle value, as determined by Equation 19-2.
5. Configure and start Timer2:
   - Clear the TMR2IF interrupt flag bit of the PIR1 register.
   - Select the Timer2 prescale value by configuring the T2CKPS bit of the T2CON register.
   - Enable Timer2 by setting the TMR2ON bit of the T2CON register.
6. Wait until the TMR2IF is set.
7. When the TMR2IF flag bit is set:
   - Clear the associated TRIS bit(s) to enable the output driver.
   - Route the signal to the desired pin by configuring the RxyPPS register.
   - Enable the PWMx module by setting the PWMxEN bit of the PWMxCON register.

In order to send a complete duty cycle and period on the first PWM output, the above steps must be followed in the order given. If it is not critical to start with a complete PWM signal, then the PWM module can be enabled during Step 2 by setting the PWMxEN bit of the PWMxCON register.

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.

27.1 Timer1 Operation

The Timer1 module is a 16-bit incrementing timer which is accessed through the TMR1H:TMR1L register pair. Writes to TMR1H or TMR1L directly update the timer.

The module can be used with either internal or external clock sources, and has the Timer1 Gate Enable function. When Timer1 is used with the Timer1 Gate Enable, the timer can measure time intervals or count signal pulses between two points of interest. When used without the Timer1 Gate Enable, the timer simply measures time intervals.

Timer1 is enabled by configuring the TMR1ON and TMR1GE bits in the T1CON and T1GCON registers, respectively. Table 27-1 displays the Timer1 enable selections.

TABLE 27-1: TIMER1 ENABLE SELECTIONS

TMR1ON	TMR1GE	Timer1 Operation
0	0	Off
0	1	Off
1	0	Always On
1	1	Count Enabled

27.2 Clock Source Selection

The TMR1CS<1:0> and T1OSC bits of the T1CON register are used to select the clock source for Timer1. Table 27-2 displays the clock source selections. The TMR1H:TMR1L register pair will increment on multiples of the clock source as determined by the Timer1 prescaler.

When either the FOSC or LFINTOSC clock source is selected, the TMR1H:TMR1L register pair will increment every rising clock edge. Reading from the TMR1H:TMR1L register pair when either the FOSC or LFINTOSC is the clock source will cause a 2 LSb loss in resolution, which can be mitigated by using an asynchronous input signal to gate the Timer1 clock input (see **Section 26.5 “Operation During Sleep”** for more information on the Timer1 Gate Enable).

When the FOSC/4 clock source is selected, the TMR1H:TMR1L register pair increments every instruction cycle (once every four FOSC pulses).

In addition to the internal clock sources, Timer1 has a dedicated external clock input pin, T1CKI. T1CKI can be either synchronized to the system clock or can run asynchronously via the T1SYNC bit of the T1CON register. When the T1CKI pin is used as the clock source, the TMR1H:TMR1L register pair increments on the rising edge of the T1CKI clock input.

Note: When using Timer1 to count events, a falling edge must be registered by the counter prior to the first incrementing rising edge after any one or more of the following conditions:

- Timer1 enabled after POR
- Write to TMR1H or TMR1L
- Timer1 is disabled
- Timer1 is disabled (TMR1ON = 0) when T1CKI is high then Timer1 is enabled (TMR1ON=1) when T1CKI is low.

TABLE 27-2: CLOCK SOURCE SELECTIONS

TMR1CS<1:0>	Clock Source
11	LFINTOSC
10	External Clocking on T1CKI Pin
01	System Clock (FOSC)
00	Instruction Clock (FOSC/4)

27.10 Register Definitions: Timer1/3/5 Control

REGISTER 27-1: TxCON(1): TIMERx CONTROL REGISTER

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	U-0	R/W-0/u
TMRxCS<1:0>		TxCKPS<1:0>		TxSOSC	$\overline{\text{TxSYNC}}$	—	TMRxON
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 **TMRxCS<1:0>:** Timerx Clock Source Select bits
- 11 = Timerx clock Source is LFINTOSC
- 10 = Timerx clock source is pin or oscillator:
  - If TxSOSC = 0:
  - External clock from TxCKIPPS pin (on the rising edge)
  - If TxSOSC = 1:
  - Clock from SOSC, either crystal oscillator on TxSOSCI/TxSOSCO pins, or SOSCIN input
- 01 = Timerx clock source is system clock (FOSC)
- 00 = Timerx clock source is instruction clock (FOSC/4)

bit 5-4 **TxCKPS<1:0>:** Timerx Input Clock Prescale Select bits
- 11 = 1:8 Prescale value
- 10 = 1:4 Prescale value
- 01 = 1:2 Prescale value
- 00 = 1:1 Prescale value

bit 3 **TxSOSC:** LP Oscillator Enable Control bit
- 1 = SOSC requested as the clock source
- 0 = TxCKI enabled as the clock source

bit 2 **$\overline{\text{TxSYNC}}$:** Timer1 Synchronization Control bit

TMRxCS<1:0> = 1x
- 1 = Do not synchronize external clock input
- 0 = Synchronize external clock input with system clock

TMRxCS<1:0> = 0x

This bit is ignored. Timer1 uses the internal clock and no additional synchronization is performed.

bit 1 **Unimplemented:** Read as '0'

bit 0 **TMRxON:** Timer1 On bit
- 1 = Enables Timerx
- 0 = Stops Timerx and clears Timerx gate flip-flop

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

29.5 Register Definitions: CCP Control

REGISTER 29-1: CCPxCON: CCPx CONTROL REGISTER

R/W-0/0	U-0	R-x/x	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
CCPxEN	—	CCPxOUT	CCPxFMT	CCPxMODE<3:0>			
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 Reset
'1' = Bit is set	'0' = Bit is cleared	

bit 7 **CCPxEN:** CCP Module Enable bit
0 = CCP is disabled
1 = CCP is enabled

bit 6 **Unimplemented:** Read as '0'

bit 5 **CCPxOUT:** CCPx Output Data (read-only) bit

bit 4 **CCPxFMT:** CCPW (pulse width) Alignment bit
CCPxMODE = Capture mode
Unused
CCPxMODE = Compare mode
Unused
CCPxMODE = PWM mode
0 = Right-aligned format
1 = Left-aligned format

bit 3-0 **CCPxMODE<3:0>:** CCPx Mode Select bits(1)
1111 = PWM mode
1110 = Reserved
1101 = Reserved
1100 = Reserved

1011 = Compare mode: output will pulse 0-1-0; Clears TMR1/3/5
1010 = Compare mode: output will pulse 0-1-0
1001 = Compare mode: clear output on compare match
1000 = Compare mode: set output on compare match

0111 = Capture mode: every 16th rising edge of CCPx input
0110 = Capture mode: every 4th rising edge of CCPx input
0101 = Capture mode: every rising edge of CCPx input
0100 = Capture mode: every falling edge of CCPx input

0011 = Capture mode: every edge of CCPx input
0010 = Compare mode: toggle output on match
0001 = Compare mode: toggle output on match; clear TMR1/3/5
0000 = Capture/Compare/PWM off (resets CCPx module)

Note 1: All modes will set the CCPxIF bit and will trigger an ADC conversion if CCPx is selected as the ADC trigger source.

30.5.3 SLAVE TRANSMISSION

When the R/$\overline{W}$ bit of the incoming address byte is set and an address match occurs, the R/$\overline{W}$ bit of the SSPxSTAT register is set. The received address is loaded into the SSPxBUF register, and an $\overline{ACK}$ pulse is sent by the slave on the ninth bit.

Following the $\overline{ACK}$, slave hardware clears the CKP bit and the SCL pin is held low (see **Section 30.5.6 "Clock Stretching"** for more detail). By stretching the clock, the master will be unable to assert another clock pulse until the slave is done preparing the transmit data.

The transmit data must be loaded into the SSPxBUF register which also loads the SSPxSR register. Then the SCL pin should be released by setting the CKP bit of the SSPxCON1 register. The eight data bits are shifted out on the falling edge of the SCL input. This ensures that the SDA signal is valid during the SCL high time.

The $\overline{ACK}$ pulse from the master-receiver is latched on the rising edge of the ninth SCL input pulse. This $\overline{ACK}$ value is copied to the ACKSTAT bit of the SSPxCON2 register. If ACKSTAT is set (not $\overline{ACK}$), then the data transfer is complete. When the not $\overline{ACK}$ is latched by the slave, the slave goes idle and waits for another occurrence of the Start bit. If the SDA line was low ($\overline{ACK}$), the next transmit data must be loaded into the SSPxBUF register. Again, the SCL pin must be released by setting bit CKP.

An MSSP interrupt is generated for each data transfer byte. The SSPxIF bit must be cleared by software and the SSPxSTAT register is used to determine the status of the byte. The SSPxIF bit is set on the falling edge of the ninth clock pulse.

30.5.3.1 Slave Mode Bus Collision

A slave receives a Read request and begins shifting data out on the SDA line. If a bus collision is detected and the SBCDE bit of the SSPxCON3 register is set, the BCLIF bit of the PIR register is set. Once a bus collision is detected, the slave goes idle and waits to be addressed again. User software can use the BCLIF bit to handle a slave bus collision.

30.5.3.2 7-bit Transmission

A master device can transmit a read request to a slave, and then clock data out of the slave. The list below outlines what software for a slave will need to do to accomplish a standard transmission. Figure 30-18 can be used as a reference to this list.

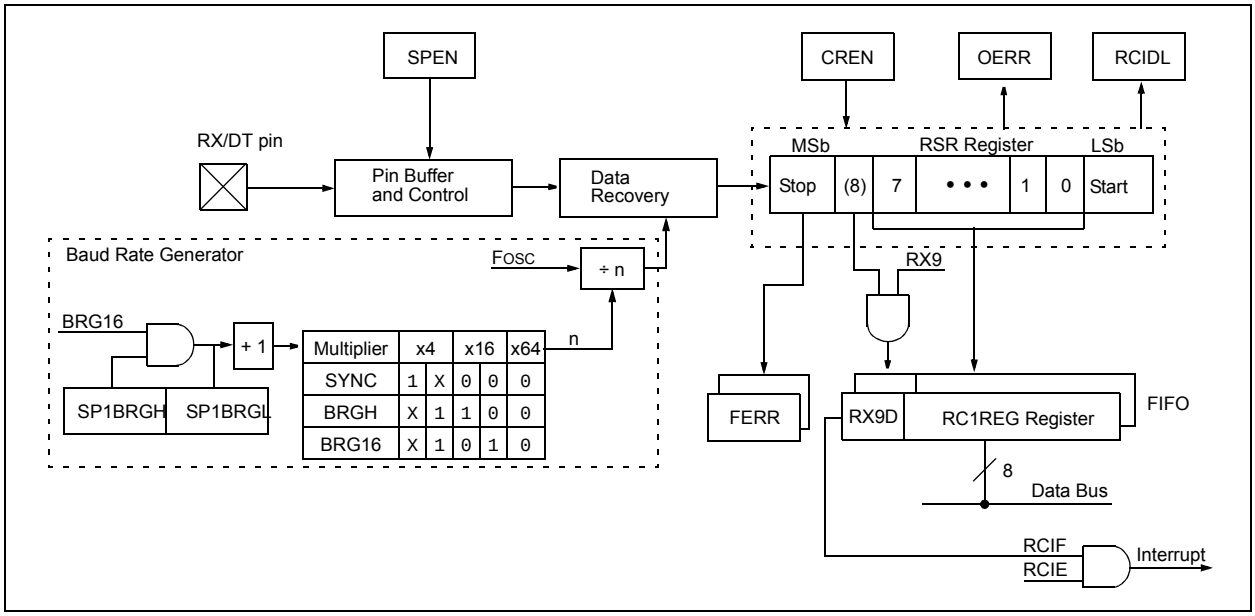
1. Master sends a Start condition on SDA and SCL.
2. S bit of SSPxSTAT is set; SSPxIF is set if interrupt on Start detect is enabled.
3. Matching address with R/$\overline{W}$ bit set is received by the Slave setting SSPxIF bit.
4. Slave hardware generates an $\overline{ACK}$ and sets SSPxIF.
5. SSPxIF bit is cleared by user.
6. Software reads the received address from SSPxBUF, clearing BF.
7. R/$\overline{W}$ is set so CKP was automatically cleared after the $\overline{ACK}$.
8. The slave software loads the transmit data into SSPxBUF.
9. CKP bit is set releasing SCL, allowing the master to clock the data out of the slave.
10. SSPxIF is set after the $\overline{ACK}$ response from the master is loaded into the ACKSTAT register.
11. SSPxIF bit is cleared.
12. The slave software checks the ACKSTAT bit to see if the master wants to clock out more data.

Note 1: If the master $\overline{ACK}$s the clock will be stretched.

2: ACKSTAT is the only bit updated on the rising edge of SCL (9th) rather than the falling.

13. Steps 9-13 are repeated for each transmitted byte.
14. If the master sends a not $\overline{ACK}$; the clock is not held, but SSPxIF is still set.
15. The master sends a Restart condition or a Stop.
16. The slave is no longer addressed.

FIGURE 31-2: EUSART1 RECEIVE BLOCK DIAGRAM



The operation of the EUSART1 module is controlled through three registers:

- Transmit Status and Control (TX1STA)
- Receive Status and Control (RC1STA)
- Baud Rate Control (BAUD1CON)

These registers are detailed in Register 31-1, Register 31-2 and Register 31-3, respectively.

The RX and CK input pins are selected with the RXPPS and CKPPS registers, respectively. TX, CK, and DT output pins are selected with each pin's RxyPPS register. Since the RX input is coupled with the DT output in Synchronous mode, it is the user's responsibility to select the same pin for both of these functions when operating in Synchronous mode. The EUSART1 control logic will control the data direction drivers automatically.

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.

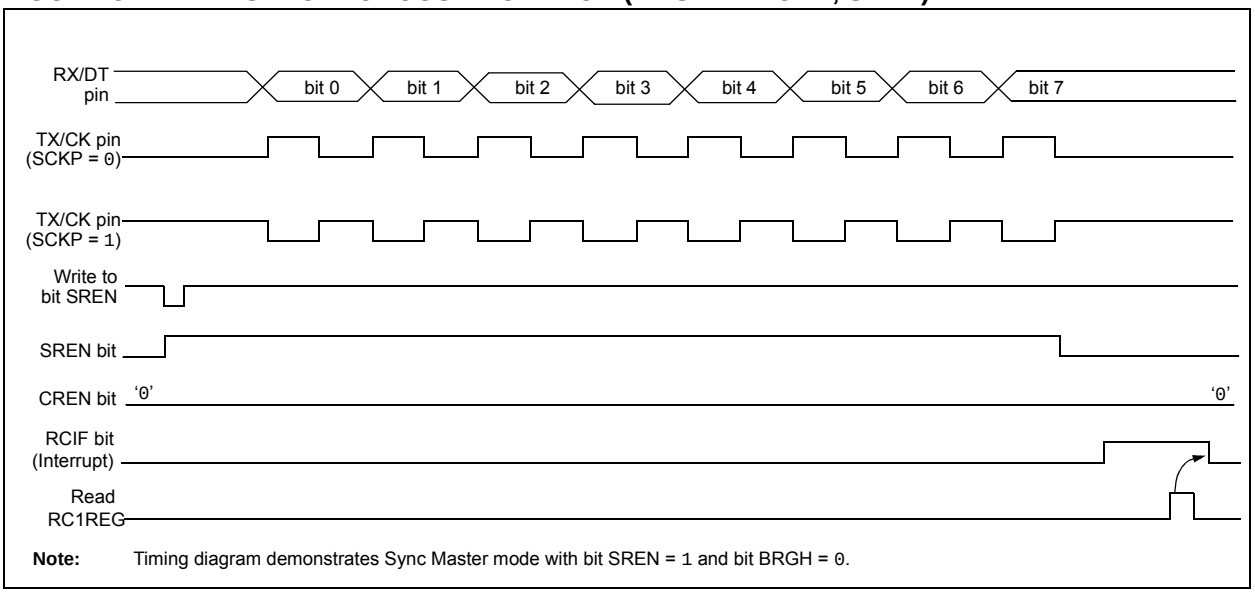
31.4.1.8 Receiving 9-bit Characters

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)



Note: Timing diagram demonstrates Sync Master mode with bit SREN = 1 and bit BRGH = 0.

TABLE 31-4: BAUD RATE FOR ASYNCHRONOUS MODES (CONTINUED)

BAUD RATE	SYNC = 0, BRGH = 0, BRG16 = 1											
	Fosc = 8.000 MHz			Fosc = 4.000 MHz			Fosc = 3.6864 MHz			Fosc = 1.000 MHz		
	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	299.9	-0.02	1666	300.1	0.04	832	300.0	0.00	767	300.5	0.16	207
1200	1199	-0.08	416	1202	0.16	207	1200	0.00	191	1202	0.16	51
2400	2404	0.16	207	2404	0.16	103	2400	0.00	95	2404	0.16	25
9600	9615	0.16	51	9615	0.16	25	9600	0.00	23	—	—	—
10417	10417	0.00	47	10417	0.00	23	10473	0.53	21	10417	0.00	5
19.2k	19.23k	0.16	25	19.23k	0.16	12	19.20k	0.00	11	—	—	—
57.6k	55556	-3.55	8	—	—	—	57.60k	0.00	3	—	—	—
115.2k	—	—	—	—	—	—	115.2k	0.00	1	—	—	—

BAUD RATE	SYNC = 0, BRGH = 1, BRG16 = 1 or SYNC = 1, BRG16 = 1											
	Fosc = 32.000 MHz			Fosc = 20.000 MHz			Fosc = 18.432 MHz			Fosc = 11.0592 MHz		
	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	300.0	0.00	26666	300.0	0.00	16665	300.0	0.00	15359	300.0	0.00	9215
1200	1200	0.00	6666	1200	-0.01	4166	1200	0.00	3839	1200	0.00	2303
2400	2400	0.01	3332	2400	0.02	2082	2400	0.00	1919	2400	0.00	1151
9600	9604	0.04	832	9597	-0.03	520	9600	0.00	479	9600	0.00	287
10417	10417	0.00	767	10417	0.00	479	10425	0.08	441	10433	0.16	264
19.2k	19.18k	-0.08	416	19.23k	0.16	259	19.20k	0.00	239	19.20k	0.00	143
57.6k	57.55k	-0.08	138	57.47k	-0.22	86	57.60k	0.00	79	57.60k	0.00	47
115.2k	115.9k	0.64	68	116.3k	0.94	42	115.2k	0.00	39	115.2k	0.00	23

BAUD RATE	SYNC = 0, BRGH = 1, BRG16 = 1 or SYNC = 1, BRG16 = 1											
	Fosc = 8.000 MHz			Fosc = 4.000 MHz			Fosc = 3.6864 MHz			Fosc = 1.000 MHz		
	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)	Actual Rate	% Error	SPBRG value (decimal)
300	300.0	0.00	6666	300.0	0.01	3332	300.0	0.00	3071	300.1	0.04	832
1200	1200	-0.02	1666	1200	0.04	832	1200	0.00	767	1202	0.16	207
2400	2401	0.04	832	2398	0.08	416	2400	0.00	383	2404	0.16	103
9600	9615	0.16	207	9615	0.16	103	9600	0.00	95	9615	0.16	25
10417	10417	0	191	10417	0.00	95	10473	0.53	87	10417	0.00	23
19.2k	19.23k	0.16	103	19.23k	0.16	51	19.20k	0.00	47	19.23k	0.16	12
57.6k	57.14k	-0.79	34	58.82k	2.12	16	57.60k	0.00	15	—	—	—
115.2k	117.6k	2.12	16	111.1k	-3.55	8	115.2k	0.00	7	—	—	—

TABLE 35-12: ANALOG-TO-DIGITAL CONVERTER (ADC) CHARACTERISTICS(1,2)

Standard Operating Conditions (unless otherwise stated) VDD = 3.0V, TA = 25°C							
Param. No.	**Sym.**	**Characteristic**	**Min.**	**Typ.†**	**Max.**	**Units**	**Conditions**
AD01	NR	Resolution	—	—	10	bit	
AD02	EIL	Integral Error	—	±0.1	±1.0	LSb	ADCREF+ = 3.0V, ADCREF- = 0V
AD03	EDL	Differential Error	—	±0.1	±1.0	LSb	ADCREF+ = 3.0V, ADCREF- = 0V
AD04	EOFF	Offset Error	—	0.5	2	LSb	ADCREF+ = 3.0V, ADCREF- = 0V
AD05	EGN	Gain Error	—	±0.2	±1.0	LSb	ADCREF+ = 3.0V, ADCREF- = 0V
AD06	VADREF	ADC Reference Voltage (ADREF+)(3)	1.8	—	VDD	V	
AD07	VAIN	Full-Scale Range	VSS	—	ADREF+	V	
AD06	VADREF	ADC Reference Voltage (ADREF+ - ADREF-)(3)	1.8	—	VDD	V	
AD07	VAIN	Full-Scale Range	ADREF-	—	ADREF+	V	
AD08	ZAIN	Recommended Impedance of Analog Voltage Source	—	10	—	kΩ	
AD09	RVREF	ADC Voltage Reference Ladder Impedance	—	—	—	kΩ	

* These parameters are characterized but not tested.

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

Note 1: Total Absolute Error is the sum of the offset, gain and integral non-linearity (INL) errors.

2: The ADC conversion result never decreases with an increase in the input and has no missing codes.

TABLE 35-14: COMPARATOR SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated) V_{DD} = 3.0V, T_A = 25°C See **Section 36.0 "DC and AC Characteristics Graphs and Charts"** for operating characterization.							
Param No.	**Sym.**	**Characteristics**	**Min.**	**Typ.**	**Max.**	**Units**	**Comments**
CM01	V_{IOFF}	Input Offset Voltage	—	—	±40	mV	V_{ICM} = V_{DD}/2
CM02	V_{ICM}	Input Common Mode Voltage	GND	—	V_{DD}	V	
CM03	CMRR	Common Mode Input Rejection Ratio	—	50	—	dB	
CM04	C_{HYST}	Comparator Hysteresis	15	25	35	mV	
CM05	T_{RESP}(1)	Response Time, Rising Edge	—	300	600	ns	
		Response Time, Falling Edge	—	220	500	ns	
CM06*	T_{MCV2VO}(2)	Mode Change to Valid Output	—	—	10	us	

* These parameters are characterized but not tested.

Note 1: Response time measured with one comparator input at V_{DD}/2, while the other input transitions from V_{SS} to V_{DD}.

2: A mode change includes changing any of the control register values, including module enable.

TABLE 35-15: DIGITAL-TO-ANALOG CONVERTER (DAC) SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated) V_{DD} = 3.0V, T_A = 25°C							
Param No.	**Sym.**	**Characteristics**	**Min.**	**Typ.†**	**Max.**	**Units**	**Comments**
DSB01	V_{LSB}	Step Size	—	V_{DD}/32	—	V	
DSB01	V_{ACC}	Absolute Accuracy	—	—	± 0.5	LSb	
DSB03*	R_{UNIT}	Unit Resistor Value	—	6000	—	Ω	
DSB04*	T_{ST}	Settling Time(1)	—	—	10	µs	

* These parameters are characterized but not tested.

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

Note 1: Settling time measured while DACR<4:0> transitions from '00000' to '01111'.

TABLE 35-16: FIXED VOLTAGE REFERENCE (FVR) SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated)							
Param. No.	**Symbol**	**Characteristic**	**Min.**	**Typ.**	**Max.**	**Units**	**Conditions**
FVR01	V_{FVR1}	1x Gain (1.024V nominal)	-4	—	4	%	V_{DD} ≥ 2.5V, -40°C to 85°C
FVR02	V_{FVR2}	2x Gain (2.048V nominal)	-4	—	4	%	V_{DD} ≥ 2.5V, -40°C to 85°C
FVR03	V_{FVR4}	4x Gain (4.096V nominal)	-5	—	5	%	V_{DD} ≥ 4.75V, -40°C to 85°C
FVR04	T_{FVRST}	FVR Start-up Time	—	—	—	µs	

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

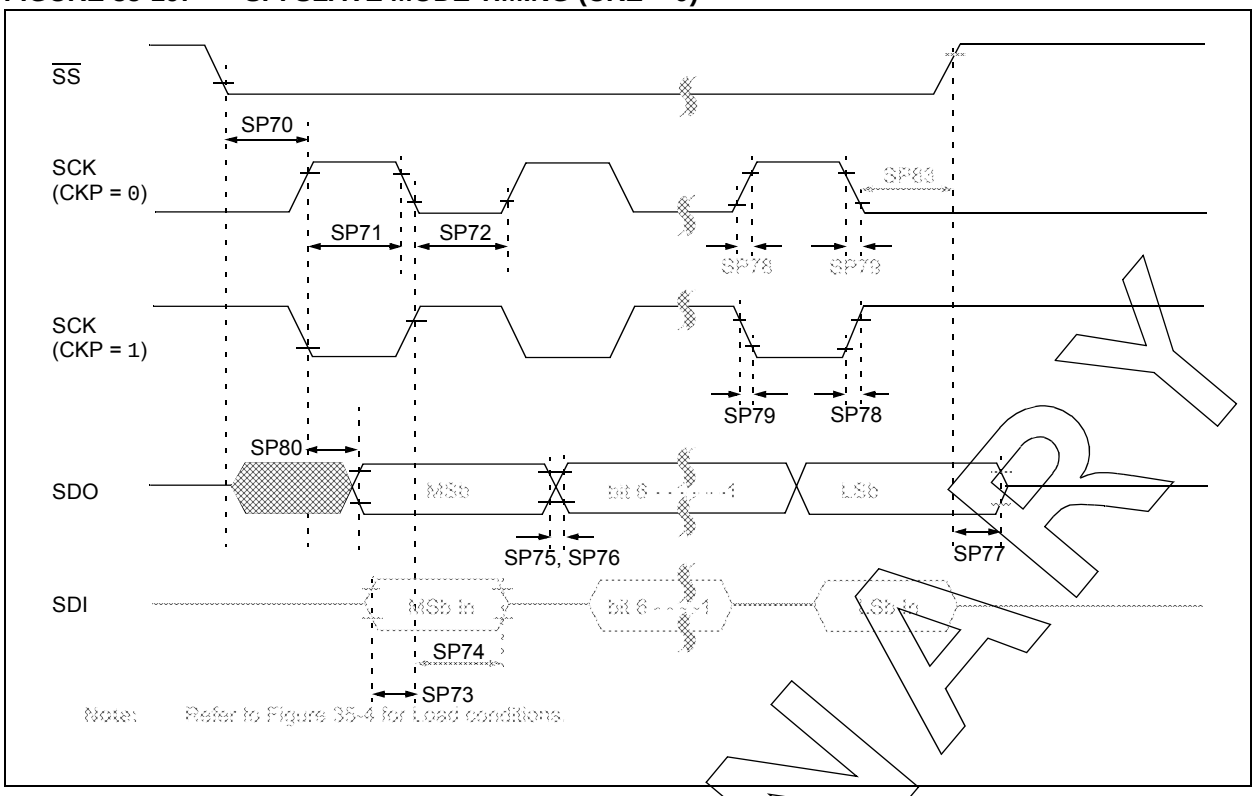


FIGURE 35-20: SPI SLAVE MODE TIMING (CKE = 1)