

Welcome to [E-XFL.COM](https://www.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, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	16
Program Memory Size	7KB (4K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	384 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 12x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	18-SOIC (0.295", 7.50mm Width)
Supplier Device Package	18-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f1827-i-so

PIC16(L)F1826/27

TABLE 3-4: PIC16(L)F1826/27 MEMORY MAP (CONTINUED)

Bank 31	
F80h	Core Registers (Table 3-2)
F8Bh F8Ch	
FE3h	Unimplemented Read as '0'
FE4h	
FE5h	
FE6h	
FE7h	
FE8h	
FE9h	
FEAh	
FEBh	
FECb	
FEDh	STKPTR
FEEh	TOSL
FEFh	TOSH
FF0h	Common RAM (Accesses 70h – 7Fh)
FFFh	

■ = Unimplemented data memory locations, read as '0',

PIC16(L)F1826/27

FIGURE 3-6: ACCESSING THE STACK EXAMPLE 2

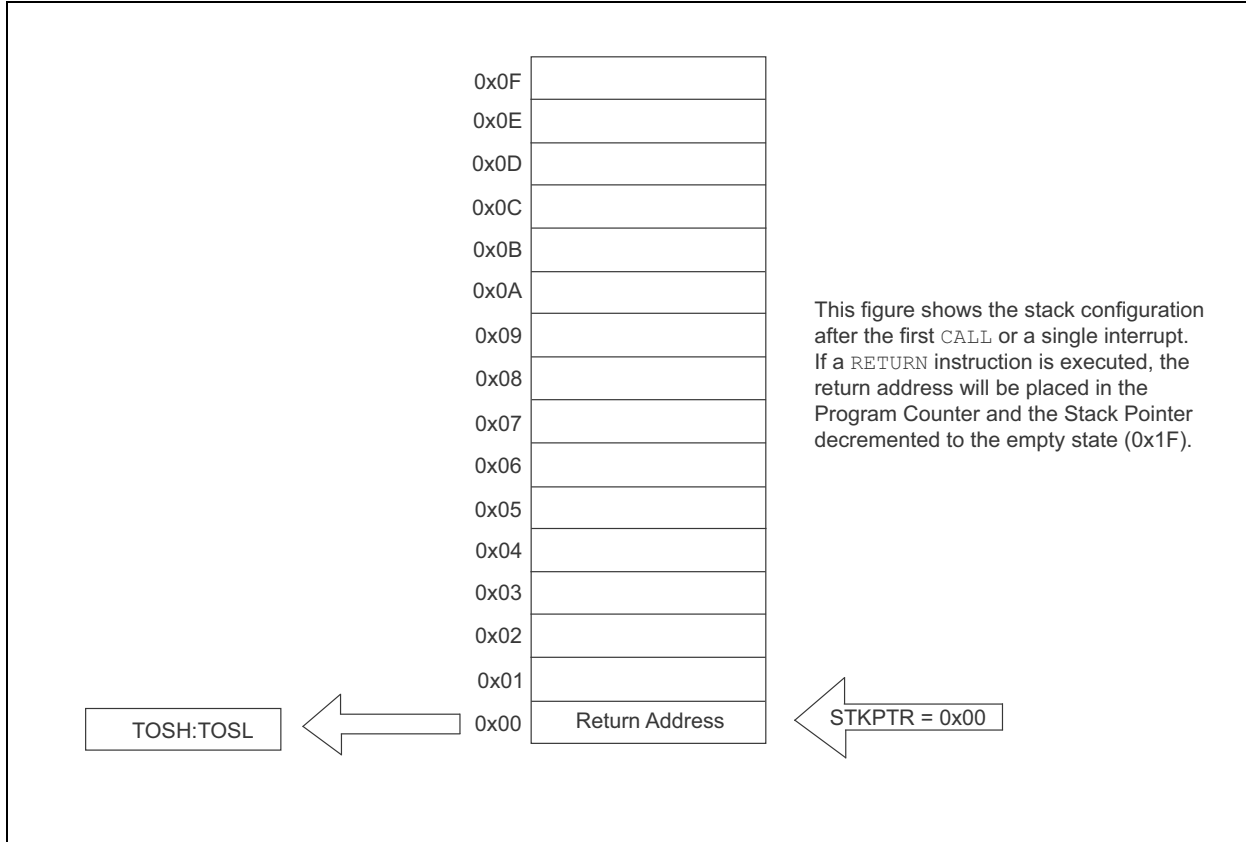
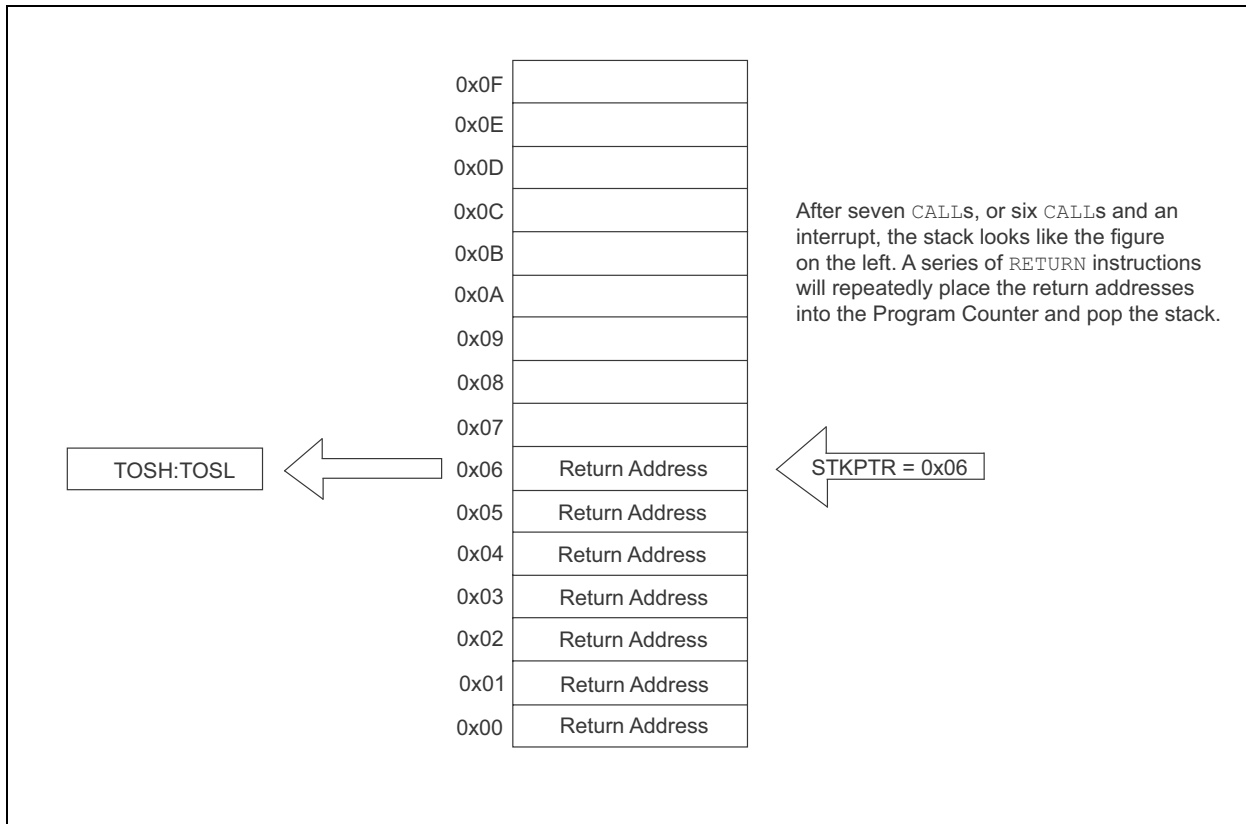


FIGURE 3-7: ACCESSING THE STACK EXAMPLE 3



7.0 RESETS

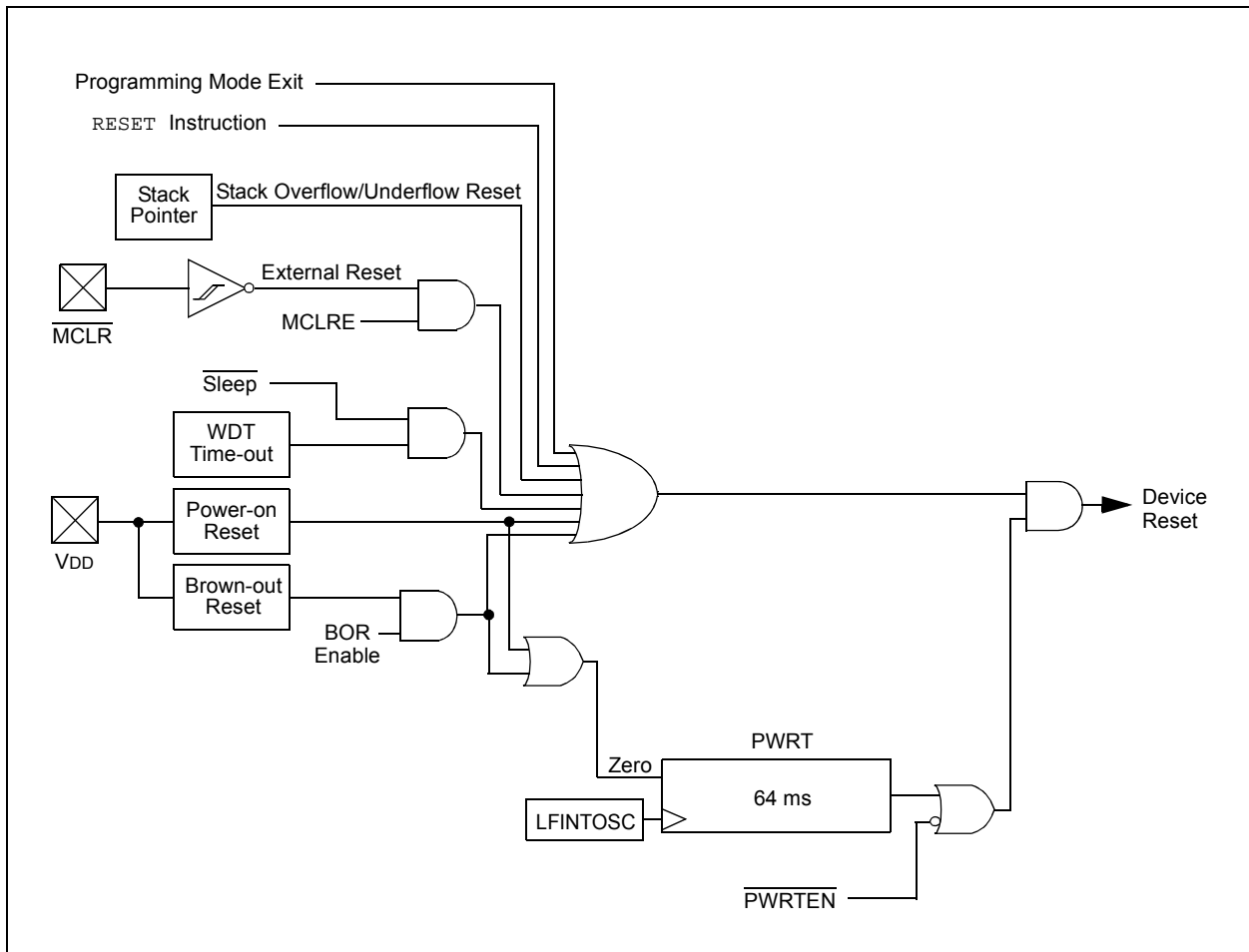
There are multiple ways to reset this device:

- Power-on Reset (POR)
- Brown-out Reset (BOR)
- MCLR Reset
- WDT Reset
- RESET instruction
- Stack Overflow
- Stack Underflow
- Programming mode exit

To allow VDD to stabilize, an optional power-up timer can be enabled to extend the Reset time after a BOR or POR event.

A simplified block diagram of the On-Chip Reset Circuit is shown in Figure 7-1.

FIGURE 7-1: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



7.11 Power Control (PCON) Register

The Power Control (PCON) register contains flag bits to differentiate between a:

- Power-on Reset ($\overline{\text{POR}}$)
- Brown-out Reset ($\overline{\text{BOR}}$)
- Reset Instruction Reset ($\overline{\text{RI}}$)
- MCLR Reset ($\overline{\text{RMCLR}}$)
- Stack Underflow Reset (STKUNF)
- Stack Overflow Reset (STKOVF)

The PCON register bits are shown in Register 7-2.

REGISTER 7-2: PCON: POWER CONTROL REGISTER

R/W/HS-0/q	R/W/HS-0/q	U-0	U-0	R/W/HC-1/q	R/W/HC-1/q	R/W/HC-q/u	R/W/HC-q/u
STKOVF	STKUNF	—	—	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$
bit 7							bit 0

Legend:

HC = Bit is cleared by hardware

HS = Bit is set by hardware

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

u = Bit is unchanged

x = Bit is unknown

-m/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 **STKOVF:** Stack Overflow Flag bit
 1 = A Stack Overflow occurred
 0 = A Stack Overflow has not occurred or set to '0' by firmware
- bit 6 **STKUNF:** Stack Underflow Flag bit
 1 = A Stack Underflow occurred
 0 = A Stack Underflow has not occurred or set to '0' by firmware
- bit 5-4 **Unimplemented:** Read as '0'
- bit 3 **$\overline{\text{RMCLR}}$:** MCLR Reset Flag bit
 1 = A $\overline{\text{MCLR}}$ Reset has not occurred or set to '1' by firmware
 0 = A $\overline{\text{MCLR}}$ Reset has occurred (set to '0' in hardware when a $\overline{\text{MCLR}}$ Reset occurs)
- bit 2 **$\overline{\text{RI}}$:** RESET Instruction Flag bit
 1 = A RESET instruction has not been executed or set to '1' by firmware
 0 = A RESET instruction has been executed (set to '0' in hardware upon executing a RESET instruction)
- bit 1 **$\overline{\text{POR}}$:** Power-on Reset Status bit
 1 = No Power-on Reset occurred
 0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)
- bit 0 **$\overline{\text{BOR}}$:** Brown-out Reset Status bit
 1 = No Brown-out Reset occurred
 0 = A Brown-out Reset occurred (must be set in software after a Power-on Reset or Brown-out Reset occurs)

PIC16(L)F1826/27

10.1 Independent Clock Source

The WDT derives its time base from the 31 kHz LFINTOSC internal oscillator. Time intervals in this chapter are based on a nominal interval of 1 ms. See **Section 30.0 “Electrical Specifications”** for the LFINTOSC tolerances.

10.2 WDT Operating Modes

The Watchdog Timer module has four operating modes controlled by the WDTE<1:0> bits in Configuration Word 1. See Table 10-1.

10.2.1 WDT IS ALWAYS ON

When the WDTE bits of Configuration Word 1 are set to ‘11’, the WDT is always on.

WDT protection is active during Sleep.

10.2.2 WDT IS OFF IN SLEEP

When the WDTE bits of Configuration Word 1 are set to ‘10’, the WDT is on, except in Sleep.

WDT protection is not active during Sleep.

10.2.3 WDT CONTROLLED BY SOFTWARE

When the WDTE bits of Configuration Word 1 are set to ‘01’, the WDT is controlled by the SWDTEN bit of the WDTCON register.

WDT protection is unchanged by Sleep. See Table 10-1 for more details.

TABLE 10-1: WDT OPERATING MODES

WDTE<1:0>	SWDTEN	Device Mode	WDT Mode
11	X	X	Active
10	X	Awake	Active
		Sleep	Disabled
01	1	X	Active
	0		Disabled
00	X	X	Disabled

TABLE 10-2: WDT CLEARING CONDITIONS

Conditions	WDT
WDTE<1:0> = 00	Cleared
WDTE<1:0> = 01 and SWDTEN = 0	
WDTE<1:0> = 10 and enter Sleep	
CLRWDT Command	
Oscillator Fail Detected	
Exit Sleep + System Clock = T1OSC, EXTRC, INTOSC, EXTCLK	
Exit Sleep + System Clock = XT, HS, LP	Cleared until the end of OST
Change INTOSC divider (IRCF bits)	Unaffected

10.3 Time-Out Period

The WDTPS bits of the WDTCON register set the time-out period from 1 ms to 256 seconds (nominal). After a Reset, the default time-out period is 2 seconds.

10.4 Clearing the WDT

The WDT is cleared when any of the following conditions occur:

- Any Reset
- CLRWDT instruction is executed
- Device enters Sleep
- Device wakes up from Sleep
- Oscillator fail event
- WDT is disabled
- Oscillator Start-up Timer (OST) is running

See Table 10-2 for more information.

10.5 Operation During Sleep

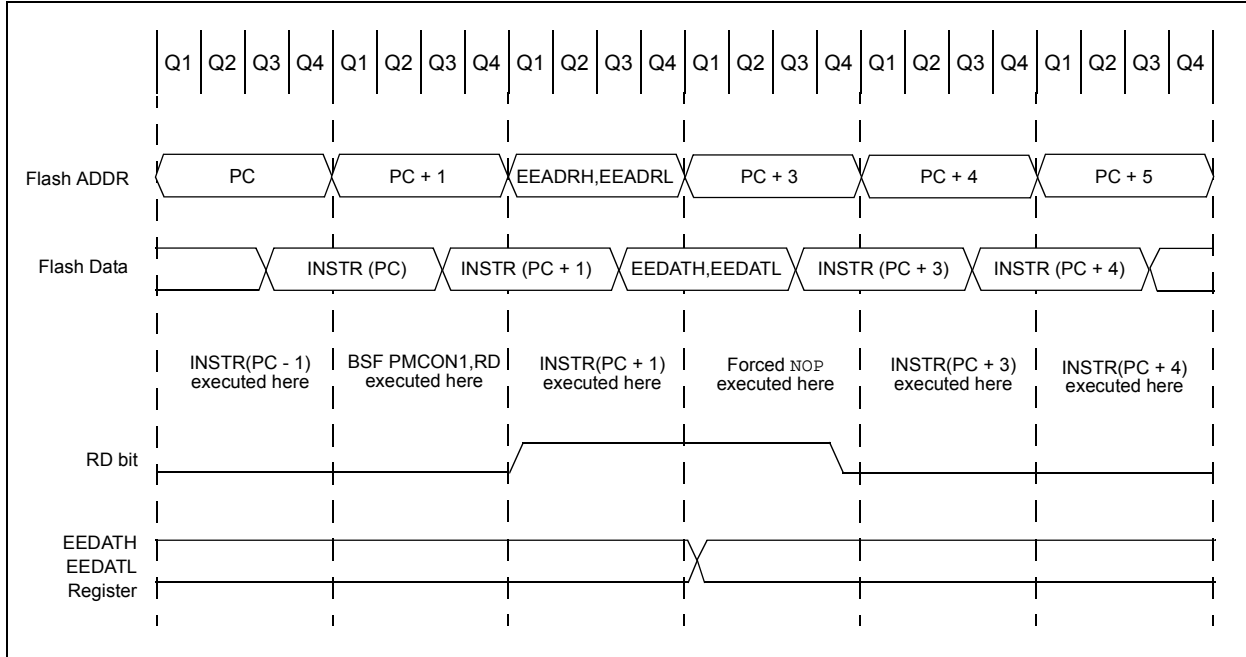
When the device enters Sleep, the WDT is cleared. If the WDT is enabled during Sleep, the WDT resumes counting.

When the device exits Sleep, the WDT is cleared again. The WDT remains clear until the OST, if enabled, completes. See **Section 5.0 “Oscillator Module (With Fail-Safe Clock Monitor)”** for more information on the OST.

When a WDT time-out occurs while the device is in Sleep, no Reset is generated. Instead, the device wakes up and resumes operation. The TO and PD bits in the STATUS register are changed to indicate the event. See **Register 3-1** for more information.

PIC16(L)F1826/27

FIGURE 11-1: FLASH PROGRAM MEMORY READ CYCLE EXECUTION



11.4 Modifying Flash Program Memory

When modifying existing data in a program memory row, and data within that row must be preserved, it must first be read and saved in a RAM image. Program memory is modified using the following steps:

1. Load the starting address of the row to be modified.
2. Read the existing data from the row into a RAM image.
3. Modify the RAM image to contain the new data to be written into program memory.
4. Load the starting address of the row to be rewritten.
5. Erase the program memory row.
6. Load the write latches with data from the RAM image.
7. Initiate a programming operation.
8. Repeat steps 6 and 7 as many times as required to reprogram the erased row.

11.5 User ID, Device ID and Configuration Word Access

Instead of accessing program memory or EEPROM data memory, the User ID's, Device ID/Revision ID and Configuration Words can be accessed when CFGS = 1 in the EECON1 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 11-2.

When read access is initiated on an address outside the parameters listed in Table 11-2, the EEDATH:EEDATL register pair is cleared.

TABLE 11-2: USER ID, DEVICE ID AND CONFIGURATION WORD ACCESS (CFGS = 1)

Address	Function	Read Access	Write Access
8000h-8003h	User IDs	Yes	Yes
8006h	Device ID/Revision ID	Yes	No
8007h-8008h	Configuration Words 1 and 2	Yes	No

EXAMPLE 11-3: CONFIGURATION WORD AND DEVICE ID ACCESS

```

* This code block will read 1 word of program memory at the memory address:
*   PROG_ADDR_LO (must be 00h-08h) data will be returned in the variables;
*   PROG_DATA_HI, PROG_DATA_LO

BANKSEL  EEADRL          ; Select correct Bank
MOVLW   PROG_ADDR_LO    ;
MOVWF   EEADRL          ; Store LSB of address
CLRF   EEADRH          ; Clear MSB of address

BSF     EECON1,CFGS     ; Select Configuration Space
BCF     INTCON,GIE     ; Disable interrupts
BSF     EECON1,RD      ; Initiate read
NOP     ; Executed (See Figure 11-1)
NOP     ; Ignored (See Figure 11-1)
BSF     INTCON,GIE     ; Restore interrupts

MOVF   EEDATL,W        ; Get LSB of word
MOVWF  PROG_DATA_LO    ; Store in user location
MOVF   EEDATH,W        ; Get MSB of word
MOVWF  PROG_DATA_HI    ; Store in user location
    
```


PIC16(L)F1826/27

11.6 Write Verify

Depending on the application, good programming practice may dictate that the value written to the data EEPROM or program memory should be verified (see Example 11-6) to the desired value to be written. Example 11-6 shows how to verify a write to EEPROM.

EXAMPLE 11-6: EEPROM WRITE VERIFY

```
BANKSEL EEDATL      ;
MOVWF  EEDATL, W    ;EEDATL not changed
                        ;from previous write
BSF    EECON1, RD   ;YES, Read the
                        ;value written
XORWF  EEDATL, W    ;
BTFSS  STATUS, Z    ;Is data the same
GOTO   WRITE_ERR    ;No, handle error
:      ;Yes, continue
```

17.0 DIGITAL-TO-ANALOG CONVERTER (DAC) MODULE

The Digital-to-Analog Converter supplies a variable voltage reference, ratiometric with the input source, with 32 selectable output levels.

The input of the DAC can be connected to:

- External VREF pins
- VDD supply voltage
- FVR (Fixed Voltage Reference)

The output of the DAC can be configured to supply a reference voltage to the following:

- Comparator positive input
- ADC input channel
- DACOUT pin
- Capacitive Sensing module (CSM)

The Digital-to-Analog Converter (DAC) can be enabled by setting the DACEN bit of the DACCON0 register.

17.1 Output Voltage Selection

The DAC has 32 voltage level ranges. The 32 levels are set with the DACR<4:0> bits of the DACCON1 register.

The DAC output voltage is determined by the equations in Equation 17-1.

EQUATION 17-1: DAC OUTPUT VOLTAGE

IF DACEN = 1

$$V_{OUT} = \left((V_{SOURCE+} - V_{SOURCE-}) \times \frac{DACR[4:0]}{2^5} \right) + V_{SOURCE-}$$

IF DACEN = 0 and DACLPS = 1 and DACR[4:0] = 1111

$$V_{OUT} = V_{SOURCE+}$$

IF DACEN = 0 and DACLPS = 0 and DACR[4:0] = 0000

$$V_{OUT} = V_{SOURCE-}$$

$$V_{SOURCE+} = V_{DD}, V_{REF}, \text{ or } FVR \text{ BUFFER } 2$$

$$V_{SOURCE-} = V_{SS}$$

17.2 Ratiometric Output Level

The DAC output value is derived using a resistor ladder with each end of the ladder tied to a positive and negative voltage reference input source. If the voltage of either input source fluctuates, a similar fluctuation will result in the DAC output value.

The value of the individual resistors within the ladder can be found in **Section 29.0 “Electrical Specifications”**.

17.3 DAC Voltage Reference Output

The DAC can be output to the DACOUT pin by setting the DACOE bit of the DACCON0 register to ‘1’. Selecting the DAC reference voltage for output on the DACOUT pin automatically overrides the digital output buffer and digital input threshold detector functions of that pin. Reading the DACOUT pin when it has been configured for DAC reference voltage output will always return a ‘0’.

Due to the limited current drive capability, a buffer must be used on the DAC voltage reference output for external connections to DACOUT. Figure 17-2 shows an example buffering technique.

18.0 SR LATCH

The module consists of a single SR Latch with multiple Set and Reset inputs as well as separate latch outputs. The SR Latch module includes the following features:

- Programmable input selection
- SR Latch output is available externally
- Separate Q and \bar{Q} outputs
- Firmware Set and Reset

The SR Latch can be used in a variety of analog applications, including oscillator circuits, one-shot circuit, hysteretic controllers, and analog timing applications.

18.1 Latch Operation

The latch is a Set-Reset Latch that does not depend on a clock source. Each of the Set and Reset inputs are active-high. The latch can be Set or Reset by:

- Software control (SRPS and SRPR bits)
- Comparator C1 output (SYNCC1OUT)
- Comparator C2 output (SYNCC2OUT)
- SRI pin
- Programmable clock (SRCLK)

The SRPS and the SRPR bits of the SRCON0 register may be used to set or reset the SR Latch, respectively. The latch is Reset-dominant. Therefore, if both Set and Reset inputs are high, the latch will go to the Reset state. Both the SRPS and SRPR bits are self resetting which means that a single write to either of the bits is all that is necessary to complete a latch Set or Reset operation.

The output from Comparator C1 or C2 can be used as the Set or Reset inputs of the SR Latch. The output of either comparator can be synchronized to the Timer1 clock source. See **Section 19.0 “Comparator Module”** and **Section 21.0 “Timer1 Module with Gate Control”** for more information.

An external source on the SRI pin can be used as the Set or Reset inputs of the SR Latch.

An internal clock source is available that can periodically set or reset the SR Latch. The SRCLK<2:0> bits in the SRCON0 register are used to select the clock source period. The SRSCKE and SRRCKE bits of the SRCON1 register enable the clock source to set or reset the SR Latch, respectively.

Note: Enabling both the Set and Reset inputs from any one source at the same time may result in indeterminate operation, as the Reset dominance cannot be assured.

18.2 Latch Output

The SRQEN and SRNQEN bits of the SRCON0 register control the Q and \bar{Q} latch outputs. Both of the SR Latch outputs may be directly output to an I/O pin at the same time.

The applicable TRIS bit of the corresponding port must be cleared to enable the port pin output driver.

18.3 Effects of a Reset

Upon any device Reset, the SR Latch output is not initialized to a known state. The user's firmware is responsible for initializing the latch output before enabling the output pins.

PIC16(L)F1826/27

NOTES:

25.2.6 SPI OPERATION IN SLEEP MODE

In SPI Master mode, module clocks may be operating at a different speed than when in full power mode; in the case of the Sleep mode, all clocks are halted.

Special care must be taken by the user when the MSSPx clock is much faster than the system clock.

In Slave mode, when MSSPx interrupts are enabled, after the master completes sending data, an MSSPx interrupt will wake the controller from Sleep.

If an exit from Sleep mode is not desired, MSSPx interrupts should be disabled.

In SPI Master mode, when the Sleep mode is selected, all module clocks are halted and the transmission/reception will remain in that state until the device wakes. After the device returns to Run mode, the module will resume transmitting and receiving data.

In SPI Slave mode, the SPI Transmit/Receive Shift register operates asynchronously to the device. This allows the device to be placed in Sleep mode and data to be shifted into the SPI Transmit/Receive Shift register. When all 8 bits have been received, the MSSPx interrupt flag bit will be set and if enabled, will wake the device.

TABLE 25-1: SUMMARY OF REGISTERS ASSOCIATED WITH SPI OPERATION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON0	RXDTSEL	SDO1SEL	SS1SEL	P2BSEL ⁽¹⁾	CCP2SEL ⁽¹⁾	P1DSEL	P1CSEL	CCP1SEL	119
ANSELA	—	—	—	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	123
ANSELB	ANSB7	ANSB6	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	—	128
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	86
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	87
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	91
SSPxBUF	Synchronous Serial Port Receive Buffer/Transmit Register								235*
SSPxCON1	WCOL	SSPxOV	SSPxEN	CKP	SSPxM3	SSPxM2	SSPxM1	SSPxM0	280
SSPxCON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	282
SSPxSTAT	SMP	CKE	D/ \bar{A}	P	S	R/ \bar{W}	UA	BF	279
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	122
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	127

Legend: — = Unimplemented location, read as '0'. Shaded cells are not used by the MSSPx in SPI mode.

* Page provides register information.

Note 1: PIC16(L)F1827 only.

PIC16(L)F1826/27

25.6.8 ACKNOWLEDGE SEQUENCE TIMING

An Acknowledge sequence is enabled by setting the Acknowledge Sequence Enable bit, ACKEN bit of the SSPxCON2 register. When this bit is set, the SCLx pin is pulled low and the contents of the Acknowledge data bit are presented on the SDAx pin. If the user wishes to generate an Acknowledge, then the ACKDT bit should be cleared. If not, the user should set the ACKDT bit before starting an Acknowledge sequence. The Baud Rate Generator then counts for one rollover period (TBRG) and the SCLx pin is deasserted (pulled high). When the SCLx pin is sampled high (clock arbitration), the Baud Rate Generator counts for TBRG. The SCLx pin is then pulled low. Following this, the ACKEN bit is automatically cleared, the Baud Rate Generator is turned off and the MSSPx module then goes into Idle mode (Figure 25-29).

25.6.8.1 WCOL Status Flag

If the user writes the SSPxBUF when an Acknowledge sequence is in progress, then WCOL is set and the contents of the buffer are unchanged (the write does not occur).

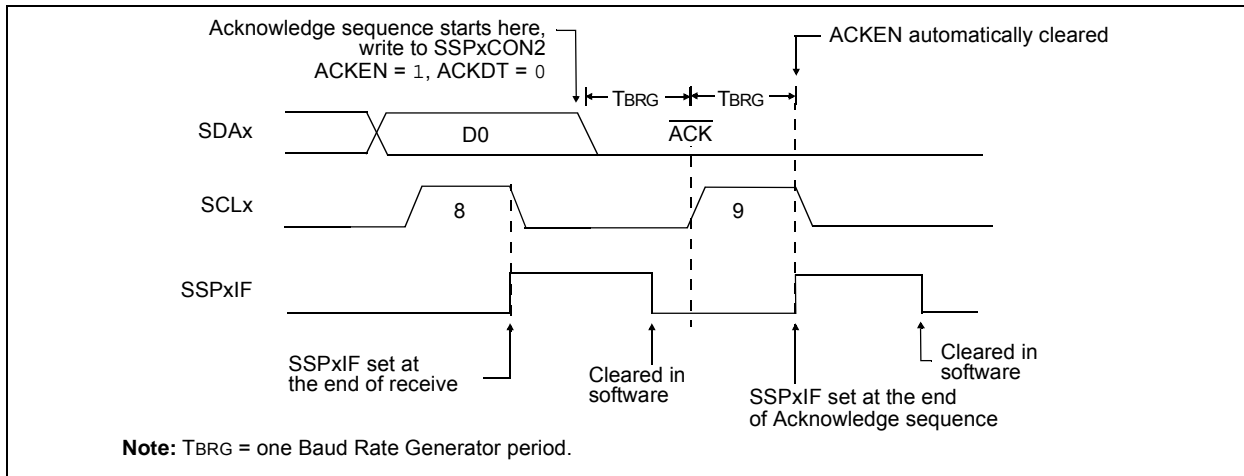
25.6.9 STOP CONDITION TIMING

A Stop bit is asserted on the SDAx pin at the end of a receive/transmit by setting the Stop Sequence Enable bit, PEN bit of the SSPxCON2 register. At the end of a receive/transmit, the SCLx line is held low after the falling edge of the ninth clock. When the PEN bit is set, the master will assert the SDAx line low. When the SDAx line is sampled low, the Baud Rate Generator is reloaded and counts down to '0'. When the Baud Rate Generator times out, the SCLx pin will be brought high and one TBRG (Baud Rate Generator rollover count) later, the SDAx pin will be deasserted. When the SDAx pin is sampled high while SCLx is high, the P bit of the SSPxSTAT register is set. A TBRG later, the PEN bit is cleared and the SSPxIF bit is set (Figure 25-30).

25.6.9.1 WCOL Status Flag

If the user writes the SSPxBUF when a Stop sequence is in progress, then the WCOL bit is set and the contents of the buffer are unchanged (the write does not occur).

FIGURE 25-30: ACKNOWLEDGE SEQUENCE WAVEFORM



PIC16(L)F1826/27

26.1.2 EUSART ASYNCHRONOUS RECEIVER

The Asynchronous mode is typically used in RS-232 systems. The receiver block diagram is shown in Figure 26-2. The data is received on the RX/DT pin and drives the data recovery block. The data recovery block is actually a high-speed shifter operating at 16 times the baud rate, whereas the serial Receive Shift Register (RSR) operates at the bit rate. When all 8 or 9 bits of the character have been shifted in, they are immediately transferred to a two character First-In-First-Out (FIFO) memory. The FIFO buffering allows reception of two complete characters and the start of a third character before software must start servicing the EUSART receiver. The FIFO and RSR registers are not directly accessible by software. Access to the received data is via the RCREG register.

26.1.2.1 Enabling the Receiver

The EUSART receiver is enabled for asynchronous operation by configuring the following three control bits:

- CREN = 1
- SYNC = 0
- SPEN = 1

All other EUSART control bits are assumed to be in their default state.

Setting the CREN bit of the RCSTA register enables the receiver circuitry of the EUSART. Clearing the SYNC bit of the TXSTA register configures the EUSART for asynchronous operation. Setting the SPEN bit of the RCSTA register enables the EUSART. The programmer must set the corresponding TRIS bit to configure the RX/DT I/O pin as an input.

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

26.1.2.2 Receiving Data

The receiver data recovery circuit initiates character reception on the falling edge of the first bit. The first bit, also known as the Start bit, is always a zero. The data recovery circuit counts one-half bit time to the center of the Start bit and verifies that the bit is still a zero. If it is not a zero then the data recovery circuit aborts character reception, without generating an error, and resumes looking for the falling edge of the Start bit. If the Start bit zero verification succeeds then the data recovery circuit counts a full bit time to the center of the next bit. The bit is then sampled by a majority detect circuit and the resulting '0' or '1' is shifted into the RSR. This repeats until all data bits have been sampled and shifted into the RSR. One final bit time is measured and the level sampled. This is the Stop bit, which is always a '1'. If the data recovery circuit samples a '0' in the Stop bit position then a framing error is set for this character, otherwise the framing error is cleared for this character. See **Section 26.1.2.4 "Receive Framing Error"** for more information on framing errors.

Immediately after all data bits and the Stop bit have been received, the character in the RSR is transferred to the EUSART receive FIFO and the RCIF interrupt flag bit of the PIR1 register is set. The top character in the FIFO is transferred out of the FIFO by reading the RCREG register.

Note: If the receive FIFO is overrun, no additional characters will be received until the overrun condition is cleared. See **Section 26.1.2.5 "Receive Overrun Error"** for more information on overrun errors.

26.1.2.3 Receive Interrupts

The RCIF interrupt flag bit of the PIR1 register is set whenever the EUSART receiver is enabled and there is an unread character in the receive FIFO. The RCIF interrupt flag bit is read-only, it cannot be set or cleared by software.

RCIF interrupts are enabled by setting all of the following bits:

- RCIE interrupt enable bit of the PIE1 register
- PEIE peripheral interrupt enable bit of the INTCON register
- GIE global interrupt enable bit of the INTCON register

The RCIF interrupt flag bit will be set when there is an unread character in the FIFO, regardless of the state of interrupt enable bits.

REGISTER 26-2: RCSTA: RECEIVE STATUS AND CONTROL REGISTER⁽¹⁾

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-0/0	R-0/0	R-0/0
SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
bit 7						bit 0	

Legend:

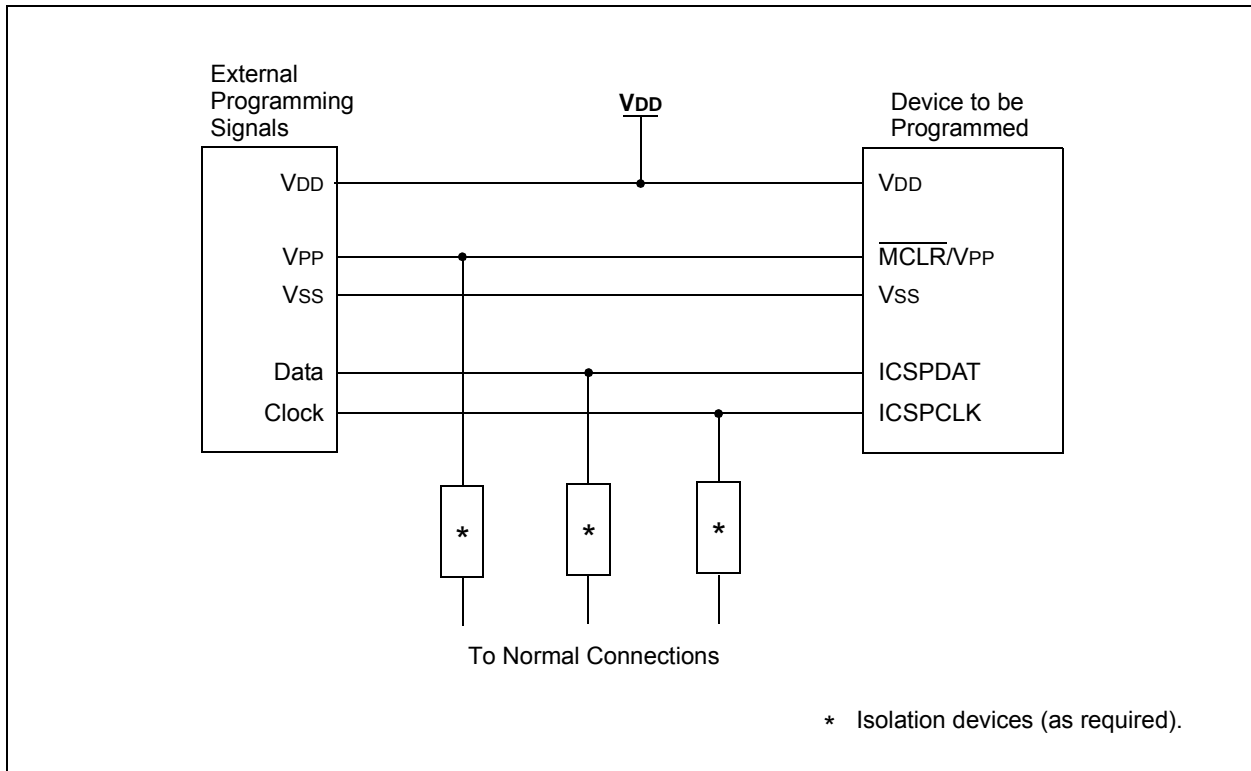
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

- bit 7 **SPEN:** Serial Port Enable bit
1 = Serial port enabled (configures RX/DT and TX/CK pins as serial port pins)
0 = Serial port disabled (held in Reset)
- bit 6 **RX9:** 9-bit Receive Enable bit
1 = Selects 9-bit reception
0 = Selects 8-bit reception
- bit 5 **SREN:** Single Receive Enable bit
Asynchronous mode:
Don't care
Synchronous mode – Master:
1 = Enables single receive
0 = Disables single receive
This bit is cleared after reception is complete.
Synchronous mode – Slave
Don't care
- bit 4 **CREN:** Continuous Receive Enable bit
Asynchronous mode:
1 = Enables receiver
0 = Disables receiver
Synchronous mode:
1 = Enables continuous receive until enable bit CREN is cleared (CREN overrides SREN)
0 = Disables continuous receive
- bit 3 **ADDEN:** Address Detect Enable bit
Asynchronous mode 9-bit (RX9 = 1):
1 = Enables address detection, enable interrupt and load the receive buffer when RSR<8> is set
0 = Disables address detection, all bytes are received and ninth bit can be used as parity bit
Asynchronous mode 8-bit (RX9 = 0):
Don't care
- bit 2 **FERR:** Framing Error bit
1 = Framing error (can be updated by reading RCREG register and receive next valid byte)
0 = No framing error
- bit 1 **OERR:** Overrun Error bit
1 = Overrun error (can be cleared by clearing bit CREN)
0 = No overrun error
- bit 0 **RX9D:** Ninth bit of Received Data
This can be address/data bit or a parity bit and must be calculated by user firmware.

For additional interface recommendations, refer to your specific device programmer manual prior to PCB design.

It is recommended that isolation devices be used to separate the programming pins from other circuitry. The type of isolation is highly dependent on the specific application and may include devices such as resistors, diodes, or even jumpers. See Figure 28-4 for more information.

FIGURE 28-4: TYPICAL CONNECTION FOR ICSP™ PROGRAMMING



PIC16(L)F1826/27

30.4 DC Characteristics: PIC16(L)F1826/27-I/E

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended				
Param No.	Sym.	Characteristic	Min.	Typ†	Max.	Units	Conditions
D030 D030A D031 D032 D033	V _{IL}	Input Low Voltage					
		I/O PORT:					
		with TTL buffer	—	—	0.8	V	$4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$
		with Schmitt Trigger buffer	—	—	$0.15 V_{DD}$	V	$1.8\text{V} \leq V_{DD} \leq 4.5\text{V}$
		with I ² C™ levels	—	—	$0.2 V_{DD}$	V	$2.0\text{V} \leq V_{DD} \leq 5.5\text{V}$
		with SMBus™ levels	—	—	0.8	V	$2.7\text{V} \leq V_{DD} \leq 5.5\text{V}$
		MCLR, OSC1 (RC mode) ⁽¹⁾	—	—	$0.2 V_{DD}$	V	
D033		OSC1 (HS mode)	—	—	$0.3 V_{DD}$	V	
D040 D040A D041 D042 D043A D043B	V _{IH}	Input High Voltage					
		I/O ports:					
		with TTL buffer	2.0	—	—	V	$4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$
			$0.25 V_{DD} + 0.8$	—	—	V	$1.8\text{V} \leq V_{DD} \leq 4.5\text{V}$
		with Schmitt Trigger buffer	$0.8 V_{DD}$	—	—	V	$2.0\text{V} \leq V_{DD} \leq 5.5\text{V}$
		with I ² C™ levels	$0.7 V_{DD}$	—	—	V	
		with SMBus™ levels	2.1	—	—	V	$2.7\text{V} \leq V_{DD} \leq 5.5\text{V}$
		MCLR	$0.8 V_{DD}$	—	—	V	
D043A		OSC1 (HS mode)	$0.7 V_{DD}$	—	—	V	
D043B		OSC1 (RC mode)	$0.9 V_{DD}$	—	—	V	(Note 1)
D060 D061	I _{IL}	Input Leakage Current⁽²⁾					
		I/O ports	—	± 5	± 100	nA	$V_{SS} \leq V_{PIN} \leq V_{DD}$, Pin at high-impedance at 85°C
D061		MCLR ⁽³⁾	—	± 50	± 1000	nA	$V_{SS} \leq V_{PIN} \leq V_{DD}$ at 85°C
D070*	I _{PUR}	Weak Pull-up Current					
			25 25	100 140	200 300	μA	$V_{DD} = 3.3\text{V}$, $V_{PIN} = V_{SS}$ $V_{DD} = 5.0\text{V}$, $V_{PIN} = V_{SS}$
D080	V _{OL}	Output Low Voltage⁽⁴⁾					
		I/O ports	—	—	0.6	V	I _{OL} = 8mA, V _{DD} = 5V I _{OL} = 6mA, V _{DD} = 3.3V I _{OL} = 1.8mA, V _{DD} = 1.8V
D090	V _{OH}	Output High Voltage⁽⁴⁾					
		I/O ports	$V_{DD} - 0.7$	—	—	V	I _{OH} = 3.5mA, V _{DD} = 5V I _{OH} = 3mA, V _{DD} = 3.3V I _{OH} = 1mA, V _{DD} = 1.8V
D101* D101A*	C _{OSC2} C _{IO}	Capacitive Loading Specs on Output Pins					
		OSC2 pin	—	—	15	pF	In XT, HS and LP modes when external clock is used to drive OSC1
D101A*		All I/O pins	—	—	50	pF	

* 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: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended to use an external clock in RC mode.

2: Negative current is defined as current sourced by the pin.

3: The leakage current on the MCLR pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

4: Including OSC2 in CLKOUT mode.

PIC16(L)F1826/27

TABLE 30-6: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

Standard Operating Conditions (unless otherwise stated) Operating Temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$								
Param No.	Sym.	Characteristic		Min.	Typ†	Max.	Units	Conditions
40*	Tt0H	T0CKI High Pulse Width	No Prescaler	$0.5 T_{CY} + 20$	—	—	ns	
			With Prescaler	10	—	—	ns	
41*	Tt0L	T0CKI Low Pulse Width	No Prescaler	$0.5 T_{CY} + 20$	—	—	ns	
			With Prescaler	10	—	—	ns	
42*	Tt0P	T0CKI Period		Greater of: 20 or $\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (2, 4, ..., 256)
45*	Tt1H	T1CKI High Time	Synchronous, No Prescaler	$0.5 T_{CY} + 20$	—	—	ns	
			Synchronous, with Prescaler	15	—	—	ns	
			Asynchronous	30	—	—	ns	
46*	Tt1L	T1CKI Low Time	Synchronous, No Prescaler	$0.5 T_{CY} + 20$	—	—	ns	
			Synchronous, with Prescaler	15	—	—	ns	
			Asynchronous	30	—	—	ns	
47*	Tt1P	T1CKI Input Period	Synchronous	Greater of: 30 or $\frac{T_{CY} + 40}{N}$	—	—	ns	
			Asynchronous	60	—	—	ns	
48	Ft1	Timer1 Oscillator Input Frequency Range (oscillator enabled by setting bit T1OSCEN)		32.4	32.768	33.1	kHz	
49*	TCKEZTMR1	Delay from External Clock Edge to Timer Increment		2 TOSC	—	7 TOSC	—	Timers in Sync mode

* 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.

FIGURE 30-11: CAPTURE/COMPARE/PWM TIMINGS (CCP)

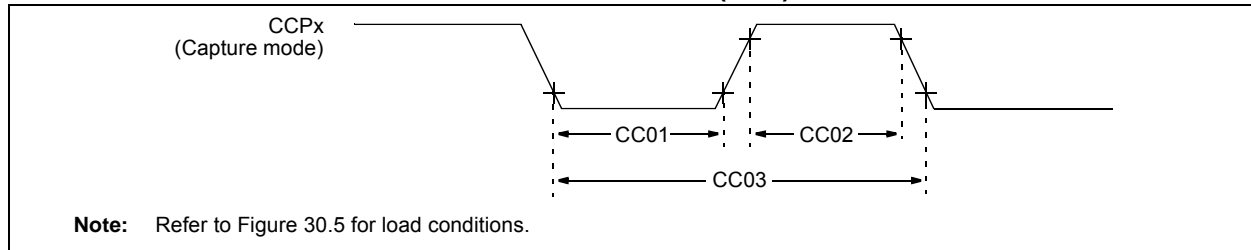


TABLE 30-7: CAPTURE/COMPARE/PWM REQUIREMENTS (CCP)

Standard Operating Conditions (unless otherwise stated) Operating Temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$								
Param No.	Sym.	Characteristic		Min.	Typ†	Max.	Units	Conditions
CC01*	TccL	CCPx Input Low Time	No Prescaler	$0.5 T_{CY} + 20$	—	—	ns	
			With Prescaler	20	—	—	ns	
CC02*	TccH	CCPx Input High Time	No Prescaler	$0.5 T_{CY} + 20$	—	—	ns	
			With Prescaler	20	—	—	ns	
CC03*	TccP	CCPx Input Period		$\frac{3 T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1, 4 or 16)

* 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.

32.0 DEVELOPMENT SUPPORT

The PIC[®] microcontrollers and dsPIC[®] digital signal controllers are supported with a full range of software and hardware development tools:

- Integrated Development Environment
 - MPLAB[®] IDE Software
- Compilers/Assemblers/Linkers
 - MPLAB C Compiler for Various Device Families
 - HI-TECH C for Various Device Families
 - MPASM[™] Assembler
 - MPLINK[™] Object Linker/
MPLIB[™] Object Librarian
 - MPLAB Assembler/Linker/Librarian for Various Device Families
- Simulators
 - MPLAB SIM Software Simulator
- Emulators
 - MPLAB REAL ICE[™] In-Circuit Emulator
- In-Circuit Debuggers
 - MPLAB ICD 3
 - PICKit[™] 3 Debug Express
- Device Programmers
 - PICKit[™] 2 Programmer
 - MPLAB PM3 Device Programmer
- Low-Cost Demonstration/Development Boards, Evaluation Kits, and Starter Kits

32.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8/16/32-bit microcontroller market. The MPLAB IDE is a Windows[®] operating system-based application that contains:

- A single graphical interface to all debugging tools
 - Simulator
 - Programmer (sold separately)
 - In-Circuit Emulator (sold separately)
 - In-Circuit Debugger (sold separately)
- A full-featured editor with color-coded context
- A multiple project manager
- Customizable data windows with direct edit of contents
- High-level source code debugging
- Mouse over variable inspection
- Drag and drop variables from source to watch windows
- Extensive on-line help
- Integration of select third party tools, such as IAR C Compilers

The MPLAB IDE allows you to:

- Edit your source files (either C or assembly)
- One-touch compile or assemble, and download to emulator and simulator tools (automatically updates all project information)
- Debug using:
 - Source files (C or assembly)
 - Mixed C and assembly
 - Machine code

MPLAB IDE supports multiple debugging tools in a single development paradigm, from the cost-effective simulators, through low-cost in-circuit debuggers, to full-featured emulators. This eliminates the learning curve when upgrading to tools with increased flexibility and power.

Steering Synchronization	224
PWM Mode. See Enhanced Capture/Compare/PWM	212
PWM Steering	223
PWMxCON Register	229

R

RCREG	292
RCREG Register	29
RCSTA Register	29, 295
Reader Response	404
Read-Modify-Write Operations	325
Reference Clock	69
Associated Registers	71
Registers	
ADCON0 (ADC Control 0)	145
ADCON1 (ADC Control 1)	146
ADRESH (ADC Result High) with ADFM = 0)	147
ADRESH (ADC Result High) with ADFM = 1)	148
ADRESL (ADC Result Low) with ADFM = 0)	147
ADRESL (ADC Result Low) with ADFM = 1)	148
ANSELA (PORTA Analog Select)	123
ANSELB (PORTB Analog Select)	128
APFCON0 (Alternate Pin Function Control 0)	119
APFCON1 (Alternate Pin Function Control 1)	119
BAUDCON (Baud Rate Control)	296
BORCON Brown-out Reset Control)	75
CCPTMRS (PWM Timer Selection Control)	227
CCPxAS (CCPx Auto-Shutdown Control)	228
CCPxCON (ECCPx Control)	226
CLKRCON (Reference Clock Control)	70
CMOUT (Comparator Output)	171
CMxCON0 (Cx Control)	170
CMxCON1 (Cx Control 1)	171
Configuration Word 1	44
Configuration Word 2	46
Core Function, Summary	27
CPSCON0 (Capacitive Sensing Control Register 0)	318
CPSCON1 (Capacitive Sensing Control Register 1)	319
DACCON0	156
DACCON1	156
Device ID	49
EEADRL (EEPROM Address)	113
EECON1 (EEPROM Control 1)	115
EECON2 (EEPROM Control 2)	116
EEDATH (EEPROM Data)	113, 114
EEDATL (EEPROM Data)	113
FVRCON	136
INTCON (Interrupt Control)	86
IOCBF (Interrupt-on-Change Flag)	132
IOCBN (Interrupt-on-Change Negative Edge)	132
IOCBP (Interrupt-on-Change Positive Edge)	132
LATA (Data Latch PORTA)	122
LATB (Data Latch PORTB)	127
MDCARH (Modulation High Carrier Control Register)	200
MDCARL (Modulation Low Carrier Control Register)	201
MDCON (Modulation Control Register)	198
MDSRC (Modulation Source Control Register)	199
OPTION_REG (OPTION)	176
OSCCON (Oscillator Control)	65
OSCSTAT (Oscillator Status)	66
OSCTUNE (Oscillator Tuning)	67
PCON (Power Control Register)	79
PCON (Power Control)	79
PIE1 (Peripheral Interrupt Enable 1)	87
PIE2 (Peripheral Interrupt Enable 2)	88

PIE3 (Peripheral Interrupt Enable 3)	89
PIE4 (Peripheral Interrupt Enable 4)	90
PIR1 (Peripheral Interrupt Register 1)	91
PIR2 (Peripheral Interrupt Request 2)	92
PIR3 (Peripheral Interrupt Request 3)	93
PIR4 (Peripheral Interrupt Request 4)	94
PORTA	122
PORTB	127
PSTRxCON (PWM Steering Control)	230
PWMxCON (Enhanced PWM Control)	229
RCREG	302
RCSTA (Receive Status and Control)	295
SPBRGH	297
SPBRGL	297
Special Function, Summary	28
SRCON0 (SR Latch Control 0)	159
SRCON1 (SR Latch Control 1)	160
SSPxADD (MSSPx Address and Baud Rate, I ² C Mode)	283
SSPxCON1 (MSSPx Control 1)	280
SSPxCON2 (SSPx Control 2)	281
SSPxCON3 (SSPx Control 3)	282
SSPxMSK (SSPx Mask)	283
SSPxSTAT (SSPx Status)	279
STATUS	21
T1CON (Timer1 Control)	185
T1GCON (Timer1 Gate Control)	186
TRISA (Tri-State PORTA)	122
TRISB (Tri-State PORTB)	127
TXCON	191
TXSTA (Transmit Status and Control)	294
WDTCON (Watchdog Timer Control)	99
WPUB (Weak Pull-up PORTB)	123, 128
RESET	335
Reset	73
Reset Instruction	76
Resets	73
Associated Registers	80
Revision History	393

S

Shoot-through Current	222
Software Simulator (MPLAB SIM)	381
SPBRG Register	29
SPBRGH Register	297
SPBRGL Register	297
Special Event Trigger	143
Special Function Registers (SFRs)	28
SPI Mode (MSSPx)	
Associated Registers	241
SPI Clock	237
SR Latch	157
Associated registers w/ SR Latch	161
SRCON0 Register	159
SRCON1 Register	160
SSP1ADD Register	30
SSP1BUF Register	30
SSP1CON Register	30
SSP1CON2 Register	30
SSP1CON3 Register	30
SSP1MSK Register	30
SSP1STAT Register	30
SSP2ADD Register	30
SSP2BUF Register	30
SSP2CON1 Register	30
SSP2CON2 Register	30