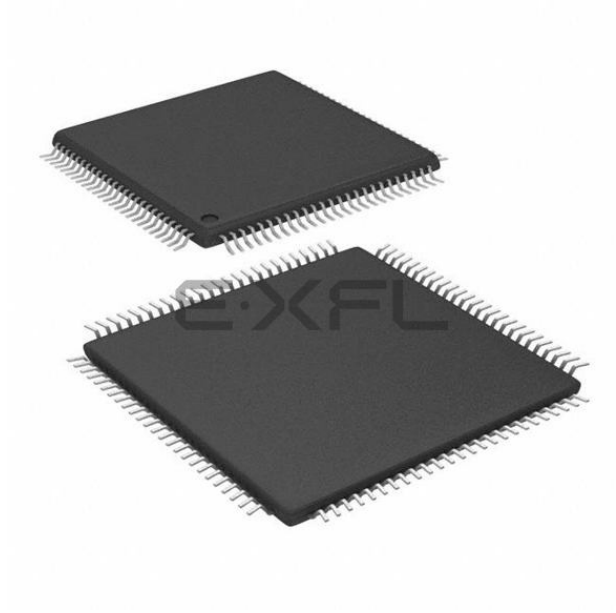


Welcome to E-XFL.COM

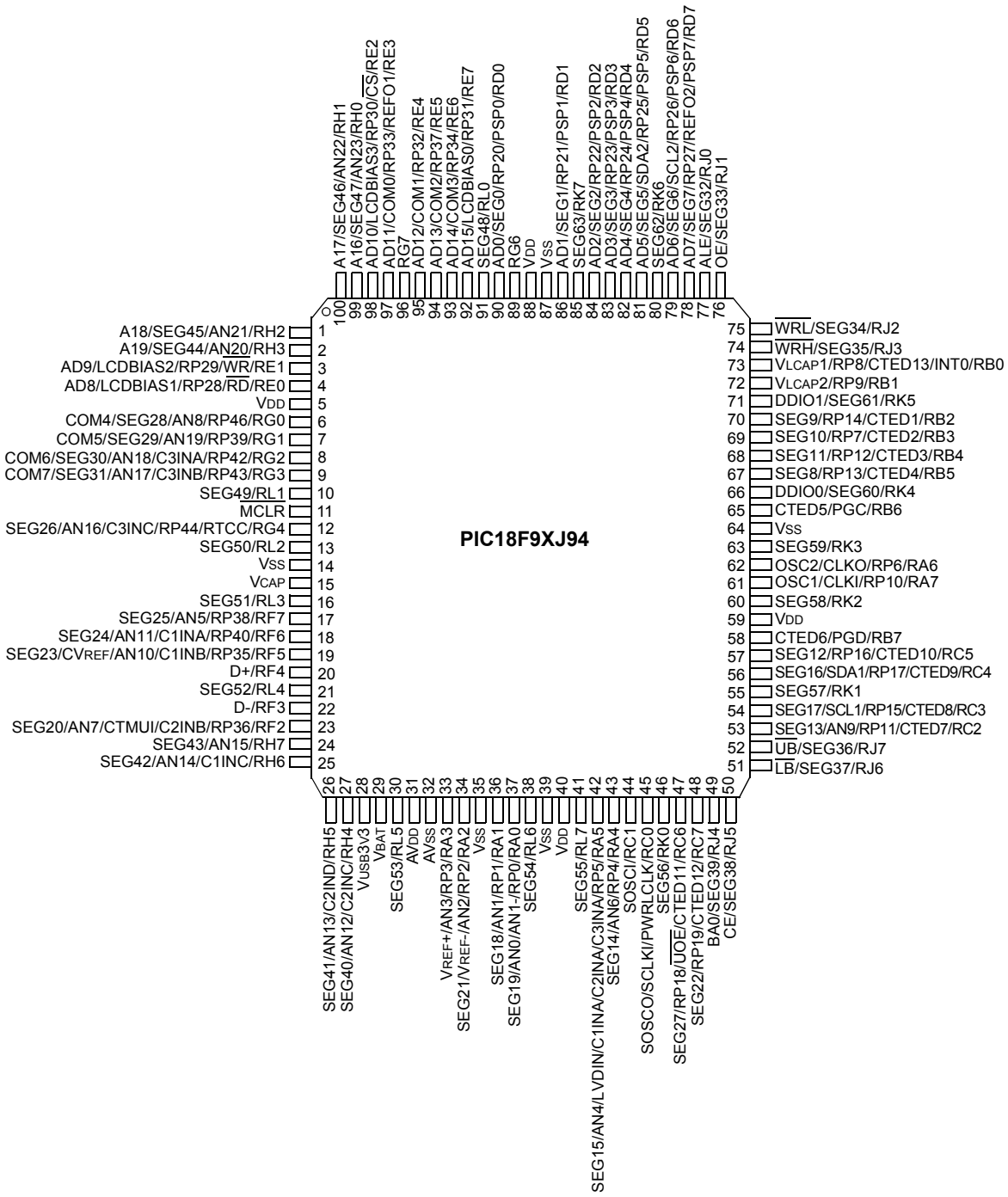
What is "Embedded - Microcontrollers"?

"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, IrDA, LINbus, SPI, UART/USART, USB
Peripherals	Brown-out Detect/Reset, HLVD, LCD, POR, PWM, WDT
Number of I/O	86
Program Memory Size	32KB (16K x 16)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 3.6V
Data Converters	A/D 24x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	100-TQFP
Supplier Device Package	100-TQFP (12x12)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f95j94t-i-pt

FIGURE 3: 100-PIN TQFP DIAGRAM FOR PIC18F9XJ94



Note 1: Pinouts are subject to change.

2: See Table 4 for the pin allocation table.

TABLE 4: 100-PIN ALLOCATION TABLE (PIC18F9XJ94) (CONTINUED)

I/O	100-Pin TQFP	ADC	Comparator	HLVD	CTMU	USB	LCD	MSSP	PSP	Interrupt	REFO	EMB	PPS-Lite(1)	Pull-up	Basic
RL6	38	—	—	—	—	—	SEG54	—	—	—	—	—	—	Y	—
RL7	41	—	—	—	—	—	SEG55	—	—	—	—	—	—	Y	—
AVDD	31	AVDD	—	—	—	—	—	—	—	—	—	—	—	—	—
AVSS	32	AVSS	—	—	—	—	—	—	—	—	—	—	—	—	—
VBAT	29	—	—	—	—	—	—	—	—	—	—	—	—	—	VBAT
VCAP/ VDDCORE	15	—	—	—	—	—	—	—	—	—	—	—	—	—	VCAP/ VDDCORE
VDD	5, 40, 59, 88	—	—	—	—	—	—	—	—	—	—	—	—	—	VDD
VSS	14, 35, 39, 64, 87	—	—	—	—	—	—	—	—	—	—	—	—	—	VSS
VUSB3V3	28	—	—	—	—	—	—	—	—	—	—	—	—	—	VUSB3V3

Note 1: The peripheral inputs and outputs that support PPS have no default pins.

Table of Contents

In cases where the frequency accuracy is not met for USB operation, the FRCPLL mode should not be used when USB is active.

Note:	Using FRC postscaler values, other than '000'or '001', will cause the clock input to the PLL to be below the operating frequency input range and may cause undesirable operation.

3.11 Internal Low-Power RC Oscillator (LPRC)

The LPRC Oscillator is separate from the FRC and oscillates at a nominal frequency of 31 kHz. LPRC is the clock source for the Power-up Timer (PWRT), Watchdog Timer (WDT) and FSCM circuits. It may also be used to provide a low-frequency clock source option for the device, in those applications where power consumption is critical and timing accuracy is not required.

3.11.1 ENABLING THE LPRC OSCILLATOR

Since it serves the Power-up Timer (PWRT) clock source, the LPRC Oscillator is enabled at POR events whenever the on-board voltage regulator is disabled. After the PWRT expires, the LPRC Oscillator will remain on if any one of the following is true:

- The FSCM is enabled.
- The WDT is enabled.
- The LPRC Oscillator is selected as the system clock (COSC<2:0> = 101).

If none of the above is true, the LPRC will shut off after the PWRT expires.

3.12 Fail-Safe Clock Monitor (FSCM)

The Fail-Safe Clock Monitor (FSCM) allows the device to continue to operate, even in the event of an oscillator failure. The FSCM function is enabled by programming the FSCMx (Clock Switch and Monitor) bits in CONFIG3L<5:4>. FSCM is only enabled when the FSCM<1:0> bits (CONFIG3L<5:4>) = 00. When FSCM is enabled, the internal LPRC Oscillator will run at all times (except during Sleep mode).

In the event of an oscillator failure, the FSCM will generate a clock failure trap and will switch the system clock to the FRC Oscillator. The user will then have the option to either attempt to restart the oscillator or execute a controlled shutdown. FSCM will monitor the system clock source regardless of its source or oscillator mode. This includes the Primary Oscillator for all oscillator modes and the Secondary Oscillator, SOSC, when configured as the system clock.

The FSCM module takes the following actions when switching to the FRC Oscillator:

1. The COSCx bits are loaded with '000'.
2. The CF Status bit is set to indicate the clock failure.

Note:	For more information about the oscillator failure trap, refer to **Section 10.0 "Interrupts"**.

3.12.1 FSCM DELAY

On a POR, BOR or wake from Sleep mode event, a nominal delay (T_{FSCM}) may be inserted before the FSCM begins to monitor the system clock source. The purpose of the FSCM delay is to provide time for the oscillator and/or PLL to stabilize when the PWRT is not utilized. The FSCM delay will be generated after the internal System Reset signal, $\overline{SYSRST}$, has been released. Refer to **Section 28.4 "Fail-Safe Clock Monitor"** for FSCM delay timing information.

The T_{FSCM} interval is applied whenever the FSCM is enabled and the EC, HS or SOSC Oscillator modes are selected as the system clock.

Note:	Refer to the "**Electrical Characteristics**" section of the specific device data sheet for T_{FSCM} specification values.

3.12.2 FSCM AND SLOW OSCILLATOR START-UP

If the chosen device oscillator has a slow start-up time coming out of POR, BOR or Sleep mode, it is possible that the FSCM delay will expire before the oscillator has started. In this case, the FSCM will initiate a clock failure trap. As this happens, the COSCx bits are loaded with the FRC Oscillator selection. This will effectively shut off the original oscillator that was trying to start. The user can detect this situation and initiate a clock switch back to the desired oscillator in the Trap Service Routine (TSR).

3.12.3 FSCM AND WDT

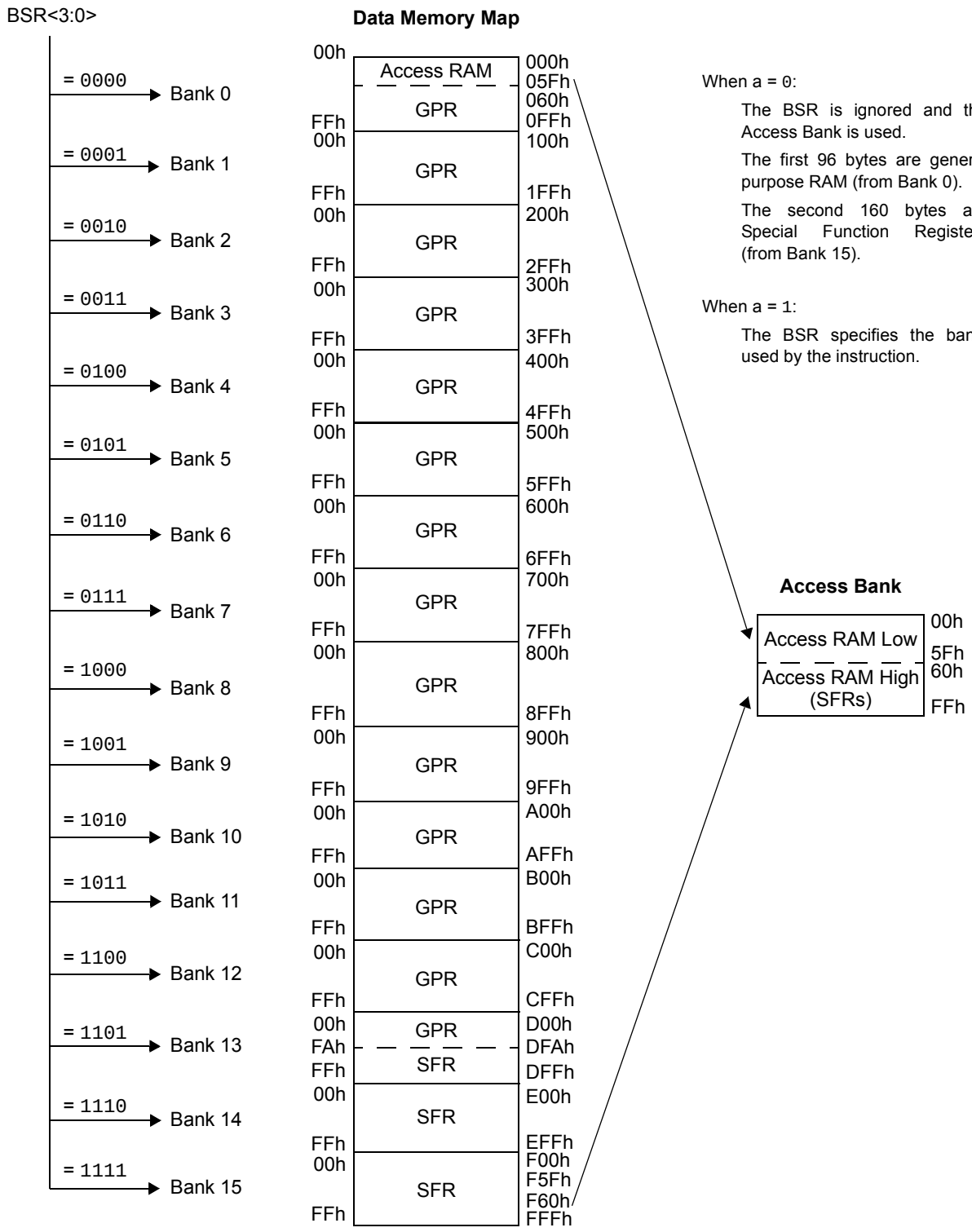
The FSCM and the WDT both use the LPRC Oscillator as their time base. In the event of a clock failure, the WDT is unaffected and continues to run on the LPRC.

3.13 Clock Switching Operation

With few limitations, applications are free to switch between any of the four clock sources (Primary, SOSC, FRC and LPRC) under software control and at any time. To limit the possible side effects that could result from this flexibility, PIC18F devices have a safeguard lock built into the switch process.

Note:	Primary Oscillator mode has three different submodes (MS, HS and EC), which are determined by the POSCMDx Configuration bits. While an application can switch to and from Primary Oscillator mode, in software, it cannot switch between the different primary submodes without reprogramming the device.

FIGURE 6-6: DATA MEMORY MAP FOR PIC18F97J94 FAMILY DEVICES



Note 1: Addresses, DFAh through F5Fh, are also SFRs, but are not part of the Access RAM. Users must always use the complete address, or load the proper BSR value, to access these registers.

7.0 FLASH PROGRAM MEMORY

The Flash program memory is readable, writable and erasable during normal operation over the entire VDD range.

A read from program memory is executed on 1 byte at a time. A write to program memory is executed on blocks of 64 bytes at a time or 2 bytes at a time. Program memory is erased in blocks of 512 bytes at a time. A bulk erase operation may not be issued from user code.

Writing or erasing program memory will cease instruction fetches until the operation is complete. The program memory cannot be accessed during the write or erase, therefore, code cannot execute. An internal programming timer terminates program memory writes and erases.

A value written to program memory does not need to be a valid instruction. Executing a program memory location that forms an invalid instruction results in a NOP.

7.1 Table Reads and Table Writes

In order to read and write program memory, there are two operations that allow the processor to move bytes between the program memory space and the data RAM:

- Table Read (TBLRD)
- Table Write (TBLWT)

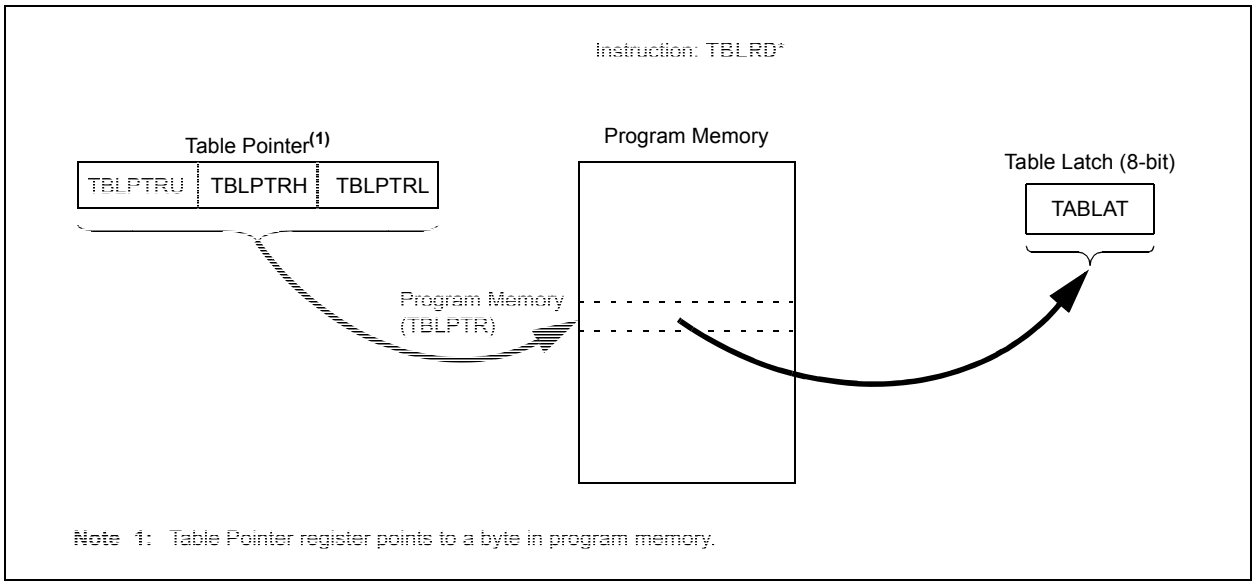
The program memory space is 16 bits wide, while the data RAM space is 8 bits wide. Table reads and table writes move data between these two memory spaces through an 8-bit register (TABLAT).

Table read operations retrieve data from program memory and place it into the data RAM space. Figure 7-1 shows the operation of a table read with program memory and data RAM.

Table write operations store data from the data memory space into holding registers in program memory. The procedure to write the contents of the holding registers into program memory is detailed in **Section 7.5 "Writing to Flash Program Memory"**. Figure 7-2 shows the operation of a table write with program memory and data RAM.

Table operations work with byte entities. A table block containing data, rather than program instructions, is not required to be word-aligned. Therefore, a table block can start and end at any byte address. If a table write is being used to write executable code into program memory, program instructions will need to be word-aligned.

FIGURE 7-1: TABLE READ OPERATION



REGISTER 10-3: INTCON3: INTERRUPT CONTROL REGISTER 3

R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
INT2IP	INT1IP	INT3IE	INT2IE	INT1IE	INT3IF	INT2IF	INT1IF
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 **INT2IP:** INT2 External Interrupt Priority bit
1 = High priority
0 = Low priority

bit 6 **INT1IP:** INT1 External Interrupt Priority bit
1 = High priority
0 = Low priority

bit 5 **INT3IE:** INT3 External Interrupt Enable bit
1 = Enables the INT3 external interrupt
0 = Disables the INT3 external interrupt

bit 4 **INT2IE:** INT2 External Interrupt Enable bit
1 = Enables the INT2 external interrupt
0 = Disables the INT2 external interrupt

bit 3 **INT1IE:** INT1 External Interrupt Enable bit
1 = Enables the INT1 external interrupt
0 = Disables the INT1 external interrupt

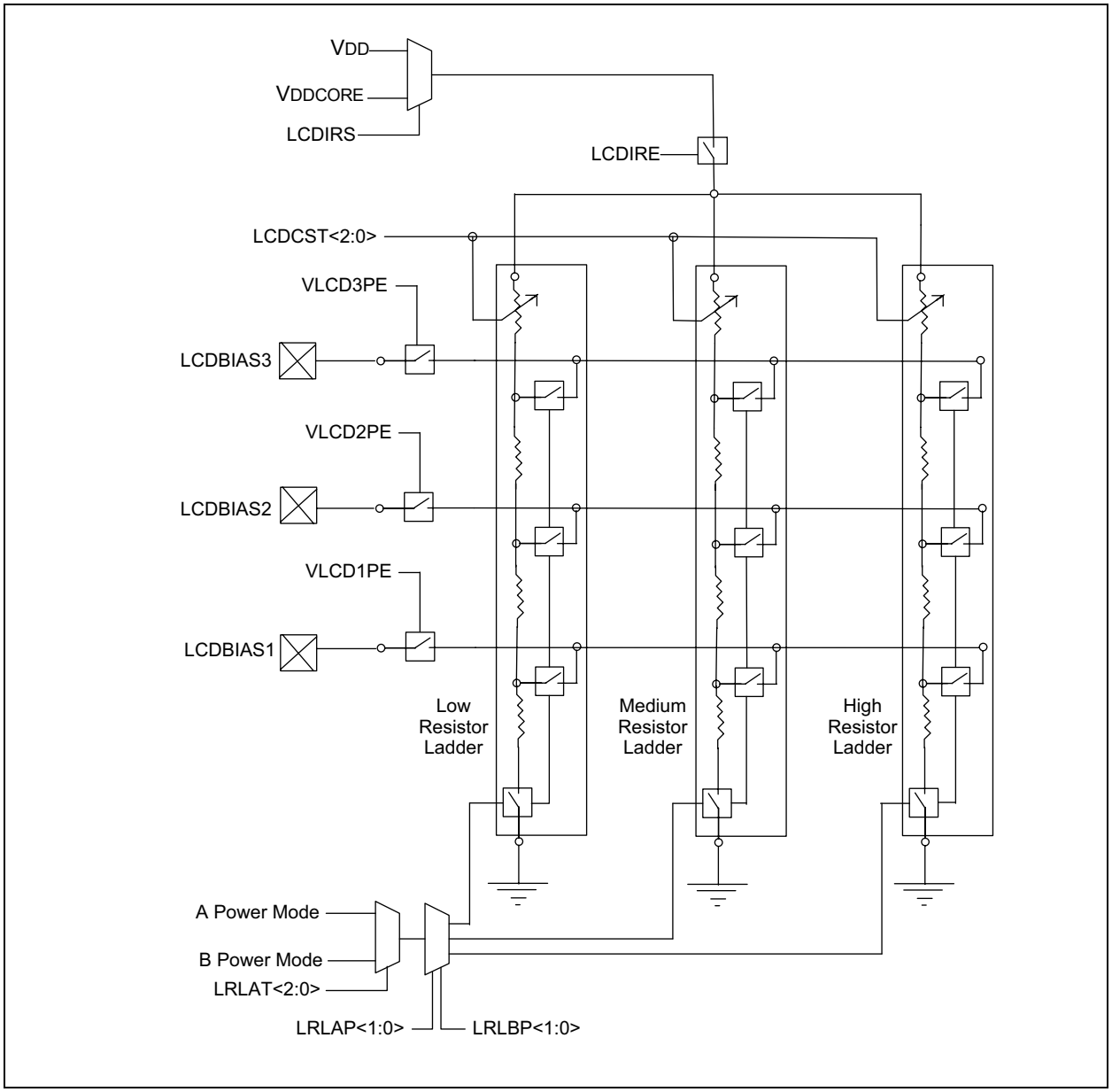
bit 2 **INT3IF:** INT3 External Interrupt Flag bit
1 = The INT3 external interrupt occurred (must be cleared in software)
0 = The INT3 external interrupt did not occur

bit 1 **INT2IF:** INT2 External Interrupt Flag bit
1 = The INT2 external interrupt occurred (must be cleared in software)
0 = The INT2 external interrupt did not occur

bit 0 **INT1IF:** INT1 External Interrupt Flag bit
1 = The INT1 external interrupt occurred (must be cleared in software)
0 = The INT1 external interrupt did not occur

Note: Interrupt flag bits are set when an interrupt condition occurs regardless of the state of its corresponding enable bit or the Global Interrupt Enable bit. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt. This feature allows for software polling.

FIGURE 13-3: LCD BIAS INTERNAL RESISTOR LADDER CONNECTION DIAGRAM



There are two power modes, designated as “Mode A” and “Mode B”. Mode A is set by the LRLAP<1:0> bits and Mode B by the LRLB<1:0> bits. 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 the LCDCST<2:0> bits (LCDREF<5:3>). 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 LRLAP<1:0> (LCDRL<7:6>) = 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.

REGISTER 13-7: LCDRL: LCD REFERENCE LADDER CONTROL REGISTER LOW

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
LRLAP1	LRLAP0	LRLBP1	LRLBP0	—	LRLAT2	LRLAT1	LRLAT0
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 **LRLAP<1:0>**: LCD Reference Ladder A Time Power Control bits

During Time Interval A:

11 = Internal LCD reference ladder is powered in High-Power mode
10 = Internal LCD reference ladder is powered in Medium Power mode
01 = Internal LCD reference ladder is powered in Low-Power mode
00 = Internal LCD reference ladder is powered down and unconnected

bit 5-4 **LRLBP<1:0>**: LCD Reference Ladder B Time Power Control bits

During Time Interval B:

11 = Internal LCD reference ladder is powered in High-Power mode
10 = Internal LCD reference ladder is powered in Medium Power mode
01 = Internal LCD reference ladder is powered in Low-Power mode
00 = Internal LCD reference ladder is powered down and unconnected

bit 3 **Unimplemented**: Read as '0'

bit 2-0 **LRLAT<2:0>**: LCD Reference Ladder A Time Interval Control bits

Sets the number of 32 clock counts when the A Time Interval Power mode is active.

For Type-A Waveforms (WFT = 0):

111 = Internal LCD reference ladder is in A Power mode for 7 clocks and B Power mode for 9 clocks
110 = Internal LCD reference ladder is in A Power mode for 6 clocks and B Power mode for 10 clocks
101 = Internal LCD reference ladder is in A Power mode for 5 clocks and B Power mode for 11 clocks
100 = Internal LCD reference ladder is in A Power mode for 4 clocks and B Power mode for 12 clocks
011 = Internal LCD reference ladder is in A Power mode for 3 clocks and B Power mode for 13 clocks
010 = Internal LCD reference ladder is in A Power mode for 2 clocks and B Power mode for 14 clocks
001 = Internal LCD reference ladder is in A Power mode for 1 clock and B Power mode for 15 clocks
000 = Internal LCD reference ladder is always in B Power mode

For Type-B Waveforms (WFT = 1):

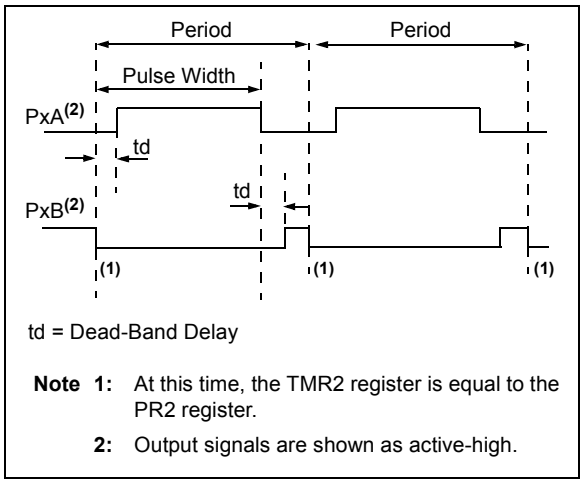
111 = Internal LCD reference ladder is in A Power mode for 7 clocks and B Power mode for 25 clocks
110 = Internal LCD reference ladder is in A Power mode for 6 clocks and B Power mode for 26 clocks
101 = Internal LCD reference ladder is in A Power mode for 5 clocks and B Power mode for 27 clocks
100 = Internal LCD reference ladder is in A Power mode for 4 clocks and B Power mode for 28 clocks
011 = Internal LCD reference ladder is in A Power mode for 3 clocks and B Power mode for 29 clocks
010 = Internal LCD reference ladder is in A Power mode for 2 clocks and B Power mode for 30 clocks
001 = Internal LCD reference ladder is in A Power mode for 1 clock and B Power mode for 31 clocks
000 = Internal LCD reference ladder is always in B Power mode

18.4.6 PROGRAMMABLE DEAD-BAND DELAY MODE

In half-bridge applications, where all power switches are modulated at the PWM frequency, the power switches normally require more time to turn off than to turn on. If both the upper and lower power switches are switched at the same time (one turned on and the other turned off), both switches may be on for a short period until one switch completely turns off. During this brief interval, a very high current (shoot-through current) will flow through both power switches, shorting the bridge supply. To avoid this potentially destructive shoot-through current from flowing during switching, turning on either of the power switches is normally delayed to allow the other switch to completely turn off.

In Half-Bridge mode, a digitally programmable dead-band delay is available to avoid shoot-through current from destroying the bridge power switches. The delay occurs at the signal transition from the non-active state to the active state. For an illustration, see Figure 18-14. The lower seven bits of the associated ECCPxDEL register (Register 18-4) set the delay period in terms of microcontroller instruction cycles (TCY or 4 TOSC).

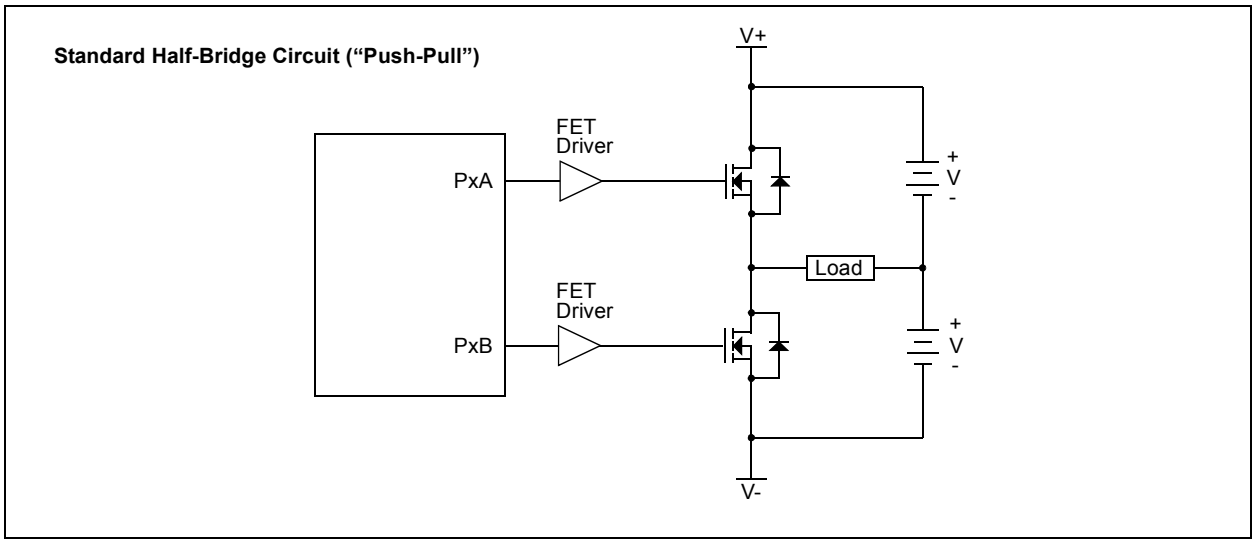
FIGURE 18-14: EXAMPLE OF HALF-BRIDGE PWM OUTPUT



Note 1: At this time, the TMR2 register is equal to the PR2 register.

2: Output signals are shown as active-high.

FIGURE 18-15: EXAMPLE OF HALF-BRIDGE APPLICATIONS



20.3.1 REGISTERS

Each MSSP module has four registers for SPI mode operation. These are:

- MSSPx Control Register 1 (SSPxCON1)
- MSSPx STATUS Register (SSPxSTAT)
- MSSPx Control Register 3 (SSPxCON3)
- Serial Receive/Transmit Buffer Register (SSPxBUF)
- MSSPx Shift Register (SSPxSR) – Not directly accessible

SSPxCON1, SSPxCON3 and SSPxSTAT are the control and STATUS registers in SPI mode operation. The SSPxCON1 and SSPxCON3 registers are readable and writable. The lower 6 bits of the SSPxSTAT are read-only. The upper two bits of the SSPxSTAT are read/write.

SSPxSR is the shift register used for shifting data in or out. SSPxBUF is the buffer register to which data bytes are written to or read from.

In receive operations, SSPxSR and SSPxBUF together, create a double-buffered receiver. When SSPxSR receives a complete byte, it is transferred to SSPxBUF and the SSPxIF interrupt is set.

During transmission, the SSPxBUF is not double-buffered. A write to SSPxBUF will write to both SSPxBUF and SSPxSR.

REGISTER 20-1: SSPxSTAT: MSSPx STATUS REGISTER (SPI MODE)

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE(1)	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF
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 **SMP:** Sample bit
SPI Master mode:
1 = Input data is sampled at the end of data output time
0 = Input data is sampled at the middle of data output time
SPI Slave mode:
SMP must be cleared when SPI is used in Slave mode.

bit 6 **CKE:** SPI Clock Select bit(1)
1 = Transmit occurs on the transition from active to Idle clock state
0 = Transmit occurs on the transition from Idle to active clock state

bit 5 **D/$\overline{A}$:** Data/$\overline{\text{Address}}$ bit
Used in I^2C mode only.

bit 4 **P:** Stop bit
Used in I^2C mode only. This bit is cleared when the MSSPx module is disabled; SSPEN is cleared.

bit 3 **S:** Start bit
Used in I^2C mode only.

bit 2 **R/$\overline{W}$:** Read/$\overline{\text{Write}}$ Information bit
Used in I^2C mode only.

bit 1 **UA:** Update Address bit
Used in I^2C mode only.

bit 0 **BF:** Buffer Full Status bit (Receive mode only)
1 = Receive is complete, SSPxBUF is full
0 = Receive is not complete, SSPxBUF is empty

Note 1: Polarity of clock state is set by the CKP bit (SSPxCON1<4>).

20.3.4 ENABLING SPI I/O

To enable the serial port, the peripheral must first be mapped to I/O pins using the PPS-Lite feature. To enable the SPI peripheral, the MSSPx Enable bit, SSPEN (SSPxCON1<5>) must be set. To reset or reconfigure SPI mode, clear the SSPEN bit, re-initialize the SSPxCON registers and then set the SSPEN bit. This configures the SDIx, SDOx, SCKx and $\overline{SSx}$ pins as serial port pins. For the pins to behave as the serial port function, some must have their data direction bits (in the TRIS register) appropriately programmed as follows:

- SDIx is automatically controlled by the SPI module
- SDOx must have the TRIS bit cleared for the corresponding RPn pin.
- SCKx (Master mode) must have the TRIS bit cleared for the corresponding RPn pin
- SCKx (Slave mode) must have the TRIS bit set for the corresponding RPn pin
- $\overline{SSx}$ must have the TRIS bit set for the corresponding RPn pin.

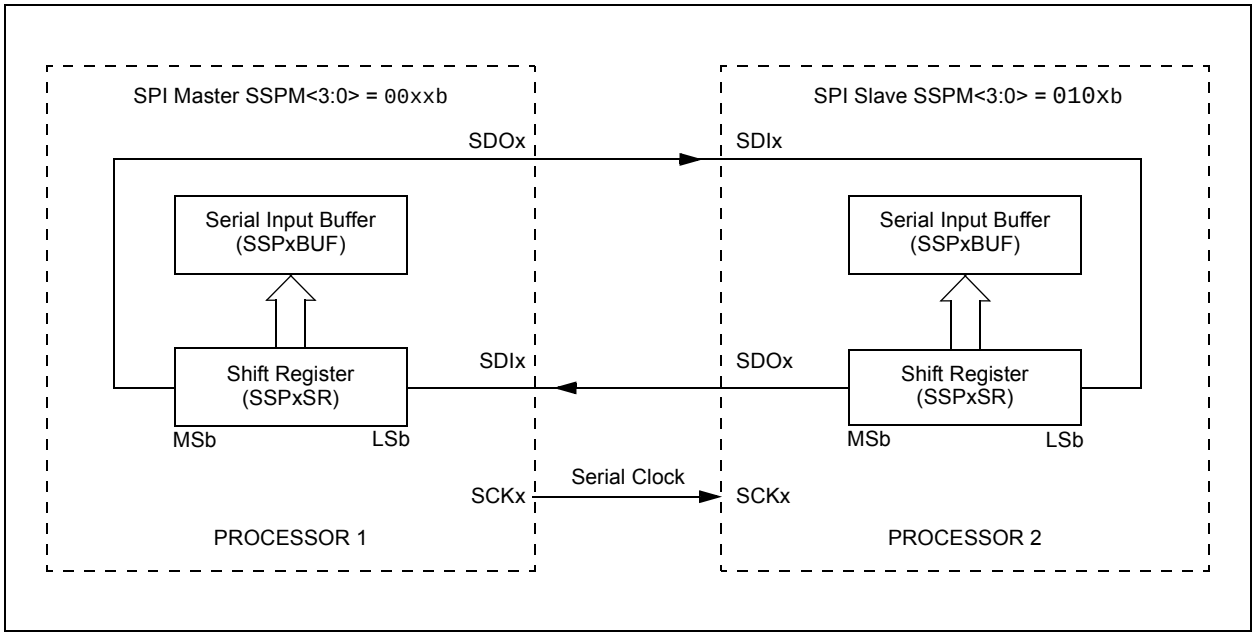
Any serial port function that is not desired may be overridden by programming the corresponding Data Direction (TRIS) register to the opposite value.

20.3.5 TYPICAL CONNECTION

Figure 20-2 shows a typical connection between two microcontrollers. The master controller (Processor 1) initiates the data transfer by sending the SCKx signal. Data is shifted out of both shift registers on their programmed clock edge and latched on the opposite edge of the clock. Both processors should be programmed to the same Clock Polarity (CKP), then both controllers would send and receive data at the same time. Whether the data is meaningful (or dummy data) depends on the application software. This leads to three scenarios for data transmission:

- Master sends data – Slave sends dummy data
- Master sends data – Slave sends data
- Master sends dummy data – Slave sends data

FIGURE 20-2: SPI MASTER/SLAVE CONNECTION



20.4.6 USING THE SPI DMA MODULE

The following steps would typically be taken to enable and use the SPI DMA module:

1. Configure the I/O pins, which will be used by MSSP2:
   a) Assign SCK1, SDO1, SDI1 and $\overline{SS1}$ to the RPn pins, as appropriate for the SPI mode which will be used. Only functions which will be used need to be assigned to a pin.
   b) Initialize the associated LATx registers for the desired Idle SPI bus state.
   c) If Open-Drain Output mode on SDO1 and SCK1 (Master mode) is desired, set ODCON1<1>.
   d) Configure the corresponding TRISx bits for each I/O pin used.
2. Configure and enable MSSP1 for the desired SPI operating mode:
   a) Select the desired operating mode (Master or Slave, SPI Mode 0, 1, 2 and 3) and configure the module by writing to the SSP1STAT and SSP1CON1 registers.
   b) Enable MSSP1 by setting SSP1CON1<5> = 1.
3. Configure the SPI DMA engine:
   a) Select the desired operating mode by writing the appropriate values to DMACON2 and DMACON1.
   b) Initialize the TXADDRH/TXADDRL Pointer (Full-Duplex or Half-Duplex Transmit Only mode).
   c) Initialize the RXADDRH/RXADDRL Pointer (Full-Duplex or Half-Duplex Receive Only mode).
   d) Initialize the DMABCH/DMABCL Byte Count register with the number of bytes to be transferred in the next SPI DMA operation.
   e) Set the DMAEN bit (DMACON1<0>).

   In SPI Master modes, this will initiate a DMA transaction. In SPI Slave modes, this will complete the initialization process, and the module will now be ready to begin receiving and/or transmitting data to the master device once the master starts the transaction.
4. Detect the SSP1IF interrupt condition (PIR1<3):
   a) If the interrupt was configured to occur at the completion of the SPI DMA transaction, the DMAEN bit (DMACON1<0>) will be clear. User firmware may prepare the module for another transaction by repeating Steps 3.b through 3.e.
   b) If the interrupt was configured to occur prior to the completion of the SPI DMA transaction, the DMAEN bit may still be set, indicating the transaction is still in progress. User firmware would typically use this interrupt condition to begin preparing new data for the next DMA transaction. Firmware should not repeat Steps 3.b. through 3.e. until the DMAEN bit is cleared by the hardware, indicating the transaction is complete.

Example 20-3 provides example code, demonstrating the initialization process and the steps needed to use the SPI DMA module to perform a 512-byte Full-Duplex Master mode transfer.

20.5.7 CLOCK STRETCHING

Both 7-Bit and 10-Bit Slave modes implement automatic clock stretching during a transmit sequence.

The SEN bit (SSPxCON2<0>) allows clock stretching to be enabled during receives. Setting SEN will cause the SCLx pin to be held low at the end of each data receive sequence.

20.5.7.1 Clock Stretching for 7-Bit Slave Receive Mode (SEN = 1)

In 7-Bit Slave Receive mode, on the falling edge of the ninth clock at the end of the $\overline{ACK}$ sequence, if the BF bit is set, the CKP bit in the SSPxCON1 register is automatically cleared, forcing the SCLx output to be held low. The CKP bit, being cleared to '0', will assert the SCLx line low. The CKP bit must be set in the user's ISR before reception is allowed to continue. By holding the SCLx line low, the user has time to service the ISR and read the contents of the SSPxBUF before the master device can initiate another receive sequence. This will prevent buffer overruns from occurring (see Figure 20-15).

Note 1: If the user reads the contents of the SSPxBUF before the falling edge of the ninth clock, thus clearing the BF bit, the CKP bit will not be cleared and clock stretching will not occur.

2: The CKP bit can be set in software regardless of the state of the BF bit. The user should be careful to clear the BF bit in the ISR before the next receive sequence in order to prevent an overflow condition.

20.5.7.2 Clock Stretching for 10-Bit Slave Receive Mode (SEN = 1)

In 10-Bit Slave Receive mode, during the address sequence, clock stretching automatically takes place but CKP is not cleared. During this time, if the UA bit is set after the ninth clock, clock stretching is initiated. The UA bit is set after receiving the upper byte of the 10-bit address, and following the receive of the second byte of the 10-bit address, with the R/$\overline{W}$ bit cleared to '0'. The release of the clock line occurs upon updating SSPxADD. Clock stretching will occur on each data receive sequence as described in 7-bit mode.

Note: If the user polls the UA bit and clears it by updating the SSPxADD register before the falling edge of the ninth clock occurs, and if the user hasn't cleared the BF bit by reading the SSPxBUF register before that time, then the CKP bit will still NOT be asserted low. Clock stretching, on the basis of the state of the BF bit, only occurs during a data sequence, not an address sequence.

20.5.7.3 Clock Stretching for 7-Bit Slave Transmit Mode

The 7-Bit Slave Transmit mode implements clock stretching by clearing the CKP bit after the falling edge of the ninth clock if the BF bit is clear. This occurs regardless of the state of the SEN bit.

The user's ISR must set the CKP bit before transmission is allowed to continue. By holding the SCLx line low, the user has time to service the ISR and load the contents of the SSPxBUF before the master device can initiate another transmit sequence (see Figure 20-10).

Note 1: If the user loads the contents of SSPxBUF, setting the BF bit before the falling edge of the ninth clock, the CKP bit will not be cleared and clock stretching will not occur.

2: The CKP bit can be set in software, regardless of the state of the BF bit.

20.5.7.4 Clock Stretching for 10-Bit Slave Transmit Mode

In 10-Bit Slave Transmit mode, clock stretching is controlled during the first two address sequences by the state of the UA bit, just as it is in 10-Bit Slave Receive mode. The first two addresses are followed by a third address sequence, which contains the high-order bits of the 10-bit address and the R/$\overline{W}$ bit set to '1'. After the third address sequence is performed, the UA bit is not set, the module is now configured in Transmit mode and clock stretching is controlled by the BF flag as in 7-Bit Slave Transmit mode (see Figure 20-13).

22.5.2 CLOCKED CONVERSION TRIGGER

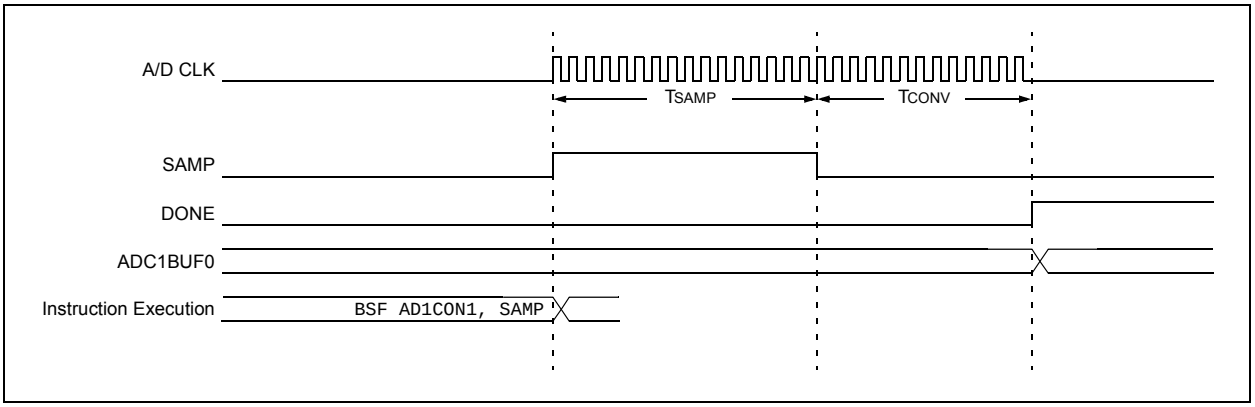
When ADRC = 1, the conversion trigger is under A/D clock control. The SAMCx bits (ADCON3H<4:0>) select the number of TAD clock cycles between the start of sampling and the start of conversion. After the start of sampling, the module will count a number of TAD clocks specified by the SAMCx bits. The SAMCx bits must always be programmed for at least one clock cycle to ensure sampling requirements are met.

EQUATION 22-2: CLOCKED CONVERSION TRIGGER TIME

$$T_{SMP} = SAMC<4:0> * T_{AD}$$

Figure 22-6 shows how to use the clocked conversion trigger with the sampling started by the user software.

FIGURE 22-6: CONVERTING ONE CHANNEL, MANUAL SAMPLE START, TAD-BASED CONVERSION START

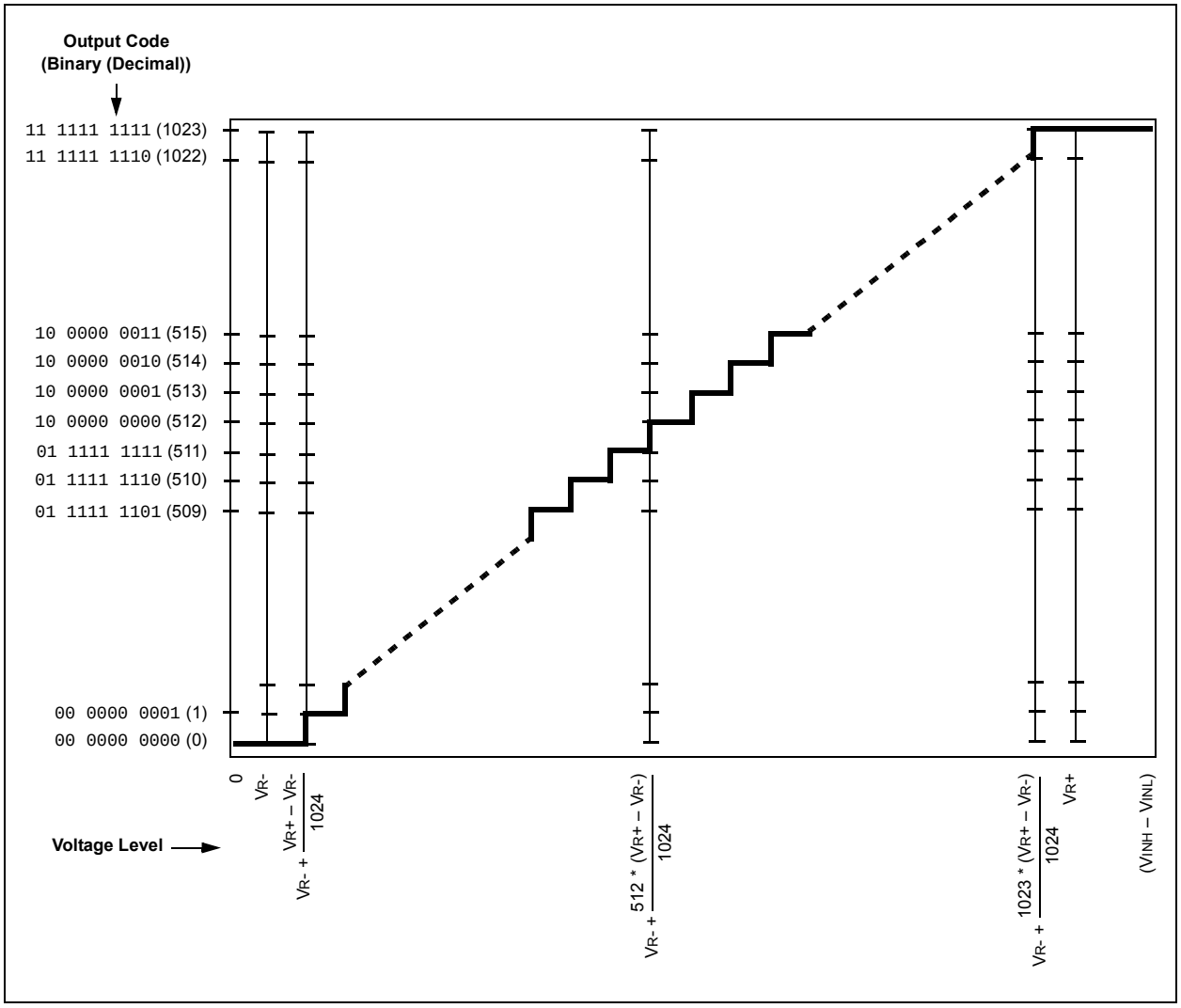


EXAMPLE 22-2: CONVERTING ONE CHANNEL, MANUAL SAMPLE START, TAD-BASED CONVERSION START CODE

```
int ADCValue;

ANCON2  = 0x10;                     // all PORTB = Digital; RB12 = analog
ADCON1L = 0x70;                     // SSRC<2:0> = 111 implies internal counter ends sampling
                                    // and starts converting.
ADCHS0L = 0x0C;                     // Connect AN12 as S/H input.
                                    // in this example AN12 is the input
ADCSS0H = 0;
ADCON3H = 1F;                       // Sample time = 31Tad, Tad = 3Tcy
ADCON3L=02;
ADCON2L = 0;
ADCON1Hbits.ADON = 1;               // turn ADC ON
while (1)                           // repeat continuously
{
    ADCON1Lbits.SAMP = 1;           // start sampling, then after 31Tad go to conversion
    while (!ADCON1Lbits.DONE){};    // conversion done?
    ADCValue = ADCBUF0;             // yes then get ADC value
}                                   // repeat
```

FIGURE 22-22: 10-BIT A/D TRANSFER FUNCTION



22.11 Operation During Sleep and Idle Modes

Sleep and Idle modes are useful for minimizing conversion noise because the digital activity of the CPU, buses and other peripherals is minimized.

22.11.1 CPU SLEEP MODE WITHOUT RC A/D CLOCK

When the device enters Sleep mode, all clock sources to the module are shut down and stay at logic '0'.

If Sleep occurs in the middle of a conversion, the conversion is aborted unless the A/D is clocked from its internal RC clock generator. The converter will not resume a partially completed conversion on exiting from Sleep mode.

Register contents are not affected by the device entering or leaving Sleep mode.

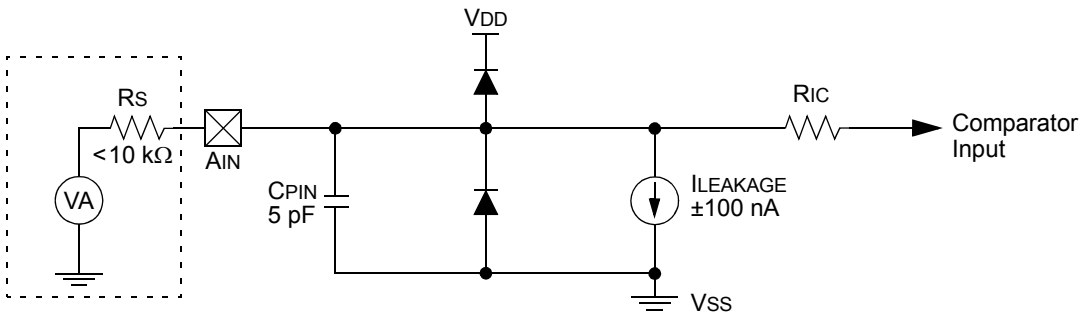
22.11.2 CPU SLEEP MODE WITH RC A/D CLOCK

The A/D module can operate during Sleep mode if the A/D clock source is set to the internal A/D RC oscillator (ADRC = 1). This eliminates digital switching noise from the conversion. When the conversion is completed, the DONE bit will be set and the result is loaded into the A/D Result Buffer n, ADCBUFn.

If the A/D interrupt is enabled (ADIE = 1), the device will wake-up from Sleep when the A/D interrupt occurs. Program execution will resume at the A/D Interrupt Service Routine (ISR). After the ISR completes execution will continue from the instruction after the SLEEP instruction that placed the device in Sleep mode.

If the A/D interrupt is not enabled, the A/D module will then be turned off, although the ADON bit will remain set.

FIGURE 23-3: COMPARATOR ANALOG INPUT MODEL



Legend:

CPIN	=	Input Capacitance
VT	=	Threshold Voltage
ILEAKAGE	=	Leakage Current at the pin due to various junctions
RIC	=	Interconnect Resistance
RS	=	Source Impedance
VA	=	Analog Voltage

23.5 Comparator Control and Configuration

Each comparator has up to eight possible combinations of inputs: up to four external analog inputs and one of two internal voltage references.

All of the comparators allow a selection of the signal from pin, CxINA, or the voltage from the Comparator Reference (CVREF) on the non-inverting channel. This is compared to either CxINB, CxINC, C2IND or the microcontroller's fixed internal reference voltage (VBG, 1.2V nominal) on the inverting channel. The comparator inputs and outputs are tied to fixed I/O pins, defined in Table 23-1. The available comparator configurations and their corresponding bit settings are shown in Figure 23-4.

TABLE 23-1: COMPARATOR INPUTS AND OUTPUTS

Comparator	Input or Output	I/O Pin(†)
1	C1INA (VIN+)	RA5/RF6
	C1INB (VIN-)	RF5
	C1INC (VIN-)	RH6(2)
	C2INB(VIN-)	RF2
	CVREF (VIN+)	RF5
	C1OUT	RPn(1)
2	C2INA (VIN+)	RA5
	C2INB (VIN-)	RF2
	C2INC (VIN-)	RH4(2)
	C2IND (VIN-)	RH5(2)
	CVREF (VIN+)	RF5
	C2OUT	RPn(1)
3	C3INA (VIN+)	RA5/RG2
	C3INB (VIN-)	RG3
	C2INB (VIN-)	RF2
	C3INC (VIN-)	RG4
	CVREF (VIN+)	RF5
	C3OUT	RPn(1)

† The I/O pin is dependent on package type.

Note 1: These pins are remappable I/Os.

2: These pins are not available on 64-pin devices.

FIGURE 30-14: EXAMPLE SPI SLAVE MODE TIMING (CKE = 0)

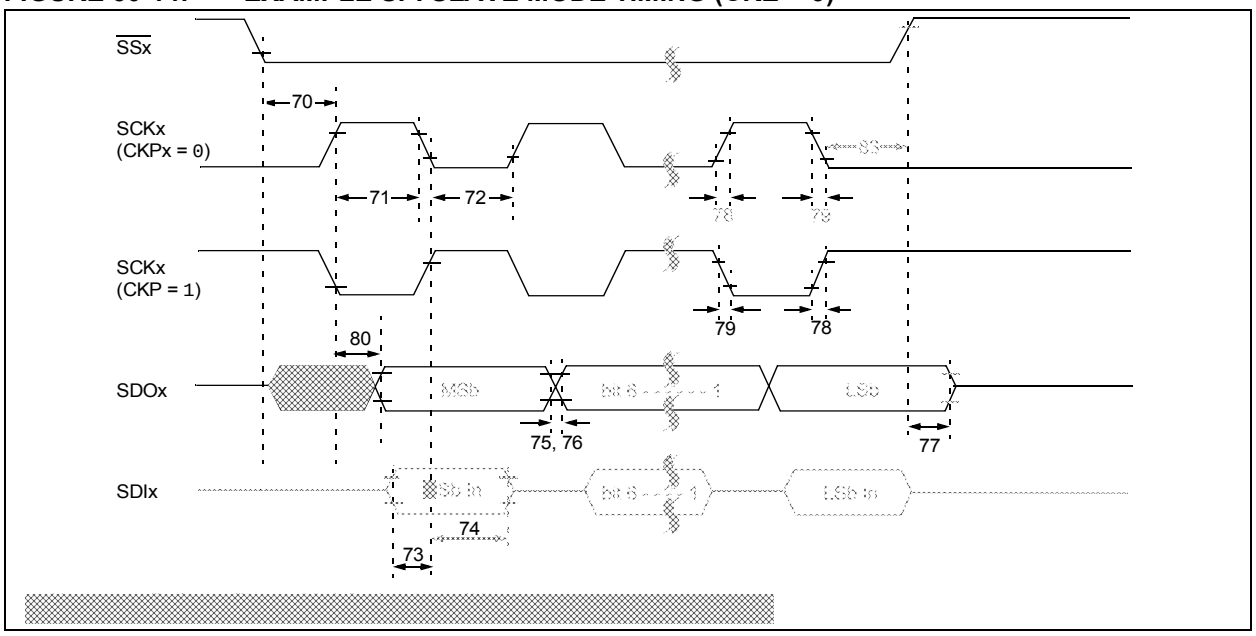


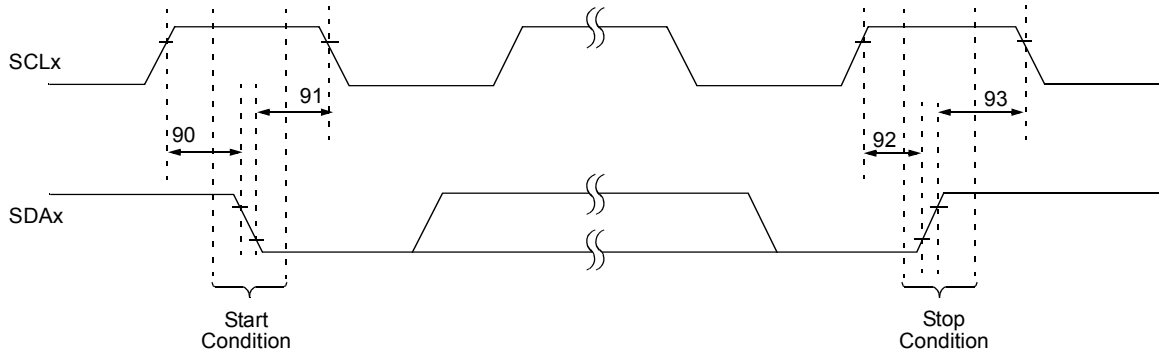
TABLE 30-32: EXAMPLE SPI MODE REQUIREMENTS (SLAVE MODE TIMING, CKE = 0)

Param. No.	Symbol	Characteristic		Min.	Max.	Units	Conditions
70	TssL2scH, TssL2scL	$\overline{SSx}$ ↓ to SCKx ↓ or SCKx ↑ Input		3 Tcy	—	ns	
70A	TssL2WB	$\overline{SSx}$ to write to SSPBUF		3 Tcy	—	ns	
71	TscH	SCKx Input High Time (Slave mode)	Continuous	1.25 Tcy + 30	—	ns	
71A			Single Byte	40	—	ns	**(Note 1)**
72	TscL	SCKx Input Low Time (Slave mode)	Continuous	1.25 Tcy + 30	—	ns	
72A			Single Byte	40	—	ns	**(Note 1)**
73	TdiV2scH, TdiV2scL	Setup Time of SDIx Data Input to SCKx Edge		20	—	ns	
73A	Tb2b	Last Clock Edge of Byte 1 to the First Clock Edge of Byte 2		1.5 Tcy + 40	—	ns	**(Note 2)**
74	TscH2diL, TscL2diL	Hold Time of SDIx Data Input to SCKx Edge		40	—	ns	
75	TdoR	SDOx Data Output Rise Time		—	25	ns	
76	TdoF	SDOx Data Output Fall Time		—	25	ns	
77	TssH2doZ	$\overline{SSx}$ ↑ to SDOx Output High-impedance		10	50	ns	
78	TscR	SCKx Output Rise Time (Master mode)		—	25	ns	
79	TscF	SCKx Output Fall Time (Master mode)		—	25	ns	
80	TscH2doV, TscL2doV	SDOx Data Output Valid after SCKx Edge		—	50	ns	
83	TscH2ssH, TscL2ssH	$\overline{SSx}$ ↑ after SCKx Edge		1.5 Tcy + 40	—	ns	

Note 1: Requires the use of Parameter #73A.

2: Only if Parameter #71A and #72A are used.

FIGURE 30-16: I²C BUS START/STOP BITS TIMING

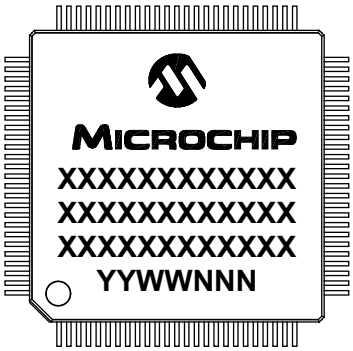


Note: Refer to Figure 30-2 for load conditions.

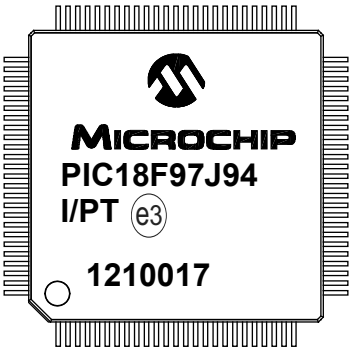
TABLE 30-34: I²C BUS START/STOP BITS REQUIREMENTS (SLAVE MODE)

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	—		

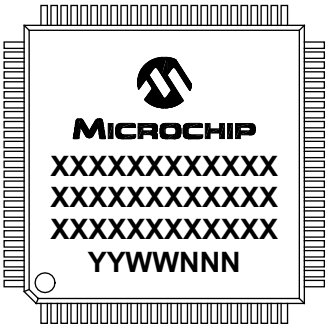
100-Lead TQFP (12x12x1 mm)



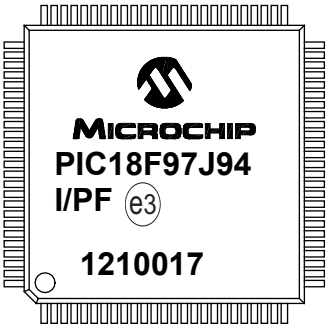
Example



100-Lead TQFP (14x14x1 mm)



Example



Legend:	XX...X	Customer-specific information
	Y	Year code (last digit of calendar year)
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week ‘01’)
	NNN	Alphanumeric traceability code
	(e3)	Pb-free JEDEC® designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC® designator ((e3)) can be found on the outer packaging for this package.
Note:	In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.	