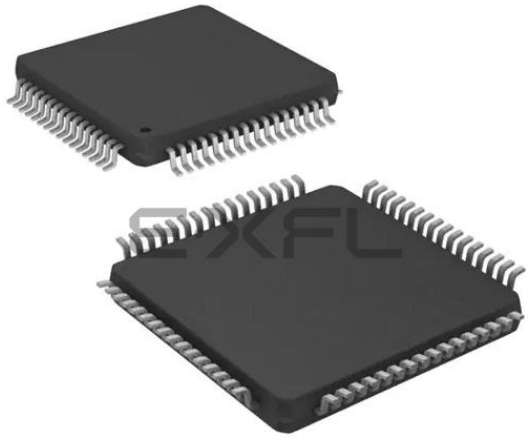


Welcome to E-XFL.COM

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	128KB (64K x 16)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	4K 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-TQFP
Supplier Device Package	64-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f67k90-e-pt

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Trademarks

ISBN: 978-1-61341-351-7

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
ISO/TS 16949:2009

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

TABLE 1-3: PIC18F6XK90 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number	Pin Type	Buffer Type	Description
	QFN/TQFP			
				PORTC is a bidirectional I/O port.
RC0/SOSCO/SCLKI	30			
RC0		I/O	ST	Digital I/O.
SOSCO		O	—	SOSC oscillator output.
SCLKI		I	ST	Digital SOSC input.
RC1/SOSCI/ECCP2/P2A/ SEG32	29			
RC1		I/O	ST	Digital I/O.
SOSCI		I	CMOS	SOSC oscillator input.
ECCP2(1)		I/O	ST	Capture 2 input/Compare 2 output/PWM2 output.
P2A		O	—	Enhanced PWM2 Output A.
SEG32		O	Analog	SEG32 output for LCD.
RC2/ECCP1/P1A/SEG13	33			
RC2		I/O	ST	Digital I/O.
ECCP1		I/O	ST	Capture 1 input/Compare 1 output/PWM1 output.
P1A		O	—	Enhanced PWM1 Output A.
SEG13		O	Analog	SEG13 output for LCD.
RC3/SCK1/SCL1/SEG17	34			
RC3		I/O	ST	Digital I/O.
SCK1		I/O	ST	Synchronous serial clock input/output for SPI mode.
SCL1		I/O	I^2C	Synchronous serial clock input/output for I^2C™ mode.
SEG17		O	Analog	SEG17 output for LCD.
RC4/SDI1/SDA1/SEG16	35			
RC4		I/O	ST	Digital I/O.
SDI1		I	ST	SPI data in.
SDA1		I/O	I^2C	I^2C data I/O.
SEG16		O	Analog	SEG16 output for LCD.
RC5/SDO1/SEG12	36			
RC5		I/O	ST	Digital I/O.
SDO1		O	—	SPI data out.
SEG12		O	Analog	SEG12 output for LCD.
RC6/TX1/CK1/SEG27	31			
RC6		I/O	ST	Digital I/O.
TX1		O	—	EUSART asynchronous transmit.
CK1		I/O	ST	EUSART synchronous clock (see related RX1/DT1).
SEG27		O	Analog	SEG27 output for LCD.
RC7/RX1/DT1/SEG28	32			
RC7		I/O	ST	Digital I/O.
RX1		I	ST	EUSART asynchronous receive.
DT1		I/O	ST	EUSART synchronous data (see related TX1/CK1).
SEG28		O	Analog	SEG28 output for LCD.

Legend: TTL = TTL compatible input; CMOS = CMOS compatible input or output; ST = Schmitt Trigger input with CMOS levels; Analog = Analog input; I = Input; O = Output; P = Power; OD = Open-Drain (no P diode to V_{DD}); I^2C™ = I^2C/SMBus

Note 1: Default assignment for ECCP2 when the CCP2MX Configuration bit is set.
2: Alternate assignment for ECCP2 when the CCP2MX Configuration bit is cleared.
3: Not available on PIC18F65K90 and PIC18F85K90 devices.

REGISTER 3-3: OSCTUNE: OSCILLATOR TUNING 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
INTSRC	PLLEN	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0
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 **INTSRC:** Internal Oscillator Low-Frequency Source Select bit
1 = 31.25 kHz device clock is derived from 16 MHz INTOSC source (divide-by-512 enabled, HF-INTOSC)
0 = 31 kHz device clock is derived from INTRC 31 kHz oscillator (LF-INTOSC)

bit 6 **PLLEN:** Frequency Multiplier PLL Enable bit
1 = PLL is enabled
0 = PLL is disabled

bit 5-0 **TUN<5:0>:** Fast RC Oscillator (INTOSC) Frequency Tuning bits
011111 = Maximum frequency
•
•
000001
000000 = Center frequency. Fast RC oscillator is running at the calibrated frequency.
111111
•
•
100000 = Minimum frequency

3.3 Clock Sources and Oscillator Switching

Essentially, PIC18F87K90 family devices have these independent clock sources:

- Primary oscillators
- Secondary oscillators
- Internal oscillator

The **primary oscillators** can be thought of as the main device oscillators. These are any external oscillators connected to the OSC1 and OSC2 pins, and include the External Crystal and Resonator modes and the External Clock modes. If selected by the OSC<3:0> Configuration bits (CONFIG1H<3:0>), the internal oscillator block may be considered a primary oscillator. The internal oscillator block can be one of the following:

- 31 kHz LF-INTRC source
- 31 kHz to 500 kHz MF-INTOSC source
- 31 kHz to 16 MHz HF-INTOSC source

The particular mode is defined by the OSC Configuration bits. The details of these modes are covered in **Section 3.4 "External Oscillator Modes"**.

The **secondary oscillators** are external clock sources that are not connected to the OSC1 or OSC2 pin. These sources may continue to operate, even after the controller is placed in a power-managed mode. PIC18F87K90 family devices offer the SOSC (Timer1/3/5/7) oscillator as a secondary oscillator source. This oscillator, in all power-managed modes, is often the time base for functions, such as a Real-Time Clock (RTC).

The SOSCEN bit in the corresponding timer should be set correctly for the enabled SOSC. The SOSCEL<1:0> bits (CONFIG1L<4:3>) decide the SOSC mode of operation:

- 11 = High-power SOSC circuit
- 10 = Digital (SCLKI) mode
- 01 = Low-power SOSC circuit

In addition to being a primary clock source in some circumstances, the **internal oscillator** is available as a power-managed mode clock source. The LF-INTOSC source is also used as the clock source for several special features, such as the WDT and Fail-Safe Clock Monitor. The internal oscillator block is discussed in more detail in **Section 3.6 "Internal Oscillator Block"**.

The PIC18F87K90 family includes features that allow the device clock source to be switched from the main oscillator, chosen by device configuration, to one of the alternate clock sources. When an alternate clock source is enabled, various power-managed operating modes are available.

4.6 Exiting Idle and Sleep Modes

An exit from Sleep mode or any of the Idle modes is triggered by an interrupt, a Reset or a WDT time-out. This section discusses the triggers that cause exits from power-managed modes. The clocking subsystem actions are discussed in each of the power-managed modes (see **Section 4.2 "Run Modes"**, **Section 4.3 "Sleep Mode"** and **Section 4.4 "Idle Modes"**).

4.6.1 EXIT BY INTERRUPT

Any of the available interrupt sources can cause the device to exit from an Idle or Sleep mode to a Run mode. To enable this functionality, an interrupt source must be enabled by setting its enable bit in one of the INTCONx or PIEx registers. The exit sequence is initiated when the corresponding interrupt flag bit is set.

On all exits from Idle or Sleep modes by interrupt, code execution branches to the interrupt vector if the GIE/GIEH bit (INTCON<7>) is set. Otherwise, code execution continues or resumes without branching (see **Section 10.0 "Interrupts"**).

4.6.2 EXIT BY WDT TIME-OUT

A WDT time-out will cause different actions depending on which power-managed mode the device is in when the time-out occurs.

If the device is not executing code (all Idle modes and Sleep mode), the time-out will result in an exit from the power-managed mode (see **Section 4.2 "Run Modes"** and **Section 4.3 "Sleep Mode"**). If the device is executing code (all Run modes), the time-out will result in a WDT Reset (see **Section 28.2 "Watchdog Timer (WDT)"**).

Executing a SLEEP or CLRWDT instruction clears the WDT timer and postscaler, loses the currently selected clock source (if the Fail-Safe Clock Monitor is enabled) and modifies the IRCF bits in the OSCCON register (if the internal oscillator block is the device clock source).

4.6.3 EXIT BY RESET

Normally, the device is held in Reset by the Oscillator Start-up Timer (OST) until the primary clock becomes ready. At that time, the OSTS bit is set and the device begins executing code. If the internal oscillator block is the new clock source, the HFIOFS/MFIOFS bits are set instead.

The exit delay time from Reset to the start of code execution depends on both the clock sources before and after the wake-up, and the type of oscillator if the new clock source is the primary clock. Exit delays are summarized in Table 4-4.

Code execution can begin before the primary clock becomes ready. If either the Two-Speed Start-up (see **Section 28.4 "Two-Speed Start-up"**) or Fail-Safe Clock Monitor (see **Section 28.5 "Fail-Safe Clock Monitor"**) is enabled, the device may begin execution as soon as the Reset source has cleared. Execution is clocked by the INTOSC multiplexer, driven by the internal oscillator block. Execution is clocked by the internal oscillator block until either the primary clock becomes ready or a power-managed mode is entered before the primary clock becomes ready; the primary clock is then shut down.

4.6.4 EXIT WITHOUT AN OSCILLATOR START-UP DELAY

Certain exits from power-managed modes do not invoke the OST at all. The two cases are:

- When in PRI_IDLE mode, where the primary clock source is not stopped
- When the primary clock source is not any of the LP, XT, HS or HSPLL modes

In these instances, the primary clock source either does not require an oscillator start-up delay, since it is already running (PRI_IDLE), or normally does not require an oscillator start-up delay (RC, EC and INTIO Oscillator modes). However, a fixed delay of interval, T_{CSD}, following the wake event, is still required when leaving Sleep and Idle modes to allow the CPU to prepare for execution. Instruction execution resumes on the first clock cycle following this delay.

TABLE 6-1: PIC18F87K90 FAMILY SPECIAL FUNCTION REGISTER MAP(5) (CONTINUED)

Addr.	Name	Addr.	Name	Addr.	Name	Addr.	Name	Addr.	Name	Addr.	Name
F3Eh	TMR7L(4)	F31h	PR12(4)	F24h	ANCON1	F17h	PMD2	F0Ah	CCPR6L	EFDh	LCDREF
F3Dh	T7CON(4)	F30h	T12CON(4)	F23h	ANCON2	F16h	PMD3	F09h	CCP6CON	EFCh	LCDRL
F3Ch	T7GCON(4)	F2Fh	CM2CON	F22h	RCSTA2	F15h	TMR5H	F08h	CCPR7H	EFBh	LCDSE5(3)
F3Bh	TMR6	F2Eh	CM3CON	F21h	TXSTA2	F14h	TMR5L	F07h	CCPR7L	EFAh	LCDSE4
F3Ah	PR6	F2Dh	CCPTMRS0	F20h	BAUDCON2	F13h	T5CON	F06h	CCP7CON	EF9h	LCDSE3
F39H	T6CON	F2Ch	CCPTMRS1	F1Fh	SPBRGH2	F12h	T5GCON	F05h	TMR4	EF8h	LCDSE2
F38h	TMR8	F2Bh	CCPTMRS2	F1Eh	SPBRG2	F11h	CCPR4H	F04h	PR4	EF7h	LCDSE1
F37h	PR8	F2Ah	REFOCON	F1Dh	RCREG2	F10h	CCPR4L	F03h	T4CON	EF6h	LCDSE0
F36h	T8CON	F29H	ODCON1	F1Ch	TXREG2	F0Fh	CCP4CON	F02h	SSP2BUF	EF5h	LCDPS
F35h	TMR10(4)	F28h	ODCON2	F1Bh	PSTR2CON	F0Eh	CCPR5H	F01h	SSP2ADD	EF4h	LCDCON
F34h	PR10(4)	F27h	ODCON3	F1Ah	PSTR3CON	F0Dh	CCPR5L	F00h	SSP2STAT		
F33h	T10CON(4)	F26h	—	F19h	PMD0	F0Ch	CCP5CON	EFFh	SSP2CON1		

Note 1: This is not a physical register.
2: Unimplemented registers are read as '0'.
3: This register is not available in 64-pin devices (PIC18F6XK90).
4: This register is not available in devices with a program memory of 32 Kbytes (PIC18FX5K90).
5: Addresses, EF4h through F5Fh, are also used by SFRs, but are not part of the Access RAM. Users must always load the proper BSR value to access these registers.

6.4.3.1 FSR Registers and the INDF Operand

At the core of Indirect Addressing are three sets of registers: FSR0, FSR1 and FSR2. Each represents a pair of 8-bit registers: FSRnH and FSRnL. The four upper bits of the FSRnH register are not used, so each FSR pair holds a 12-bit value. This represents a value that can address the entire range of the data memory in a linear fashion. The FSR register pairs, then, serve as pointers to data memory locations.

Indirect Addressing is accomplished with a set of Indirect File Operands, INDF0 through INDF2. These can be thought of as "virtual" registers. The operands are mapped in the SFR space, but are not physically implemented. Reading or writing to a particular INDF register actually accesses its corresponding FSR register pair. A read from INDF1, for example, reads the data at the address indicated by FSR1H:FSR1L.

Instructions that use the INDF registers as operands actually use the contents of their corresponding FSR as a pointer to the instruction's target. The INDF operand is just a convenient way of using the pointer.

Because Indirect Addressing uses a full 12-bit address, data RAM banking is not necessary. Thus, the current contents of the BSR and the Access RAM bit have no effect on determining the target address.

FIGURE 6-8: INDIRECT ADDRESSING

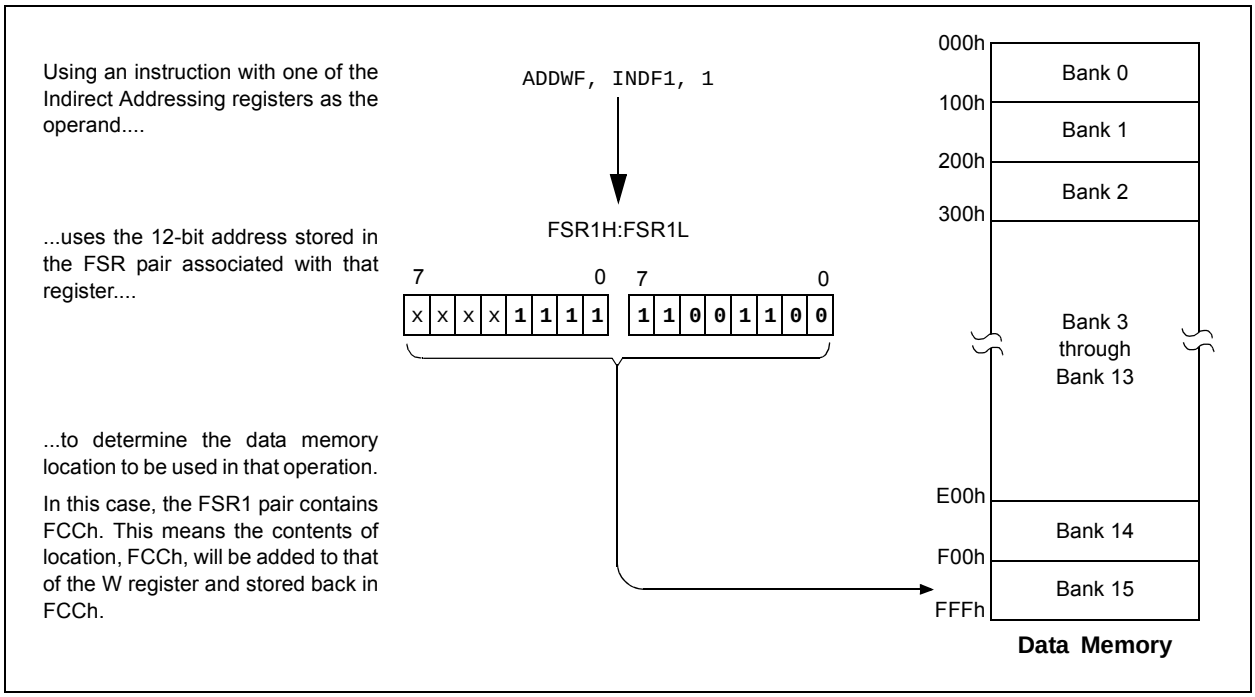
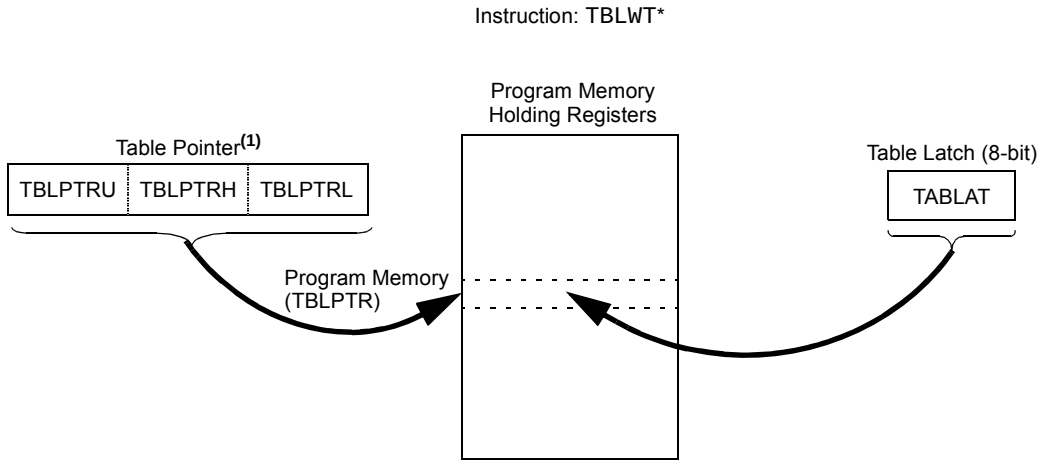


FIGURE 7-2: TABLE WRITE OPERATION



Note 1: The Table Pointer actually points to one of 64 holding registers; the address of which is determined by TBLPTRL<5:0>. The process for physically writing data to the program memory array is discussed in **Section 7.5 "Writing to Flash Program Memory"**.

7.2 Control Registers

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

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

7.2.1 EECON1 AND EECON2 REGISTERS

The EECON1 register (Register 7-1) is the control register for memory accesses. The EECON2 register, not a physical register, is used exclusively in the memory write and erase sequences. Reading EECON2 will read all '0's.

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

The CFGS control bit determines if the access is to the Configuration/Calibration registers or to program memory/data EEPROM memory. When set, subsequent operations operate on Configuration registers regardless of EEPGD (see **Section 28.0 "Special Features of the CPU"**). When clear, memory selection access is determined by EEPGD.

The FREE bit, when set, allows a program memory erase operation. When FREE is set, the erase operation is initiated on the next WR command. When FREE is clear, only writes are enabled.

The WREN bit, when set, allows a write operation. On power-up, the WREN bit is clear. The WRERR bit is set in hardware when the WR bit is set and cleared when the internal programming timer expires and the write operation is complete.

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

The WR control bit initiates write operations. The bit cannot be cleared, only set, in software. It is cleared in hardware at the completion of the write operation.

Note:	The EEIF interrupt flag bit (PIR6<4>) is set when the write is complete. It must be cleared in software.

FIGURE 13-7: TIMER1 GATE SINGLE PULSE AND TOGGLE COMBINED MODE

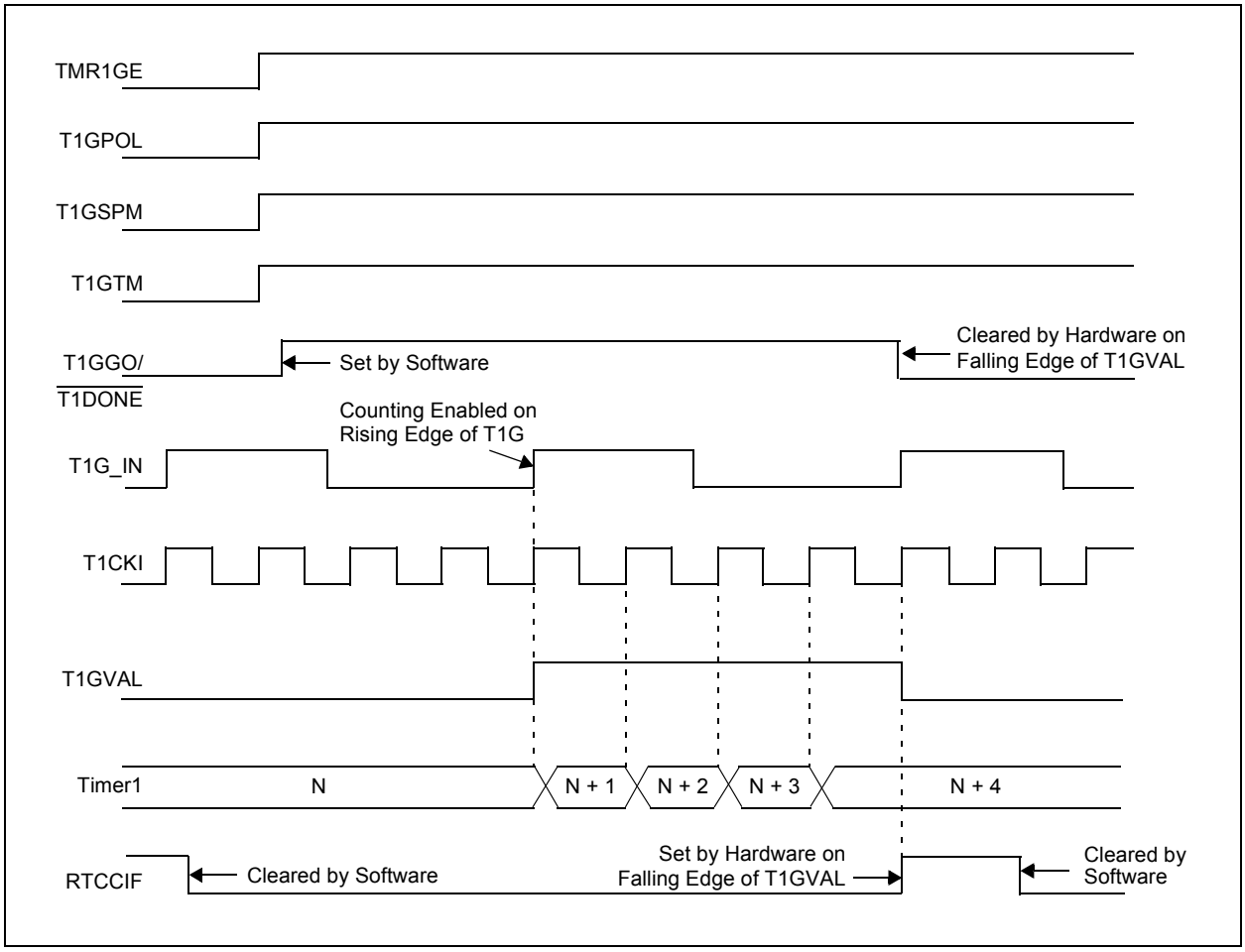


TABLE 13-5: REGISTERS ASSOCIATED WITH TIMER1 AS A TIMER/COUNTER

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
PIR1	—	ADIF	RC1IF	TX1IF	SSP1IF	TMR1GIF	TMR2IF	TMR1IF	77
PIE1	—	ADIE	RC1IE	TX1IE	SSP1IE	TMR1GIE	TMR2IE	TMR1IE	77
IPR1	—	ADIP	RC1IP	TX1IP	SSP1IP	TMR1GIP	TMR2IP	TMR1IP	77
TMR1L	Timer1 Register Low Byte								76
TMR1H	Timer1 Register High Byte								76
T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	SOSCEN	$\overline{\text{T1SYNC}}$	RD16	TMR1ON	76
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	$\overline{\text{T1GGO}}$/T1DONE	T1GVAL	T1GSS1	T1GSS0	77
OSCCON2	—	SOSCRUN	—	—	SOSCGO	—	MFIOFS	MFIOSEL	79
CCPTMRS0	C3TSEL1	C3TSEL0	C2TSEL2	C2TSEL1	C2TSEL0	C1TSEL2	C1TSEL1	C1TSEL0	81
CCPTMRS1	C7TSEL1	C7TSEL0	—	C6TSEL0	—	C5TSEL0	C4TSEL1	C4TSEL0	81
CCPTMRS2	—	—	—	C10TSEL0	—	C9TSEL0	C8TSEL1	C8TSEL0	81

Legend: Shaded cells are not used by the Timer1 module.

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.

18.0 CAPTURE/COMPARE/PWM (CCP) MODULES

PIC18F87K90 family devices have seven CCP (Capture/Compare/PWM) modules, designated CCP4 through CCP10. All the modules implement standard Capture, Compare and Pulse-Width Modulation (PWM) modes.

Note:	Throughout this section, generic references are used for register and bit names that are the same, except for an 'x' variable that indicates the item's association with the specific CCP module. For example, the control register is named CCPxCON and refers to CCP4CON through CCP10CON.

Each CCP module contains a 16-bit register that can operate as a 16-bit Capture register, a 16-bit Compare register or a PWM Master/Slave Duty Cycle register. For the sake of clarity, all CCP module operation in the following sections is described with respect to CCP4, but is equally applicable to CCP5 through CCP10.

Note:	The CCP9 and CCP10 modules are disabled on the devices with 32 Kbytes of program memory (PIC18FX5K90).

REGISTER 18-1: CCPxCON: CCPx CONTROL REGISTER (CCP4-CCP10 MODULES)(1)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	DCxB1	DCxB0	CCPxM3	CCPxM2	CCPxM1	CCPxM0
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 **Unimplemented:** Read as '0'

bit 5-4 **DCxB<1:0>**: PWM Duty Cycle for CCPx Module bits (bit 1, bit 0)

Capture mode:
Unused.

Compare mode:
Unused.

PWM mode:
These bits are the two Least Significant bits (bit 1 and bit 0) of the 10-bit PWM duty cycle. The eight Most Significant bits (DCxB<9:2>) of the duty cycle are found in CCPRxL.

bit 3-0 **CCPxM<3:0>**: CCPx Module Mode Select bits

0000 = Capture/Compare/PWM disabled (resets CCPx module)
0001 = Reserved
0010 = Compare mode: toggle output on match (CCPxIF bit is set)
0011 = Reserved
0100 = Capture mode: every falling edge
0101 = Capture mode: every rising edge
0110 = Capture mode: every 4th rising edge
0111 = Capture mode: every 16th rising edge
1000 = Compare mode: initialize CCPx pin low; on compare match, force CCPx pin high (CCPxIF bit is set)
1001 = Compare mode: initialize CCPx pin high; on compare match, force CCPx pin low (CCPxIF bit is set)
1010 = Compare mode: generate software interrupt on compare match (CCPxIF bit is set, CCPx pin reflects I/O state)
1011 = Compare mode: Special Event Trigger; reset timer on CCPx match (CCPxIF bit is set)(2)
11xx = PWM mode

Note 1: The CCP9 and CCP10 modules are not available on devices with 32 Kbytes of program memory (PIC18FX5K90).

2: CCPxM<3:0> = 1011 will only reset the timer and not start the A/D conversion on a CCPx match.

There are two power modes designated as “Mode A” and “Mode B”. Mode A is set by the bits, LRLAP<1:0> and Mode B by LRLB<1:0>. The resistor ladder to use for Modes A and B are selected by the bits, LRLAP<1:0> and LRLBP<1:0>, respectively

Each ladder has a matching contrast control ladder, tuned to the nominal resistance of the reference ladder. This contrast control resistor can be controlled by LCDREF<5:3> (LCDCST<2:0>). Disabling the internal reference ladder results in all of the ladders being disconnected, allowing external voltages to be supplied.

To get additional current in High-Power mode, when LCDRL<7:6> (LRLAP<1:0>) = 11, both the medium and high-power resistor ladders are activated.

Whenever the LCD module is inactive (LCDA (LCDPS<5>) = 0), the reference ladder will be turned off.

20.3.2.1 Automatic Power Mode Switching

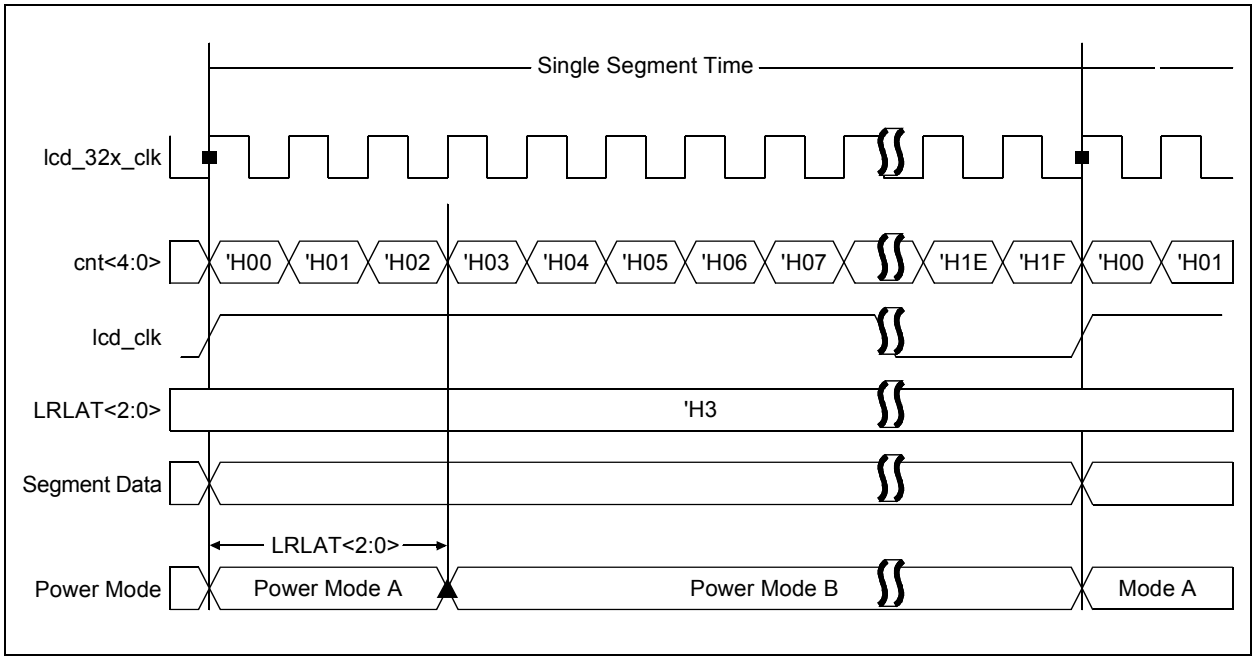
As an LCD segment is electrically only a capacitor, current is drawn only during the interval when the voltage is switching. To minimize total device current, the LCD reference ladder can be operated in a different power mode for the transition portion of the duration. This is controlled by the LCDRL register.

Mode A Power mode is active for a programmable time, beginning at the time when the LCD segment waveform is transitioning. The LCDRL<2:1> (LRLAT<2:0>) bits select how long, or if the Mode A is active. Mode B Power mode is active for the remaining time before the segments or commons change again.

As shown in Figure 20-5, there are 32 counts in a single segment time. Type-A can be chosen during the time when the wave form is in transition. Type-B can be used when the clock is stable or not in transition.

By using this feature of automatic power switching, using Type-A/Type-B, the power consumption can be optimized for a given contrast.

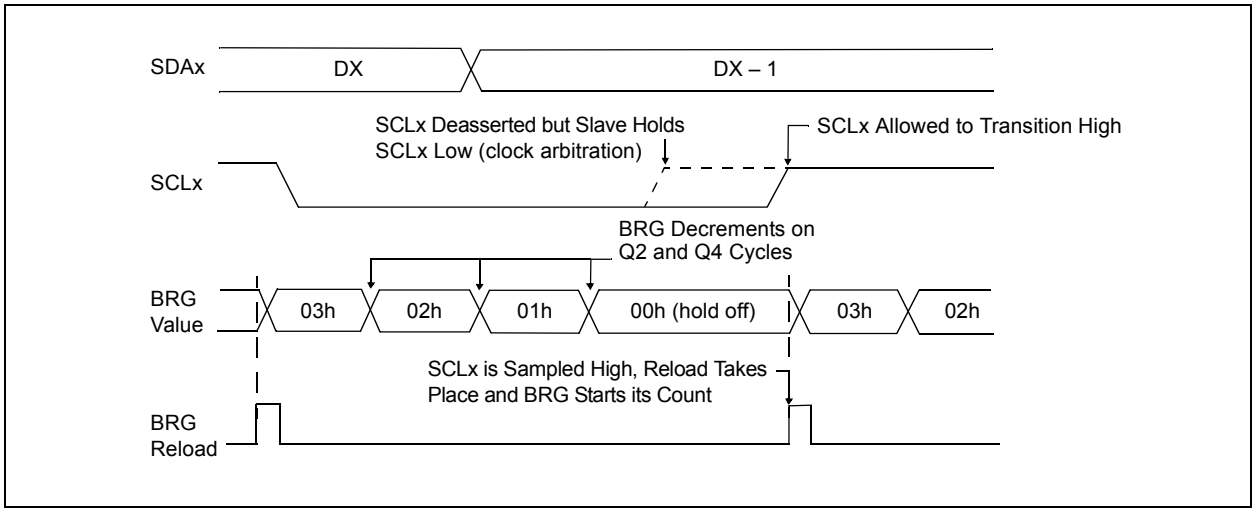
FIGURE 20-5: LCD REFERENCE LADDER POWER MODE SWITCHING DIAGRAM



21.4.7.2 Clock Arbitration

Clock arbitration occurs when the master, during any receive, transmit or Repeated Start/Stop condition, deasserts the SCLx pin (SCLx allowed to float high). When the SCLx pin is allowed to float high, the Baud Rate Generator (BRG) is suspended from counting until the SCLx pin is actually sampled high. When the SCLx pin is sampled high, the Baud Rate Generator is reloaded with the contents of SSPxADD<6:0> and begins counting. This ensures that the SCLx high time will always be at least one BRG rollover count in the event that the clock is held low by an external device (Figure 21-20).

FIGURE 21-20: BAUD RATE GENERATOR TIMING WITH CLOCK ARBITRATION



21.4.8 I²C™ MASTER MODE START CONDITION TIMING

To initiate a Start condition, the user sets the Start Enable bit, SEN (SSPxCON2<0>). If the SDAx and SCLx pins are sampled high, the Baud Rate Generator is reloaded with the contents of SSPxADD<6:0> and starts its count. If SCLx and SDAx are both sampled high when the Baud Rate Generator times out (TBRG), the SDAx pin is driven low. The action of the SDAx being driven low while SCLx is high is the Start condition and causes the S bit (SSPxSTAT<3>) to be set. Following this, the Baud Rate Generator is reloaded with the contents of SSPxADD<6:0> and resumes its count. When the Baud Rate Generator times out (TBRG), the SEN bit (SSPxCON2<0>) will be automatically cleared by hardware. The Baud Rate Generator is suspended, leaving the SDAx line held low and the Start condition is complete.

Note:	If, at the beginning of the Start condition, the SDAx and SCLx pins are already sampled low, or if during the Start condition, the SCLx line is sampled low before the SDAx line is driven low, a bus collision occurs, the Bus Collision Interrupt Flag, BCLxIF, is set, the Start condition is aborted and the I²C module is reset into its Idle state.

21.4.8.1 WCOL Status Flag

If the user writes the SSPxBUF when a Start sequence is in progress, the WCOL bit is set and the contents of the buffer are unchanged (the write doesn't occur).

Note:	Because queueing of events is not allowed, writing to the lower 5 bits of SSPxCON2 is disabled until the Start condition is complete.

FIGURE 21-21: FIRST START BIT TIMING

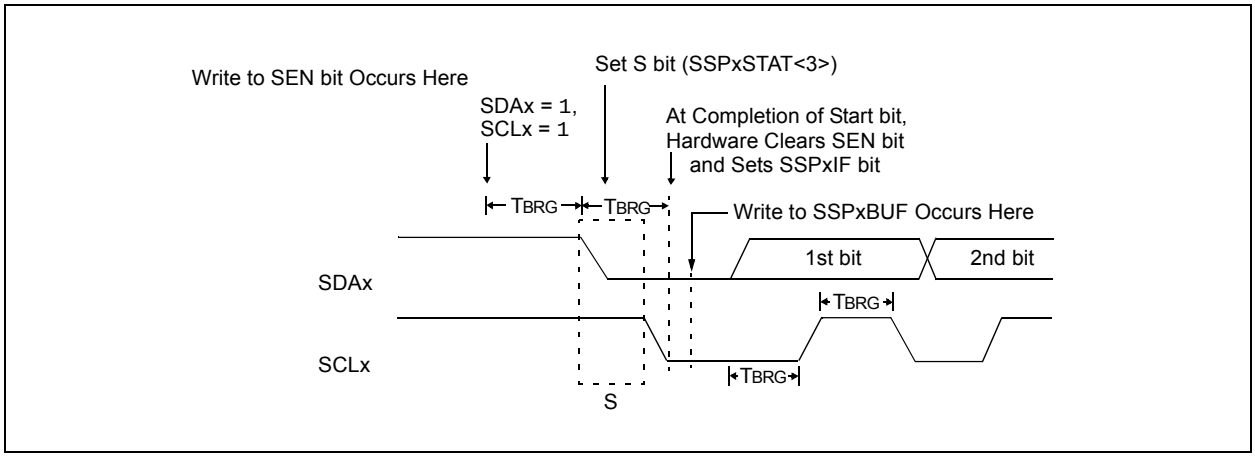


TABLE 22-5: REGISTERS ASSOCIATED WITH ASYNCHRONOUS TRANSMISSION

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
PIR1	—	ADIF	RC1IF	TX1IF	SSP1IF	TMR1GIF	TMR2IF	TMR1IF	77
PIE1	—	ADIE	RC1IE	TX1IE	SSP1IE	TMR1GIE	TMR2IE	TMR1IE	77
IPR1	—	ADIP	RC1IP	TX1IP	SSP1IP	TMR1GIP	TMR2IP	TMR1IP	77
PIR3	TMR5GIF	LCDIF	RC2IF	TX2IF	CTMUIF	CCP2IF	CCP1IF	RTCCIF	77
PIE3	TMR5GIE	LCDIE	RC2IE	TX2IE	CTMUIE	CCP2IE	CCP1IE	RTCCIE	77
IPR3	TMR5GIP	LCDIP	RC2IP	TX2IP	CTMUIP	CCP2IP	CCP1IP	RTCCIP	77
RCSTA1	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	77
TXREG1	EUSART1 Transmit Register								77
TXSTA1	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	77
BAUDCON1	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN	79
SPBRGH1	EUSART1 Baud Rate Generator Register High Byte								76
SPBRG1	EUSART1 Baud Rate Generator Register Low Byte								77
RCSTA2	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	81
TXREG2	EUSART2 Transmit Register								82
TXSTA2	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	81
BAUDCON2	ABDOVF	RCIDL	RXDTP	TXCKP	BRG16	—	WUE	ABDEN	81
SPBRGH2	EUSART2 Baud Rate Generator Register High Byte								82
SPBRG2	EUSART2 Baud Rate Generator Register Low Byte								82

Legend: — = unimplemented locations read as '0'. Shaded cells are not used for asynchronous transmission.

27.4.2 CAPACITANCE CALIBRATION

There is a small amount of capacitance from the internal A/D Converter sample capacitor, as well as stray capacitance from the circuit board traces and pads that affect the precision of capacitance measurements. A measurement of the stray capacitance can be taken by making sure the desired capacitance to be measured has been removed.

After removing the capacitance to be measured:

1. Initialize the A/D Converter and the CTMU.
2. Set EDG1STAT (= 1).
3. Wait for a fixed delay of time, *t*.
4. Clear EDG1STAT.
5. Perform an A/D conversion.
6. Calculate the stray and A/D sample capacitances:

$$C_{OFFSET} = C_{STRAY} + C_{AD} = (I \bullet t)/V$$

Where:

- I is known from the current source measurement step
- t is a fixed delay
- V is measured by performing an A/D conversion

This measured value is then stored and used for calculations of time measurement or subtracted for capacitance measurement. For calibration, it is expected that the capacitance of C_{STRAY} + C_{AD} is approximately known; C_{AD} is approximately 4 pF.

An iterative process may be required to adjust the time, *t*, that the circuit is charged to obtain a reasonable voltage reading from the A/D Converter. The value of *t* may be determined by setting C_{OFFSET} to a theoretical value and solving for *t*. For example, if C_{STRAY} is theoretically calculated to be 11 pF, and V is expected to be 70% of V_{DD} or 2.31V, t would be:

$$(4\ pF + 11\ pF) \bullet 2.31V/0.55\ \mu A$$

or 63 μs.

See Example 27-3 for a typical routine for CTMU capacitance calibration.

REGISTER 28-5: CONFIG3L: CONFIGURATION REGISTER 3 LOW (BYTE ADDRESS 300004h)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/P-1
—	—	—	—	—	—	—	RTCOSC
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 7-1 **Unimplemented:** Read as ‘0’

bit 0 **RTCOSC:** RTCC Reference Clock Select bit
1 = RTCC uses SOSC as a reference clock
0 = RTCC uses LF-INTOSC as a reference clock

REGISTER 28-6: CONFIG3H: CONFIGURATION REGISTER 3 HIGH (BYTE ADDRESS 300005h)

R/P-1	U-0	U-0	U-0	R/P-1	U-0	R/P-1	R/P-1
MCLRE	—	—	—	MSSPMSK	—	ECCPMX(1)	CCP2MX
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 **MCLRE:** $\overline{\text{MCLR}}$ Pin Enable bit
1 = $\overline{\text{MCLR}}$ pin is enabled; RG5 input pin is disabled
0 = RG5 input pin is enabled; $\overline{\text{MCLR}}$ is disabled

bit 6-4 **Unimplemented:** Read as ‘0’

bit 3 **MSSPMSK:** MSSP V3 7-Bit Address Masking Mode Enable bit
1 = 7-Bit Address Masking mode is enabled
0 = 5-Bit Address Masking mode is enabled

bit 2 **Unimplemented:** Read as ‘0’

bit 1 **ECCPMX:** ECCP MUX bit(1)
1 = Enhanced ECCP1 (P1B/P1C) is multiplexed onto RE6 and RE5, CCP6 onto RE6 and CCP7 onto RE5
Enhanced ECCP3 (P3B/P3C) is multiplexed onto RE4 and RE3, CCP8 onto RE4 and CCP9 onto RE3
0 = Enhanced ECCP1 (P1B/P1C) is multiplexed onto RH7 and RH6, CCP6 onto RH7 and CCP7 onto RH6
Enhanced ECCP3 (P3B/P3C) is multiplexed onto RH5 and RH4, CCP8 onto RH5 and CCP9 onto RH4

bit 0 **CCP2MX:** ECCP2 MUX bit
1 = ECCP2 is multiplexed with RC1
0 = ECCP2 input/output is multiplexed with RE7(1)

Note 1: This feature is only available on 80-pin devices.

REGISTER 28-10: CONFIG6L: CONFIGURATION REGISTER 6 LOW (BYTE ADDRESS 30000Ah)(2)

R/C-1	R/C-1	R/C-1	R/C-1	R/C-1	R/C-1	R/C-1	R/C-1
WRT7(1)	WRT6(1)	WRT5(1)	WRT4(1)	WRT3	WRT2	WRT1	WRT0
bit 7							bit 0

Legend:	C = Clearable 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 **WRT7:** Write Protection bit(1)
1 = Block 7 is not write-protected
0 = Block 7 is write-protected

bit 6 **WRT6:** Write Protection bit(1)
1 = Block 6 is not write-protected
0 = Block 6 is write-protected

bit 5 **WRT5:** Write Protection bit(1)
1 = Block 5 is not write-protected
0 = Block 5 is write-protected

bit 4 **WRT4:** Write Protection bit(1)
1 = Block 4 is not write-protected
0 = Block 4 is write-protected

bit 3 **WRT3:** Write Protection bit
1 = Block 3 is not write-protected
0 = Block 3 is write-protected

bit 2 **WRT2:** Write Protection bit
1 = Block 2 is not write-protected
0 = Block 2 is write-protected

bit 1 **WRT1:** Write Protection bit
1 = Block 1 is not write-protected
0 = Block 1 is write-protected

bit 0 **WRT0:** Write Protection bit
1 = Block 0 is not write-protected
0 = Block 0 is write-protected

Note 1: This bit is only available on PIC18F67K90 and PIC18F87K90.
2: For the memory size of the blocks, refer to Figure 28-6.

SLEEP **Enter Sleep Mode**

Syntax: SLEEP

Operands: None

Operation: 00h → WDT,
0 → WDT postscaler,
1 → $\overline{TO}$,
0 → $\overline{PD}$

Status Affected: $\overline{TO}$, $\overline{PD}$

Encoding:

0000	0000	0000	0011

Description: The Power-Down status bit ($\overline{PD}$) is cleared. The Time-out status bit ($\overline{TO}$) is set. The Watchdog Timer and its postscaler are cleared.

The processor is put into Sleep mode with the oscillator stopped.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	No operation	Process Data	Go to Sleep

Example: SLEEP

Before Instruction
$\overline{TO}$ = ?
$\overline{PD}$ = ?
After Instruction
$\overline{TO}$ = 1 †
$\overline{PD}$ = 0

† If WDT causes wake-up, this bit is cleared.

SUBFWB **Subtract f from W with Borrow**

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

Operands: 0 ≤ f ≤ 255
d ∈ [0,1]
a ∈ [0,1]

Operation: (W) – (f) – ($\overline{C}$) → dest

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

Encoding:

0101	01da	ffff	ffff

Description: Subtract register 'f' and Carry flag (borrow) from W (2's complement method). If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored 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 ≤ 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 1: SUBFWB REG, 1, 0

Before Instruction
REG = 3
W = 2
C = 1
After Instruction
REG = FF
W = 2
C = 0
Z = 0
N = 1 ; result is negative

Example 2: SUBFWB REG, 0, 0

Before Instruction
REG = 2
W = 5
C = 1
After Instruction
REG = 2
W = 3
C = 1
Z = 0
N = 0 ; result is positive

Example 3: SUBFWB REG, 1, 0

Before Instruction
REG = 1
W = 2
C = 0
After Instruction
REG = 0
W = 2
C = 1
Z = 1 ; result is zero
N = 0

MOVSS **Move Indexed to Indexed**

Syntax: MOVSS [z_s], [z_d]

Operands: $0 \le z_s \le 127$
$0 \le z_d \le 127$

Operation: ((FSR2) + z_s) → ((FSR2) + z_d)

Status Affected: None

Encoding:

1st word (source)	1110	1011	1zzz	$zzzz_s$
2nd word (dest.)	1111	xxxx	xzzz	$zzzz_d$

Description: The contents of the source register are moved to the destination register. The addresses of the source and destination registers are determined by adding the 7-bit literal offsets, 'z_s' or 'z_d', respectively, to the value of FSR2. Both registers can be located anywhere in the 4096-byte data memory space (000h to FFFh).

The MOVSS instruction cannot use the PCL, TOSU, TOSH or TOSL as the destination register.

If the resultant source address points to an Indirect Addressing register, the value returned will be 00h. If the resultant destination address points to an Indirect Addressing register, the instruction will execute as a NOP.

Words: 2

Cycles: 2

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Determine source addr	Determine source addr	Read source reg
Decode	Determine dest addr	Determine dest addr	Write to dest reg

Example: MOVSS [05h], [06h]

Before Instruction
FSR2 = 80h
Contents of 85h = 33h
Contents of 86h = 11h
After Instruction
FSR2 = 80h
Contents of 85h = 33h
Contents of 86h = 33h

PUSHL **Store Literal at FSR2, Decrement FSR2**

Syntax: PUSHL k

Operands: $0 \le k \le 255$

Operation: k → (FSR2),
FSR2 – 1 → FSR2

Status Affected: None

Encoding:

1110	1010	kkkk	kkkk

Description: The 8-bit literal 'k' is written to the data memory address specified by FSR2. FSR2 is decremented by 1 after the operation.

This instruction allows users to push values onto a software stack.

Words: 1

Cycles: 1

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read 'k'	Process data	Write to destination

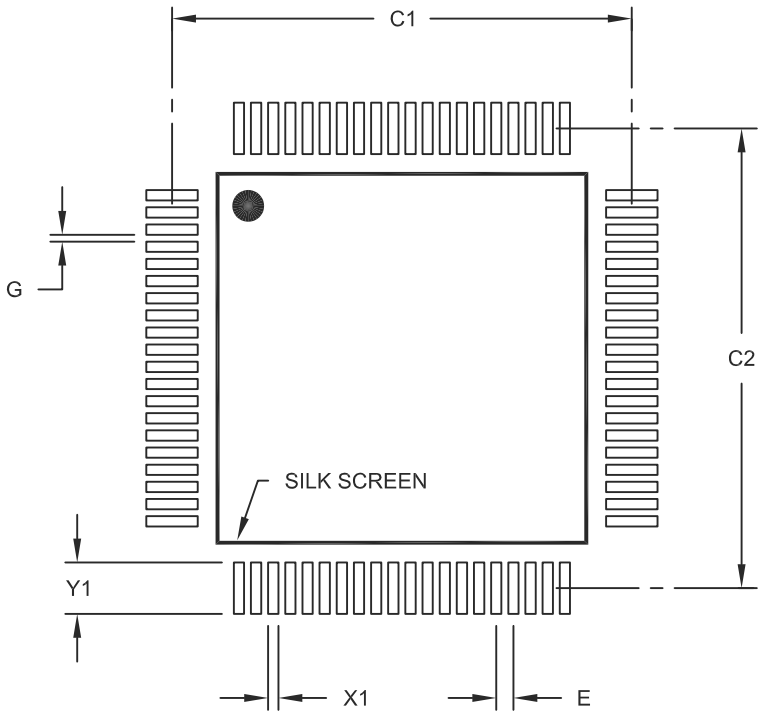
Example: PUSHL 08h

Before Instruction
FSR2H:FSR2L = 01ECh
Memory (01ECh) = 00h

After Instruction
FSR2H:FSR2L = 01EBh
Memory (01ECh) = 08h

80-Lead Plastic Thin Quad Flatpack (PT) – 12x12x1 mm Body, 2.00 mm [TQFP]

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		0.50 BSC	
Contact Pad Spacing	C1		13.40	
Contact Pad Spacing	C2		13.40	
Contact Pad Width (X80)	X1			0.30
Contact Pad Length (X80)	Y1			1.50
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

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

Microchip Technology Drawing No. C04-2092A