

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, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	17
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 12x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-UQFN Exposed Pad
Supplier Device Package	20-UQFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16lf1829-i-gz

Table of Contents

1.0	Device Overview	9
2.0	Enhanced Mid-range CPU	17
3.0	Memory Organization	19
4.0	Device Configuration	47
5.0	Oscillator Module (With Fail-Safe Clock Monitor).....	53
6.0	Reference Clock Module	71
7.0	Resets	74
8.0	Interrupts	82
9.0	Power-Down Mode (Sleep)	96
10.0	Watchdog Timer	98
11.0	Data EEPROM and Flash Program Memory Control.....	102
12.0	I/O Ports	116
13.0	Interrupt-on-Change	136
14.0	Fixed Voltage Reference (FVR)	141
15.0	Temperature Indicator Module	143
16.0	Analog-to-Digital Converter (ADC) Module	144
17.0	Digital-to-Analog Converter (DAC) Module	157
18.0	SR Latch.....	162
19.0	Comparator Module.....	167
20.0	Timer0 Module	174
21.0	Timer1 Module with Gate Control.....	177
22.0	Timer2/4/6 Modules.....	188
23.0	Data Signal Modulator.....	192
24.0	Capture/Compare/PWM Modules	201
25.0	Master Synchronous Serial Port (MSSP1 and MSSP2) Module	229
26.0	Enhanced Universal Synchronous Asynchronous Receiver Transmitter (EUSART).....	281
27.0	Capacitive Sensing (CPS) Module	309
28.0	In-Circuit Serial Programming™ (ICSP™)	317
29.0	Instruction Set Summary	320
30.0	Electrical Specifications.....	334
31.0	DC and AC Characteristics Graphs and Charts	370
32.0	Development Support.....	401
33.0	Packaging Information.....	405
	Appendix A: Data Sheet Revision History.....	432
	Appendix B: Migrating From Other PIC® Devices	432
	The Microchip Web Site	433
	Customer Change Notification Service	433
	Customer Support.....	433
	Product Identification System	434

PIC16(L)F1825/9

TABLE 3-8: SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	Name	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 1												
080h ⁽¹⁾	INDF0	Addressing this location uses contents of FSR0H/FSR0L to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx	
081h ⁽¹⁾	INDF1	Addressing this location uses contents of FSR1H/FSR1L to address data memory (not a physical register)								xxxx xxxx	xxxx xxxx	
082h ⁽¹⁾	PCL	Program Counter (PC) Least Significant Byte								0000 0000	0000 0000	
083h ⁽¹⁾	STATUS	—	—	—	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	---1 1000	---q quuu	
084h ⁽¹⁾	FSR0L	Indirect Data Memory Address 0 Low Pointer								0000 0000	uuuu uuuu	
085h ⁽¹⁾	FSR0H	Indirect Data Memory Address 0 High Pointer								0000 0000	0000 0000	
086h ⁽¹⁾	FSR1L	Indirect Data Memory Address 1 Low Pointer								0000 0000	uuuu uuuu	
087h ⁽¹⁾	FSR1H	Indirect Data Memory Address 1 High Pointer								0000 0000	0000 0000	
088h ⁽¹⁾	BSR	—	—	—	BSR<4:0>					---0 0000	---0 0000	
089h ⁽¹⁾	WREG	Working Register								0000 0000	uuuu uuuu	
08Ah ⁽¹⁾	PCLATH	—	Write Buffer for the upper 7 bits of the Program Counter								-000 0000	-000 0000
08Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 0000	0000 0000	
08Ch	TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	--11 1111	--11 1111	
08Dh	TRISB ⁽²⁾	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	1111 ----	1111 ----	
08Eh	TRISC	TRISC7 ⁽²⁾	TRISC6 ⁽²⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111	
08Fh	—	Unimplemented								—	—	
090h	—	Unimplemented								—	—	
091h	PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000	
092h	PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	—	—	CCP2IE	0000 0--0	0000 0--0	
093h	PIE3	—	—	CCP4IE	CCP3IE	TMR6IE	—	TMR4IE	—	--00 0-0-	--00 0-0-	
094h	PIE4 ⁽²⁾	—	—	—	—	—	—	BCL2IE	SSP2IE	---- --00	---- --00	
095h	OPTION_REG	$\overline{\text{WPUEN}}$	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			1111 1111	1111 1111	
096h	PCON	STKOVF	STKUNF	—	—	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	00-- 11qq	qq-- qqqu	
097h	WDTCON	—	—	WDTPS<4:0>					SWDTEN	--01 0110	--01 0110	
098h	OSCTUNE	—	—	TUN<5:0>							--00 0000	--00 0000
099h	OSCCON	SPLLEN	IRCF<3:0>				—	SCS<1:0>		0011 1-00	0011 1-00	
09Ah	OSCSTAT	T1OSCR	PLLRF	OSTS	HFIOFR	HFIOFL	MFIOFR	LFIOFR	HFIOFS	10q0 0q00	qqqq qq0q	
09Bh	ADRESL	A/D Result Register Low								xxxx xxxx	uuuu uuuu	
09Ch	ADRESH	A/D Result Register High								xxxx xxxx	uuuu uuuu	
09Dh	ADCON0	—	CHS<4:0>					$\overline{\text{GO/DONE}}$	ADON	-000 0000	-000 0000	
09Eh	ADCON1	ADFM	ADCS<2:0>			—	ADNREF	ADPREF<1:0>		0000 -000	0000 -000	
09Fh	—	Unimplemented								—	—	

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

- Note** 1: These registers can be addressed from any bank.
2: PIC16(L)F1829 only.
3: PIC16(L)F1825 only.
4: Unimplemented, read as '1'.

5.2 Clock Source Types

Clock sources can be classified as external or internal.

External clock sources rely on external circuitry for the clock source to function. Examples are: oscillator modules (EC mode), quartz crystal resonators or ceramic resonators (LP, XT and HS modes) and Resistor-Capacitor (RC) mode circuits.

Internal clock sources are contained internally within the oscillator module. The internal oscillator block has two internal oscillators and a dedicated Phase-Lock Loop (HFPLL) that are used to generate three internal system clock sources: the 16 MHz High-Frequency Internal Oscillator (HFINTOSC), 500 kHz (MFINTOSC) and the 31 kHz Low-Frequency Internal Oscillator (LFINTOSC).

The system clock can be selected between external or internal clock sources via the System Clock Select (SCS) bits in the OSCCON register. See **Section 5.3 “Clock Switching”** for additional information.

5.2.1 EXTERNAL CLOCK SOURCES

An external clock source can be used as the device system clock by performing one of the following actions:

- Program the FOSC<2:0> bits in the Configuration Word 1 to select an external clock source that will be used as the default system clock upon a device Reset.
- Write the SCS<1:0> bits in the OSCCON register to switch the system clock source to:
 - Timer1 Oscillator during run-time, or
 - An external clock source determined by the value of the FOSC bits.

See **Section 5.3 “Clock Switching”** for more information.

5.2.1.1 EC Mode

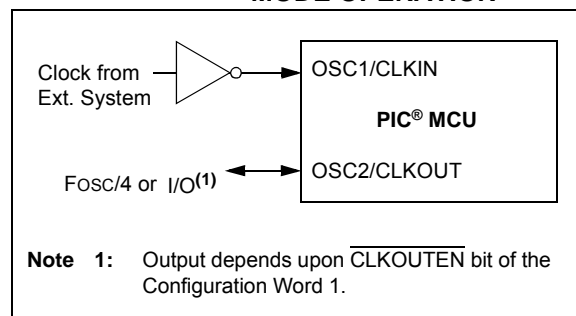
The External Clock (EC) mode allows an externally generated logic level signal to be the system clock source. When operating in this mode, an external clock source is connected to the OSC1 input. OSC2/CLKOUT is available for general purpose I/O or CLKOUT. Figure 5-2 shows the pin connections for EC mode.

EC mode has three power modes to select from through Configuration Word 1:

- High power, 4-32 MHz (FOSC = 111)
- Medium power, 0.5-4 MHz (FOSC = 110)
- Low power, 0-0.5 MHz (FOSC = 101)

The Oscillator Start-up Timer (OST) is disabled when EC mode is selected. Therefore, there is no delay in operation after a Power-on Reset (POR) or wake-up from Sleep. Because the PIC® MCU design is fully static, stopping the external clock input will have the effect of halting the device while leaving all data intact. Upon restarting the external clock, the device will resume operation as if no time had elapsed.

FIGURE 5-2: EXTERNAL CLOCK (EC) MODE OPERATION



5.2.1.2 LP, XT, HS Modes

The LP, XT and HS modes support the use of quartz crystal resonators or ceramic resonators connected to OSC1 and OSC2 (Figure 5-3). The three modes select a low, medium or high gain setting of the internal inverter-amplifier to support various resonator types and speed.

LP Oscillator mode selects the lowest gain setting of the internal inverter-amplifier. LP mode current consumption is the least of the three modes. This mode is designed to drive only 32.768 kHz tuning-fork type crystals (watch crystals).

XT Oscillator mode selects the intermediate gain setting of the internal inverter-amplifier. XT mode current consumption is the medium of the three modes. This mode is best suited to drive resonators with a medium drive level specification.

HS Oscillator mode selects the highest gain setting of the internal inverter-amplifier. HS mode current consumption is the highest of the three modes. This mode is best suited for resonators that require a high drive setting.

Figure 5-3 and Figure 5-4 show typical circuits for quartz crystal and ceramic resonators, respectively.

11.3.2 ERASING FLASH PROGRAM MEMORY

While executing code, program memory can only be erased by rows. To erase a row:

1. Load the EEADRH:EEADRL register pair with the address of new row to be erased.
2. Clear the CFGS bit of the EECON1 register.
3. Set the EEPGD, FREE, and WREN bits of the EECON1 register.
4. Write 55h, then AAh, to EECON2 (Flash programming unlock sequence).
5. Set control bit WR of the EECON1 register to begin the erase operation.
6. Poll the FREE bit in the EECON1 register to determine when the row erase has completed.

See Example 11-4.

After the “BSF EECON1, WR” instruction, the processor requires two cycles to set up the erase operation. The user must place two NOP instructions after the WR bit is set. The processor will halt internal operations for the typical 2 ms erase time. This is not Sleep mode as the clocks and peripherals will continue to run. After the erase cycle, the processor will resume operation with the third instruction after the EECON1 write instruction.

11.3.3 WRITING TO FLASH PROGRAM MEMORY

Program memory is programmed using the following steps:

1. Load the starting address of the word(s) to be programmed.
2. Load the write latches with data.
3. Initiate a programming operation.
4. Repeat steps 1 through 3 until all data is written.

Before writing to program memory, the word(s) to be written must be erased or previously unwritten. Program memory can only be erased one row at a time. No automatic erase occurs upon the initiation of the write.

Program memory can be written one or more words at a time. The maximum number of words written at one time is equal to the number of write latches. See Figure 11-2 for more details. The write latches are aligned to the address boundary defined by EEADRL as shown in Table 11-1. Write operations do not cross these boundaries. At the completion of a program memory write operation, the write latches are reset to contain 0x3FFF.

The following steps should be completed to load the write latches and program a block of program memory. These steps are divided into two parts. First, all write latches are loaded with data except for the last program memory location. Then, the last write latch is loaded and the programming sequence is initiated. A special unlock sequence is required to load a write latch with data or initiate a Flash programming operation. This unlock sequence should not be interrupted.

1. Set the EEPGD and WREN bits of the EECON1 register.
2. Clear the CFGS bit of the EECON1 register.
3. Set the LWLO bit of the EECON1 register. When the LWLO bit of the EECON1 register is ‘1’, the write sequence will only load the write latches and will not initiate the write to Flash program memory.
4. Load the EEADRH:EEADRL register pair with the address of the location to be written.
5. Load the EEDATH:EEDATL register pair with the program memory data to be written.
6. Write 55h, then AAh, to EECON2, then set the WR bit of the EECON1 register (Flash programming unlock sequence). The write latch is now loaded.
7. Increment the EEADRH:EEADRL register pair to point to the next location.
8. Repeat steps 5 through 7 until all but the last write latch has been loaded.
9. Clear the LWLO bit of the EECON1 register. When the LWLO bit of the EECON1 register is ‘0’, the write sequence will initiate the write to Flash program memory.
10. Load the EEDATH:EEDATL register pair with the program memory data to be written.
11. Write 55h, then AAh, to EECON2, then set the WR bit of the EECON1 register (Flash programming unlock sequence). The entire latch block is now written to Flash program memory.

It is not necessary to load the entire write latch block with user program data. However, the entire write latch block will be written to program memory.

An example of the complete write sequence for eight words is shown in Example 11-5. The initial address is loaded into the EEADRH:EEADRL register pair; the eight words of data are loaded using indirect addressing.

PIC16(L)F1825/9

REGISTER 12-1: APFCON0: ALTERNATE PIN FUNCTION CONTROL REGISTER 0

R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	U-0	U-0
RXDTSEL	SDO1SEL	SS1SEL	—	T1GSEL	TXCKSEL	—	—
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 **RXDTSEL:** Pin Selection bit
For 14 Pin Devices (PIC16(L)F1825):
0 = RX/DT function is on RC5
1 = RX/DT function is on RA1
For 20 Pin Devices (PIC16(L)F1829):
0 = RX/DT function is on RB5
1 = RX/DT function is on RC5
- bit 6 **SDO1SEL:** Pin Selection bit
For 14 Pin Devices (PIC16(L)F1825):
0 = SDO1 function is on RC2
1 = SDO1 function is on RA4
For 20 Pin Devices (PIC16(L)F1829):
Bit is read-only, '0'
SDO1 function is always on RC7.
- bit 5 **SS1SEL:** Pin Selection bit
For 14 Pin Devices (PIC16(L)F1825):
0 = SS1 function is on RC3
1 = SS1 function is on RA3
For 20 Pin Devices (PIC16(L)F1829):
Bit is read-only, '0'
SS1 function is always on RC6.
- bit 4 **Unimplemented:** Read as '0'
- bit 3 **T1GSEL:** Pin Selection bit
0 = T1G function is on RA4
1 = T1G function is on RA3
- bit 2 **TXCKSEL:** Pin Selection bit
For 14 Pin Devices (PIC16(L)F1825):
0 = TX/CK function is on RC4
1 = TX/CK function is on RA0
For 20 Pin Devices (PIC16(L)F1829):
0 = TX/CK function is on RB7
1 = TX/CK function is on RC4
- bit 1-0 **Unimplemented:** Read as '0'

PIC16(L)F1825/9

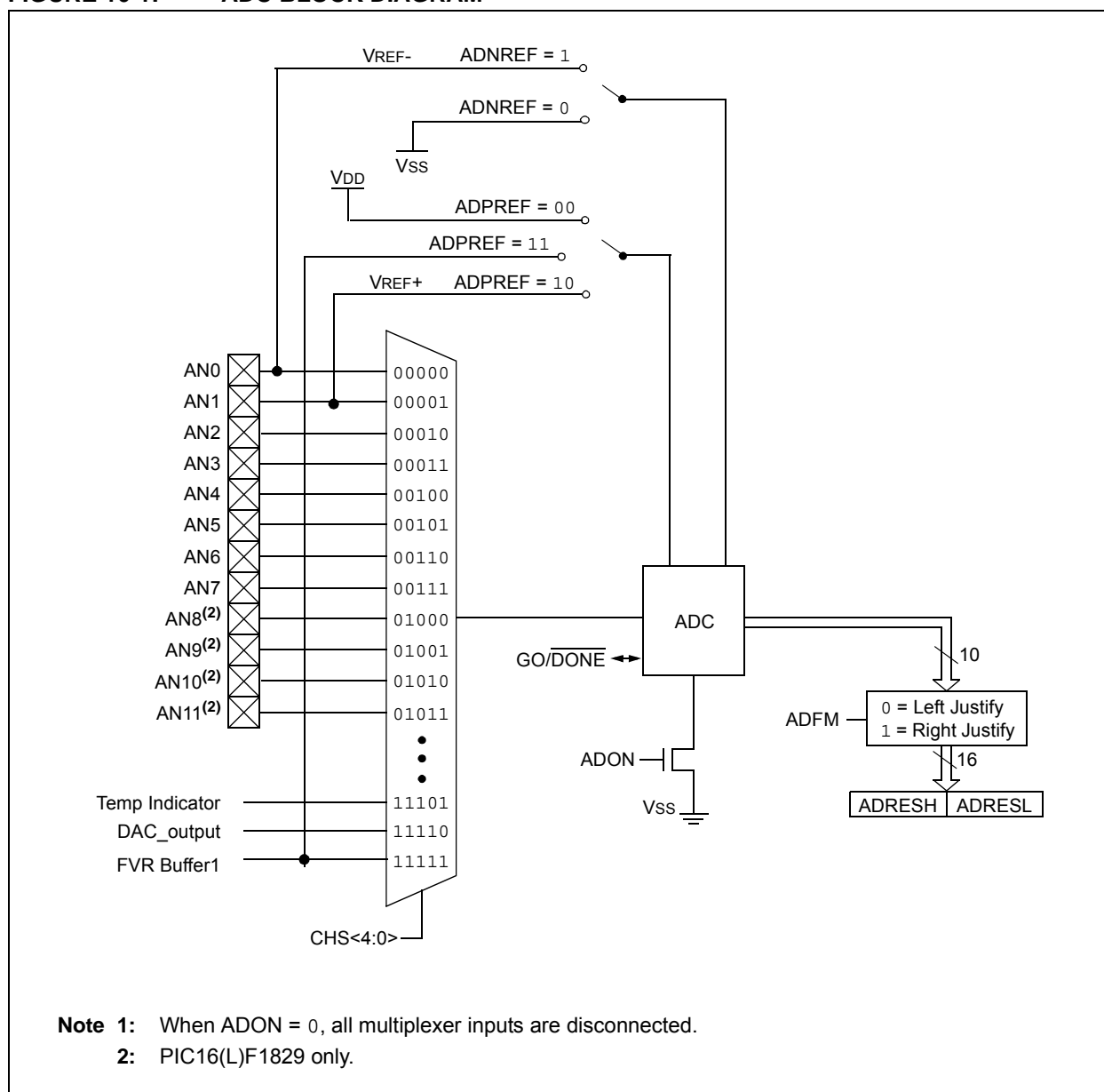
16.0 ANALOG-TO-DIGITAL CONVERTER (ADC) MODULE

The Analog-to-Digital Converter (ADC) allows conversion of an analog input signal to a 10-bit binary representation of that signal. This device uses analog inputs, which are multiplexed into a single sample and hold circuit. The output of the sample and hold is connected to the input of the converter. The converter generates a 10-bit binary result via successive approximation and stores the conversion result into the ADC result registers (ADRESH:ADRESL register pair). Figure 16-1 shows the block diagram of the ADC.

The ADC voltage reference is software selectable to be either internally generated or externally supplied.

The ADC can generate an interrupt upon completion of a conversion. This interrupt can be used to wake-up the device from Sleep.

FIGURE 16-1: ADC BLOCK DIAGRAM



16.1 ADC Configuration

When configuring and using the ADC the following functions must be considered:

- Port configuration
- Channel selection
- ADC voltage reference selection
- ADC conversion clock source
- Interrupt control
- Result formatting

16.1.1 PORT CONFIGURATION

The ADC can be used to convert both analog and digital signals. When converting analog signals, the I/O pin should be configured for analog by setting the associated TRIS and ANSEL bits. Refer to **Section 12.0 “I/O Ports”** for more information.

Note: Analog voltages on any pin that is defined as a digital input may cause the input buffer to conduct excess current.

16.1.2 CHANNEL SELECTION

There are up to 15 channel selections available:

- AN<7:0> pins (PIC16(L)F1825 only)
- AN<11:0> pins (PIC16(L)F1829 only)
- Temperature Indicator
- DAC_output
- FVR Buffer1

Refer to **Section 17.0 “Digital-to-Analog Converter (DAC) Module”**, **Section 14.0 “Fixed Voltage Reference (FVR)”** and **Section 15.0 “Temperature Indicator Module”** for more information on these channel selections.

The CHS bits of the ADCON0 register determine which channel is connected to the sample and hold circuit.

When changing channels, a delay is required before starting the next conversion. Refer to **Section 16.2 “ADC Operation”** for more information.

16.1.3 ADC VOLTAGE REFERENCE

The ADPREF bits of the ADCON1 register provides control of the positive voltage reference. The positive voltage reference can be:

- VREF+ pin
- VDD
- FVR 2.048V
- FVR 4.096V (Not available on LF devices)

The ADNREF bits of the ADCON1 register provides control of the negative voltage reference. The negative voltage reference can be:

- VREF- pin
- VSS

See **Section 14.0 “Fixed Voltage Reference (FVR)”** for more details on the Fixed Voltage Reference.

16.1.4 CONVERSION CLOCK

The source of the conversion clock is software selectable via the ADCS bits of the ADCON1 register. There are seven possible clock options:

- Fosc/2
- Fosc/4
- Fosc/8
- Fosc/16
- Fosc/32
- Fosc/64
- FRC (dedicated internal oscillator)

The time to complete one bit conversion is defined as TAD. One full 10-bit conversion requires 11.5 TAD periods as shown in Figure 16-2.

For correct conversion, the appropriate TAD specification must be met. Refer to the A/D conversion requirements in **Section 30.0 “Electrical Specifications”** for more information. Table 16-1 gives examples of appropriate ADC clock selections.

Note: Unless using the FRC, any changes in the system clock frequency will change the ADC clock frequency, which may adversely affect the ADC result.

17.4 Low-Power Voltage State

In order for the DAC module to consume the least amount of power, one of the two voltage reference input sources to the resistor ladder must be disconnected. Either the positive voltage source, (VSOURCE+), or the negative voltage source, (VSOURCE-) can be disabled.

The negative voltage source is disabled by setting the DACLPS bit in the DACCON0 register. Clearing the DACLPS bit in the DACCON0 register disables the positive voltage source.

17.4.1 OUTPUT CLAMPED TO POSITIVE VOLTAGE SOURCE

The DAC output voltage can be set to VSOURCE+ with the least amount of power consumption by performing the following:

- Clearing the DACEN bit in the DACCON0 register.
- Setting the DACLPS bit in the DACCON0 register.
- Configuring the DACPSS bits to the proper positive source.
- Configuring the DACR<4:0> bits to '11111' in the DACCON1 register.

This is also the method used to output the voltage level from the FVR to an output pin. See **Section 17.5 “Operation During Sleep”** for more information.

Reference Figure 17-3 for output clamping examples.

17.4.2 OUTPUT CLAMPED TO NEGATIVE VOLTAGE SOURCE

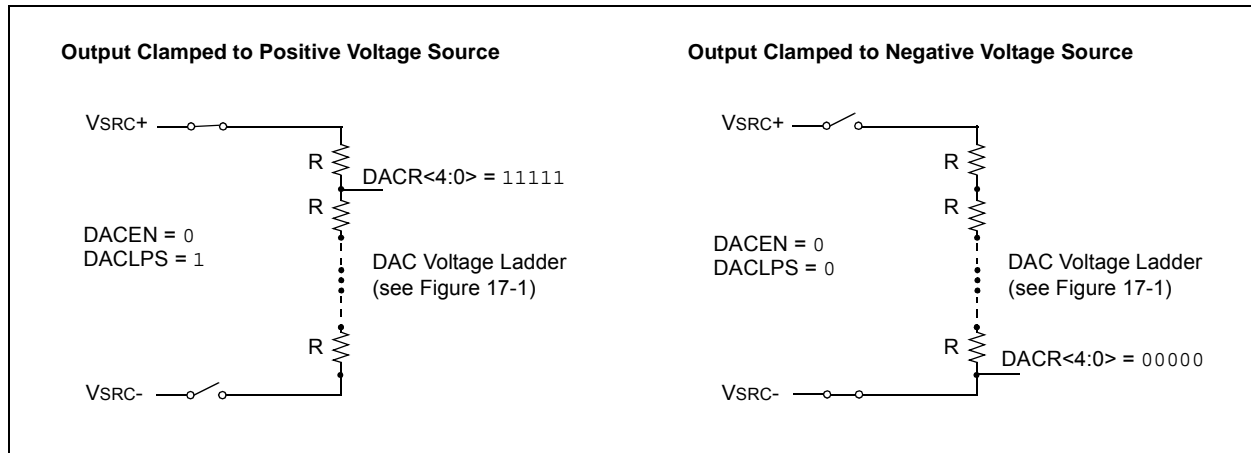
The DAC output voltage can be set to VSOURCE- with the least amount of power consumption by performing the following:

- Clearing the DACEN bit in the DACCON0 register.
- Clearing the DACLPS bit in the DACCON0 register.
- Configuring the DACNSS bits to the proper negative source.
- Configuring the DACR<4:0> bits to '00000' in the DACCON1 register.

This allows the comparator to detect a zero-crossing while not consuming additional current through the DAC module.

Reference Figure 17-3 for output clamping examples.

FIGURE 17-3: OUTPUT VOLTAGE CLAMPING EXAMPLES



17.5 Operation During Sleep

When the device wakes up from Sleep through an interrupt or a Watchdog Timer time-out, the contents of the DACCON0 register are not affected. To minimize current consumption in Sleep mode, the voltage reference should be disabled.

17.6 Effects of a Reset

A device Reset affects the following:

- DAC is disabled.
- DAC output voltage is removed from the DACOUT pin.
- The DACR<4:0> range select bits are cleared.

19.4 Timer1 Gate Operation

The output resulting from a comparator operation can be used as a source for gate control of Timer1. See **Section 21.6 “Timer1 Gate”** for more information. This feature is useful for timing the duration or interval of an analog event.

It is recommended that the comparator output be synchronized to Timer1. This ensures that Timer1 does not increment while a change in the comparator is occurring.

19.4.1 COMPARATOR OUTPUT SYNCHRONIZATION

The output from either comparator, C1 or C2, can be synchronized with Timer1 by setting the CxSYNC bit of the CMxCON0 register.

Once enabled, the comparator output is latched on the falling edge of the Timer1 source clock. If a prescaler is used with Timer1, the comparator output is latched after the prescaling function. To prevent a race condition, the comparator output is latched on the falling edge of the Timer1 clock source and Timer1 increments on the rising edge of its clock source. See the Comparator Block Diagram (Figure 19-2) and the Timer1 Block Diagram (Figure 21-1) for more information.

19.5 Comparator Interrupt

An interrupt can be generated upon a change in the output value of the comparator for each comparator, a rising edge detector and a Falling edge detector are present.

When either edge detector is triggered and its associated enable bit is set (CxINTP and/or CxINTN bits of the CMxCON1 register), the Corresponding Interrupt Flag bit (CxIF bit of the PIR2 register) will be set.

To enable the interrupt, you must set the following bits:

- CxON, CxPOL and CxSP bits of the CMxCON0 register
- CxIE bit of the PIE2 register
- CxINTP bit of the CMxCON1 register (for a rising edge detection)
- CxINTN bit of the CMxCON1 register (for a falling edge detection)
- PEIE and GIE bits of the INTCON register

The associated interrupt flag bit, CxIF bit of the PIR2 register, must be cleared in software. If another edge is detected while this flag is being cleared, the flag will still be set at the end of the sequence.

Note: Although a comparator is disabled, an interrupt can be generated by changing the output polarity with the CxPOL bit of the CMxCON0 register, or by switching the comparator on or off with the CxON bit of the CMxCON0 register.

19.6 Comparator Positive Input Selection

Configuring the CxPCH<1:0> bits of the CMxCON1 register directs an internal voltage reference or an analog pin to the non-inverting input of the comparator:

- CxIN+ analog pin
- DAC_output
- FVR Buffer2
- Vss (Ground)

See **Section 14.0 “Fixed Voltage Reference (FVR)”** for more information on the Fixed Voltage Reference module.

See **Section 17.0 “Digital-to-Analog Converter (DAC) Module”** for more information on the DAC input signal.

Any time the comparator is disabled (CxON = 0), all comparator inputs are disabled.

19.7 Comparator Negative Input Selection

The CxNCH<1:0> bits of the CMxCON0 register direct one of four analog pins to the comparator inverting input.

Note: To use CxIN+ and CxINx- pins as analog input, the appropriate bits must be set in the ANSEL register and the corresponding TRIS bits must also be set to disable the output drivers.

19.8 Comparator Response Time

The comparator output is indeterminate for a period of time after the change of an input source or the selection of a new reference voltage. This period is referred to as the response time. The response time of the comparator differs from the settling time of the voltage reference. Therefore, both of these times must be considered when determining the total response time to a comparator input change. See the Comparator and Voltage Reference Specifications in **Section 30.0 “Electrical Specifications”** for more details.

19.9 Interaction with ECCP Logic

In some devices, a comparator output signal can be used to trigger the auto-shutdown feature found within the ECCP module. When the ECCP auto-shutdown feature is enabled and a comparator output signal is selected as the source, the comparator can be used simultaneously as a general purpose comparator and as the ECCP auto-shutdown source. In addition, the comparator output signal can also be routed to the designated I/O pin. If the ECCP Auto-Restart mode is also enabled, the comparators can be used as a closed loop analog feedback circuit to the ECCP, thereby creating an analog controlled PWM.

Please see section

for more information.

Note: When the comparator module is first initialized the output state is unknown. Upon initialization, the user should verify the output state of the comparator prior to relying on the result, primarily when using the result in connection with other peripheral features, such as the ECCP Auto-Shutdown mode.

19.10 Analog Input Connection Considerations

A simplified circuit for an analog input is shown in Figure 19-3. Since the analog input pins share their connection with a digital input, they have reverse biased ESD protection diodes to V_{DD} and V_{SS} . The analog input, therefore, must be between V_{SS} and V_{DD} .

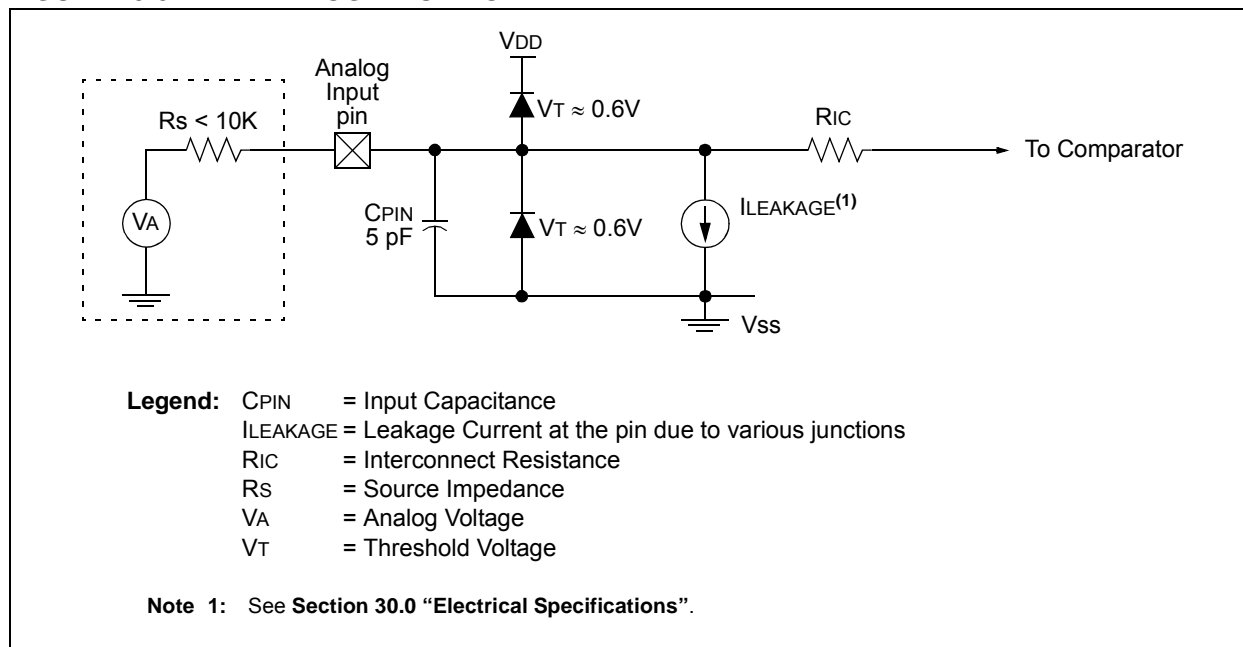
If the input voltage deviates from this range by more than 0.6V in either direction, one of the diodes is forward biased and a latch-up may occur.

A maximum source impedance of 10 k Ω is recommended for the analog sources. Also, any external component connected to an analog input pin, such as a capacitor or a Zener diode, should have very little leakage current to minimize inaccuracies introduced.

Note 1: When reading a PORT register, all pins configured as analog inputs will read as a '0'. Pins configured as digital inputs will convert as an analog input, according to the input specification.

2: Analog levels on any pin defined as a digital input, may cause the input buffer to consume more current than is specified.

FIGURE 19-3: ANALOG INPUT MODEL



PIC16(L)F1825/9

21.1 Timer1 Operation

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

When used with an internal clock source, the module is a timer and increments on every instruction cycle. When used with an external clock source, the module can be used as either a timer or counter and increments on every selected edge of the external source.

Timer1 is enabled by configuring the TMR1ON and TMR1GE bits in the T1CON and T1GCON registers, respectively. Table 21-1 displays the Timer1 enable selections.

TABLE 21-1: TIMER1 ENABLE SELECTIONS

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

21.2 Clock Source Selection

The TMR1CS<1:0> and T1OSCEN bits of the T1CON register are used to select the clock source for Timer1. Table 21-2 displays the clock source selections.

21.2.1 INTERNAL CLOCK SOURCE

When the internal clock source is selected the TMR1H:TMR1L register pair will increment on multiples of Fosc as determined by the Timer1 prescaler.

When the Fosc internal clock source is selected, the Timer1 register value will increment by four counts every instruction clock cycle. Due to this condition, a 2 LSB error in resolution will occur when reading the Timer1 value. To utilize the full resolution of Timer1, an asynchronous input signal must be used to gate the Timer1 clock input.

The following asynchronous sources may be used:

- Asynchronous event on the T1G pin to Timer1 gate
- C1 or C2 comparator input to Timer1 gate

21.2.2 EXTERNAL CLOCK SOURCE

When the external clock source is selected, the Timer1 module may work as a timer or a counter.

When enabled to count, Timer1 is incremented on the rising edge of the external clock input T1CKI or the capacitive sensing oscillator signal. Either of these external clock sources can be synchronized to the microcontroller system clock or they can run asynchronously.

When used as a timer with a clock oscillator, an external 32.768 kHz crystal can be used in conjunction with the dedicated internal oscillator circuit.

Note: In Counter mode, 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 21-2: CLOCK SOURCE SELECTIONS

TMR1CS1	TMR1CS0	T1OSCEN	Clock Source
0	0	x	Instruction Clock (Fosc/4)
0	1	x	System Clock (Fosc)
1	0	0	External Clocking on T1CKI Pin
1	0	0	External Clocking on T1CKI Pin
1	1	x	Capacitive Sensing Oscillator

23.0 DATA SIGNAL MODULATOR

The Data Signal Modulator (DSM) is a peripheral which allows the user to mix a data stream, also known as a modulator signal, with a carrier signal to produce a modulated output.

Both the carrier and the modulator signals are supplied to the DSM module either internally, from the output of a peripheral, or externally through an input pin.

The modulated output signal is generated by performing a logical “AND” operation of both the carrier and modulator signals and then provided to the MDOUT pin.

The carrier signal is comprised of two distinct and separate signals. A carrier high (CARH) signal and a carrier low (CARL) signal. During the time in which the modulator (MOD) signal is in a logic high state, the DSM mixes the carrier high signal with the modulator signal. When the modulator signal is in a logic low state, the DSM mixes the carrier low signal with the modulator signal.

Using this method, the DSM can generate the following types of Key Modulation schemes:

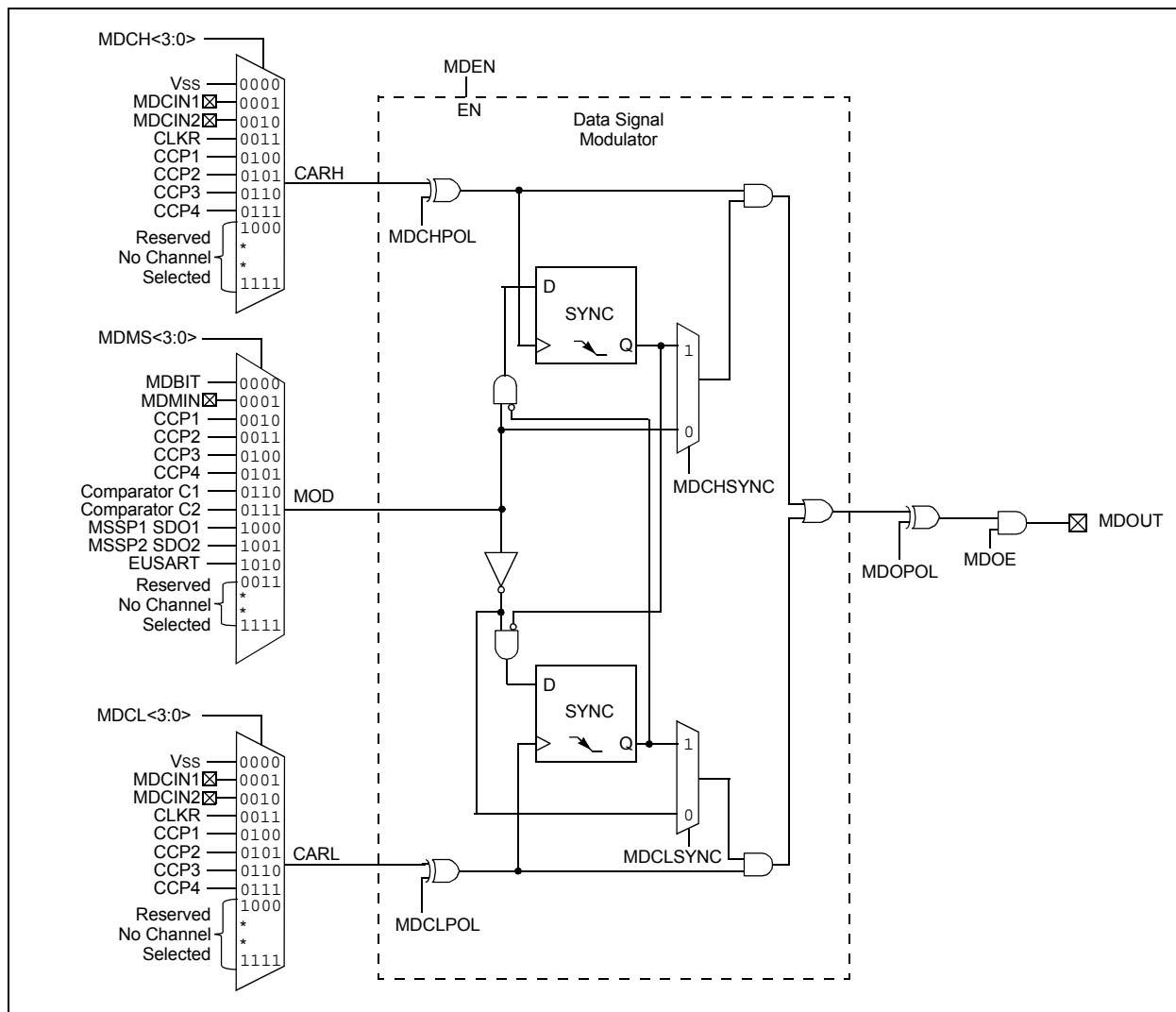
- Frequency-Shift Keying (FSK)
- Phase-Shift Keying (PSK)
- On-Off Keying (OOK)

Additionally, the following features are provided within the DSM module:

- Carrier Synchronization
- Carrier Source Polarity Select
- Carrier Source Pin Disable
- Programmable Modulator Data
- Modulator Source Pin Disable
- Modulated Output Polarity Select
- Slew Rate Control

Figure 23-1 shows a Simplified Block Diagram of the Data Signal Modulator peripheral.

FIGURE 23-1: SIMPLIFIED BLOCK DIAGRAM OF THE DATA SIGNAL MODULATOR



24.4 PWM (Enhanced Mode)

The enhanced PWM function described in this section is available for CCP modules ECCP1 and ECCP2, with any differences between modules noted.

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

- PRx registers
- TxCON registers
- CCPRxL registers
- CCPxCON registers

The ECCP modules have the following additional PWM registers which control Auto-shutdown, Auto-restart, Dead-band Delay and PWM Steering modes:

- CCPxAS registers
- PSTRxCON registers
- PWMxCON registers

The enhanced PWM module can generate the following five PWM Output modes:

- Single PWM
- Half-Bridge PWM
- Full-Bridge PWM, Forward mode
- Full-Bridge PWM, Reverse mode
- Single PWM with PWM Steering mode

To select an Enhanced PWM Output mode, the PxM bits of the CCPxCON register must be configured appropriately.

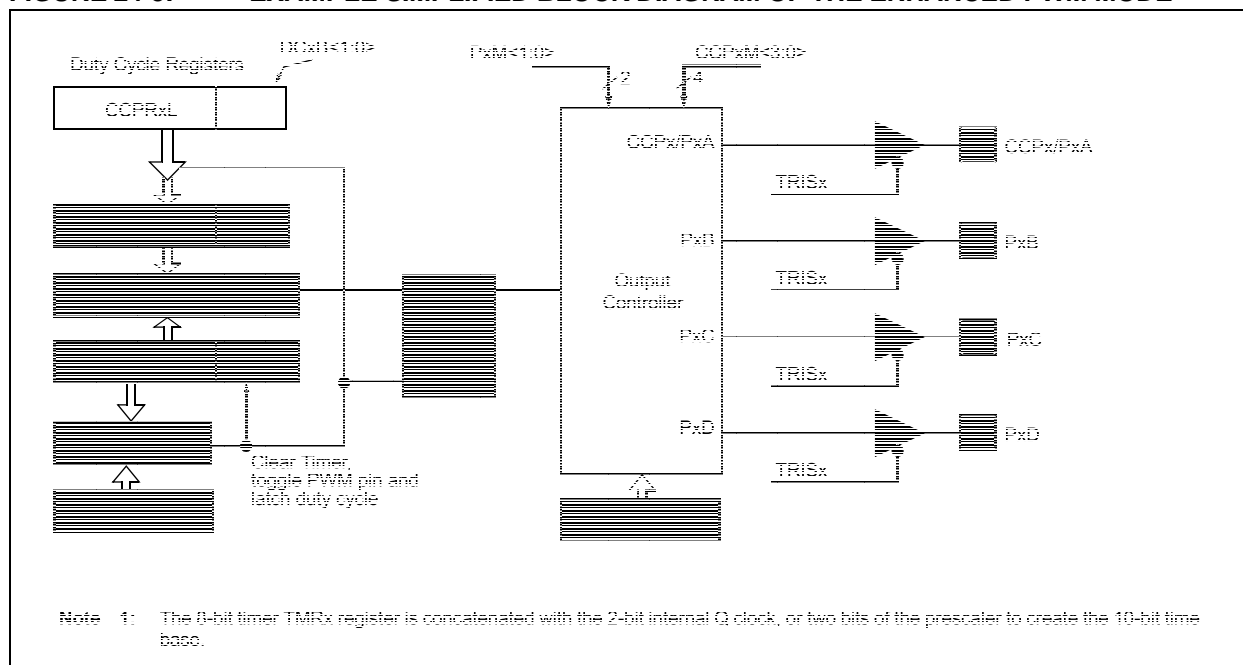
The PWM outputs are multiplexed with I/O pins and are designated PxA, PxB, PxC and PxD. The polarity of the PWM pins is configurable and is selected by setting the CCPxM bits in the CCPxCON register appropriately.

Figure 24-5 shows an example of a simplified block diagram of the Enhanced PWM module.

Figure 24-8 shows the pin assignments for various Enhanced PWM modes.

- Note 1:** The corresponding TRIS bit must be cleared to enable the PWM output on the CCPx pin.
- 2:** Clearing the CCPxCON register will relinquish control of the CCPx pin.
- 3:** Any pin not used in the enhanced PWM mode is available for alternate pin functions, if applicable.
- 4:** To prevent the generation of an incomplete waveform when the PWM is first enabled, the ECCP module waits until the start of a new PWM period before generating a PWM signal.

FIGURE 24-5: EXAMPLE SIMPLIFIED BLOCK DIAGRAM OF THE ENHANCED PWM MODE



PIC16(L)F1825/9

The I²C interface supports the following modes and features:

- Master mode
- Slave mode
- Byte NACKing (Slave mode)
- Limited Multi-master support
- 7-bit and 10-bit addressing
- Start and Stop interrupts
- Interrupt masking
- Clock stretching
- Bus collision detection
- General call address matching
- Address masking
- Address Hold and Data Hold modes
- Selectable SDAx hold times

Figure 25-2 is a block diagram of the I²C interface module in Master mode. Figure 25-3 is a diagram of the I²C interface module in Slave mode.

The PIC16F1829 has two MSSP modules, MSSP1 and MSSP2, each module operating independently from the other.

Note 1: In devices with more than one MSSP module, it is very important to pay close attention to SSPxCONx register names. SSP1CON1 and SSP1CON2 registers control different operational aspects of the same module, while SSP1CON1 and SSP2CON1 control the same features for two different modules.

2: Throughout this section, generic references to an MSSP module in any of its operating modes may be interpreted as being equally applicable to MSSP1 or MSSP2. register names, module I/O signals, and bit names may use the generic designator 'x' to indicate the use of a numeral to distinguish a particular module when required.

FIGURE 25-2: MSSPx BLOCK DIAGRAM (I²C™ MASTER MODE)

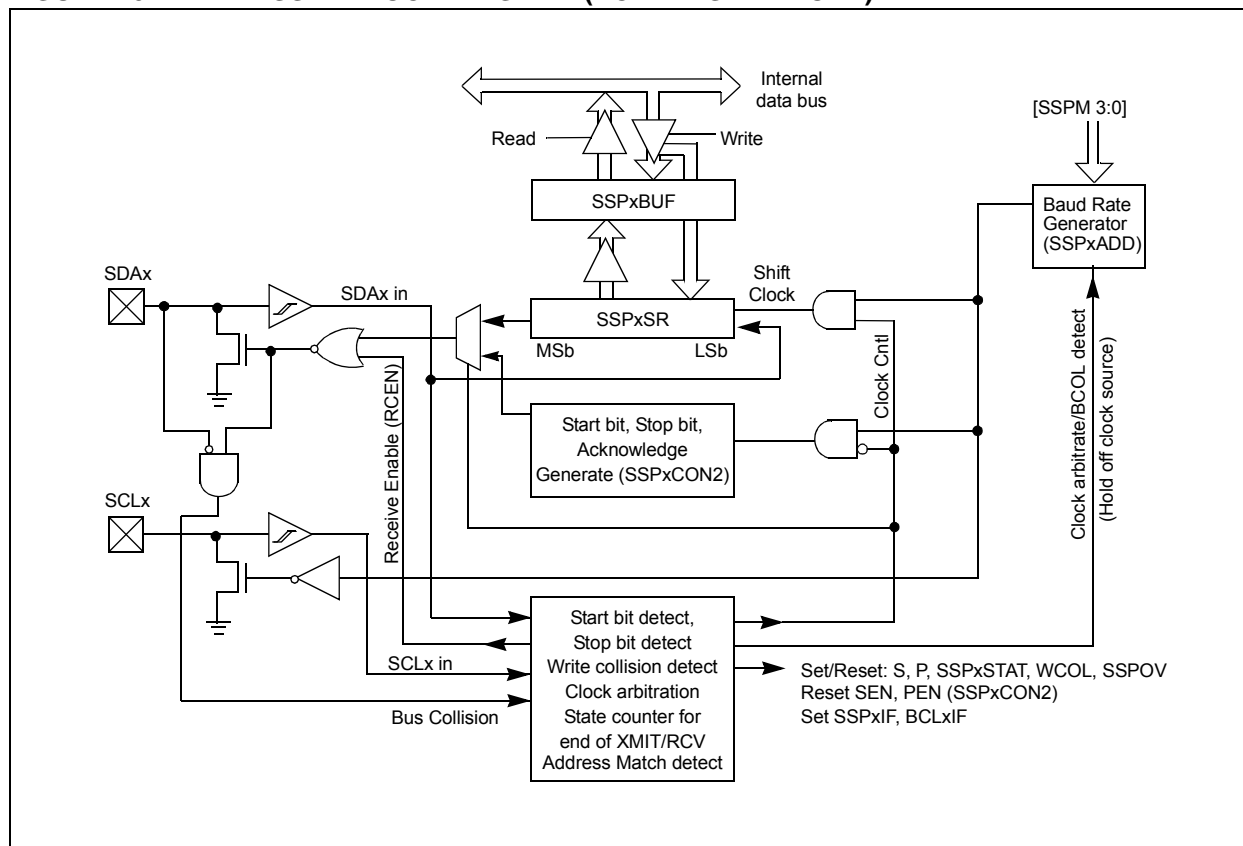


FIGURE 25-17: I²C SLAVE, 7-BIT ADDRESS, RECEPTION (SEN = 1, AHEN = 1, DHEN = 1)

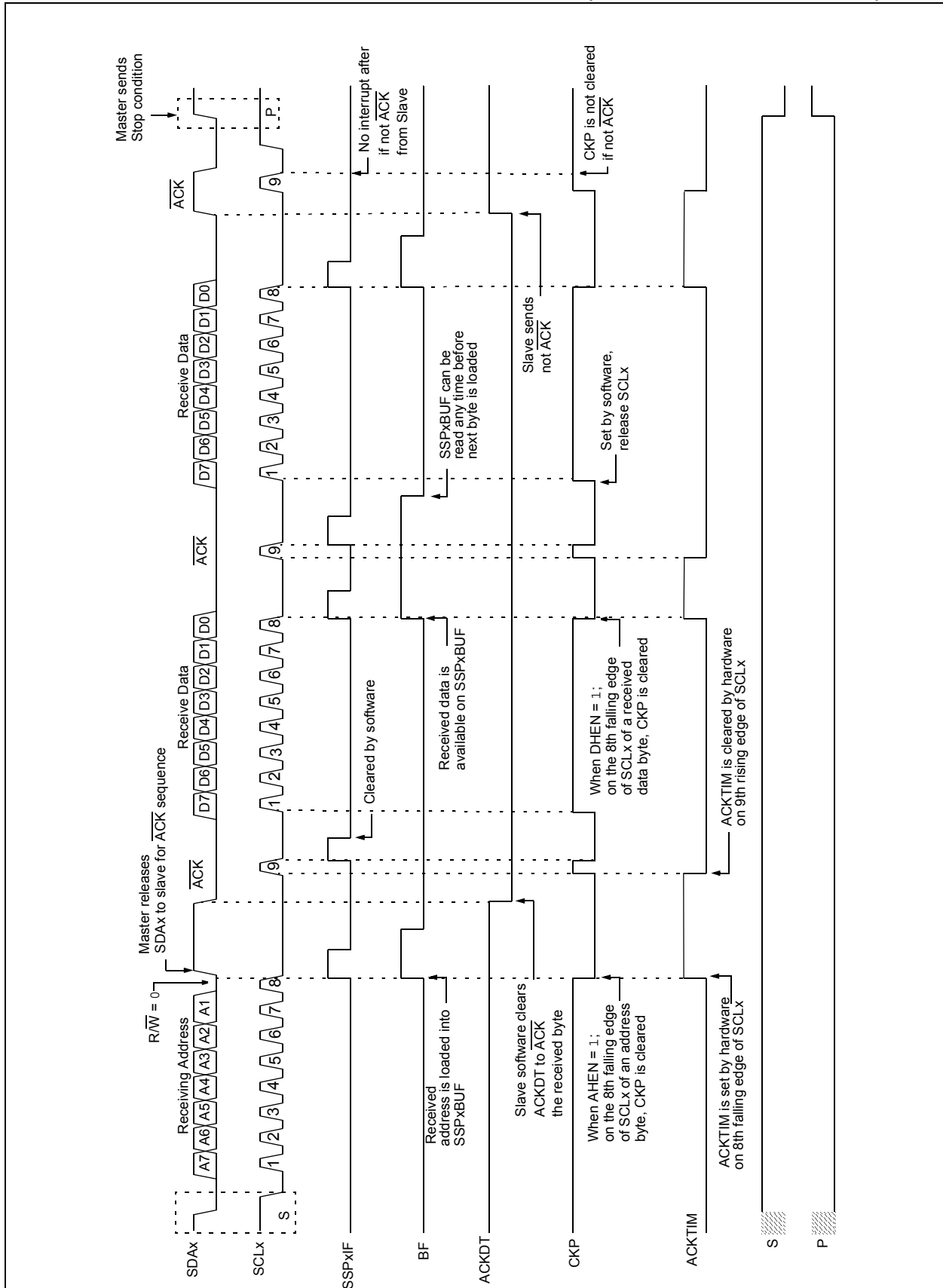


FIGURE 25-29: I²C MASTER MODE WAVEFORM (RECEPTION, 7-BIT ADDRESS)

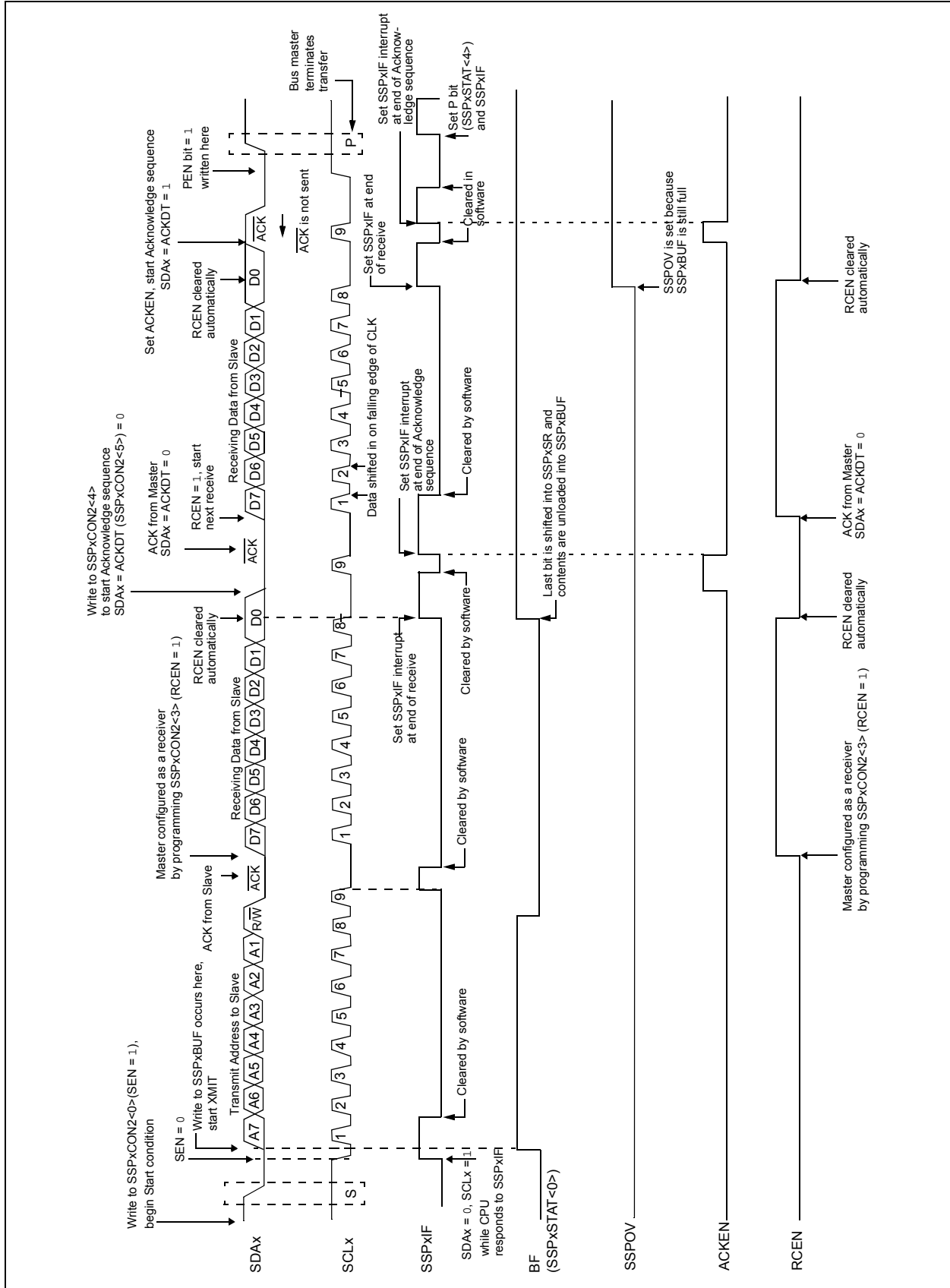


TABLE 26-2: SUMMARY OF REGISTERS ASSOCIATED WITH ASYNCHRONOUS RECEPTION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON0	RXDTSEL	SDO1SEL ⁽²⁾	SS1SEL ⁽²⁾	—	T1GSEL	TXCKSEL	—	—	118
BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	292
INLVLA ⁽³⁾	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	124
INVLVB ⁽¹⁾	INVLVB7	INVLVB6	INVLVB5	INVLVB4	—	—	—	—	129
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	135
INTCON	GIE	PEIE	TMR0IE	INTE	IOCFE	TMR0IF	INTF	IOCF	87
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	88
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	92
RCREG	EUSART Receive Data Register								286*
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	291
SPBRGL	SPBRG<7:0>								293*
SPBRGH	SPBRG<15:8>								293*
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	122
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	128
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	133
TXSTA	CSRC	TX9	TXEN	SYNC	SEnDB	BRGH	TRMT	TX9D	290

Legend: — Unimplemented location, read as '0'. Shaded cells are not used for asynchronous reception.

* Page provides register information.

Note 1: PIC16(L)F1829 only.
2: PIC16(L)F1825 only.
3: Unshaded cells apply to PIC16(L)F1825 only.

PIC16(L)F1825/9

FIGURE 30-22: PIC16F1825/9 VOLTAGE FREQUENCY GRAPH, $-40^{\circ}\text{C} \leq T_A \leq +150^{\circ}\text{C}$

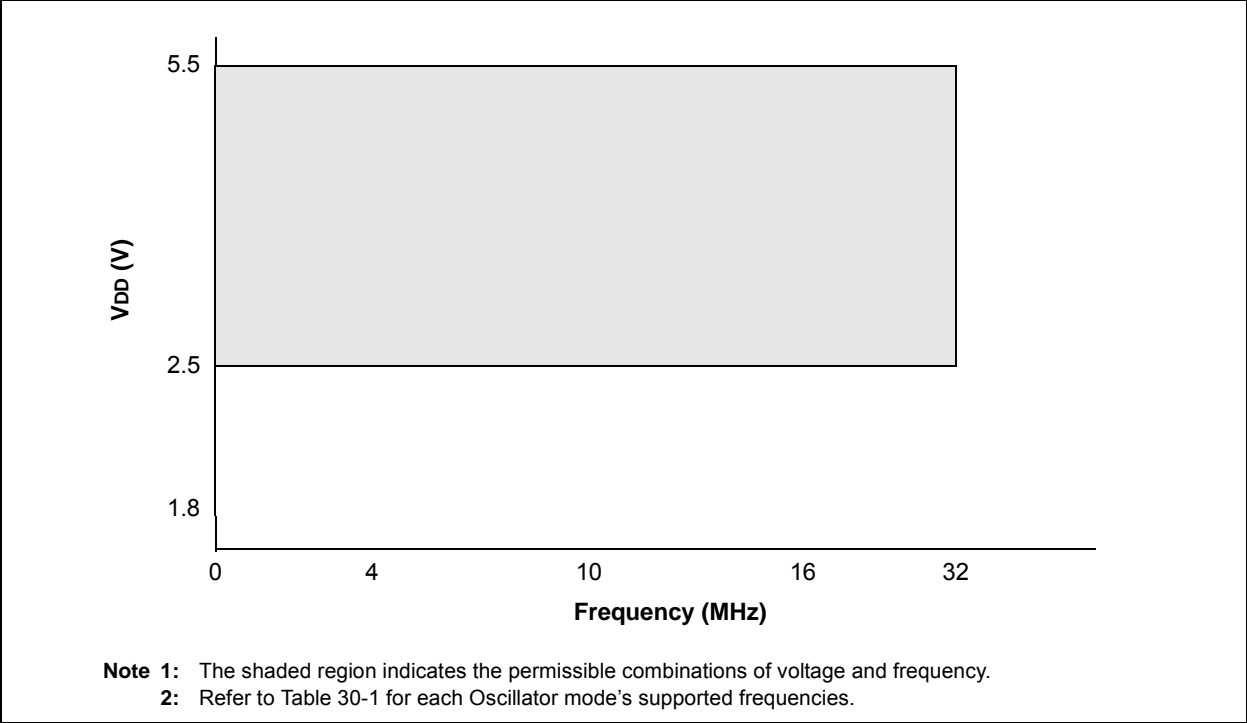


FIGURE 30-23: HFINTOSC FREQUENCY ACCURACY OVER DEVICE V_{DD} AND TEMPERATURE

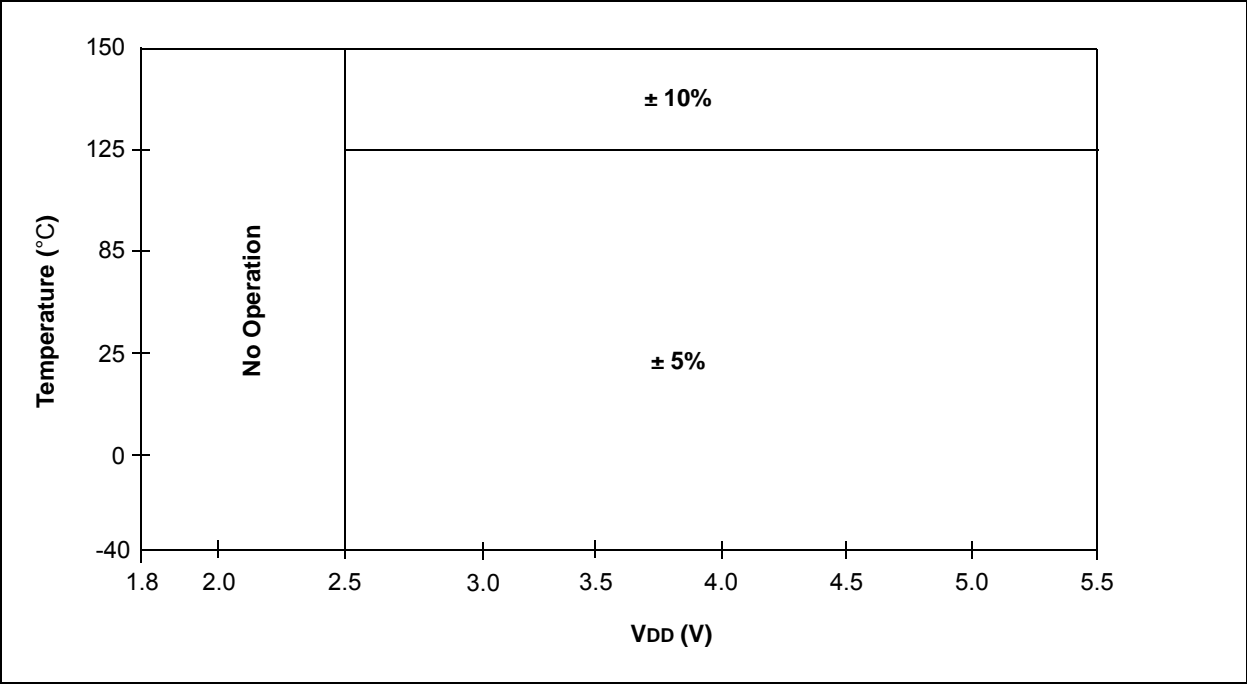


FIGURE 31-41: V_{OH} vs. I_{OH} OVER TEMPERATURE ($V_{DD} = 5.0V$), PIC16F1825/9 ONLY

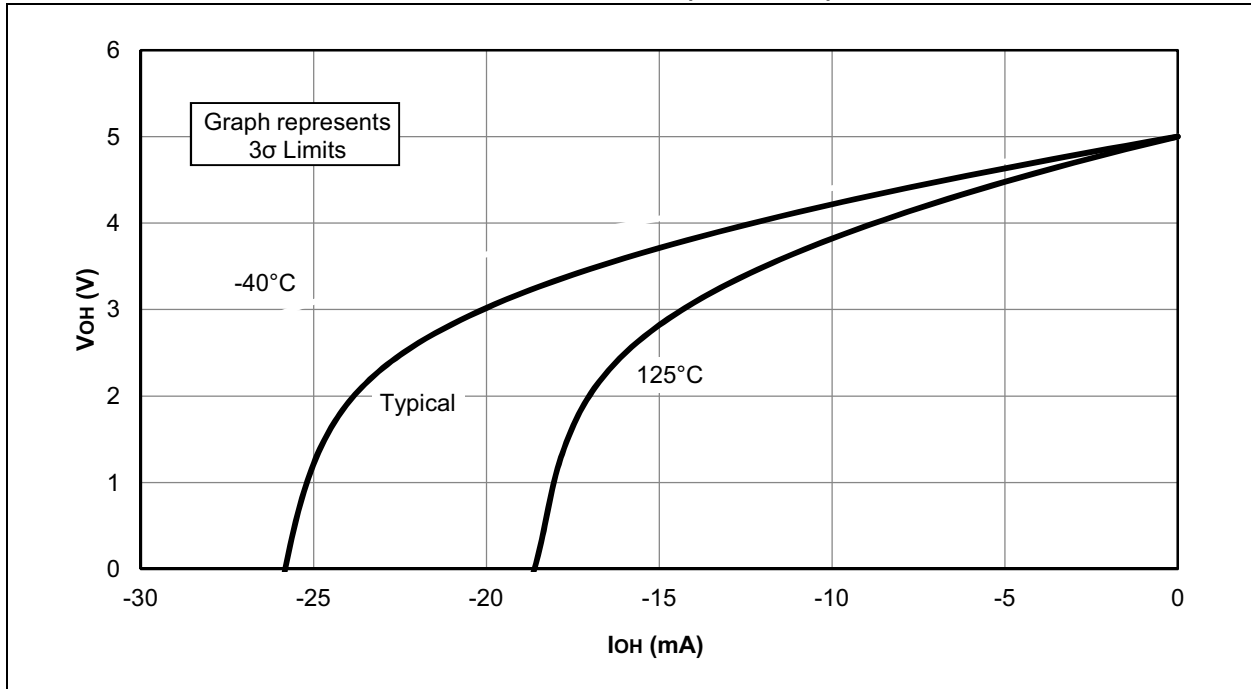
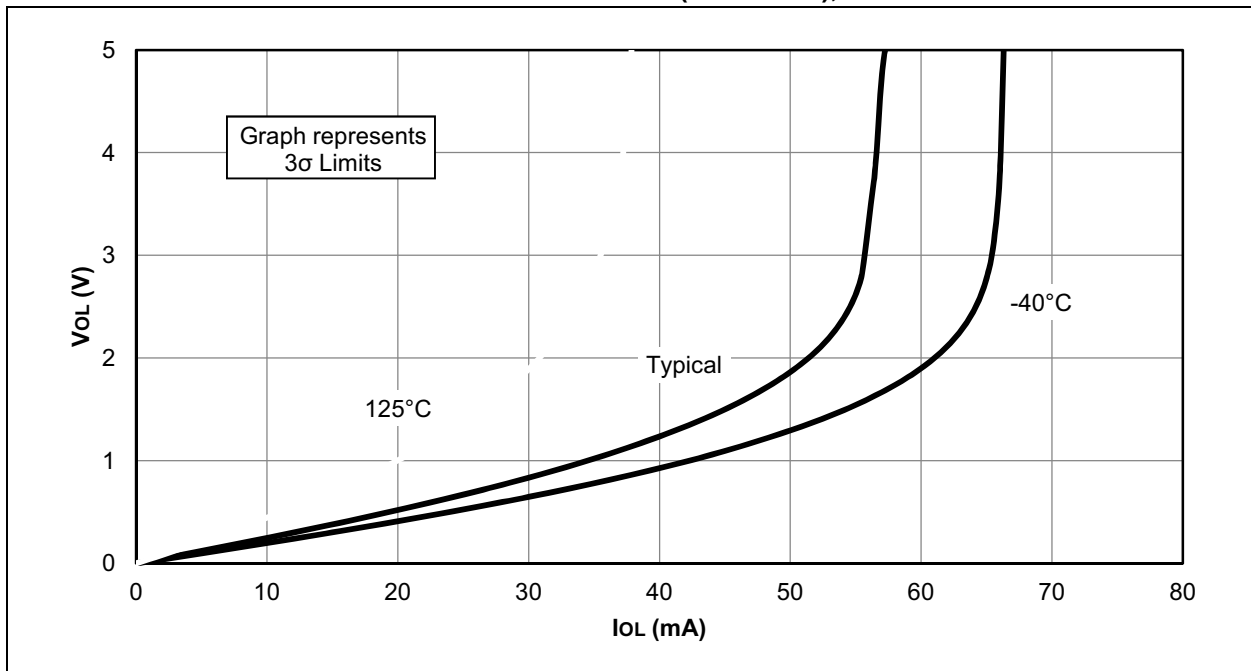


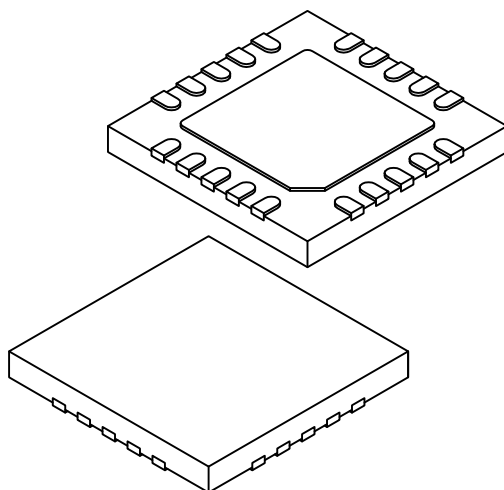
FIGURE 31-42: V_{OL} vs. I_{OL} OVER TEMPERATURE ($V_{DD} = 5.0V$), PIC16F1825/9 ONLY



PIC16(L)F1825/9

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>



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Terminals	N	20		
Pitch	e	0.50 BSC		
Overall Height	A	0.45	0.50	0.55
Standoff	A1	0.00	0.02	0.05
Terminal Thickness	A3	0.127 REF		
Overall Width	E	4.00 BSC		
Exposed Pad Width	E2	2.60	2.70	2.80
Overall Length	D	4.00 BSC		
Exposed Pad Length	D2	2.60	2.70	2.80
Terminal Width	b	0.20	0.25	0.30
Terminal Length	L	0.30	0.40	0.50
Terminal-to-Exposed-Pad	K	0.20	-	-

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Package is saw singulated
3. Dimensioning and tolerancing per ASME Y14.5M

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

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-255A Sheet 2 of 2