



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	64MHz
Connectivity	I ² C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, HLVD, POR, PWM, WDT
Number of I/O	35
Program Memory Size	64KB (32K x 16)
Program Memory Type	FLASH
EEPROM Size	1К х 8
RAM Size	3.8K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 30x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f46k22-i-pt

Email: info@E-XFL.COM

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

2.2 Oscillator Control

The OSCCON, OSCCON2 and OSCTUNE registers (Register 2-1 to Register 2-3) control several aspects of the device clock's operation, both in full-power operation and in power-managed modes.

- Main System Clock Selection (SCS)
- Primary Oscillator Circuit Shutdown (PRISD)
- Secondary Oscillator Enable (SOSCGO)
- Primary Clock Frequency 4x multiplier (PLLEN)
- Internal Frequency selection bits (IRCF, INTSRC)
- Clock Status bits (OSTS, HFIOFS, MFIOFS, LFIOFS. SOSCRUN, PLLRDY)
- Power management selection (IDLEN)

2.2.1 MAIN SYSTEM CLOCK SELECTION

The System Clock Select bits, SCS<1:0>, select the main clock source. The available clock sources are

- Primary clock defined by the FOSC<3:0> bits of CONFIG1H. The primary clock can be the primary oscillator, an external clock, or the internal oscillator block.
- Secondary clock (secondary oscillator)
- Internal oscillator block (HFINTOSC, MFINTOSC and LFINTOSC).

The clock source changes immediately after one or more of the bits is written to, following a brief clock transition interval. The SCS bits are cleared to select the primary clock on all forms of Reset.

2.2.2 INTERNAL FREQUENCY SELECTION

The Internal Oscillator Frequency Select bits (IRCF<2:0>) select the frequency output of the internal oscillator block. The choices are the LFINTOSC source (31.25 kHz), the MFINTOSC source (31.25 kHz, 250 kHz or 500 kHz) and the HFINTOSC source (16 MHz) or one of the frequencies derived from the HFINTOSC postscaler (31.25 kHz to 8 MHz). If the internal oscillator block is supplying the main clock, changing the states of these bits will have an immediate change on the internal oscillator's output. On device Resets, the output frequency of the internal oscillator is set to the default frequency of 1 MHz.

2.2.3 LOW FREQUENCY SELECTION

When a nominal output frequency of 31.25 kHz is selected (IRCF<2:0> = 000), users may choose which internal oscillator acts as the source. This is done with the INTSRC bit of the OSCTUNE register and MFIOSEL bit of the OSCCON2 register. See Figure 2-2 and Register 2-1 for specific 31.25 kHz selection. This option allows users to select a 31.25 kHz clock (MFINTOSC or HFINTOSC) that can be tuned using the TUN<5:0> bits in OSCTUNE register, while maintaining power savings with a very low clock speed. LFINTOSC always remains the clock source for features such as the Watchdog Timer and the Fail-Safe Clock Monitor, regardless of the setting of INTSRC and MFIOSEL bits

This option allows users to select the tunable and more precise HFINTOSC as a clock source, while maintaining power savings with a very low clock speed.

2.2.4 POWER MANAGEMENT

The IDLEN bit of the OSCCON register determines whether the device goes into Sleep mode or one of the Idle modes when the SLEEP instruction is executed.

5.3.3 INSTRUCTIONS IN PROGRAM MEMORY

The program memory is addressed in bytes. Instructions are stored as either two bytes or four bytes in program memory. The Least Significant Byte of an instruction word is always stored in a program memory location with an even address (LSb = 0). To maintain alignment with instruction boundaries, the PC increments in steps of two and the LSb will always read '0' (see Section 5.1.1 "Program Counter").

Figure 5-4 shows an example of how instruction words are stored in the program memory.

The CALL and GOTO instructions have the absolute program memory address embedded into the instruction. Since instructions are always stored on word boundaries, the data contained in the instruction is a word address. The word address is written to PC<20:1>, which accesses the desired byte address in program memory. Instruction #2 in Figure 5-4 shows how the instruction GOTO 0006h is encoded in the program memory. Program branch instructions, which encode a relative address offset, operate in the same manner. The offset value stored in a branch instruction represents the number of single-word instructions that the PC will be offset by. Section 25.0 "Instruction Set Summary" provides further details of the instruction set.

			LSB = 1	LSB = 0	Word Address \downarrow	
	Program N	lemory			000000h	
	Byte Locations \rightarrow				000002h	
					000004h	
					000006h	
Instruction 1:	MOVLW	055h	0Fh	55h	000008h	
Instruction 2:	GOTO	0006h	EFh	03h	00000Ah	
			F0h	00h	00000Ch	
Instruction 3:	MOVFF	123h, 456h	C1h	23h	00000Eh	
			F4h	56h	000010h	
					000012h	
					000014h	

FIGURE 5-4: INSTRUCTIONS IN PROGRAM MEMORY

5.3.4 TWO-WORD INSTRUCTIONS

The standard PIC18 instruction set has four two-word instructions: CALL, MOVFF, GOTO and LSFR. In all cases, the second word of the instruction always has '1111' as its four Most Significant bits; the other 12 bits are literal data, usually a data memory address.

The use of '1111' in the 4 MSbs of an instruction specifies a special form of NOP. If the instruction is executed in proper sequence – immediately after the first word – the data in the second word is accessed and used by the instruction sequence.

If the first word is skipped for some reason and the second word is executed by itself, a NOP is executed instead. This is necessary for cases when the two-word instruction is preceded by a conditional instruction that changes the PC. Example 5-4 shows how this works.

Note:	See Section 5.8 "PIC18 Instruction Execution and the Extended
	Instruction Set" for information on
	two-word instructions in the extended instruction set.

CASE 1:		
Object Code	Source Code	
0110 0110 0000 0000	TSTFSZ REG1	; is RAM location 0?
1100 0001 0010 0011	MOVFF REG1, RE	G2 ; No, skip this word
1111 0100 0101 0110		; Execute this word as a NOP
0010 0100 0000 0000	ADDWF REG3	; continue code
CASE 2:		
Object Code	Source Code	
0110 0110 0000 0000	TSTFSZ REG1	; is RAM location 0?
1100 0001 0010 0011	MOVFF REG1, RE	G2 ; Yes, execute this word
1111 0100 0101 0110		; 2nd word of instruction
0010 0100 0000 0000	ADDWF REG3	; continue code

EXAMPLE 5-4: TWO-WORD INSTRUCTIONS

6.3.1 TABLAT – TABLE LATCH REGISTER

The Table Latch (TABLAT) is an 8-bit register mapped into the SFR space. The Table Latch register is used to hold 8-bit data during data transfers between program memory and data RAM.

6.3.2 TBLPTR – TABLE POINTER REGISTER

The Table Pointer (TBLPTR) register addresses a byte within the program memory. The TBLPTR is comprised of three SFR registers: Table Pointer Upper Byte, Table Pointer High Byte and Table Pointer Low Byte (TBLPTRU:TBLPTRH:TBLPTRL). These three registers join to form a 22-bit wide pointer. The low-order 21 bits allow the device to address up to 2 Mbytes of program memory space. The 22nd bit allows access to the device ID, the user ID and the Configuration bits.

The Table Pointer register, TBLPTR, is used by the TBLRD and TBLWT instructions. These instructions can update the TBLPTR in one of four ways based on the table operation. These operations on the TBLPTR affect only the low-order 21 bits.

6.3.3 TABLE POINTER BOUNDARIES

TBLPTR is used in reads, writes and erases of the Flash program memory.

When a TBLRD is executed, all 22 bits of the TBLPTR determine which byte is read from program memory directly into the TABLAT register.

When a TBLWT is executed the byte in the TABLAT register is written, not to Flash memory but, to a holding register in preparation for a program memory write. The holding registers constitute a write block which varies depending on the device (see Table 6-1). The 3, 4, or 5 LSbs of the TBLPTRL register determine which specific address within the holding register block is written to. The MSBs of the Table Pointer have no effect during TBLWT operations.

When a program memory write is executed the entire holding register block is written to the Flash memory at the address determined by the MSbs of the TBLPTR. The 3, 4, or 5 LSBs are ignored during Flash memory writes. For more detail, see **Section 6.6** "**Writing to Flash Program Memory**".

When an erase of program memory is executed, the 16 MSbs of the Table Pointer register (TBLPTR<21:6>) point to the 64-byte block that will be erased. The Least Significant bits (TBLPTR<5:0>) are ignored.

Figure 6-3 describes the relevant boundaries of TBLPTR based on Flash program memory operations.

TABLE 6-1: TABLE POINTER OPERATIONS WITH TBLRD AND TBLWT INSTRUCTIONS

Example	Operation on Table Pointer					
TBLRD* TBLWT*	TBLPTR is not modified					
TBLRD*+ TBLWT*+	TBLPTR is incremented after the read/write					
TBLRD*- TBLWT*-	TBLPTR is decremented after the read/write					
TBLRD+* TBLWT+*	TBLPTR is incremented before the read/write					

FIGURE 6-3: TABLE POINTER BOUNDARIES BASED ON OPERATION



R/W-1	R/W-1	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
INT2IP	INT1IP	—	INT2IE	INT1IE	—	INT2IF	INT1IF
bit 7							bit 0
Legend:							
R = Reada	able bit	W = Writable	bit	U = Unimple	mented bit, read	1 as '0'	
-n = Value	at POR	'1' = Bit is set		'0' = Bit is cle	eared	x = Bit is unkr	nown
bit 7	INT2IP: INT2	External Interr	upt Priority bi	t			
	1 = High prio	rity					
L'HO	0 = Low prior	ity Easterne et lasterne					
DIT 6	INTTIP: INTT	External Interr	upt Priority bi	τ			
	1 = High pho 0 = Low prior	itv					
bit 5	Unimplemen	ted: Read as '	0'				
bit 4	INT2IE: INT2	External Interr	upt Enable bi	t			
	1 = Enables t	the INT2 extern	nal interrupt				
	0 = Disables	the INT2 exter	nal interrupt				
bit 3	INT1IE: INT1	External Interr	upt Enable bi	t			
	1 = Enables t 0 = Disables	the INT1 extern the INT1 extern	nal interrupt				
bit 2	Unimplemen	ted: Read as '	0'				
bit 1	INT2IF: INT2	External Interr	upt Flag bit				
	1 = The INT2	external inter	upt occurred	(must be clear	ed bv software)		
	0 = The INT2	external inter	rupt did not o	ccur	,		
bit 0	INT1IF: INT1	External Interr	upt Flag bit				
	1 = The INT1	external interi	upt occurred	(must be clear	ed by software)		
	0 = The INT1	external inter	rupt did not o	ccur			
Note:	Interrupt flag bits a	re set when an	interrupt				
	condition occurs, re	egardless of the	e state of				
	its corresponding e	enable bit or the	ne global				
	the appropriate inte	errupt flag bits	are clear				
	prior to enabling a	n interrupt. Thi	s feature				
	allows for software	polling.					

REGISTER 9-3: INTCON3: INTERRUPT CONTROL 3 REGISTER

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W/HC-0/u	R-x/x	R/W-0/u	R/W-0/u
TMRxGE	TxGPOL	TxGTM	TxGSPM	TxGGO/DONE	TxGVAL	TxGSS	6<1:0>
bit 7	·						bit 0
Legend:							
R = Readable	bit	W = Writable	bit	U = Unimplemer	nted bit, read a	as '0'	
u = Bit is unch	anged	x = Bit is unkr	nown	-n/n = Value at F	POR and BOR	/Value at all ot	her Resets
'1' = Bit is set		'0' = Bit is cle	ared	HC = Bit is clear	ed by hardwar	re	
bit 7	bit 7 TMRxGE: Timer1/3/5 Gate Enable bit <u>If TMRxON = 0</u> : This bit is ignored <u>If TMRxON = 1</u> : 1 = Timer1/3/5 counting is controlled by the Timer1/3/5 gate function						
bit 6	TxGPOL: Tim 1 = Timer1/3 0 = Timer1/3	ner1/3/5 Gate F /5 gate is activ /5 gate is activ	Polarity bit e-high (Timer1 e-low (Timer1/	/3/5 counts when 3/5 counts when g	gate is high) gate is low)		
bit 5	TxGTM: Time 1 = Timer1/3 0 = Timer1/3 Timer1/3/5 ga	er1/3/5 Gate To /5 Gate Toggle /5 Gate Toggle ate flip-flop togg	ggle Mode bit mode is enab mode is disat gles on every r	led bled and toggle flip rising edge.	o-flop is cleare	d	
bit 4	TxGSPM: Tin 1 = Timer1/3 0 = Timer1/3	ner1/3/5 Gate 3 /5 gate Single- /5 gate Single-	Single-Pulse M Pulse mode is Pulse mode is	lode bit enabled and is co disabled	ontrolling Time	r1/3/5 gate	
bit 3 TxGGO/DONE: Timer1/3/5 Gate Single-Pulse Acquisition Status bit 1 = Timer1/3/5 gate single-pulse acquisition is ready, waiting for an edge 0 = Timer1/3/5 gate single-pulse acquisition has completed or has not been started This bit is automatically cleared when TxGSPM is cleared.							
bit 2	t 2 TxGVAL: Timer1/3/5 Gate Current State bit Indicates the current state of the Timer1/3/5 gate that could be provided to TMRxH:TMRxL. Unaffected by Timer1/3/5 Gate Enable (TMRxGE).						
bit 1-0	Unaffected by Timer1/3/5 Gate Enable (TMRxGE). bit 1-0 TxGSS<1:0>: Timer1/3/5 Gate Source Select bits 00 = Timer1/3/5 Gate pin 01 = Timer2/4/6 Match PR2/4/6 output (See Table 12-5 for proper timer match selection) 10 = Comparator 1 optionally synchronized output (sync_C1OUT) 11 = Comparator 2 optionally synchronized output (sync_C2OUT)						

REGISTER 12-2: TXGCON: TIMER1/3/5 GATE CONTROL REGISTER

14.4.2 FULL-BRIDGE MODE

In Full-Bridge mode, all four pins are used as outputs. An example of full-bridge application is shown in Figure 14-10.

In the Forward mode, pin CCPx/PxA is driven to its active state, pin PxD is modulated, while PxB and PxC will be driven to their inactive state as shown in Figure 14-11.

In the Reverse mode, PxC is driven to its active state, pin PxB is modulated, while PxA and PxD will be driven to their inactive state as shown Figure 14-11.

PxA, PxB, PxC and PxD outputs are multiplexed with the PORT data latches. The associated TRIS bits must be cleared to configure the PxA, PxB, PxC and PxD pins as outputs.

FIGURE 14-10: EXAMPLE OF FULL-BRIDGE APPLICATION



The $\mathsf{I}^2\mathsf{C}$ 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 15-2 is a block diagram of the I^2C interface module in Master mode. Figure 15-3 is a diagram of the I^2C interface module in Slave mode.

The PIC18(L)F2X/4XK22 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 15-2: MSSPx BLOCK DIAGRAM (I²C MASTER MODE)



15.5.8 GENERAL CALL ADDRESS SUPPORT

The addressing procedure for the I^2C bus is such that the first byte after the Start condition usually determines which device will be the slave addressed by the master device. The exception is the general call address which can address all devices. When this address is used, all devices should, in theory, respond with an acknowledge.

The general call address is a reserved address in the I^2C protocol, defined as address 0x00. When the GCEN bit of the SSPxCON2 register is set, the slave module will automatically \overline{ACK} the reception of this address regardless of the value stored in SSPxADD. After the slave clocks in an address of all zeros with the R/W bit clear, an interrupt is generated and slave software can read SSPxBUF and respond. Figure 15-24 shows a general call reception sequence.

In 10-bit Address mode, the UA bit will not be set on the reception of the general call address. The slave will prepare to receive the second byte as data, just as it would in 7-bit mode.

If the AHEN bit of the SSPxCON3 register is set, just as with any other address reception, the slave hardware will stretch the clock after the 8th falling edge of SCLx. The slave must then set its ACKDT value and release the clock with communication progressing as it would normally.



FIGURE 15-24: SLAVE MODE GENERAL CALL ADDRESS SEQUENCE

15.5.9 SSPx MASK REGISTER

An SSPx Mask (SSPxMSK) register (Register 15-6) is available in I²C Slave mode as a mask for the value held in the SSPxSR register during an address comparison operation. A zero ('0') bit in the SSPxMSK register has the effect of making the corresponding bit of the received address a "don't care".

This register is reset to all '1's upon any Reset condition and, therefore, has no effect on standard SSPx operation until written with a mask value.

The SSPx Mask register is active during:

- 7-bit Address mode: address compare of A<7:1>.
- 10-bit Address mode: address compare of A<7:0> only. The SSPx mask has no effect during the reception of the first (high) byte of the address.

16.2 Clock Accuracy with Asynchronous Operation

The factory calibrates the internal oscillator block output (HFINTOSC). However, the HFINTOSC frequency may drift as VDD or temperature changes, and this directly affects the asynchronous baud rate. Two methods may be used to adjust the baud rate clock, but both require a reference clock source of some kind.

The first (preferred) method uses the OSCTUNE register to adjust the HFINTOSC output. Adjusting the value in the OSCTUNE register allows for fine resolution changes to the system clock source. See **Section 2.6** "Internal Clock Modes" for more information.

The other method adjusts the value in the Baud Rate Generator. This can be done automatically with the Auto-Baud Detect feature (see **Section 16.4.1 "Auto-Baud Detect"**). There may not be fine enough resolution when adjusting the Baud Rate Generator to compensate for a gradual change in the peripheral clock frequency.

16.4.1 AUTO-BAUD DETECT

The EUSART module supports automatic detection and calibration of the baud rate.

In the Auto-Baud Detect (ABD) mode, the clock to the BRG is reversed. Rather than the BRG clocking the incoming RXx signal, the RXx signal is timing the BRG. The Baud Rate Generator is used to time the period of a received 55h (ASCII "U") which is the Sync character for the LIN bus. The unique feature of this character is that it has five rising edges including the Stop bit edge.

Setting the ABDEN bit of the BAUDCONx register starts the auto-baud calibration sequence (Section 16.4.2 "Auto-baud Overflow"). While the ABD sequence takes place, the EUSART state machine is held in Idle. On the first rising edge of the receive line, after the Start bit, the SPBRGx begins counting up using the BRG counter clock as shown in Table 16-6. The fifth rising edge will occur on the RXx/ DTx pin at the end of the eighth bit period. At that time, an accumulated value totaling the proper BRG period is left in the SPBRGHx:SPBRGx register pair, the ABDEN bit is automatically cleared, and the RCxIF interrupt flag is set. A read operation on the RCREGx needs to be performed to clear the RCxIF interrupt. RCREGx content should be discarded. When calibrating for modes that do not use the SPBRGHx register the user can verify that the SPBRGx register did not overflow by checking for 00h in the SPBRGHx register.

The BRG auto-baud clock is determined by the BRG16 and BRGH bits as shown in Table 16-6. During ABD, both the SPBRGHx and SPBRGx registers are used as a 16-bit counter, independent of the BRG16 bit setting. While calibrating the baud rate period, the SPBRGHx and SPBRGx registers are clocked at 1/8th the BRG base clock rate. The resulting byte measurement is the average bit time when clocked at full speed.

- Note 1: If the WUE bit is set with the ABDEN bit, auto-baud detection will occur on the byte <u>following</u> the Break character (see Section 16.4.3 "Auto-Wake-up on Break").
 - 2: It is up to the user to determine that the incoming character baud rate is within the range of the selected BRG clock source. Some combinations of oscillator frequency and EUSART baud rates are not possible.
 - 3: During the auto-baud process, the autobaud counter starts counting at one. Upon completion of the auto-baud sequence, to achieve maximum accuracy, subtract one from the SPBRGHx:SPBRGx register pair.

TABLE 16-6:	BRG COUNTER CLOCK
	RATES

BRG16	BRGH	BRG Base Clock	BRG ABD Clock
0	0	Fosc/64	Fosc/512
0	1	Fosc/16	Fosc/128
1	0	Fosc/16	Fosc/128
1	1	Fosc/4	Fosc/32

Note: During the ABD sequence, SPBRGx and SPBRGHx registers are both used as a 16-bit counter, independent of BRG16 setting.

BRG Value	XXXXh	0000h		001Ch
RXx/DTx pin		Start	Edge #1 Edge #2 Edge #3 Edge #4	it 7 Stop bit
BRG Clock		huuuuuu		
	Set by User —	I I		Auto Cleared
ABDEN bit	·			
RCIDL		' ' '		
RCxIF bit		·		
(Interrupt)		1	1	
Read		1 		
RCREGX		1	1	
SPBRGx		 	XXh	(1Ch
SPBRGHx		1	XXh) 00h
Note	I: The ABD sequ	ence requires the EUSA	RT module to be configured in Asynchronous mode.	

FIGURE 16-6: AUTOMATIC BAUD RATE CALIBRATION





FIGURE 17-6: ADC TRANSFER FUNCTION



19.6 Creating a Delay with the CTMU Module

A unique feature on board the CTMU module is its ability to generate system clock independent output pulses based on an external capacitor value. This is accomplished using the internal comparator voltage reference module, Comparator 2 input pin and an external capacitor. The pulse is output onto the CTPLS pin. To enable this mode, set the TGEN bit.

See Figure 19-4 for an example circuit. *C*PULSE is chosen by the user to determine the output pulse width on CTPLS. The pulse width is calculated by T = (CPULSE/I)*V, where *I* is known from the current source measurement step (Section 19.3.1 "Current Source Calibration") and *V* is the internal reference voltage (CVREF).

An example use of this feature is for interfacing with variable capacitive-based sensors, such as a humidity sensor. As the humidity varies, the pulse width output on CTPLS will vary. The CTPLS output pin can be connected to an input capture pin and the varying pulse width is measured to determine the humidity in the application.

Follow these steps to use this feature:

- 1. Initialize Comparator 2.
- 2. Initialize the comparator voltage reference.
- 3. Initialize the CTMU and enable time delay generation by setting the TGEN bit.
- 4. Set EDG1STAT.
- 5. When CPULSE charges to the value of the voltage reference trip point, an output pulse is generated on CTPLS.

FIGURE 19-4: TYPICAL CONNECTIONS AND INTERNAL CONFIGURATION FOR PULSE DELAY GENERATION



19.7 Operation During Sleep/Idle Modes

19.7.1 SLEEP MODE AND DEEP SLEEP MODES

When the device enters any Sleep mode, the CTMU module current source is always disabled. If the CTMU is performing an operation that depends on the current source when Sleep mode is invoked, the operation may not terminate correctly. Capacitance and time measurements may return erroneous values.

19.7.2 IDLE MODE

The behavior of the CTMU in Idle mode is determined by the CTMUSIDL bit (CTMUCONH<5>). If CTMUSIDL is cleared, the module will continue to operate in Idle mode. If CTMUSIDL is set, the module's current source is disabled when the device enters Idle mode. If the module is performing an operation when Idle mode is invoked, in this case, the results will be similar to those with Sleep mode.

19.8 CTMU Peripheral Module Disable (PMD)

When this peripheral is not used, the Peripheral Module Disable bit can be set to disconnect all clock sources to the module, reducing power consumption to an absolute minimum. See **Section 3.6** "**Selective Peripheral Module Control**".

The module is enabled by setting the HLVDEN bit (HLVDCON<4>). Each time the HLVD module is enabled, the circuitry requires some time to stabilize. The IRVST bit (HLVDCON<5>) is a read-only bit 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 (HLVDCON<7>) 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.

23.2 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 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 the flexibility of configuring the High/Low-Voltage Detect interrupt to occur at any voltage in the valid operating range.

FIGURE 23-1: HLVD MODULE BLOCK DIAGRAM (WITH EXTERNAL INPUT)



26.2 MPLAB XC Compilers

The MPLAB XC Compilers are complete ANSI C compilers for all of Microchip's 8, 16, and 32-bit MCU and DSC devices. These compilers provide powerful integration capabilities, superior code optimization and ease of use. MPLAB XC Compilers run on Windows, Linux or MAC OS X.

For easy source level debugging, the compilers provide debug information that is optimized to the MPLAB X IDE.

The free MPLAB XC Compiler editions support all devices and commands, with no time or memory restrictions, and offer sufficient code optimization for most applications.

MPLAB XC Compilers include an assembler, linker and utilities. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. MPLAB XC Compiler uses the assembler to produce its object file. Notable features of the assembler include:

- · Support for the entire device instruction set
- · Support for fixed-point and floating-point data
- Command-line interface
- · Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

26.3 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel[®] standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code, and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB X IDE projects
- User-defined macros to streamline
 assembly code
- Conditional assembly for multipurpose source files
- Directives that allow complete control over the assembly process

26.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

26.5 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC DSC devices. MPLAB XC Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- Support for the entire device instruction set
- · Support for fixed-point and floating-point data
- Command-line interface
- Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

Param. No.	Symbol	Characteristic		Min	Max	Units	Conditions
100	Тнідн	Clock High Time	100 kHz mode	4.0		μS	Must operate at a minimum of 1.5 MHz
			400 kHz mode	0.6	_	μS	Must operate at a minimum of 10 MHz
			SSP Module	1.5 TCY	_		
101	TLOW	Clock Low Time	100 kHz mode	4.7	_	μS	Must operate at a minimum of 1.5 MHz
			400 kHz mode	1.3		μS	Must operate at a minimum of 10 MHz
			SSP Module	1.5 TCY	_		
102	TR	SDA and SCL Rise	100 kHz mode	—	1000	ns	
		Time	400 kHz mode	20 + 0.1 Св	300	ns	CB is specified to be from 10 to 400 pF
103	03 TF SDA and SCL Fal	SDA and SCL Fall	100 kHz mode	—	300	ns	
		Time	400 kHz mode	20 + 0.1 Св	300	ns	CB is specified to be from 10 to 400 pF
90	TSU:STA	Start Condition	100 kHz mode	4.7	—	μS	Only relevant for Repeated
		Setup Time	400 kHz mode	0.6	—	μS	Start condition
91	THD:STA	Start Condition	100 kHz mode	4.0	—	μS	After this period, the first
		Hold Time	400 kHz mode	0.6	—	μS	clock pulse is generated
106	THD:DA	Data Input Hold	100 kHz mode	0	_	ns	
	Т	Time	400 kHz mode	0	0.9	μS	
107	TSU:DAT	Data Input Setup	100 kHz mode	250	_	ns	(Note 2)
		Time	400 kHz mode	100	_	ns	
92	TSU:STO	Stop Condition	100 kHz mode	4.7	—	μS	
		Setup Time	400 kHz mode	0.6	—	μS	
109	ΤΑΑ	Output Valid from	100 kHz mode	—	3500	ns	(Note 1)
		Clock	400 kHz mode	—	—	ns	
110	TBUF	Bus Free Time	100 kHz mode	4.7	—	μS	Time the bus must be free
			400 kHz mode	1.3	_	μS	before a new transmission can start
D102	Св	Bus Capacitive Load	ding	—	400	pF	

TABLE 27-16:	I ² C BUS DATA	REQUIREMENTS	(SLAVE MODE)
--------------	---------------------------	--------------	--------------

Note 1: As a transmitter, the device must provide this internal minimum delay time to bridge the undefined region (min. 300 ns) of the falling edge of SCL to avoid unintended generation of Start or Stop conditions.

2: A fast mode I²C bus device can be used in a standard mode I²C bus system but the requirement, TSU:DAT ≥ 250 ns, must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line, TR max. + TSU:DAT = 1000 + 250 = 1250 ns (according to the standard mode I²C bus specification), before the SCL line is released.



FIGURE 28-55: PIC18F2X/4XK22 MAXIMUM IDD: PRI_RUN EC HIGH POWER



© 2010-2016 Microchip Technology Inc.



FIGURE 28-60: PIC18LF2X/4XK22 TYPICAL IDD: PRI_IDLE EC MEDIUM POWER





28-Lead Plastic Quad Flat, No Lead Package (ML) – 6x6 mm Body [QFN] with 0.55 mm Contact Length

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
Contact Pitch	E	0.65 BSC		
Optional Center Pad Width	W2			4.25
Optional Center Pad Length	T2			4.25
Contact Pad Spacing	C1		5.70	
Contact Pad Spacing	C2		5.70	
Contact Pad Width (X28)	X1			0.37
Contact Pad Length (X28)	Y1			1.00
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2105A

44-Lead Plastic Quad Flat, No Lead Package (ML) - 8x8 mm Body [QFN or VQFN]

Note: For the most current package drawings, please see the Microchip Packaging Specification located at http://www.microchip.com/packaging



Microchip Technology Drawing C04-103D Sheet 1 of 2

THE MICROCHIP WEBSITE

Microchip provides online support via our website at www.microchip.com. This website is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the website contains the following information:

- Product Support Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- General Technical Support Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- Business of Microchip Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip website at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the website at: http://microchip.com/support