



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	24
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)	1.8V ~ 3.6V
Data Converters	A/D 19x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SSOP (0.209", 5.30mm Width)
Supplier Device Package	28-SSOP
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lf25k22-i-ss

Email: info@E-XFL.COM

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

PIC18(L)F2X/4XK22







PIC18(L)F2X/4XK22

TABLE 2: F	PIC18(L)F2XK22	PIN SUMMARY
------------	----------------	--------------------

28-SSOP, SOIC 28-SPDIP	28-QFN, UQFN	0/1	Analog	Comparator	CTMU	SR Latch	Reference	(E)CCP	EUSART	MSSP	Timers	Interrupts	Pull-up	Basic
2	27	RA0	AN0	C12IN0-										
3	28	RA1	AN1	C12IN1-										
4	1	RA2	AN2	C2IN+			VREF- DACOUT							
5	2	RA3	AN3	C1IN+			VREF+							
6	3	RA4		C1OUT		SRQ		CCP5			TOCKI			
7	4	RA5	AN4	C2OUT		SRNQ	HLVDIN			SS1				
10	7	RA6												OSC2 CLKO
9	6	RA7												OSC1 CLKI
21	18	RB0	AN12			SRI		CCP4 FLT0		SS2		INT0	Y	
22	19	RB1	AN10	C12IN3-				P1C		SCK2 SCL2		INT1	Y	
23	20	RB2	AN8		CTED1			P1B		SDI2 SDA2		INT2	Y	
24	21	RB3	AN9	C12IN2-	CTED2			CCP2 P2A ⁽¹⁾		SDO2			Y	
25	22	RB4	AN11					P1D			T5G	IOC	Y	
26	23	RB5	AN13					CCP3 P3A ⁽⁴⁾ P2B ⁽³⁾			T1G T3CKI ⁽²⁾	IOC	Y	
27	24	RB6							TX2/CK2			IOC	Y	PGC
28	25	RB7							RX2/DT2			IOC	Y	PGD
11	8	RC0						P2B ⁽³⁾			SOSCO T1CKI T3CKI ⁽²⁾ T3G			
12	9	RC1						CCP2 P2A ⁽¹⁾			SOSCI			
13	10	RC2	AN14		CTPLS			CCP1 P1A			T5CKI			
14	11	RC3	AN15							SCK1 SCL1				
15	12	RC4	AN16							SDI1 SDA1				
16	13	RC5	AN17							SDO1				
17	14	RC6	AN18					CCP3 P3A ⁽⁴⁾	TX1/CK1					
18	15	RC7	AN19					P3B	RX1/DT1					
1	26	RE3												MCLR VPP
8, 19 19	5, 16 16	Vss												Vss
20	17	Vdd												Vdd

 CCP2/P2A multiplexed in fuses.
 T3CKI multiplexed in fuses.
 P2B multiplexed in fuses.
 CCP3/P3A multiplexed in fuses. Note 1:

	Pin M	lumber		Din Nome	Pin	Buffer	Description
PDIP	TQFP	QFN	UQFN	Pin Name	Туре	Туре	Description
10	27	27	25	RE2/CCP5/AN7			
				RE2	I/O	ST	Digital I/O.
				CCP5	I/O	ST	Capture 5 input/Compare 5 output/PWM 5 output
				AN7	Ι	Analog	Analog input 7.
1	18	18	16	RE3/VPP/MCLR			
				RE3	Ι	ST	Digital input.
				Vpp	Р		Programming voltage input.
				MCLR	I	ST	Active-low Master Clear (device Reset) input.
11,32	7, 28	7, 8, 28, 29	7, 26	Vdd	Р	—	Positive supply for logic and I/O pins.
12,31	6, 29	6,30, 31	6, 27	Vss	Р	_	Ground reference for logic and I/O pins.
	12,13, 33,34	13		NC			

TABLE 1-3: PIC18(L)F4XK22 PINOUT I/O DESCRIPTIONS (CONTINUED)

Legend: TTL = TTL compatible input CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; I = Input; O = Output; P = Power.

Note 1: Default pin assignment for P2B, T3CKI, CCP3/P3A and CCP2/P2A when Configuration bits PB2MX, T3CMX, CCP3MX and CCP2MX are set.

2: Alternate pin assignment for P2B, T3CKI, CCP3/P3A and CCP2/P2A when Configuration bits PB2MX, T3CMX, CCP3MX and CCP2MX are clear.

OSC Mode	OSC1 Pin	OSC2 Pin			
RC, INTOSC with CLKOUT	Floating, external resistor should pull high	At logic low (clock/4 output)			
RC with IO	Floating, external resistor should pull high	Configured as PORTA, bit 6			
INTOSC with IO	Configured as PORTA, bit 7	Configured as PORTA, bit 6			
EC with IO	Floating, pulled by external clock	Configured as PORTA, bit 6			
EC with CLKOUT	Floating, pulled by external clock	At logic low (clock/4 output)			
LP, XT, HS	Feedback inverter disabled at quiescent voltage level	Feedback inverter disabled at quiescent voltage level			

TABLE 2-3:OSC1 AND OSC2 PIN STATES IN SLEEP MODE

Note: See Table 4-2 in Section 4.0 "Reset" for time-outs due to Sleep and MCLR Reset.

2.11 Clock Switching

The system clock source can be switched between external and internal clock sources via software using the System Clock Select (SCS<1:0>) bits of the OSCCON register.

PIC18(L)F2X/4XK22 devices contain circuitry to prevent clock "glitches" when switching between clock sources. A short pause in the device clock occurs during the clock switch. The length of this pause is the sum of two cycles of the old clock source and three to four cycles of the new clock source. This formula assumes that the new clock source is stable.

Clock transitions are discussed in greater detail in **Section 3.1.2 "Entering Power-Managed Modes"**.

2.11.1 SYSTEM CLOCK SELECT (SCS<1:0>) BITS

The System Clock Select (SCS<1:0>) bits of the OSCCON register select the system clock source that is used for the CPU and peripherals.

- When SCS<1:0> = 00, the system clock source is determined by configuration of the FOSC<3:0> bits in the CONFIG1H Configuration register.
- When SCS<1:0> = 10, the system clock source is chosen by the internal oscillator frequency selected by the INTSRC bit of the OSCTUNE register, the MFIOSEL bit of the OSCCON2 register and the IRCF<2:0> bits of the OSCCON register.
- When SCS<1:0> = 01, the system clock source is the 32.768 kHz secondary oscillator shared with Timer1, Timer3 and Timer5.

After a Reset, the SCS<1:0> bits of the OSCCON register are always cleared.

Note: Any automatic clock switch, which may occur from Two-Speed Start-up or Fail-Safe Clock Monitor, does not update the SCS<1:0> bits of the OSCCON register. The user can monitor the SOSCRUN, MFIOFS and LFIOFS bits of the OSCCON2 register, and the HFIOFS and OSTS bits of the OSCCON register to determine the current system clock source.

2.11.2 OSCILLATOR START-UP TIME-OUT STATUS (OSTS) BIT

The Oscillator Start-up Time-out Status (OSTS) bit of the OSCCON register indicates whether the system clock is running from the external clock source, as defined by the FOSC<3:0> bits in the CONFIG1H Configuration register, or from the internal clock source. In particular, when the primary oscillator is the source of the primary clock, OSTS indicates that the Oscillator Start-up Timer (OST) has timed out for LP, XT or HS modes.

4.2 Register Definitions: Reset Control

REGISTER 4-1: RCON: RESET CONTROL REGISTER

R/W-0/	0 R/W-q/u	U-0	R/W-1/a	R-1/q	R-1/q	R/W-q/u	R/W-0/a				
IPEN	SBOREN ⁽¹⁾	_	RI	то	PD	POR ⁽²⁾	BOR				
bit 7	I						bit 0				
							,				
Legend:											
R = Reada	able bit	W = Writable	bit	U = Unimplemented bit, read as '0'							
'1' = Bit is	set	'0' = Bit is cle	ared	-n/n = Value at POR and BOR/Value at all other Resets							
x = Bit is	unknown	u = unchang	ed	q = depends	on condition						
bit 7	IPEN: Interrup 1 = Enable pr 0 = Disable pr	ot Priority Enat iority levels on riority levels on	ble bit interrupts i interrupts (P	IC16CXXX Co	mpatibility mode	•)					
bit 6	bit 6 $\begin{array}{l} \text{SBOREN: BOR Software Enable bit}^{(1)} \\ \hline \text{If BOREN<1:0> = 01:} \\ 1 = \text{BOR is enabled} \\ 0 = \text{BOR is disabled} \\ \hline \text{If BOREN<1:0> = 00, 10 or 11:} \\ \hline \text{Bit is disabled and read as '0'.} \end{array}$										
bit 5	Unimplemen	ted: Read as '	0'								
bit 4	RI: RESET INS	struction Flag b	oit								
	1 = The RESE 0 = The RESE code-exe	ET instruction v ET instruction cuted Reset of	vas not execu was executec ccurs)	ited (set by firm d causing a de	ware or Power- vice Reset (mu	on Reset) st be set in fin	mware after a				
bit 3	TO: Watchdog	g Time-out Flag	g bit								
	1 = Set by po 0 = A WDT ti	wer-up, CLRW	DT instruction ed	or SLEEP instr	uction						
bit 2	PD: Power-do	own Detection	Flag bit								
	1 = Set by po	ower-up or by t	he CLRWDT in	struction							
L	0 = Set by ex	ecution of the	SLEEP INStruc	Ction							
DIT		on Reset Statu	S DIT-								
	1 = NO POWer 0 0 = A Power 0	on Reset occu	rred (must be	set in software	after a Power-o	on Reset occur	s)				
bit 0	BOR: Brown-	out Reset State	us bit ⁽³⁾				- /				
	1 = A Brown- 0 = A Brown-	out Reset has out Reset occi	not occurred urred (must be	(set by firmwai e set by firmwa	e only) re after a POR o	or Brown-out R	eset occurs)				
Note 1:	When CONFIG2L[2:1] = 01, then	the SBOREN	Reset state is	; '1'; otherwise.	it is '0'.					
2:	The actual Reset v	alue of POR is	determined b	by the type of c	levice Reset. Se	e the notes fol	lowing this				

register and Section 4.7 "Reset State of Registers" for additional information.

3: See Table 4-1.

Note 1: Brown-out Reset is indicated when BOR is '0' and POR is '1' (assuming that both POR and BOR were set to '1' by firmware immediately after POR).

2: It is recommended that the POR bit be set after a Power-on Reset has been detected so that subsequent Power-on Resets may be detected.





6.2 Control Registers

Several control registers are used in conjunction with the TBLRD and TBLWT instructions. These include the:

- EECON1 register
- EECON2 register
- TABLAT register
- TBLPTR registers

6.2.1 EECON1 AND EECON2 REGISTERS

The EECON1 register (Register 6-1) is the control register for memory accesses. The EECON2 register is not a physical register; it is used exclusively in the memory write and erase sequences. Reading EECON2 will read all '0's.

The EEPGD control bit determines if the access will be a program or data EEPROM memory access. When EEPGD is clear, any subsequent operations will operate on the data EEPROM memory. When EEPGD is set, any subsequent operations will operate on the program memory.

The CFGS control bit determines if the access will be to the Configuration/Calibration registers or to program memory/data EEPROM memory. When CFGS is set, subsequent operations will operate on Configuration registers regardless of EEPGD (see **Section 24.0 "Special Features of the CPU"**). When CFGS is clear, memory selection access is determined by EEPGD. The FREE bit allows the program memory erase operation. When FREE is set, an erase operation is initiated on the next WR command. When FREE is clear, only writes are enabled.

The WREN bit, when set, will allow a write operation. The WREN bit is clear on power-up.

The WRERR bit is set by hardware when the WR bit is set and cleared when the internal programming timer expires and the write operation is complete.

Note: During normal operation, the WRERR is read as '1'. This can indicate that a write operation was prematurely terminated by a Reset, or a write operation was attempted improperly.

The WR control bit initiates write operations. The WR bit cannot be cleared, only set, by firmware. Then WR bit is cleared by hardware at the completion of the write operation.

Note: The EEIF interrupt flag bit of the PIR2 register is set when the write is complete. The EEIF flag stays set until cleared by firmware.

9.0 INTERRUPTS

The PIC18(L)F2X/4XK22 devices have multiple interrupt sources and an interrupt priority feature that allows most interrupt sources to be assigned a high or low priority level (INT0 does not have a priority bit, it is always a high priority). The high priority interrupt vector is at 0008h and the low priority interrupt vector is at 0018h. A high priority interrupt event will interrupt a low priority interrupt that may be in progress.

There are 19 registers used to control interrupt operation.

These registers are:

- INTCON, INTCON2, INTCON3
- PIR1, PIR2, PIR3, PIR4, PIR5
- PIE1, PIE2, PIE3, PIE4, PIE5
- IPR1, IPR2, IPR3, IPR4, IPR5
- RCON

It is recommended that the Microchip header files supplied with MPLAB[®] IDE be used for the symbolic bit names in these registers. This allows the assembler/ compiler to automatically take care of the placement of these bits within the specified register.

In general, interrupt sources have three bits to control their operation. They are:

- Flag bit to indicate that an interrupt event occurred
- Enable bit that allows program execution to branch to the interrupt vector address when the flag bit is set
- **Priority bit** to select high priority or low priority

9.1 Mid-Range Compatibility

When the IPEN bit is cleared (default state), the interrupt priority feature is disabled and interrupts are compatible with PIC[®] microcontroller mid-range devices. In Compatibility mode, the interrupt priority bits of the IPRx registers have no effect. The PEIE/GIEL bit of the INTCON register is the global interrupt enable for the peripherals. The PEIE/GIEL bit disables only the peripheral interrupt sources and enables the peripheral interrupt sources when the GIE/GIEH bit is also set. The GIE/GIEH bit of the INTCON register is the global interrupt sources and enables all non-peripheral interrupt sources and disables all interrupt sources, including the peripherals. All interrupts branch to address 0008h in Compatibility mode.

9.2 Interrupt Priority

The interrupt priority feature is enabled by setting the IPEN bit of the RCON register. When interrupt priority is enabled the GIE/GIEH and PEIE/GIEL global interrupt enable bits of Compatibility mode are replaced by the GIEH high priority, and GIEL low priority, global interrupt enables. When set, the GIEH bit of the INTCON register enables all interrupts that have their associated IPRx register or INTCONx register priority bit set (high priority). When clear, the GIEH bit disables all interrupt sources including those selected as low priority. When clear, the GIEL bit of the INTCON register disables only the interrupts that have their associated priority bit cleared (low priority). When set, the GIEL bit enables the low priority sources when the GIEH bit is also set.

When the interrupt flag, enable bit and appropriate Global Interrupt Enable (GIE) bit are all set, the interrupt will vector immediately to address 0008h for high priority, or 0018h for low priority, depending on level of the interrupting source's priority bit. Individual interrupts can be disabled through their corresponding interrupt enable bits.

9.3 Interrupt Response

When an interrupt is responded to, the Global Interrupt Enable bit is cleared to disable further interrupts. The GIE/GIEH bit is the Global Interrupt Enable when the IPEN bit is cleared. When the IPEN bit is set, enabling interrupt priority levels, the GIEH bit is the high priority global interrupt enable and the GIEL bit is the low priority Global Interrupt Enable. High priority interrupt sources can interrupt a low priority interrupt. Low priority interrupts are not processed while high priority interrupts are in progress.

The return address is pushed onto the stack and the PC is loaded with the interrupt vector address (0008h or 0018h). Once in the Interrupt Service Routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits in the INTCONx and PIRx registers. The interrupt flag bits must be cleared by software before re-enabling interrupts to avoid repeating the same interrupt.

The "return from interrupt" instruction, RETFIE, exits the interrupt routine and sets the GIE/GIEH bit (GIEH or GIEL if priority levels are used), which re-enables interrupts.

For external interrupt events, such as the INT pins or the PORTB interrupt-on-change, the interrupt latency will be three to four instruction cycles. The exact latency is the same for one-cycle or two-cycle instructions. Individual interrupt flag bits are set, regardless of the status of their corresponding enable bits or the Global Interrupt Enable bit.

13.6 Register Definitions: Timer2/4/6 Control

REGISTER 13-1: TxCON: TIMER2/TIMER4/TIMER6 CONTROL REGISTER

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0				
_		TxOUT	PS<3:0>		TMRxON	TxCKP	S<1:0>				
bit 7							bit (
Lowende											
L egena: R – Roadah	le hit	W – Writable	hit	II – I Inimple	mented hit read	as 'O'					
		v – Pit is upkr			at DOP and POI		other Decete				
	changeu	X = DILIS ULKI			at FOR and BOI	R/Value at all 0					
T = BIT IS Se	et	0 = Bit is clear	ared								
oit 7	Unimpleme	ented: Read as '	0'								
bit 6-3	TxOUTPS<	: 3:0>: TimerX Οι	utput Postscal	ler Select bits							
	0000 = 1:1	Postscaler									
	0001 = 1:2	Postscaler									
	0010 = 1:3	Postscaler									
	0011 = 1 :4	Postscaler									
	0100 = 1:5	0100 = 1:5 Postscaler									
	0101 = 1:6	Postscaler									
	0110 = 1:7	Postscaler									
	0111 = 1:8	0111 = 1:8 Postscaler									
	1000 = 1.9										
	1001 = 1.10										
	1010 - 1.11	2 Postscaler									
	1100 = 1.12	3 Postscaler									
	1100 = 1:14	4 Postscaler									
	1110 = 1:15	5 Postscaler									
	1111 = 1:16	1111 = 1:16 Postscaler									
oit 2	TMRxON:	FimerX On bit									
	1 = TimerX	1 = TimerX is on									
	0 = TimerX	is off									
bit 1-0	TxCKPS<1	:0>: Timer2-type	Clock Presc	ale Select bits							
	00 = Presca	aler is 1									
	01 = Presca	aler is 4									
	1 Dreese	- I									

14.1.4 CCP PRESCALER

There are four prescaler settings specified by the CCPxM<3:0> bits of the CCPxCON register. Whenever the CCP module is turned off, or the CCP module is not in Capture mode, the prescaler counter is cleared. Any Reset will clear the prescaler counter.

Switching from one capture prescaler to another does not clear the prescaler and may generate a false interrupt. To avoid this unexpected operation, turn the module off by clearing the CCPxCON register before changing the prescaler. Example 14-1 demonstrates the code to perform this function.

EXAMPLE 14-1: CHANGING BETWEEN CAPTURE PRESCALERS

#define NEW_CAPT_PS 0x06	//Capture
	// Prescale 4th
	// rising edge
CCPxCON = 0;	// Turn the CCP
	// Module Off
CCPxCON = NEW_CAPT_PS;	// Turn CCP module
	// on with new
	<pre>// prescale value</pre>

14.1.5 CAPTURE DURING SLEEP

Capture mode requires a 16-bit TimerX module for use as a time base. There are four options for driving the 16-bit TimerX module in Capture mode. It can be driven by the system clock (Fosc), the instruction clock (Fosc/ 4), or by the external clock sources, the Secondary Oscillator (Sosc), or the TxCKI clock input. When the 16-bit TimerX resource is clocked by Fosc or Fosc/4, TimerX will not increment during Sleep. When the device wakes from Sleep, TimerX will continue from its previous state. Capture mode will operate during Sleep when the 16-bit TimerX resource is clocked by one of the external clock sources (Sosc or the TxCKI pin).

TABLE 14-3: REGISTERS ASSOCIATED WITH CAPTURE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page		
CCP1CON	P1M-	<1:0>	DC1B-	<1:0>	CCP1M<3:0>				198		
CCP2CON	P2M	<1:0>	DC2B	<1:0>		198					
CCP3CON	P3M-	<1:0>	DC3B	<1:0>		198					
CCP4CON	—	—	DC4B	<1:0>		CCP4M<	3:0>		198		
CCP5CON	_	_	DC5B-	<1:0>		CCP5M<	3:0>		198		
CCPR1H			Capture/Co	mpare/PWM F	Register 1 High By	te (MSB)			_		
CCPR1L			Capture/Co	mpare/PWM	Register 1 Low By	rte (LSB)			_		
CCPR2H	Capture/Compare/PWM Register 2 High Byte (MSB)										
CCPR2L	Capture/Compare/PWM Register 2 Low Byte (LSB)										
CCPR3H	Capture/Compare/PWM Register 3 High Byte (MSB)										
CCPR3L	Capture/Compare/PWM Register 3 Low Byte (LSB)										
CCPR4H			Capture/Co	mpare/PWM F	Register 4 High By	te (MSB)			—		
CCPR4L			Capture/Co	mpare/PWM	Register 4 Low By	rte (LSB)			—		
CCPR5H			Capture/Co	mpare/PWM F	Register 5 High By	te (MSB)			—		
CCPR5L			Capture/Co	mpare/PWM	Register 5 Low By	rte (LSB)			—		
CCPTMRS0	C3TSE	L<1:0>	_	C2TS	SEL<1:0>	—	C1TSEL	_<1:0>	201		
CCPTMRS1	—	—	_	_	C5TSEL∢	<1:0>	C4TSEL	_<1:0>	201		
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INTOIE	RBIE	TMR0IF	INTOIF	RBIF	109		
IPR1	_	ADIP	RC1IP	TX1IP	SSP1IP	CCP1IP	TMR2IP	TMR1IP	121		
IPR2	OSCFIP	C1IP	C2IP	EEIP	BCL1IP	HLVDIP	TMR3IP	CCP2IP	122		
IPR4	_	—	—	—	_	CCP5IP	CCP4IP	CCP3IP	124		
PIE1	_	ADIE	RC1IE	TX1IE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	117		

Legend: — = Unimplemented location, read as '0'. Shaded bits are not used by Capture mode.

Note 1: These registers/bits are available on PIC18(L)F4XK22 devices.

15.2.4 SPI SLAVE MODE

In Slave mode, the data is transmitted and received as external clock pulses appear on SCKx. When the last bit is latched, the SSPxIF interrupt flag bit is set.

Before enabling the module in SPI Slave mode, the clock line must match the proper Idle state. The clock line can be observed by reading the SCKx pin. The Idle state is determined by the CKP bit of the SSPxCON1 register.

While in Slave mode, the external clock is supplied by the external clock source on the SCKx pin. This external clock must meet the minimum high and low times as specified in the electrical specifications.

While in Sleep mode, the slave can transmit/receive data. The shift register is clocked from the SCKx pin input and when a byte is received, the device will generate an interrupt. If enabled, the device will wake up from Sleep.

15.2.4.1 Daisy-Chain Configuration

The SPI bus can sometimes be connected in a daisychain configuration. The first slave output is connected to the second slave input, the second slave output is connected to the third slave input, and so on. The final slave output is connected to the master input. Each slave sends out, during a second group of clock pulses, an exact copy of what was received during the first group of clock pulses. The whole chain acts as one large communication shift register. The daisychain feature only requires a single Slave Select line from the master device.

Figure 15-7 shows the block diagram of a typical daisy-chain connection when operating in SPI Mode.

In a daisy-chain configuration, only the most recent byte on the bus is required by the slave. Setting the BOEN bit of the SSPxCON3 register will enable writes to the SSPxBUF register, even if the previous byte has not been read. This allows the software to ignore data that may not apply to it.

15.2.5 SLAVE SELECT SYNCHRONIZATION

The Slave Select can also be used to synchronize communication. The Slave Select line is held high until the master device is ready to communicate. When the Slave Select line is pulled low, the slave knows that a new transmission is starting.

If the slave fails to receive the communication properly, it will be reset at the end of the transmission, when the Slave Select line returns to a high state. The slave is then ready to receive a new transmission when the Slave Select line is pulled low again. If the Slave Select line is not used, there is a risk that the slave will eventually become out of sync with the master. If the slave misses a bit, it will always be one bit off in future transmissions. Use of the Slave Select line allows the slave and master to align themselves at the beginning of each transmission (Figure 15-8).

The \overline{SSx} pin allows a Synchronous Slave mode. The SPI must be in Slave mode with \overline{SSx} pin control enabled (SSPxCON1<3:0> = 0100).

When the \overline{SSx} pin is low, transmission and reception are enabled and the SDOx pin is driven.

When the \overline{SSx} pin goes high, the SDOx pin is no longer driven, even if in the middle of a transmitted byte and becomes a floating output. External pull-up/pull-down resistors may be desirable depending on the application.

- Note 1: When the SPI is in Slave mode with SSx pin control enabled (SSPxCON1<3:0> = 0100), the SPI module will reset if the SSx pin is set to VDD.
 - 2: When the SPI is used in Slave mode with CKE set; the user must enable SSx pin control.
 - **3:** While operated in SPI Slave mode the SMP bit of the SSPxSTAT register must remain clear.

When the SPI module resets, the bit counter is forced to '0'. This can be done by either forcing the SSx pin to a high level or clearing the SSPxEN bit.

15.5.3.3 7-bit Transmission with Address Hold Enabled

Setting the AHEN bit of the SSPxCON3 register enables additional clock stretching and interrupt generation after the 8th falling edge of a received matching address. Once a matching address has been clocked in, CKP is cleared and the SSPxIF interrupt is set.

Figure 15-19 displays a standard waveform of a 7-bit Address Slave Transmission with AHEN enabled.

- 1. Bus starts Idle.
- Master sends Start condition; the S bit of SSPxSTAT is set; SSPxIF is set if interrupt on Start detect is enabled.
- Master sends matching address with R/W bit set. After the 8th falling edge of the SCLx line the CKP bit is cleared and SSPxIF interrupt is generated.
- 4. Slave software clears SSPxIF.
- Slave software reads ACKTIM bit of SSPxCON3 register, and R/W and D/A of the SSPxSTAT register to determine the source of the interrupt.
- 6. Slave reads the address value from the SSPxBUF register clearing the BF bit.
- Slave software decides from this information if it wishes to ACK or not ACK and sets ACKDT bit of the SSPxCON2 register accordingly.
- 8. Slave sets the CKP bit releasing SCLx.
- 9. Master clocks in the \overline{ACK} value from the slave.
- 10. Slave hardware automatically clears the CKP bit and sets SSPxIF after the ACK if the R/W bit is set.
- 11. Slave software clears SSPxIF.
- 12. Slave loads value to transmit to the master into SSPxBUF setting the BF bit.

Note: <u>SSPxBUF</u> cannot be loaded until after the ACK.

- 13. Slave sets CKP bit releasing the clock.
- 14. Master clocks out the data from the slave and sends an ACK value on the 9th SCLx pulse.
- 15. Slave hardware copies the ACK value into the ACKSTAT bit of the SSPxCON2 register.
- 16. Steps 10-15 are repeated for each byte transmitted to the master from the slave.
- 17. If the master sends a not ACK the slave releases the bus allowing the master to send a Stop and end the communication.

Note: Master must send a not ACK on the last byte to ensure that the slave releases the SCLx line to receive a Stop.

15.6.8 ACKNOWLEDGE SEQUENCE TIMING

An Acknowledge sequence is enabled by setting the Acknowledge Sequence Enable bit, ACKEN, of the SSPxCON2 register. When this bit is set, the SCLx pin is pulled low and the contents of the Acknowledge data bit are presented on the SDAx pin. If the user wishes to generate an Acknowledge, then the ACKDT bit should be cleared. If not, the user should set the ACKDT bit before starting an Acknowledge sequence. The Baud Rate Generator then counts for one rollover period (TBRG) and the SCLx pin is deasserted (pulled high). When the SCLx pin is sampled high (clock arbitration), the Baud Rate Generator counts for TBRG. The SCLx pin is then pulled low. Following this, the ACKEN bit is automatically cleared, the Baud Rate Generator is turned off and the MSSPx module then goes into Idle mode (Figure 15-30).

15.6.8.1 WCOL Status Flag

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

15.6.9 STOP CONDITION TIMING

A Stop bit is asserted on the SDAx pin at the end of a receive/transmit by setting the Stop Sequence Enable bit, PEN, of the SSPxCON2 register. At the end of a receive/transmit, the SCLx line is held low after the falling edge of the ninth clock. When the PEN bit is set, the master will assert the SDAx line low. When the SDAx line is sampled low, the Baud Rate Generator is reloaded and counts down to '0'. When the Baud Rate Generator times out, the SCLx pin will be brought high and one TBRG (Baud Rate Generator rollover count) later, the SDAx pin will be deasserted. When the SDAx pin is sampled high while SCLx is high, the P bit of the SSPxSTAT register is set. A TBRG later, the PEN bit is cleared and the SSPxIF bit is set (Figure 15-31).

15.6.9.1 WCOL Status Flag

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

FIGURE 15-30: ACKNOWLEDGE SEQUENCE WAVEFORM



17.1 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
- Results formatting

17.1.1 PORT CONFIGURATION

The ANSELx and TRISx registers configure the A/D port pins. Any port pin needed as an analog input should have its corresponding ANSx bit set to disable the digital input buffer and TRISx bit set to disable the digital output driver. If the TRISx bit is cleared, the digital output level (VOH or VOL) will be converted.

The A/D operation is independent of the state of the ANSx bits and the TRIS bits.

- Note 1: When reading the PORT register, all pins with their corresponding ANSx bit set read as cleared (a low level). However, analog conversion of pins configured as digital inputs (ANSx bit cleared and TRISx bit set) will be accurately converted.
 - 2: Analog levels on any pin with the corresponding ANSx bit cleared may cause the digital input buffer to consume current out of the device's specification limits.
 - 3: The PBADEN bit in Configuration Register 3H configures PORTB pins to reset as analog or digital pins by controlling how the bits in ANSELB are reset.

17.1.2 CHANNEL SELECTION

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 17.2** "**ADC Operation**" for more information.

17.1.3 ADC VOLTAGE REFERENCE

The PVCFG<1:0> and NVCFG<1:0> bits of the ADCON1 register provide independent control of the positive and negative voltage references.

The positive voltage reference can be:

- Vdd
- the fixed voltage reference (FVR BUF2)
- an external voltage source (VREF+)

The negative voltage reference can be:

- Vss
- an external voltage source (VREF-)

17.1.4 SELECTING AND CONFIGURING ACQUISITION TIME

The ADCON2 register allows the user to select an acquisition time that occurs each time the GO/\overline{DONE} bit is set.

Acquisition time is set with the ACQT<2:0> bits of the ADCON2 register. Acquisition delays cover 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 is 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> = 000. 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. When an acquisition time is programmed, there is no indication of when the acquisition time ends and the conversion begins.

19.4 Measuring Capacitance with the CTMU

There are two separate methods of measuring capacitance with the CTMU. The first is the absolute method, in which the actual capacitance value is desired. The second is the relative method, in which the actual capacitance is not needed, rather an indication of a change in capacitance is required.

19.4.1 ABSOLUTE CAPACITANCE MEASUREMENT

For absolute capacitance measurements, both the current and capacitance calibration steps found in **Section 19.3 "Calibrating the CTMU Module"** should be followed. Capacitance measurements are then performed using the following steps:

- 1. Initialize the A/D Converter.
- 2. Initialize the CTMU.
- 3. Set EDG1STAT.
- 4. Wait for a fixed delay, *T*.
- 5. Clear EDG1STAT.
- 6. Perform an A/D conversion.
- 7. Calculate the total capacitance, CTOTAL = (I * T)/V, where *I* is known from the current source measurement step (see **Section 19.3.1 "Current Source Calibration"**), *T* is a fixed delay and *V* is measured by performing an A/D conversion.
- 8. Subtract the stray and A/D capacitance (*C*OFFSET from **Section 19.3.2** "**Capacitance Calibration**") from *CTOTAL* to determine the measured capacitance.

19.4.2 RELATIVE CHARGE MEASUREMENT

An application may not require precise capacitance measurements. For example, when detecting a valid press of a capacitance-based switch, detecting a relative change of capacitance is of interest. In this type of application, when the switch is open (or not touched), the total capacitance is the capacitance of the combination of the board traces, the A/D Converter, etc. A larger voltage will be measured by the A/D Converter. When the switch is closed (or is touched), the total capacitance is larger due to the addition of the capacitances, and a smaller voltage will be measured by the A/D Converter.

Detecting capacitance changes is easily accomplished with the CTMU using these steps:

- 1. Initialize the A/D Converter and the CTMU.
- 2. Set EDG1STAT.
- 3. Wait for a fixed delay.
- 4. Clear EDG1STAT.
- 5. Perform an A/D conversion.

The voltage measured by performing the A/D conversion is an indication of the relative capacitance. Note that in this case, no calibration of the current source or circuit capacitance measurement is needed. See Example 19-4 for a sample software routine for a capacitive touch switch.

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0		
—	—	—			DACR<4:0>				
bit 7							bit 0		
Legend:									
R = Readable	bit	W = Writable	bit	U = Unimplemented bit, read as '0'					
u = Bit is unchanged x = Bit is unknown			-n/n = Value at POR and BOR/Value at all other Resets						
'1' = Bit is set		'0' = Bit is clea	ared						

REGISTER 22-2: VREFCON2: VOLTAGE REFERENCE CONTROL REGISTER 1

bit 7-5 Unimplemented: Read as '0'

bit 4-0 DACR<4:0>: DAC Voltage Output Select bits VOUT = ((VSRC+) - (VSRC-))*(DACR<4:0>/(2⁵)) + VSRC-

TABLE 22-1: REGISTERS ASSOCIATED WITH DAC MODULE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page	
VREFCON0	FVREN	FVRST	FVRS	FVRS<1:0>		—	—	—	332	
VREFCON1	DACEN	DACLPS	DACOE	—	DACPS	DACPSS<1:0>		DACNSS	335	
VREFCON2	—	—	—		DACR<4:0>					

Legend: — = Unimplemented locations, read as '0'. Shaded bits are not used by the DAC module.

25.2 Extended Instruction Set

In addition to the standard 75 instructions of the PIC18 instruction set, PIC18(L)F2X/4XK22 devices also provide an optional extension to the core CPU functionality. The added features include eight additional instructions that augment indirect and indexed addressing operations and the implementation of Indexed Literal Offset Addressing mode for many of the standard PIC18 instructions.

The additional features of the extended instruction set are disabled by default. To enable them, users must set the XINST Configuration bit.

The instructions in the extended set can all be classified as literal operations, which either manipulate the File Select Registers, or use them for indexed addressing. Two of the instructions, ADDFSR and SUBFSR, each have an additional special instantiation for using FSR2. These versions (ADDULNK and SUBULNK) allow for automatic return after execution.

The extended instructions are specifically implemented to optimize re-entrant program code (that is, code that is recursive or that uses a software stack) written in high-level languages, particularly C. Among other things, they allow users working in high-level languages to perform certain operations on data structures more efficiently. These include:

- dynamic allocation and deallocation of software stack space when entering and leaving subroutines
- function pointer invocation
- software Stack Pointer manipulation
- manipulation of variables located in a software stack

A summary of the instructions in the extended instruction set is provided in Table 25-3. Detailed descriptions are provided in **Section 25.2.2 "Extended Instruction Set**". The opcode field descriptions in Table 25-1 apply to both the standard and extended PIC18 instruction sets.

Note: The instruction set extension and the Indexed Literal Offset Addressing mode were designed for optimizing applications written in C; the user may likely never use these instructions directly in assembler. The syntax for these commands is provided as a reference for users who may be reviewing code that has been generated by a compiler.

25.2.1 EXTENDED INSTRUCTION SYNTAX

Most of the extended instructions use indexed arguments, using one of the File Select Registers and some offset to specify a source or destination register. When an argument for an instruction serves as part of indexed addressing, it is enclosed in square brackets ("[]"). This is done to indicate that the argument is used as an index or offset. MPASM[™] Assembler will flag an error if it determines that an index or offset value is not bracketed.

When the extended instruction set is enabled, brackets are also used to indicate index arguments in byteoriented and bit-oriented instructions. This is in addition to other changes in their syntax. For more details, see Section 25.2.3.1 "Extended Instruction Syntax with Standard PIC18 Commands".

Note: In the past, square brackets have been used to denote optional arguments in the PIC18 and earlier instruction sets. In this text and going forward, optional arguments are denoted by braces ("{ }").

Mnemo	Mnemonic, Operands	Description	Cyclos	16-	Bit Instru	uction W	ord	Status	
Opera	nds	Description	Cycles	MSb			LSb	Affected	
ADDFSR	f, k	Add literal to FSR	1	1110	1000	ffkk	kkkk	None	
ADDULNK	k	Add literal to FSR2 and return	2	1110	1000	11kk	kkkk	None	
CALLW		Call subroutine using WREG	2	0000	0000	0001	0100	None	
MOVSF	z _s , f _d	Move z _s (source) to 1st word	2	1110	1011	0zzz	ZZZZ	None	
		f _d (destination) 2nd word		1111	ffff	ffff	ffff		
MOVSS	z _s , z _d	Move z _s (source) to 1st word	2	1110	1011	lzzz	ZZZZ	None	
		z _d (destination) 2nd word		1111	XXXX	XZZZ	ZZZZ		
PUSHL	k	Store literal at FSR2,	1	1110	1010	kkkk	kkkk	None	
		decrement FSR2							
SUBESR	f, K	Subtract literal from FSR	1	1110	1001	ÍÍKK	kkkk	None	
SUBULNK	k	Subtract literal from FSR2 and	2	1110	1001	11kk	kkkk	None	
		return							

TABLE 25-3: EXTENSIONS TO THE PIC18 INSTRUCTION SET

PIC18LF2X/4XK22		Standard Operating Conditions (unless otherwise stated)Operating temperature $-40^{\circ}C \le TA \le +125^{\circ}C$					
PIC18F2X/4XK22		Standard Operating Conditions (unless otherwise stated)Operating temperature $-40^{\circ}C \le TA \le +125^{\circ}C$					
Param No.	Device Characteristics	Тур	Max	Units	Conditions		
D055		0.25	0.40	mA	-40°C to +125°C	VDD = 1.8V	Fosc = 1 MHz (RC_IDLE mode, HFINTOSC source)
D056		0.35	0.50	mA	-40°C to +125°C	VDD = 3.0V	
D057		0.30	0.45	mA	-40°C to +125°C	VDD = 2.3V	Fosc = 1 MHz (RC_IDLE mode, HFINTOSC source)
D058		0.40	0.50	mA	-40°C to +125°C	VDD = 3.0V	
D059		0.45	0.60	mA	-40°C to +125°C	Vdd = 5.0V	
D060		0.50	0.7	mA	-40°C to +125°C	VDD = 1.8V	Fosc = 16 MHz (RC_IDLE mode, HFINTOSC source)
D061		0.80	1.1	mA	-40°C to +125°C	VDD = 3.0V	
D062		0.65	1.0	mA	-40°C to +125°C	VDD = 2.3V	Fosc = 16 MHz (RC_IDLE mode, HFINTOSC source)
D063		0.80	1.1	mA	-40°C to +125°C	VDD = 3.0V	
D064		0.95	1.2	mA	-40°C to +125°C	VDD = 5.0V	
D066		2.5	3.5	mA	-40°C to +125°C	VDD = 3.0V	Fosc = 64 MHz (RC_IDLE mode, HFINTOSC + PLL source)
D068		2.5	3.5	mA	-40°C to +125°C	VDD = 3.0V	Fosc = 64 MHz (RC_IDLE mode, HFINTOSC + PLL source)
D069		3.0	4.5	mA	-40°C to +125°C	VDD = 5.0V	

27.4 DC Characteristics: RC Idle Supply Current, PIC18(L)F2X/4XK22 (Continued)

Note 1: The supply current is mainly a function of operating voltage, frequency and mode. Other factors, such as I/O pin loading and switching rate, oscillator type and circuit, internal code execution pattern and temperature, also have an impact on the current consumption.

Test condition: All Peripheral Module Control bits in PMD0, PMD1 and PMD2 set to '1'.

2: The test conditions for all IDD measurements in active operation mode are:

All I/O pins set as outputs driven to Vss;

OSC1 = external square wave, from rail-to-rail (PRI_RUN and PRI_IDLE only).

PIC18(L)F2X/4XK22









FIGURE 28-101: PIC18LF2X/4XK22 TYPICAL LF-INTOSC FREQUENCY vs. VDD Min/Max = 31.25 kHz ± 15%, T = -40°C to +85°C





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

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



Microchip Technology Drawing C04-105C Sheet 1 of 2