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	69
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 24x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-TQFP
Supplier Device Package	80-TQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f85k90t-i-ptrsl

The PIC18F87K90 family is also largely pin-compatible with other PIC18 families, such as the PIC18F8720, PIC18F8722, PIC18F85J11, PIC18F8490, PIC18F85J90, PIC18F87J90 and PIC18F87J93 families of microcontrollers with LCD drivers. This allows a new dimension to the evolution of applications, allowing developers to select different price points within Microchip's PIC18 portfolio, while maintaining a similar feature set.

1.2 LCD Driver

The on-chip LCD driver includes many features that ease the integration of displays in low-power applications. These include an integrated internal resistor ladder, so bias voltages can be generated internally. This enables software-controlled contrast control and eliminates the need for external bias voltage resistors.

1.3 Other Special Features

- **Communications:** The PIC18F87K90 family incorporates a range of serial communication peripherals including two Enhanced USART, that support LIN/J2602, and two Master SSP modules capable of both SPI and I^2C™ (Master and Slave) modes of operation.
- **CCP Modules:** PIC18F87K90 family devices incorporate up to seven or five Capture/ Compare/PWM (CCP) modules. Up to six different time bases can be used to perform several different operations at once.
- **ECCP Modules:** The PIC18F87K90 family has three Enhanced CCP (ECCP) modules to maximize flexibility in control applications:
  - Up to eight different time bases for performing several different operations at once
  - Up to four PWM outputs for each module, for a total of 12 PWMs
  - Other beneficial features, such as polarity selection, programmable dead time, auto-shutdown and restart, and Half-Bridge and Full-Bridge Output modes
- **12-Bit A/D Converter:** The PIC18F87K90 family has differential ADC. It incorporates programmable acquisition time, allowing for a channel to be selected and a conversion to be initiated without waiting for a sampling period, and thus, reducing code overhead.
- **Charge Time Measurement Unit (CTMU):** The CTMU is a flexible analog module that provides accurate differential time measurement between pulse sources, as well as asynchronous pulse generation.

  Together with other on-chip analog modules, the CTMU can precisely measure time, measure capacitance or relative changes in capacitance, or generate output pulses that are independent of the system clock.
- **LP Watchdog Timer (WDT):** This enhanced version incorporates a 22-bit prescaler, allowing an extended time-out range that is stable across operating voltage and temperature. See **Section 31.0 "Electrical Characteristics"** for time-out periods.
- **Real-Time Clock and Calendar Module (RTCC):** The RTCC module is intended for applications requiring that accurate time be maintained for extended periods of time with minimum to no intervention from the CPU.

  The module is a 100-year clock and calendar with automatic leap year detection. The range of the clock is from 00:00:00 (midnight) on January 1, 2000 to 23:59:59 on December 31, 2099.

1.4 Details on Individual Family Members

Devices in the PIC18F87K90 family are available in 64-pin and 80-pin packages. Block diagrams for the two groups are shown in Figure 1-1 and Figure 1-2.

The devices are differentiated from each other in these ways:

- Flash Program Memory:
  - PIC18FX5K90 (PIC18F65K90 and PIC18F85K90) – 32 Kbytes
  - PIC18FX6K90 (PIC18F66K90 and PIC18F86K90) – 64 Kbytes
  - PIC18FX7K90 (PIC18F67K90 and PIC18F87K90) – 128 Kbytes
- Data RAM:
  - All devices except PIC18FX5K90 – 4 Kbytes
  - PIC18FX5K90 – 2 Kbytes
- I/O Ports:
  - PIC18F6XK90 (64-pin devices) – 7 bidirectional ports
  - PIC18F8XK90 (80-pin devices) – 9 bidirectional ports
- LCD Pixels:
  - PIC18F6XK90 – 132 pixels (33 SEGs x 4 COMs)
  - PIC18F8XK90 – 192 pixels (48 SEGs x 4 COMs)
- CCP Module:
  - All devices except PIC18FX5K90 have seven CCP modules, PIC18FX5K90 has only five CCP modules
- Timers:
  - All devices except 18FX5K90 have six 8-bit timers and five 16-bit timers, PIC18FX5K90 has only four 8-bit timers and four 16-bit timers.
- A/D Channels:
  - All PIC18F8XK90 devices have 24 A/D channels, all PIC18F6XK90 devices have 16 A/D channels

All other features for devices in this family are identical. These are summarized in Table 1-1 and Table 1-2.

The pinouts for all devices are listed in Table 1-3 and Table 1-4.

© 2009-2011 Microchip Technology Inc.

TABLE 6-2: PIC18F87K90 FAMILY REGISTER FILE SUMMARY

Address	File Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR
EF4h	LCDCON	LCDEN	SLPEN	WERR	—	CS1	CS0	LMUX1	LMUX0	000- 0000
EF5h	LCDPS	WFT	BIASMD	LCDA	WA	LP3	LP2	LP1	LP0	0000 0000
EF6h	LCDSE0	SE07	SE06	SE05	SE04	SE03	SE02	SE01	SE00	0000 0000
EF7h	LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE09	SE08	0000 0000
EF8h	LCDSE2	SE23	SE22	SE21	SE20	SE19	SE18	SE17	SE16	0000 0000
EF9h	LCDSE3	SE31	SE30	SE29	SE28	SE27	SE26	SE25	SE24	0000 0000
EFAh	LCDSE4	SE39	SE38	S37	SE36	SE35	SE34	SE33	SE32	0000 0000
EFBh	LCDSE5(2)	SE47	SE46	SE45	SE44	SE43	SE42	SE41	SE40	0000 0000
EFCh	LCDRL	LRLAP1	LRLAP0	LRLBP1	LRLBP0	—	LRLAT2	LRLAT1	LRLAT0	0000 -000
EFDh	LCDREF	LCDIRE	LCDIRS	LCDCST2	LCDCST1	LCDCST0	VLCD3PE	VLCD2PE	VLCD1PE	0000 0000
EFEh	SSP2CON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000
EFFh	SSP2CON1	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000
F00h	SSP2STAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	0000 0000
F01h	SSP2ADD	MSSP Address Register in I^{2}C™ Slave Mode. SSP1 Baud Rate Reload Register in I^{2}C Master Mode								0000 0000
F02h	SSP2BUF	MSSP Receive Buffer/Transmit Register								xxxx xxxx
F03h	T4CON	—	T4OUTPS3	T4OUTPS2	T4OUTPS1	T4OUTPS0	TMR4ON	T4CKPS1	T4CKPS0	-000 0000
F04h	PR4	Timer4 Period Register								0000 0000
F05h	TMR4	Timer4 Register								1111 1111
F06h	CCP7CON	—	—	DC7B1	DC7B0	CCP7M3	CCP7M2	CCP7M1	CCP7M0	--00 0000
F07h	CCPR7L	Capture/Compare/PWM Register 7 Low Byte								xxxx xxxx
F08h	CCPR7H	Capture/Compare/PWM Register7 High Byte								xxxx xxxx
F09h	CCP6CON	—	—	DC6B1	DC6B0	CCP6M3	CCP6M2	CCP6M1	CCP6M0	--00 0000
F0Ah	CCPR6L	Capture/Compare/PWM Register 6 Low Byte								xxxx xxxx
F0Bh	CCPR6H	Capture/Compare/PWM Register6 High Byte								xxxx xxxx
F0Ch	CCP5CON	—	—	DC5B1	DC5B0	CCP5M3	CCP5M2	CCP5M1	CCP5M0	--00 0000
F0Dh	CCPR5L	Capture/Compare/PWM Register 5 Low Byte								xxxx xxxx
F0Eh	CCPR5H	Capture/Compare/PWM Register 5 High Byte								xxxx xxxx
F0Fh	CCP4CON	—	—	DC4B1	DC4B0	CCP4M3	CCP4M2	CCP4M1	CCP4M0	--00 0000
F10h	CCPR4L	Capture/Compare/PWM Register 4 Low Byte								xxxx xxxx
F11h	CCPR4H	Capture/Compare/PWM Register 4 High Byte								xxxx xxxx
F12h	T5GCON	TMR5GE	T5GPOL	T5GTM	T5GSPM	T5GGO/$\overline{T5DONE}$	T5GVAL	T5GSS1	T5GSS0	0000 0000
F13h	T5CON	TMR5CS1	TMR5CS0	T5CKPS1	T5CKPS0	SOSCEN	$\overline{T5SYNC}$	RD16	TMR5ON	0000 0000
F14h	TMR5L	Timer5 Register Low Byte								0000 0000
F15h	TMR5H	Timer5 Register High Byte								xxxx xxxx
F16h	PMD3	CCP10MD(3)	CCP9MD(3)	CCP8MD	CCP7MD	CCP6MD	CCP5MD	CCP4MD	TMR12MD(3)	0000 0000
F17h	PMD2	TMR10MD(3)	TMR8MD	TMR7MD(3)	TMR6MD	TMR5MD	CMP3MD	CMP2MD	CMP1MD	0000 0000
F18h	PMD1	—	CTMUMD	RTCCMD	TMR4MD	TMR3MD	TMR2MD	TMR1MD	—	-000 000-
F19h	PMD0	CCP3MD	CCP2MD	CCP1MD	UART2MD	UART1MD	SSP2MD	SSP1MD	ADCMD	0000 0000
F1Ah	PSTR3CON	CMPL1	CMPL0	—	STRSYNC	STRD	STRC	STRB	STRA	00-0 0001
F1Bh	PSTR2CON	CMPL1	CMPL0	—	STRSYNC	STRD	STRC	STRB	STRA	00-0 0001
F1Ch	TXREG2	Transmit Data FIFO								xxxx xxxx
F1Dh	RCREG2	Receive Data FIFO								0000 0000
F1Eh	SPBRG2	USART2 Baud Rate Generator Low Byte								0000 0000
F1Fh	SPBRGH2	USART2 Baud Rate Generator High Byte								0000 0000
F20h	BAUDCON2	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN	0100 0-00
F21h	TXSTA2	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	0000 0010
F22h	RCSTA2	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x
F23h	ANCON2	ANSEL23	ANSEL22	ANSEL21	ANSEL20	ANSEL19	ANSEL18	ANSEL17	ANSEL16	1111 1111

Note 1: This bit is available when Master Clear is disabled (MCLRE = 0). When MCLRE is set, the bit is unimplemented.
2: Unimplemented in 64-pin devices (PIC18F6XK90).
3: Unimplemented in devices with a program memory of 32 Kbytes (PIC18FX5K90).

9.0 8 x 8 HARDWARE MULTIPLIER

9.1 Introduction

All PIC18 devices include an 8 x 8 hardware multiplier as part of the ALU. The multiplier performs an unsigned operation and yields a 16-bit result that is stored in the product register pair, PRODH:PRODL. The multiplier's operation does not affect any flags in the STATUS register.

Making multiplication a hardware operation allows it to be completed in a single instruction cycle. This has the advantages of higher computational throughput and reduced code size for multiplication algorithms and allows PIC18 devices to be used in many applications previously reserved for digital-signal processors. A comparison of various hardware and software multiply operations, along with the savings in memory and execution time, is shown in Table 9-1.

9.2 Operation

Example 9-1 shows the instruction sequence for an 8 x 8 unsigned multiplication. Only one instruction is required when one of the arguments is already loaded in the WREG register.

Example 9-2 shows the sequence to do an 8 x 8 signed multiplication. To account for the sign bits of the arguments, each argument's Most Significant bit (MSb) is tested and the appropriate subtractions are done.

EXAMPLE 9-1: 8 x 8 UNSIGNED MULTIPLY ROUTINE

```
MOVF    ARG1, W     ;
MULWF   ARG2        ; ARG1 * ARG2 ->
                    ; PRODH:PRODL
```

EXAMPLE 9-2: 8 x 8 SIGNED MULTIPLY ROUTINE

```
MOVF    ARG1, W
MULWF   ARG2        ; ARG1 * ARG2 ->
                    ; PRODH:PRODL
BTFSC   ARG2, SB    ; Test Sign Bit
SUBWF   PRODH, F    ; PRODH = PRODH
                    ;         - ARG1
MOVF    ARG2, W
BTFSC   ARG1, SB    ; Test Sign Bit
SUBWF   PRODH, F    ; PRODH = PRODH
                    ;         - ARG2
```

TABLE 9-1: PERFORMANCE COMPARISON FOR VARIOUS MULTIPLY OPERATIONS

Routine	Multiply Method	Program Memory (Words)	Cycles (Max)	Time			
				@ 64 MHz	@ 48 MHz	@ 10 MHz	@ 4 MHz
8 x 8 Unsigned	Without Hardware Multiply	13	69	4.3 μs	5.7 μs	27.6 μs	69 μs
	Hardware Multiply	1	1	62.5 ns	83.3 ns	400 ns	1 μs
8 x 8 Signed	Without Hardware Multiply	33	91	5.6 μs	7.5 μs	36.4 μs	91 μs
	Hardware Multiply	6	6	375 ns	500 ns	2.4 μs	6 μs
16 x 16 Unsigned	Without Hardware Multiply	21	242	15.1 μs	20.1 μs	96.8 μs	242 μs
	Hardware Multiply	28	28	1.7 μs	2.3 μs	11.2 μs	28 μs
16 x 16 Signed	Without Hardware Multiply	52	254	15.8 μs	21.2 μs	101.6 μs	254 μs
	Hardware Multiply	35	40	2.5 μs	3.3 μs	16.0 μs	40 μs

REGISTER 10-8: PIR5: PERIPHERAL INTERRUPT FLAG REGISTER 5

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TMR7GIF(1)	TMR12IF(1)	TMR10IF(1)	TMR8IF	TMR7IF(1)	TMR6IF	TMR5IF	TMR4IF
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 **TMR7GIF:** TMR7 Gate Interrupt Flag bits(1)
1 = TMR gate interrupt occurred (must be cleared in software)
0 = No TMR gate interrupt occurred

bit 6 **TMR12IF:** TMR12 to PR12 Match Interrupt Flag bit(1)
1 = TMR12 to PR12 match occurred (must be cleared in software)
0 = No TMR12 to PR12 match occurred

bit 5 **TMR10IF:** TMR10 to PR10 Match Interrupt Flag bit(1)
1 = TMR10 to PR10 match occurred (must be cleared in software)
0 = No TMR10 to PR10 match occurred

bit 4 **TMR8IF:** TMR8 to PR8 Match Interrupt Flag bit
1 = TMR8 to PR8 match occurred (must be cleared in software)
0 = No TMR8 to PR8 match occurred

bit 3 **TMR7IF:** TMR7 Overflow Interrupt Flag bit(1)
1 = TMR7 register overflowed (must be cleared in software)
0 = TMR7 register did not overflow

bit 2 **TMR6IF:** TMR6 to PR6 Match Interrupt Flag bit
1 = TMR6 to PR6 match occurred (must be cleared in software)
0 = No TMR6 to PR6 match occurred

bit 1 **TMR5IF:** TMR5 Overflow Interrupt Flag bit
1 = TMR5 register overflowed (must be cleared in software)
0 = TMR5 register did not overflow

bit 0 **TMR4IF:** TMR4 to PR4 Match Interrupt Flag bit
1 = TMR4 to PR4 match occurred (must be cleared in software)
0 = No TMR4 to PR4 match occurred

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

© 2009-2011 Microchip Technology Inc.

TABLE 11-4: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on Page:
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	78
LATB	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	78
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	78
INTCON	GIE/GIEH	PEIE/GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	75
INTCON2	$\overline{\text{RBPU}}$	INTEDG0	INTEDG1	INTEDG2	INTEDG3	TMR0IP	INT3IP	RBIP	75
INTCON3	INT2IP	INT1IP	INT3IE	INT2IE	INT1IE	INT3IF	INT2IF	INT1IF	75
LCDSE1	SE15	SE14	SE13	SE12	SE11	SE10	SE09	SE08	83
LCDSE3	SE31	SE30	SE29	SE28	SE27	SE26	SE25	SE24	83

Legend: Shaded cells are not used by PORTB.

© 2009-2011 Microchip Technology Inc.

13.2 Timer1 Operation

The Timer1 module is an 8 or 16-bit incrementing counter that is accessed through the TMR1H:TMR1L register pair.

When used with an internal clock source, the module is a timer and increments on every instruction cycle. When used with an external clock source, the module can be used as either a timer or counter. It increments on every selected edge of the external source.

Timer1 is enabled by configuring the TMR1ON and TMR1GE bits in the T1CON and T1GCON registers, respectively.

When SOSC is selected as a Crystal mode (by SOSCEL), the RC1/SOSCI/ECCP2/P2A/SEG32 and RC0/SOSCO/SCLKI pins become inputs. This means the values of TRISC<1:0> are ignored and the pins are read as '0'.

13.3 Clock Source Selection

The TMR1CS<1:0> and SOSCEN bits of the T1CON register are used to select the clock source for Timer1. Register 13-1 displays the clock source selections.

13.3.1 INTERNAL CLOCK SOURCE

When the internal clock source is selected, the TMR1H:TMR1L register pair will increment on multiples of FOSC, as determined by the Timer1 prescaler.

13.3.2 EXTERNAL CLOCK SOURCE

When the external clock source is selected, the Timer1 module may work as a timer or a counter.

When enabled to count, Timer1 is incremented on the rising edge of the external clock input, T1CKI. Either of these external clock sources can be synchronized to the microcontroller system clock or they can run asynchronously.

When used as a timer with a clock oscillator, an external, 32.768 kHz crystal can be used in conjunction with the dedicated internal oscillator circuit.

Note: In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge after any one or more of the following conditions:

- Timer1 is enabled after a POR Reset
- Write to TMR1H or TMR1L
- Timer1 is disabled
- Timer1 is disabled (TMR1ON = 0)

When T1CKI is high, Timer1 is enabled (TMR1ON = 1) when T1CKI is low.

TABLE 13-1: TIMER1 CLOCK SOURCE SELECTION

TMR1CS1	TMR1CS0	SOSCEN	Clock Source
0	1	x	Clock Source (FOSC)
0	0	x	Instruction Clock (FOSC/4)
1	0	0	External Clock on T1CKI Pin
1	0	1	Oscillator Circuit on SOSCI/SOSCO Pins

REGISTER 15-1: TxCON: TIMER3/5/7 CONTROL REGISTER

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TMRxCS1	TMRxCS0	TxCKPS1	TxCKPS0	SOSCEN	$\overline{\text{TxSYNC}}$	RD16	TMRxON
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 7-6 **TMRxCS<1:0>:** Timerx Clock Source Select bits

10 = The Timer1 clock source is either a pin or an oscillator depending on the SOSCEN bit.
SOSCEN = 0:
External clock is from the T1CKI pin (on the rising edge).
SOSCEN = 1:
Crystal oscillator is on the SOSCI/SOSCO pins.
01 = Timerx clock source is the system clock (FOSC)(1)
00 = Timerx clock source is the instruction clock (FOSC/4)

bit 5-4 **TxCKPS<1:0>**: Timerx Input Clock Prescale Select bits

11 = 1:8 Prescale value
10 = 1:4 Prescale value
01 = 1:2 Prescale value
00 = 1:1 Prescale value

bit 3 **SOSCEN**: SOSC Oscillator Enable bit

1 = SOSC is enabled for Timerx (based on SOSCSEL fuses)
0 = SOSC is disabled for Timerx

bit 2 **TxSYNC:** Timerx External Clock Input Synchronization Control bit
(Not usable if the device clock comes from Timer1/Timer3.)

When TMRxCS<1:0> = 10:
1 = Do not synchronize external clock input
0 = Synchronize external clock input
When TMRxCS<1:0> = 0x:
This bit is ignored; Timer3 uses the internal clock.

bit 1 **RD16:** 16-Bit Read/Write Mode Enable bit

1 = Enables register read/write of Timerx in one 16-bit operation
0 = Enables register read/write of Timerx in two 8-bit operations

bit 0 **TMRxON:** Timerx On bit

1 = Enables Timerx
0 = Stops Timerx

Note 1: The FOSC clock source should not be selected if the timer will be used with the ECCP capture/compare features.

© 2009-2011 Microchip Technology Inc.

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

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Reset Values on Page:
CCP6CON	—	—	DC6B1	DC6B0	CCP6M3	CCP6M2	CCP6M1	CCP6M0	82
CCP7CON	—	—	DC7B1	DC7B0	CCP7M3	CCP7M2	CCP7M1	CCP7M0	82
CCP8CON	—	—	DC8B1	DC8B0	CCP8M3	CCP8M2	CCP8M1	CCP8M0	80
CCP9CON(1)	—	—	DC9B1	DC9B0	CCP9M3	CCP9M2	CCP9M1	CCP9M0	80
CCP10CON(1)	—	—	DC10B1	DC10B0	CCP10M3	CCP10M2	CCP10M1	CCP10M0	81
CCPTMRS1	C7TSEL1	C7TSEL0	—	C6TSEL0	—	C5TSEL0	C4TSEL1	C4TSEL0	81
CCPTMRS2	—	—	—	C10TSEL0	—	C9TSEL0	C8TSEL1	C8TSEL0	81

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.

18.4 PWM Mode

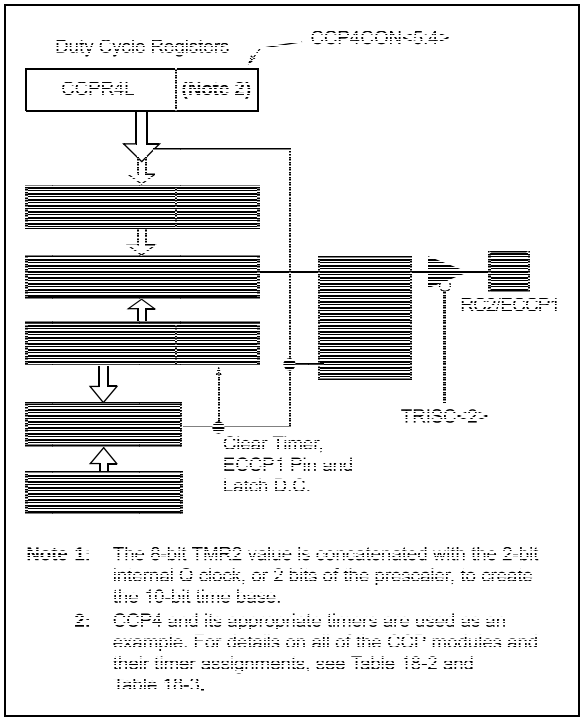
In Pulse-Width Modulation (PWM) mode, the CCP4 pin produces up to a 10-bit resolution PWM output. Since the CCP4 pin is multiplexed with a PORTC or PORTE data latch, the appropriate TRIS bit must be cleared to make the CCP4 pin an output.

Note:	Clearing the CCP4CON register will force the RC1 or RE7 output latch (depending on device configuration) to the default low level. This is not the PORTC or PORTE I/O data latch.

Figure 18-3 shows a simplified block diagram of the ECCP1 module in PWM mode.

For a step-by-step procedure on how to set up the CCP module for PWM operation, see **Section 18.4.3 "Setup for PWM Operation"**.

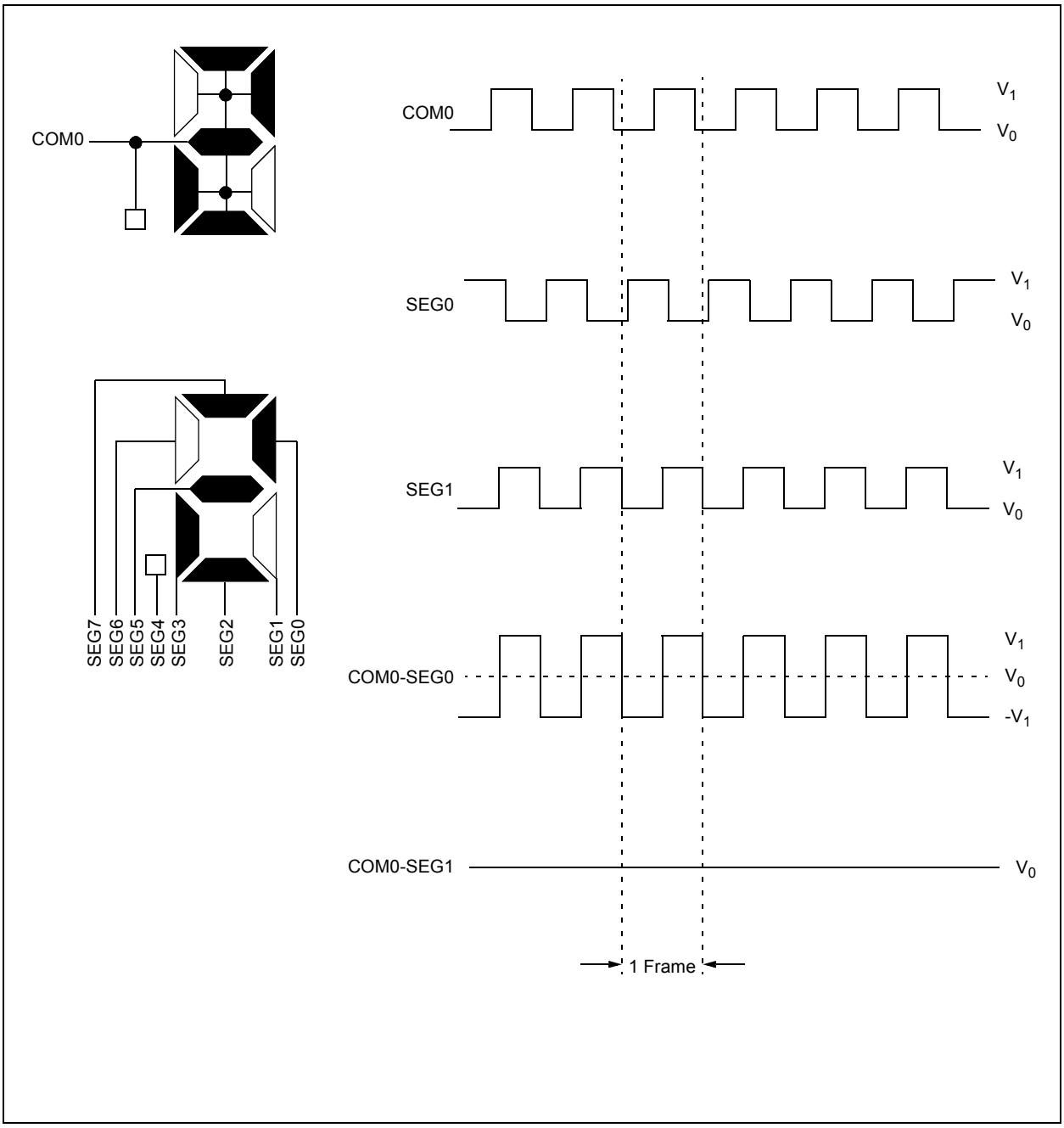
FIGURE 18-3: SIMPLIFIED PWM BLOCK DIAGRAM



Note 1: The 8-bit TMR2 value is concatenated with the 2-bit internal Q clock, or 2 bits of the prescaler, to create the 10-bit time base.

2: CCP4 and its appropriate timers are used as an example. For details on all of the CCP modules and their timer assignments, see Table 18-2 and Table 18-1.

FIGURE 20-7: TYPE-A/TYPE-B WAVEFORMS IN STATIC DRIVE



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.

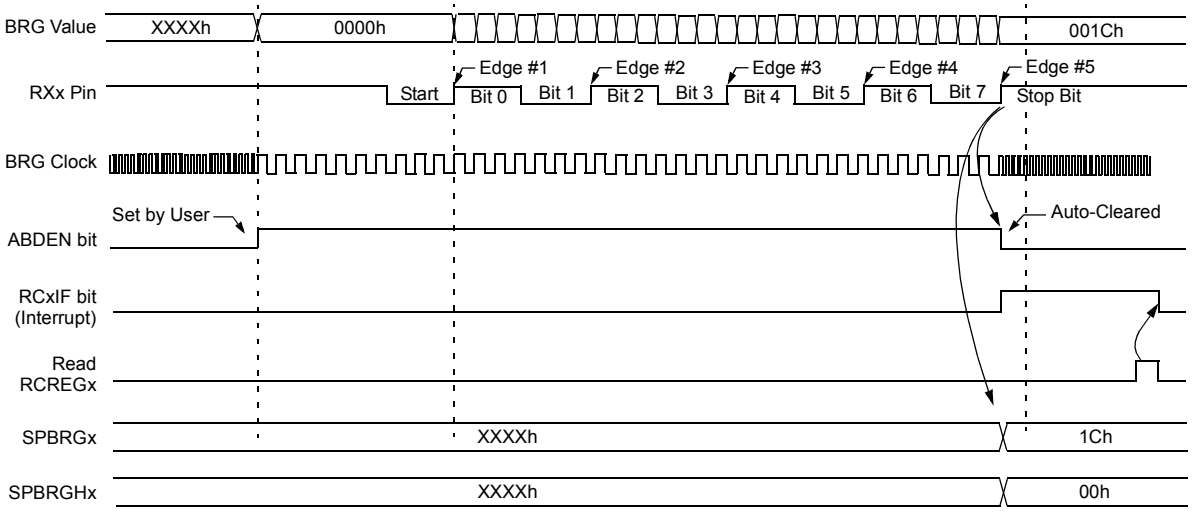
The operation of each Enhanced USART module is controlled through three registers:

- 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.

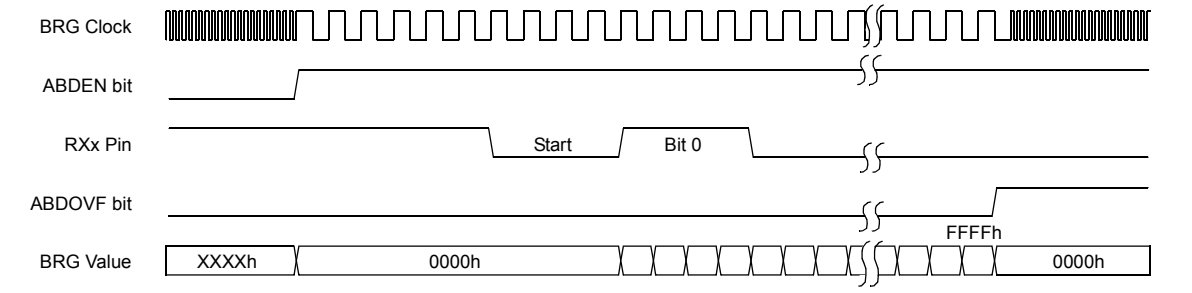
Note:	Throughout this section, references to register and bit names that may be associated with a specific EUSART module are referred to generically by the use of ‘x’ in place of the specific module number. Thus, “RCSTAx” might refer to the Receive Status register for either EUSART1 or EUSART2.

FIGURE 22-1: AUTOMATIC BAUD RATE CALCULATION



Note: The ABD sequence requires the EUSART module to be configured in Asynchronous mode and WUE = 0.

FIGURE 22-2: BRG OVERFLOW SEQUENCE



© 2009-2011 Microchip Technology Inc.

22.2.4 AUTO-WAKE-UP ON SYNC BREAK CHARACTER

During Sleep mode, all clocks to the EUSART are suspended. Because of this, the Baud Rate Generator is inactive and a proper byte reception cannot be performed. The auto-wake-up feature allows the controller to wake-up due to activity on the RXx/DTx line while the EUSART is operating in Asynchronous mode.

The auto-wake-up feature is enabled by setting the WUE bit (BAUDCONx<1>). Once set, the typical receive sequence on RXx/DTx is disabled and the EUSART remains in an Idle state, monitoring for a wake-up event independent of the CPU mode. A wake-up event consists of a high-to-low transition on the RXx/DTx line. (This coincides with the start of a Sync Break or a Wake-up Signal character for the LIN/J2602 protocol.)

Following a wake-up event, the module generates an RCxIF interrupt. The interrupt is generated synchronously to the Q clocks in normal operating modes (Figure 22-8) and asynchronously if the device is in Sleep mode (Figure 22-9). The interrupt condition is cleared by reading the RCREGx register.

The WUE bit is automatically cleared once a low-to-high transition is observed on the RXx line following the wake-up event. At this point, the EUSART module is in Idle mode and returns to normal operation. This signals to the user that the Sync Break event is over.

22.2.4.1 Special Considerations Using Auto-Wake-up

Since auto-wake-up functions by sensing rising edge transitions on RXx/DTx, information with any state changes before the Stop bit may signal a false End-of-Character (EOC) and cause data or framing errors. To work properly, therefore, the initial character in the transmission must be all ‘0’s. This can be 00h (8 bits) for standard RS-232 devices or 000h (12 bits) for LIN/J2602 bus.

Oscillator start-up time must also be considered, especially in applications using oscillators with longer start-up intervals (i.e., HS or HSPLL mode). The Sync Break (or Wake-up Signal) character must be of sufficient length and be followed by a sufficient interval to allow enough time for the selected oscillator to start and provide proper initialization of the EUSART.

© 2009-2011 Microchip Technology Inc.

The module is enabled by setting the HLVDEN bit (HLVDCON<4>). Each time the HLVD module is enabled, the circuitry requires some time to stabilize. The IRVST bit (HLVDCON<5>) is a read-only bit used to indicate when the circuit is stable. The module can only generate an interrupt after the circuit is stable and IRVST is set.

The VDIRMAG bit (HLVDCON<7>) determines the overall operation of the module. When VDIRMAG is cleared, the module monitors for drops in VDD below a predetermined set point. When the bit is set, the module monitors for rises in VDD above the set point.

26.1 Operation

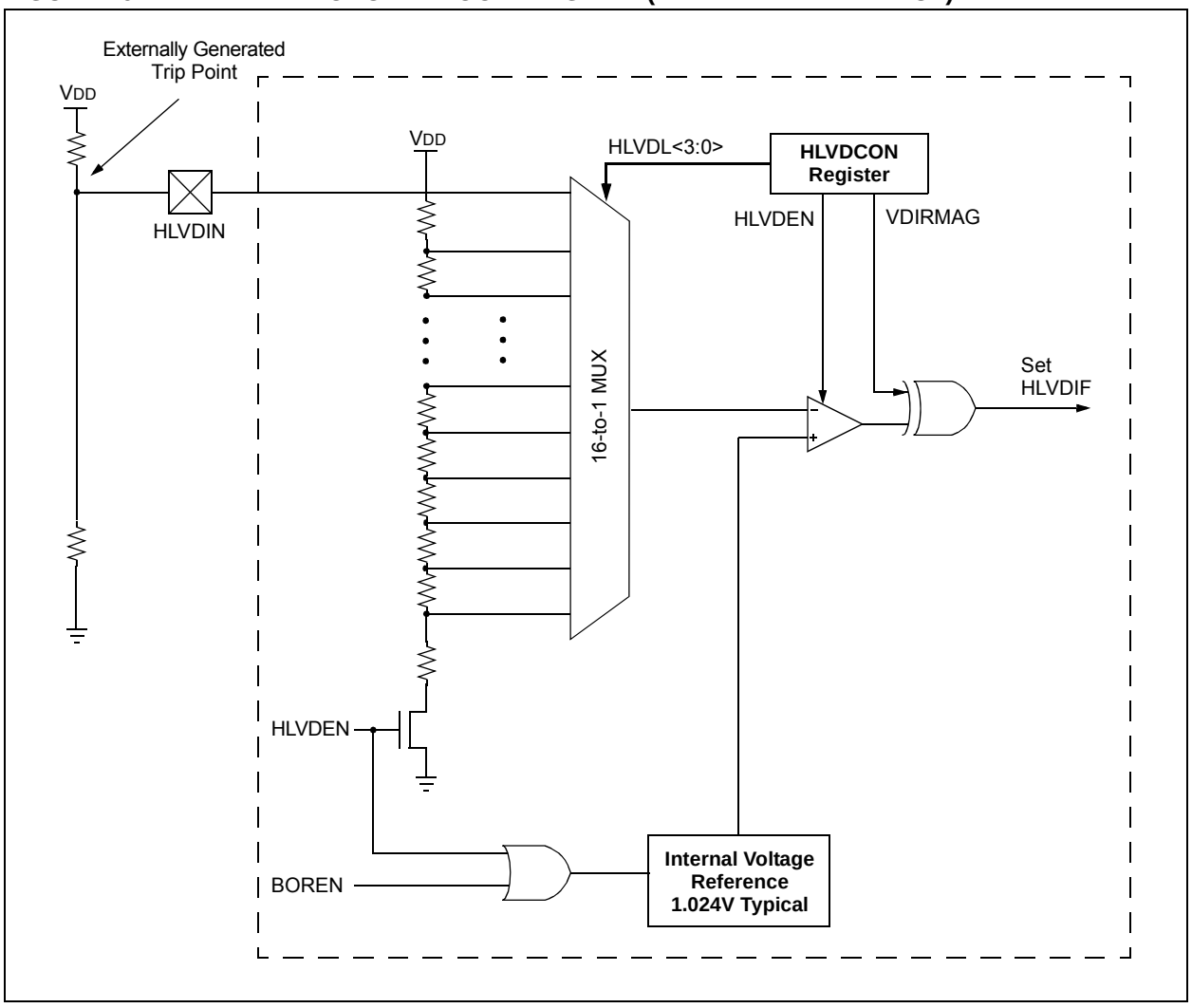
When the HLVD module is enabled, a comparator uses an internally generated reference voltage as the set point. The set point is compared with the trip point, where each node in the resistor divider represents a trip point voltage. The "trip point" voltage is the voltage level at which the device detects a high or low-voltage event, depending on the configuration of the module.

When the supply voltage is equal to the trip point, the voltage tapped off of the resistor array is equal to the internal reference voltage generated by the voltage reference module. The comparator then generates an interrupt signal by setting the HLVDIF bit.

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

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

FIGURE 26-1: HLVD MODULE BLOCK DIAGRAM (WITH EXTERNAL INPUT)



© 2009-2011 Microchip Technology Inc.

EXAMPLE 27-2: CURRENT CALIBRATION ROUTINE

```
#include "p18cxxx.h"

#define COUNT 500                       //@ 8MHz = 125uS.
#define DELAY for(i=0;i<COUNT;i++)
#define RCAL .027                       //R value is 4200000 (4.2M)
                                        //scaled so that result is in
                                        //1/100th of uA
#define ADSCALE 1023                    //for unsigned conversion 10 sig bits
#define ADREF 3.3                       //Vdd connected to A/D Vr+

int main(void)
{
    int i;
    int j = 0; //index for loop
    unsigned int Vread = 0;
    double VTot = 0;
    float Vavg=0, Vcal=0, CTMUISrc = 0;  //float values stored for calcs

//assume CTMU and A/D have been setup correctly
//see Example 25-1 for CTMU & A/D setup
setup();

CTMUCONHbits.CTMUEN = 1;                //Enable the CTMU
    for(j=0;j<10;j++)
    {
        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
        PIR1bits.ADIF = 0;              //Clear A/D Interrupt Flag
        VTot += Vread;                  //Add the reading to the total
    }

    Vavg = (float)(VTot/10.000);        //Average of 10 readings
    Vcal = (float)(Vavg/ADSCALE*ADREF);
    CTMUISrc = Vcal/RCAL;               //CTMUISrc is in 1/100ths of uA

}
```

REGISTER 28-4: CONFIG2H: CONFIGURATION REGISTER 2 HIGH (BYTE ADDRESS 300003h)

U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	WDTPS4	WDTPS3	WDTPS2	WDTPS1	WDTPS0	WDTEN1	WDTEN0
bit 7							bit 0

Legend:	P = Programmable bit		
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 **Unimplemented:** Read as ‘0’

bit 6-2 **WDTPS<4:0>:** Watchdog Timer Postscale Select bits

11111 = 1:1,048,576
10011 = 1:524,288
10010 = 1:262,144
10001 = 1:131,072
10000 = 1:65,536
01111 = 1:32,768
01110 = 1:16,384
01101 = 1:8,192
01100 = 1:4,096
01011 = 1:2,048
01010 = 1:1,024
01001 = 1:512
01000 = 1:256
00111 = 1:128
00110 = 1:64
00101 = 1:32
00100 = 1:16
00011 = 1:8
00010 = 1:4
00001 = 1:2
00000 = 1:1

bit 1-0 **WDTEN<1:0>:** Watchdog Timer Enable bits

11 = WDT is enabled in hardware; SWDTEN bit is disabled
10 = WDT is controlled by the SWDTEN bit setting
01 = WDT is enabled only while device is active and is disabled in Sleep mode; SWDTEN bit is disabled
00 = WDT is disabled in hardware; SWDTEN bit is disabled

© 2009-2011 Microchip Technology Inc.

28.2 Watchdog Timer (WDT)

For the PIC18F87K90 family of devices, the WDT is driven by the LF-INTOSC source. When the WDT is enabled, the clock source is also enabled. The nominal WDT period is 4 ms and has the same stability as the LF-INTOSC oscillator.

The 4 ms period of the WDT is multiplied by a 16-bit postscaler. Any output of the WDT postscaler is selected by a multiplexer, controlled by bits in Configuration Register 2H. Available periods range from 4 ms to 4,194 seconds (about one hour). The WDT and postscaler are cleared when any of the following events occur: a SLEEP or CLRWDT instruction is executed, the IRCF bits (OSCCON<6:4>) are changed or a clock failure has occurred.

The WDT can be operated in one of four modes as determined by the CONFIG2H bits (WDTEN<1:0>) The four modes are:

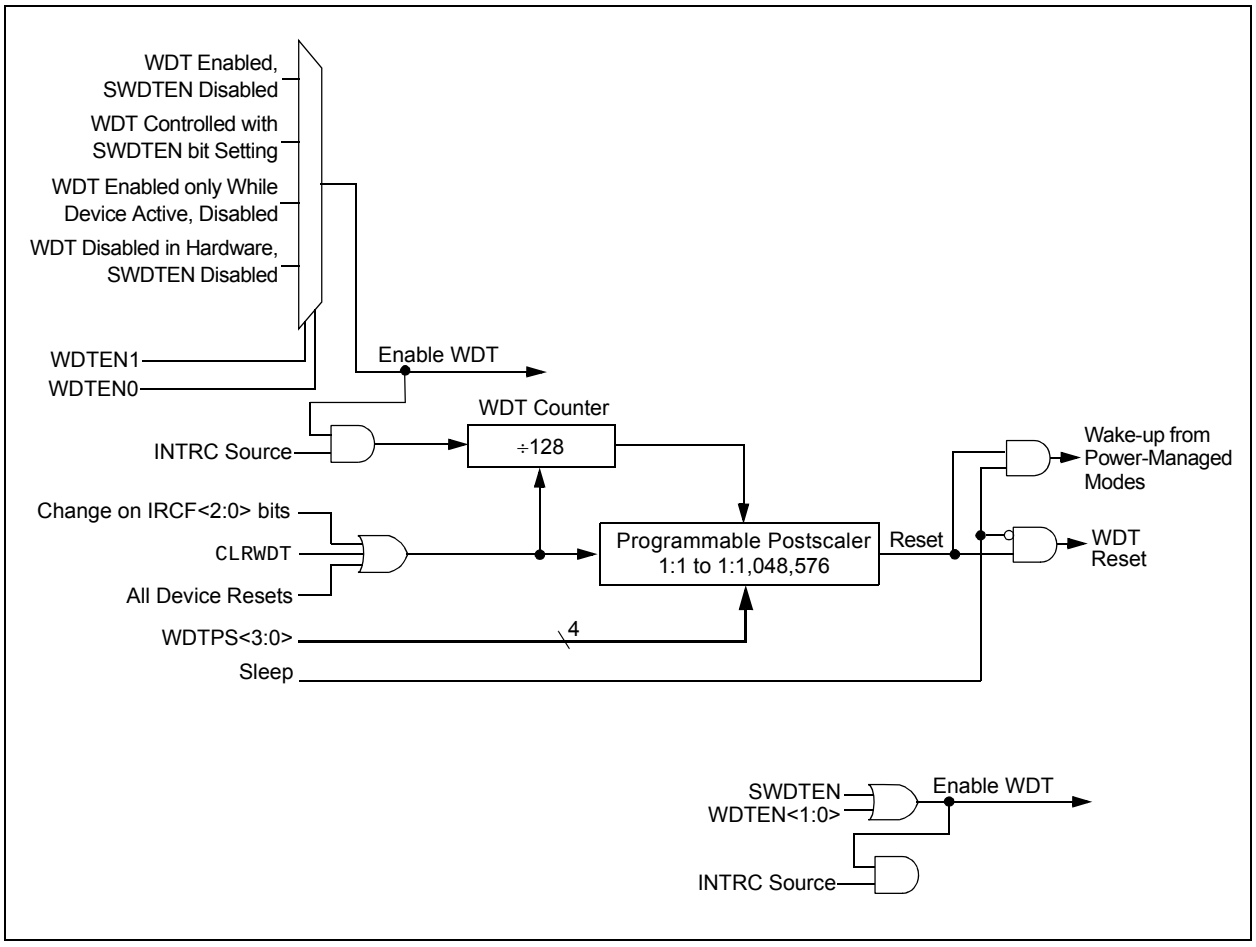
- WDT Enabled
- WDT Disabled
- WDT under Software Control (WDTCON<0>, SWDTEN)
- WDT
  - Enabled during normal operation
  - Disabled during Sleep

Note 1: The CLRWDT and SLEEP instructions clear the WDT and postscaler counts when executed.

2: Changing the setting of the IRCF bits (OSCCON<6:4>) clears the WDT and postscaler counts.

3: When a CLRWDT instruction is executed, the postscaler count will be cleared.

FIGURE 28-1: WDT BLOCK DIAGRAM



© 2009-2011 Microchip Technology Inc.

GOTO **Unconditional Branch**

Syntax: GOTO k

Operands: $0 \le k \le 1048575$

Operation: $k \rightarrow PC<20:1>$

Status Affected: None

Encoding:

1st word (k<7:0>)	1110	1111	k_7kkk	$kkkk_0$
2nd word(k<19:8>)	1111	$k_{19}kkk$	kkkk	$kkkk_8$

Description: GOTO allows an unconditional branch anywhere within entire 2-Mbyte memory range. The 20-bit value 'k' is loaded into PC<20:1>. GOTO is always a two-cycle instruction.

Words: 2

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read literal 'k'<7:0>,	No operation	Read literal 'k'<19:8>, Write to PC
No operation	No operation	No operation	No operation

Example: GOTO THERE

After Instruction
PC = Address (THERE)

INCF **Increment f**

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

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

Operation: (f) + 1 → dest

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

Encoding:

0010	10da	ffff	ffff

Description: The contents of register 'f' are incremented. If 'd' is '0', the result is placed in W. If 'd' is '1', the result is placed back in register '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 29.2.3 "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: INCF CNT, 1, 0

Before Instruction
CNT = FFh
Z = 0
C = ?
DC = ?
After Instruction
CNT = 00h
Z = 1
C = 1
DC = 1

© 2009-2011 Microchip Technology Inc.

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

© 2009-2011 Microchip Technology Inc.

31.0 ELECTRICAL CHARACTERISTICS

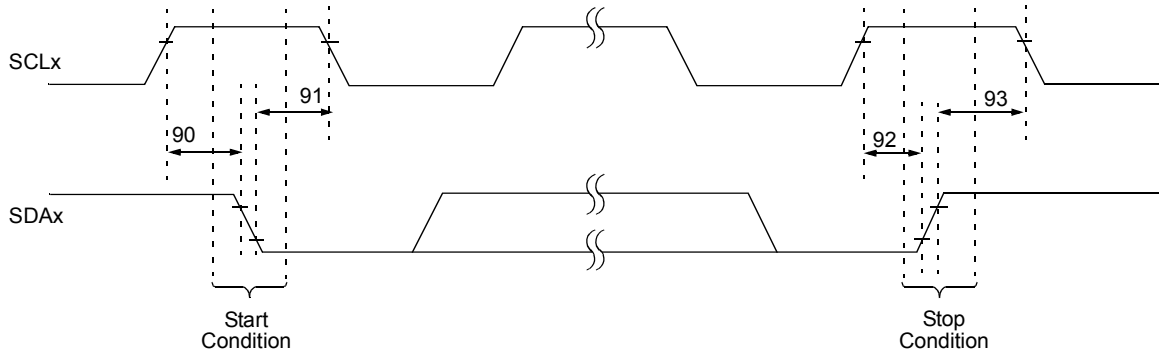
Absolute Maximum Ratings(†)

Ambient temperature under bias ..... -40°C to +125°C
Storage temperature ..... -65°C to +150°C
Voltage on any digital only I/O pin with respect to V_{SS} (except V_{DD}) ..... -0.3V to 7.5V
Voltage on $\overline{MCLR}$ with respect to V_{SS} ..... -0.3V to +9.0V
Voltage on any combined digital and analog pin with respect to V_{SS} (except V_{DD} and $\overline{MCLR}$) ..... -0.3V to (V_{DD} + 0.3V)
Voltage on V_{DD} with respect to V_{SS} (regulator enabled) ..... -0.3V to 5.5V
Voltage on V_{DD} with respect to V_{SS} (regulator disabled) ..... -0.3V to 3.6V
Total power dissipation **(Note 1)** 1W
Maximum current out of V_{SS} pin ..... 300 mA
Maximum current into V_{DD} pin ..... 250 mA
Input clamp current, I_{IK} (V_I < 0 or V_I > V_{DD}) ..... ±20 mA
Output clamp current, I_{OK} (V_O < 0 or V_O > V_{DD}) ..... ±20 mA
Maximum output current sunk by PORTA<7:6> and any PORTB and PORTC I/O pins ..... 25 mA
Maximum output current sunk by any PORTD, PORTE and PORTJ I/O pins ..... 8 mA
Maximum output current sunk by PORTA<5:0> and any PORTF, PORTG and PORTH I/O pins ..... 2 mA
Maximum output current sourced by PORTA<7:6> and any PORTB and PORTC I/O pins ..... 25 mA
Maximum output current sourced by any PORTD, PORTE and PORTJ I/O pins ..... 8 mA
Maximum output current sourced by PORTA<5:0> and any PORTF, PORTG and PORTH I/O pins ..... 2 mA
Maximum current sunk by all ports combined ..... 200 mA

Note 1: Power dissipation is calculated as follows:
$Pdis = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum(V_{OL} \times I_{OL})$

† **NOTICE:** Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

FIGURE 31-15: I²C™ BUS START/STOP BITS TIMING



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

TABLE 31-18: I²C™ BUS START/STOP BITS REQUIREMENTS (SLAVE MODE)

Param. No.	Symbol	Characteristic		Min	Max	Units	Conditions
90	TSU:STA	Start Condition Setup Time	100 kHz mode	4700	—	ns	Only relevant for Repeated Start condition
			400 kHz mode	600	—		
91	THD:STA	Start Condition Hold Time	100 kHz mode	4000	—	ns	After this period, the first clock pulse is generated
			400 kHz mode	600	—		
92	TSU:STO	Stop Condition Setup Time	100 kHz mode	4700	—	ns	
			400 kHz mode	600	—		
93	THD:STO	Stop Condition Hold Time	100 kHz mode	4000	—	ns	
			400 kHz mode	600	—		