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	Obsolete
Core Processor	PIC
Core Size	8-Bit
Speed	25MHz
Connectivity	EBI/EMI, I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	26
Program Memory Size	-
Program Memory Type	ROMless
EEPROM Size	-
RAM Size	1.5K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	68-LCC (J-Lead)
Supplier Device Package	68-PLCC (24.23x24.23)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18lc601-i-l

TO OUR VALUED CUSTOMERS

It is our intention to provide our valued customers with the best documentation possible to ensure successful use of your Microchip products. To this end, we will continue to improve our publications to better suit your needs. Our publications will be refined and enhanced as new volumes and updates are introduced.

If you have any questions or comments regarding this publication, please contact the Marketing Communications Department via E-mail at **docerrors@mail.microchip.com** or fax the **Reader Response Form** in the back of this data sheet to (480) 792-4150. We welcome your feedback.

Most Current Data Sheet

To obtain the most up-to-date version of this data sheet, please register at our Worldwide Web site at:

http://www.microchip.com

You can determine the version of a data sheet by examining its literature number found on the bottom outside corner of any page. The last character of the literature number is the version number, (e.g., DS30000A is version A of document DS30000).

Errata

An errata sheet, describing minor operational differences from the data sheet and recommended workarounds, may exist for current devices. As device/documentation issues become known to us, we will publish an errata sheet. The errata will specify the revision of silicon and revision of document to which it applies.

To determine if an errata sheet exists for a particular device, please check with one of the following:

- Microchip's Worldwide Web site; http://www.microchip.com
- Your local Microchip sales office (see last page)
- The Microchip Corporate Literature Center; U.S. FAX: (480) 792-7277

When contacting a sales office or the literature center, please specify which device, revision of silicon and data sheet (include literature number) you are using.

Customer Notification System

Register on our web site at **www.microchip.com/cn** to receive the most current information on all of our products.

TABLE 1-2: PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number				Pin Type	Buffer Type	Description
	PIC18C601		PIC18C801				
	TQFP	PLCC	TQFP	PLCC			
							PORTA is a bi-directional I/O port.
RA0/AN0	24	34	30	42			
RA0					I/O	TTL	Digital I/O.
AN0					I	Analog	Analog input 0.
RA1/AN1	23	33	29	41			
RA1					I/O	TTL	Digital I/O.
AN1					I	Analog	Analog input 1.
RA2/AN2/VREF-	22	32	28	40			
RA2					I/O	TTL	Digital I/O.
AN2					I	Analog	Analog input 2.
VREF-					I	Analog	A/D reference voltage (Low) input.
RA3/AN3/VREF+	21	31	27	39			
RA3					I/O	TTL	Digital I/O.
AN3					I	Analog	Analog input 3.
VREF+					I	Analog	A/D reference voltage (High) input.
RA4/T0CKI	28	39	34	47			
RA4					I/O	ST/OD	Digital I/O – Open drain when configured as output.
T0CKI					I	ST	Timer0 external clock input.
RA5/AN4/$\overline{SS}$/LVDIN	27	38	33	46			
RA5					I/O	TTL	Digital I/O.
AN4					I	Analog	Analog input 4.
$\overline{SS}$					I	ST	SPI slave select input.
LVDIN					I	Analog	Low voltage detect input.

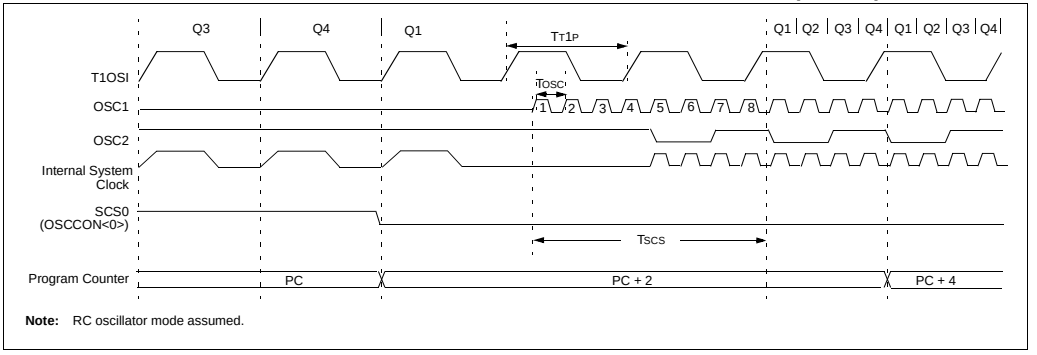
Legend: TTL = TTL compatible input
ST = Schmitt Trigger input with CMOS levels
I = Input
P = Power
CMOS = CMOS compatible input or output
Analog = Analog input
O = Output
OD = Open Drain (no P diode to VDD)

TABLE 1-2: PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number				Pin Type	Buffer Type	Description
	PIC18C601		PIC18C801				
	TQFP	PLCC	TQFP	PLCC			
							PORTD is a bi-directional I/O port. These pins have TTL input buffers when external memory is enabled.
RD0/AD0	58	3	72	3			
RD0					I/O	ST	Digital I/O.
AD0					I/O	TTL	External memory address/data 0.
RD1/AD1	55	67	69	83			
RD1					I/O	ST	Digital I/O.
AD1					I/O	TTL	External memory address/data 1.
RD2/AD2	54	66	68	82			
RD2					I/O	ST	Digital I/O.
AD2					I/O	TTL	External memory address/data 2.
RD3/AD3	53	65	67	81			
RD3					I/O	ST	Digital I/O.
AD3					I/O	TTL	External memory address/data 3.
RD4/AD4	52	64	66	80			
RD4					I/O	ST	Digital I/O.
AD4					I/O	TTL	External memory address/data 4.
RD5/AD5	51	63	65	79			
RD5					I/O	ST	Digital I/O.
AD5					I/O	TTL	External memory address/data 5.
RD6/AD6	50	62	64	78			
RD6					I/O	ST	Digital I/O.
AD6					I/O	TTL	External memory address/data 6.
RD7/AD7	49	61	63	77			
RD7					I/O	ST	Digital I/O.
AD7					I/O	TTL	External memory address/data 7.

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

FIGURE 2-10: TIMING FOR TRANSITION BETWEEN TIMER1 AND OSC1 (RC, EC)



Note: RC oscillator mode assumed.

2.6.3 SCS0, SCS1 PRIORITY

If both SCS0 and SCS1 are set to '1' simultaneously, the SCS0 bit has priority over the SCS1 bit. This means that the low power option will take precedence over the PLL option. If both bits are cleared simultaneously, the system clock will come from OSC1, after a TOST time-out. If only the SCS0 bit is cleared, the system clock will come from the PLL output, following TOST and TPLL time.

TABLE 2-3: SCS0, SCS1 PRIORITY

SCS1	SCS0	Clock Source
0	0	Ext Oscillator OSC1
0	1	Timer1 Oscillator
1	0	HS + PLL
1	1	Timer1 Oscillator

2.7 Effects of SLEEP Mode on the On-Chip Oscillator

When the device executes a SLEEP instruction, the on-chip clocks and oscillator are turned off and the device is held at the beginning of an instruction cycle (Q1 state). With the oscillator off, the OSC1 and OSC2 signals will stop oscillating. Since all the transistor switching currents have been removed, SLEEP mode achieves the lowest current consumption of the device (only leakage currents). Enabling any on-chip feature that will operate during SLEEP, will increase the current consumed during SLEEP. The user can wake from SLEEP through external RESET, Watchdog Timer Reset, or through an interrupt.

2.8 Power-up Delays

Power-up delays are controlled by two timers, so that no external RESET circuitry is required for most applications. The delays ensure that the device is kept in RESET until the device power supply and clock are stable. For additional information on RESET operation, see Section 3.0 RESET.

The first timer is the Power-up Timer (PWRT), which optionally provides a fixed delay of TPWRT (parameter #33) on power-up only. The second timer is the Oscillator Start-up Timer (OST), intended to keep the chip in RESET until the crystal oscillator is stable.

PIC18C601/801 devices provide a configuration bit, PWRTEN in CONFIG2L register, to enable or disable the Power-up Timer. By default, the Power-up Timer is enabled.

With the PLL enabled (HS4 oscillator mode), the time-out sequence following a Power-on Reset is different from other oscillator modes. The time-out sequence is as follows: the PWRT time-out is invoked after a POR time delay has expired, then, the Oscillator Start-up Timer (OST) is invoked. However, this is still not a sufficient amount of time to allow the PLL to lock at high frequencies. The PWRT timer is used to provide an additional time-out, called TPLL (parameter #7), to allow the PLL ample time to lock to the incoming clock frequency.

TABLE 2-4: OSC1 AND OSC2 PIN STATES IN SLEEP MODE

OSC Mode	OSC1 Pin	OSC2 Pin
RC	Floating, external resistor should pull high	At logic low
EC	Floating	At logic low
LP and HS	Feedback inverter disabled, at quiescent voltage level	Feedback inverter disabled, at quiescent voltage level

Note: See Table 3-1 in Section 3.0 RESET, for time-outs due to SLEEP and MCLR Reset.

4.9 Data Memory Organization

The data memory is implemented as static RAM. Each register in the data memory has a 12-bit address, allowing up to 4096 bytes of data memory. Figure 4-8 shows the data memory organization for PIC18C601/801 devices.

The data memory map is divided into banks that contain 256 bytes each. The lower four bits of the Bank Select Register (BSR<3:0>) select which bank will be accessed. The upper 4 bits for the BSR are not implemented.

The data memory contains Special Function Registers (SFR) and General Purpose Registers (GPR). The SFR's are used for control and status of the controller and peripheral functions, while GPR's are used for data storage and scratch pad operations in the user's application. The SFR's start at the last location of Bank 15 (0FFFh) and grow downwards. GPR's start at the first location of Bank 0 and grow upwards. Any read of an unimplemented location will read as '0's.

GPR banks 4 and 5 serve as a Program Memory called "Boot RAM", when PGRM bit in MEMCON is set. When PGRM bit is set, any read from "Boot RAM" returns '0's, while any write to it is ignored.

The entire data memory may be accessed directly or indirectly. Direct addressing may require the use of the BSR register. Indirect addressing requires the use of a File Select Register (FSR). Each FSR holds a 12-bit address value that can be used to access any location in the Data Memory map without banking.

The instruction set and architecture allow operations across all banks. This may be accomplished by indirect addressing, or by the use of the `MOVFF` instruction. The `MOVFF` instruction is a two-word/two-cycle instruction that moves a value from one register to another.

To ensure that commonly used registers (SFRs and select GPRs) can be accessed in a single cycle, regardless of the current BSR values, an Access Bank is implemented. A segment of Bank 0 and a segment of Bank 15 comprise the Access bank. Section 4.10 provides a detailed description of the Access bank.

4.9.1 GENERAL PURPOSE REGISTER FILE

The register file can be accessed either directly or indirectly. Indirect addressing operates through the File Select Registers (FSR). The operation of indirect addressing is shown in Section 4.12.

PIC18C601/801 devices have banked memory in the GPR area. GPRs are not initialized by a Power-on Reset and are unchanged on all other RESETS.

Data RAM is available for use as GPR registers by all instructions. Bank 15 (0F80h to 0FFFh) contains SFR's. All other banks of data memory contain GPR registers starting with bank 0.

4.9.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers (SFRs) are registers used by the CPU and Peripheral Modules for controlling the desired operation of the device. These registers are implemented as static RAM. A list of these registers is given in Table 4-2.

The SFR's can be classified into two sets: those associated with the "core" function and those related to the peripheral functions. Those registers related to the "core" are described in this section, while those related to the operation of the peripheral features are described in the section of that peripheral feature.

The SFRs are typically distributed among the peripherals whose functions they control.

The unused SFR locations are unimplemented and read as '0's. See Table 4-2 for addresses for the SFRs.

4.9.3 SECURED ACCESS REGISTERS

PIC18C601/801 devices contain software programming options for safety critical peripherals. Because these safety critical peripherals can be programmed in software, registers used to control these peripherals are given limited access by the user code. This way, errant code will not accidentally change settings in peripherals that could cause catastrophic results.

The registers that are considered safety critical are the Watchdog Timer register (WDTCON), the External Memory Control register (MEMCON), the Oscillator Control register (OSCCON) and the Chip Select registers (CSSEL2 and CSELIO).

Two bits called Combination Lock (CMLK) bits, located in the lower two bits of the PSPCON register, must be set in sequence by user code to gain access to Secured Access registers.

6.3 Table Write

Table Write operations store data from the data memory space into external program memory.

PIC18C601/801devices perform Table Writes one byte at a time. Table Writes to external memory are two-cycle instructions, unless wait states are enabled. The last cycle writes the data to the external memory location.

16-bit interface Table Writes depend on the type of external device that is connected and the WM<1:0> bits in the MEMCON register (See Figure 5-2).

Example 6-2 describes how to use TBLWT.

EXAMPLE 6-2: TABLE WRITE CODE EXAMPLE

```
; Write a byte to location 0020h
CLRF    TBLPTRU            ; clear upper 5 bits of TBLPTR
CLRF    TBLPTRH            ; clear higher 8 bits of TBLPTR
MOVLW   20h                ; Load 20h into
MOVWF   TBLPTRL            ; TBLPTRL
MOVLW   55h                ; Load 55h into
MOVWF   TBLAT              ; TBLAT
TBLWT*                     ; Write it
```

7.0 8 X 8 HARDWARE MULTIPLIER

An 8 x 8 hardware multiplier is included in the ALU of PIC18C601/801 devices. By making the multiply a hardware operation, it completes in a single instruction cycle. This is an unsigned multiply that gives a 16-bit result. The result is stored into the 16-bit product register pair (PRODH:PRODL). The multiplier does not affect any flags in the STATUS register.

Making the 8 x 8 multiplier execute in a single cycle gives the following advantages:

- Higher computational throughput
- Reduces code size requirements for multiply algorithms

The performance increase allows the device to be used in some applications previously reserved for Digital Signal Processors.

Table 7-1 shows a performance comparison between enhanced devices using the single cycle hardware multiply, and performing the same function without the hardware multiply.

TABLE 7-1: PERFORMANCE COMPARISON

Routine	Multiply Method	Program Memory (Words)	Cycles (Max)	Time		
				@ 25 MHz	@ 10 MHz	@ 4 MHz
8 x 8 unsigned	Without hardware multiply	13	69	11.0 μs	27.6 μs	69.0 μs
	Hardware multiply	1	1	160.0 ns	400.0 ns	1.0 μs
8 x 8 signed	Without hardware multiply	33	91	14.6 μs	36.4 μs	91.0 μs
	Hardware multiply	6	6	960.0 ns	2.4 μs	6.0 μs
16 x 16 unsigned	Without hardware multiply	21	242	38.7 μs	96.8 μs	242.0 μs
	Hardware multiply	24	24	3.8 μs	9.6 μs	24.0 μs
16 x 16 signed	Without hardware multiply	52	254	40.6 μs	102.6 μs	254.0 μs
	Hardware multiply	36	36	5.8 μs	14.4 μs	36.0 μs

9.3 PORTC, TRISC and LATC Registers

PORTC is an 8-bit wide, bi-directional port. The corresponding data direction register is TRISC. Setting a TRISC bit (= 1) will make the corresponding PORTC pin an input (i.e., put the corresponding output driver in a Hi-Impedance mode). Clearing a TRISC bit (= 0) will make the corresponding PORTC pin an output (i.e., put the contents of the output latch on the selected pin).

Read-modify-write operations on the LATC register, read and write the latched output value for PORTC.

PORTC is multiplexed with several peripheral functions (Table 9-5). PORTC pins have Schmitt Trigger input buffers.

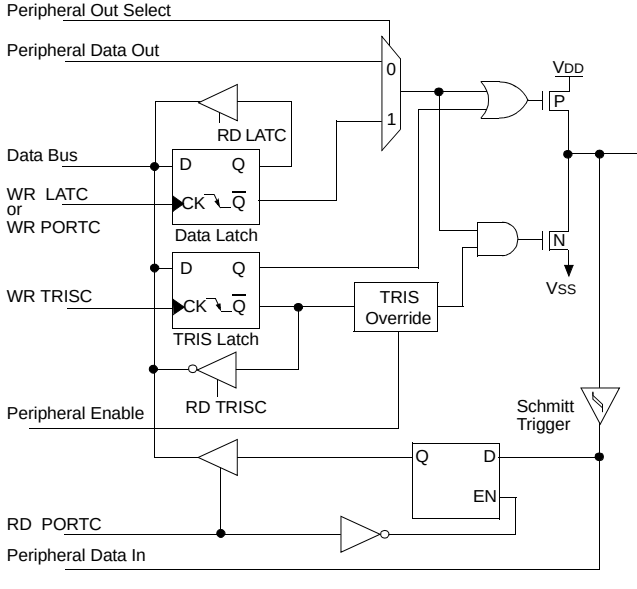
When enabling peripheral functions, care should be taken in defining TRIS bits for each PORTC pin. Some peripherals override the TRIS bit to make a pin an output, while other peripherals override the TRIS bit to make a pin an input. The user should refer to the corresponding peripheral section for the correct TRIS bit settings.

The pin override value is not loaded into the TRIS register. This allows read-modify-write of the TRIS register, without concern due to peripheral overrides.

EXAMPLE 9-3: INITIALIZING PORTC

```
CLRF    PORTC      ; Initialize PORTC by
                   ; clearing output
                   ; data latches
CLRF    LATC       ; Alternate method
                   ; to clear output
                   ; data latches
MOVLW   0CFh       ; Value used to
                   ; initialize data
                   ; direction
MOVWF   TRISC      ; Set RC3:RC0 as inputs
                   ; RC5:RC4 as outputs
                   ; RC7:RC6 as inputs
```

FIGURE 9-6: PORTC BLOCK DIAGRAM (PERIPHERAL OUTPUT OVERRIDE)



TRIS OVERRIDE

Pin	Override	Peripheral
RC0	Yes	Timer1 OSC for Timer1/Timer3
RC1	Yes	Timer1 OSC for Timer1/Timer3
RC2	No	—
RC3	Yes	SPI/I^{2}C Master Clock
RC4	Yes	I^{2}C Data Out
RC5	Yes	SPI Data Out
RC6	Yes	USART Async Xmit, Sync Clock
RC7	Yes	USART Sync Data Out

Note 1: I/O pins have diode protection to VDD and VSS.

TABLE 9-9: PORTE FUNCTIONS

Name	Bit#	Buffer Type	Function
RE0/AD8/A8(2)	bit0	ST/TTL(1)	Input/output port pin or Address/Data bit 8
RE1/AD9/A9(2)	bit1	ST/TTL(1)	Input/output port pin or Address/Data bit 9
RE2/AD10/A10(2)	bit2	ST/TTL(1)	Input/output port pin or Address/Data bit 10
RE3/AD11/A11(2)	bit3	ST/TTL(1)	Input/output port pin or Address/Data bit 11
RE4/AD12/A12(2)	bit4	ST/TTL(1)	Input/output port pin or Address/Data bit 12
RE5/AD13/A13(2)	bit5	ST/TTL(1)	Input/output port pin or Address/Data bit 13
RE6/AD14/A14(2)	bit6	ST/TTL(1)	Input/output port pin or Address/Data bit 14
RE7/AD15/A15(2)	bit7	ST/TTL(1)	Input/output port pin or Address/Data bit 15

Legend: ST = Schmitt Trigger input, TTL = TTL input

Note 1: Input buffers are Schmitt Triggers when in I/O mode and TTL buffers when in System Bus mode.

2: REx is used as a multiplexed address/data bus for PIC18C601 and PIC18C801 in 16-bit mode, and as an address only for PIC18C801 in 8-bit mode.

TABLE 9-10: SUMMARY OF REGISTERS ASSOCIATED WITH PORTE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Value on all other RESETS
TRISE	PORTE Data Direction Control Register								1111 1111	1111 1111
PORTE	Read PORTE pin/Write PORTE Data Latch								xxxx xxxx	uuuu uuuu
LATE	Read PORTE Data Latch/Write PORTE Data Latch								xxxx xxxx	uuuu uuuu
MEMCON	EBDIS	PGRM	WAIT1	WAIT0	—	—	WM1	WM0	0000 --00	0000 --00

Legend: x = unknown, u = unchanged. Shaded cells are not used by PORTE.

TABLE 11-2: 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	Value on POR, BOR	Value on all other RESETS
INTCON	GIE/ GIEH	PEIE/ GIEL	TMR0IE	INT0IE	RBIE	TMR0IF	INT0IF	RBIF	0000 000x	0000 000u
PIR1	—	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	-000 0000	-000 0000
PIE1	—	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	-000 0000	-000 0000
IPR1	—	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	-000 0000	-000 0000
TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
T1CON	RD16	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	0-00 0000	u-uu uuuu

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'. Shaded cells are not used by the Timer1 module.

13.1 Timer3 Operation

Timer3 can operate in one of these modes:

- As a timer
- As a synchronous counter
- As an asynchronous counter

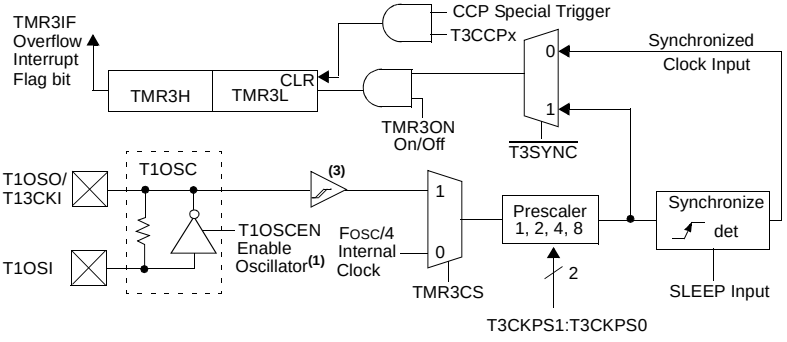
The operating mode is determined by the clock select bit, TMR3CS (T3CON register).

When TMR3CS = 0, Timer3 increments every instruction cycle. When TMR3CS = 1, Timer3 increments on every rising edge of the Timer1 external clock input or the Timer1 oscillator, if enabled.

When the Timer1 oscillator is enabled (T1OSCEN is set), the RC1/T1OSI and RC0/T1OSO/T1CKI pins become inputs. That is, the TRISC<1:0> value is ignored.

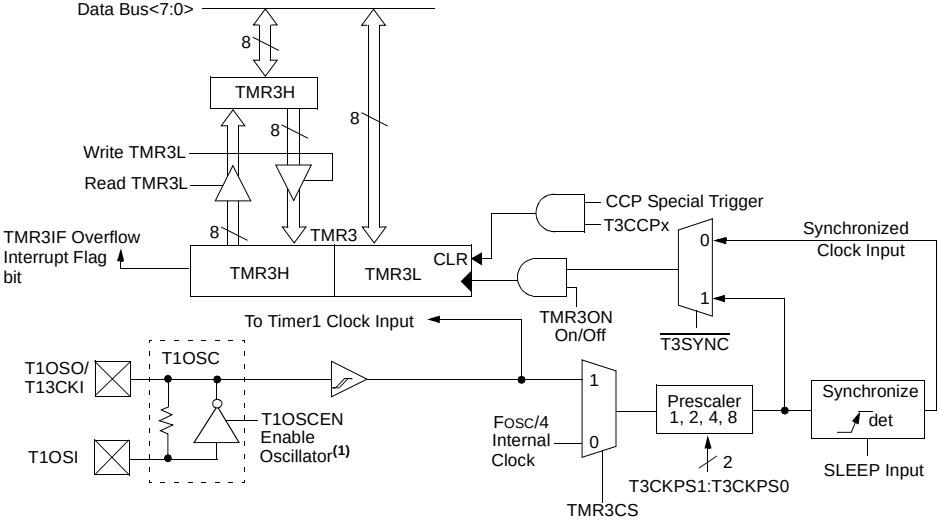
Timer3 also has an internal “RESET input”. This RESET can be generated by the CCP module (Section 13.0).

FIGURE 13-1: TIMER3 BLOCK DIAGRAM



Note 1: When enable bit T1OSCEN is cleared, the inverter and feedback resistor are turned off. This reduces power drain.

FIGURE 13-2: TIMER3 BLOCK DIAGRAM CONFIGURED IN 16-BIT READ/WRITE MODE



Note 1: When enable bit T1OSCEN is cleared, the inverter and feedback resistor are turned off. This reduces power drain.

15.0 MASTER SYNCHRONOUS SERIAL PORT (MSSP) MODULE

15.1 Master SSP (MSSP) Module Overview

The Master Synchronous Serial Port (MSSP) module is a serial interface useful for communicating with other peripheral or microcontroller devices. These peripheral devices may be Serial EEPROMs, shift registers, display drivers, A/D converters, etc. The MSSP module can operate in one of two modes:

- Serial Peripheral Interface™ (SPI)
- Inter-Integrated Circuit™ (I^2C)
  - Full Master mode
  - Slave mode (with general address call)

The I^2C interface supports the following modes in hardware:

- Master mode
- Multi-Master mode
- Slave mode

15.2 Control Registers

The MSSP module has three associated registers. These include a status register and two control registers.

Register 15-1 shows the MSSP Status Register (SSPSTAT), Register 15-2 shows the MSSP Control Register 1 (SSPCON1), and Register 15-3 shows the MSSP Control Register 2 (SSPCON2).

REGISTER 15-1: SSPSTAT REGISTER

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF
bit 7							bit 0

bit 7 **SMP:** Sample bit
SPI Master mode:
1 = Input data sampled at end of data output time
0 = Input data sampled at middle of data output time
SPI Slave mode:
SMP must be cleared when SPI is used in Slave mode
In I^2C Master or Slave mode:
1= Slew rate control disabled for standard speed mode (100 kHz and 1 MHz)
0= Slew rate control enabled for high speed mode (400 kHz)

bit 6 **CKE:** SPI Clock Edge Select
CKP = 0:
1 = Data transmitted on rising edge of SCK
0 = Data transmitted on falling edge of SCK
CKP = 1:
1 = Data transmitted on falling edge of SCK
0 = Data transmitted on rising edge of SCK

bit 5 **D/$\overline{A}$:** Data/$\overline{Address}$ bit (I^2C mode only)
1 = Indicates that the last byte received or transmitted was data
0 = Indicates that the last byte received or transmitted was address

bit 4 **P:** STOP bit
(I^2C mode only. This bit is cleared when the MSSP module is disabled, SSPEN is cleared.)
1 = Indicates that a STOP bit has been detected last (this bit is '0' on RESET)
0 = STOP bit was not detected last

bit 3 **S:** START bit
(I^2C mode only. This bit is cleared when the MSSP module is disabled, SSPEN is cleared.)
1 = Indicates that a START bit has been detected last (this bit is '0' on RESET)
0 = START bit was not detected last

bit 2 **R/$\overline{W}$:** Read/Write bit Information (I^2C mode only)
This bit holds the R/W bit information following the last address match. This bit is only valid from the address match to the next START bit, STOP bit, or not $\overline{ACK}$ bit.
In I^2C Slave mode:
1 = Read
0 = Write
In I^2C Master mode:
1 = Transmit is in progress
0 = Transmit is not in progress.
OR-ing this bit with SEN, RSEN, PEN, RCEN, or ACKEN will indicate if the MSSP is in IDLE mode.

bit 1 **UA:** Update Address (10-bit I^2C mode only)
1 = Indicates that the user needs to update the address in the SSPADD register
0 = Address does not need to be updated

bit 0 **BF:** Buffer Full Status bit
Receive (SPI and I^2C modes):
1 = Receive complete, SSPBUF is full
0 = Receive not complete, SSPBUF is empty
Transmit (I^2C mode only):
1 = Data transmit in progress (does not include the $\overline{ACK}$ and STOP bits), SSPBUF is full
0 = Data transmit complete (does not include the ACK and STOP bits), SSPBUF is empty

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

REGISTER 15-3: SSPCON2 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
GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

bit 7 **GCEN:** General Call Enable bit (In I^2C Slave mode only)
1 = Enable interrupt when a general call address (0000h) is received in the SSPSR
0 = General call address disabled

bit 6 **ACKSTAT:** Acknowledge Status bit (In I^2C Master mode only)
In Master Transmit mode:
1 = Acknowledge was not received from slave
0 = Acknowledge was received from slave

bit 5 **ACKDT:** Acknowledge Data bit (In I^2C Master mode only)
In Master Receive mode:
Value transmitted when the user initiates an Acknowledge sequence at the end of a receive
1 = Not Acknowledge
0 = Acknowledge

bit 4 **ACKEN:** Acknowledge Sequence Enable bit (In I^2C Master mode only)
In Master Receive mode:
1 = Initiate Acknowledge sequence on SDA and SCL pins, and transmit ACKDT data bit. Automatically cleared by hardware.
0 = Acknowledge sequence idle

bit 3 **RCEN:** Receive Enable bit (In I^2C Master mode only)
1 = Enables Receive mode for I^2C
0 = Receive idle

bit 2 **PEN:** STOP Condition Enable bit (In I^2C Master mode only)
SCK release control
1 = Initiate STOP condition on SDA and SCL pins. Automatically cleared by hardware.
0 = STOP condition idle

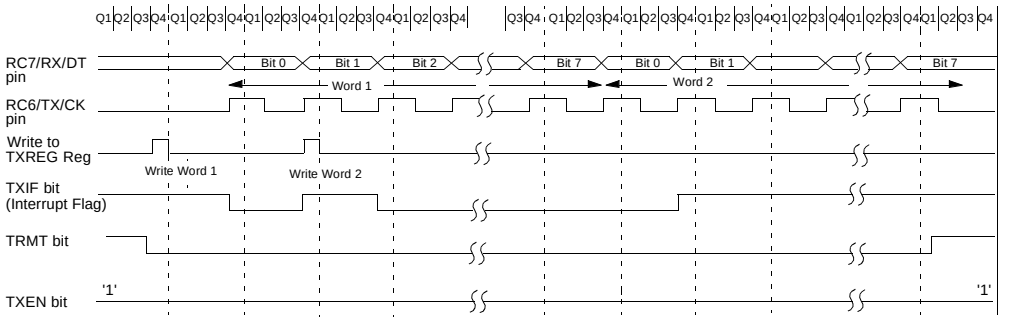
bit 1 **RSEN:** Repeated START Condition Enabled bit (In I^2C Master mode only)
1 = Initiate Repeated START condition on SDA and SCL pins. Automatically cleared by hardware.
0 = Repeated START condition idle

bit 0 **SEN:** START Condition Enabled bit (In I^2C Master mode only)
1 = Initiate START condition on SDA and SCL pins. Automatically cleared by hardware.
0 = START condition idle

Note: For bits ACKEN, RCEN, PEN, RSEN, SEN: If the I^2C module is not in the IDLE mode, this bit may not be set (no spooling) and the SSPBUF may not be written (or writes to the SSPBUF are disabled).

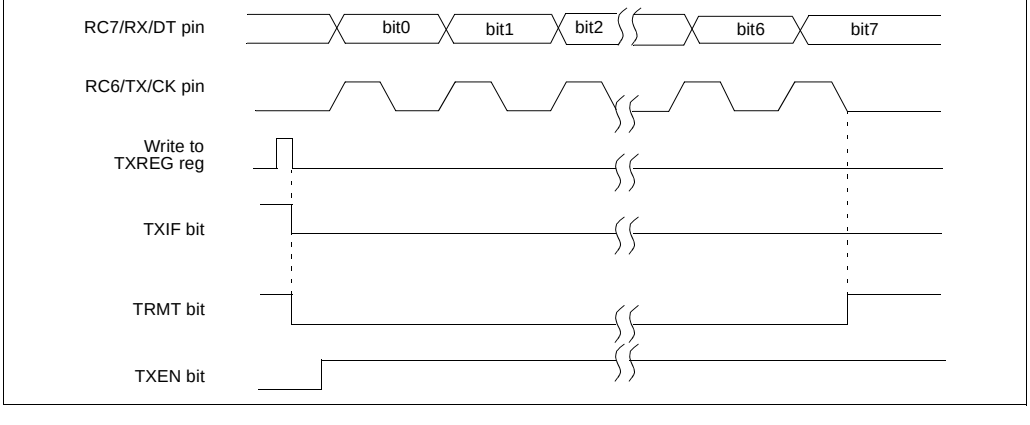
Legend:			
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'	
- n = Value at POR	'1' = Bit is set	'0' = Bit is cleared	x = Bit is unknown

FIGURE 16-6: SYNCHRONOUS TRANSMISSION



Note: Sync Master mode; SPBRG = '0'; continuous transmission of two 8-bit words.

FIGURE 16-7: SYNCHRONOUS TRANSMISSION (THROUGH TXEN)

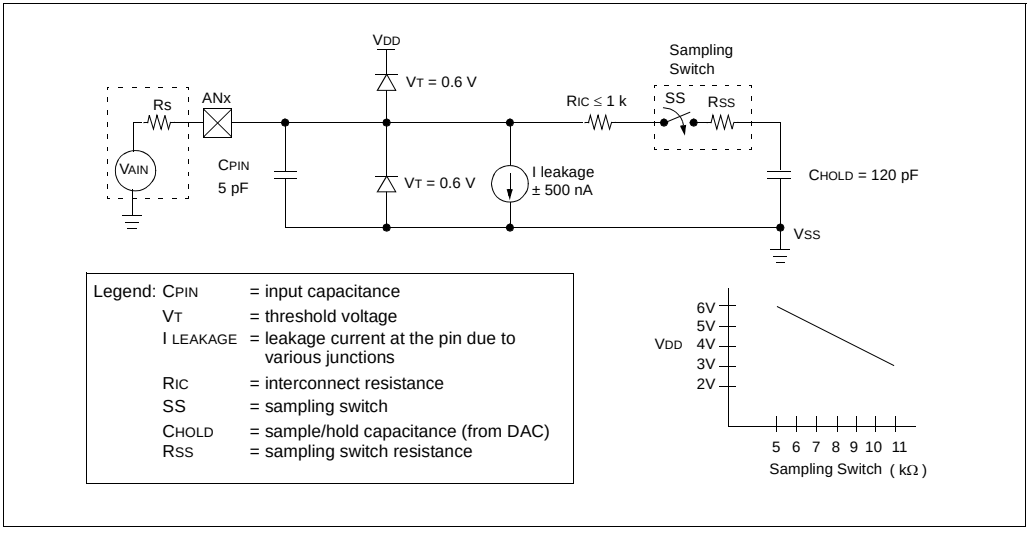


The value in the ADRESH/ADRESL registers is not modified for a Power-on Reset. The ADRESH/ADRESL registers will contain unknown data after a Power-on Reset.

After the A/D module has been configured as desired, the selected channel must be acquired before the conversion is started. The analog input channels must have their corresponding TRIS bits selected as an input. To determine acquisition time, see Section 17.1. After this acquisition time has elapsed, the A/D conversion can be started. The following steps should be followed to do an A/D conversion:

1. Configure the A/D module:
   - Configure analog pins, voltage reference and digital I/O (ADCON1)
   - Select A/D input channel (ADCON0)
   - Select A/D conversion clock (ADCON2)
   - Turn on A/D module (ADCON0)
2. Configure A/D interrupt (if desired):
   - Clear ADIF bit
   - Set ADIE bit
   - Set GIE bit
3. Wait the required acquisition time.
4. Start conversion:
   - Set GO/$\overline{\text{DONE}}$ bit (ADCON0 register)
5. Wait for A/D conversion to complete, by either:
   - Polling for the GO/$\overline{\text{DONE}}$ bit to be cleared, OR
   - Waiting for the A/D interrupt
6. Read A/D Result registers (ADRESH:ADRESL); clear bit ADIF, if required.
7. For next conversion, go to step 1 or step 2, as required. The A/D conversion time per bit is defined as TAD. A minimum wait of 2TAD is required before next acquisition starts.

FIGURE 17-2: ANALOG INPUT MODEL



BCF **Bit Clear f**

Syntax: [*label*] BCF f, b [,a]

Operands: $0 \le f \le 255$
$0 \le b \le 7$
$a \in [0,1]$

Operation: $0 \rightarrow$ f<b>

Status Affected: None

Encoding:

1001	bbba	ffff	ffff

Description: Bit 'b' in register 'f' is cleared. If 'a' is 0, the Access Bank will be selected, overriding the BSR value. If 'a' = 1, the Bank will be selected as per the BSR value.

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example: `BCF FLAG_REG, 7`

Before Instruction
FLAG_REG = 0C7h

After Instruction
FLAG_REG = 47h

BN **Branch if Negative**

Syntax: [*label*] BN n

Operands: $-128 \le n \le 127$

Operation: if negative bit is '1'
(PC) + 2 + 2n $\rightarrow$ PC

Status Affected: None

Encoding:

1110	0110	nnnn	nnnn

Description: If the Negative bit is '1', 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 BN Jump`

Before Instruction
PC = address `(HERE)`

After Instruction
If Negative = 1;
PC = address `(Jump)`
If Negative = 0;
PC = address `(HERE+2)`

BRA **Unconditional Branch**

Syntax: [*label*] BRA n

Operands: -1024 ≤ n ≤ 1023

Operation: (PC) + 2 + 2n → PC

Status Affected: None

Encoding:

1101	0nnn	nnnn	nnnn

Description: Add the 2's complement number '2n' 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 a two-cycle instruction.

Words: 1

Cycles: 2

Q Cycle Activity:

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

Example: `HERE BRA Jump`

Before Instruction
PC = address `(HERE)`

After Instruction
PC = address `(Jump)`

BSF **Bit Set f**

Syntax: [*label*] BSF f, b [,a]

Operands: 0 ≤ f ≤ 255
0 ≤ b ≤ 7
a ∈ [0,1]

Operation: 1 → f<b>

Status Affected: None

Encoding:

1000	bbba	ffff	ffff

Description: Bit 'b' in register 'f' is set. If 'a' is 0 Access Bank will be selected, overriding the BSR value. If 'a' is 1, the Bank will be selected as per the BSR value (default).

Words: 1

Cycles: 1

Q Cycle Activity:

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

Example: `BSF FLAG_REG, 7`

Before Instruction
FLAG_REG = 0Ah

After Instruction
FLAG_REG = 8Ah

CPFSGT — **Compare f with WREG, skip if f > WREG**

Syntax: [*label*] CPFSGT f [,a]

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

Operation: (f) – (WREG),
skip if (f) > (WREG)
(unsigned comparison)

Status Affected: None

Encoding:

0110	010a	ffff	ffff

Description: Compares the contents of data memory location 'f' to the contents of the WREG by performing an unsigned subtraction.

If the contents of 'f' are greater than the contents of **,** then the fetched instruction is discarded and a NOP is executed instead, making this a two-cycle instruction. If 'a' is 0, the Access Bank will be selected, overriding the BSR value. If 'a' is 1, the Bank will be selected as per the BSR value.

Words: 1

Cycles: 1(2)
Note: 3 cycles if skip and followed by a 2-word instruction.

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	No operation

If skip:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation

If skip and followed by 2-word instruction:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation
No operation	No operation	No operation	No operation

Example:
```
HERE      CPFSGT REG
NGREATER  :
GREATER   :
```
Before Instruction
PC = Address (HERE)
WREG = ?
After Instruction
If REG > WREG;
PC = Address (GREATER)
If REG ≤ WREG;
PC = Address (NGREATER)

CPFSLT — **Compare f with WREG, skip if f < WREG**

Syntax: [*label*] CPFSLT f [,a]

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

Operation: (f) – (WREG),
skip if (f) < (WREG)
(unsigned comparison)

Status Affected: None

Encoding:

0110	000a	ffff	ffff

Description: Compares the contents of data memory location 'f' to the contents of WREG by performing an unsigned subtraction.

If the contents of 'f' are less than the contents of WREG, then the fetched instruction is discarded and a NOP is executed instead, making this a two-cycle instruction. If 'a' is 0, the Access Bank will be selected. If 'a' is 1, the Bank will be selected as per the BSR value.

Words: 1

Cycles: 1(2)
Note: 3 cycles if skip and followed by a 2-word instruction.

Q Cycle Activity:

Q1	Q2	Q3	Q4
Decode	Read register 'f'	Process Data	No operation

If skip:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation

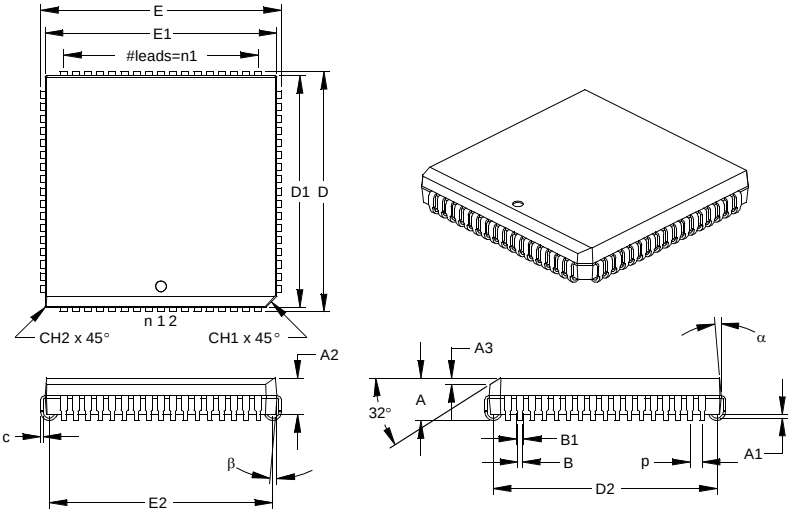
If skip and followed by 2-word instruction:

Q1	Q2	Q3	Q4
No operation	No operation	No operation	No operation
No operation	No operation	No operation	No operation

Example:
```
HERE    CPFSLT REG
NLESS   :
LESS    :
```
Before Instruction
PC = Address (HERE)
WREG = ?
After Instruction
If REG < WREG;
PC = Address (LESS)
If REG ≥ WREG;
PC = Address (NLESS)

84-Lead Plastic Leaded Chip Carrier (L) – Square (PLCC)

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



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		68			68	
Pitch	p		.050			1.27	
Pins per Side	n1		17			17	
Overall Height	A	.165	.173	.180	4.19	4.39	4.57
Molded Package Thickness	A2	.145	.153	.160	3.68	3.87	4.06
Standoff §	A1	.020	.028	.035	0.51	0.71	0.89
Side 1 Chamfer Height	A3	.024	.029	.034	0.61	0.74	0.86
Corner Chamfer 1	CH1	.040	.045	.050	1.02	1.14	1.27
Corner Chamfer (others)	CH2	.000	.005	.010	0.00	0.13	0.25
Overall Width	E	.985	.990	.995	25.02	25.15	25.27
Overall Length	D	.985	.990	.995	25.02	25.15	25.27
Molded Package Width	E1	.950	.954	.958	24.13	24.23	24.33
Molded Package Length	D1	.950	.954	.958	24.13	24.23	24.33
Footprint Width	E2	.890	.920	.930	22.61	23.37	23.62
Footprint Length	D2	.890	.920	.930	22.61	23.37	23.62
Lead Thickness	c	.008	.011	.013	0.20	0.27	0.33
Upper Lead Width	B1	.026	.029	.032	0.66	0.74	0.81
Lower Lead Width	B	.013	.020	.021	0.33	0.51	0.53
Mold Draft Angle Top	α	0	5	10	0	5	10
Mold Draft Angle Bottom	β	0	5	10	0	5	10

* Controlling Parameter
§ Significant Characteristic

Notes:
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.
JEDEC Equivalent: MO-047
Drawing No. C04-093

