



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	40MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, HLVD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	32KB (16K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	1.5K x 8
Voltage - Supply (Vcc/Vdd)	4.2V ~ 5.5V
Data Converters	A/D 13x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f4520-i-ml

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

1.0 DEVICE OVERVIEW

This document contains device-specific information for the following devices:

- PIC18F2420 PIC18LF2420
- PIC18F2520 PIC18LF2520
- PIC18F4420 PIC18LF4420
- PIC18F4520
 PIC18LF4520

This family offers the advantages of all PIC18 microcontrollers – namely, high computational performance at an economical price – with the addition of high-endurance, Enhanced Flash program memory. On top of these features, the PIC18F2420/2520/4420/4520 family introduces design enhancements that make these microcontrollers a logical choice for many high-performance, power sensitive applications.

1.1 New Core Features

1.1.1 nanoWatt TECHNOLOGY

All of the devices in the PIC18F2420/2520/4420/4520 family incorporate a range of features that can significantly reduce power consumption during operation. Key items include:

- Alternate Run Modes: By clocking the controller from the Timer1 source or the internal oscillator block, power consumption during code execution can be reduced by as much as 90%.
- **Multiple Idle Modes:** The controller can also run with its CPU core disabled but the peripherals still active. In these states, power consumption can be reduced even further, to as little as 4% of normal operation requirements.
- On-the-Fly Mode Switching: The powermanaged modes are invoked by user code during operation, allowing the user to incorporate power-saving ideas into their application's software design.
- Low Consumption in Key Modules: The power requirements for both Timer1 and the Watchdog Timer are minimized. See Section 26.0 "Electrical Characteristics" for values.

1.1.2 MULTIPLE OSCILLATOR OPTIONS AND FEATURES

All of the devices in the PIC18F2420/2520/4420/4520 family offer ten different oscillator options, allowing users a wide range of choices in developing application hardware. These include:

- Four Crystal modes, using crystals or ceramic resonators
- Two External Clock modes, offering the option of using two pins (oscillator input and a divide-by-4 clock output) or one pin (oscillator input, with the second pin reassigned as general I/O)
- Two External RC Oscillator modes with the same pin options as the External Clock modes
- An internal oscillator block which provides an 8 MHz clock and an INTRC source (approximately 31 kHz), as well as a range of 6 user-selectable clock frequencies, between 125 kHz to 4 MHz, for a total of 8 clock frequencies. This option frees the two oscillator pins for use as additional general purpose I/O.
- A Phase Lock Loop (PLL) frequency multiplier, available to both the High-Speed Crystal and Internal Oscillator modes, which allows clock speeds of up to 40 MHz. Used with the internal oscillator, the PLL gives users a complete selection of clock speeds, from 31 kHz to 32 MHz – all without using an external crystal or clock circuit.

Besides its availability as a clock source, the internal oscillator block provides a stable reference source that gives the family additional features for robust operation:

- Fail-Safe Clock Monitor: This option constantly monitors the main clock source against a reference signal provided by the internal oscillator. If a clock failure occurs, the controller is switched to the internal oscillator block, allowing for continued low-speed operation or a safe application shutdown.
- **Two-Speed Start-up:** This option allows the internal oscillator to serve as the clock source from Power-on Reset, or wake-up from Sleep mode, until the primary clock source is available.

2.7 Clock Sources and Oscillator Switching

Like previous PIC18 devices, the PIC18F2420/2520/ 4420/4520 family includes a feature that allows the device clock source to be switched from the main oscillator to an alternate low-frequency clock source. PIC18F2420/2520/4420/4520 devices offer two alternate clock sources. When an alternate clock source is enabled, the various power-managed operating modes are available.

Essentially, there are three clock sources for these devices:

- Primary oscillators
- · Secondary oscillators
- · Internal oscillator block

The **primary oscillators** include the External Crystal and Resonator modes, the External RC modes, the External Clock modes and the internal oscillator block. The particular mode is defined by the FOSC<3:0> Configuration bits. The details of these modes are covered earlier in this chapter. The **secondary oscillators** are those external sources not connected to the OSC1 or OSC2 pins. These sources may continue to operate even after the controller is placed in a power-managed mode.

PIC18F2420/2520/4420/4520 devices offer the Timer1 oscillator as a secondary oscillator. This oscillator, in all power-managed modes, is often the time base for functions such as a Real-Time Clock (RTC).

Most often, a 32.768 kHz watch crystal is connected between the RC0/T1OSO/T13CKI and RC1/T1OSI pins. Like the LP Oscillator mode circuit, loading capacitors are also connected from each pin to ground.

The Timer1 oscillator is discussed in greater detail in **Section 12.3 "Timer1 Oscillator"**.

In addition to being a primary clock source, the **internal oscillator block** is available as a power-managed mode clock source. The INTRC source is also used as the clock source for several special features, such as the WDT and Fail-Safe Clock Monitor.

The clock sources for the PIC18F2420/2520/4420/4520 devices are shown in Figure 2-8. See **Section 23.0 "Special Features of the CPU"** for Configuration register details.

FIGURE 2-8: PIC18F2420/2520/4420/4520 CLOCK DIAGRAM



5.1.2.2 Return Stack Pointer (STKPTR)

The STKPTR register (Register 5-1) contains the Stack Pointer value, the STKFUL (Stack Full) status bit and the STKUNF (Stack Underflow) status bits. The value of the Stack Pointer can be 0 through 31. The Stack Pointer increments before values are pushed onto the stack and decrements after values are popped off the stack. On Reset, the Stack Pointer value will be zero. The user may read and write the Stack Pointer value. This feature can be used by a Real-Time Operating System (RTOS) for return stack maintenance.

After the PC is pushed onto the stack 31 times (without popping any values off the stack), the STKFUL bit is set. The STKFUL bit is cleared by software or by a POR.

The action that takes place when the stack becomes full depends on the state of the STVREN (Stack Overflow Reset Enable) Configuration bit. (Refer to **Section 23.1 "Configuration Bits"** for a description of the device Configuration bits.) If STVREN is set (default), the 31st push will push the (PC + 2) value onto the stack, set the STKFUL bit and reset the device. The STKFUL bit will remain set and the Stack Pointer will be set to zero.

If STVREN is cleared, the STKFUL bit will be set on the 31st push and the Stack Pointer will increment to 31. Any additional pushes will not overwrite the 31st push and STKPTR will remain at 31.

When the stack has been popped enough times to unload the stack, the next pop will return a value of zero to the PC and sets the STKUNF bit, while the Stack Pointer remains at zero. The STKUNF bit will remain set until cleared by software or until a POR occurs.

Note:	Returning a value of zero to the PC on an				
	underflow has the effect of vectoring the				
	program to the Reset vector, where the				
	stack conditions can be verified and				
	appropriate actions can be taken. This is				
	not the same as a Reset, as the contents				
	of the SFRs are not affected.				

5.1.2.3 PUSH and POP Instructions

Since the Top-of-Stack is readable and writable, the ability to push values onto the stack and pull values off the stack without disturbing normal program execution is a desirable feature. The PIC18 instruction set includes two instructions, PUSH and POP, that permit the TOS to be manipulated under software control. TOSU, TOSH and TOSL can be modified to place data or a return address on the stack.

The PUSH instruction places the current PC value onto the stack. This increments the Stack Pointer and loads the current PC value onto the stack.

The POP instruction discards the current TOS by decrementing the Stack Pointer. The previous value pushed onto the stack then becomes the TOS value.

REGISTER 5-1: STKPTR: STACK POINTER REGISTER

R/C-0	R/C-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STKFUL ⁽¹⁾	STKUNF ⁽¹⁾	—	SP4	SP3	SP2	SP1	SP0
bit 7							bit 0
Legend:		C = Clearable	bit				
R = Readable	e bit	W = Writable b	bit	U = Unimplem	ented bit, read	l as '0'	
-n = Value at	POR	'1' = Bit is set		'0' = Bit is clea	ared	x = Bit is unkr	nown
bit 7	bit 7 STKELLI - Stock Full Elog bit ⁽¹⁾						
	1 = Stack became full or overflowed 0 = Stack has not become full or overflowed						
bit 6	STKUNF: Sta	ck Underflow F	lag bit ⁽¹⁾				
	 1 = Stack underflow occurred 0 = Stack underflow did not occur 						
bit 5	5 Unimplemented: Read as '0'						
bit 4-0	bit 4-0 SP<4:0>: Stack Pointer Location bits						
Note 1: Bit 7 and bit 6 are cleared by user software or by a POR.							

1	-			1			\ -		, ,	T
File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Details on page:
SPBRGH	EUSART Bau		0000 0000	51, 206						
SPBRG	EUSART Baud Rate Generator Register Low Byte								0000 0000	51, 206
RCREG	EUSART Red	eive Register							0000 0000	51, 213
TXREG	EUSART Tra	nsmit Register	-						0000 0000	51, 211
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	0000 0010	51, 202
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	51, 203
EEADR	EEPROM Ad	dress Registe	r						0000 0000	51, 74, 83
EEDATA	EEPROM Da	ta Register							0000 0000	51, 74, 83
EECON2	EEPROM Co	ntrol Register	2 (not a physi	cal register)					0000 0000	51, 74, 83
EECON1	EEPGD	CFGS	—	FREE	WRERR	WREN	WR	RD	xx-0 x000	51, 75, 84
IPR2	OSCFIP	CMIP	—	EEIP	BCLIP	HLVDIP	TMR3IP	CCP2IP	11-1 1111	52, 101
PIR2	OSCFIF	CMIF	—	EEIF	BCLIF	HLVDIF	TMR3IF	CCP2IF	00-0 0000	52, 97
PIE2	OSCFIE	CMIE	—	EEIE	BCLIE	HLVDIE	TMR3IE	CCP2IE	00-0 0000	52, 99
IPR1	PSPIP ⁽²⁾	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	1111 1111	52, 100
PIR1	PSPIF ⁽²⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	52, 96
PIE1	PSPIE ⁽²⁾	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	52, 98
OSCTUNE	INTSRC	PLLEN ⁽³⁾	—	TUN4	TUN3	TUN2	TUN1	TUN0	0q-0 0000	27, 52
TRISE ⁽²⁾	IBF	OBF	IBOV	PSPMODE		TRISE2	TRISE1	TRISE0	0000 -111	52, 118
TRISD ⁽²⁾	PORTD Data	Direction Reg	jister						1111 1111	52, 114
TRISC	PORTC Data	Direction Reg	jister						1111 1111	52, 111
TRISB	PORTB Data	Direction Reg	ister						1111 1111	52, 108
TRISA	TRISA7 ⁽⁵⁾	TRISA6 ⁽⁵⁾	PORTA Data	Direction Reg	ister				1111 1111	52, 105
LATE ⁽²⁾	-	_	—	—	_	PORTE Data (Read and W	Latch Registe rite to Data La	er itch)	xxx	52, 117
LATD ⁽²⁾	PORTD Data	Latch Registe	er (Read and V	Vrite to Data L	atch)				XXXX XXXX	52, 114
LATC	PORTC Data	Latch Registe	er (Read and V	Vrite to Data L	atch)				XXXX XXXX	52, 111
LATB	PORTB Data Latch Register (Read and Write to Data Latch)								xxxx xxxx	52, 108
LATA	LATA7 ⁽⁵⁾	LATA6 ⁽⁵⁾	PORTA Data	Latch Registe	r (Read and V	Vrite to Data La	atch)		xxxx xxxx	52, 105
PORTE	_	_	_	_	RE3 ⁽⁴⁾	RE2 ⁽²⁾	RE1 ⁽²⁾	RE0 ⁽²⁾	xxxx	52, 117
PORTD ⁽²⁾	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx xxxx	52, 114
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	52, 111
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	52, 108
PORTA	RA7 ⁽⁵⁾	RA6 ⁽⁵⁾	RA5	RA4	RA3	RA2	RA1	RA0	xx0x 0000	52, 105

TABLE 5-2:PIC18F2420/2520/4420/4520 REGISTER FILE SUMMARY (CONTINUED)

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

Note 1: The SBOREN bit is only available when the BOREN<1:0> Configuration bits = 01; otherwise, it is disabled and reads as '0'. See Section 4.4 "Brown-out Reset (BOR)".

2: These registers and/or bits are not implemented on 28-pin devices and are read as '0'. Reset values are shown for 40/44-pin devices; individual unimplemented bits should be interpreted as '-'.

3: The PLLEN bit is only available in specific oscillator configurations; otherwise, it is disabled and reads as '0'. See Section 2.6.4 "PLL in INTOSC Modes".

4: The RE3 bit is only available when Master Clear Reset is disabled (MCLRE Configuration bit = 0); otherwise, RE3 reads as '0'. This bit is read-only.

5: RA6/RA7 and their associated latch and direction bits are individually configured as port pins based on various primary oscillator modes. When disabled, these bits read as '0'.

6.3 Reading the Flash Program Memory

The TBLRD instruction is used to retrieve data from program memory and places it into data RAM. Table reads from program memory are performed one byte at a time.

TBLPTR points to a byte address in program space. Executing TBLRD places the byte pointed to into TABLAT. In addition, TBLPTR can be modified automatically for the next table read operation.

The internal program memory is typically organized by words. The Least Significant bit of the address selects between the high and low bytes of the word. Figure 6-4 shows the interface between the internal program memory and the TABLAT.

FIGURE 6-4: READS FROM FLASH PROGRAM MEMORY



EXAMPLE 6-1: READING A FLASH PROGRAM MEMORY WORD

	MOVLW MOVWF MOVLW MOVWF MOVLW MOVWF	CODE_ADDR_UPPER TBLPTRU CODE_ADDR_HIGH TBLPTRH CODE_ADDR_LOW TBLPTRL	;;	: Load TBLPTR with the base : address of the word
READ_WORD				
	TBLRD*-	÷	;	; read into TABLAT and increment
	MOVF	TABLAT, W	;	; get data
	MOVWF	WORD_EVEN		
	TBLRD*-	+	;	; read into TABLAT and increment
	MOVFW	TABLAT, W	;	; get data
	MOVF	WORD_ODD		

NOTES:

9.2 PIR Registers

The PIR registers contain the individual flag bits for the peripheral interrupts. Due to the number of peripheral interrupt sources, there are two Peripheral Interrupt Request Flag registers (PIR1 and PIR2).

- Note 1: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Interrupt Enable bit, GIE (INTCON<7>).
 - 2: User software should ensure the appropriate interrupt flag bits are cleared prior to enabling an interrupt and after servicing that interrupt.

REGISTER 9-4: PIR1: PERIPHERAL INTERRUPT REQUEST (FLAG) REGISTER 1

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
PSPIF ⁽¹⁾	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit 7							bit 0

Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit,	, read as '0'
-n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

. .. (1)

. ---

bit /	PSPIF: Parallel Slave Port Read/Write Interrupt Flag bit'
	 1 = A read or a write operation has taken place (must be cleared in software) 0 = No read or write has occurred
bit 6	ADIF: A/D Converter Interrupt Flag bit
	1 = An A/D conversion completed (must be cleared in software)0 = The A/D conversion is not complete
bit 5	RCIF: EUSART Receive Interrupt Flag bit
	 1 = The EUSART receive buffer, RCREG, is full (cleared when RCREG is read) 0 = The EUSART receive buffer is empty
bit 4	TXIF: EUSART Transmit Interrupt Flag bit
	 1 = The EUSART transmit buffer, TXREG, is empty (cleared when TXREG is written) 0 = The EUSART transmit buffer is full
bit 3	SSPIF: Master Synchronous Serial Port Interrupt Flag bit
	1 = The transmission/reception is complete (must be cleared in software)0 = Waiting to transmit/receive
bit 2	CCP1IF: CCP1 Interrupt Flag bit
	<u>Capture mode:</u> 1 = A TMR1 register capture occurred (must be cleared in software) 0 = No TMR1 register capture occurred
	<u>Compare mode:</u> 1 = A TMR1 register compare match occurred (must be cleared in software) 0 = No TMR1 register compare match occurred <u>PWM mode:</u> Unused in this mode.
bit 1	TMR2IF: TMR2 to PR2 Match Interrupt Flag bit
	1 = TMR2 to PR2 match occurred (must be cleared in software)0 = No TMR2 to PR2 match occurred
bit 0	<pre>TMR1IF: TMR1 Overflow Interrupt Flag bit 1 = TMR1 register overflowed (must be cleared in software) 0 = TMR1 register did not overflow</pre>

Note 1: This bit is unimplemented on 28-pin devices and will read as '0'.

16.4.4 HALF-BRIDGE MODE

In the Half-Bridge Output mode, two pins are used as outputs to drive push-pull loads. The PWM output signal is output on the P1A pin, while the complementary PWM output signal is output on the P1B pin (Figure 16-4). This mode can be used for half-bridge applications, as shown in Figure 16-5, or for full-bridge applications where four power switches are being modulated with two PWM signals.

In Half-Bridge Output mode, the programmable deadband delay can be used to prevent shoot-through current in half-bridge power devices. The value of bits, PDC<6:0>, sets the number of instruction cycles before the output is driven active. If the value is greater than the duty cycle, the corresponding output remains inactive during the entire cycle. See **Section 16.4.6 "Programmable Dead-Band Delay"** for more details of the dead-band delay operations.

Since the P1A and P1B outputs are multiplexed with the PORTC<2> and PORTD<5> data latches, the TRISC<2> and TRISD<5> bits must be cleared to configure P1A and P1B as outputs.

FIGURE 16-4: HALF-BRIDGE PWM OUTPUT



FIGURE 16-5: EXAMPLES OF HALF-BRIDGE OUTPUT MODE APPLICATIONS



18.0 ENHANCED UNIVERSAL SYNCHRONOUS ASYNCHRONOUS RECEIVER TRANSMITTER (EUSART)

The Enhanced Universal Synchronous Asynchronous Receiver Transmitter (EUSART) module is one of the two serial I/O modules. (Generically, the USART is also known as a Serial Communications Interface or SCI.) The EUSART can be configured as a full-duplex asynchronous system that can communicate with peripheral devices, such as CRT terminals and personal computers. It can also be configured as a halfduplex, synchronous system that can communicate with peripheral devices, such as A/D or D/A integrated circuits, serial EEPROMs, etc.

The Enhanced USART module implements additional features, including automatic baud rate detection and calibration, automatic wake-up on Sync Break reception and 12-bit Break character transmit. These make it ideally suited for use in Local Interconnect Network bus (LIN bus) systems.

The EUSART can be configured in the following modes:

- Asynchronous (full duplex) with:
 - Auto-wake-up on character reception
 - Auto-baud calibration
 - 12-bit Break character transmission
- Synchronous Master (half duplex) with selectable clock polarity
- Synchronous Slave (half duplex) with selectable clock polarity

The pins of the Enhanced USART are multiplexed with PORTC. In order to configure RC6/TX/CK and RC7/RX/DT as an EUSART:

- bit SPEN (RCSTA<7>) must be set (= 1)
- bit TRISC<7> must be set (= 1)
- bit TRISC<6> must be set (= 1)

Note:	The EUSART control will automatically
	reconfigure the pin from input to output as
	needed.

The operation of the Enhanced USART module is controlled through three registers:

- · Transmit Status and Control (TXSTA)
- · Receive Status and Control (RCSTA)
- Baud Rate Control (BAUDCON)

These are detailed on the following pages in Register 18-1, Register 18-2 and Register 18-3, respectively.

19.2 Selecting and Configuring Acquisition Time

The ADCON2 register allows the user to select an acquisition time that occurs each time the GO/DONE bit is set. It also gives users the option to use an automatically determined acquisition time.

Acquisition time may be set with the ACQT<2:0> bits (ADCON2<5:3>), which provides a range of 2 to 20 TAD. When the GO/DONE bit is set, the A/D module continues to sample the input for the selected acquisition time, then automatically begins a conversion. Since the acquisition time is programmed, there may be no need to wait for an acquisition time between selecting a channel and setting the GO/DONE bit.

Manual acquisition is selected when ACQT<2:0> = 0.00. When the GO/DONE bit is set, sampling is stopped and a conversion begins. The user is responsible for ensuring the required acquisition time has passed between selecting the desired input channel and setting the GO/DONE bit. This option is also the default Reset state of the ACQT<2:0> bits and is compatible with devices that do not offer programmable acquisition times.

In either case, when the conversion is completed, the GO/DONE bit is cleared, the ADIF flag is set and the A/D begins sampling the currently selected channel again. If an acquisition time is programmed, there is nothing to indicate if the acquisition time has ended or if the conversion has begun.

19.3 Selecting the A/D Conversion Clock

The A/D conversion time per bit is defined as TAD. The A/D conversion requires 11 TAD per 10-bit conversion. The source of the A/D conversion clock is software selectable. There are seven possible options for TAD:

- 2 Tosc
- 4 Tosc
- 8 Tosc
- 16 Tosc
- 32 Tosc
- 64 Tosc
- Internal RC Oscillator

For correct A/D conversions, the A/D conversion clock (TAD) must be as short as possible, but greater than the minimum TAD (see parameter 130 for more information).

Table 19-1 shows the resultant TAD times derived from the device operating frequencies and the A/D clock source selected.

AD Clock Sou	urce (TAD)	Maximum Device Frequency		
Operation	ADCS<2:0>	PIC18F2X20/4X20	PIC18LF2X2X/4X20 ⁽⁴⁾	
2 Tosc	000	2.86 MHz	1.43 kHz	
4 Tosc	100	5.71 MHz	2.86 MHz	
8 Tosc	001	11.43 MHz	5.72 MHz	
16 Tosc	101	22.86 MHz	11.43 MHz	
32 Tosc	010	40.0 MHz	22.86 MHz	
64 Tosc	110	40.0 MHz	22.86 MHz	
RC ⁽³⁾	x11	1.00 MHz ⁽¹⁾	1.00 MHz ⁽²⁾	

TABLE 19-1: TAD vs. DEVICE OPERATING FREQUENCIES

Note 1: The RC source has a typical TAD time of $1.2 \,\mu s$.

2: The RC source has a typical TAD time of $2.5 \,\mu$ s.

- **3:** For device frequencies above 1 MHz, the device must be in Sleep for the entire conversion or the A/D accuracy may be out of specification.
- 4: Low-power (PIC18LFXXXX) devices only.

The module is enabled by setting the HLVDEN bit. Each time that the HLVD module is enabled, the circuitry requires some time to stabilize. The IRVST bit is a read-only bit and is used to indicate when the circuit is stable. The module can only generate an interrupt after the circuit is stable and IRVST is set.

The VDIRMAG bit determines the overall operation of the module. When VDIRMAG is cleared, the module monitors for drops in VDD below a predetermined set point. When the bit is set, the module monitors for rises in VDD above the set point.

22.1 Operation

When the HLVD module is enabled, a comparator uses an internally generated reference voltage as the set point. The set point is compared with the trip point, where each node in the resistor divider represents a trip point voltage. The "trip point" voltage is the voltage level at which the device detects a high or low-voltage event, depending on the configuration of the module. When the supply voltage is equal to the trip point, the voltage tapped off of the resistor array is equal to the internal reference voltage generated by the voltage reference module. The comparator then generates an interrupt signal by setting the HLVDIF bit.

The trip point voltage is software programmable to any one of 16 values. The trip point is selected by programming the HLVDL<3:0> bits (HLVDCON<3:0>).

The HLVD module has an additional feature that allows the user to supply the trip voltage to the module from an external source. This mode is enabled when bits, HLVDL<3:0>, are set to '1111'. In this state, the comparator input is multiplexed from the external input pin, HLVDIN. This gives users flexibility because it allows them to configure the High/Low-Voltage Detect interrupt to occur at any voltage in the valid operating range.





REGISTER 23-6: CONFIG5L: CONFIGURATION REGISTER 5 LOW (BYTE ADDRESS 300008h)

U-0	U-0	U-0	U-0	R/C-1	R/C-1	R/C-1	R/C-1
	—	_		CP3 ⁽¹⁾	CP2 ⁽¹⁾	CP1	CP0
bit 7							bit 0
Legend:							

Logonan		
R = Readable bit	C = Clearable bit	U = Unimplemented bit, read as '0'
-n = Value when device is unp	programmed	u = Unchanged from programmed state

bit 7-4	Unimplemented: Read as '0'
bit 3	CP3: Code Protection bit ⁽¹⁾
	1 = Block 3 (006000-007FFFh) not code-protected 0 = Block 3 (006000-007FFFh) code-protected
bit 2	CP2: Code Protection bit ⁽¹⁾
	1 = Block 2 (004000-005FFFh) not code-protected
	0 = Block 2 (004000-005FFFh) code-protected
bit 1	CP1: Code Protection bit
	1 = Block 1 (002000-003FFFh) not code-protected
	0 = Block 1 (002000-003FFFh) code-protected
bit 0	CP0: Code Protection bit
	1 = Block 0 (000800-001FFFh) not code-protected
	0 = Block 0 (000800-001FFFh) code-protected

Note 1: Unimplemented in PIC18F2420/4420 devices; maintain this bit set.

REGISTER 23-7: CONFIG5H: CONFIGURATION REGISTER 5 HIGH (BYTE ADDRESS 300009h)

R/C-1	R/C-1	U-0	U-0	U-0	U-0	U-0	U-0
CPD	СРВ	—	—	—	—	—	—
bit 7							bit 0

Legend:		
R = Readable bit	C = Clearable bit	U = Unimplemented bit, read as '0'
-n = Value when device is unp	programmed	u = Unchanged from programmed state

bit 7	CPD: Data EEPROM Code Protection bit
	1 = Data EEPROM not code-protected
	0 = Data EEPROM code-protected
bit 6	CPB: Boot Block Code Protection bit
	1 = Boot block (000000-0007FFh) not code-protected
	0 = Boot block (000000-0007FFh) code-protected
bit 5-0	Unimplemented: Read as '0'

24.1.1 STANDARD INSTRUCTION SET

ADD	DLW	ADD Lite	ADD Literal to W					
Synta	ax:	ADDLW	ADDLW k					
Oper	rands:	$0 \le k \le 255$						
Oper	ration:	$(W) + k \rightarrow $	W					
Statu	is Affected:	N, OV, C, E	DC, Z					
Enco	oding:	0000	1111	kkkk	kkkk			
Desc	cription:	The conten 8-bit literal W.	its of W a 'k' and th	ire addeo e result i	to the s placed in			
Word	ds:	1	1					
Cycle	es:	1						
QC	ycle Activity:							
	Q1	Q2	Q3		Q4			
	Decode	Read literal 'k'	Proce Data	ss V a	/rite to W			
<u>Exan</u>	nple:	ADDLW 1	15h					
	Before Instruc	tion						
	VV =	10h						
	After Instruction	on						
	VV =	25h						

ADDWF	ADD W to f					
Syntax:	ADDWF f {,d {,a}}					
Operands:	0 ≤ f ≤ 255 d ∈ [0,1] a ∈ [0,1]					
Operation:	(W) + (f) \rightarrow dest					
Status Affected:	N, OV, C, DC, Z					
Encoding:	0010 01da ffff ffff					
Description:	Add W to register T. If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f' (default). If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank (default). If 'a' is '0' and the extended instruction set is enabled, this instruction operates in Indexed Literal Offset Addressing mode whenever $f \le 95$ (5Fh). See Section 24.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode" for details.					
Words:	1					
Cycles:	1					

QC	ycle Activity:					
	Q1		Q2	Q3		Q4
	Decode	Read register 'f'		Process Data		Write to destination
<u>Exan</u>	<u>nple:</u>	ADDWF		REG,	0, 0	
	Before Instruc	tion				
	W REG	= 17h = 0C2h				
	After Instruction					
	W REG	=	0D9h 0C2h			

Note:	All PIC18 instructions may take an optional label argument preceding the instruction mnemonic for use in
	symbolic addressing. If a label is used, the instruction format then becomes: {label} instruction argument(s).

ANDWF	AND W wi	th f		BC		Branch if	Carry	
Syntax:	ANDWF	f {,d {,a}}		Synta	ix:	BC n		
Operands:	$0 \le f \le 255$			Oper	ands:	-128 ≤ n ≤	127	
	d ∈ [0,1] a ∈ [0,1]			Operation	ation:	if Carry bit (PC) + 2 +	is '1', 2n → PC	
Operation:	(W) .AND. (f	f) \rightarrow dest		Statu	s Affected:	None		
Status Affected:	N, Z			Enco	ding:	1110	0010 n	nnn nnnn
Encoding: Description:	0001 The contents register 'f'. If in W. If 'd' is in register 'f' If 'a' is '0', th If 'a' is '1', th GPR bank (d If 'a' is '0' an set is enable in Indexed L mode where Section 24.3 Bit-Oriented	olda ff: s of W are AN ''d' is 'o', the r 'l', the result (default). he Access Ban he BSR is use default). hd the extenda ed, this instruct iteral Offset A ever $f \le 95$ (51 2.3 "Byte-Or d Instruction	ff ffff IDed with result is stored result is stored back back nk is selected. back nk is selected. back ed instruction back ction operates baddressing Fh). See biented and s in Indexed back	Desc Word Cycle Q Cy If Ju	ription: s: es: ycle Activity: mp: Q1	If the Carry will branch The 2's cor added to th incremente instruction, PC + 2 + 2 two-cycle in 1 1(2) Q2	y bit is '1', the mplement nu le PC. Since d to fetch the the new add n. This instru nstruction.	en the program mber '2n' is the PC will have e next lress will be loction is then a
Words:	1		uctano.		Decode	Read literal 'n'	Process Data	Write to PC
Cycles:	1				No	No	No	No
Q Cycle Activity:				If No	operation	operation	operation	operation
Q1	Q2	Q3	Q4		Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write to destination		Decode	Read literal 'n'	Process Data	No operation
Example: Before Instrue	ANDWF	REG, 0, 0		Exam	<u>iple:</u> Before Instruc	HERE	BC 5	
W REG After Instructi W REG	= 17h = C2h on = 02h = C2h				PC After Instructi If Carry PC If Carry PC	= ac on = 1; = ac = 0; = ac	Idress (HER Idress (HER Idress (HER	E) E + 12) E + 2)

SUBFSR Subtract Literal from FSR								
Synta	ax:	SUBFSR	f, k					
Oper	ands:	$0 \le k \le 63$						
		f∈[0,1,	2]					
Oper	ation:	FSR(f) – k	$s \rightarrow FSRf$					
Statu	s Affected:	None						
Enco	oding:	1110	1001	ffkk	kkkk			
Desc	ription:	The 6-bit I the conter 'f'.	The 6-bit literal 'k' is subtracted from the contents of the FSR specified by 'f'.					
Word	ls:	1	1					
Cycle	es:	1	1					
QC	ycle Activity:							
	Q1	Q2	Q3		Q4			
	Decode	Read register 'f'	Proce Data	ess a	Write to destination			

Example:	SUBFSR	2,	23h
----------	--------	----	-----

Before Ir	nstruction
-----------	------------

FSR2	=	03FFh
After Instruct	ion	
FSR2	=	03DCh

SUBULNK	Subtract I	iteral fron	n FSR2 a	nd Return	
Syntax:	SUBULNK	SUBULNK k			
Operands:	$0 \le k \le 63$				
Operation:	FSR2 – k -	→ FSR2,			
	$(TOS) \rightarrow P$	С			
Status Affected:	None				
Encoding:	1110	1001	11kk	kkkk	
	contents of the FSR2. A RETURN is then executed by loading the PC with the TOS. The instruction takes two cycles to execute; a NOP is performed during the second cycle. This may be thought of as a special case of the SUBFSR instruction, where f = 3 (binary '11'); it operates only on FSR2.				
Words:	1				
Cycles:	2				
Q Cycle Activity	y:				
Q1	Q2	T	Q3	Q4	
Decode	Rea registe	d Pro er 'f' E	ocess Data	Write to destination	
No	No		No	No	
Operatio	n Onera		oration	Operation	

Example: SUBULNK 23h

Before Instru	ction	
FSR2	=	03FFh
PC	=	0100h
After Instructi	on	
FSR2	=	03DCh
PC	=	(TOS)



TABLE 26-14: EXAMPLE SPI MODE REQUIREMENTS (MASTER MODE, CKE = 0)

Param No.	Symbol	Characteristi	Min	Max	Units	Conditions	
70	TssL2scH, TssL2scL	$\overline{SS} \downarrow$ to SCK \downarrow or SCK \uparrow Input	Тсү	—	ns		
73	TdiV2scH, TdiV2scL	Setup Time of SDI Data Input t	20	_	ns		
73A	Tb2b	Last Clock Edge of Byte 1 to th of Byte 2	1.5 Tcy + 40	_	ns	(Note 2)	
74	TscH2diL, TscL2diL	Hold Time of SDI Data Input to SCK Edge		40	-	ns	
75	TdoR	SDO Data Output Rise Time	SDO Data Output Rise Time PIC18FXXXX		25	ns	
			PIC18LFXXXX		45	ns	VDD = 2.0V
76	TdoF	SDO Data Output Fall Time			25	ns	
78	TscR	SCK Output Rise Time	PIC18FXXXX	—	25	ns	
		(Master mode)	PIC18LFXXXX	—	45	ns	VDD = 2.0V
79	TscF	SCK Output Fall Time (Master mode)		—	25	ns	
80	TscH2doV, TscL2doV	SDO Data Output Valid after	PIC18FXXXX		50	ns	
		SCK Edge	PIC18LFXXXX	_	100	ns	VDD = 2.0V

Note 1: Requires the use of Parameter #73A.

2: Only if Parameter #71A and #72A are used.



TABLE 26-20: MASTER SSP I²C[™] BUS START/STOP BITS REQUIREMENTS

Param. No.	Symbol	Characte	eristic	Min	Max	Units	Conditions
90	TSU:STA	Start Condition	100 kHz mode	2(Tosc)(BRG + 1)		ns	Only relevant for
		Setup Time	400 kHz mode	2(Tosc)(BRG + 1)	—		Repeated Start
			1 MHz mode ⁽¹⁾	2(Tosc)(BRG + 1)	_		condition
91	THD:STA	Start Condition	100 kHz mode	2(Tosc)(BRG + 1)	_	ns	After this period, the
		Hold Time	400 kHz mode	2(Tosc)(BRG + 1)	_		first clock pulse is
			1 MHz mode ⁽¹⁾	2(Tosc)(BRG + 1)	_		generated
92	Tsu:sto	Stop Condition	100 kHz mode	2(Tosc)(BRG + 1)	—	ns	3
		Setup Time	400 kHz mode	2(Tosc)(BRG + 1)	_		
			1 MHz mode ⁽¹⁾	2(Tosc)(BRG + 1)	_		
93	THD:STO	Stop Condition	100 kHz mode	2(Tosc)(BRG + 1)	—	ns	
		Hold Time	400 kHz mode	2(Tosc)(BRG + 1)	—		
			1 MHz mode ⁽¹⁾	2(Tosc)(BRG + 1)]	

Note 1: Maximum pin capacitance = 10 pF for all I^2C pins.

FIGURE 26-20: MASTER SSP I²C[™] BUS DATA TIMING



NOTES:







© 2008 Microchip Technology Inc.

W

Watchdog Timer (WDT)	249 258
Associated Registers	
Control Register	
During Oscillator Failure	
Programming Considerations	
WCOL	189, 190, 191, 194
WCOL Status Flag	189, 190, 191, 194
WWW Address	
WWW, On-Line Support	6

Х

XORLW	. 307
XORWF	. 308