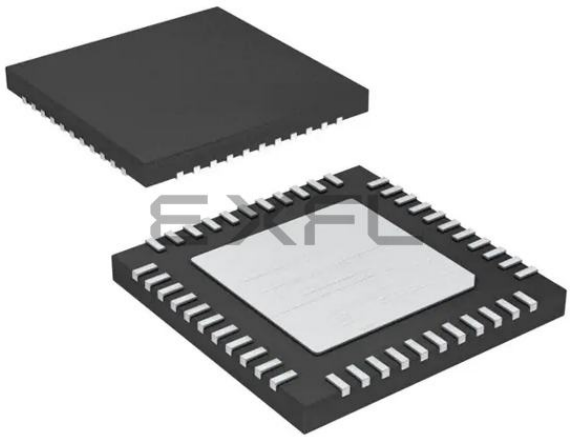


Welcome to E-XFL.COM

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 "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	64MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, HLVD, POR, PWM, WDT
Number of I/O	35
Program Memory Size	16KB (8K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	768 x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 30x10b
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/pic18lf44k22t-i-ml

FIGURE 5: 44-PIN TQFP DIAGRAM

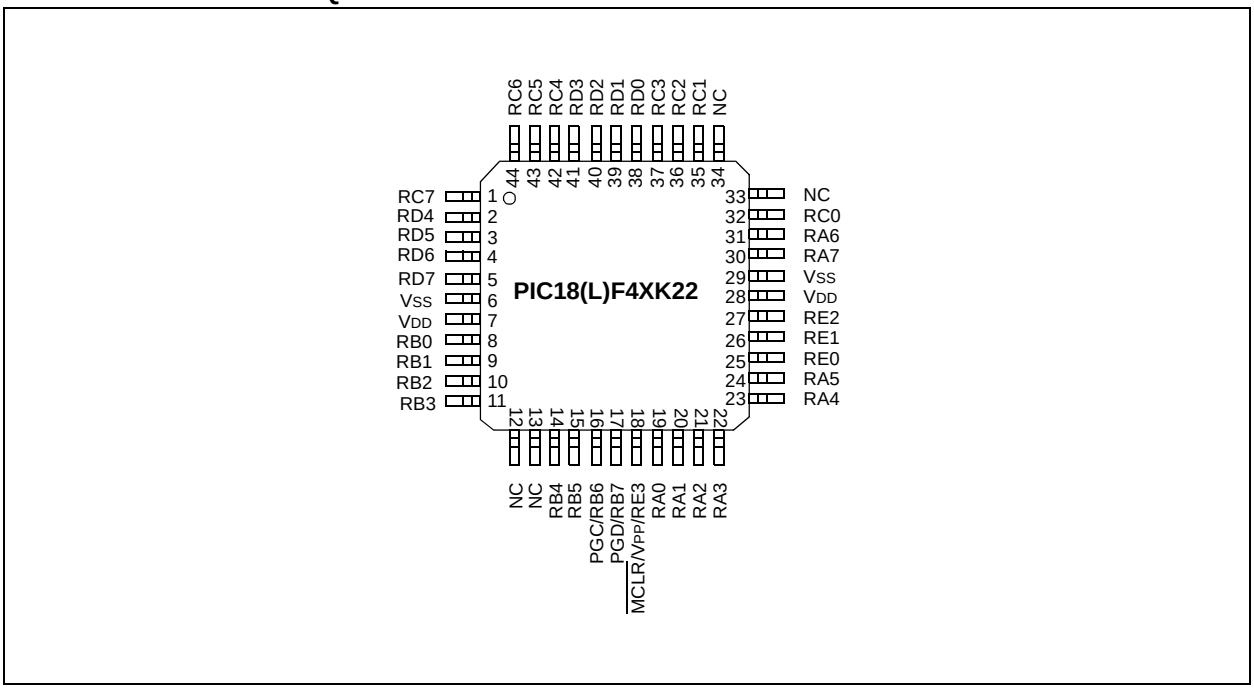


FIGURE 6: 44-PIN QFN DIAGRAM

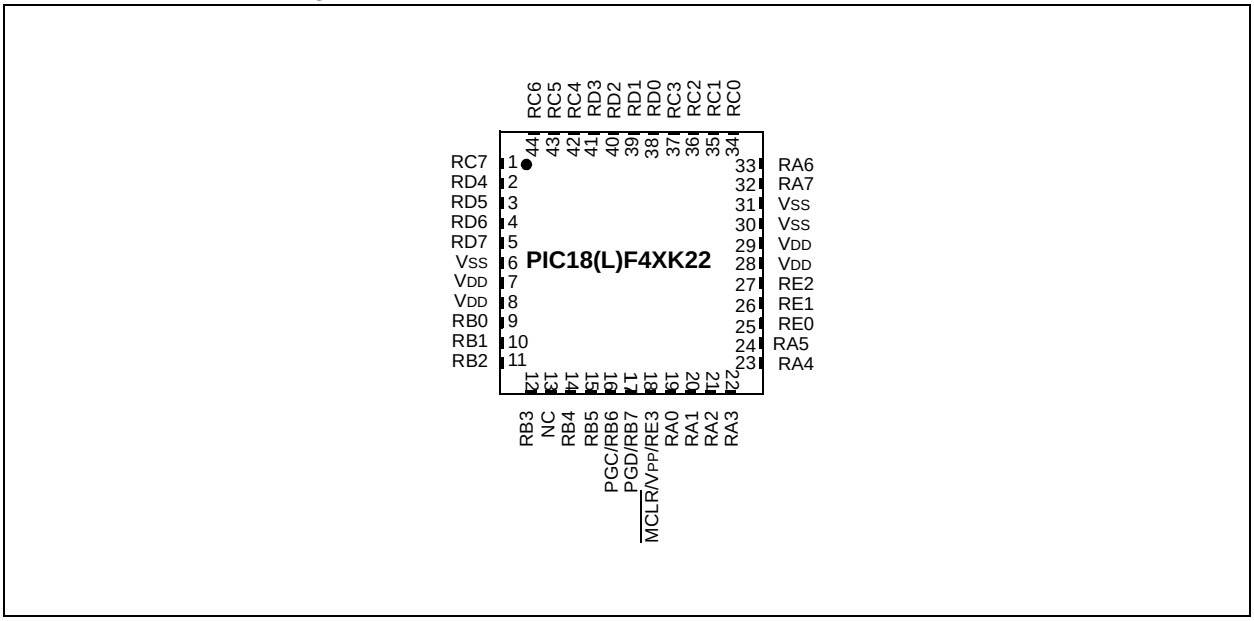


TABLE 1-3: PIC18(L)F4XK22 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Number				Pin Name	Pin Type	Buffer Type	Description
PDIP	TQFP	QFN	UQFN				
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	I	Analog	Analog input 7.
1	18	18	16	RE3/VPP/$\overline{\text{MCLR}}$			
				RE3	I	ST	Digital input.
				VPP	P		Programming voltage input.
				$\overline{\text{MCLR}}$	I	ST	Active-low Master Clear (device Reset) input.
11,32	7, 28	7, 8, 28, 29	7, 26	VDD	P	—	Positive supply for logic and I/O pins.
12,31	6, 29	6,30, 31	6, 27	VSS	P	—	Ground reference for logic and I/O pins.
	12,13, 33,34	13		NC			

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.

2.7.1 LFINTOSC

The Low-Frequency Internal Oscillator (LFINTOSC) is a 31.25 kHz internal clock source. The LFINTOSC is not tunable, but is designed to be stable across temperature and voltage. See **Section 27.0 "Electrical Specifications"** for the LFINTOSC accuracy specifications.

The output of the LFINTOSC can be a clock source to the primary clock or the INTOSC clock (see Figure 2-1). The LFINTOSC is also the clock source for the Power-up Timer (PWRT), Watchdog Timer (WDT) and Fail-Safe Clock Monitor (FSCM).

2.7.2 FREQUENCY SELECT BITS (IRCF)

The HFINTOSC (16 MHz) and MFINTOSC (500 MHz) outputs connect to a divide circuit that provides frequencies of 16 MHz to 31.25 kHz. These divide circuit frequencies, along with the 31.25 kHz LFINTOSC output, are multiplexed to provide a single INTOSC clock output (see Figure 2-1). The IRCF<2:0> bits of the OSCCON register, the MFIOSEL bit of the OSCCON2 register and the INTSRC bit of the OSCTUNE register, select the output frequency of the internal oscillators. One of eight frequencies can be selected via software:

- 16 MHz
- 8 MHz
- 4 MHz
- 2 MHz
- 1 MHz (default after Reset)
- 500 kHz (MFINTOSC or HFINTOSC)
- 250 kHz (MFINTOSC or HFINTOSC)
- 31 kHz (LFINTOSC, MFINTOSC or HFINTOSC)

2.7.3 INTOSC FREQUENCY DRIFT

The factory calibrates the internal oscillator block outputs (HFINTOSC/MFINTOSC) for 16 MHz/500 kHz. However, this frequency may drift as V_{DD} or temperature changes. It is possible to adjust the HFINTOSC/MFINTOSC frequency by modifying the value of the TUN<5:0> bits in the OSCTUNE register. This has no effect on the LFINTOSC clock source frequency.

Tuning the HFINTOSC/MFINTOSC source requires knowing when to make the adjustment, in which direction it should be made and, in some cases, how large a change is needed. Three possible compensation techniques are discussed in the following sections. However, other techniques may be used.

2.7.3.1 Compensating with the EUSART

An adjustment may be required when the EUSART begins to generate framing errors or receives data with errors while in Asynchronous mode. Framing errors indicate that the device clock frequency is too high; to adjust for this, decrement the value in OSCTUNE to reduce the clock frequency. On the other hand, errors in data may suggest that the clock speed is too low; to compensate, increment OSCTUNE to increase the clock frequency.

2.7.3.2 Compensating with the Timers

This technique compares device clock speed to some reference clock. Two timers may be used; one timer is clocked by the peripheral clock, while the other is clocked by a fixed reference source, such as the Timer1 oscillator.

Both timers are cleared, but the timer clocked by the reference generates interrupts. When an interrupt occurs, the internally clocked timer is read and both timers are cleared. If the internally clocked timer value is greater than expected, then the internal oscillator block is running too fast. To adjust for this, decrement the OSCTUNE register.

2.7.3.3 Compensating with the CCP Module in Capture Mode

A CCP module can use free running Timer1, Timer3 or Timer5 clocked by the internal oscillator block and an external event with a known period (i.e., AC power frequency). The time of the first event is captured in the CCPRxH:CCPRxL registers and is recorded for use later. When the second event causes a capture, the time of the first event is subtracted from the time of the second event. Since the period of the external event is known, the time difference between events can be calculated.

If the measured time is much greater than the calculated time, the internal oscillator block is running too fast; to compensate, decrement the OSCTUNE register. If the measured time is much less than the calculated time, the internal oscillator block is running too slow; to compensate, increment the OSCTUNE register.

TABLE 10-2: REGISTERS ASSOCIATED WITH PORTA

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
ANSELA	—	—	ANSA5	—	ANSA3	ANSA2	ANSA1	ANSA0	149
CM1CON0	C1ON	C1OUT	C1OE	C1POL	C1SP	C1R	C1CH<1:0>		308
CM2CON0	C2ON	C2OUT	C2OE	C2POL	C2SP	C2R	C2CH<1:0>		308
LATA	LATA7	LATA6	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0	152
VREFCON1	DACEN	DACLPS	DACOE	—	DACPSS<1:0>		—	DACNSS	335
VREFCON2	—	—	—	DACR<4:0>					336
HLVDCON	VDIRMAG	BGVST	IRVST	HLVDEN	HLVDL<3:0>				337
PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	148
SLRCON	—	—	—	SLRE	SLRD	SLRC	SLRB	SLRA	153
SRCON0	SRLEN	SRCLK<2:0>			SRQEN	SRNQEN	SRPS	SRPR	329
SSP1CON1	WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				253
T0CON	TMR0ON	T08BIT	T0CS	T0SE	PSA	T0PS<2:0>			154
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	151

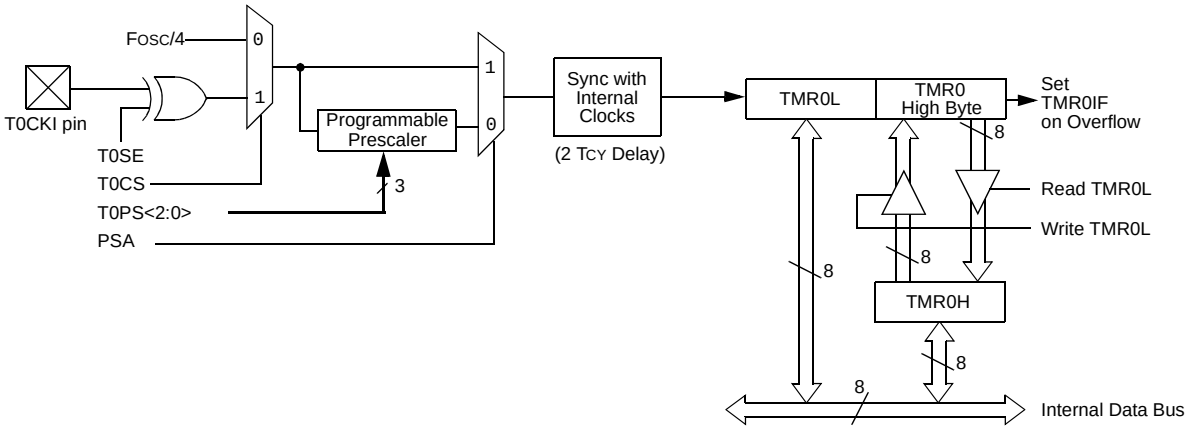
Legend: — = unimplemented locations, read as '0'. Shaded bits are not used for PORTA.

TABLE 10-3: CONFIGURATION REGISTERS ASSOCIATED WITH PORTA

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
CONFIG1H	IESO	FCMEN	PRICLKEN	PLLCFG	FOSC<3:0>				345

Legend: — = unimplemented locations, read as '0'. Shaded bits are not used for PORTA.

FIGURE 11-2: TIMER0 BLOCK DIAGRAM (16-BIT MODE)



Note: Upon Reset, Timer0 is enabled in 8-bit mode with clock input from T0CKI max. prescale.

11.4 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module. The prescaler is not directly readable or writable; its value is set by the PSA and T0PS<2:0> bits of the T0CON register which determine the prescaler assignment and prescale ratio.

Clearing the PSA bit assigns the prescaler to the Timer0 module. When the prescaler is assigned, prescale values from 1:2 through 1:256 in integer power-of-2 increments are selectable.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g., `CLRF TMR0`, `MOVWF TMR0`, `BSF TMR0`, etc.) clear the prescaler count.

Note:	Writing to TMR0 when the prescaler is assigned to Timer0 will clear the prescaler count but will not change the prescaler assignment.

11.4.1 SWITCHING PRESCALER ASSIGNMENT

The prescaler assignment is fully under software control and can be changed “on-the-fly” during program execution.

11.5 Timer0 Interrupt

The TMR0 interrupt is generated when the TMR0 register overflows from FFh to 00h in 8-bit mode, or from FFFFh to 0000h in 16-bit mode. This overflow sets the TMR0IF flag bit. The interrupt can be masked by clearing the TMR0IE bit of the INTCON register. Before re-enabling the interrupt, the TMR0IF bit must be cleared by software in the Interrupt Service Routine.

Since Timer0 is shut down in Sleep mode, the TMR0 interrupt cannot awaken the processor from Sleep.

TABLE 11-1: REGISTERS ASSOCIATED WITH TIMER0

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on page
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	109
INTCON2	$\overline{\text{RBPU}}$	INTEDG0	INTEDG1	INTEDG2	—	TMR0IP	—	RBIP	110
T0CON	TMR0ON	T08BIT	T0CS	T0SE	PSA	T0PS<2:0>			154
TMR0H	Timer0 Register, High Byte								—
TMR0L	Timer0 Register, Low Byte								—
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	151

Legend: — = unimplemented locations, read as ‘0’. Shaded bits are not used by Timer0.

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
                            // prescale value
```

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>		CCP2M<3:0>				198
CCP3CON	P3M<1:0>		DC3B<1:0>		CCP3M<3:0>				198
CCP4CON	—	—	DC4B<1:0>		CCP4M<3:0>				198
CCP5CON	—	—	DC5B<1:0>		CCP5M<3:0>				198
CCPR1H	Capture/Compare/PWM Register 1 High Byte (MSB)								—
CCPR1L	Capture/Compare/PWM Register 1 Low Byte (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/Compare/PWM Register 4 High Byte (MSB)								—
CCPR4L	Capture/Compare/PWM Register 4 Low Byte (LSB)								—
CCPR5H	Capture/Compare/PWM Register 5 High Byte (MSB)								—
CCPR5L	Capture/Compare/PWM Register 5 Low Byte (LSB)								—
CCPTMRS0	C3TSEL<1:0>		—	C2TSEL<1:0>		—	C1TSEL<1:0>		201
CCPTMRS1	—	—	—	—	C5TSEL<1:0>		C4TSEL<1:0>		201
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	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.

14.2 Compare Mode

The Compare mode function described in this section is identical for all CCP and ECCP modules available on this device family.

Compare mode makes use of the 16-bit TimerX resources, Timer1, Timer3 and Timer5. The 16-bit value of the CCPRxH:CCPRxL register pair is constantly compared against the 16-bit value of the TMRxH:TMRxL register pair. When a match occurs, one of the following events can occur:

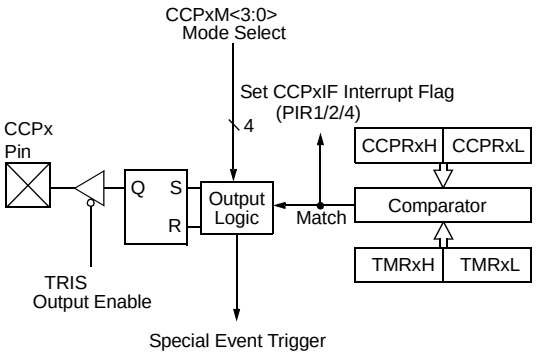
- Toggle the CCPx output
- Set the CCPx output
- Clear the CCPx output
- Generate a Special Event Trigger
- Generate a Software Interrupt

The action on the pin is based on the value of the CCPxM<3:0> control bits of the CCPxCON register. At the same time, the interrupt flag CCPxIF bit is set.

All Compare modes can generate an interrupt.

Figure 14-2 shows a simplified diagram of the Compare operation.

FIGURE 14-2: COMPARE MODE OPERATION BLOCK DIAGRAM



Special Event Trigger function on

- ECCP1, ECCP2, ECCP3, CCP4 and CCP5 will:
  - Reset TimerX – TMRxH:TMRxL = 0x0000
  - TimerX Interrupt Flag, (TMRxIF) is not set

Additional Function on

- CCP5 will
  - Set ADCON0<1>, GO/$\overline{\text{DONE}}$ bit to start an ADC Conversion if ADCON<0>, ADON = 1.

14.2.1 CCP PIN CONFIGURATION

The user must configure the CCPx pin as an output by clearing the associated TRIS bit.

Some CCPx outputs are multiplexed on a couple of pins. Table 14-2 shows the CCP output pin Multiplexing. Selection of the output pin is determined by the CCPxMX bits in Configuration register 3H (CONFIG3H). Refer to Register 24-4 for more details.

Note:	Clearing the CCPxCON register will force the CCPx compare output latch to the default low level. This is not the PORT I/O data latch.

14.2.2 TimerX MODE RESOURCE

In Compare mode, 16-bit TimerX resource must be running in either Timer mode or Synchronized Counter mode. The compare operation may not work in Asynchronous Counter mode.

See **Section 12.0 "Timer1/3/5 Module with Gate Control"** for more information on configuring the 16-bit TimerX resources.

Note:	Clocking TimerX from the system clock (FOSC) should not be used in Compare mode. In order for Compare mode to recognize the trigger event on the CCPx pin, TimerX must be clocked from the instruction clock (FOSC/4) or from an external clock source.

14.2.3 SOFTWARE INTERRUPT MODE

When Generate Software Interrupt mode is chosen (CCPxM<3:0> = 1010), the CCPx module does not assert control of the CCPx pin (see the CCPxCON register).

TABLE 14-10: REGISTERS ASSOCIATED WITH STANDARD PWM

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>		CCP2M<3:0>				198
CCP3CON	P3M<1:0>		DC3B<1:0>		CCP3M<3:0>				198
CCP4CON	—	—	DC4B<1:0>		CCP4M<3:0>				198
CCP5CON	—	—	DC5B<1:0>		CCP5M<3:0>				198
CCPTMRS0	C3TSEL<1:0>		—	C2TSEL<1:0>		—	C1TSEL<1:0>		201
CCPTMRS1	—	—	—	—	C5TSEL<1:0>		C4TSEL<1:0>		201
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	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
PIE2	OSCFIE	C1IE	C2IE	EEIE	BCL1IE	HLVDIE	TMR3IE	CCP2IE	118
PIE4	—	—	—	—	—	CCP5IE	CCP4IE	CCP3IE	120
PIR1	—	ADIF	RC1IF	TX1IF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	112
PIR2	OSCFIF	C1IF	C2IF	EEIF	BCL1IF	HLVDIF	TMR3IF	CCP2IF	113
PIR4	—	—	—	—	—	CCP5IF	CCP4IF	CCP3IF	115
PMD0	UART2MD	UART1MD	TMR6MD	TMR5MD	TMR4MD	TMR3MD	TMR2MD	TMR1MD	52
PMD1	MSSP2MD	MSSP1MD	—	CCP5MD	CCP4MD	CCP3MD	CCP2MD	CCP1MD	53
PR2	Timer2 Period Register								—
PR4	Timer4 Period Register								—
PR6	Timer6 Period Register								—
T2CON	—	T2OUTPS<3:0>				TMR2ON	T2CKPS<1:0>		166
T4CON	—	T4OUTPS<3:0>				TMR4ON	T4CKPS<1:0>		166
T6CON	—	T6OUTPS<3:0>				TMR6ON	T6CKPS<1:0>		166
TMR2	Timer2 Register								—
TMR4	Timer4 Register								—
TMR6	Timer6 Register								—
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	151
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	151
TRISD(1)	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	151
TRISE	WPUE3	—	—	—	—	TRISE2(1)	TRISE1(1)	TRISE0(1)	151

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

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

TABLE 14-11: CONFIGURATION REGISTERS ASSOCIATED WITH STANDARD PWM

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
CONFIG3H	MCLRE	—	P2BMX	T3CMX	HFOFST	CCP3MX	PBADEN	CCP2MX	348

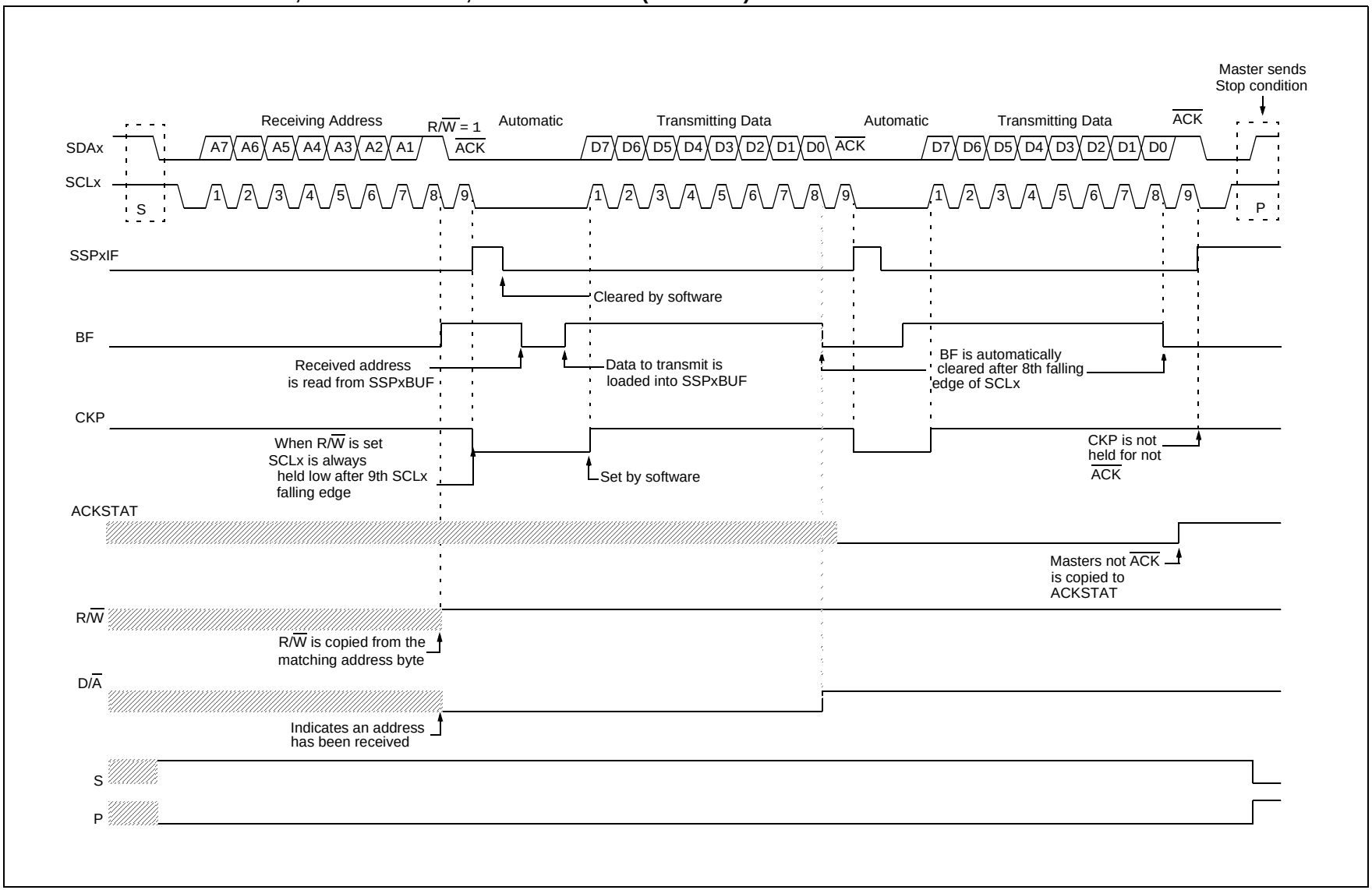
Legend: — = Unimplemented location, read as '0'. Shaded bits are not used by Standard PWM mode.

14.4.8 SETUP FOR ECCP PWM OPERATION USING ECCP1 AND TIMER2

The following steps should be taken when configuring the ECCP1 module for PWM operation using Timer2:

1. Configure the PWM pins to be used (P1A, P1B, P1C, and P1D):
   - Configure PWM outputs to be used as inputs by setting the corresponding TRIS bits. This prevents spurious outputs during setup.
   - Set the PSTR1CON bits for each PWM output to be used.
2. Select Timer2 as the period timer by configuring CCPTMR0 register bits C1TSEL<1:0> = '00'.
3. Set the PWM period by loading the PR2 register.
4. Configure auto-shutdown as OFF or select the source with the CCP1AS<2:0> bits of the ECCP1AS register.
5. Configure the auto-shutdown sources as needed:
   - Configure each comparator used.
   - Configure the comparator inputs as analog.
   - Configure the FLT0 input pin and clear ANSB0.
6. Force a shutdown condition (OFF included):
   - Configure safe starting output levels by setting the default shutdown drive states with the PSS1AC<1:0> and PSS1BD<1:0> bits of the ECCP1AS register.
   - Clear the P1RSEN bit of the PWM1CON register.
   - Set the CCP1AS bit of the ECCP1AS register.
7. Configure the ECCP1 module for the desired PWM mode and configuration by loading the CCP1CON register with the appropriate values:
   - Select one of the available output configurations and direction with the P1M<1:0> bits.
   - Select the polarities of the PWM output signals with the CCP1M<3:0> bits.
8. Set the 10-bit PWM duty cycle:
   - Load the eight MSbs into the CCPR1L register.
   - Load the two LSbs into the DC<1:0> bits of the CCP1CON register.
9. For Half-Bridge Output mode, set the dead-band delay by loading P1DC<6:0> bits of the PWM1CON register with the appropriate value.
10. Configure and start TMR2:
    - Set the TMR2 prescale value by loading the T2CKPS bits of the T2CON register.
    - Start Timer2 by setting the TMR2ON bit.
11. Enable the ECCP1/P1A, P1B, P1C and/or P1D pin outputs by clearing the respective TRIS bits.
12. Start the PWM:
    - If shutdown auto-restart is used, then set the P1RSEN bit of the PWM1CON register.
    - If shutdown auto-restart is not used, then clear the CCP1ASE bit of the ECCP1AS register.

FIGURE 15-18: I²C SLAVE, 7-BIT ADDRESS, TRANSMISSION (AHEN = 0)



19.3 Calibrating the CTMU Module

The CTMU requires calibration for precise measurements of capacitance and time, as well as for accurate time delay. If the application only requires measurement of a relative change in capacitance or time, calibration is usually not necessary. An example of this type of application would include a capacitive touch switch, in which the touch circuit has a baseline capacitance, and the added capacitance of the human body changes the overall capacitance of a circuit.

If actual capacitance or time measurement is required, two hardware calibrations must take place: the current source needs calibration to set it to a precise current, and the circuit being measured needs calibration to measure and/or nullify all other capacitance other than that to be measured.

19.3.1 CURRENT SOURCE CALIBRATION

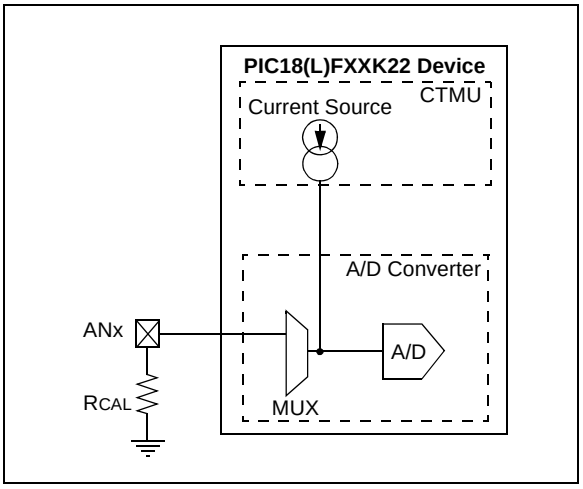
The current source on the CTMU module is trimable. Therefore, for precise measurements, it is possible to measure and adjust this current source by placing a high precision resistor, RCAL, onto an unused analog channel. An example circuit is shown in Figure 19-2. The current source measurement is performed using the following steps:

1. Initialize the A/D Converter.
2. Initialize the CTMU.
3. Enable the current source by setting EDG1STAT (CTMUCONL<0>).
4. Issue settling time delay.
5. Perform A/D conversion.
6. Calculate the current source current using $I = V/R_{CAL}$, where R_{CAL} is a high precision resistance and V is measured by performing an A/D conversion.

The CTMU current source may be trimmed with the trim bits in CTMUICON using an iterative process to get an exact desired current. Alternatively, the nominal value without adjustment may be used; it may be stored by the software for use in all subsequent capacitive or time measurements.

To calculate the value for R_{CAL}, the nominal current must be chosen, and then the resistance can be calculated. For example, if the A/D Converter reference voltage is 3.3V, use 70% of full scale, or 2.31V as the desired approximate voltage to be read by the A/D Converter. If the range of the CTMU current source is selected to be 0.55 μA, the resistor value needed is calculated as $R_{CAL} = 2.31V/0.55\ \mu A$, for a value of 4.2 MΩ. Similarly, if the current source is chosen to be 5.5 μA, R_{CAL} would be 420,000Ω, and 42,000Ω if the current source is set to 55 μA.

FIGURE 19-2: CTMU CURRENT SOURCE CALIBRATION CIRCUIT



A value of 70% of full-scale voltage is chosen to make sure that the A/D Converter was in a range that is well above the noise floor. Keep in mind that if an exact current is chosen, that is to incorporate the trimming bits from CTMUICON, the resistor value of RCAL may need to be adjusted accordingly. RCAL may also be adjusted to allow for available resistor values. RCAL should be of the highest precision available, keeping in mind the amount of precision needed for the circuit that the CTMU will be used to measure. A recommended minimum would be 0.1% tolerance.

The following examples show one typical method for performing a CTMU current calibration. Example 19-1 demonstrates how to initialize the A/D Converter and the CTMU; this routine is typical for applications using both modules. Example 19-2 demonstrates one method for the actual calibration routine.

REGISTER 24-9: CONFIG6H: CONFIGURATION REGISTER 6 HIGH

R/C-1	R/C-1	R-1	U-0	U-0	U-0	U-0	U-0
WRTD	WRTB	WRTC(1)	—	—	—	—	—
bit 7							bit 0

Legend:	
R = Readable bit	U = Unimplemented bit, read as '0'
-n = Value when device is unprogrammed	C = Clearable only bit

bit 7 **WRTD:** Data EEPROM Write Protection bit
1 = Data EEPROM not write-protected
0 = Data EEPROM write-protected

bit 6 **WRTB:** Boot Block Write Protection bit
1 = Boot Block not write-protected
0 = Boot Block write-protected

bit 5 **WRTC:** Configuration Register Write Protection bit(1)
1 = Configuration registers not write-protected
0 = Configuration registers write-protected

bit 4-0 **Unimplemented:** Read as '0'

Note 1: This bit is read-only in normal execution mode; it can be written only in Program mode.

REGISTER 24-10: CONFIG7L: CONFIGURATION REGISTER 7 LOW

U-0	U-0	U-0	U-0	R/C-1	R/C-1	R/C-1	R/C-1
—	—	—	—	EBTR3(1)	EBTR2(1)	EBTR1	EBTR0
bit 7							bit 0

Legend:	
R = Readable bit	U = Unimplemented bit, read as '0'
-n = Value when device is unprogrammed	C = Clearable only bit

bit 7-4 **Unimplemented:** Read as '0'

bit 3 **EBTR3:** Table Read Protection bit(1)
1 = Block 3 not protected from table reads executed in other blocks
0 = Block 3 protected from table reads executed in other blocks

bit 2 **EBTR2**: Table Read Protection bit(1)
1 = Block 2 not protected from table reads executed in other blocks
0 = Block 2 protected from table reads executed in other blocks

bit 1 **EBTR1:** Table Read Protection bit
1 = Block 1 not protected from table reads executed in other blocks
0 = Block 1 protected from table reads executed in other blocks

bit 0 **EBTR0:** Table Read Protection bit
1 = Block 0 not protected from table reads executed in other blocks
0 = Block 0 protected from table reads executed in other blocks

Note 1: Available on PIC18(L)FX5K22 and PIC18(L)FX6K22s devices.

24.5.1 PROGRAM MEMORY CODE PROTECTION

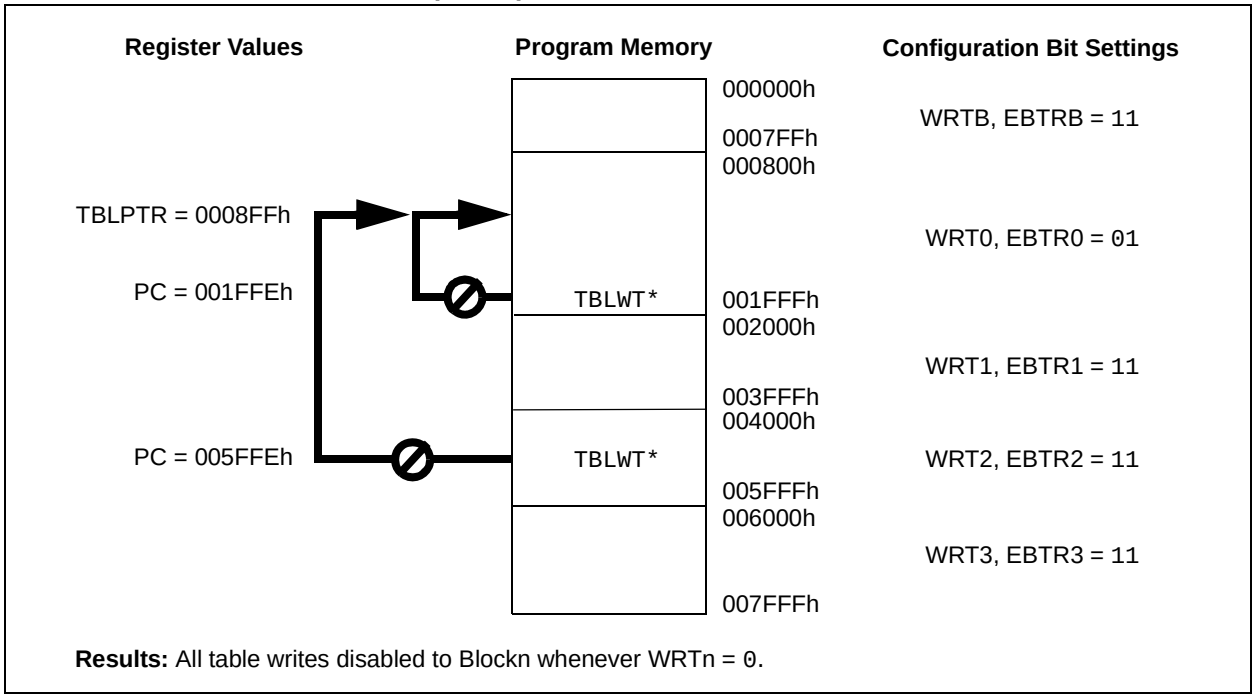
The program memory may be read to or written from any location using the table read and table write instructions. The device ID may be read with table reads. The Configuration registers may be read and written with the table read and table write instructions.

In Normal execution mode, the CPn bits have no direct effect. CPn bits inhibit external reads and writes. A block of user memory may be protected from table writes if the WRTn Configuration bit is '0'. The EBTRn bits control table reads. For a block of user memory with the EBTRn bit cleared to '0', a table READ instruction that executes from within that block is allowed to read.

A table read instruction that executes from a location outside of that block is not allowed to read and will result in reading '0's. Figures 24-3 through 24-5 illustrate table write and table read protection.

Note:	Code protection bits may only be written to a '0' from a '1' state. It is not possible to write a '1' to a bit in the '0' state. Code protection bits are only set to '1' by a full chip erase or block erase function. The full chip erase and block erase functions can only be initiated via ICSP™ or an external programmer.

FIGURE 24-3: TABLE WRITE (WRTn) DISALLOWED



Results: All table writes disabled to Blockn whenever WRTn = 0.

ADDWFC **ADD W and CARRY bit to f**

Syntax: ADDWFC f {,d {,a}}

Operands: $0 \le f \le 255$
$d \in [0,1]$
$a \in [0,1]$

Operation: (W) + (f) + (C) → dest

Status Affected: N,OV, C, DC, Z

Encoding:

0010	00da	ffff	ffff

Description: Add W, the CARRY flag and data memory location 'f'. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed in data memory location 'f'.
If 'a' is '0', the Access Bank is selected. If 'a' is '1', the BSR is used to select the GPR bank.
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 25.2.3 "Byte-Oriented and Bit-Oriented Instructions in Indexed Literal Offset Mode"** for details.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	Write to destination

Example: ADDWFC REG, 0, 1

Before Instruction
CARRY bit = 1
REG = 02h
W = 4Dh
After Instruction
CARRY bit = 0
REG = 02h
W = 50h

ANDLW **AND literal with W**

Syntax: ANDLW k

Operands: $0 \le k \le 255$

Operation: (W) .AND. k → W

Status Affected: N, Z

Encoding:

0000	1011	kkkk	kkkk

Description: The contents of W are AND'ed with the 8-bit literal 'k'. The result is placed in W.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process Data	Write to W

Example: ANDLW 05Fh

Before Instruction
W = A3h
After Instruction
W = 03h

25.2.2 EXTENDED INSTRUCTION SET

ADDFSR **Add Literal to FSR**

Syntax: ADDFSR f, k

Operands: $0 \le k \le 63$
$f \in [0, 1, 2]$

Operation: FSR(f) + k → FSR(f)

Status Affected: None

Encoding:

1110	1000	ffkk	kkkk

Description: The 6-bit literal 'k' is added to the contents of the FSR specified by 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process Data	Write to FSR

Example: `ADDFSR 2, 23h`

Before Instruction
FSR2 = 03FFh
After Instruction
FSR2 = 0422h

ADDULNK **Add Literal to FSR2 and Return**

Syntax: ADDULNK k

Operands: $0 \le k \le 63$

Operation: FSR2 + k → FSR2,
(TOS) → PC

Status Affected: None

Encoding:

1110	1000	11kk	kkkk

Description: The 6-bit literal 'k' is added to the contents of 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 ADDFSR instruction, where f = 3 (binary '11'); it operates only on FSR2.

Words: 1

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'	Process Data	Write to FSR
No Operation	No Operation	No Operation	No Operation

Example: `ADDULNK 23h`

Before Instruction
FSR2 = 03FFh
PC = 0100h
After Instruction
FSR2 = 0422h
PC = (TOS)

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 syntax then becomes: {label} instruction argument(s).

27.2 DC Characteristics: Power-Down Current, PIC18(L)F2X/4XK22 (Continued)

PIC18LF2X/4XK22		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C						
PIC18F2X/4XK22		**Standard Operating Conditions (unless otherwise stated)** Operating temperature -40°C ≤ TA ≤ +125°C						
Param No.	**Device Characteristics**	**Typ +25°C**	**Typ +60°C**	**Max +85°C**	**Max +125°C**	**Units**	**Conditions**	
							VDD	**Notes**
D015	Comparators	7	7	18	18	µA	1.8V	LP mode
		7	7	18	18	µA	3.0V	
		7	7	18	18	µA	2.3V	
		7	7	18	18	µA	3.0V	
		8	8	20	20	µA	5.0V	
D016	Comparators	38	38	95	95	µA	1.8V	HP mode
		40	40	105	105	µA	3.0V	
		39	39	95	95	µA	2.3V	
		40	40	105	105	µA	3.0V	
		40	40	105	105	µA	5.0V	
D017	DAC	14	14	25	25	µA	2.0V	
		20	20	35	35	µA	3.0V	
		15	15	30	30	µA	2.3V	
		20	20	35	35	µA	3.0V	
		32	32	60	60	µA	5.0V	
D018	FVR(2)	15	16	25	25	µA	1.8V	
		15	16	25	25	µA	3.0V	
		28	28	45	45	µA	2.3V	
		31	31	55	55	µA	3.0V	
		66	66	100	100	µA	5.0V	
D013	A/D Converter(3)	185	185	370	370	µA	1.8V	A/D on, not converting
		210	210	400	400	µA	3.0V	
		200	200	380	380	µA	2.3V	
		210	210	400	400	µA	3.0V	
		250	250	450	450	µA	5.0V	

Note 1: The power-down current in Sleep mode does not depend on the oscillator type. Power-down current is measured with the part in Sleep mode, with all I/O pins in high-impedance state and tied to VDD or VSS and all features that add delta current disabled (such as WDT, Timer1 Oscillator, BOR, etc.).

2: On PIC18LF2X/4XK22 the BOR, HLVD and FVR enable internal band gap reference. With more than one of these modules enabled, the current consumption will be less than the sum of the specifications. On PIC18F2X/4XK22, the internal band gap reference is always enabled and its current consumption is included in the Power-down Base Current (IPD).

3: A/D converter differential currents apply only in Run mode. In Sleep or Idle mode both the ADC and the FRC turn off as soon as conversion (if any) is complete.

TABLE 27-8: PLL CLOCK TIMING SPECIFICATIONS

Param. No.	Sym	Characteristic	Min	Max	Units	Conditions
F10	FOSC	Oscillator Frequency Range	4	5	MHz	VDD < 2.7V, -40°C to +85°C
			4	4	MHz	VDD < 2.7V, +85°C to +125°C
			4	16	MHz	2.7V ≤ VDD, -40°C to +85°C
			4	12	MHz	2.7V ≤ VDD, +85°C to +125°C
F11	FSYS	On-Chip VCO System Frequency	16	20	MHz	VDD < 2.7V, -40°C to +85°C
			16	16	MHz	VDD < 2.7V, +85°C to +125°C
			16	64	MHz	2.7V ≤ VDD, -40°C to +85°C
			16	48	MHz	2.7V ≤ VDD, +85°C to +125°C
F12	t_{rc}	PLL Start-up Time (Lock Time)	—	2	ms	

TABLE 27-9: AC CHARACTERISTICS:INTERNAL OSCILLATORS ACCURACY PIC18(L)F46K22

Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +125°C							
Param. No.	**Characteristics**	**Freq. Tolerance**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
OA1	Internal Calibrated HFINTOSC Frequency(1)	± 2%	—	16.0	—	MHz	0°C ≤ TA ≤ +60°C, VDD ≥ 2.5V
		± 3%	—	16.0	—	MHz	+60°C ≤ TA ≤ +85°C, VDD ≥ 2.5V
		± 5%	—	16.0	—	MHz	-40°C ≤ TA ≤ +125°C
OA2	Internal Calibrated MFINTOSC Frequency(1)	± 2%	—	500	—	kHz	0°C ≤ TA ≤ +60°C, VDD ≥ 2.5V
		± 3%	—	500	—	kHz	+60°C ≤ TA ≤ +85°C, VDD ≥ 2.5V
		± 5%	—	500	—	kHz	-40°C ≤ TA ≤ +125°C
OA3	Internal Calibrated LFINTOSC Frequency(1)	± 20%	—	31	—	kHz	-40°C ≤ TA ≤ +125°C

† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: To ensure these oscillator frequency tolerances, VDD and VSS must be capacitively decoupled as close to the device as possible. 0.1 µF and 0.01 µF values in parallel are recommended.

TABLE 27-22: A/D CONVERSION REQUIREMENTS PIC18(L)F2X/4XK22

Standard Operating Conditions (unless otherwise stated) Operating temperature Tested at +25°C							
Param. No.	**Symbol**	**Characteristic**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
130	TAD	A/D Clock Period	1	—	25	µs	-40°C to +85°C
			1	—	4	µs	+85°C to +125°C
131	TCNV	Conversion Time (not including acquisition time) **(Note 1)**	11	—	11	TAD	
132	TACQ	Acquisition Time **(Note 2)**	1.4	—	—	µs	VDD = 3V, Rs = 50Ω
135	TSWC	Switching Time from Convert → Sample	—	—	**(Note 3)**		
136	TDIS	Discharge Time	1	—	1	TCY	

Note 1: ADRES register may be read on the following TCY cycle.

2: The time for the holding capacitor to acquire the "New" input voltage when the voltage changes full scale after the conversion (VDD to VSS or VSS to VDD). The source impedance (*RS*) on the input channels is 50 Ω.

3: On the following cycle of the device clock.

FIGURE 28-40: PIC18LF2X/4XK22 TYPICAL IDD: RC_IDLE HF-INTOSC

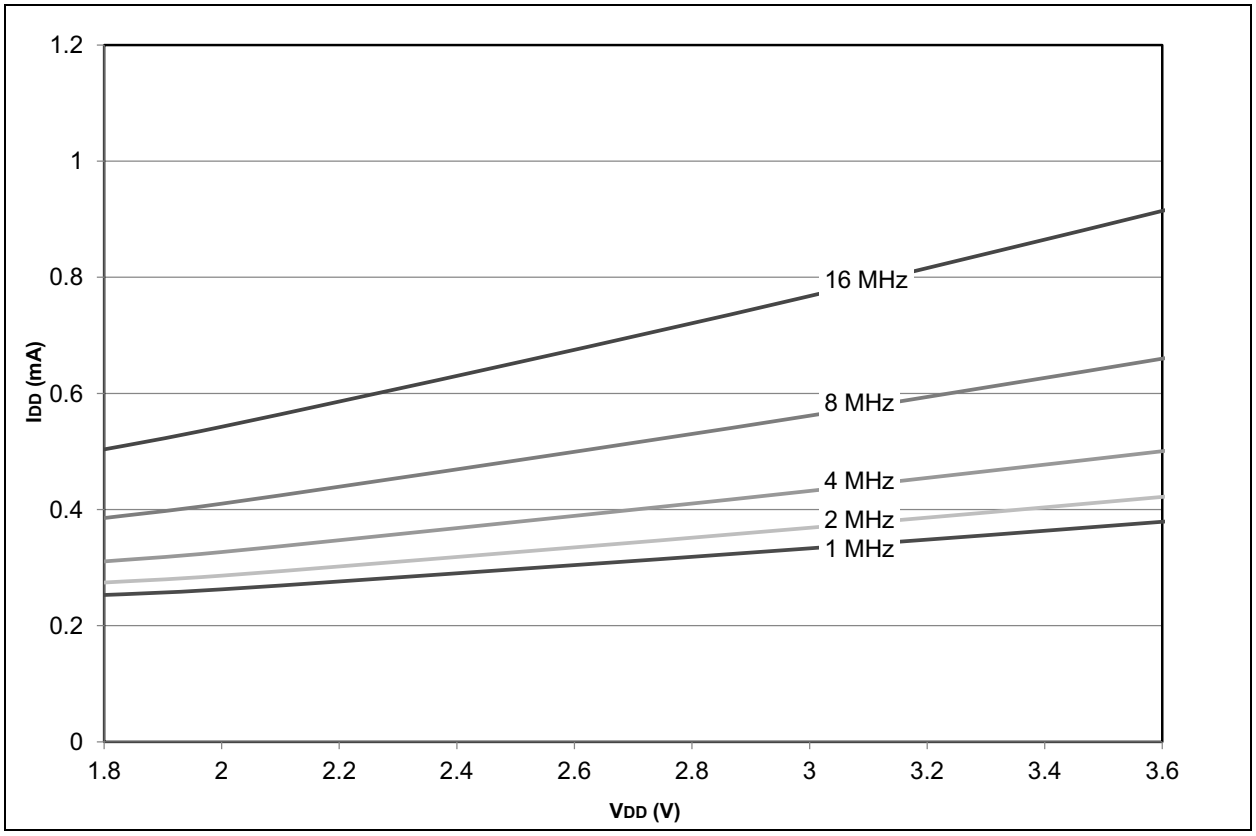
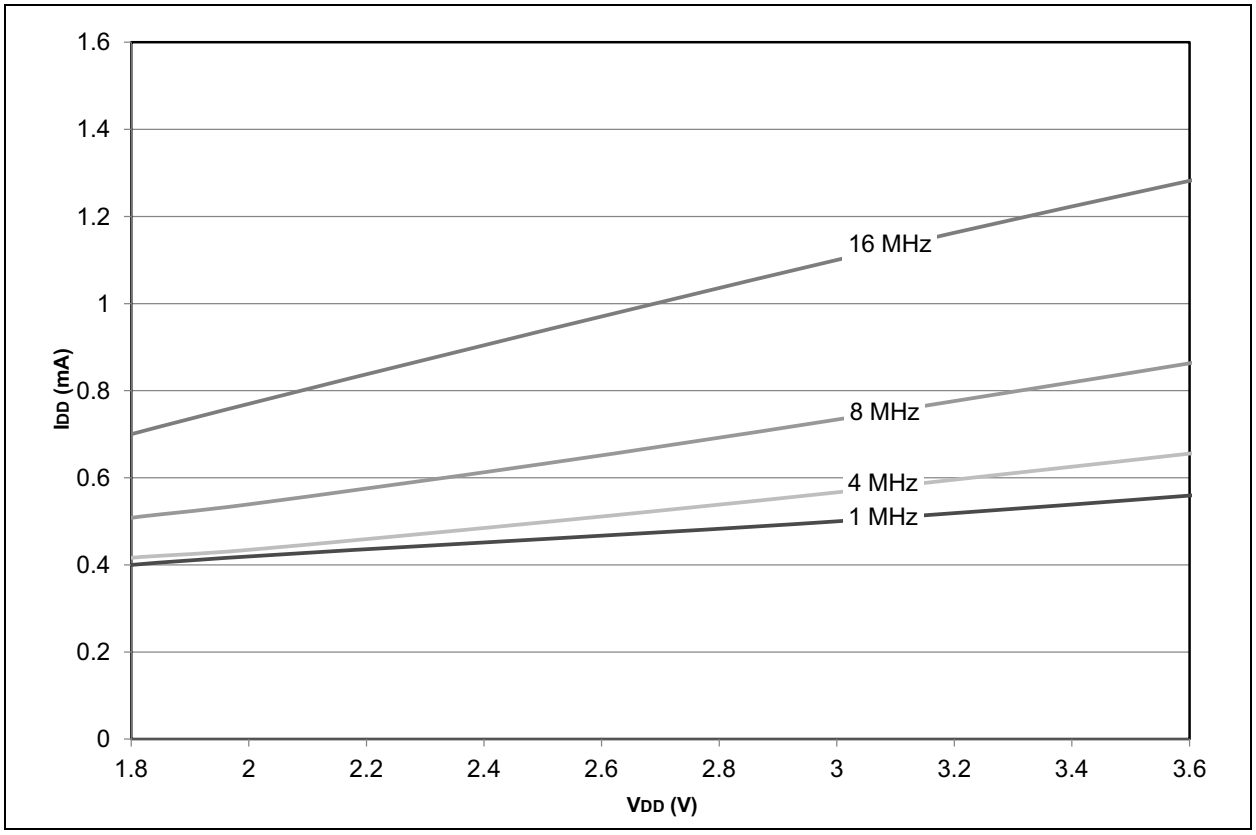


FIGURE 28-41: PIC18LF2X/4XK22 MAXIMUM IDD: RC_IDLE HF-INTOSC



PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery, refer to the factory or the listed sales office.

PART NO.	[X](2)	-	X	/XX	XXX
Device	Tape and Reel Option		Temperature Range	Package	Pattern

Device:	PIC18F23K22, PIC18LF23K22 PIC18F24K22, PIC18LF24K22 PIC18F25K22, PIC18LF25K22 PIC18F26K22, PIC18LF26K22 PIC18F43K22, PIC18LF43K22 PIC18F44K22, PIC18LF44K22 PIC18F45K22, PIC18LF45K22 PIC18F46K22, PIC18LF46K22
Tape and Reel Option:	Blank = standard packaging (tube or tray) T = Tape and Reel(1), (2)
Temperature Range:	E = -40°C to +125°C (Extended) I = -40°C to +85°C (Industrial)
Package:	ML = QFN MV = UQFN P = PDIP PT = TQFP (Thin Quad Flatpack) SO = SOIC SP = Skinny Plastic DIP SS = SSOP
Pattern:	QTP, SQTP, Code or Special Requirements (blank otherwise)

Examples:

a) PIC18(L)F45K22-E/P 301 = Extended temp., PDIP package, QTP pattern #301.

b) PIC18F46K22-I/SO = Industrial temp., SOIC package.

c) PIC18F46K22-E/P = Extended temp., PDIP package.

d) PIC18F46K22T-I/ML = Tape and reel, Industrial temp., QFN package.

Note 1: Tape and Reel option is available for ML, MV, PT, SO and SS packages with industrial Temperature Range only.

2: Tape and Reel identifier only appears in catalog part number description. This identifier is used for ordering purposes and is not printed on the device package.