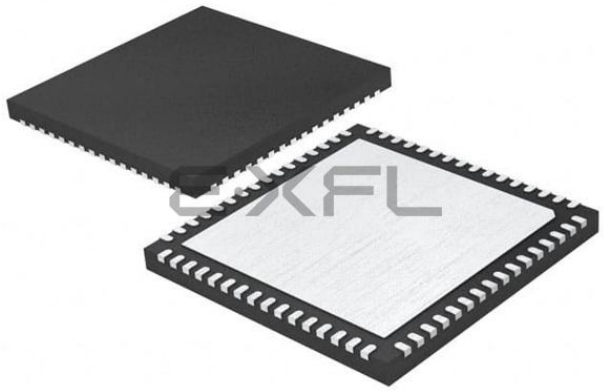


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

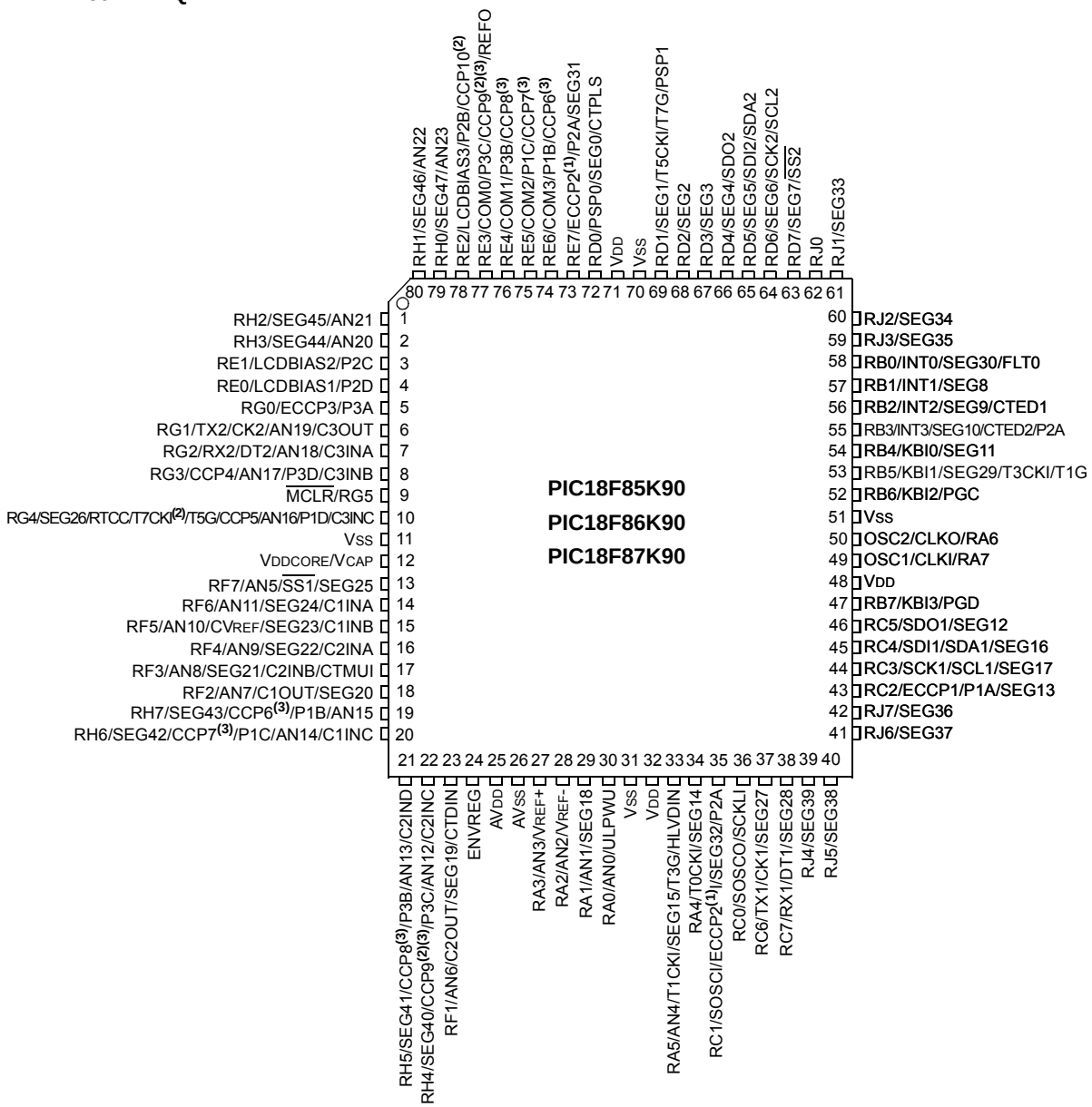
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LCD, POR, PWM, WDT
Number of I/O	53
Program Memory Size	32KB (16K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 16x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 125°C (TA)
Mounting Type	Surface Mount
Package / Case	64-VFQFN Exposed Pad
Supplier Device Package	64-VQFN (9x9)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f65k90-e-mr

Pin Diagrams – PIC18F8XK90

80-Pin TQFP



Note 1: The ECCP2 pin placement depends on the CCP2MX Configuration bit setting.

2: Not available on the PIC18F65K90 and PIC18F85K90.

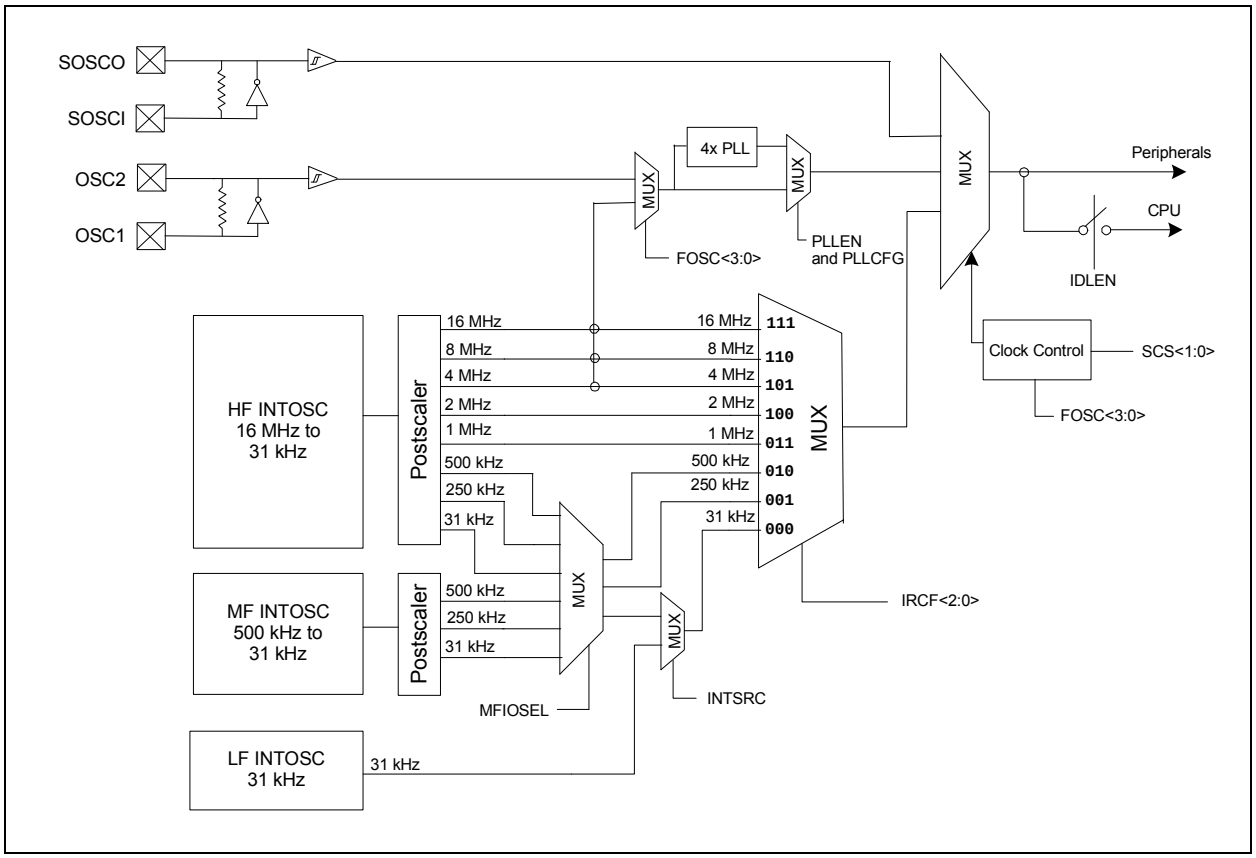
3: The CCP6, CCP7, CCP8 and CCP9 pin placement depends on the ECCPMX Configuration bit setting

© 2009-2011 Microchip Technology Inc.

TABLE 3-1: HS, EC, XT, LP AND RC MODES: RANGES AND SETTINGS

Mode	Frequency Range	OSC<3:0> Setting
EC1 (low power)	DC-160 kHz	1101
(EC1 & EC1IO)		1100
EC2 (medium power)	160 kHz-16 MHz	1011
(EC2 & EC2IO)		1010
EC3 (high power)	16 MHz-64 MHz	0101
(EC3 & EC3IO)		0100
HS1 (medium power)	4 MHz-16 MHz	0011
HS2 (high power)	16 MHz-25 MHz	0010
XT	100 kHz-4 MHz	0001
LP	31.25 kHz	0000
RC (External)	0-4 MHz	011x
INTIO	32 kHz-16 MHz	100x (and OSCCON, OSCCON2)

FIGURE 3-1: PIC18F87K90 FAMILY CLOCK DIAGRAM



© 2009-2011 Microchip Technology Inc.

4.1.3 CLOCK TRANSITIONS AND STATUS INDICATORS

The length of the transition between clock sources 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. The HF-INTOSC and MF-INTOSC are termed as INTOSC in this chapter.

Three bits indicate the current clock source and its status, as shown in Table 4-2. The three bits are:

- OSTS (OSCCON<3>)
- HFIOFS (OSCCON<2>)
- SOSCRUN (OSCCON2<6>)

TABLE 4-2: SYSTEM CLOCK INDICATOR

Main Clock Source	OSTS	HFIOFS or MFIOFS	SOSCRUN
Primary Oscillator	1	0	0
INTOSC (HF-INTOSC or MF-INTOSC)	0	1	0
Secondary Oscillator	0	0	1
MF-INTOSC or HF-INTOSC as Primary Clock Source	1	1	0
LF-INTOSC is Running or INTOSC is Not Yet Stable	0	0	0

When the OSTS bit is set, the primary clock is providing the device clock. When the HFIOFS or MFIOFS bit is set, the INTOSC output is providing a stable clock source to a divider that actually drives the device clock. When the SOSCRUN bit is set, the SOSC oscillator is providing the clock. If none of these bits are set, either the LF-INTOSC clock source is clocking the device or the INTOSC source is not yet stable.

If the internal oscillator block is configured as the primary clock source by the OSC<3:0> Configuration bits (CONFIG1H<3:0>), then the OSTS and HFIOFS or MFIOFS bits can be set when in PRI_RUN or PRI_IDLE modes. This indicates that the primary clock (INTOSC output) is generating a stable output. Entering another INTOSC power-managed mode at the same frequency would clear the OSTS bit.

Note 1: Caution should be used when modifying a single IRCF bit. At a lower VDD, it is possible to select a higher clock speed than is supportable by that VDD. Improper device operation may result if the VDD/FOSC specifications are violated.

2: Executing a SLEEP instruction does not necessarily place the device into Sleep mode. It acts as the trigger to place the controller into either the Sleep mode or one of the Idle modes, depending on the setting of the IDLEN bit.

4.1.4 MULTIPLE SLEEP COMMANDS

The power-managed mode that is invoked with the SLEEP instruction is determined by the setting of the IDLEN bit at the time the instruction is executed. If another SLEEP instruction is executed, the device will enter the power-managed mode specified by IDLEN at that time. If IDLEN has changed, the device will enter the new power-managed mode specified by the new setting.

4.2 Run Modes

In the Run modes, clocks to both the core and peripherals are active. The difference between these modes is the clock source.

4.2.1 PRI_RUN MODE

The PRI_RUN mode is the normal, Full-Power Execution mode of the microcontroller. This is also the default mode upon a device Reset, unless Two-Speed Start-up is enabled. (For details, see **Section 28.4 “Two-Speed Start-up”**.) In this mode, the OSTS bit is set. The HFIOFS or MFIOFS bit may be set if the internal oscillator block is the primary clock source. (See **Section 3.2 “Control Registers”**.)

4.2.2 SEC_RUN MODE

The SEC_RUN mode is the compatible mode to the “clock-switching” feature offered in other PIC18 devices. In this mode, the CPU and peripherals are clocked from the SOSC oscillator. This enables lower power consumption while retaining a high-accuracy clock source.

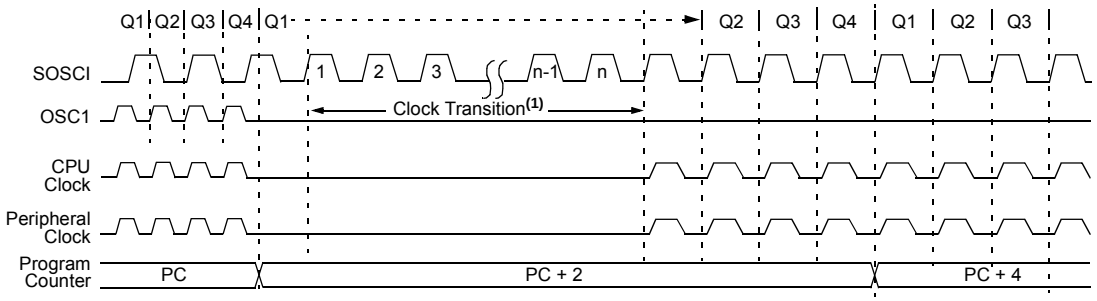
SEC_RUN mode is entered by setting the SCS<1:0> bits to ‘01’. The device clock source is switched to the SOSC oscillator (see Figure 4-1), the primary oscillator is shut down, the SOSCRUN bit (OSCCON2<6>) is set and the OSTS bit is cleared.

Note: The SOSC oscillator can be enabled by setting the SOSCGO bit (OSCCON2<3>). If this bit is set, the clock switch to the SEC_RUN mode can switch immediately once SCS<1:0> are set to ‘01’.

On transitions from SEC_RUN mode to PRI_RUN mode, the peripherals and CPU continue to be clocked from the SOSC oscillator while the primary clock is started. When the primary clock becomes ready, a clock switch back to the primary clock occurs (see Figure 4-2). When the clock switch is complete, the SOSCRUN bit is cleared, the OSTS bit is set and the primary clock is providing the clock. The IDLEN and SCS bits are not affected by the wake-up and the SOSC oscillator continues to run.

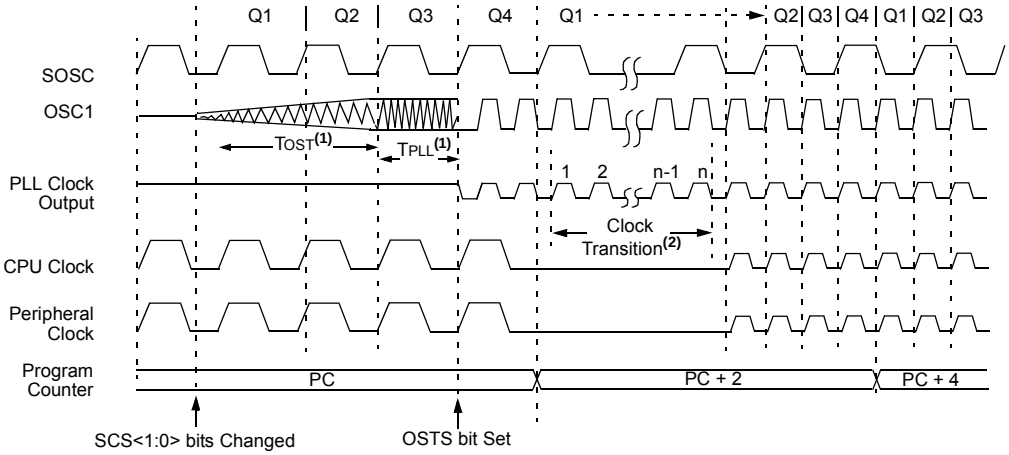
© 2009-2011 Microchip Technology Inc.

FIGURE 4-1: TRANSITION TIMING FOR ENTRY TO SEC_RUN MODE



Note 1: Clock transition typically occurs within 2-4 TOSC.

FIGURE 4-2: TRANSITION TIMING FROM SEC_RUN MODE TO PRI_RUN MODE (HSPLL)



Note 1: TOST = 1024 TOSC; TPLL = 2 ms (approx). These intervals are not shown to scale.
2: Clock transition typically occurs within 2-4 TOSC.

4.2.3 RC_RUN MODE

In RC_RUN mode, the CPU and peripherals are clocked from the internal oscillator block using the INTOSC multiplexer. In this mode, the primary clock is shut down. When using the LF-INTOSC source, this mode provides the best power conservation of all the Run modes, while still executing code. It works well for user applications which are not highly timing-sensitive or do not require high-speed clocks at all times.

If the primary clock source is the internal oscillator block – either LF-INTOSC or INTOSC (MF-INTOSC or HF-INTOSC) – there are no distinguishable differences between the PRI_RUN and RC_RUN modes during execution. Entering or exiting RC_RUN mode, however, causes a clock switch delay. Therefore, if the primary clock source is the internal oscillator block, using RC_RUN mode is not recommended.

This mode is entered by setting the SCS1 bit to '1'. To maintain software compatibility with future devices, it is recommended that the SCS0 bit also be cleared, even though the bit is ignored. When the clock source is switched to the INTOSC multiplexer (see Figure 4-3), the primary oscillator is shut down and the OSTS bit is cleared. The IRCF bits may be modified at any time to immediately change the clock speed.

Note:	Caution should be used when modifying a single IRCF bit. At a lower VDD, it is possible to select a higher clock speed than is supportable by that VDD. Improper device operation may result if the VDD/FOSC specifications are violated.

6.2.3 INSTRUCTIONS IN PROGRAM MEMORY

The program memory is addressed in bytes. Instructions are stored as two or four bytes in program memory. The Least Significant Byte (LSB) 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 6.1.2 "Program Counter"**).

Figure 6-5 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 6-5 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. For more details on the instruction set, see **Section 29.0 "Instruction Set Summary"**.

FIGURE 6-5: INSTRUCTIONS IN PROGRAM MEMORY

			LSB = 1	LSB = 0	Word Address ↓
Program Memory Byte Locations →					000000h
					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

6.2.4 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 instructions 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 6-4 shows how this works.

Note:	For information on two-word instructions in the extended instruction set, see **Section 6.5 "Program Memory and the Extended Instruction Set"**.

EXAMPLE 6-4: TWO-WORD INSTRUCTIONS

CASE 1:			
Object Code	**Source Code**		
0110 0110 0000 0000	TSTFSZ	REG1	; is RAM location 0?
1100 0001 0010 0011	MOVFF	REG1, REG2	; 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, REG2	; Yes, execute this word
1111 0100 0101 0110			; 2nd word of instruction
0010 0100 0000 0000	ADDWF	REG3	; continue code

13.8.2 TIMER1 GATE SOURCE SELECTION

The Timer1 gate source can be selected from one of four sources. Source selection is controlled by the T1GSSx bits, T1GCON<1:0> (see Table 13-4).

TABLE 13-4: TIMER1 GATE SOURCES

T1GSS<1:0>	Timer1 Gate Source
00	Timer1 Gate Pin
01	TMR2 to Match PR2 (TMR2 increments to match PR2)
10	Comparator 1 Output (Comparator logic high output)
11	Comparator 2 Output (Comparator logic high output)

The polarity for each available source is also selectable, controlled by the T1GPOL bit (T1GCON<6>).

13.8.2.1 T1G Pin Gate Operation

The T1G pin is one source for Timer1 gate control. It can be used to supply an external source to the Timer1 gate circuitry.

13.8.2.2 Timer2 Match Gate Operation

The TMR2 register will increment until it matches the value in the PR2 register. On the very next increment cycle, TMR2 will be reset to 00h. When this Reset occurs, a low-to-high pulse will automatically be generated and internally supplied to the Timer1 gate circuitry. The pulse will remain high for one instruction cycle and will return back to a low state until the next match.

Depending on T1GPOL, Timer1 increments differently when TMR2 matches PR2. When T1GPOL = 1, Timer1 increments for a single instruction cycle following a TMR2 match with PR2. When T1GPOL = 0, Timer1 increments continuously, except for the cycle following the match, when the gate signal goes from low-to-high.

13.8.2.3 Comparator 1 Output Gate Operation

The output of Comparator 1 can be internally supplied to the Timer1 gate circuitry. After setting up Comparator 1 with the CM1CON register, Timer1 will increment depending on the transition of the CMP1OUT (CMSTAT<5>) bit.

13.8.2.4 Comparator 2 Output Gate Operation

The output of Comparator 2 can be internally supplied to the Timer1 gate circuitry. After setting up Comparator 2 with the CM2CON register, Timer1 will increment depending on the transition of the CMP2OUT (CMSTAT<6>) bit.

© 2009-2011 Microchip Technology Inc.

TABLE 17-4: ALRMVAL REGISTER MAPPING

ALRMPTR<1:0>	Alarm Value Register Window	
	ALRMVALH	ALRMVALL
00	ALRMMIN	ALRMSEC
01	ALRMWD	ALRMHR
10	ALRMMNTH	ALRMDAY
11	—	—

17.2.9 CALIBRATION

The real-time crystal input can be calibrated using the periodic auto-adjust feature. When properly calibrated, the RTCC can provide an error of less than three seconds per month.

To perform this calibration, find the number of error clock pulses and store the value into the lower half of the RTCCAL register. The 8-bit, signed value, loaded into RTCCAL, is multiplied by four and will be either added or subtracted from the RTCC timer, once every minute.

To calibrate the RTCC module:

1. Use another timer resource on the device to find the error of the 32.768 kHz crystal.
2. Convert the number of error clock pulses per minute (see Equation 17-1).

EQUATION 17-1: CONVERTING ERROR CLOCK PULSES

(Ideal Frequency (32,758) – Measured Frequency) * 60 = Error Clocks per Minute

   - If the oscillator is *faster* than ideal (negative result from Step 2), the RCFGCALL register value needs to be negative. This causes the specified number of clock pulses to be subtracted from the timer counter, once every minute.
   - If the oscillator is *slower* than ideal (positive result from Step 2), the RCFGCALL register value needs to be positive. This causes the specified number of clock pulses to be added to the timer counter, once every minute.
3. Load the RTCCAL register with the correct value.

Writes to the RTCCAL register should occur only when the timer is turned off or immediately after the rising edge of the seconds pulse.

Note: In determining the crystal's error value, it is the user's responsibility to include the crystal's initial error from drift due to temperature or crystal aging.

17.3 Alarm

The Alarm features and characteristics are:

- Configurable from half a second to one year
- Enabled using the ALRMEN bit (ALRMCFG<7>, Register 17-4)
- Offers one-time and repeat alarm options

17.3.1 CONFIGURING THE ALARM

The alarm feature is enabled using the ALRMEN bit. This bit is cleared when an alarm is issued. The bit will not be cleared if the CHIME bit = 1 or if ALRMRPT ≠ 0.

The interval selection of the alarm is configured through the ALRMCFG bits (AMASK<3:0>) (see Figure 17-5). These bits determine which, and how many, digits of the alarm must match the clock value for the alarm to occur.

The alarm can also be configured to repeat based on a preconfigured interval. The number of times this occurs, after the alarm is enabled, is stored in the ALRMRPT register.

Note: While the alarm is enabled (ALRMEN = 1), changing any of the registers, other than the RTCCAL, ALRMCFG and ALRMRPT registers and the CHIME bit, can result in a false alarm event leading to a false alarm interrupt. To avoid this, only change the timer and alarm values while the alarm is disabled (ALRMEN = 0). It is recommended that the ALRMCFG and ALRMRPT registers and CHIME bit be changed when RTCSYNC = 0.

© 2009-2011 Microchip Technology Inc.

TABLE 18-5: REGISTERS ASSOCIATED WITH CAPTURE, COMPARE, TIMER1/3/5/7

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	75
RCON	IPEN	SBOREN	$\overline{CM}$	$\overline{RI}$	$\overline{TO}$	$\overline{PD}$	$\overline{POR}$	$\overline{BOR}$	76
PIR4	CCP10IF(1)	CCP9IF(1)	CCP8IF	CCP7IF	CCP6IF	CCP5IF	CCP4IF	CCP3IF	77
PIE4	CCP10IE(1)	CCP9IE(1)	CCP8IE	CCP7IE	CCP6IE	CCP5IE	CCP4IE	CCP3IE	77
IPR4	CCP10IP(1)	CCP9IP(1)	CCP8IP	CCP7IP	CCP6IP	CCP5IP	CCP4IP	CCP3IP	77
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	78
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	78
TRISE	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2	TRISE1	TRISE0	78
TRISH(2)	TRISH7	TRISH6	TRISH5	TRISH4	TRISH3	TRISH2	TRISH1	TRISH0	78
TMR1L	Timer1 Register Low Byte								76
TMR1H	Timer1 Register High Byte								76
TMR3L	Timer3 Register Low Byte								77
TMR3H	Timer3 Register High Byte								77
TMR5L	Timer5 Register Low Byte								82
TMR5H	Timer5 Register High Byte								82
TMR7L(1)	Timer7 Register Low Byte								81
TMR7H(1)	Timer7 Register High Byte								81
T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	SOSCEN	$\overline{T1SYNC}$	RD16	TMR1ON	76
T3CON	TMR3CS1	TMR3CS0	T3CKPS1	T3CKPS0	SOSCEN	$\overline{T3SYNC}$	RD16	TMR3ON	77
T5CON	TMR5CS1	TMR5CS0	T5CKPS1	T5CKPS0	SOSCEN	$\overline{T5SYNC}$	RD16	TMR5ON	82
T7CON(1)	TMR7CS1	TMR7CS0	T7CKPS1	T7CKPS0	SOSCEN	$\overline{T7SYNC}$	RD16	TMR7ON	81
CCPR4L	Capture/Compare/PWM Register 4 Low Byte								82
CCPR4H	Capture/Compare/PWM Register 4 High Byte								82
CCPR5L	Capture/Compare/PWM Register 5 Low Byte								82
CCPR5H	Capture/Compare/PWM Register 5 High Byte								82
CCPR6L	Capture/Compare/PWM Register 6 Low Byte								82
CCPR6H	Capture/Compare/PWM Register 6 High Byte								82
CCPR7L	Capture/Compare/PWM Register 7 Low Byte								82
CCPR7H	Capture/Compare/PWM Register 7 High Byte								82
CCPR8L	Capture/Compare/PWM Register 8 Low Byte								80
CCPR8H	Capture/Compare/PWM Register 8 High Byte								80
CCPR9L(1)	Capture/Compare/PWM Register 9 Low Byte								80
CCPR9H(1)	Capture/Compare/PWM Register 9 High Byte								80
CCPR10L(1)	Capture/Compare/PWM Register 10 Low Byte								81
CCPR10H(1)	Capture/Compare/PWM Register 10 High Byte								80
CCP4CON	—	—	DC4B1	DC4B0	CCP4M3	CCP4M2	CCP4M1	CCP4M0	82
CCP5CON	—	—	DC5B1	DC5B0	CCP5M3	CCP5M2	CCP5M1	CCP5M0	82

Legend: — = unimplemented, read as '0'. Shaded cells are not used by Capture/Compare or Timer1/3/5/7.

Note 1: Unimplemented in devices with a program memory of 32 Kbytes (PIC18FX5K90).

2: Unimplemented in 64-pin devices.

© 2009-2011 Microchip Technology Inc.

19.2 Capture Mode

In Capture mode, the CCPRxH:CCPRxL register pair captures the 16-bit value of the TMR1 or TMR3 registers when an event occurs on the corresponding ECCPx pin. An event is defined as one of the following:

- Every falling edge
- Every rising edge
- Every fourth rising edge
- Every 16th rising edge

The event is selected by the mode select bits, CCPxM<3:0> (CCPxCON register<3:0>). When a capture is made, the interrupt request flag bit, CCPxIF, is set (see Table 19-2). The flag must be cleared by software. If another capture occurs before the value in the CCPRxH/L register is read, the old captured value is overwritten by the new captured value.

TABLE 19-2: ECCP1/2/3 INTERRUPT FLAG BITS

ECCP Module	Flag Bit
1	PIR3<1>
2	PIR3<2>
3	PIR4<0>

19.2.1 ECCP PIN CONFIGURATION

In Capture mode, the appropriate ECCPx pin should be configured as an input by setting the corresponding TRIS direction bit.

Note:	If the ECCPx pin is configured as an output, a write to the PORT can cause a capture condition.

19.2.2 TIMER1/TIMER3 MODE SELECTION

The timers that are to be used with the capture feature (Timer1 and/or Timer3) must be running in Timer mode or Synchronized Counter mode. In Asynchronous Counter mode, the capture operation may not work. The timer to be used with each ECCP module is selected in the CCPTMRS0 register (Register 19-2).

19.2.3 SOFTWARE INTERRUPT

When the Capture mode is changed, a false capture interrupt may be generated. The user should keep the CCPxIE interrupt enable bit clear to avoid false interrupts. The interrupt flag bit, CCPxIF, should also be cleared following any such change in operating mode.

19.2.4 ECCP PRESCALER

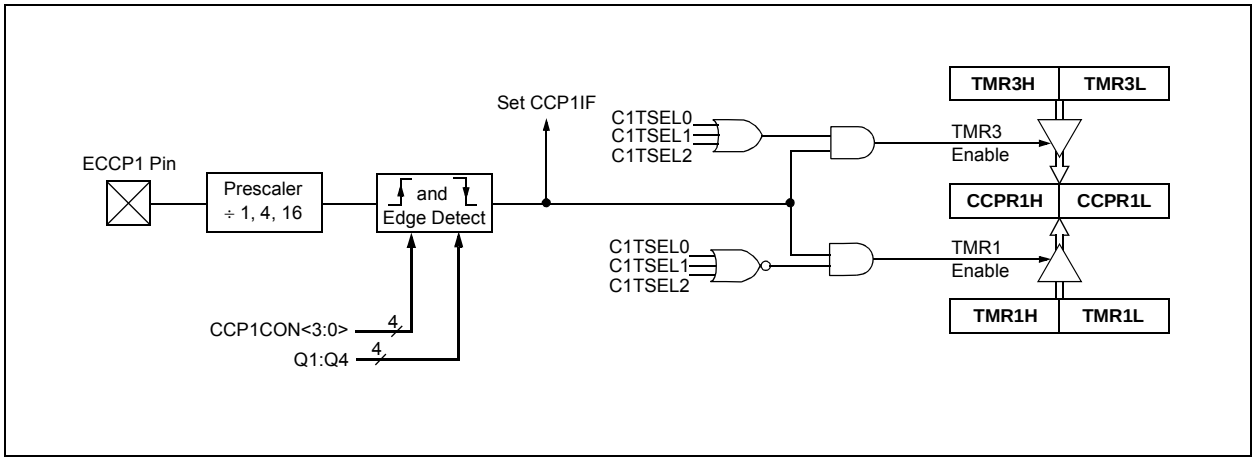
There are four prescaler settings in Capture mode; they are specified as part of the operating mode selected by the mode select bits (CCPxM<3:0>). Whenever the ECCP module is turned off, or Capture mode is disabled, the prescaler counter is cleared. This means that any Reset will clear the prescaler counter.

Switching from one capture prescaler to another may generate an interrupt. Also, the prescaler counter will not be cleared; therefore, the first capture may be from a non-zero prescaler. Example 19-1 provides the recommended method for switching between capture prescalers. This example also clears the prescaler counter and will not generate the “false” interrupt.

EXAMPLE 19-1: CHANGING BETWEEN CAPTURE PRESCALERS

```
CLRF   CCP1CON     ; Turn ECCP module off
MOVLW  NEW_CAPT_PS ; Load WREG with the
                   ; new prescaler mode
                   ; value and ECCP ON
MOVWF  CCP1CON     ; Load CCP1CON with
                   ; this value
```

FIGURE 19-1: CAPTURE MODE OPERATION BLOCK DIAGRAM



22.0 ENHANCED UNIVERSAL SYNCHRONOUS ASYNCHRONOUS RECEIVER TRANSMITTER (EUSART)

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

The Enhanced USART module implements additional features, including automatic baud rate detection and calibration, automatic wake-up on Sync Break reception and 12-bit Break character transmit. These make it ideally suited for use in Local Interconnect Network bus (LIN/J2602 bus) systems.

All members of the PIC18F87K90 family are equipped with two independent EUSART modules, referred to as EUSART1 and EUSART2. They can be configured in the following modes:

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

The pins of EUSART1 and EUSART2 are multiplexed with the functions of PORTC (RC6/TX1/CK1/SEG27 and RC7/RX1/DT1/SEG28) and PORTG (RG1/TX2/CK2/AN19/C3OUT and RG2/RX2/DT2/AN18/C3INA), respectively. In order to configure these pins as an EUSART:

- For EUSART1:
  - SPEN (RCSTA1<7>) bit must be set (= 1)
  - TRISC<7> bit must be set (= 1)
  - TRISC<6> bit must be cleared (= 0) for Asynchronous and Synchronous Master modes
  - TRISC<6> bit must be set (= 1) for Synchronous Slave mode
- For EUSART2:
  - SPEN (RCSTA2<7>) bit must be set (= 1)
  - TRISG<2> bit must be set (= 1)
  - TRISG<1> bit must be cleared (= 0) for Asynchronous and Synchronous Master modes
  - TRISC<6> bit must be set (= 1) for Synchronous Slave mode

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

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

- Transmit Status and Control (TXSTAx)
- Receive Status and Control (RCSTAx)
- Baud Rate Control (BAUDCONx)

These are detailed in Register 22-1, Register 22-2 and Register 22-3, respectively, on the following pages.

Note:	Throughout this section, references to register and bit names that may be associated with a specific EUSART module are referred to generically by the use of ‘x’ in place of the specific module number. Thus, “RCSTAx” might refer to the Receive Status register for either EUSART1 or EUSART2.

REGISTER 23-6: ADRESH: A/D RESULT HIGH BYTE REGISTER, RIGHT JUSTIFIED (ADFM = 1)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
ADSGN	ADSGN	ADSGN	ADSGN	ADRES11	ADRES10	ADRES9	ADRES8
bit 7							bit 0

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

bit 7-4 **ADSGN:** A/D Result Sign bits
1 = A/D result is negative
0 = A/D result is positive

bit 3-0 **ADRES<11:8>:** A/D Result High Byte bits

REGISTER 23-7: ADRESL: A/D RESULT LOW BYTE REGISTER, RIGHT JUSTIFIED (ADFM = 1)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2	ADRES1	ADRES0
bit 7							bit 0

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

bit 7-0 **ADRES<7:0>:** A/D Result Low Byte bits

23.8 Use of the Special Event Triggers

A/D conversion can be started by the Special Event Trigger of any of these modules:

- ECCP2 – Requires CCP2M<3:0> bits (CCP2CON<3:0>) set at '1011'
- CTMU – Requires the setting of the CTTRIG bit (CTMUCONH<0>)
- Timer1
- RTCC

To start an A/D conversion:

- The A/D module must be enabled (ADON = 1)
- The appropriate analog input channel is selected
- The minimum acquisition period is set in one of these ways:
  - Timing provided by the user
  - Selection made of an appropriate TACQ time

With these conditions met, the trigger sets the GO/$\overline{\text{DONE}}$ bit and the A/D acquisition starts.

If the A/D module is not enabled (ADON = 0), the module ignores the Special Event Trigger.

Note:	With an ECCP2 trigger, Timer1 or Timer3 is cleared. The timers reset to automatically repeat the A/D acquisition period with minimal software overhead (moving ADRESH:ADRESL to the desired location). If the A/D module is not enabled, the Special Event Trigger is ignored by the module, but the timer's counter resets.

23.9 Operation in Power-Managed Modes

The selection of the automatic acquisition time and A/D conversion clock is determined, in part, by the clock source and frequency while in a power-managed mode.

If the A/D is expected to operate while the device is in a power-managed mode, the ACQT<2:0> and ADCS<2:0> bits in ADCON2 should be updated in accordance with the power-managed mode clock that will be used.

After the power-managed mode is entered (either of the power-managed Run modes), an A/D acquisition or conversion may be started. Once an acquisition or conversion is started, the device should continue to be clocked by the same power-managed mode clock source until the conversion has been completed. If desired, the device may be placed into the corresponding power-managed Idle mode during the conversion.

If the power-managed mode clock frequency is less than 1 MHz, the A/D RC clock source should be selected.

Operation in Sleep mode requires that the A/D RC clock be selected. If bits, ACQT<2:0>, are set to '000' and a conversion is started, the conversion will be delayed one instruction cycle to allow execution of the SLEEP instruction and entry into Sleep mode. The IDLEN and SCS<1:0> bits in the OSCCON register must have already been cleared prior to starting the conversion.

27.2 CTMU Operation

The CTMU works by using a fixed current source to charge a circuit. The type of circuit depends on the type of measurement being made.

In the case of charge measurement, the current is fixed and the amount of time the current is applied to the circuit is fixed. The amount of voltage read by the A/D becomes a measurement of the circuit's capacitance.

In the case of time measurement, the current, as well as the capacitance of the circuit, is fixed. In this case, the voltage read by the A/D is representative of the amount of time elapsed from the time the current source starts and stops charging the circuit.

If the CTMU is being used as a time delay, both capacitance and current source are fixed, as well as the voltage supplied to the comparator circuit. The delay of a signal is determined by the amount of time it takes the voltage to charge to the comparator threshold voltage.

27.2.1 THEORY OF OPERATION

The operation of the CTMU is based on the equation for charge:

$$C = I \bullet \frac{dV}{dT}$$

More simply, the amount of charge measured in coulombs in a circuit is defined as current in amperes (I) multiplied by the amount of time in seconds that the current flows (t). Charge is also defined as the capacitance in farads (C) multiplied by the voltage of the circuit (V). It follows that:

$$I \bullet t = C \bullet V$$

The CTMU module provides a constant, known current source. The A/D Converter is used to measure (V) in the equation, leaving two unknowns: capacitance (C) and time (t). The above equation can be used to calculate capacitance or time, by either the relationship using the known fixed capacitance of the circuit:

$$t = (C \bullet V)/I$$

or by:

$$C = (I \bullet t)/V$$

using a fixed time that the current source is applied to the circuit.

27.2.2 CURRENT SOURCE

At the heart of the CTMU is a precision current source, designed to provide a constant reference for measurements. The level of current is user-selectable across three ranges or a total of two orders of magnitude, with the ability to trim the output in ±2% increments (nominal). The current range is selected by the IRNG<1:0> bits (CTMUICON<1:0>), with a value of '00' representing the lowest range.

Current trim is provided by the ITRIM<5:0> bits (CTMUICON<7:2>). These six bits allow trimming of the current source in steps of approximately 2% per step. Half of the range adjusts the current source positively and the other half reduces the current source. A value of '000000' is the neutral position (no change). A value of '100000' is the maximum negative adjustment (approximately -62%) and '011111' is the maximum positive adjustment (approximately +62%).

27.2.3 EDGE SELECTION AND CONTROL

CTMU measurements are controlled by edge events occurring on the module's two input channels. Each channel, referred to as Edge 1 and Edge 2, can be configured to receive input pulses from one of the edge input pins (CTED1 and CTED2) or CCPx Special Event Triggers. The input channels are level-sensitive, responding to the instantaneous level on the channel rather than a transition between levels. The inputs are selected using the EDG1SEL and EDG2SEL bit pairs (CTMUCONL<3:2, 6:5>).

In addition to source, each channel can be configured for event polarity using the EDGE2POL and EDGE1POL bits (CTMUCONL<7,4>). The input channels can also be filtered for an edge event sequence (Edge 1 occurring before Edge 2) by setting the EDGSEQEN bit (CTMUCONH<2>).

27.2.4 EDGE STATUS

The CTMUCON register also contains two status bits, EDG2STAT and EDG1STAT (CTMUCONL<1:0>). Their primary function is to show if an edge response has occurred on the corresponding channel. The CTMU automatically sets a particular bit when an edge response is detected on its channel. The level-sensitive nature of the input channels also means that the status bits become set immediately if the channel's configuration is changed and matches the channel's current state.

The module uses the edge status bits to control the current source output to external analog modules (such as the A/D Converter). Current is only supplied to external modules when only one (not both) of the status bits is set. Current is shut off when both bits are either set or cleared. This allows the CTMU to measure current only during the interval between edges. After both status bits are set, it is necessary to clear them before another measurement is taken. Both bits should be cleared simultaneously, if possible, to avoid re-enabling the CTMU current source.

In addition to being set by the CTMU hardware, the edge status bits can also be set by software. This permits a user application to manually enable or disable the current source. Setting either (but not both) of the bits enables the current source. Setting or clearing both bits at once disables the source.

EXAMPLE 27-4: ROUTINE FOR CAPACITIVE TOUCH SWITCH

```
#include "p18cxxx.h"

#define COUNT 500                       //@ 8MHz = 125uS.
#define DELAY for(i=0;i<COUNT;i++)
#define OPENSW 1000                     //Un-pressed switch value
#define TRIP 300                        //Difference between pressed
                                        //and un-pressed switch
#define HYST 65                         //amount to change
                                        //from pressed to un-pressed
#define PRESSED 1
#define UNPRESSED 0

int main(void)
{
    unsigned int Vread;                 //storage for reading
    unsigned int switchState;
    int i;

    //assume CTMU and A/D have been setup correctly
    //see Example 27-1 for CTMU & A/D setup
    setup();

    CTMUCONHbits.CTMUEN = 1;            //Enable the CTMU

    CTMUCONHbits.IDISSEN = 1;           //drain charge on the circuit
    DELAY;                              //wait 125us
    CTMUCONHbits.IDISSEN = 0;           //end drain of circuit

    CTMUCONLbits.EDG1STAT = 1;          //Begin charging the circuit
                                        //using CTMU current source
    DELAY;                              //wait for 125us
    CTMUCONLbits.EDG1STAT = 0;          //Stop charging circuit

    PIR1bits.ADIF = 0;                  //make sure A/D Int not set
    ADCON0bits.GO=1;                    //and begin A/D conv.
    while(!PIR1bits.ADIF);              //Wait for A/D convert complete

    Vread = ADRES;                      //Get the value from the A/D

    if(Vread < OPENSW - TRIP)
    {
        switchState = PRESSED;
    }
    else if(Vread > OPENSW - TRIP + HYST)
    {
        switchState = UNPRESSED;
    }
}
```

BNOV **Branch if Not Overflow**

Syntax: BNOV n

Operands: -128 ≤ n ≤ 127

Operation: if Overflow bit is '0',
(PC) + 2 + 2n → PC

Status Affected: None

Encoding:

1110	0101	nnnn	nnnn

Description: If the Overflow bit is '0', then the program will branch.

The 2's complement number '2n' is added to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is then a two-cycle instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:

If Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	Write to PC
No operation	No operation	No operation	No operation

If No Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	No operation

Example: HERE BNOV Jump

Before Instruction
PC = address (HERE)
After Instruction
If Overflow = 0;
PC = address (Jump)
If Overflow = 1;
PC = address (HERE + 2)

BNZ **Branch if Not Zero**

Syntax: BNZ n

Operands: -128 ≤ n ≤ 127

Operation: if Zero bit is '0',
(PC) + 2 + 2n → PC

Status Affected: None

Encoding:

1110	0001	nnnn	nnnn

Description: If the Zero bit is '0', then the program will branch.

The 2's complement number '2n' is added to the PC. Since the PC will have incremented to fetch the next instruction, the new address will be PC + 2 + 2n. This instruction is then a two-cycle instruction.

Words: 1

Cycles: 1(2)

Q Cycle Activity:

If Jump:

Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	Write to PC
No operation	No operation	No operation	No operation

If No Jump:

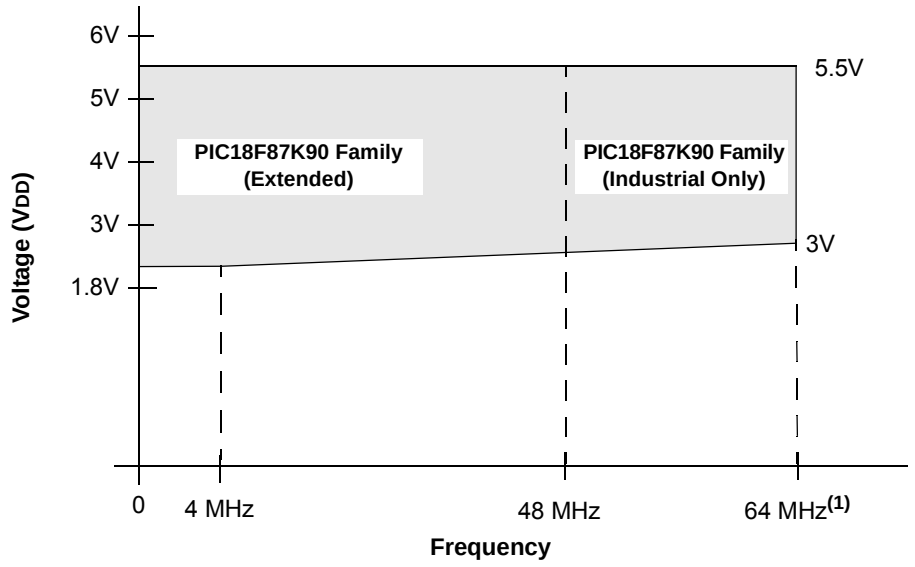
Q1	Q2	Q3	Q4
Decode	Read literal 'n'	Process Data	No operation

Example: HERE BNZ Jump

Before Instruction
PC = address (HERE)
After Instruction
If Zero = 0;
PC = address (Jump)
If Zero = 1;
PC = address (HERE + 2)

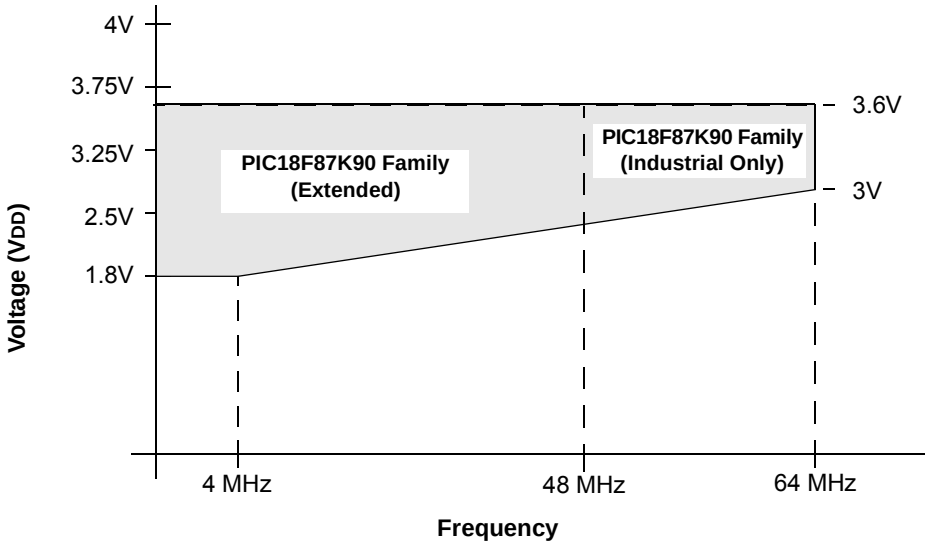
© 2009-2011 Microchip Technology Inc.

FIGURE 31-1: VOLTAGE-FREQUENCY GRAPH, REGULATOR ENABLED (INDUSTRIAL/EXTENDED)(1)



Note 1: FMAX = 64 MHz in all other modes. For VDD values, 1.8V to 3V, FMAX = (VDD – 1.72)/0.02 MHz.

FIGURE 31-2: VOLTAGE-FREQUENCY GRAPH, REGULATOR DISABLED (INDUSTRIAL/EXTENDED)(1,2)



Note 1: When the on-chip voltage regulator is disabled, VDD must be maintained so that VDD ≤ 3.6V.

2: For VDD values, 1.8V to 3V, FMAX = (VDD – 1.72)/0.02 MHz.

© 2009-2011 Microchip Technology Inc.

31.2 DC Characteristics: Power-Down and Supply Current PIC18F87K90 Family (Industrial/Extended) (Continued)

PIC18F87K90 Family		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended					
Param No.	**Device**	**Typ**	**Max**	**Units**	**Conditions**		
	Supply Current (IDD) Cont.(2,3)						
	All devices	42	73	µA	-40°C	VDD = 1.8V(4) Regulator Disabled	FOSC = 1 MHz (**PRI_IDLE** mode, EC oscillator)
		42	73	µA	+25°C		
		43	74	µA	+85°C		
		53	100	µA	+125°C		
	All devices	110	190	µA	-40°C	VDD = 3.3V(4) Regulator Disabled	
		110	195	µA	+25°C		
		110	195	µA	+85°C		
		130	250	µA	+125°C		
	All devices	280	450	µA	-40°C	VDD = 5V(5) Regulator Enabled	
		290	440	µA	+25°C		
		300	460	µA	+85°C		
		330	500	µA	+125°C		
	All devices	160	360	µA	-40°C	VDD = 1.8V(4) Regulator Disabled	FOSC = 4 MHz (**PRI_IDLE** mode, EC oscillator)
		160	360	µA	+25°C		
		170	370	µA	+85°C		
		200	400	µA	+125°C		
	All devices	330	650	µA	-40°C	VDD = 3.3V(4) Regulator Disabled	
		340	660	µA	+25°C		
		340	660	µA	+85°C		
		370	700	µA	+125°C		
	All devices	510	900	µA	-40°C	VDD = 5V(5) Regulator Enabled	
		520	950	µA	+25°C		
		540	990	µA	+85°C		
		600	1200	µA	+125°C		
	All devices	4.7	9	mA	-40°C	VDD = 3.3V(4) Regulator Disabled	FOSC = 64 MHz (**PRI_IDLE** mode, EC oscillator)
		4.8	9	mA	+25°C		
		4.8	10	mA	+85°C		
		5.2	12	mA	+125°C(7)		
	All devices	5.1	11	mA	-40°C	VDD = 5V(5) Regulator Enabled	
		5.1	11	mA	+25°C		
		5.2	12	mA	+85°C		
		5.7	14	mA	+125°C(7)		

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 a high-impedance state and tied to VDD or VSS, and all features that add delta current are disabled (such as WDT, SOSC oscillator, BOR, etc.).

2: 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.
The test conditions for all IDD measurements in active operation mode are:
OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD;
MCLR = VDD; WDT enabled/disabled as specified.

3: Standard, low-cost 32 kHz crystals have an operating temperature range of -10°C to +70°C. Extended temperature crystals are available at a much higher cost.

4: Voltage regulator disabled (ENVREG = 0, tied to VSS, $\overline{\text{RETEN}}$ (CONFIG1L<0>) = 1).

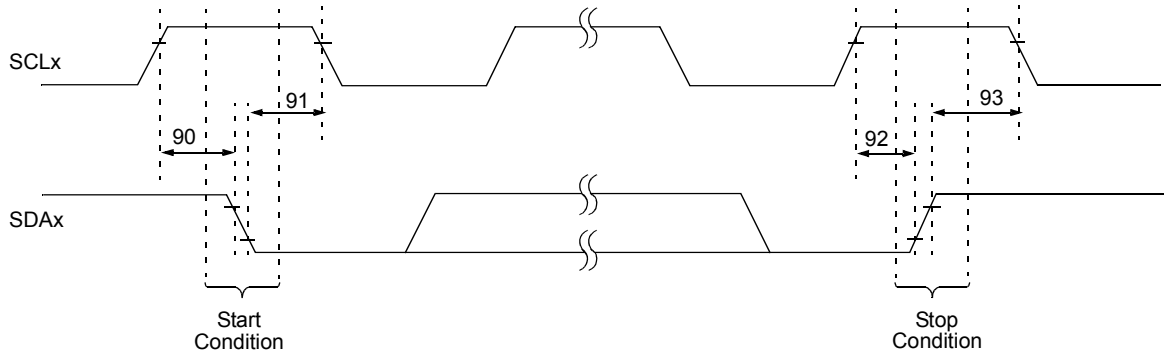
5: Voltage regulator enabled (ENVREG = 1, tied to VDD, SRETEN (WDTCON<4>) = 1 and $\overline{\text{RETEN}}$ (CONFIG1L<0>) = 0).

6: LCD glass is not connected; resistor current is not included.

7: 48 MHz maximum frequency at 125°C.

© 2009-2011 Microchip Technology Inc.

FIGURE 31-17: MSSP I²C™ BUS START/STOP BITS TIMING WAVEFORMS



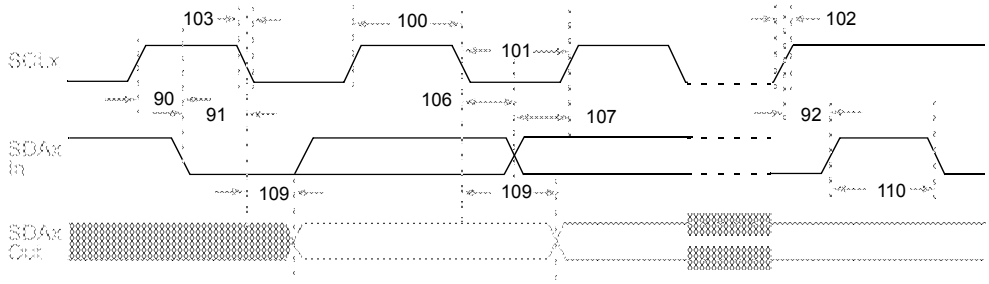
Note: Refer to Figure 31-3 for load conditions.

TABLE 31-20: MSSP I²C™ BUS START/STOP BITS REQUIREMENTS

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

Note 1: Maximum pin capacitance = 10 pF for all I²C™ pins.

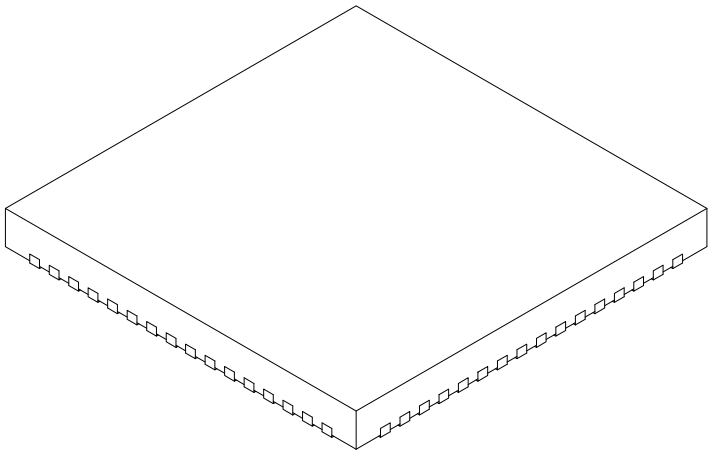
FIGURE 31-18: MSSP I²C™ BUS DATA TIMING



Note: Refer to Figure 31-3 for load conditions.

64-Lead Plastic Quad Flat, No Lead Package (MR) – 9x9x0.9 mm Body [QFN]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		64	
Pitch	e		0.50 BSC	
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3		0.20 REF	
Overall Width	E		9.00 BSC	
Exposed Pad Width	E2	7.05	7.15	7.50
Overall Length	D		9.00 BSC	
Exposed Pad Length	D2	7.05	7.15	7.50
Contact Width	b	0.18	0.25	0.30
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	-	-

Notes:

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

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

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

Microchip Technology Drawing C04-149B Sheet 2 of 2