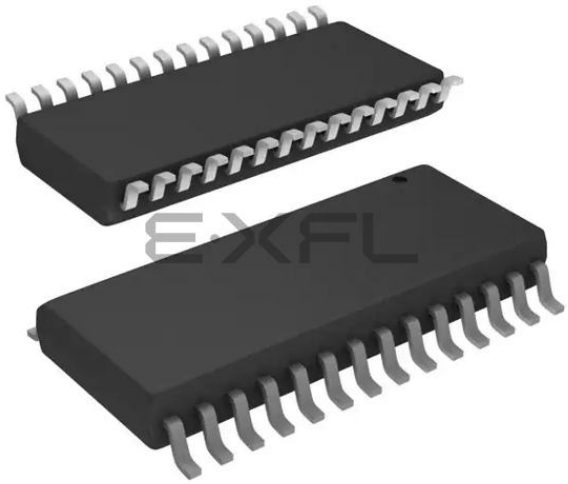


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

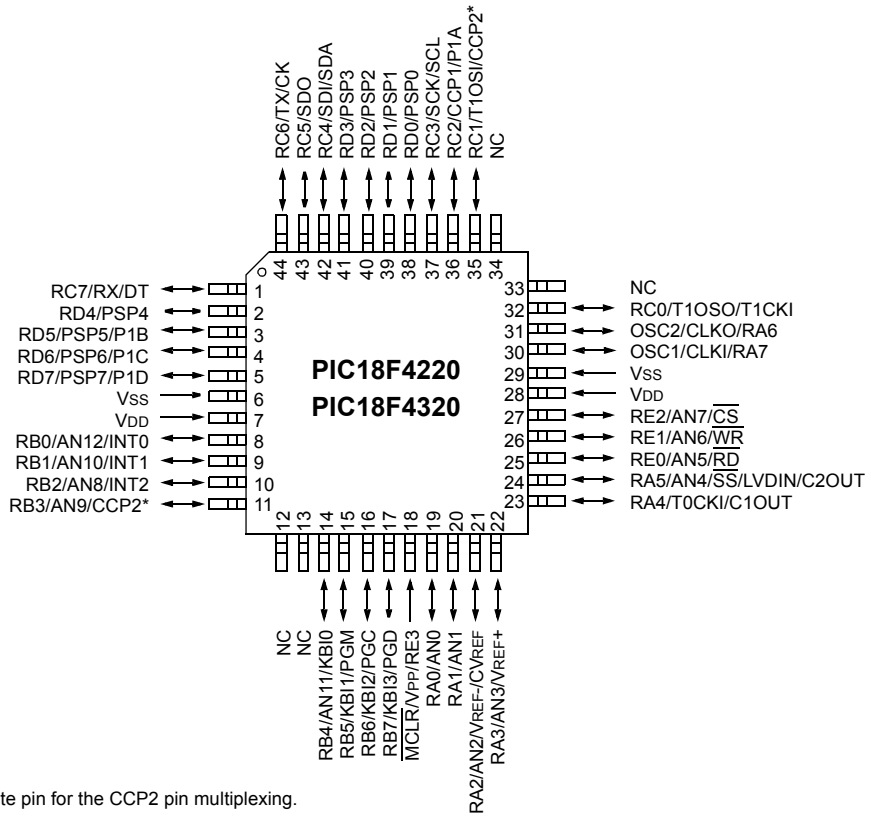
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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	40MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LVD, POR, PWM, WDT
Number of I/O	25
Program Memory Size	8KB (4K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	4.2V ~ 5.5V
Data Converters	A/D 10x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic18f2320t-i-so

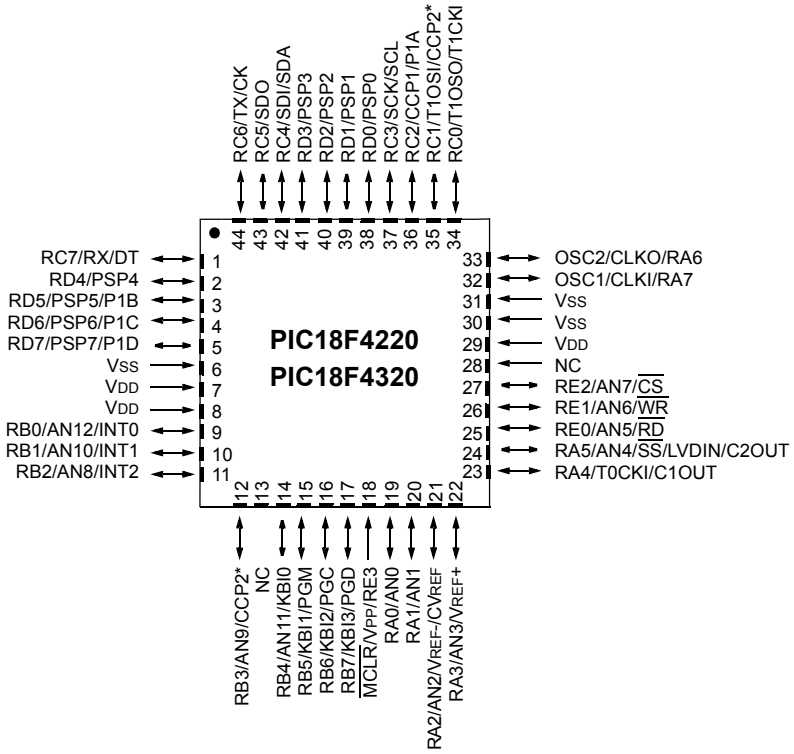
Pin Diagrams (Cont.'d)

44-Pin TQFP



* RB3 is the alternate pin for the CCP2 pin multiplexing.

44-Pin QFN



* RB3 is the alternate pin for the CCP2 pin multiplexing.

TABLE 1-3: PIC18F4220/4320 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	PDIP	TQFP	QFN			
						PORTD is a bidirectional I/O port or a Parallel Slave Port (PSP) for interfacing to a microprocessor port. These pins have TTL input buffers when PSP module is enabled.
RD0/PSP0	19	38	38			
RD0				I/O	ST	Digital I/O.
PSP0				I/O	TTL	Parallel Slave Port data.
RD1/PSP1	20	39	39			
RD1				I/O	ST	Digital I/O.
PSP1				I/O	TTL	Parallel Slave Port data.
RD2/PSP2	21	40	40			
RD2				I/O	ST	Digital I/O.
PSP2				I/O	TTL	Parallel Slave Port data.
RD3/PSP3	22	41	41			
RD3				I/O	ST	Digital I/O.
PSP3				I/O	TTL	Parallel Slave Port data.
RD4/PSP4	27	2	2			
RD4				I/O	ST	Digital I/O.
PSP4				I/O	TTL	Parallel Slave Port data.
RD5/PSP5/P1B	28	3	3			
RD5				I/O	ST	Digital I/O.
PSP5				I/O	TTL	Parallel Slave Port data.
P1B				O	—	Enhanced CCP1 output.
RD6/PSP6/P1C	29	4	4			
RD6				I/O	ST	Digital I/O.
PSP6				I/O	TTL	Parallel Slave Port data.
P1C				O	—	Enhanced CCP1 output.
RD7/PSP7/P1D	30	5	5			
RD7				I/O	ST	Digital I/O.
PSP7				I/O	TTL	Parallel Slave Port data.
P1D				O	—	Enhanced CCP1 output.

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

Note 1: Alternate assignment for CCP2 when CCP2MX is cleared.
2: Default assignment for CCP2 when CCP2MX (CONFIG3H<0>) is set.

TABLE 1-3: PIC18F4220/4320 PINOUT I/O DESCRIPTIONS (CONTINUED)

Pin Name	Pin Number			Pin Type	Buffer Type	Description
	PDIP	TQFP	QFN			
						PORTE is a bidirectional I/O port.
RE0/AN5/$\overline{RD}$	8	25	25			
RE0				I/O	ST	Digital I/O.
AN5				I	Analog	Analog input 5.
$\overline{RD}$				I	TTL	Read control for Parallel Slave Port (see also $\overline{WR}$ and $\overline{CS}$ pins).
RE1/AN6/$\overline{WR}$	9	26	26			
RE1				I/O	ST	Digital I/O.
AN6				I	Analog	Analog input 6.
$\overline{WR}$				I	TTL	Write control for Parallel Slave Port (see $\overline{CS}$ and $\overline{RD}$ pins).
RE2/AN7/$\overline{CS}$	10	27	27			
RE2				I/O	ST	Digital I/O.
AN7				I	Analog	Analog input 7.
$\overline{CS}$				I	TTL	Chip select control for Parallel Slave Port (see related $\overline{RD}$ and $\overline{WR}$).
RE3	1	18	18	—	—	See $\overline{MCLR}$/VPP/RE3 pin.
VSS	12, 31	6, 29	6, 30, 31	P	—	Ground reference for logic and I/O pins.
VDD	11, 32	7, 28	7, 8 29	P	—	Positive supply for logic and I/O pins.
NC	—	—	13, 28	NC	NC	No connect.

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

Note 1: Alternate assignment for CCP2 when CCP2MX is cleared.

2: Default assignment for CCP2 when CCP2MX (CONFIG3H<0>) is set.

TABLE 3-3: ACTIVITY AND EXIT DELAY ON WAKE-UP FROM SLEEP MODE OR ANY IDLE MODE (BY CLOCK SOURCES)

Clock in Power-Managed Mode	Primary System Clock	Power-Managed Mode Exit Delay	Clock Ready Status Bit (OSCCON)	Activity During Wake-up from Power-Managed Mode	
				Exit by Interrupt	Exit by Reset
Primary System Clock (PRI_IDLE mode)	LP, XT, HS	5-10 μs(5)	OSTS	CPU and peripherals clocked by primary clock and executing instructions.	Not clocked or Two-Speed Start-up (if enabled)(3).
	HSPLL				
	EC, RC, INTRC(1)		—		
	INTOSC(2)		IOFS		
T1OSC or INTRC(1)	LP, XT, HS	OST	OSTS	CPU and peripherals clocked by selected power-managed mode clock and executing instructions until primary clock source becomes ready.	
	HSPLL	OST + 2 ms			
	EC, RC, INTRC(1)	5-10 μs(5)	—		
	INTOSC(2)	5-10 μs(4)	IOFS		
INTOSC(2)	LP, XT, HS	OST	OSTS		
	HSPLL	OST + 2 ms			
	EC, RC, INTRC(1)	5-10 μs(5)	—		
	INTOSC(2)	None	IOFS		
Sleep mode	LP, XT, HS	OST	OSTS	Not clocked or Two-Speed Start-up (if enabled) until primary clock source becomes ready(3).	
	HSPLL	OST + 2 ms			
	EC, RC, INTRC(1)	5-10 μs(5)	—		
	INTOSC(2)	5-10 μs(4)	IOFS		

Note 1: In this instance, refers specifically to the INTRC clock source.

2: Includes both the INTOSC 8 MHz source and postscaler derived frequencies.

3: Two-Speed Start-up is covered in greater detail in **Section 23.3 "Two-Speed Start-up"**.

4: Execution continues during the INTOSC stabilization period.

5: Required delay when waking from Sleep and all Idle modes. This delay runs concurrently with any other required delays (see **Section 3.3 "Idle Modes"**).

5.10 Access Bank

The Access Bank is an architectural enhancement which is very useful for C compiler code optimization. The techniques used by the C compiler may also be useful for programs written in assembly.

This data memory region can be used for:

- Intermediate computational values
- Local variables of subroutines
- Faster context saving/switching of variables
- Common variables
- Faster evaluation/control of SFRs (no banking)

The Access Bank is comprised of the last 128 bytes in Bank 15 (SFRs) and the first 128 bytes in Bank 0. These two sections will be referred to as Access RAM High and Access RAM Low, respectively. Figure 5-6 indicates the Access RAM areas.

A bit in the instruction word specifies if the operation is to occur in the bank specified by the BSR register or in the Access Bank. This bit is denoted as the 'a' bit (for access bit).

When forced in the Access Bank (a = 0), the last address in Access RAM Low is followed by the first address in Access RAM High. Access RAM High maps the Special Function Registers, so these registers can be accessed without any software overhead. This is useful for testing status flags and modifying control bits.

5.11 Bank Select Register (BSR)

The need for a large general purpose memory space dictates a RAM banking scheme. The data memory is partitioned into as many as sixteen banks. When using direct addressing, the BSR should be configured for the desired bank.

BSR<3:0> holds the upper 4 bits of the 12-bit RAM address. The BSR<7:4> bits will always read '0's and writes will have no effect (see Figure 5-7).

A MOVLB instruction has been provided in the instruction set to assist in selecting banks.

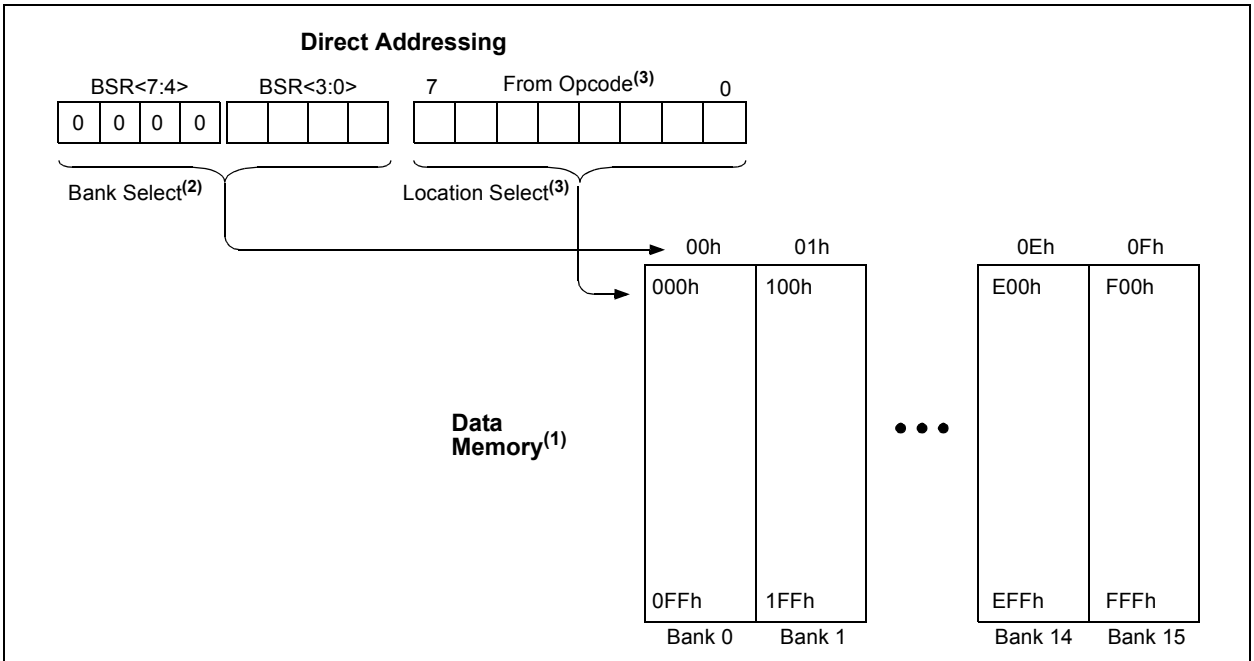
If the currently selected bank is not implemented, any read will return all '0's and all writes are ignored. The STATUS register bits will be set/cleared as appropriate for the instruction performed.

Each Bank extends up to FFh (256 bytes). All data memory is implemented as static RAM.

A MOVFF instruction ignores the BSR since the 12-bit addresses are embedded into the instruction word.

Section 5.12 "Indirect Addressing, INDF and FSR Registers" provides a description of indirect addressing which allows linear addressing of the entire RAM space.

FIGURE 5-7: DIRECT ADDRESSING



Note 1: For register file map detail, see Table 5-1.
2: The access bit of the instruction can be used to force an override of the selected bank (BSR<3:0>) to the registers of the Access Bank.
3: The MOVFF instruction embeds the entire 12-bit address in the instruction.

5.14 RCON Register

The Reset Control (RCON) register contains flag bits that allow differentiation between the sources of a device Reset. These flags include the $\overline{TO}$, $\overline{PD}$, $\overline{POR}$, $\overline{BOR}$ and $\overline{RI}$ bits. This register is readable and writable.

Note 1: If the BOREN Configuration bit is set (Brown-out Reset enabled), the $\overline{BOR}$ bit is '1' on a Power-on Reset. After a Brown-out Reset has occurred, the $\overline{BOR}$ bit will be cleared and must be set by firmware to indicate the occurrence of the next Brown-out Reset.

2: It is recommended that the $\overline{POR}$ bit be set after a Power-on Reset has been detected so that subsequent Power-on Resets may be detected.

REGISTER 5-3: RCON: RESET CONTROL REGISTER

R/W-0	U-0	U-0	R/W-1	R-1	R-1	R/W-0	R/W-0
IPEN	—	—	$\overline{RI}$	$\overline{TO}$	$\overline{PD}$	$\overline{POR}$	$\overline{BOR}$
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 **IPEN:** Interrupt Priority Enable bit
- 1 = Enable priority levels on interrupts
- 0 = Disable priority levels on interrupts (PIC16CXXX Compatibility mode)

bit 6-5 **Unimplemented:** Read as '0'

bit 4 **$\overline{RI}$:** RESET Instruction Flag bit
- 1 = The RESET instruction was not executed (set by firmware only)
- 0 = The RESET instruction was executed causing a device Reset (must be set in software after a Brown-out Reset occurs)

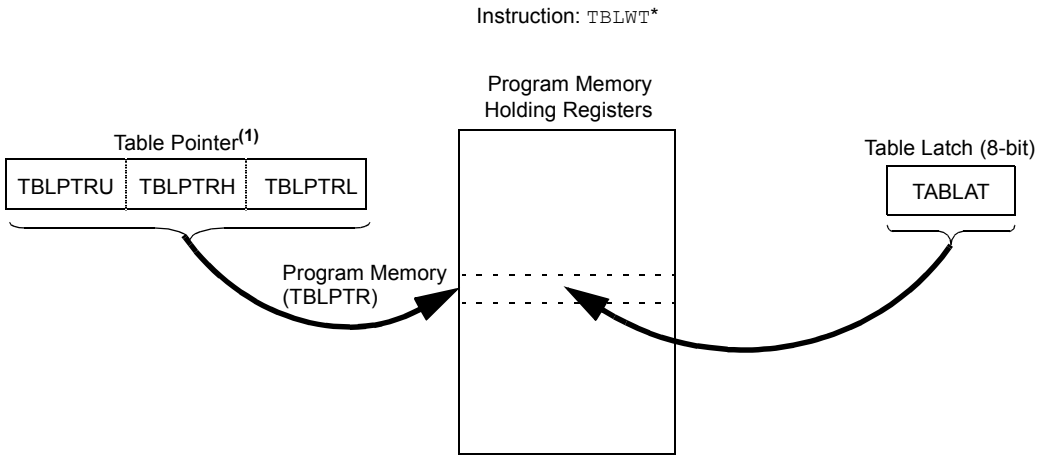
bit 3 **$\overline{TO}$:** Watchdog Time-out Flag bit
- 1 = Set by power-up, CLRWDT instruction or SLEEP instruction
- 0 = A WDT time-out occurred

bit 2 **$\overline{PD}$:** Power-down Detection Flag bit
- 1 = Set by power-up or by the CLRWDT instruction
- 0 = Cleared by execution of the SLEEP instruction

bit 1 **$\overline{POR}$:** Power-on Reset Status bit
- 1 = A Power-on Reset has not occurred (set by firmware only)
- 0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

bit 0 **$\overline{BOR}$:** Brown-out Reset Status bit
- 1 = A Brown-out Reset has not occurred (set by firmware only)
- 0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

FIGURE 6-2: TABLE WRITE OPERATION



Note 1: Table Pointer actually points to one of eight holding registers, the address of which is determined by TBLPTRL<2:0>. The process for physically writing data to the program memory array is discussed in **Section 6.5 "Writing to Flash Program Memory"**.

6.2 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

6.2.1 EECON1 AND EECON2 REGISTERS

EECON1 is the control register for memory accesses.

EECON2 is not a physical register. Reading EECON2 will read all '0's. The EECON2 register is used exclusively in the memory write and erase sequences.

Control bit, EEPGD, determines if the access will be to program or data EEPROM memory. When clear, operations will access the data EEPROM memory. When set, program memory is accessed.

Control bit, CFGS, determines if the access will be to the Configuration registers or to program memory/data EEPROM memory. When set, subsequent operations access Configuration registers. When CFGS is clear, the EEPGD bit selects either program Flash or data EEPROM memory.

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

The WREN bit enables and disables erase and write operations. When set, erase and write operations are allowed. When clear, erase and write operations are disabled – the WR bit cannot be set while the WREN bit is clear. This process helps to prevent accidental writes to memory due to errant (unexpected) code execution.

Firmware should keep the WREN bit clear at all times except when starting erase or write operations. Once firmware has set the WR bit, the WREN bit may be cleared. Clearing the WREN bit will not affect the operation in progress.

The WRERR bit is set when a write operation is interrupted by a Reset. In these situations, the user can check the WRERR bit and rewrite the location. It will be necessary to reload the data and address registers (EEDATA and EEADR) as these registers have cleared as a result of the Reset.

Control bits, RD and WR, start read and erase/write operations, respectively. These bits are set by firmware and cleared by hardware at the completion of the operation.

The RD bit cannot be set when accessing program memory (EEPGD = 1). Program memory is read using table read instructions. See **Section 6.3 "Reading the Flash Program Memory"** regarding table reads.

Note:	Interrupt flag bit, EEIF in the PIR2 register, is set when the write is complete. It must be cleared in software.

NOTES:

NOTES:

17.3.3 ENABLING SPI I/O

To enable the serial port, MSSP Enable bit, SSPEN (SSPCON1<5>), must be set. To reset or reconfigure SPI mode, clear the SSPEN bit, re-initialize the SSPCON registers and then set the SSPEN bit. This configures the SDI, SDO, SCK and $\overline{SS}$ 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. That is:

- SDI must have TRISC<4> bit cleared
- SDO must have TRISC<5> bit cleared
- SCK (Master mode) must have TRISC<3> bit cleared
- SCK (Slave mode) must have TRISC<3> bit set
- $\overline{SS}$ must have TRISA<5> bit set

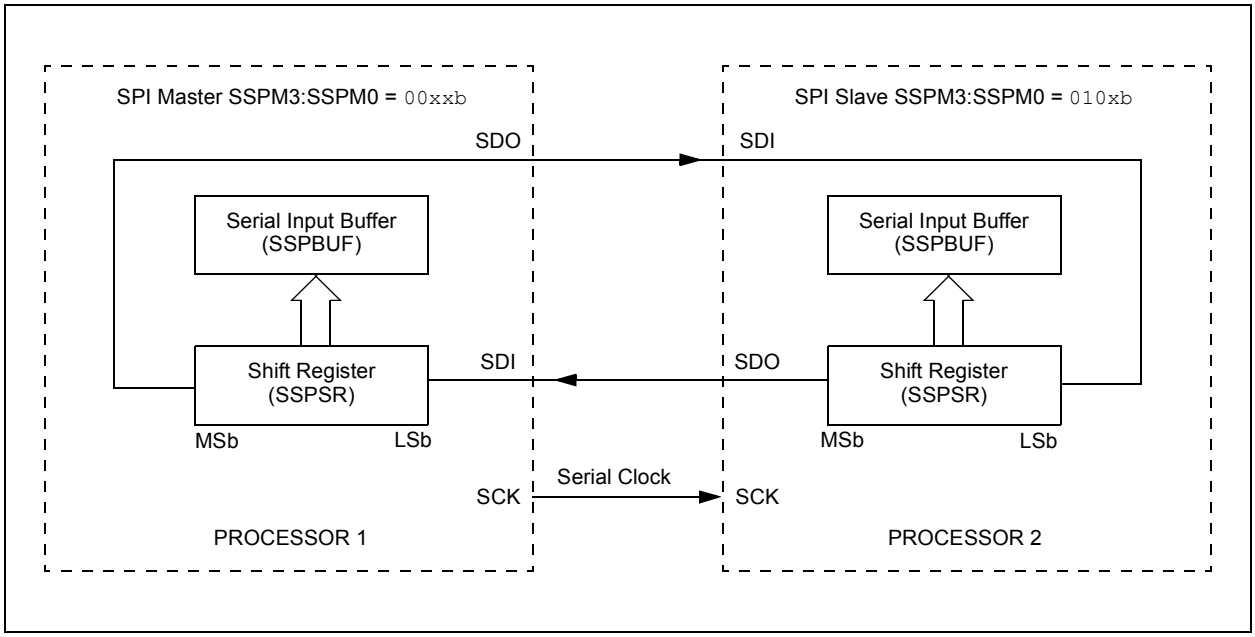
Any serial port function that is not desired may be overridden by programming the corresponding data direction (TRIS) register to the opposite value.

17.3.4 TYPICAL CONNECTION

Register 17-2 shows a typical connection between two microcontrollers. The master controller (Processor 1) initiates the data transfer by sending the SCK 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 17-2: SPI MASTER/SLAVE CONNECTION



17.3.6 SLAVE MODE

In Slave mode, the data is transmitted and received as the external clock pulses appear on SCK. When the last bit is latched, the SSPIF interrupt flag bit is set.

While in Slave mode, the external clock is supplied by the external clock source on the SCK pin. This external clock must meet the minimum high and low times as specified in the electrical specifications.

While in power-managed modes, the slave can transmit/receive data. When a byte is received, the device will wake-up from power-managed modes.

17.3.7 SLAVE SELECT CONTROL

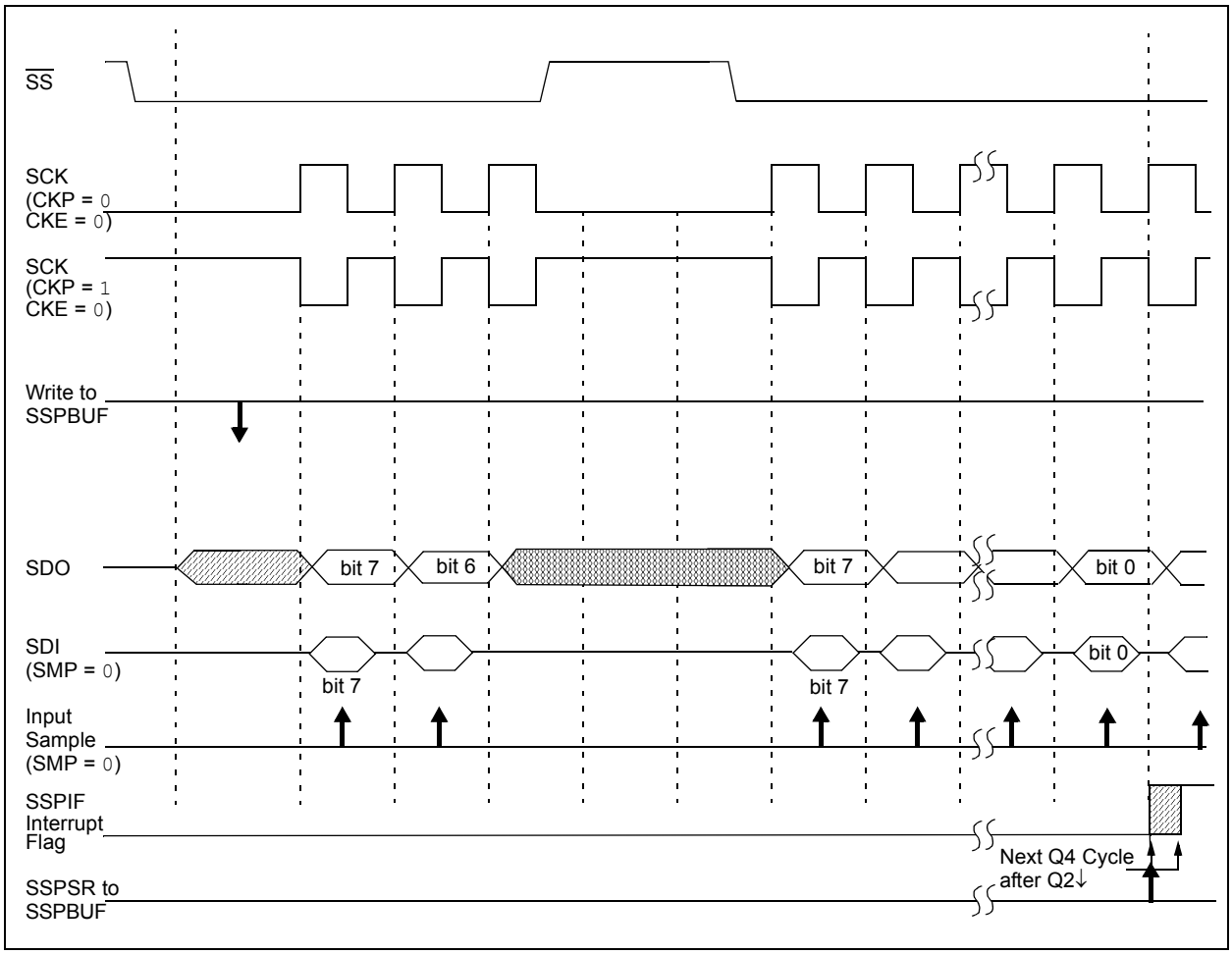
The $\overline{SS}$ pin allows a master controller to select one of several slave controllers for communications in systems with more than one slave. The SPI must be in Slave mode with $\overline{SS}$ pin control enabled (SSPCON1<3:0> = 04h). The $\overline{SS}$ pin is configured for input by setting TRISA<5>. When the $\overline{SS}$ pin is low, transmission and reception are enabled and the SDO pin is driven. When the $\overline{SS}$ pin goes high, the SDO pin is tri-stated, even if in the middle of a transmitted byte. External pull-up/pull-down resistors may be desirable, depending on the application.

Note 1:	When the SPI is in Slave mode with $\overline{SS}$ pin control enabled (SSPCON1<3:0> = 0100), the SPI module will reset when the $\overline{SS}$ pin is set high.
2:	If the SPI is used in Slave mode with CKE set, then the $\overline{SS}$ pin control must be enabled.

When the SPI module resets, SSPSR is cleared. This can be done by either driving the $\overline{SS}$ pin to a high level or clearing the SSPEN bit.

To emulate two-wire communication, the SDO pin can be connected to the SDI pin. When the SPI needs to operate as a receiver the SDO pin can be configured as an input. This disables transmissions from the SDO. The SDI can always be left as an input (SDI function) since it cannot create a bus conflict.

FIGURE 17-4: SLAVE SYNCHRONIZATION WAVEFORM



REGISTER 17-3: SSPSTAT: MSSP STATUS REGISTER (I²C™ MODE)

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/$\overline{A}$	P(1)	S(2)	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:** Slew Rate Control bit

In Master or Slave mode:
1 = Slew rate control disabled
0 = Slew rate control enabled

bit 6 **CKE:** SMBus Select bit

In Master or Slave mode:
1 = Enable SMBus specific inputs
0 = Disable SMBus specific inputs

bit 5 **D/$\overline{A}$:** Data/$\overline{\text{Address}}$ bit

In Master mode:
Reserved.

In Slave mode:
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(1)

1 = Indicates that a Stop bit has been detected last
0 = Stop bit was not detected last

bit 3 **S:** Start bit(2)

1 = Indicates that a Start bit has been detected last
0 = Start bit was not detected last

bit 2 **R/$\overline{W}$:** Read/$\overline{\text{Write}}$ bit Information (I²C mode only)

In Slave mode:(3)
1 = Read
0 = Write

In Master mode:(4)
1 = Transmit is in progress
0 = Transmit is not in progress

bit 1 **UA:** Update Address bit (10-Bit Slave 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

In Transmit mode:
1 = Data transmit in progress (does not include the $\overline{\text{ACK}}$ and Stop bits), SSPBUF is full
0 = Data transmit complete (does not include the $\overline{\text{ACK}}$ and Stop bits), SSPBUF is empty

In Receive mode:
1 = Receive complete, SSPBUF is full
0 = Receive not complete, SSPBUF is empty

Note 1: This bit is cleared on Reset when SSPEN is cleared or a Start bit has been detected.

2: This bit is cleared on Reset when SSPEN is cleared or a Stop bit has been detected.

3: 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 $\overline{\text{ACK}}$ bit.

4: ORing this bit with the SSPCON2 bits, SEN, RSEN, PEN, RCEN or ACKEN will indicate if the MSSP is in Idle mode.

FIGURE 18-2: ASYNCHRONOUS TRANSMISSION

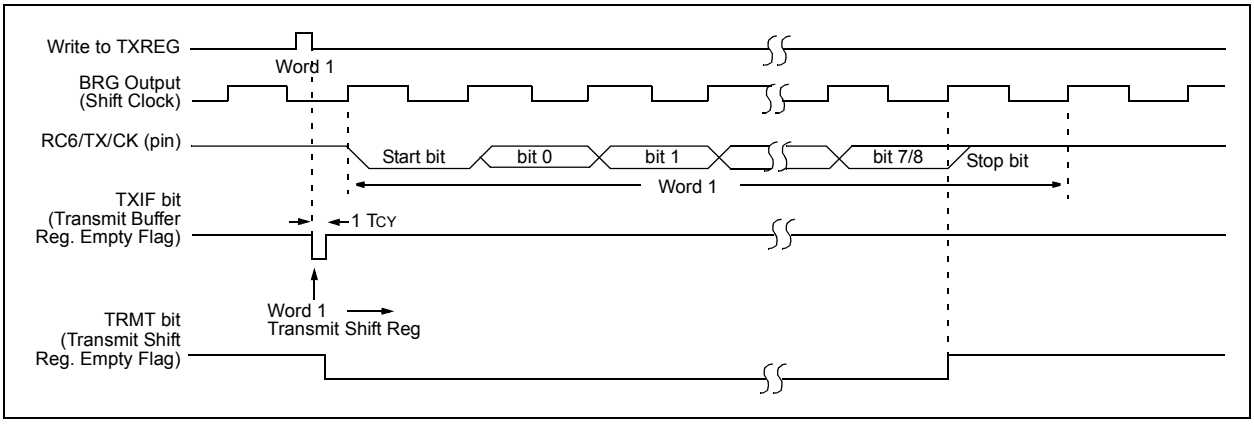
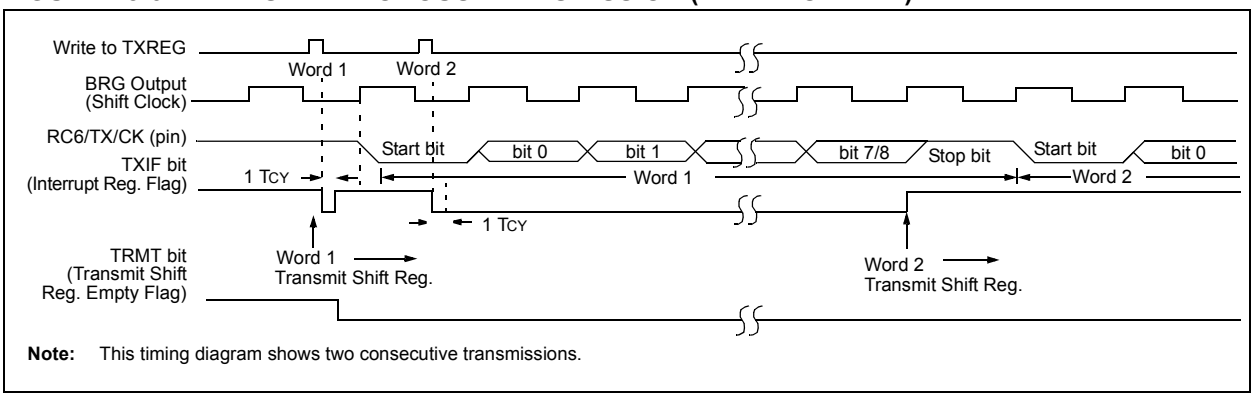


FIGURE 18-3: ASYNCHRONOUS TRANSMISSION (BACK TO BACK)



Note: This timing diagram shows two consecutive transmissions.

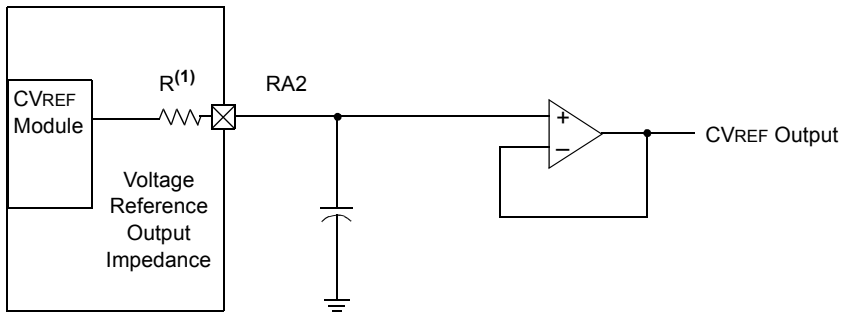
TABLE 18-6: REGISTERS ASSOCIATED WITH ASYNCHRONOUS TRANSMISSION

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	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
IPR1	PSPIP(1)	ADIP	RCIP	TXIP	SSPIP	CCP1IP	TMR2IP	TMR1IP	1111 1111	1111 1111
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
TXREG	USART Transmit Register								0000 0000	0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
SPBRG	Baud Rate Generator Register								0000 0000	0000 0000

Legend: x = unknown, - = unimplemented locations read as '0'. Shaded cells are not used for asynchronous transmission.

Note 1: The PSPIF, PSPIE and PSPIP bits are reserved on the PIC18F2X20 devices; always maintain these bits clear.

FIGURE 21-2: VOLTAGE REFERENCE OUTPUT BUFFER EXAMPLE



Note 1: R is dependent upon the voltage reference configuration bits (CVRCON<3:0> and CVRCON<5>).

TABLE 21-1: REGISTERS ASSOCIATED WITH COMPARATOR VOLTAGE REFERENCE

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR	Value on all other Resets
CVRCON	CVREN	CVROE	CVRR	—	CVR3	CVR2	CVR1	CVR0	000- 0000	000- 0000
CMCON	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0111	0000 0111
TRISA	RA7(1)	RA6(1)	RA5	RA4	RA3	RA2	RA1	RA0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented, read as '0'.
Shaded cells are not used with the comparator voltage reference.

Note 1: These pins are enabled based on oscillator configuration (see Configuration Register 1H).

22.2 Operation

Depending on the power source for the device voltage, the voltage normally decreases relatively slowly. This means that the LVD module does not need to be constantly operating. To decrease the current requirements, the LVD circuitry only needs to be enabled for short periods where the voltage is checked. After doing the check, the LVD module may be disabled.

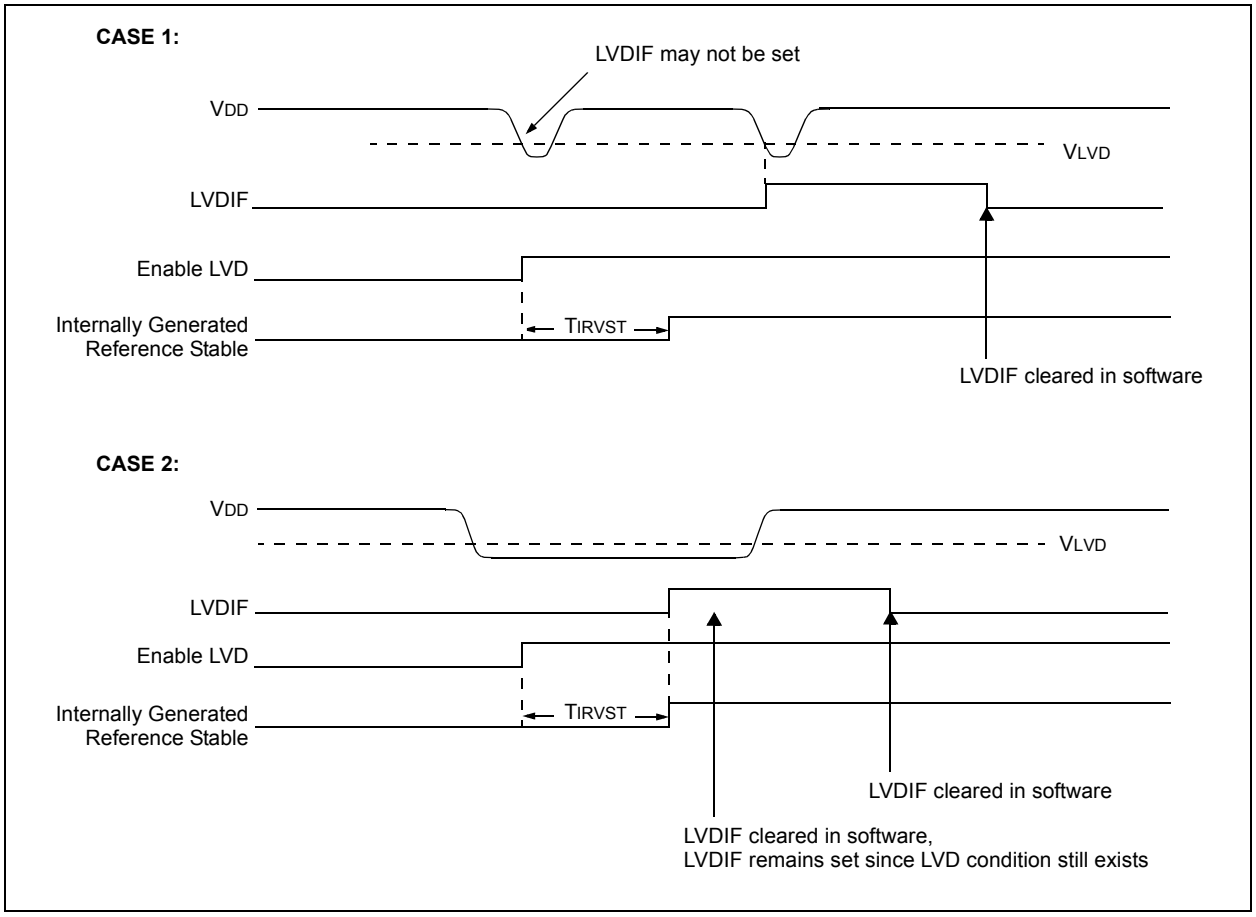
Each time that the LVD module is enabled, the circuitry requires some time to stabilize. After the circuitry has stabilized, all status flags may be cleared. The module will then indicate the proper state of the system.

The following steps are needed to set up the LVD module:

1. Write the value to the LVDL3:LVDL0 bits (LVDCON register) which selects the desired LVD trip point.
2. Ensure that LVD interrupts are disabled (the LVDIE bit is cleared or the GIE bit is cleared).
3. Enable the LVD module (set the LVDEN bit in the LVDCON register).
4. Wait for the LVD module to stabilize (the IRVST bit to become set).
5. Clear the LVD interrupt flag, which may have falsely become set, until the LVD module has stabilized (clear the LVDIF bit).
6. Enable the LVD interrupt (set the LVDIE and the GIE bits).

Figure 22-4 shows typical waveforms that the LVD module may be used to detect.

FIGURE 22-4: LOW-VOLTAGE DETECT WAVEFORMS



BZ **Branch if Zero**

Syntax: [*label*] BZ n

Operands: -128 ≤ n ≤ 127

Operation: if Zero bit is '1',
(PC) + 2 + 2n → PC

Status Affected: None

Encoding:

1110	0000	nnnn	nnnn

Description: If the Zero 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 BZ Jump`

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

CALL **Subroutine Call**

Syntax: [*label*] CALL k [,s]

Operands: 0 ≤ k ≤ 1048575
s ∈ [0,1]

Operation: (PC) + 4 → TOS,
k → PC<20:1>;
if s = 1,
(W) → WS,
(STATUS) → STATUSS,
(BSR) → BSRS

Status Affected: None

Encoding:

1st word (k<7:0>)	1110	110s	$k_7$kkk	kkkk$_0$
2nd word(k<19:8>)	1111	k_{19}kkk	kkkk	kkkk$_8$

Description: Subroutine call of entire 2 Mbyte memory range. First, return address (PC + 4) is pushed onto the return stack. If 's' = 1, the W, STATUS and BSR registers are also pushed into their respective shadow registers, WS, STATUSS and BSRS. If 's' = 0, no update occurs (default). Then, the 20-bit value 'k' is loaded into PC<20:1>. CALL is a two-cycle instruction.

Words: 2

Cycles: 2

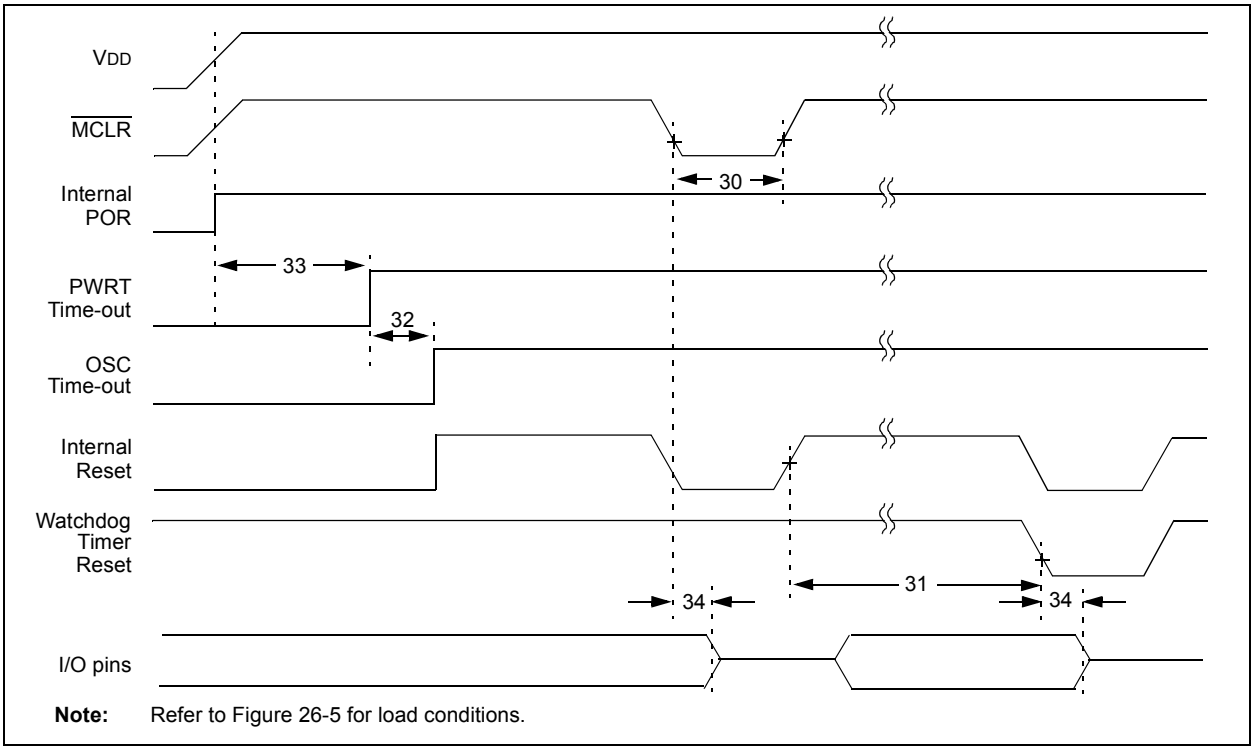
Q Cycle Activity:

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

Example: `HERE CALL THERE,FAST`

Before Instruction
PC = address `(HERE)`
After Instruction
PC = address `(THERE)`
TOS = address `(HERE + 4)`
WS = W
BSRS = BSR
STATUSS= STATUS

FIGURE 26-8: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING



Note: Refer to Figure 26-5 for load conditions.

FIGURE 26-9: BROWN-OUT RESET TIMING

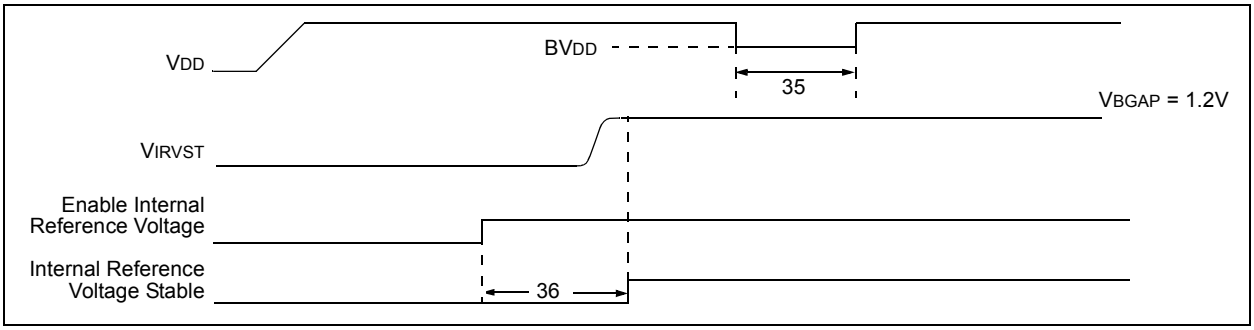
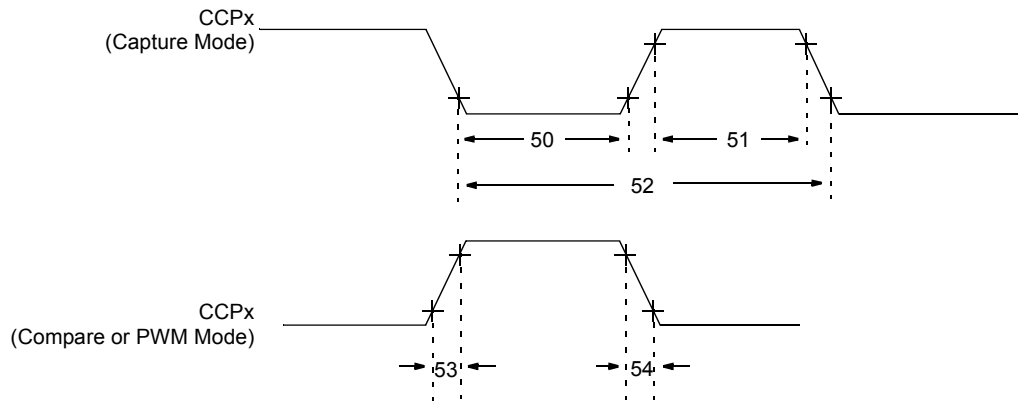


TABLE 26-11: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

Param No.	Symbol	Characteristic			Min	Max	Units	Conditions
40	TT0H	T0CKI High Pulse Width		No prescaler	0.5 TCY + 20	—	ns	
				With prescaler	10	—	ns	
41	TT0L	T0CKI Low Pulse Width		No prescaler	0.5 TCY + 20	—	ns	
				With prescaler	10	—	ns	
42	TT0P	T0CKI Period		No prescaler	TCY + 10	—	ns	
				With prescaler	Greater of: 20 ns or $\frac{T_{CY} + 40}{N}$	—	ns	N = prescale value (1, 2, 4,..., 256)
45	TT1H	T1CKI High Time	Synchronous, no prescaler		0.5 TCY + 20	—	ns	
			Synchronous, with prescaler	PIC18**F**XX20	10	—	ns	
				PIC18**LF**XX20	25	—	ns	
			Asynchronous	PIC18**F**XX20	30	—	ns	
				PIC18**LF**XX20	50	—	ns	
46	TT1L	T1CKI Low Time	Synchronous, no prescaler		0.5 TCY + 5	—	ns	
			Synchronous, with prescaler	PIC18**F**XX20	10	—	ns	
				PIC18**LF**XX20	25	—	ns	
			Asynchronous	PIC18**F**XX20	30	—	ns	
				PIC18**LF**XX20	50	—	ns	
47	TT1P	T1CKI Input Period	Synchronous		Greater of: 20 ns or $\frac{T_{CY} + 40}{N}$	—	ns	N = prescale value (1, 2, 4, 8)
			Asynchronous		60	—	ns	
	FT1	T1CKI Oscillator Input Frequency Range			DC	50	kHz	
48	TCKE2TMRI	Delay from External T1CKI Clock Edge to Timer Increment			2 TOSC	7 TOSC	—	

FIGURE 26-11: CAPTURE/COMPARE/PWM TIMINGS (ALL CCP MODULES)



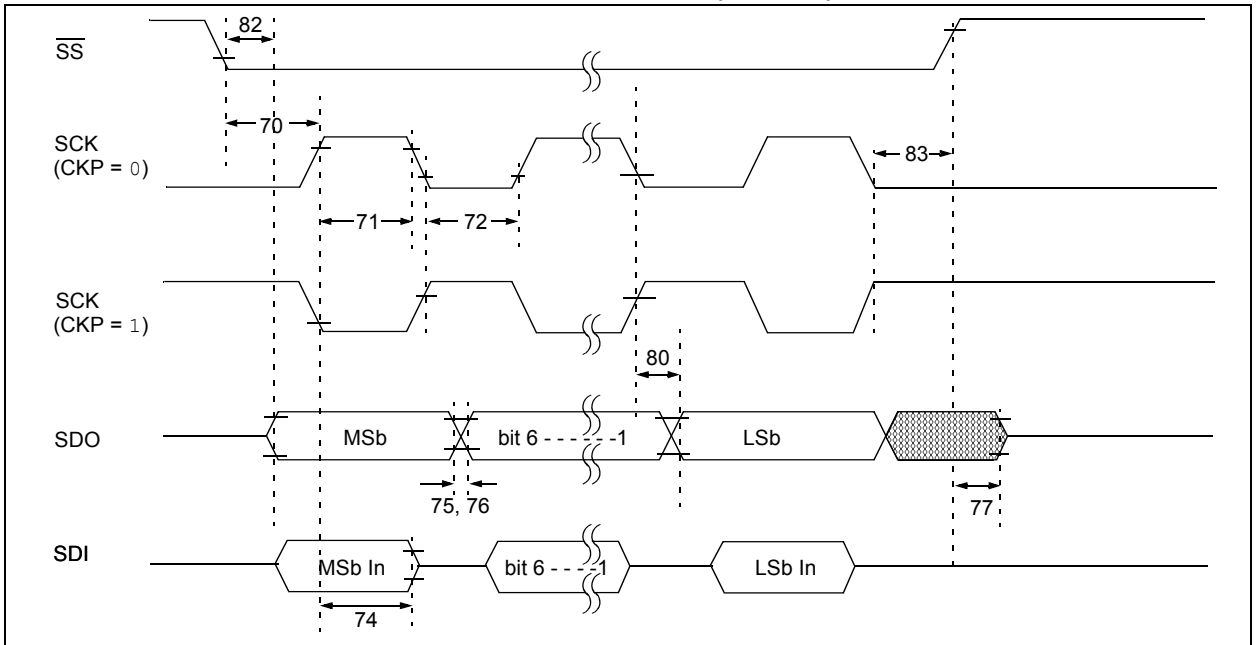
Note: Refer to Figure 26-5 for load conditions.

TABLE 26-16: EXAMPLE SPI MODE REQUIREMENTS (SLAVE MODE TIMING, CKE = 0)

Param No.	Symbol	Characteristic		Min	Max	Units	Conditions
70	TssL2scH, TssL2scL	$\overline{SS}$ ↓ to SCK ↓ or SCK ↑ Input		Tcy	—	ns	
71	TscH	SCK Input High Time (Slave mode)	Continuous	1.25 Tcy + 30	—	ns	
71A			Single Byte	40	—	ns	**(Note 1)**
72	TscL	SCK Input Low Time (Slave mode)	Continuous	1.25 Tcy + 30	—	ns	
72A			Single Byte	40	—	ns	**(Note 1)**
73	TdiV2scH, TdiV2scL	Setup Time of SDI Data Input to SCK Edge		100	—	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 SDI Data Input to SCK Edge		100	—	ns	
75	TdoR	SDO Data Output Rise Time	PIC18**F**XX20	—	25	ns	
			PIC18**LF**XX20		45	ns	
76	TdoF	SDO Data Output Fall Time		—	25	ns	
77	TssH2doZ	$\overline{SS}$ ↑ to SDO Output High-Impedance		10	50	ns	
78	TscR	SCK Output Rise Time (Master mode)	PIC18**F**XX20	—	25	ns	
			PIC18**LF**XX20		45	ns	
79	TscF	SCK Output Fall Time (Master mode)		—	25	ns	
80	TscH2doV, TscL2doV	SDO Data Output Valid after SCK Edge	PIC18**F**XX20	—	50	ns	
			PIC18**LF**XX20		100	ns	
83	TscH2ssH, TscL2ssH	$\overline{SS}$ ↑ after SCK Edge		1.5 Tcy + 40	—	ns	

Note 1: Requires the use of Parameter # 73A.
2: Only if Parameter # 71A and # 72A are used.

FIGURE 26-16: EXAMPLE SPI SLAVE MODE TIMING (CKE = 1)



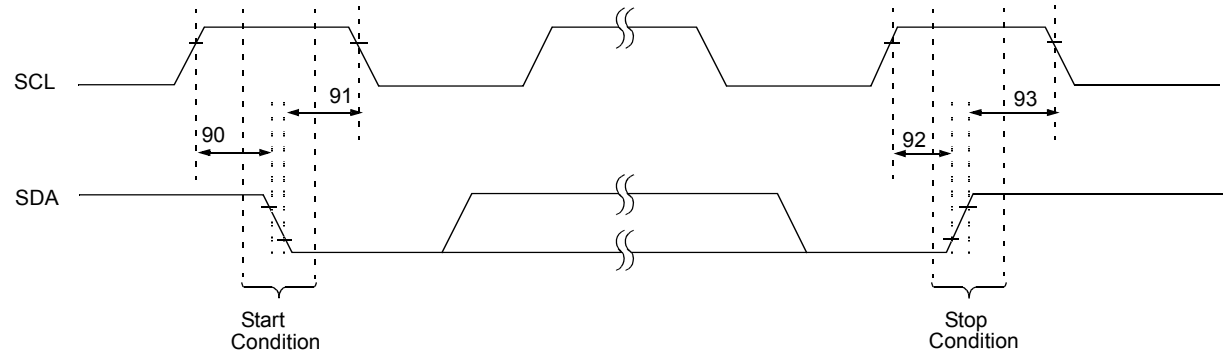
Note: Refer to Figure 26-5 for load conditions.

TABLE 26-17: EXAMPLE SPI SLAVE MODE REQUIREMENTS (CKE = 1)

Param No.	Symbol	Characteristic		Min	Max	Units	Conditions
70	TssL2scH, TssL2scL	$\overline{SS}$ ↓ to SCK ↓ or SCK ↑ Input		Tcy	—	ns	
71	TscH	SCK Input High Time (Slave mode)	Continuous	1.25 Tcy + 30	—	ns	
71A			Single Byte	40	—	ns	**(Note 1)**
72	TscL	SCK Input Low Time (Slave mode)	Continuous	1.25 Tcy + 30	—	ns	
72A			Single Byte	40	—	ns	**(Note 1)**
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 SDI Data Input to SCK Edge		100	—	ns	
75	TdoR	SDO Data Output Rise Time	PIC18**F**XX20	—	25	ns	
			PIC18**LF**XX20		45	ns	
76	TdoF	SDO Data Output Fall Time		—	25	ns	
77	TssH2doZ	$\overline{SS}$↑ to SDO Output High-Impedance		10	50	ns	
78	TscR	SCK Output Rise Time (Master mode)	PIC18**F**XX20	—	25	ns	
			PIC18**LF**XX20	—	45	ns	
79	TscF	SCK Output Fall Time (Master mode)		—	25	ns	
80	TscH2doV, TscL2doV	SDO Data Output Valid after SCK Edge	PIC18**F**XX20	—	50	ns	
			PIC18**LF**XX20	—	100	ns	
82	TssL2doV	SDO Data Output Valid after $\overline{SS}$ ↓ Edge	PIC18**F**XX20	—	50	ns	
			PIC18**LF**XX20	—	100	ns	
83	TscH2ssH, TscL2ssH	$\overline{SS}$ ↑ after SCK edge		1.5 Tcy + 40	—	ns	

Note 1: Requires the use of Parameter # 73A.
2: Only if Parameter # 71A and # 72A are used.

FIGURE 26-17: I²C™ BUS START/STOP BITS TIMING



Note: Refer to Figure 26-5 for load conditions.