PIC16C717/770/771

18/20-Pin, 8-Bit CMOS Microcontrollers with 10/12-Bit A/D

Microcontroller Core Features:

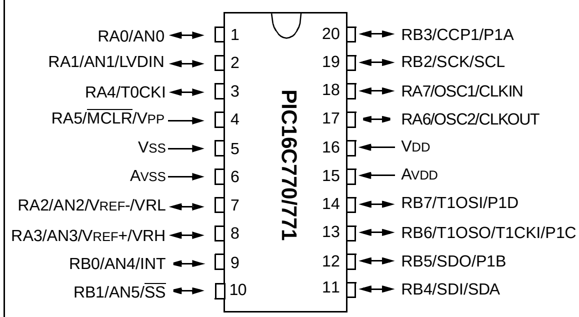
- High-performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches which are two cycle
- Operating speed: DC - 20 MHz clock input
  DC - 200 ns instruction cycle

Device	Memory		Pins	A/D Resolution	A/D Channels
	Program x14	Data x8			
PIC16C717	2K	256	18, 20	10 bits	6
PIC16C770	2K	256	20	12 bits	6
PIC16C771	4K	256	20	12 bits	6

- Interrupt capability (up to 10 internal/external interrupt sources)
- Eight level deep hardware stack
- Direct, indirect and relative addressing modes
- Power-on Reset (POR)
- Power-up Timer (PWRT) and Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own on-chip RC oscillator for reliable operation
- Selectable oscillator options:
  - INTRC - Internal RC, dual speed (4MHz and 37KHz) dynamically switchable for power savings
  - ER - External resistor, dual speed (user selectable frequency and 37KHz) dynamically switchable for power savings
  - EC - External clock
  - HS - High speed crystal/resonator
  - XT - Crystal/resonator
  - LP - Low power crystal
- Low-power, high-speed CMOS EPROM technology
- In-Circuit Serial Programming™ (ISCP)
- Wide operating voltage range: 2.5V to 5.5V
- 15 I/O pins with individual control for:
  - Direction (15 pins)
  - Digital/Analog input (6 pins)
  - PORTB interrupt on change (8 pins)
  - PORTB weak pull-up (8 pins)
  - High voltage open drain (1 pin)
- Commercial and Industrial temperature ranges
- Low-power consumption:
  - < 2 mA @ 5V, 4 MHz
  - 22.5 µA typical @ 3V, 32 kHz
  - < 1 µA typical standby current

Pin Diagram

20-Pin PDIP, SOIC, SSOP

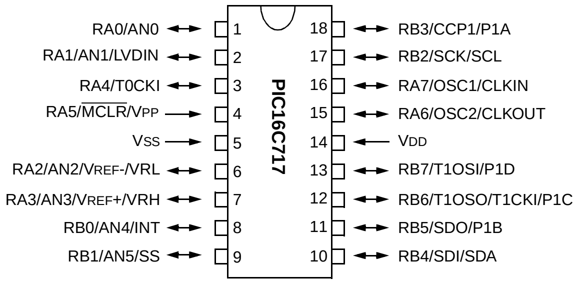


Peripheral Features:

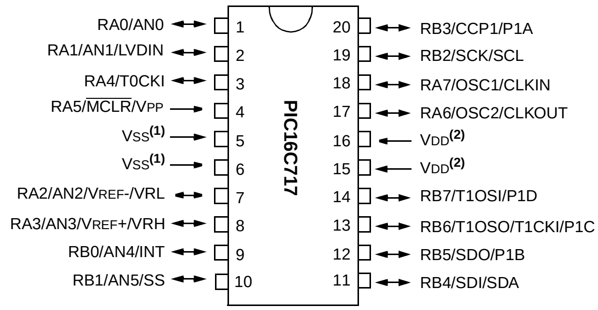
- Timer0: 8-bit timer/counter with 8-bit prescaler
- Timer1: 16-bit timer/counter with prescaler, can be incremented during sleep via external crystal/clock
- Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler
- Enhanced Capture, Compare, PWM (ECCP) module
  - Capture is 16 bit, max. resolution is 12.5 ns
  - Compare is 16 bit, max. resolution is 200 ns
  - PWM max. resolution is 10 bit
  - Enhanced PWM:
    - Single, Half-Bridge and Full-Bridge output modes
    - Digitally programmable deadband delay
- Analog-to-Digital converter:
  - PIC16C770/771 12-bit resolution
  - PIC16C717 10-bit resolution
- On-chip absolute bandgap voltage reference generator
- Programmable Brown-out Reset (PBOR) circuitry
- Programmable Low-Voltage Detection (PLVD) circuitry
- Master Synchronous Serial Port (MSSP) with two modes of operation:
  - 3-wire SPI™ (supports all 4 SPI modes)
  - I²C™ compatible including master mode support
- Program Memory Read (PMR) capability for look-up table, character string storage and checksum calculation purposes

Pin Diagrams

18-Pin PDIP, SOIC



20-Pin SSOP



Note 1: VSS pins 5 and 6 must be tied together.
2: VDD pins 15 and 16 must be tied together.

Key Features PICmicro™ Mid-Range Reference Manual (DS33023)	PIC16C717	PIC16C770	PIC16C771
Operating Frequency	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz
Resets (and Delays)	POR, BOR, $\overline{\text{MCLR}}$, WDT (PWRT, OST)	POR, BOR, $\overline{\text{MCLR}}$, WDT (PWRT, OST)	POR, BOR, $\overline{\text{MCLR}}$, WDT (PWRT, OST)
Program Memory (14-bit words)	2K	2K	4K
Data Memory (bytes)	256	256	256
Interrupts	10	10	10
I/O Ports	Ports A,B	Ports A,B	Ports A,B
Timers	3	3	3
Enhanced Capture/Compare/PWM (ECCP) modules	1	1	1
Serial Communications	MSSP	MSSP	MSSP
12-bit Analog-to-Digital Module	—	6 input channels	6 input channels
10-bit Analog-to-Digital Module	6 input channels	—	—
Instruction Set	35 Instructions	35 Instructions	35 Instructions

Table of Contents

To Our Valued Customers

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.

New Customer Notification System

Register on our web site (www.microchip.com/cn) to receive the most current information on our products.

Errata

An errata sheet may exist for current devices, describing minor operational differences (from the data sheet) and recommended workarounds. 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) 786-7277

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.

Corrections to this Data Sheet

We constantly strive to improve the quality of all our products and documentation. We have spent a great deal of time to ensure that this document is correct. However, we realize that we may have missed a few things. If you find any information that is missing or appears in error, please:

- Fill out and mail in the reader response form in the back of this data sheet.
- E-mail us at webmaster@microchip.com.

We appreciate your assistance in making this a better document.

NOTES:

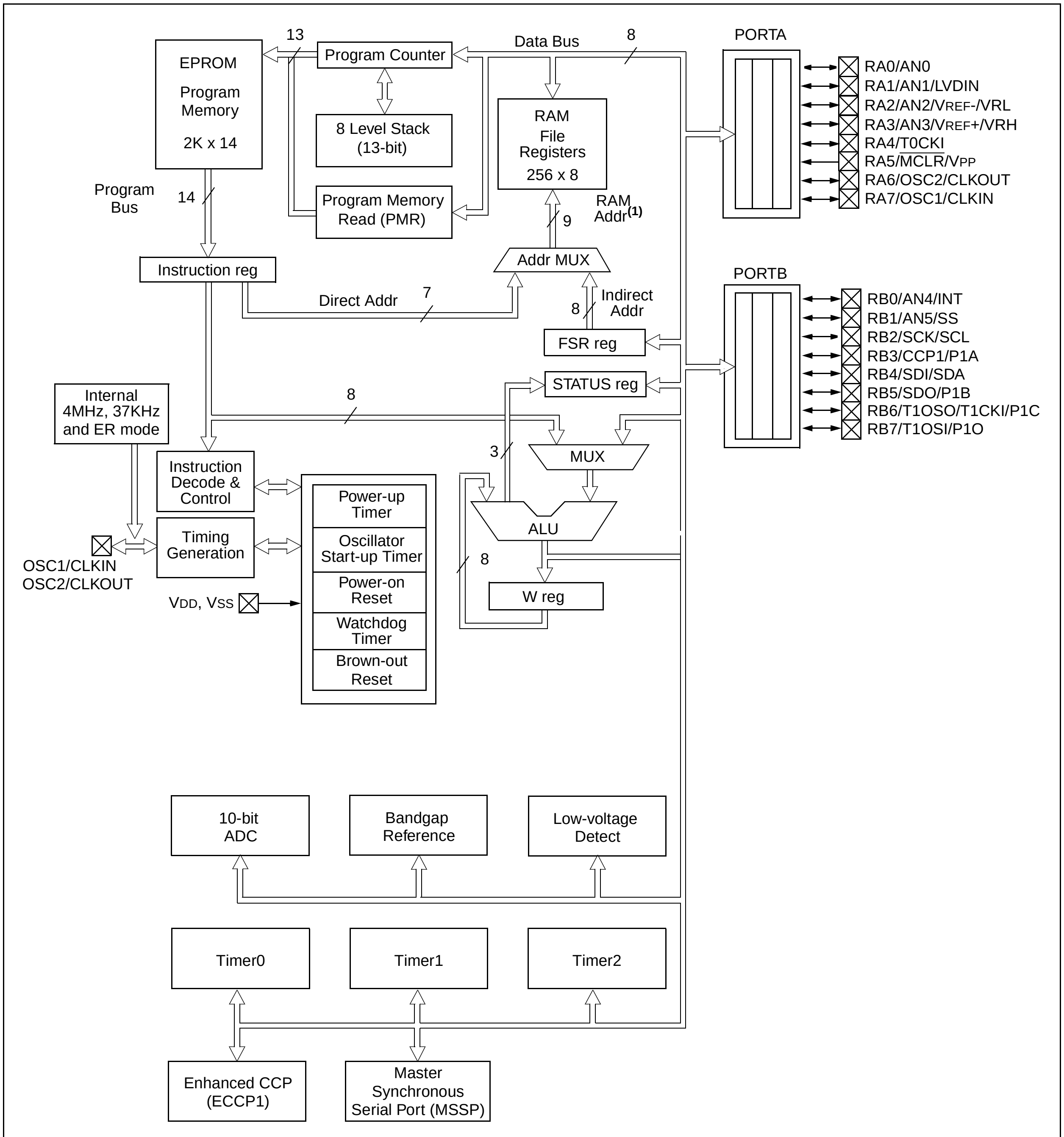
1.0 DEVICE OVERVIEW

This document contains device-specific information. Additional information may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023), which may be obtained from your local Microchip Sales Representative or downloaded from the Microchip website. The Reference Manual should be considered a complementary document to this data sheet, and is highly recommended reading for a better understanding of the device architecture and operation of the peripheral modules.

There are three devices (PIC16C717, PIC16C770 and PIC16C771) covered by this datasheet. The PIC16C717 device comes in 18/20-pin packages and the PIC16C770/771 devices come in 20-pin packages.

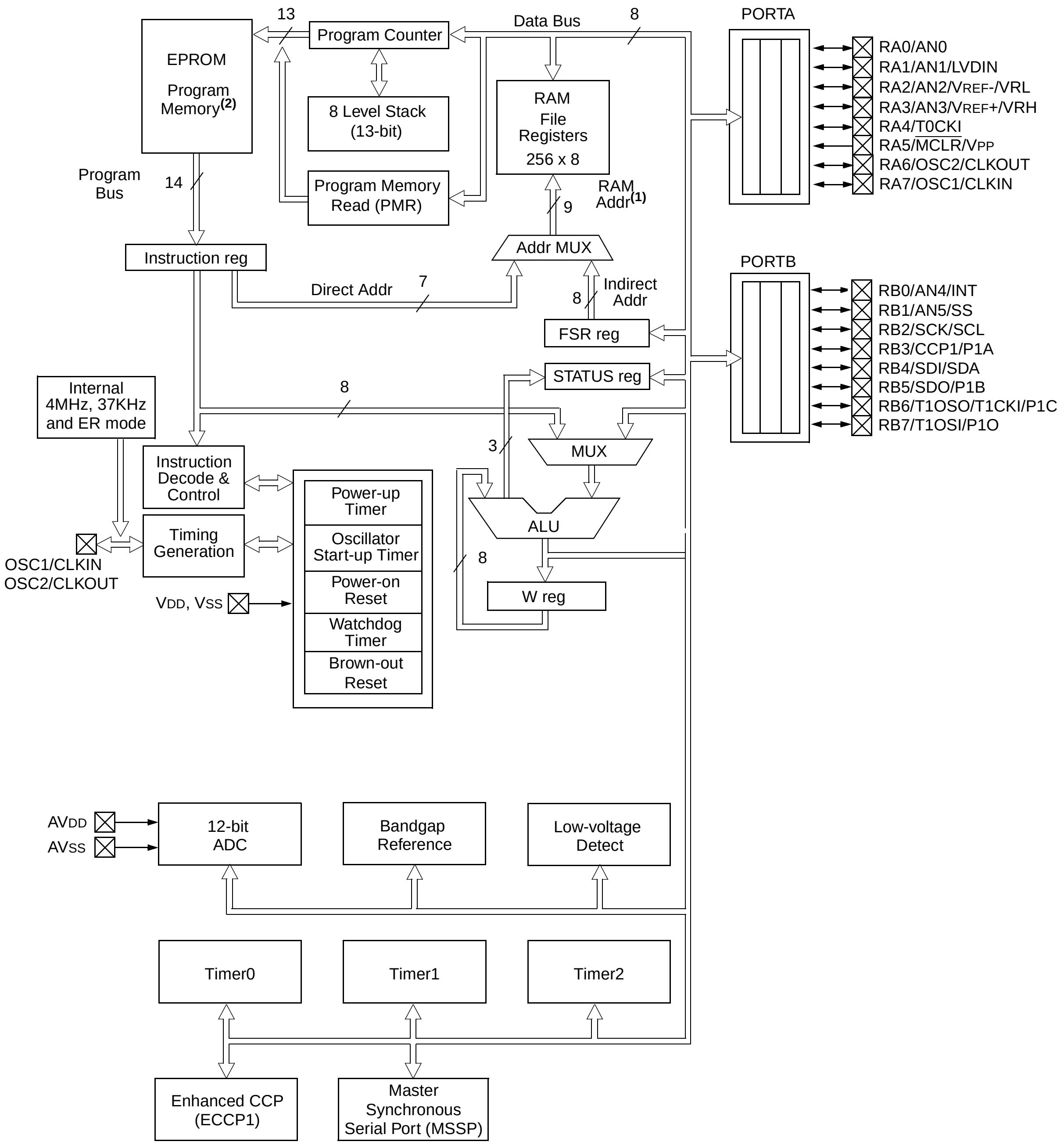
The following two figures are device block diagrams of the PIC16C717 and the PIC16C770/771.

FIGURE 1-1: PIC16C717 BLOCK DIAGRAM



Note 1: Higher order bits are from the STATUS register.

FIGURE 1-2: PIC16C770/771 BLOCK DIAGRAM



Note 1: Higher order bits are from the STATUS register.

2: Program memory for PIC16C770 is 2K x 14. Program memory for PIC16C771 is 4K x 14.

 Advanced Information

TABLE 1-1: PIC16C770/771 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RA0/AN0	RA0	ST	CMOS	Bi-directional I/O
	AN0	AN		A/D input
RA1/AN1/LVDIN	RA1	ST	CMOS	Bi-directional I/O
	AN1	AN		A/D input
	LVDIN	AN		LVD input reference
RA2/AN2/VREF-/VRL	RA2	ST	CMOS	Bi-directional I/O
	AN2	AN		A/D input
	VREF-	AN		Negative analog reference input
	VRL		AN	Internal voltage reference low output
RA3/AN3/VREF+/VRH	RA3	ST	CMOS	Bi-directional I/O
	AN3	AN		A/D input
	VREF+	AN		Positive analog reference input
	VRH		AN	Internal voltage reference high output
RA4/T0CKI	RA4	ST	OD	Bi-directional I/O
	T0CKI	ST		TMR0 clock input
RA5/$\overline{\text{MCLR}}$/VPP	RA5	ST		Input port
	$\overline{\text{MCLR}}$	ST		Master clear
	VPP	Power		Programming voltage
RA6/OSC2/CLKOUT	RA6	ST	CMOS	Bi-directional I/O
	OSC2		XTAL	Crystal/resonator
	CLKOUT		CMOS	FOSC/4 output
RA7/OSC1/CLKIN	RA7	ST	CMOS	Bi-directional I/O
	OSC1	XTAL		Crystal/resonator
	CLKIN	ST		External clock input/ER resistor connection
RB0/AN4/INT	RB0	TTL	CMOS	Bi-directional I/O(1)
	AN4	AN		A/D input
	INT	ST		Interrupt input
RB1/AN5/$\overline{\text{SS}}$	RB1	TTL	CMOS	Bi-directional I/O(1)
	AN5	AN		A/D input
	$\overline{\text{SS}}$	ST		SSP slave select input
RB2/SCK/SCL	RB2	TTL	CMOS	Bi-directional input(1)
	SCK	ST	CMOS	Serial clock I/O for SPI
	SCL	ST	OD	Serial clock I/O for I^2C
RB3/CCP1/P1A	RB3	TTL	CMOS	Bi-directional input(1)
	CCP1	ST	CMOS	Capture 1 input/Compare 1 output
	P1A		CMOS	PWM P1A output
RB4/SDI/SDA	RB4	TTL	CMOS	Bi-directional input(1)
	SDI	ST		Serial data in for SPI
	SDA	ST	OD	Serial data I/O for I^2C
RB5/SDO/P1B	RB5	ST	CMOS	Bi-directional I/O(1)
	SDO		CMOS	Serial data out for SPI
	P1B		CMOS	PWM P1B output

Note 1: Bit programmable pull-ups.

TABLE 1-1: PIC16C770/771 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RB6/T1OSO/T1CKI/P1C	RB6	TTL	CMOS	Bi-directional I/O(1)
	T1OSO		XTAL	Crystal/Resonator
	T1CKI	ST		TMR1 clock input
	P1C		CMOS	PWM P1C output
RB7/T1OSI/P1D	RB7	TTL	CMOS	Bi-directional I/O(1)
	T1OSI	XTAL		TMR1 crystal/resonator
	P1D		CMOS	PWM P1D output
V_{SS}	V_{SS}	Power		Ground reference for logic and I/O pins
V_{DD}	V_{DD}	Power		Positive supply for logic and I/O pins
AV_{SS}	AV_{SS}	Power		Ground reference for analog
AV_{DD}	AV_{DD}	Power		Positive supply for analog

Note 1: Bit programmable pull-ups.

TABLE 1-2: PIC16C717 PINOUT DESCRIPTION

Name	Function	Input Type	Output Type	Description
RA0/AN0	RA0	ST	CMOS	Bi-directional I/O
	AN0	AN		A/D input
RA1/AN1/LVDIN	RA1	ST	CMOS	Bi-directional I/O
	AN1	AN		A/D input reference
	LVDIN	AN		LVD input reference
RA2/AN2/VREF-/VRL	RA2	ST	CMOS	Bi-directional I/O
	AN2	AN		A/D input
	VREF-	AN		Negative analog reference input
	VRL		AN	Internal voltage reference low output
RA3/AN3/VREF+/VRH	RA3	ST	CMOS	Bi-directional I/O
	AN3	AN		A/D input
	VREF+	AN		Positive analog reference high output
	VRH		AN	Internal voltage reference high output
RA4/T0CKI	RA4	ST	OD	Bi-directional I/O
	T0CKI	ST		TMR0 clock input
RA5/$\overline{\text{MCLR}}$/VPP	RA5	ST		Input port
	$\overline{\text{MCLR}}$	ST		Master Clear
	VPP	Power		Programming Voltage
RA6/OSC2/CLKOUT	RA6	ST	CMOS	Bi-directional I/O
	OSC2		XTAL	Crystal/Resonator
	CLKOUT		CMOS	FOSC/4 output
RA7/OSC1/CLKIN	RA7	ST	CMOS	Bi-directional I/O
	OSC1	XTAL		Crystal/Resonator
	CLKIN	ST		External clock input/ER resistor connection
RB0/AN4/INT	RB0	TTL	CMOS	Bi-directional I/O(1)
	AN4	AN		A/D input
	INT	ST		Interrupt input
RB1/AN5/$\overline{\text{SS}}$	RB1	TTL	CMOS	Bi-directional I/O(1)
	AN5	AN		A/D input
	$\overline{\text{SS}}$	ST		SSP slave select input
RB2/SCK/SCL	RB2	TTL	CMOS	Bi-directional input(1)
	SCK	ST	CMOS	Serial clock I/O for SPI
	SCL	ST	OD	Serial clock I/O for I^2C
RB3/CCP1/P1A	RB3	TTL	CMOS	Bi-directional input(1)
	CCP1	ST	CMOS	Capture 1 input/Compare 1 output
	P1A		CMOS	PWM P1A output
RB4/SDI/SDA	RB4	TTL	CMOS	Bi-directional input(1)
	SDI	ST		Serial data in for SPI
	SDA	ST	OD	Serial data I/O for I^2C
RB5/SDO/P1B	RB5	ST	CMOS	Bi-directional I/O(1)
	SDO		CMOS	Serial data out for SPI
	P1B		CMOS	PWM P1B output

Note 1: Bit programmable pull-ups.

TABLE 1-2: PIC16C717 PINOUT DESCRIPTION (CONTINUED)

Name	Function	Input Type	Output Type	Description
RB6/T1OSO/T1CKI/P1C	RB6	TTL	CMOS	Bi-directional I/O(1)
	T1OSO		XTAL	TMR1 Crystal/Resonator
	T1CKI	ST		TMR1 Clock input
	P1C		CMOS	PWM P1C output
RB7/T1OSI/P1D	RB7	TTL	CMOS	Bi-directional I/O(1)
	T1OSI	XTAL		TMR1 Crystal/Resonator
	P1D		CMOS	PWM P1D output
V_{SS}	V_{SS}	Power		Ground
V_{DD}	V_{DD}	Power		Positive Supply

Note 1: Bit programmable pull-ups.

2.0 MEMORY ORGANIZATION

There are two memory blocks in each of these PICmicro® microcontrollers. Each block (Program Memory and Data Memory) has its own bus, so that concurrent access can occur.

Additional information on device memory may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023).

2.1 Program Memory Organization

The PIC16C717/770/771 devices have a 13-bit program counter capable of addressing an 8K x 14 program memory space. The PIC16C717 and the PIC16C770 have 2K x 14 words of program memory. The PIC16C771 has 4K x 14 words of program memory. Accessing a location above the physically implemented address will cause a wraparound.

The reset vector is at 0000h and the interrupt vector is at 0004h.

FIGURE 2-1: PROGRAM MEMORY MAP AND STACK OF THE PIC16C717 AND PIC16C770

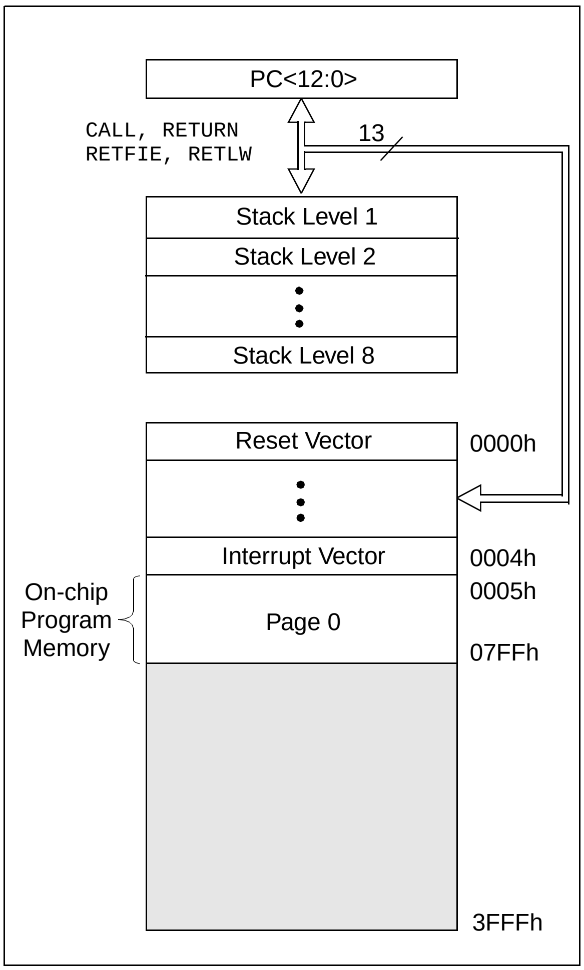
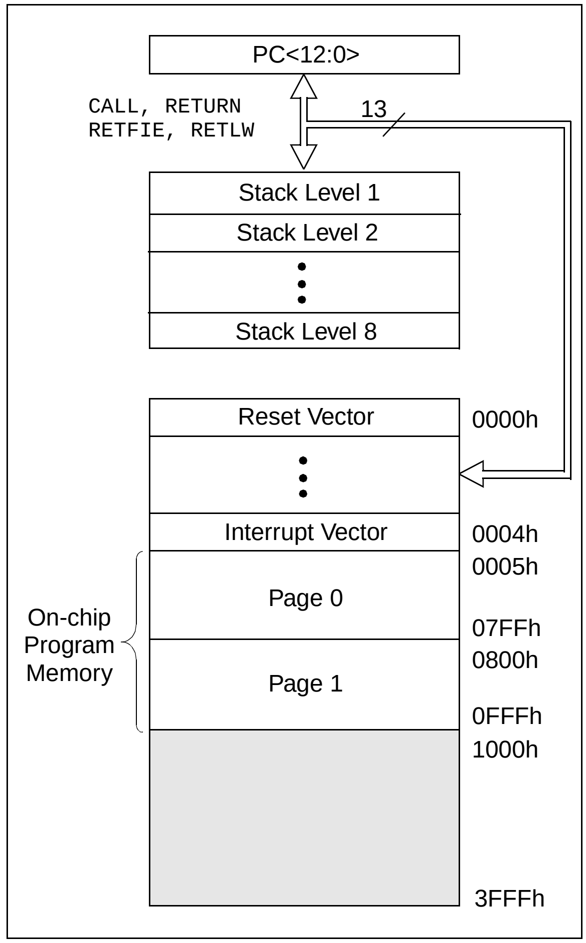


FIGURE 2-2: PROGRAM MEMORY MAP AND STACK OF THE PIC16C771



2.2 Data Memory Organization

The data memory is partitioned into multiple banks, which contain the General Purpose Registers and the Special Function Registers. Bits RP1 and RP0 are the bank select bits.

RP1	RP0

(STATUS<6:5>)

= 00 → Bank0
= 01 → Bank1
= 10 → Bank2
= 11 → Bank3

Each bank extends up to 7Fh (128 bytes). The lower locations of each bank are reserved for the Special Function Registers. Above the Special Function Registers are General Purpose Registers, implemented as static RAM. All implemented banks contain special function registers. Some frequently used special function registers from one bank are mirrored in another bank for code reduction and quicker access.

2.2.1 GENERAL PURPOSE REGISTER FILE

The register file can be accessed either directly, or indirectly, through the File Select Register FSR.

FIGURE 2-3: REGISTER FILE MAP

Bank 0	File Address	Bank 1	File Address	Bank 2	File Address	Bank 3	File Address
Indirect addr.(*)	00h	Indirect addr.(*)	80h	Indirect addr.(*)	100h	Indirect addr.(*)	180h
TMR0	01h	OPTION_REG	81h	TMR0	101h	OPTION_REG	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h		105h		185h
PORTB	06h	TRISB	86h	PORTB	106h	TRISB	186h
	07h		87h		107h		187h
	08h		88h		108h		188h
	09h		89h		109h		189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	PMDATL	10Ch	PMCON1	18Ch
PIR2	0Dh	PIE2	8Dh	PMADRL	10Dh		18Dh
TMR1L	0Eh	PCON	8Eh	PMDATH	10Eh		18Eh
TMR1H	0Fh		8Fh	PMADRH	10Fh		18Fh
T1CON	10h		90h		110h		190h
TMR2	11h	SSPCON2	91h		111h		191h
T2CON	12h	PR2	92h		112h		192h
SSPBUF	13h	SSPADD	93h		113h		193h
SSPCON	14h	SSPSTAT	94h		114h		194h
CCPR1L	15h	WPUB	95h		115h		195h
CCPR1H	16h	IOCB	96h		116h		196h
CCP1CON	17h	P1DEL	97h		117h		197h
	18h		98h		118h		198h
	19h		99h		119h		199h
	1Ah		9Ah		11Ah		19Ah
	1Bh	REFCON	9Bh		11Bh		19Bh
	1Ch	LVDCON	9Ch		11Ch		19Ch
	1Dh	ANSEL	9Dh		11Dh		19Dh
ADRESH	1Eh	ADRESL	9Eh		11Eh		19Eh
ADCON0	1Fh	ADCON1	9Fh		11Fh		19Fh
General Purpose Register 96 Bytes	20h–7Fh	General Purpose Register 80 Bytes	A0h–EFh	General Purpose Register 80 Bytes	120h–6Fh		1A0h–1EFh
		accesses 70h-7Fh	F0h–FFh	accesses 70h - 7Fh	70h–17Fh	accesses 70h - 7Fh	1F0h–1FFh

Shaded cells: Unimplemented data memory locations, read as '0'.

* Not a physical register.

 Advanced Information

2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers 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 2-1.

The special function registers can be classified into two sets; core (CPU) and peripheral. Those registers associated with the core functions are described in detail in this section. Those related to the operation of the peripheral features are described in detail in that peripheral feature section.

TABLE 2-1: PIC16C717/770/771 SPECIAL FUNCTION REGISTER SUMMARY

Address	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 (2)
Bank 0											
00h(3)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
01h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
02h(3)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
03h(3)	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
04h(3)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
05h	PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx 0000	uuuu 0000
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xx00	uuuu uu00
07h	—	Unimplemented								—	—
08h	—	Unimplemented								—	—
09h	—	Unimplemented								—	—
0Ah(1,3)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
0Bh(3)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF	-0-- 0000	-0-- 0000
0Dh	PIR2	LVDIF	—	—	—	BCLIF	—	—	—	0--- 0---	0--- 0---
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYNC}$	TMR1CS	TMR1ON	--00 0000	--uu uuuu
11h	TMR2	Timer2 module's register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
15h	CCPR1L	Capture/Compare/PWM Register1 (LSB)								xxxx xxxx	uuuu uuuu
16h	CCPR1H	Capture/Compare/PWM Register1 (MSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	PWM1M1	PWM1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	0000 0000
18h	—	Unimplemented								—	—
19h	—	Unimplemented								—	—
1Ah	—	Unimplemented								—	—
1Bh	—	Unimplemented								—	—
1Ch	—	Unimplemented								—	—
1Dh	—	Unimplemented								—	—
1Eh	ADRESH	A/D High Byte Result Register								xxxx xxxx	uuuu uuuu
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{DONE}$	CHS3	ADON	0000 0000	0000 0000

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0'. Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.

2: Other (non power-up) resets include external reset through $\overline{MCLR}$ and Watchdog Timer Reset.

3: These registers can be addressed from any bank.

TABLE 2-1: PIC16C717/770/771 SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	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 (2)
Bank 1											
80h(3)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
81h	OPTION_REG	$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
82h(3)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
83h(3)	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000q quuu
84h(3)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
85h	TRISA	PORTA Data Direction Register								1111 1111	1111 1111
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
87h	—	Unimplemented								—	—
88h	—	Unimplemented								—	—
89h	—	Unimplemented								—	—
8Ah(1,3)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
8Bh(3)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
8Ch	PIE1	—	ADIE	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE	-0-- 0000	-0-- 0000
8Dh	PIE2	LVDIE	—	—	—	BCLIE	—	—	—	0--- 0---	0--- 0---
8Eh	PCON	—	—	—	—	OSCF	—	$\overline{POR}$	$\overline{BOR}$	---- 1-qq	---- 1-uu
8Fh	—	Unimplemented								—	—
90h	—	Unimplemented								—	—
91h	SSPCON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
92h	PR2	Timer2 Period Register								1111 1111	1111 1111
93h	SSPADD	Synchronous Serial Port (I^2C mode) Address Register								0000 0000	0000 0000
94h	SSPSTAT	SMP	CKE	D/$\overline{A}$	P	S	R/$\overline{W}$	UA	BF	0000 0000	0000 0000
95h	WPUB	PORTB Weak Pull-up Control								1111 1111	1111 1111
96h	IOCB	PORTB Interrupt on Change Control								1111 0000	1111 0000
97h	P1DEL	PWM 1 Delay value								0000 0000	0000 0000
98h	—	Unimplemented								—	—
99h	—	Unimplemented								—	—
9Ah	—	Unimplemented								—	—
9Bh	REFCON	VRHEN	VRLEN	VRHOEN	VRLOEN	—	—	—	—	0000 ----	0000 ----
9Ch	LVDCON	—	—	BGST	LVDEN	LVV3	LVV2	LVV1	LVV0	--00 0101	--00 0101
9Dh	ANSEL			Analog Channel Select						1111 1111	1111 1111
9Eh	ADRESL	A/D Low Byte Result Register								xxxx xxxx	uuuu uuuu
9Fh	ADCON1	ADFM	VCFG2	VCFG1	VCFG0					0000 0000	0000 0000

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0'.
Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.

2: Other (non power-up) resets include external reset through $\overline{MCLR}$ and Watchdog Timer Reset.

3: These registers can be addressed from any bank.

TABLE 2-1: PIC16C717/770/771 SPECIAL FUNCTION REGISTER SUMMARY (CONTINUED)

Address	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 (2)
Bank 2											
100h(3)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
101h	TMR0	Timer0 module's register								xxxx xxxx	uuuu uuuu
102h(3)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
103h(3)	STATUS	IRP	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0001 1xxx	000q quuu
104h(3)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
105h	—	Unimplemented								—	—
106h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xx00	uuuu uu00
107h	—	Unimplemented								—	—
108h	—	Unimplemented								—	—
109h	—	Unimplemented								—	—
10Ah(1,3)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
10Bh(3)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
10Ch	PMDATL	Program memory read data low								xxxx xxxx	uuuu uuuu
10Dh	PMADRL	Program memory read address low								xxxx xxxx	uuuu uuuu
10Eh	PMDATH	—	—	Program memory read data high						--xx xxxx	--uu uuuu
10Fh	PMADRH	—	—	—	—	Program memory read address high				---- xxxx	---- uuuu
110h-11Fh	—	Unimplemented								—	—
Bank 3											
180h(3)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								0000 0000	0000 0000
181h	OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
182h(3)	PCL	Program Counter's (PC) Least Significant Byte								0000 0000	0000 0000
183h(3)	STATUS	IRP	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0001 1xxx	000q quuu
184h(3)	FSR	Indirect data memory address pointer								xxxx xxxx	uuuu uuuu
185h	—	Unimplemented								—	—
186h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
187h	—	Unimplemented								—	—
188h	—	Unimplemented								—	—
189h	—	Unimplemented								—	—
18Ah(1,3)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	---0 0000
18Bh(3)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
18Ch	PMCON1	Reserved	—	—	—	—	—	—	RD	1--- ---0	1--- ---0
18Dh-18Fh	—	Unimplemented								—	—

Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented read as '0'. Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.

2: Other (non power-up) resets include external reset through $\overline{\text{MCLR}}$ and Watchdog Timer Reset.

3: These registers can be addressed from any bank.

2.2.2.1 STATUS REGISTER

The STATUS register, shown in Register 2-1, contains the arithmetic status of the ALU, the RESET status and the bank select bits for data memory.

The STATUS register can be the destination for any instruction, as with any other register. If the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. These bits are set or cleared according to the device logic. Furthermore, the $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits are not writable. Therefore, the result of an instruction with the STATUS register as destination may be different than intended.

For example, `CLRF STATUS` will clear the upper-three bits and set the Z bit. This leaves the STATUS register as `000u u1uu` (where u = unchanged).

It is recommended, therefore, that only `BCF`, `BSF`, `SWAPF` and `MOVWF` instructions are used to alter the STATUS register, because these instructions do not affect the Z, C or DC bits from the STATUS register. For other instructions not affecting any status bits, see the "Instruction Set Summary."

Note 2:	The C and DC bits operate as a $\overline{\text{borrow}}$ and $\overline{\text{digit borrow}}$ bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

REGISTER 2-1: STATUS REGISTER (STATUS: 03h, 83h, 103h, 183h)

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as ‘0’
- n = Value at POR reset

bit 7: **IRP:** Register Bank Select bit (used for indirect addressing)
`1` = Bank 2, 3 (100h - 1FFh)
`0` = Bank 0, 1 (00h - FFh)

bit 6-5: **RP<1:0>:** Register Bank Select bits (used for direct addressing)
`11` = Bank 3 (180h - 1FFh)
`10` = Bank 2 (100h - 17Fh)
`01` = Bank 1 (80h - FFh)
`00` = Bank 0 (00h - 7Fh)
Each bank is 128 bytes

bit 4: **$\overline{\text{TO}}$:** Time-out bit
`1` = After power-up, `CLRWDT` instruction, or `SLEEP` instruction
`0` = A WDT time-out occurred

bit 3: **$\overline{\text{PD}}$:** Power-down bit
`1` = After power-up or by the `CLRWDT` instruction
`0` = By execution of the `SLEEP` instruction

bit 2: **Z:** Zero bit
`1` = The result of an arithmetic or logic operation is zero
`0` = The result of an arithmetic or logic operation is not zero

bit 1: **DC:** Digit carry/$\overline{\text{borrow}}$ bit (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions) (for $\overline{\text{borrow}}$ the polarity is reversed)
`1` = A carry-out from the 4th low order bit of the result occurred
`0` = No carry-out from the 4th low order bit of the result

bit 0: **C:** Carry/$\overline{\text{borrow}}$ bit (`ADDWF`, `ADDLW`, `SUBLW`, `SUBWF` instructions)
`1` = A carry-out from the most significant bit of the result occurred
`0` = No carry-out from the most significant bit of the result occurred

Note: For $\overline{\text{borrow}}$, the polarity is reversed. A subtraction is executed by adding the two’s complement of the second operand. For rotate (`RRF, RLF`) instructions, this bit is loaded with either the high or low order bit of the source register.

2.2.2.2 OPTION_REG REGISTER

The OPTION_REG register is a readable and writable register, which contains various control bits to configure the TMR0 prescaler/WDT postscaler (single assignable register known also as the prescaler), the External INT Interrupt, TMR0 and the weak pull-ups on PORTB.

> **Note:** To achieve a 1:1 prescaler assignment for the TMR0 register, assign the prescaler to the Watchdog Timer.

REGISTER 2-2: OPTION REGISTER (OPTION_REG: 81h, 181h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **$\overline{RBPU}$:** PORTB Pull-up Enable bit(1)
1 = PORTB weak pull-ups are disabled
0 = PORTB weak pull-ups are enabled by the WPUB register

bit 6: **INTEDG:** Interrupt Edge Select bit
1 = Interrupt on rising edge of RB0/INT pin
0 = Interrupt on falling edge of RB0/INT pin

bit 5: **T0CS:** TMR0 Clock Source Select bit
1 = Transition on RA4/T0CKI pin
0 = Internal instruction cycle clock (CLKOUT)

bit 4: **T0SE:** TMR0 Source Edge Select bit
1 = Increment on high-to-low transition on RA4/T0CKI pin
0 = Increment on low-to-high transition on RA4/T0CKI pin

bit 3: **PSA:** Prescaler Assignment bit
1 = Prescaler is assigned to the WDT
0 = Prescaler is assigned to the Timer0 module

bit 2-0: **PS<2:0>:** Prescaler Rate Select bits

Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

Note 1: Individual weak pull-up on RB pins can be enabled/disabled from the weak pull-up PORTB Register (WPUB).

2.2.2.3 INTCON REGISTER

The INTCON Register is a readable and writable register, which contains various enable and flag bits for the TMR0 register overflow, RB Port change and External RB0/INT pin interrupts.

> **Note:** Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-3: INTERRUPT CONTROL REGISTER (INTCON: 0Bh, 8Bh, 10Bh, 18Bh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as ‘0’
- n = Value at POR reset

bit 7: **GIE:** Global Interrupt Enable bit
1 = Enables all un-masked interrupts
0 = Disables all interrupts

bit 6: **PEIE:** Peripheral Interrupt Enable bit
1 = Enables all un-masked peripheral interrupts
0 = Disables all peripheral interrupts

bit 5: **T0IE:** TMR0 Overflow Interrupt Enable bit
1 = Enables the TMR0 interrupt
0 = Disables the TMR0 interrupt

bit 4: **INTE:** RB0/INT External Interrupt Enable bit
1 = Enables the RB0/INT external interrupt
0 = Disables the RB0/INT external interrupt

bit 3: **RBIE:** RB Port Change Interrupt Enable bit(1)
1 = Enables the RB port change interrupt
0 = Disables the RB port change interrupt

bit 2: **T0IF:** TMR0 Overflow Interrupt Flag bit
1 = TMR0 register has overflowed (must be cleared in software)
0 = TMR0 register did not overflow

bit 1: **INTF:** RB0/INT External Interrupt Flag bit
1 = The RB0/INT external interrupt occurred (must be cleared in software)
0 = The RB0/INT external interrupt did not occur

bit 0: **RBIF:** RB Port Change Interrupt Flag bit(1)
1 = At least one of the RB<7:0> pins changed state (must be cleared in software)
0 = None of the RB<7:0> pins have changed state

Note 1: Individual RB pin interrupt on change can be enabled/disabled from the Interrupt on Change PORTB register (IOCB).

2.2.2.4 PIE1 REGISTER

This register contains the individual enable bits for the peripheral interrupts.

Note:	Bit PEIE (INTCON<6>) must be set to enable any peripheral interrupt.

REGISTER 2-4: PERIPHERAL INTERRUPT ENABLE REGISTER 1 (PIE1: 8Ch)

U-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	ADIE	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as ‘0’
\- n = Value at POR reset

bit 7: **Unimplemented:** Read as ’0’

bit 6: **ADIE:** A/D Converter Interrupt Enable bit
1 = Enables the A/D interrupt
0 = Disables the A/D interrupt

bit 5-4: **Unimplemented:** Read as ’0’

bit 3: **SSPIE:** Synchronous Serial Port Interrupt Enable bit
1 = Enables the SSP interrupt
0 = Disables the SSP interrupt

bit 2: **CCP1IE:** CCP1 Interrupt Enable bit
1 = Enables the CCP1 interrupt
0 = Disables the CCP1 interrupt

bit 1: **TMR2IE:** TMR2 to PR2 Match Interrupt Enable bit
1 = Enables the TMR2 to PR2 match interrupt
0 = Disables the TMR2 to PR2 match interrupt

bit 0: **TMR1IE:** TMR1 Overflow Interrupt Enable bit
1 = Enables the TMR1 overflow interrupt
0 = Disables the TMR1 overflow interrupt

2.2.2.5 PIR1 REGISTER

This register contains the individual flag bits for the peripheral interrupts.

Note:	Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-5: PERIPHERAL INTERRUPT REGISTER 1 (PIR1: 0Ch)

U-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	ADIF	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **Unimplemented:** Read as '0'.

bit 6: **ADIF:** A/D Converter Interrupt Flag bit
1 = An A/D conversion completed
0 = The A/D conversion is not complete

bit 5-4: **Unimplemented:** Read as '0'.

bit 3: **SSPIF:** Synchronous Serial Port (SSP) Interrupt Flag
1 = The SSP interrupt condition has occurred, and must be cleared in software before returning from the interrupt service routine. The conditions that will set this bit are:
SPI
A transmission/reception has taken place.
I^2C Slave / Master
A transmission/reception has taken place.
I^2C Master
The initiated start condition was completed by the SSP module.
The initiated stop condition was completed by the SSP module.
The initiated restart condition was completed by the SSP module.
The initiated acknowledge condition was completed by the SSP module.
A start condition occurred while the SSP module was idle (Multimaster system).
A stop condition occurred while the SSP module was idle (Multimaster system).
0 = No SSP interrupt condition has occurred.

bit 2: **CCP1IF:** CCP1 Interrupt Flag bit
Capture Mode
1 = A TMR1 register capture occurred (must be cleared in software)
0 = No TMR1 register capture occurred
Compare Mode
1 = A TMR1 register compare match occurred (must be cleared in software)
0 = No TMR1 register compare match occurred
PWM Mode
Unused in this mode

bit 1: **TMR2IF:** TMR2 to PR2 Match Interrupt Flag bit
1 = TMR2 to PR2 match occurred (must be cleared in software)
0 = No TMR2 to PR2 match occurred

bit 0: **TMR1IF:** TMR1 Overflow Interrupt Flag bit
1 = TMR1 register overflowed (must be cleared in software)
0 = TMR1 register did not overflow

2.2.2.6 PIE2 REGISTER

This register contains the individual enable bits for the SSP bus collision and low voltage detect interrupts.

REGISTER 2-6: PERIPHERAL INTERRUPT REGISTER 2 (PIE2: 8Dh)

R/W-0	U-0	U-0	U-0	R/W-0	U-0	U-0	U-0
LVDIE	—	—	—	BCLIE	—	—	—
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **LVDIE:** Low-voltage Detect Interrupt Enable bit
1 = LVD Interrupt is enabled
0 = LVD Interrupt is disabled

bit 6-4: **Unimplemented:** Read as '0'

bit 3: **BCLIE:** Bus Collision Interrupt Enable bit
1 = Bus Collision interrupt is enabled
0 = Bus Collision interrupt is disabled

bit 2-0: **Unimplemented:** Read as '0'

2.2.2.7 PIR2 REGISTER

This register contains the SSP Bus Collision and low-voltage detect interrupt flag bits.

.

Note:	Interrupt flag bits get set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>). User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.

REGISTER 2-7: PERIPHERAL INTERRUPT REGISTER 2 (PIR2: 0Dh)

R/W-0	U-0	U-0	U-0	R/W-0	U-0	U-0	U-0
LVDIF	—	—	—	BCLIF	—	—	—
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as ‘0’
- n = Value at POR reset

bit 7: **LVDIF:** Low-voltage Detect Interrupt Flag bit
`1` = The supply voltage has fallen below the specified LVD voltage (must be cleared in software)
`0` = The supply voltage is greater than the specified LVD voltage

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

bit 3: **BCLIF:** Bus Collision Interrupt Flag bit
`1` = A bus collision has occurred while the SSP module configured in I^2C Master was transmitting (must be cleared in software)
`0` = No bus collision occurred

bit 2-0: **Unimplemented:** Read as ’0’

 Advanced Information

2.2.2.8 PCON REGISTER

The Power Control (PCON) register contains a flag bit to allow differentiation between a Power-on Reset (POR) to an external $\overline{MCLR}$ Reset or WDT Reset. Those devices with brown-out detection circuitry contain an additional bit to differentiate a Brown-out Reset condition from a Power-on Reset condition.

The PCON register also contains the frequency select bit of the INTRC or ER oscillator.

Note:	$\overline{BOR}$ is unknown on Power-on Reset. It must then be set by the user and checked on subsequent resets to see if $\overline{BOR}$ is clear, indicating a brown-out has occurred. The $\overline{BOR}$ status bit is a don't care and is not necessarily predictable if the brown-out circuit is disabled (by clearing the BODEN bit in the Configuration word).

REGISTER 2-8: POWER CONTROL REGISTER (PCON: 8Eh)

U-0	U-0	U-0	U-0	R/W-1	U-0	R/W-q	R/W-q
—	—	—	—	OSCF	—	$\overline{POR}$	$\overline{BOR}$
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7-4,2: **Unimplemented:** Read as '0'

bit 3: **OSCF:** Oscillator speed
INTRC Mode
1 = 4 MHz nominal
0 = 37 KHz nominal
ER Mode
1 = Oscillator frequency depends on the external resistor value on the OSC1 pin.
0 = 37 KHz nominal
All other modes
x = Ignored

bit 1: **$\overline{POR}$:** Power-on Reset Status bit
1 = No Power-on Reset occurred
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 = No Brown-out Reset occurred
0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

2.3 PCL and PCLATH

The program counter (PC) specifies the address of the instruction to fetch for execution. The PC is 13 bits wide. The low byte is called the PCL register. This register is readable and writable. The high byte is called the PCH register. This register contains the PC<12:8> bits and is not directly readable or writable. All updates to the PCH register occur through the PCLATH register.

2.3.1 PROGRAM MEMORY PAGING

PIC16C717/770/771 devices are capable of addressing a continuous 8K word block of program memory. The `CALL` and `GOTO` instructions provide only 11 bits of address to allow branching within any 2K program memory page. When doing a `CALL` or `GOTO` instruction, the upper 2 bits of the address are provided by PCLATH<4:3>. When doing a `CALL` or `GOTO` instruction, the user must ensure that the page select bits are programmed so that the desired program memory page is addressed. A return instruction pops a PC address off the stack onto the PC register. Therefore, manipulation of the PCLATH<4:3> bits are not required for the return instructions (which POPs the address from the stack).

2.4 Stack

The stack allows a combination of up to 8 program calls and interrupts to occur. The stack contains the return address from this branch in program execution.

Mid-range devices have an 8-level deep x 13-bit wide hardware stack. The stack space is not part of either program or data space and the stack pointer is not readable or writable. The PC is PUSHed onto the stack when a `CALL` instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a `RETURN`, `RETLW` or a `RETFIE` instruction execution. PCLATH is not modified when the stack is PUSHed or POPed.

After the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

The INDF register is not a physical register. Addressing INDF actually addresses the register whose address is contained in the FSR register (FSR is a *pointer*). This is indirect addressing.

Reading INDF itself indirectly (FSR = 0) will produce 00h. Writing to the INDF register indirectly results in a no-operation (although STATUS bits may be affected).

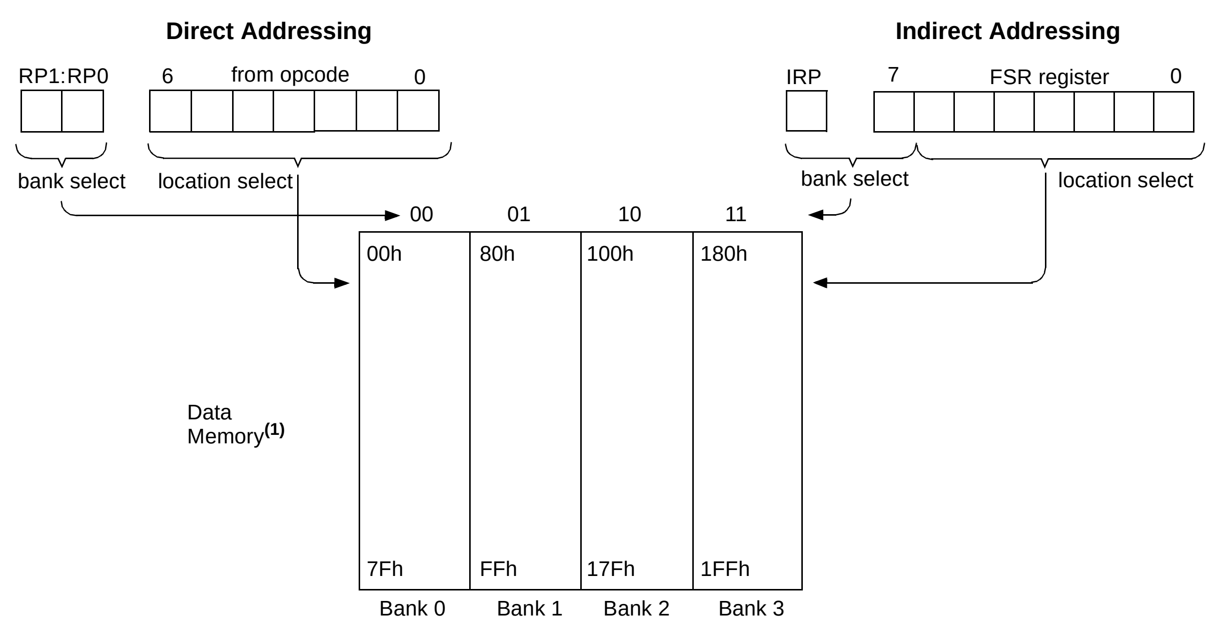
A simple program to clear RAM locations 20h-2Fh using indirect addressing is shown in Example 2-1.

EXAMPLE 2-1: HOW TO CLEAR RAM USING INDIRECT ADDRESSING

```
          movlw  0x20   ;initialize pointer
          movwf  FSR    ;  to RAM
NEXT      clrf   INDF   ;clear INDF register
          incf   FSR    ;inc pointer
          btfss  FSR,4  ;all done?
          goto   NEXT   ;NO, clear next
CONTINUE
          :             ;YES, continue
```

An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit (STATUS<7>), as shown in Figure 2-4.

FIGURE 2-4: DIRECT/INDIRECT ADDRESSING



Note 1: For register file map detail see Figure 2-3.

NOTES:

3.0 I/O PORTS

Some pins for these I/O ports are multiplexed with an alternate function for the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

Additional information on I/O ports may be found in the PICmicro™ Mid-Range Reference Manual, (DS33023).

3.1 I/O Port Analog/Digital Mode

The PIC16C717/770/771 have two I/O ports: PORTA and PORTB. Some of these port pins are mixed-signal (can be digital or analog). When an analog signal is present on a pin, the pin must be configured as an analog input to prevent unnecessary current draw from the power supply. The Analog Select Register (ANSEL) allows the user to individually select the digital/analog mode on these pins. When the analog mode is active, the port pin will always read 0.

> **Note 1:** On a Power-on Reset, the ANSEL register configures these mixed-signal pins as analog mode.
>
> **2:** If a pin is configured as analog mode, the pin will always read '0', even if the digital output is active.

REGISTER 3-1: ANALOG SELECT REGISTER (ANSEL: 9Dh)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
		ANS5	ANS4	ANS3	ANS2	ANS1	ANS0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as ‘0’
-n = Value at POR reset

bit 7-6: **Reserved:** Do not use

bit 5-0: **ANS<5:0>:** Analog Select between analog or digital function on pins AN<5:0>, respectively.
0 = Digital I/O. Pin is assigned to port or special function.
1 = Analog Input. Pin is assigned as analog input.

Note: Setting a pin to an analog input disables digital inputs and any pull-up that may be present. The corresponding TRIS bit should be set to input mode when using pins as analog inputs.

3.2 PORTA and the TRISA Register

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

Reading the PORTA register reads the status of the pins, whereas writing to it will write to the port latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read, this value is modified, and then written to the port data latch.

Pins RA<3:0> are multiplexed with analog functions, such as analog inputs to the A/D converter, analog VREF inputs, and the on-board bandgap reference outputs. When the analog peripherals are using any of these pins as analog input/output, the ANSEL register must have the proper value to individually select the analog mode of the corresponding pins.

> **Note:** Upon reset, the ANSEL register configures the RA<3:0> pins as analog inputs. All RA<3:0> pins will read as ’0’.

Pin RA4 is multiplexed with the Timer0 module clock input to become the RA4/T0CKI pin. The RA4/T0CKI pin is a Schmitt Trigger input and an open drain output.

Pin RA5 is multiplexed with the device reset ($\overline{MCLR}$) and programming input (VPP) functions. The RA5/$\overline{MCLR}$/VPP input only pin has a Schmitt Trigger input buffer. All other RA port pins have Schmitt Trigger input buffers and full CMOS output buffers.

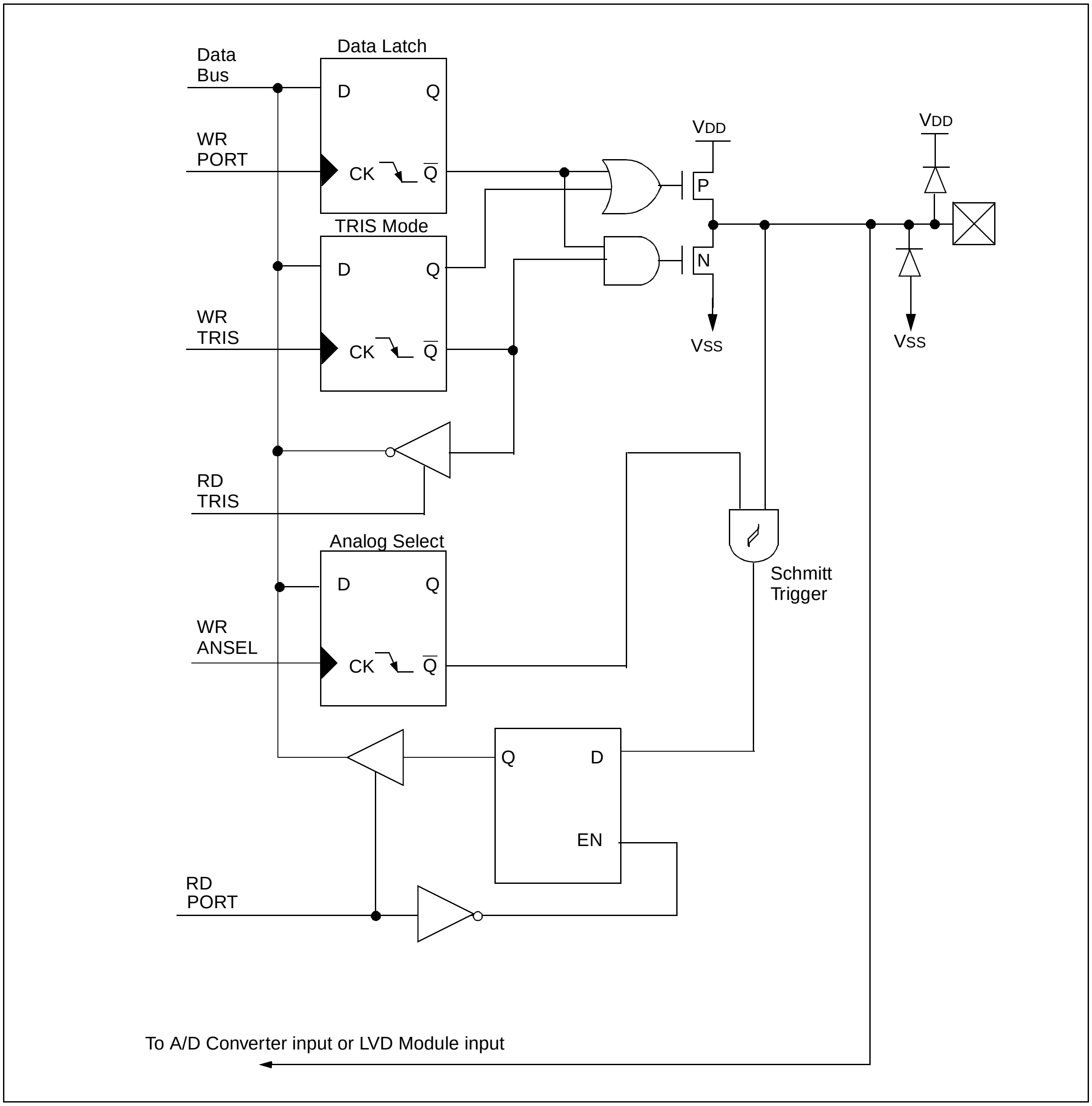
Pins RA6 and RA7 are multiplexed with the oscillator input and output functions.

The TRISA register controls the direction of the RA pins, even when they are being used as analog inputs. The user must ensure the bits in the TRISA register are maintained set when using them as analog inputs.

EXAMPLE 3-1: INITIALIZING PORTA

```
BCF     STATUS, RP0  ; Select Bank 0
CLRF    PORTA        ; Initialize PORTA by
                     ; clearing output
                     ; data latches
BSF     STATUS, RP0  ; Select Bank 1
MOVLW   0Fh          ; Value used to
                     ; initialize data
                     ; direction
MOVWF   TRISA        ; Set RA<3:0> as inputs
                     ; RA<7:4> as outputs. RA<7:6>availability depends on oscillator selection.
MOVLW   03           ; Set RA<1:0> as analog inputs, RA<7:2> are digital I/O
MOVWF   ANSEL
BCF     STATUS, RP0  ; Return to Bank 0
```

FIGURE 3-1: BLOCK DIAGRAM OF RA0/AN0, RA1/AN1/LVDIN



 Advanced Information

FIGURE 3-2: BLOCK DIAGRAM OF RA2/AN2/VREF-/VRL AND RA3/AN3/VREF+/VRH

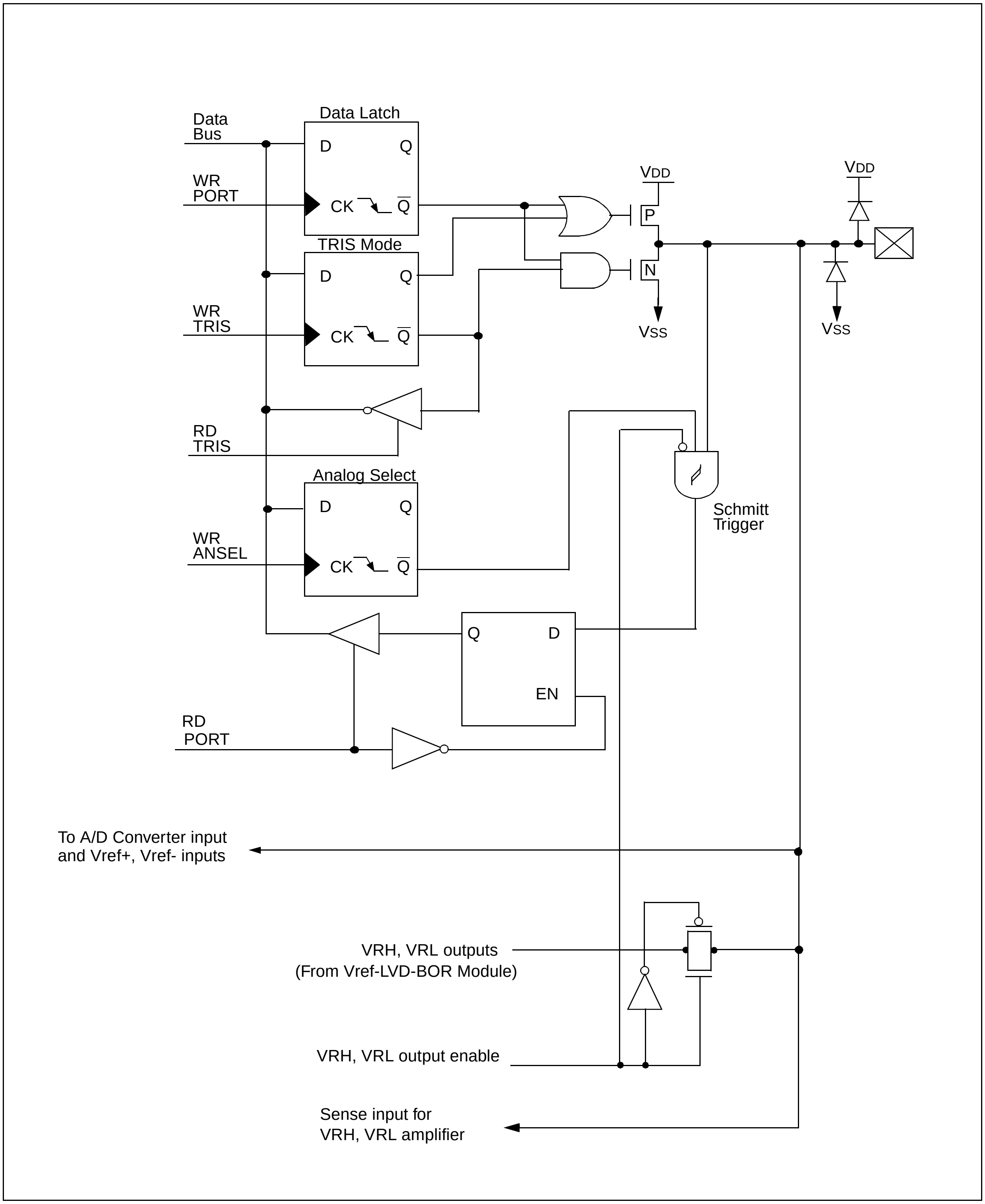
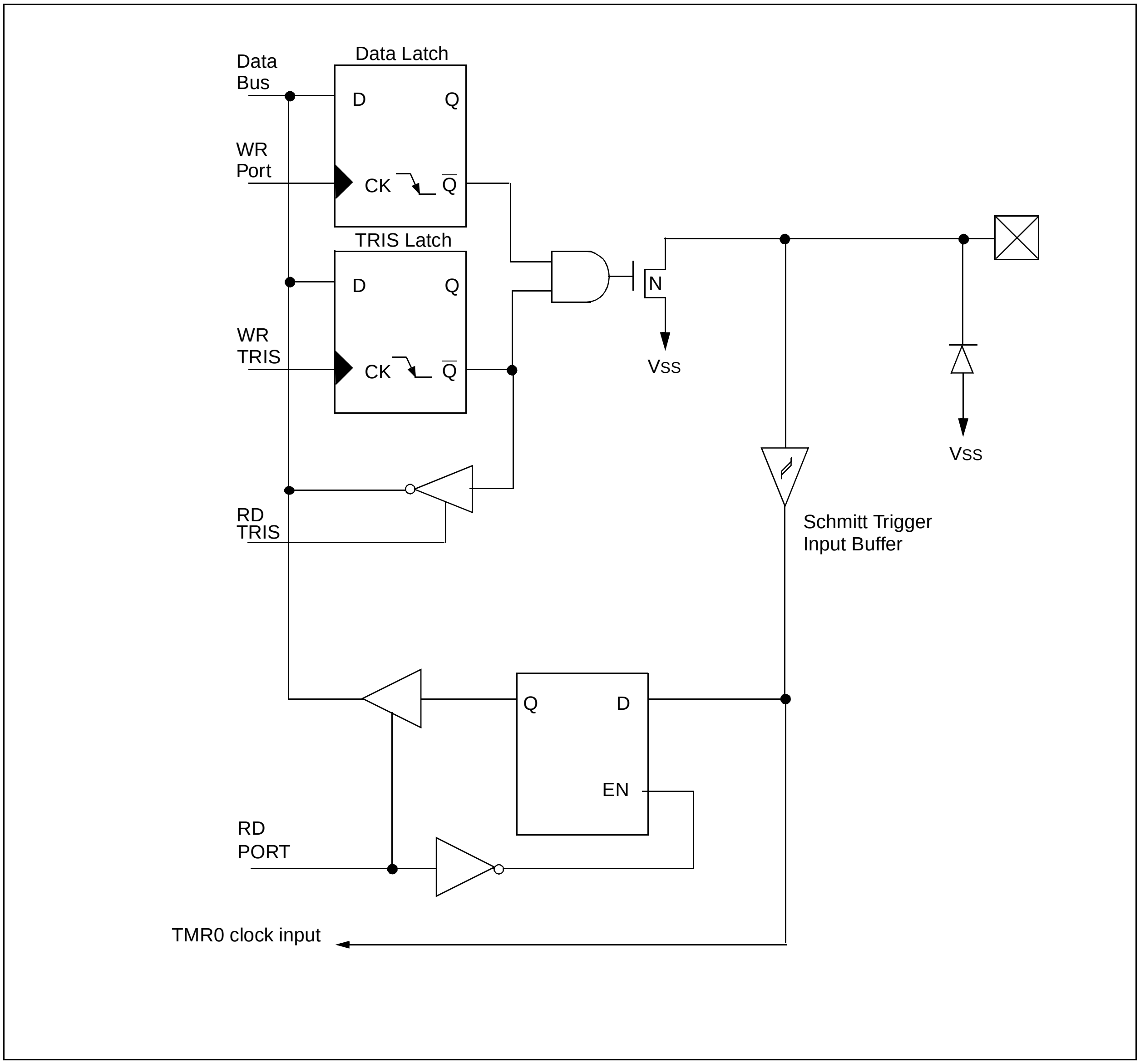


FIGURE 3-3: BLOCK DIAGRAM OF RA4/T0CKI



 Advanced Information

FIGURE 3-4: BLOCK DIAGRAM OF RA5/$\overline{MCLR}$/VPP

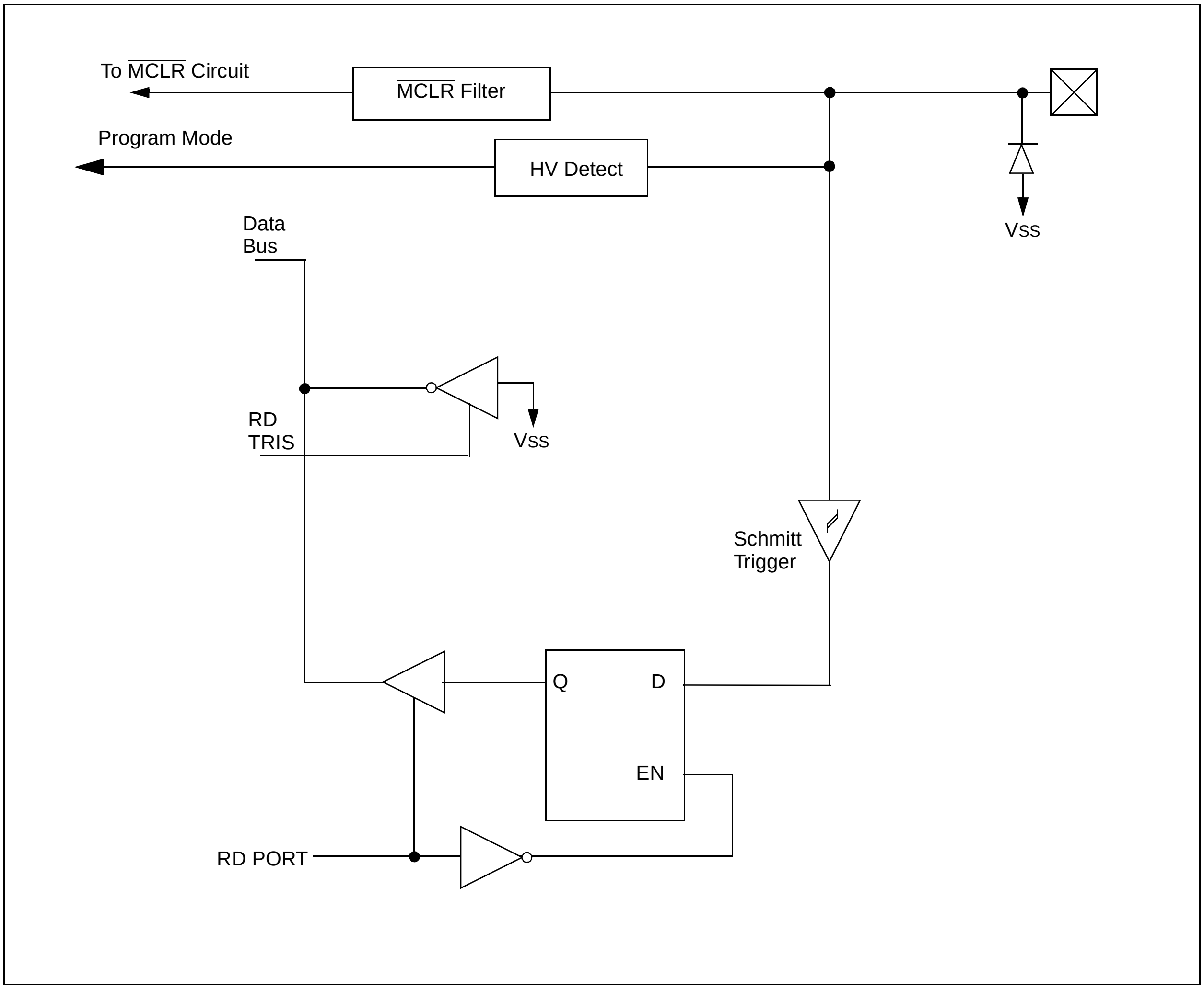
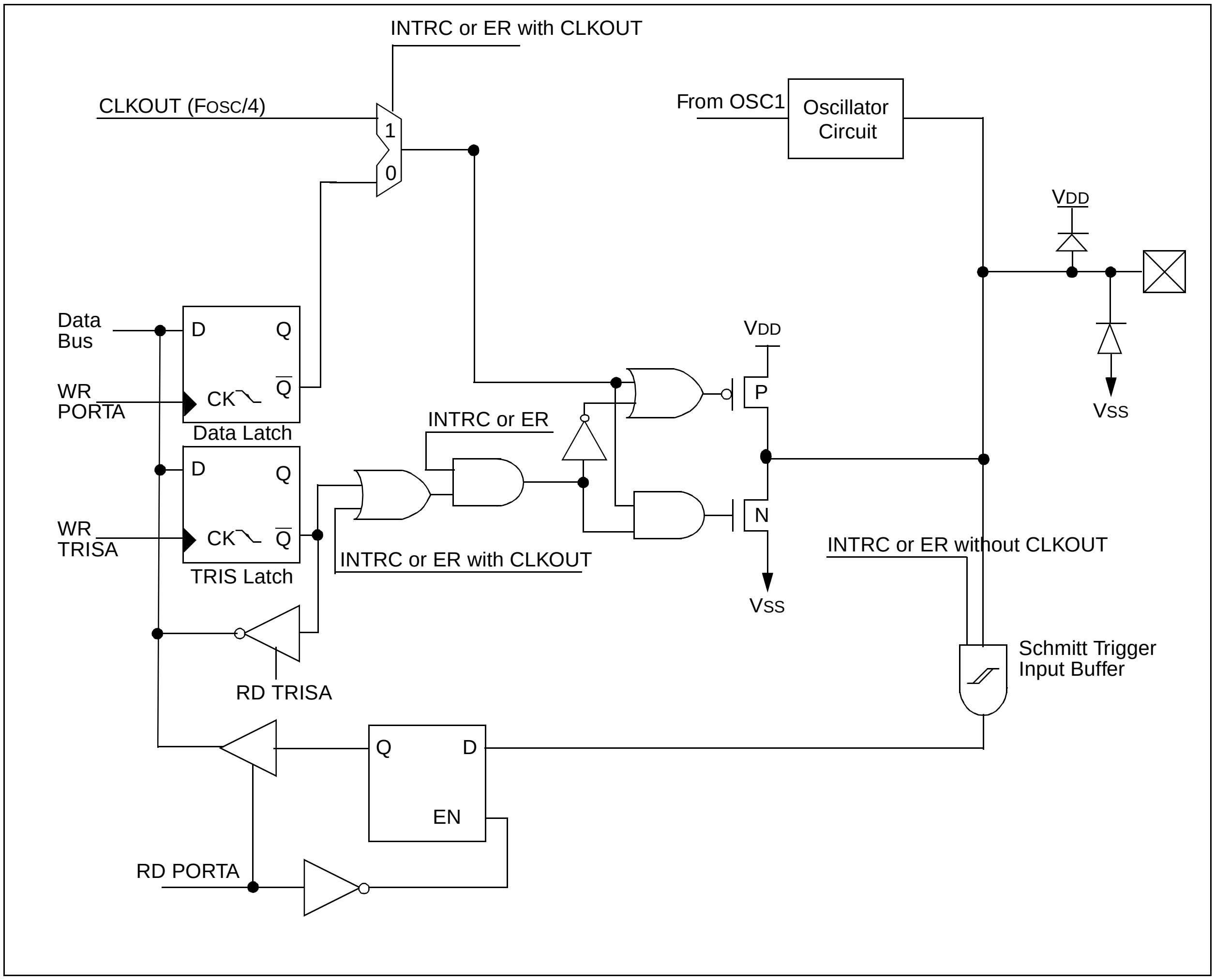


FIGURE 3-5: BLOCK DIAGRAM OF RA6/OSC2/CLKOUT PIN



 Advanced Information

FIGURE 3-6: BLOCK DIAGRAM OF RA7/OSC1/CLKIN PIN

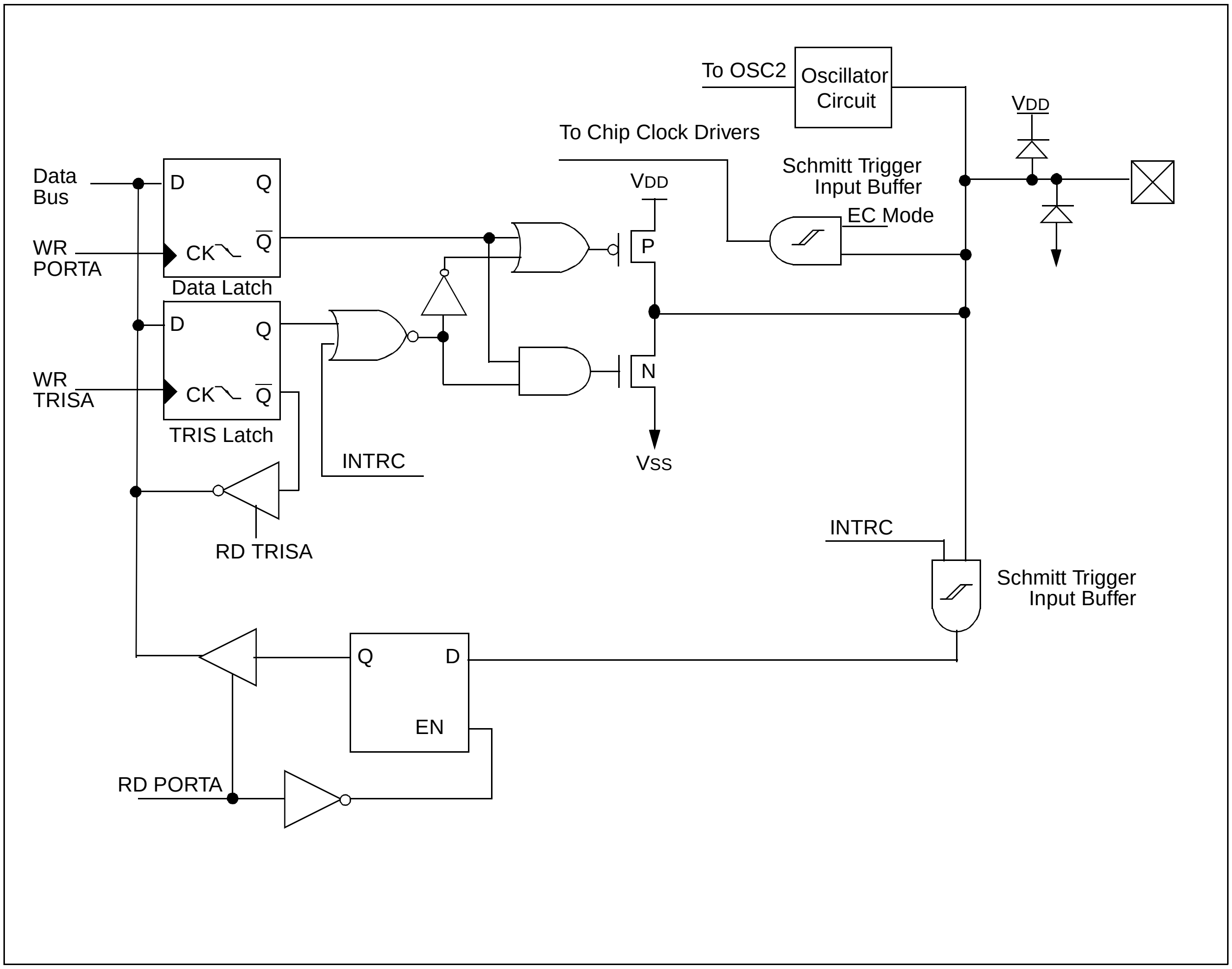


TABLE 3-1: PORTA FUNCTIONS

Name	Function	Input Type	Output Type	Description
RA0/AN0	RA0	ST	CMOS	Bi-directional I/O
	AN0	AN		A/D input
RA1/AN1/LVDIN	RA1	ST	CMOS	Bi-directional I/O
	AN1	AN		A/D input
	LVDIN	AN		LVD input reference
RA2/AN2/VREF-/VRL	RA2	ST	CMOS	Bi-directional I/O
	AN2	AN		A/D input
	VREF-	AN		Negative analog reference input
	VRL		AN	Internal voltage reference low output
RA3/AN3/VREF+/VRH	RA3	ST	CMOS	Bi-directional I/O
	AN3	AN		A/D input
	VREF+	AN		Positive analog reference input
	VRH		AN	Internal voltage reference high output
RA4/T0CKI	RA4	ST	OD	Bi-directional I/O
	T0CKI	ST		TMR0 clock input
RA5/$\overline{MCLR}$/VPP	RA5	ST		Input port
	$\overline{MCLR}$	ST		Master clear
	VPP	Power		Programming voltage
RA6/OSC2/CLKOUT	RA6	ST	CMOS	Bi-directional I/O
	OSC2		XTAL	Crystal/resonator
	CLKOUT		CMOS	FOSC/4 output
RA7/OSC1/CLKIN	RA7	ST	CMOS	Bi-directional I/O
	OSC1	XTAL		Crystal/resonator
	CLKIN	ST		External clock input/ER resistor connection

TABLE 3-2: SUMMARY OF REGISTERS ASSOCIATED WITH PORTA

Address	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
05h	PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx 0000	uuuu 0000
85h	TRISA	PORTA Data Direction Register								1111 1111	1111 1111
9Dh	ANSEL			ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111

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

3.3 PORTB and the TRISB Register

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

EXAMPLE 3-2: INITIALIZING PORTB

```
BCF    STATUS, RP0  ;
CLRF   PORTB        ; Initialize PORTB by
                    ; clearing output
                    ; data latches
BSF    STATUS, RP0  ; Select Bank 1
MOVLW  0xCF         ; Value used to
                    ; initialize data
                    ; direction
MOVWF  TRISB        ; Set RB<3:0> as inputs
                    ; RB<5:4> as outputs
                    ; RB<7:6> as inputs
MOVLW  03           ; Set RB<1:0> as analog
                      inputs
MOVWF  ANSEL        ;
BCF    STATUS, RP0  ; Return to Bank 0
```

Each of the PORTB pins has an internal pull-up, which can be individually enabled from the WPUB register. A single global enable bit can turn on/off the enabled pull-ups. Clearing the $\overline{\text{RBPU}}$ bit, (OPTION_REG<7>), enables the weak pull-up resistors. The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset.

Each of the PORTB pins, if configured as input, also has an interrupt on change feature, which can be individually selected from the IOCB register. The RBIE bit in the INTCON register functions as a global enable bit to turn on/off the interrupt on change feature. The selected inputs are compared to the old value latched on the last read of PORTB. The "mismatch" outputs are OR'ed together to generate the RB Port Change Interrupt with flag bit RBIF (INTCON<0>).

This interrupt can wake the device from SLEEP. The user, in the interrupt service routine, can clear the interrupt in the following manner:

a) Any read or write of PORTB. This will end the mismatch condition.
b) Clear flag bit RBIF.

A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition and allow flag bit RBIF to be cleared.

The interrupt on change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt on change feature. Polling of PORTB is not recommended while using the interrupt on change feature.

REGISTER 3-2: WEAK PULL UP PORTB REGISTER (WPUB: 95h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
-n = Value at POR reset

bit 7-0: **WPUB<7:0>:** PORTB Weak Pull-Up Control
1 = Weak pull up enabled.
0 = Weak pull up disabled

Note 1: For the WPUB register setting to take effect, the $\overline{\text{RBPU}}$ bit in the OPTION_REG Register must be cleared.
2: The weak pull up device is automatically disabled if the pin is in output mode (TRIS = 0).

REGISTER 3-3: INTERRUPT ON CHANGE PORTB REGISTER (IOCB: 96h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
IOCB7	IOCB6	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0
bit7							bit0

R =	Readable bit
W =	Writable bit
U =	Unimplemented bit, read as ‘0’
-n =	Value at POR reset

bit 7-0: **IOCB<7:0>: Interrupt on Change PORTB Control**
`1` = Interrupt on change enabled.
`0` = Interrupt on change disabled.

Note 1: The interrupt enable bits GIE and RBIE in the INTCON Register must be set for individual interrupts to be recognized.

The RB0 pin is multiplexed with the A/D converter analog input 4 and the external interrupt input (RB0/AN4/INT). When the pin is used as analog input, the ANSEL register must have the proper value to select the RB0 pin as analog mode.

The RB1 pin is multiplexed with the A/D converter analog input 5 and the MSSP module slave select input (RB1/AN5/SS). When the pin is used as analog input, the ANSEL register must have the proper value to select the RB1 pin as analog mode.

Note:	Upon reset, the ANSEL register configures the RB1 and RB0 pins as analog inputs. Both RB1 and RB0 pins will read as ’0’.

FIGURE 3-7: BLOCK DIAGRAM OF RB0/AN4/INT, RB1/AN5/$\overline{SS}$ PIN

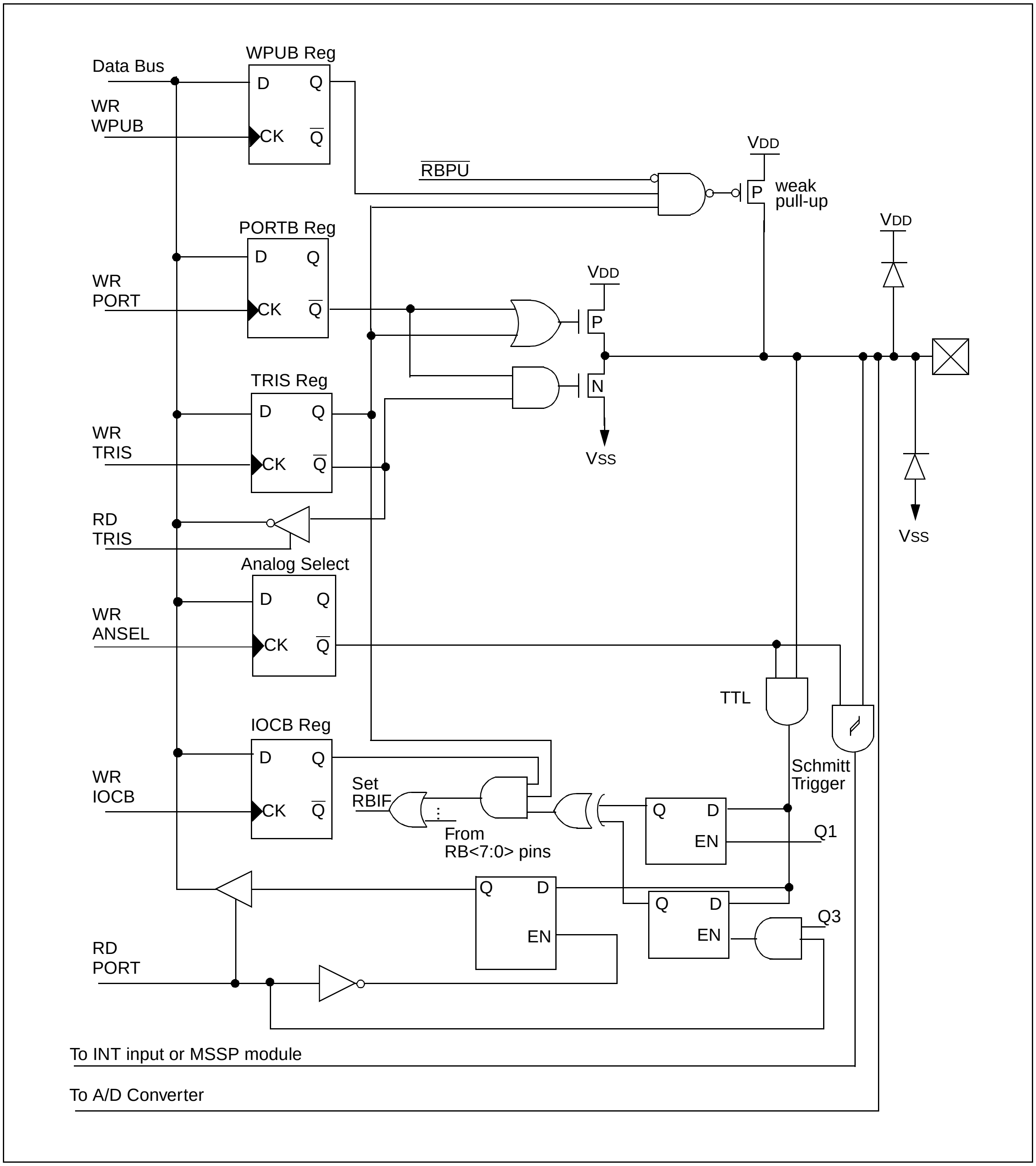
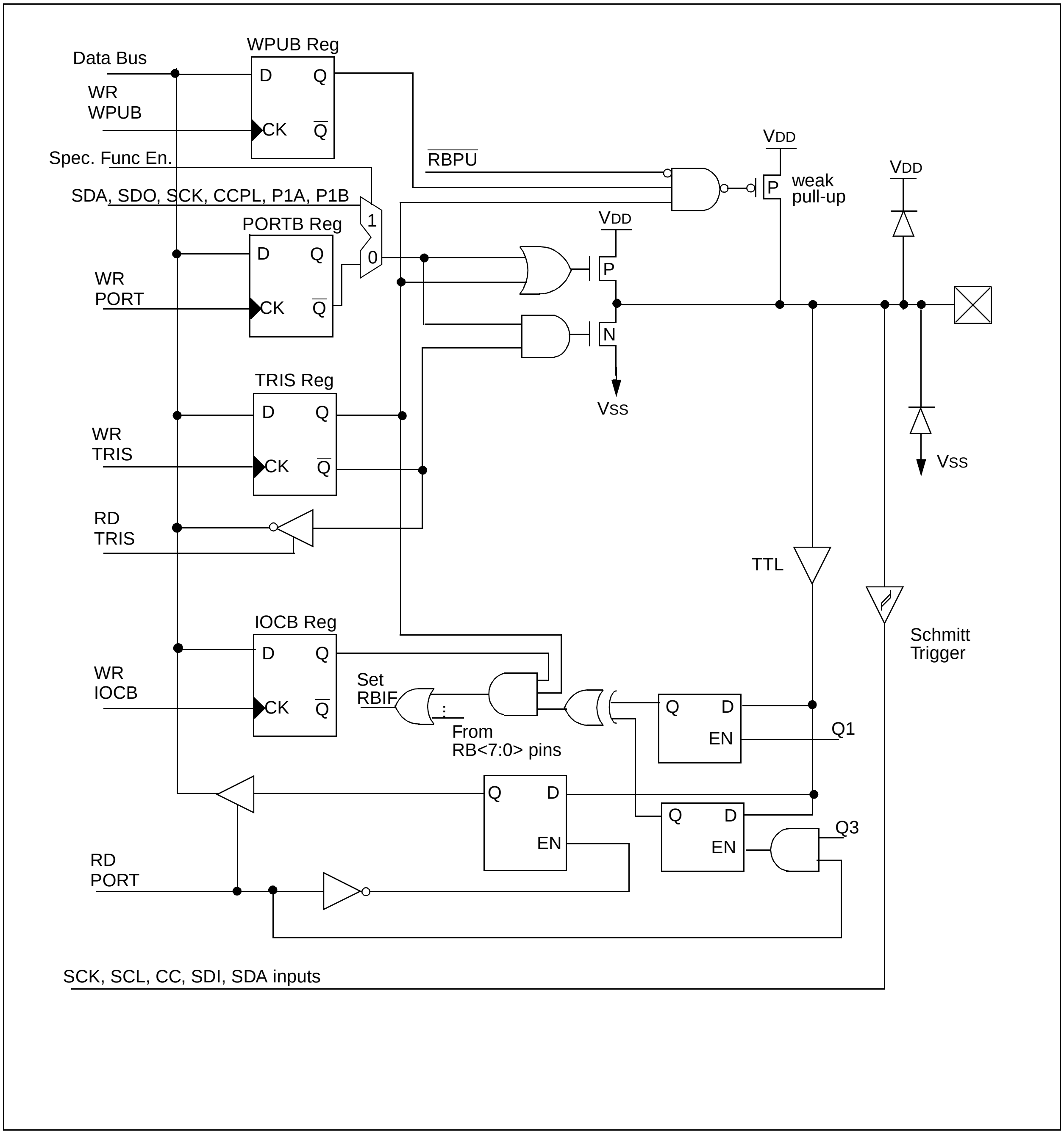
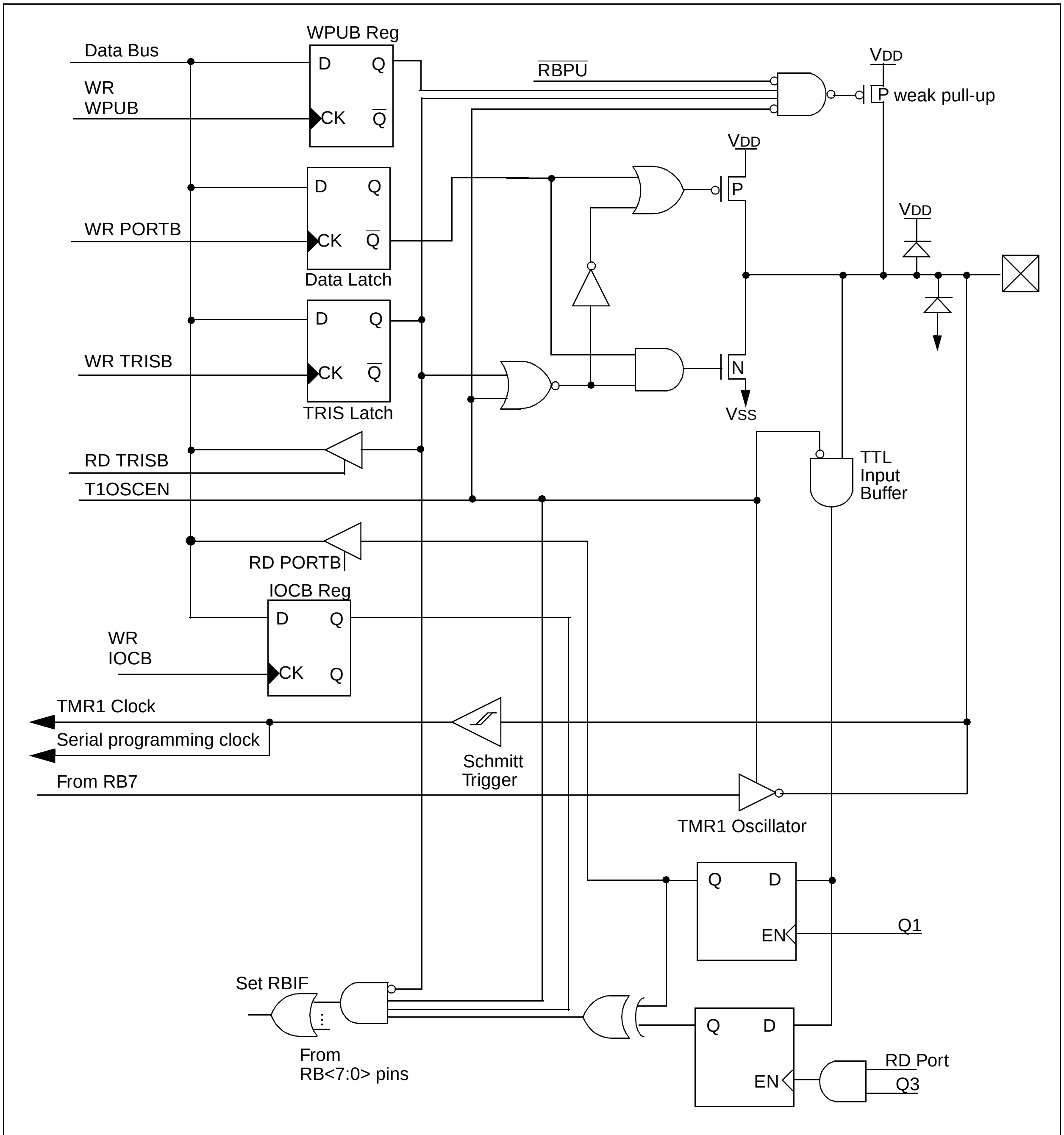


FIGURE 3-8: BLOCK DIAGRAM OF RB2/SCK/SCL, RB3/CCP1/P1A, RB4/SDI/SDA, RB5/SDO/P1B



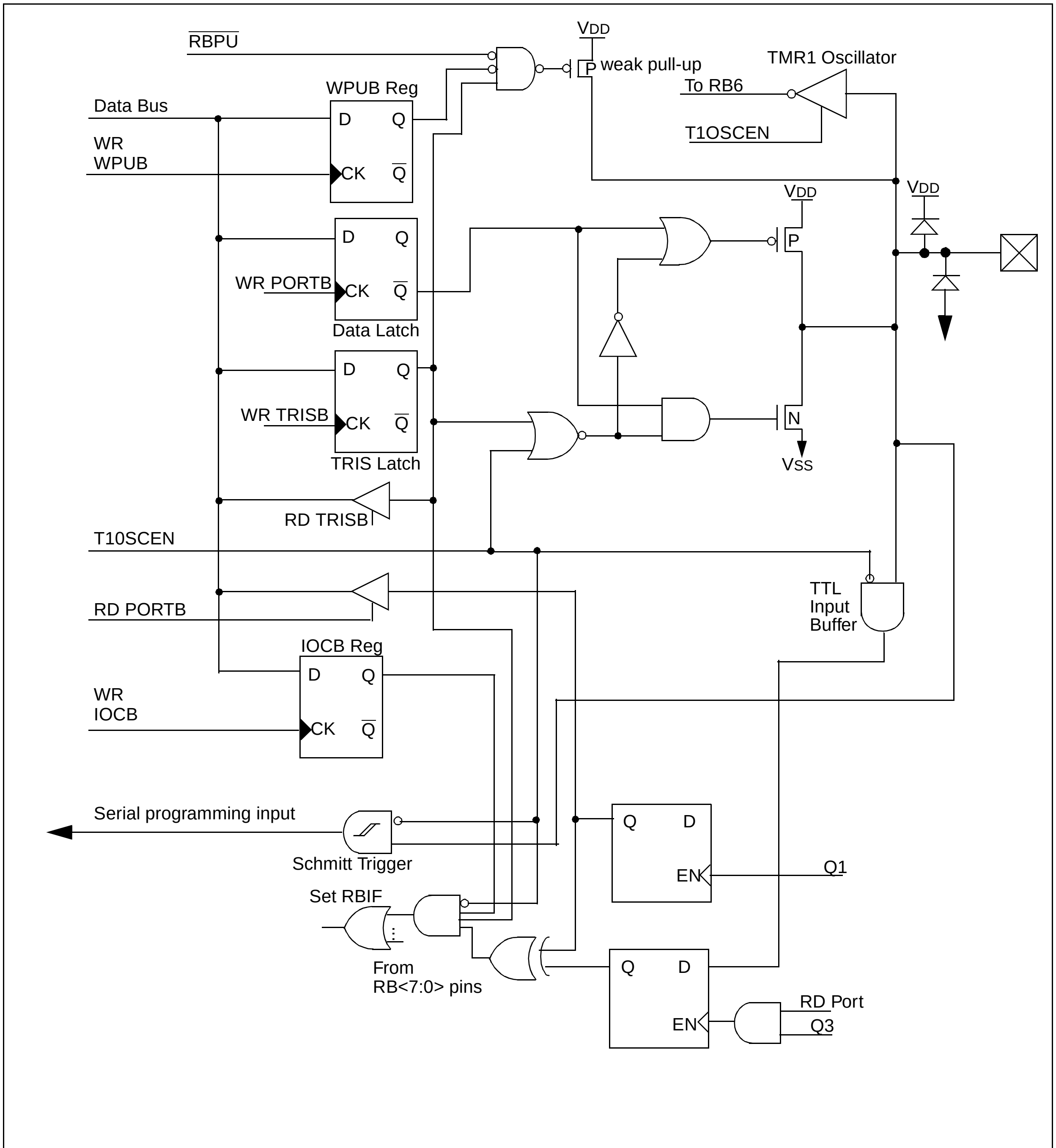
 Advanced Information

FIGURE 3-9: BLOCK DIAGRAM OF RB6/T1OSO/T1CKI/P1C



Note: The TMR1 oscillator enable (T1OSCEN = 1) overrides the RB6 I/O port and P1C functions.

FIGURE 3-10: BLOCK DIAGRAM OF THE RB7/T1OSI/P1D



Note: The TMR1 oscillator enable (T1OSCEN = 1) overrides the RB7 I/O port and P1D functions.

 Advanced Information

TABLE 3-3: PORTB FUNCTIONS

Name	Function	Input Type	Output Type	Description
RB0/AN4/INT	RB0	TTL	CMOS	Bi-directional I/O(1)
	AN4	AN		A/D input
	INT	ST		Interrupt input
RB1/AN5/$\overline{SS}$	RB1	TTL	CMOS	Bi-directional I/O(1)
	AN5	AN		A/D input
	$\overline{SS}$	ST		SSP slave select input
RB2/SCK/SCL	RB2	TTL	CMOS	Bi-directional input(1)
	SCK	ST	CMOS	Serial clock I/O for SPI
	SCL	ST	OD	Serial clock I/O for I^2C
RB3/CCP1/P1A	RB3	TTL	CMOS	Bi-directional input(1)
	CCP1	ST	CMOS	Capture 1 input/Compare 1 output
	P1A		CMOS	PWM P1A output
RB4/SDI/SDA	RB4	TTL	CMOS	Bi-directional input(1)
	SDI	ST		Serial data in for SPI
	SDA	ST	OD	Serial data I/O for I^2C
RB5/SDO/P1B	RB5	ST	CMOS	Bi-directional I/O(1)
	SDO		CMOS	Serial data out for SPI
	P1B		CMOS	PWM P1B output
RB6/T1OSO/T1CKI/P1C	RB6	TTL	CMOS	Bi-directional I/O(1)
	T1OSO		XTAL	Crystal/Resonator
	T1CKI	ST		TMR1 clock input
	P1C		CMOS	PWM P1C output
RB7/T1OSI/P1D	RB7	TTL	CMOS	Bi-directional I/O(1)
	T1OSI	XTAL		TMR1 crystal/resonator
	P1D		CMOS	PWM P1D output

Note 1: Bit programmable pull-ups.

TABLE 3-4: SUMMARY OF REGISTERS ASSOCIATED WITH PORTB

Address	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
06h, 106h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xx00	uuuu uu00
86h, 186h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
81h, 181h	OPTION_REG	$\overline{RBPU}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
95h	WPUB	PORTB Weak Pull-up Control								1111 1111	1111 1111
96h	IOCB	PORTB Interrupt on Change Control								1111 0000	1111 0000
9Dh	ANSEL			ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111

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

NOTES:

4.0 PROGRAM MEMORY READ (PMR)

Program memory is readable during normal operation (full VDD range). It is indirectly addressed through the Special Function Registers:

- PMCON1
- PMDATH
- PMDATL
- PMADRH
- PMADRL

When interfacing the program memory block, the PMDATH & PMDATL registers form a 2-byte word, which holds the 14-bit data. The PMADRH & PMADRL registers form a 2-byte word, which holds the 12-bit address of the program memory location being accessed. Mid-range devices have up to 8K words of program EPROM with an address range from 0h to 3FFFh. When the device contains less memory than the full address range of the PMADRH:PMARDL registers, the most significant bits of the PMADRH register are ignored.

4.0.1 PMCON1 REGISTER

PMCON1 is the control register for program memory accesses.

Control bit RD initiates a read operation. This bit cannot be cleared, only set, in software. It is cleared in hardware at completion of the read operation.

REGISTER 4-1: PROGRAM MEMORY READ CONTROL REGISTER 1 (PMCON1: 18Ch)

R-1	U-0	U-0	U-0	U-0	U-0	U-0	R/S-0
Reserved	—	—	—	—	—	—	RD
bit7							bit0

R = Readable bit
W = Writable bit
S = Settable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **Reserved:** Read as '1'

bit 6-1: **Unimplemented:** Read as '0'

bit 0: **RD**: Read Control bit
1 = Initiates a Program memory read (read takes 2 cycles. RD is cleared in hardware.
0 = Reserved

4.0.2 PMDATH AND PMDATL REGISTERS

The PMDATH:PMDATL registers are loaded with the contents of program memory addressed by the PMADRH and PMADRL registers upon completion of a Program Memory Read command.

REGISTER 4-2: PROGRAM MEMORY DATA HIGH (PMDATH: 10Eh)

U-0	U-0	R-x	R-x	R-x	R-x	R-x	R-x
—	—	PMD13	PMD12	PMD11	PMD10	PMD9	PMD8
bit7							bit0

R = Readable bit
W = Writable bit
S = Settable bit
U = Unimplemented bit, read as ‘0’
- n = Value at POR reset

bit 7-6: **Unimplemented:** Read as '0'

bit 5-0: **PMD<13:8>:** The value of the program memory word pointed to by PMADRH and PMADRL after a program memory read command.

REGISTER 4-3: PROGRAM MEMORY DATA LOW (PMDATL: 10Ch)

R-x	R-x	R-x	R-x	R-x	R-x	R-x	R-x
PMD7	PMD6	PMD5	PMD4	PMD3	PMD2	PMD1	PMD0
bit7							bit0

R = Readable bit
W = Writable bit
S = Settable bit
U = Unimplemented bit, read as ‘0’
- n = Value at POR reset

bit 7-0: **PMD<7:0>:** The value of the program memory word pointed to by PMADRH and PMADRL after a program memory read command.

REGISTER 4-4: PROGRAM MEMORY ADDRESS HIGH (PMADRH: 10Fh)

U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	—	PMA11	PMA10	PMA9	PMA8
bit7							bit0

R = Readable bit
W = Writable bit
S = Settable bit
U = Unimplemented bit, read as ‘0’
- n = Value at POR reset

bit 7-4: **Unimplemented:** Read as '0'

bit 3-0: **PMA<11:8>:** PMR Address bits

REGISTER 4-5: PROGRAM MEMORY ADDRESS LOW (PMADRL: 10Dh)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
PMA7	PMA6	PMA5	PMA4	PMA3	PMA2	PMA1	PMA0
bit7							bit0

R = Readable bit
W = Writable bit
S = Settable bit
U = Unimplemented bit, read as ‘0’
- n = Value at POR reset

bit 7-0: **PMA<7:0>:** PMR Address bits

4.0.3 READING THE EPROM PROGRAM MEMORY

To read a program memory location, the user must write 2 bytes of the address to the PMADRH and PMADRL registers, then set control bit RD (PMCON1<0>). Once the read control bit is set, the Program Memory Read (PMR) controller will use the second instruction cycle after to read the data. This causes the second instruction immediately following the "`BSF PMCON1,RD`" instruction to be ignored. The data is available, in the very next cycle, in the PMDATH and PMDATL registers; therefore it can be read as 2 bytes in the following instructions. PMDATH and PMDATL registers will hold this value until another read or until it is written to by the user.

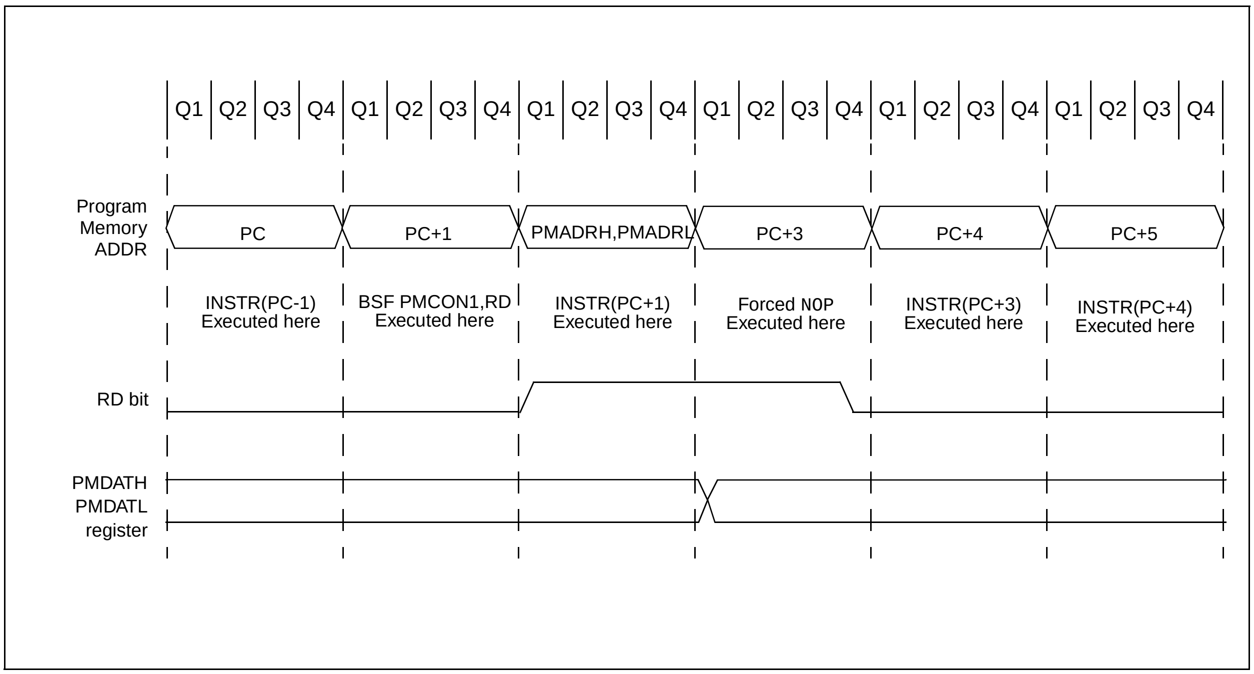
EXAMPLE 4-1: OTP PROGRAM MEMORY READ

```
BSF     STATUS, RP1      ;
BCF     STATUS, RP0      ; Bank 2
MOVLW   MS_PROG_PM_ADDR  ;
MOVWF   PMADRH           ; MS Byte of Program Memory Address to read
MOVLW   LS_PROG_PM_ADDR  ;
MOVWF   PMADRL           ; LS Byte of Program Memory Address to read
BSF     STATUS, RP0      ; Bank 3
BSF     PMCON1, RD       ; Program Memory Read
NOP                      ; This instruction is executed
NOP                      ; This instruction must be a NOP
next instruction         ; PMDATH:PMDATL now has the data
```

4.0.4 OPERATION DURING CODE PROTECT

When the device is code protected, the CPU can still perform the program memory read function.

FIGURE 4-1: PROGRAM MEMORY READ CYCLE EXECUTION



NOTES:

5.0 TIMER0 MODULE

The Timer0 module timer/counter has the following features:

- 8-bit timer/counter
- Readable and writable
- Internal or external clock select
- Edge select for external clock
- 8-bit software programmable prescaler
- Interrupt on overflow from FFh to 00h

Figure 5-1 is a simplified block diagram of the Timer0 module.

Additional information on timer modules is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

5.1 Timer0 Operation

Timer0 can operate as a timer or as a counter.

Timer mode is selected by clearing bit T0CS (OPTION_REG<5>). In timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If the TMR0 register is written, the increment is inhibited for the following two instruction cycles. The user can work around this by writing an adjusted value to the TMR0 register.

Counter mode is selected by setting bit T0CS (OPTION_REG<5>). In counter mode, Timer0 will increment either on every rising or falling edge of pin RA4/T0CKI. The incrementing edge is determined by the Timer0 Source Edge Select bit T0SE (OPTION_REG<4>). Clearing bit T0SE selects the rising edge. Restrictions on the external clock input are discussed in below.

When an external clock input is used for Timer0, it must meet certain requirements. The requirements ensure the external clock can be synchronized with the internal phase clock (TOSC). Also, there is a delay in the actual incrementing of Timer0 after synchronization.

Additional information on external clock requirements is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

5.2 Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module, or as a postscaler for the Watchdog Timer, respectively (Figure 5-2). For simplicity, this counter is being referred to as “prescaler” throughout this data sheet. Note that there is only one prescaler available which is mutually exclusively shared between the Timer0 module and the Watchdog Timer. Thus, a prescaler assignment for the Timer0 module means that there is no prescaler for the Watchdog Timer, and vice-versa.

The prescaler is not readable or writable.

The PSA and PS<2:0> bits (OPTION_REG<3:0>) determine the prescaler assignment and prescale ratio.

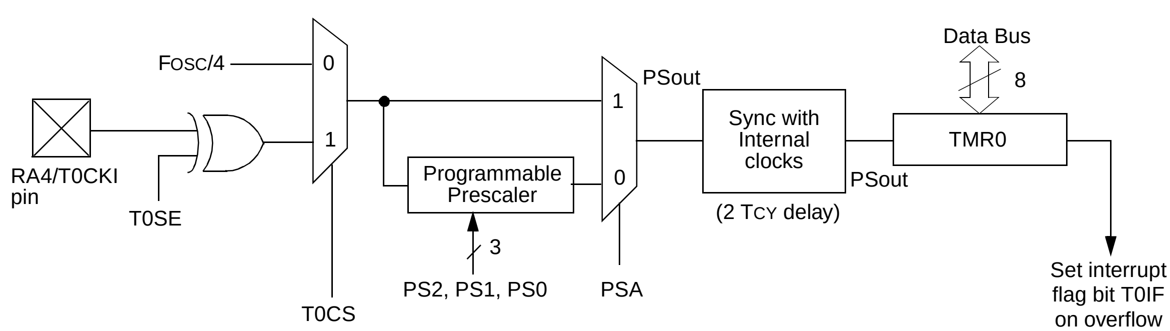
Clearing bit PSA will assign the prescaler to the Timer0 module. When the prescaler is assigned to the Timer0 module, prescale values of 1:2, 1:4, ..., 1:256 are selectable.

Setting bit PSA will assign the prescaler to the Watchdog Timer (WDT). When the prescaler is assigned to the WDT, prescale values of 1:1, 1:2, ..., 1:128 are selectable.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g. `CLRF 1, MOVWF 1, BSF 1, x....etc.`) will clear the prescaler. When assigned to WDT, a `CLRWDT` instruction will clear the prescaler along with the WDT.

Note:	Writing to TMR0 when the prescaler is assigned to Timer0 will clear the prescaler count, but will not change the prescaler assignment.

FIGURE 5-1: TIMER0 BLOCK DIAGRAM



Note 1: T0CS, T0SE, PSA, PS<2:0> (OPTION_REG<5:0>).
2: The prescaler is shared with Watchdog Timer (refer to Figure 5-2 for detailed block diagram).

5.2.1 SWITCHING PRESCALER ASSIGNMENT

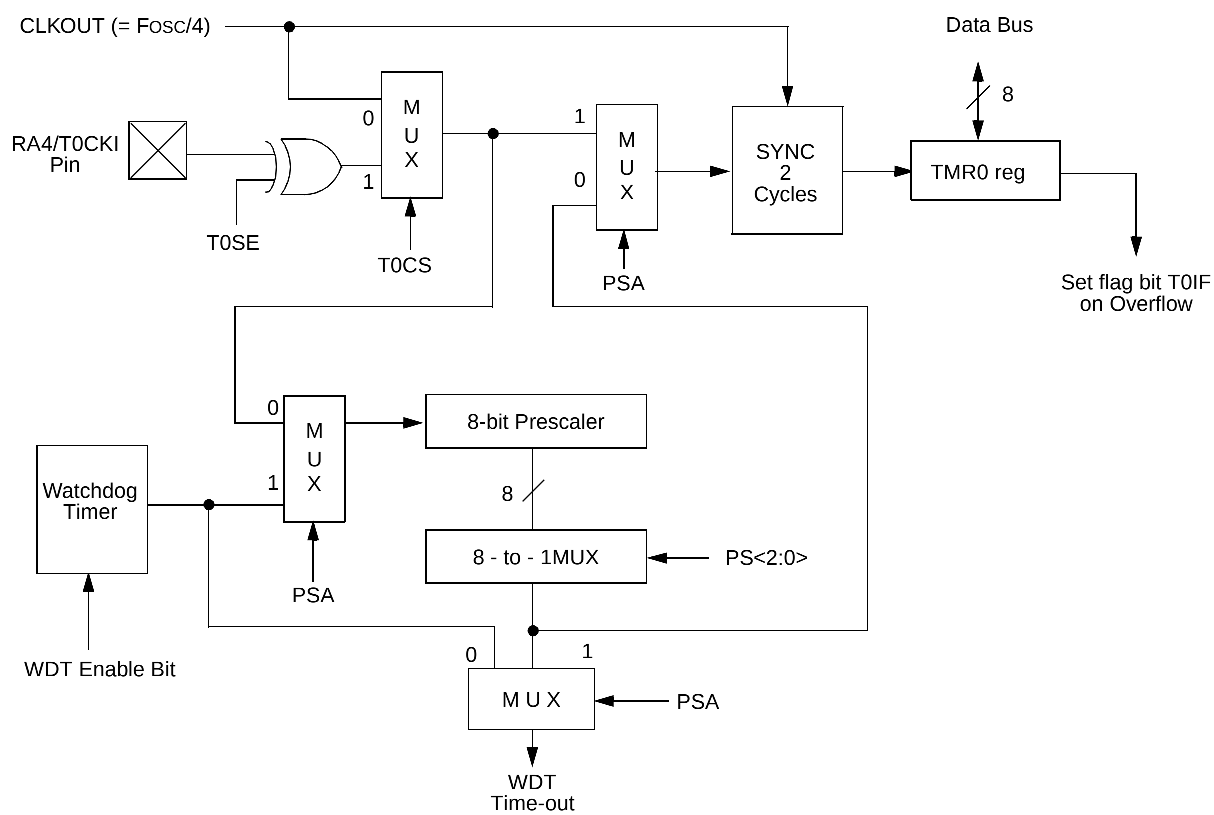
The prescaler assignment is fully under software control, i.e., it can be changed “on-the-fly” during program execution.

Note:	To avoid an unintended device RESET, a specific instruction sequence (shown in the PICmicro™ Mid-Range Reference Manual, DS33023) must be executed when changing the prescaler assignment from Timer0 to the WDT. This sequence must be followed even if the WDT is disabled.

5.3 Timer0 Interrupt

The TMR0 interrupt is generated when the TMR0 register overflows from FFh to 00h. This overflow sets bit T0IF (INTCON<2>). The interrupt can be masked by clearing bit T0IE (INTCON<5>). Bit T0IF must be cleared in software by the Timer0 module interrupt service routine before re-enabling this interrupt. The TMR0 interrupt cannot awaken the processor from SLEEP since the timer is shut off during SLEEP.

FIGURE 5-2: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER



Note: T0CS, T0SE, PSA, PS<2:0> are (OPTION_REG<5:0>).

TABLE 5-1: REGISTERS ASSOCIATED WITH TIMER0

Address	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
01h,101h	TMR0	Timer0 register								xxxx xxxx	uuuu uuuu
0Bh,8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
81h,181h	OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
85h	TRISA	PORTA Data Direction Register								1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented locations read as '0'. Shaded cells are not used by Timer0.

6.0 TIMER1 MODULE

The Timer1 module timer/counter has the following features:

- 16-bit timer/counter
  (Two 8-bit registers; TMR1H and TMR1L)
- Readable and writable (Both registers)
- Internal or external clock select
- Interrupt on overflow from FFFFh to 0000h
- Reset from ECCP module trigger

Timer1 has a control register, shown in Register 6-1. Timer1 can be enabled/disabled by setting/clearing control bit TMR1ON (T1CON<0>).

Figure 6-2 is a simplified block diagram of the Timer1 module.

Additional information on timer modules is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

6.1 Timer1 Operation

Timer1 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, TMR1CS (T1CON<1>).

In timer mode, Timer1 increments every instruction cycle. In counter mode, it increments on every rising edge of the external clock input.

When the Timer1 oscillator is enabled (T1OSCEN is set), the RB7/T1OSI/P1D and RB6/T1OSO/T1CKI/P1C pins are no longer available as I/O ports or PWM outputs. That is, the TRISB<7:6> value is ignored.

Timer1 also has an internal “reset input”. This reset can be generated by the ECCP module (Section 7.0).

REGISTER 6-1: TIMER1 CONTROL REGISTER (T1CON: 10h)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as ‘0’
- n = Value at POR reset

bit 7-6: **Unimplemented:** Read as ’0’

bit 5-4: **T1CKPS<1:0>:** Timer1 Input Clock Prescale Select bits
11 = 1:8 Prescale value
10 = 1:4 Prescale value
01 = 1:2 Prescale value
00 = 1:1 Prescale value

bit 3: **T1OSCEN:** Timer1 Oscillator Enable Control bit
1 = Oscillator is enabled
0 = Oscillator is shut off
Note: The oscillator inverter and feedback resistor are turned off to eliminate power drain

bit 2: **$\overline{\text{T1SYNC}}$:** Timer1 External Clock Input Synchronization Control bit

TMR1CS = 1
1 = Do not synchronize external clock input
0 = Synchronize external clock input

TMR1CS = 0
This bit is ignored. Timer1 uses the internal clock when TMR1CS = 0.

bit 1: **TMR1CS:** Timer1 Clock Source Select bit
1 = External clock from pin RB6/T1OSO/T1CKI /P1C(on the rising edge)
0 = Internal clock (FOSC/4)

bit 0: **TMR1ON:** Timer1 On bit
1 = Enables Timer1
0 = Stops Timer1

6.1.1 TIMER1 COUNTER OPERATION

In this mode, Timer1 is being incremented via an external source. Increments occur on a rising edge. After Timer1 is enabled in counter mode, the module must first have a falling edge before the counter begins to increment.

FIGURE 6-1: TIMER1 INCREMENTING EDGE

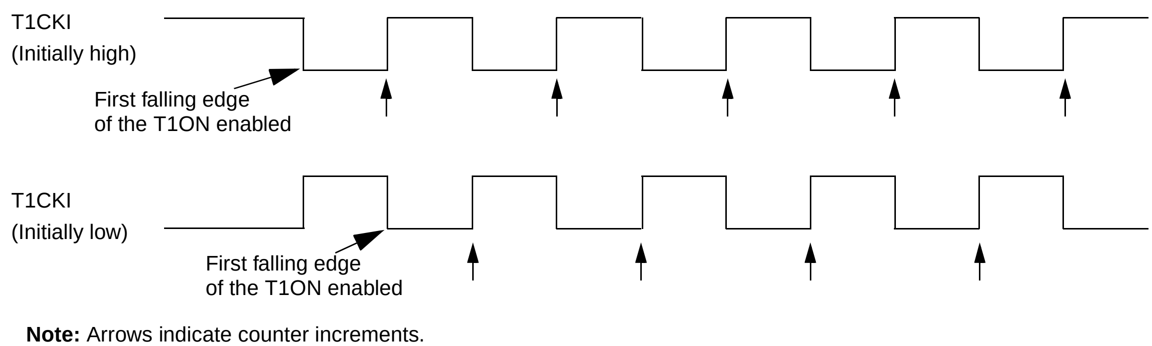
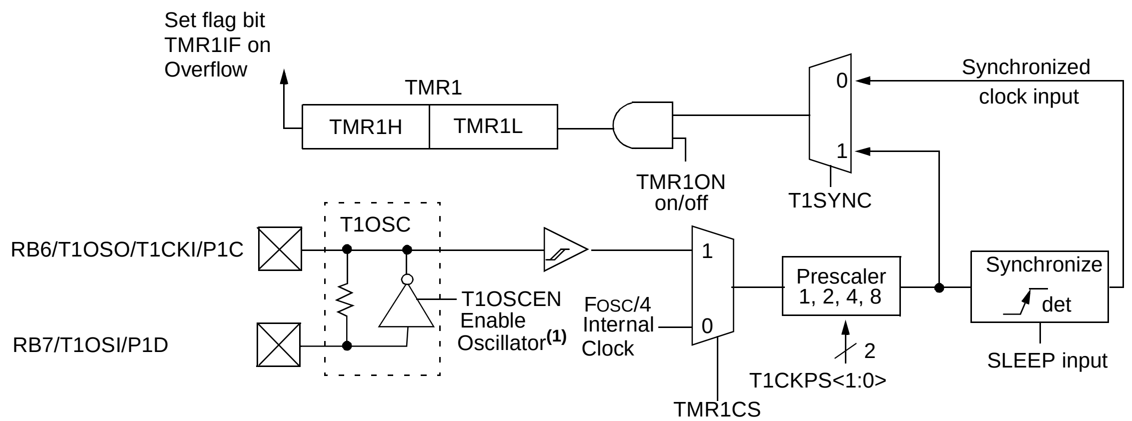


FIGURE 6-2: TIMER1 BLOCK DIAGRAM



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

6.2 Timer1 Oscillator

A crystal oscillator circuit is built in between pins T1OSI (input) and T1OSO (amplifier output). It is enabled by setting control bit T1OSCEN (T1CON<3>). The oscillator is a low power oscillator rated up to 200 kHz. It will continue to run during SLEEP. It is primarily intended for a 32 kHz crystal. Table 6-1 shows the capacitor selection for the Timer1 oscillator.

The Timer1 oscillator is identical to the LP oscillator. The user must provide a software time delay to ensure proper oscillator start-up.

TABLE 6-1: CAPACITOR SELECTION FOR THE TIMER1 OSCILLATOR

Osc Type	Freq	C1	C2
LP	32 kHz	33 pF	33 pF
	100 kHz	15 pF	15 pF
	200 kHz	15 pF	15 pF
These values are for design guidance only.			

Note 1: Higher capacitance increases the stability of oscillator but also increases the start-up time.

2: Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components.

6.3 Timer1 Interrupt

The TMR1 Register pair (TMR1H:TMR1L) increments from 0000h to FFFFh and rolls over to 0000h. The TMR1 Interrupt, if enabled, is generated on overflow which is latched in interrupt flag bit TMR1IF (PIR1<0>). This interrupt can be enabled/disabled by setting/clearing TMR1 interrupt enable bit TMR1IE (PIE1<0>).

6.4 Resetting Timer1 using a CCP Trigger Output

If the ECCP module is configured in compare mode to generate a "special event trigger" (CCP1M<3:0> = `1011`), this signal will reset Timer1 and start an A/D conversion (if the A/D module is enabled).

Note: The special event triggers from the CCP1 module will not set interrupt flag bit TMR1IF (PIR1<0>).

Timer1 must be configured for either timer or synchronized counter mode to take advantage of this feature. If Timer1 is running in asynchronous counter mode, this reset operation may not work.

In the event that a write to Timer1 coincides with a special event trigger from ECCP1, the write will take precedence.

In this mode of operation, the CCPR1H:CCPR1L registers pair effectively becomes the period register for Timer1.

TABLE 6-2: REGISTERS ASSOCIATED WITH TIMER1 AS A TIMER/COUNTER

Address	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
0Bh,8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF	-0-- 0000	-0-- 0000
8Ch	PIE1	—	ADIE	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE	-0-- 0000	-0-- 0000
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 register								xxxx xxxx	uuuu uuuu
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	--00 0000	--uu uuuu

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

NOTES:

7.0 TIMER2 MODULE

The Timer2 module timer has the following features:

- 8-bit timer (TMR2 register)
- 8-bit period register (PR2)
- Readable and writable (Both registers)
- Software programmable prescaler (1:1, 1:4, 1:16)
- Software programmable postscaler (1:1 to 1:16)
- Interrupt on TMR2 match of PR2
- SSP module optional use of TMR2 output to generate clock shift

Timer2 has a control register, shown in Register 7-1. Timer2 can be shut off by clearing control bit TMR2ON (T2CON<2>) to minimize power consumption.

Figure 7-1 is a simplified block diagram of the Timer2 module.

Additional information on timer modules is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

7.1 Timer2 Operation

Timer2 can be used as the PWM time-base for PWM mode of the ECCP module.

The TMR2 register is readable and writable, and is cleared on any device reset.

The input clock (FOSC/4) has a prescale option of 1:1, 1:4 or 1:16, selected by control bits T2CKPS<1:0> (T2CON<1:0>).

The match output of TMR2 goes through a 4-bit postscaler (which gives a 1:1 to 1:16 scaling inclusive) to generate a TMR2 interrupt (latched in flag bit TMR2IF, (PIR1<1>)).

The prescaler and postscaler counters are cleared when any of the following occurs:

- a write to the TMR2 register
- a write to the T2CON register
- any device reset (Power-on Reset, $\overline{\text{MCLR}}$ Reset, Watchdog Timer Reset, or Brown-out Reset)

TMR2 is not cleared when T2CON is written.

REGISTER 7-1: TIMER2 CONTROL REGISTER (T2CON1: 12h)

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **Unimplemented:** Read as '0'

bit 6-3: **TOUTPS<3:0>:** Timer2 Output Postscale Select bits
`0000` = 1:1 Postscale
`0001` = 1:2 Postscale
•
•
•
`1111` = 1:16 Postscale

bit 2: **TMR2ON:** Timer2 On bit
`1` = Timer2 is on
`0` = Timer2 is off

bit 1-0: **2CKPS<1:0>:** Timer2 Clock Prescale Select bits
`00` = Prescaler is 1
`01` = Prescaler is 4
`1x` = Prescaler is 16

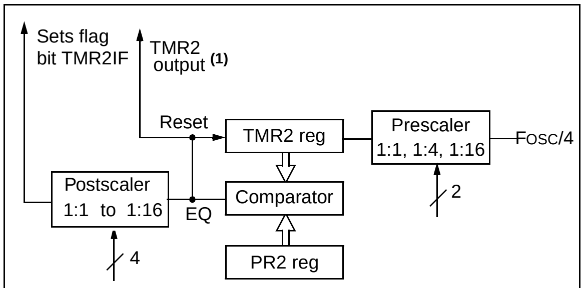
7.2 Timer2 Interrupt

The Timer2 module has an 8-bit period register PR2. Timer2 increments from 00h until it matches PR2 and then resets to 00h on the next increment cycle. PR2 is a readable and writable register. The PR2 register is initialized to FFh upon reset.

7.3 Output of TMR2

The output of TMR2 (before the postscaler) is fed to the Synchronous Serial Port module which optionally uses it to generate shift clock.

FIGURE 7-1: TIMER2 BLOCK DIAGRAM



Note 1: TMR2 register output can be software selected by the SSP Module as a baud clock.

TABLE 7-1: REGISTERS ASSOCIATED WITH TIMER2 AS A TIMER/COUNTER

Address	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
0Bh,8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF	-0-- 0000	-0-- 0000
8Ch	PIE1	—	ADIE	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE	-0-- 0000	-0-- 0000
11h	TMR2	Timer2 register								0000 0000	0000 0000
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
92h	PR2	Timer2 Period Register								1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by the Timer2 module.

 Advanced Information

8.0 ENHANCED CAPTURE/ COMPARE/PWM(ECCP) MODULES

The ECCP (Enhanced Capture/Compare/PWM) module contains a 16-bit register which can operate as a 16-bit capture register, as a 16-bit compare register or as a PWM master/slave Duty Cycle register. Table 8-1 shows the timer resources of the ECCP module modes.

Capture/Compare/PWM Register1 (CCPR1) is comprised of two 8-bit registers: CCPR1L (low byte) and CCPR1H (high byte). The CCP1CON and P1DEL registers control the operation of ECCP. All are readable and writable.

REGISTER 8-1: CCP1 CONTROL REGISTER (CCP1CON: 17h)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PWM1M1	PWM1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0
bit7							bit0

R = Readable bit
W= Writable bit
U = Unimplemented bit, read as ‘0’
- n = Value at POR reset

bit 7-6: **PWM1M<1:0>:** PWM Output Configuration

IF CCP1M<3:2> = `00, 01, 10`
`xx` - P1A assigned as Capture/Compare input. P1B, P1C, P1D assigned as Port pins.

IF CCP1M<3:2> = `11`
`00` - Single output. P1A modulated. P1B, P1C, P1D assigned as Port pins.
`01` - Full-bridge output forward. P1D modulated. P1A active. P1B, P1C inactive.
`10` - Half-bridge output. P1A, P1B modulated with deadband control. P1C, P1D assigned as Port pins.
`11` - Full-bridge output reverse. P1B modulated. P1C active. P1A, P1D inactive.

bit 5-4: **DC1B<1:0>:** PWM Duty Cycle Least Significant bits
Capture Mode: Unused
Compare Mode: Unused
PWM Mode: These bits are the two LSbs of the PWM duty cycle. The eight MSbs are found in CCPRnL.

bit 3-0: **CCP1M<3:0>:** ECCP1 Mode Select bits
`0000` = Capture/Compare/PWM off (resets ECCP module)
`0001` = Unused (reserved)
`0010` = Compare mode, toggle output on match (CCP1IF bit is set)
`0011` = Unused (reserved)
`0100` = Capture mode, every falling edge
`0101` = Capture mode, every rising edge
`0110` = Capture mode, every 4th rising edge
`0111` = Capture mode, every 16th rising edge
`1000` = Compare mode, set output on match (CCP1IF bit is set)
`1001` = Compare mode, clear output on match (CCP1IF bit is set)
`1010` = Compare mode, generate software interrupt on match (CCP1IF bit is set, CCP1 pin is unaffected)
`1011` = Compare mode, trigger special event (CCP1IF bit is set; ECCP resets TMR1, and starts an A/D conversion, if the A/D module is enabled.)
`1100` = PWM mode. P1A, P1C active high. P1B, P1D active high.
`1101` = PWM mode. P1A, P1C active high. P1B, P1D active low.
`1110` = PWM mode. P1A, P1C active low. P1B, P1D active high.
`1111` = PWM mode. P1A, P1C active low. P1B, P1D active low.

TABLE 8-1: ECCP MODE - TIMER RESOURCE

ECCP1 Mode	Timer Resource
Capture	Timer1
Compare	Timer1
PWM	Timer2

8.1 Capture Mode

In Capture mode, CCPR1H:CCPR1L captures the 16-bit value of the TMR1 register when an event occurs on pin CCP1. An event is defined as:

- every falling edge
- every rising edge
- every 4th rising edge
- every 16th rising edge

An event is selected by control bits CCP1M<3:0> (CCP1CON<3:0>). When a capture is made, the interrupt request flag bit CCP1IF (PIR1<2>) is set. It must be cleared in software. If another capture occurs before the value in register CCPR1 is read, the old captured value will be lost.

8.1.1 CCP1 PIN CONFIGURATION

In Capture mode, the CCP1 pin should be configured as an input by setting the TRISB<3> bit.

Note:	If the RB3/CCP1/P1A pin is configured as an output, a write to the port can cause a capture condition.

8.1.2 TIMER1 MODE SELECTION

Timer1 must be running in timer mode or synchronized counter mode. In asynchronous counter mode, the capture operation may not work.

8.1.3 SOFTWARE INTERRUPT

When the Capture mode is changed, a false capture interrupt may be generated. The user should keep bit CCP1IE (PIE1<2>) clear to avoid false interrupts and should clear the flag bit CCP1IF following any such change in operating mode.

8.1.4 ECCP PRESCALER

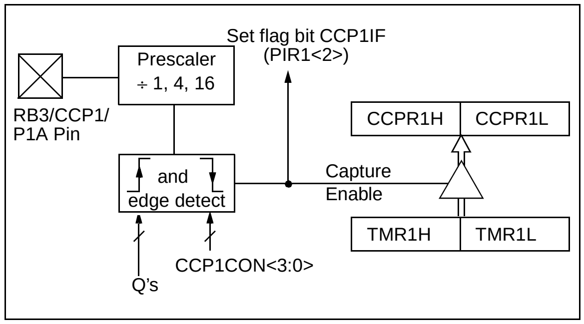
There are four prescaler settings, specified by bits CCP1M<3:0>. Whenever the ECCP module is turned off or the ECCP1 module is not in capture mode, the prescaler counter is cleared. This means that any reset will clear the prescaler counter.

Switching from one capture prescaler to another may generate an interrupt. Also, the prescaler counter will not be cleared, therefore the first capture may be from a non-zero prescaler. Example 8-1 shows the recommended method for switching between capture prescalers. This example also clears the prescaler counter and will not generate the “false” interrupt.

EXAMPLE 8-1: CHANGING BETWEEN CAPTURE PRESCALERS

```
CLRF    CCP1CON, F ; Turn ECCP module off
MOVLW   NEW_CAPT_PS ; Load WREG with the
                   ; new prescaler mode
                   ; value and ECCP ON
MOVWF   CCP1CON    ; Load CCP1CON with
                   ; this value
```

FIGURE 8-1: CAPTURE MODE OPERATION BLOCK DIAGRAM



8.2 Compare Mode

In Compare mode, the 16-bit CCPR1 register value is constantly compared against the TMR1 register pair value. When a match occurs, the CCP1 pin is:

- driven High
- driven Low
- toggle output (High to Low or Low to High)
- remains Unchanged

The action on the pin is based on the value of control bits CCP1M<3:0>. At the same time, interrupt flag bit CCP1IF is set.

8.2.1 CCP1 PIN CONFIGURATION

The user must configure the CCP1 pin as an output by clearing the appropriate TRISB bit.

Note:	Clearing the CCP1CON register will force the CCP1 compare output latch to the default low level. This is not the port data latch.

8.2.2 TIMER1 MODE SELECTION

Timer1 must be running in Timer mode or Synchronized Counter mode if the ECCP module is using the compare feature. In Asynchronous Counter mode, the compare operation may not work.

8.2.3 SOFTWARE INTERRUPT MODE

When generate software interrupt is chosen, the CCP1 pin is not affected. Only an ECCP interrupt is generated (if enabled).

8.2.4 SPECIAL EVENT TRIGGER

In this mode, an internal hardware trigger is generated, which may be used to initiate an action.

The special event trigger output of ECCP resets the TMR1 register pair. This allows the CCPR1 register to effectively be a 16-bit programmable period register for Timer1.

The special event trigger output of ECCP module will also start an A/D conversion if the A/D module is enabled.

Note:	The special event trigger will not set the interrupt flag bit TMR1IF (PIR1<0>).

FIGURE 8-2: COMPARE MODE OPERATION BLOCK DIAGRAM

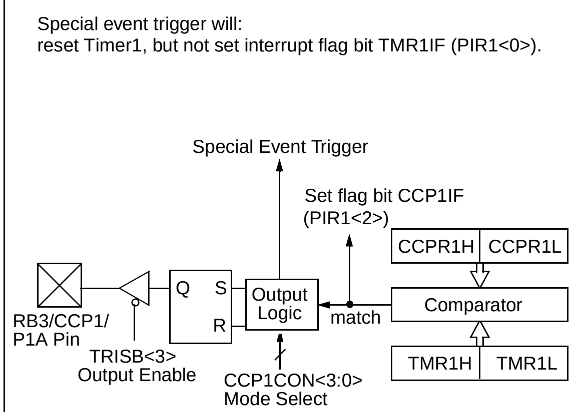


TABLE 8-2: REGISTERS ASSOCIATED WITH CAPTURE, COMPARE AND TIMER1

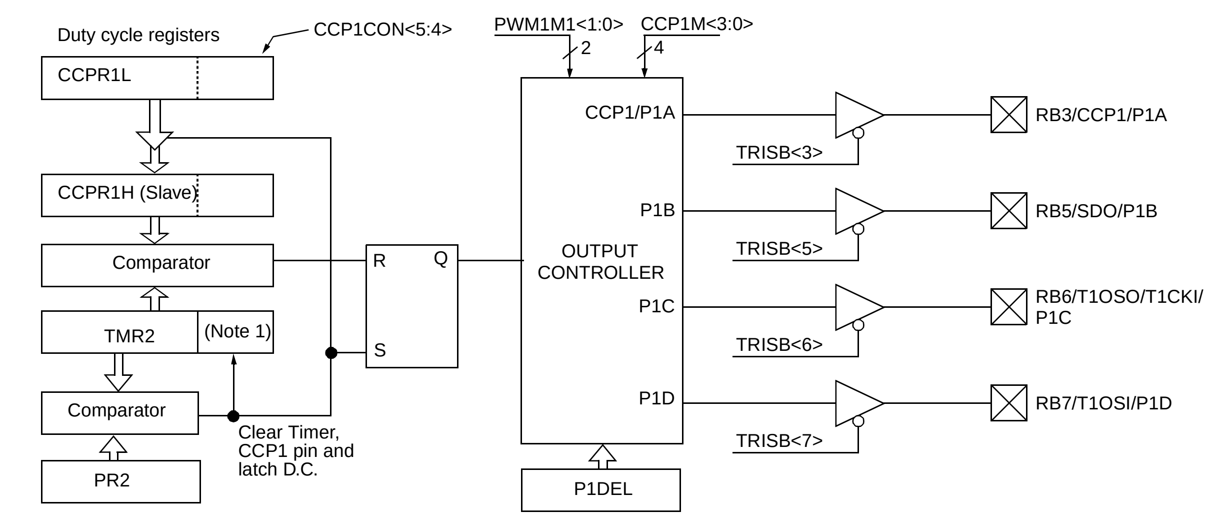
Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other resets
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
PIR1	PSPIF(1)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIE1	PSPIE(1)	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
TRISB	PORTB Data Direction Register								1111 1111	1111 1111
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 TMR1register								xxxx xxxx	uuuu uuuu
T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON	--00 0000	--uu uuuu
CCPR1L	Capture/Compare/PWM register1 (LSB)								xxxx xxxx	uuuu uuuu
CCPR1H	Capture/Compare/PWM register1 (MSB)								xxxx xxxx	uuuu uuuu
CCP1CON	PWM1M1	PWM1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	0000 0000

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

8.3 PWM Mode

In Pulse Width Modulation (PWM) mode, the ECCP module produces up to a 10-bit resolution PWM output. Figure 8-3 shows the simplified PWM block diagram.

FIGURE 8-3: SIMPLIFIED PWM BLOCK DIAGRAM



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

8.3.1 PWM PERIOD

The PWM period is specified by writing to the PR2 register. The PWM period can be calculated using the following formula:

$$\text{PWM PERIOD} = [(PR2) + 1] \cdot 4 \cdot T_{OSC} \cdot (\text{TMR2 PRESCALE VALUE})$$

PWM frequency is defined as 1 / [PWM period].

When TMR2 is equal to PR2, the following three events occur on the next increment cycle:

- TMR2 is cleared
- The CCP1 pin is set (exception: if PWM duty cycle = 0%, the CCP1 pin will not be set)
- The PWM duty cycle is latched from CCPR1L into CCPR1H

Note:	The Timer2 postscaler (see Section 7.0) is not used in the determination of the PWM frequency. The postscaler could be used to have a servo update rate at a different frequency than the PWM output.

 Advanced Information

8.3.2 PWM DUTY CYCLE

The PWM duty cycle is specified by writing to the CCPR1L register and to the CCP1CON<5:4> bits. Up to 10-bit resolution is available. The CCPR1L contains the eight MSbs and the CCP1CON<5:4> contains the two LSbs. This 10-bit value is represented by CCPR1L:CCP1CON<5:4>. The following equation is used to calculate the PWM duty cycle in time:

$$\text{PWM duty cycle} = (\text{CCPR1L:CCP1CON<5:4>}) \cdot T_{OSC} \cdot (\text{TMR2 prescale value})$$

CCPR1L and CCP1CON<5:4> can be written to at any time, but the duty cycle value is not latched into CCPR1H until after a match between PR2 and TMR2 occurs (i.e., the period is complete). In PWM mode, CCPR1H is a read-only register.

The CCPR1H register and a 2-bit internal latch are used to double buffer the PWM duty cycle. This double buffering is essential for glitchless PWM operation.

When the CCPR1H and 2-bit latch match TMR2 concatenated with an internal 2-bit Q clock or 2 bits of the TMR2 prescaler, the CCP1 pin is cleared.

Maximum PWM resolution (bits) for a given PWM frequency:

$$= \frac{\log\left(\frac{F_{OSC}}{F_{PWM}}\right)}{\log(2)} \text{bits}$$

Note:	If the PWM duty cycle value is longer than the PWM period, the CCP1 pin will not be cleared.

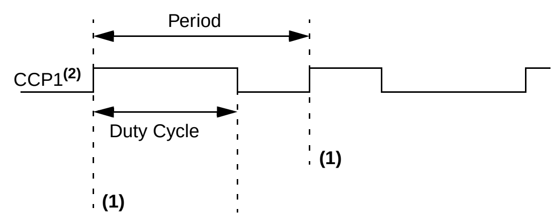
8.3.3 PWM OUTPUT CONFIGURATIONS

The PWM1M1 bits in the CCP1CON register allows one of the following configurations:

- Single output
- Half-Bridge output
- Full-Bridge output, Forward mode
- Full-Bridge output, Reverse mode

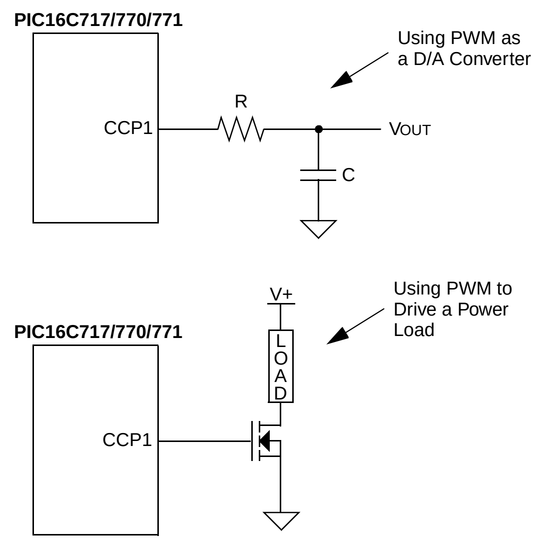
In the Single Output mode, the RB3/CCP1/P1A pin is used as the PWM output. Since the CCP1 output is multiplexed with the PORTB<3> data latch, the TRISB<3> bit must be cleared to make the CCP1 pin an output.

FIGURE 8-4: SINGLE PWM OUTPUT



Note 1: At this time, the TMR2 register is equal to the PR2 register.
2: Output signal is shown as asserted high.

FIGURE 8-5: EXAMPLE OF SINGLE OUTPUT APPLICATION



In the Half-Bridge output mode, two pins are used as outputs. The RB3/CCP1/P1A pin has the PWM output signal, while the RB5/SDO/P1B pin has the complementary PWM output signal. This mode can be used for half-bridge applications, as shown on Figure 8-7, or for full-bridge applications, where four power switches are being modulated with two PWM signal.

Since the P1A and P1B outputs are multiplexed with the PORTB<3> and PORTB<5> data latches, the TRISB<3> and TRISB<5> bits must be cleared to configure P1A and P1B as outputs.

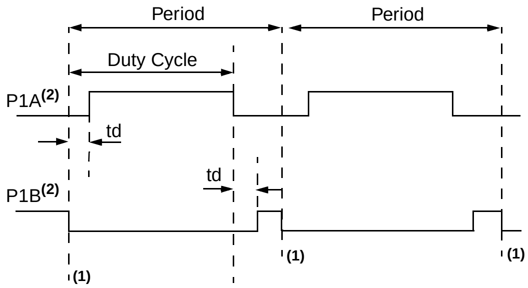
In Half-Bridge output mode, the programmable deadband delay can be used to prevent shoot-through current in bridge power devices. See Section 8.3.5 for more details of the deadband delay operations.

8.3.4 OUTPUT POLARITY CONFIGURATION

The CCP1M<1:0> bits in the CCP1CON register allow user to choose the logic conventions (asserted high/ low) for each of the outputs. See Register 8-1 for further details.

The PWM output polarities must be selected before the PWM outputs are enabled. Charging the polarity configuration while the PWM outputs are active is not recommended, since it may result in unpredictable operation.

FIGURE 8-6: HALF-BRIDGE PWM OUTPUT

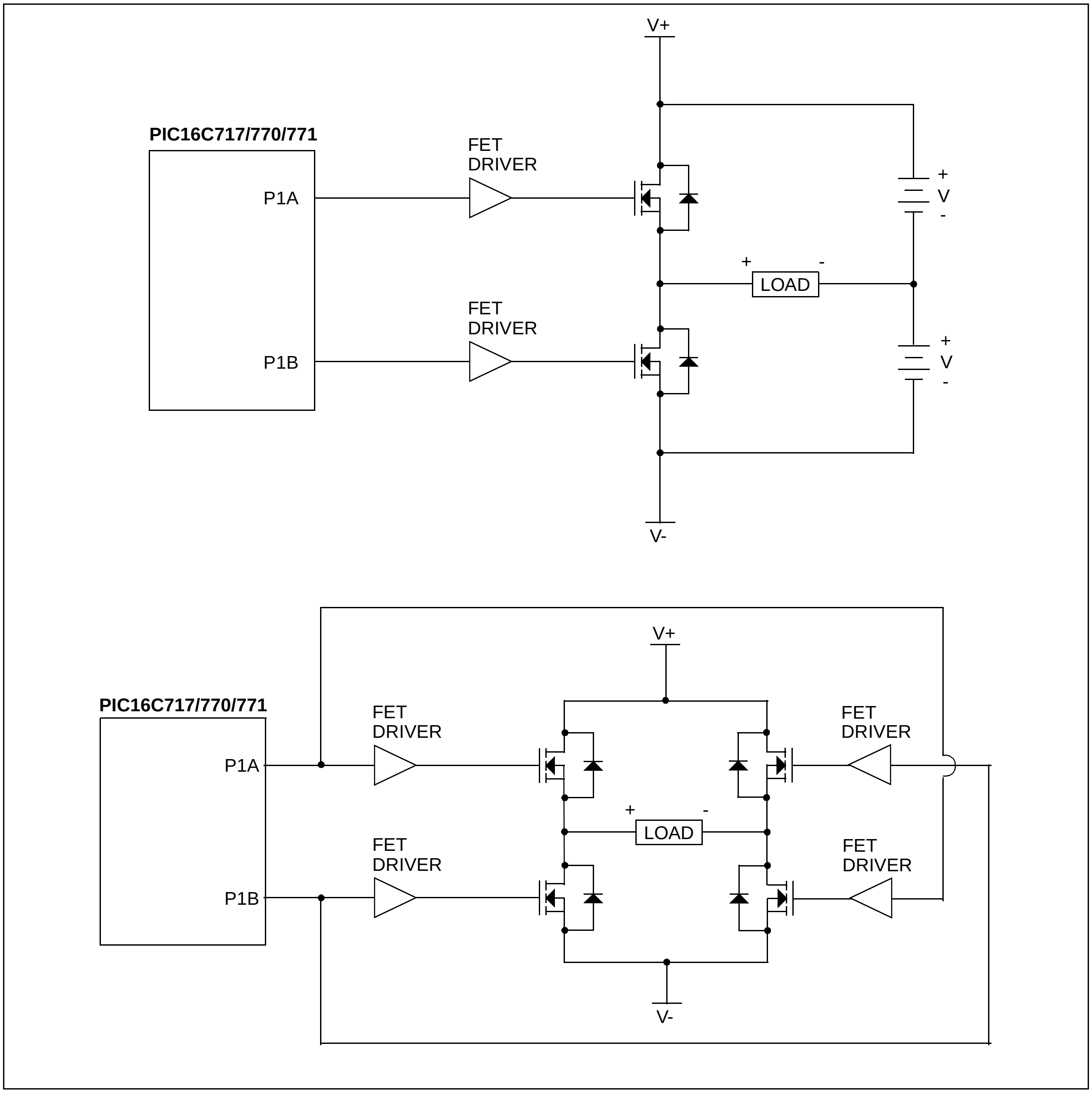


td = Deadband Delay

Note 1: At this time, the TMR2 register is equal to the PR2 register.
2: Output signals are shown as asserted high.

 Advanced Information

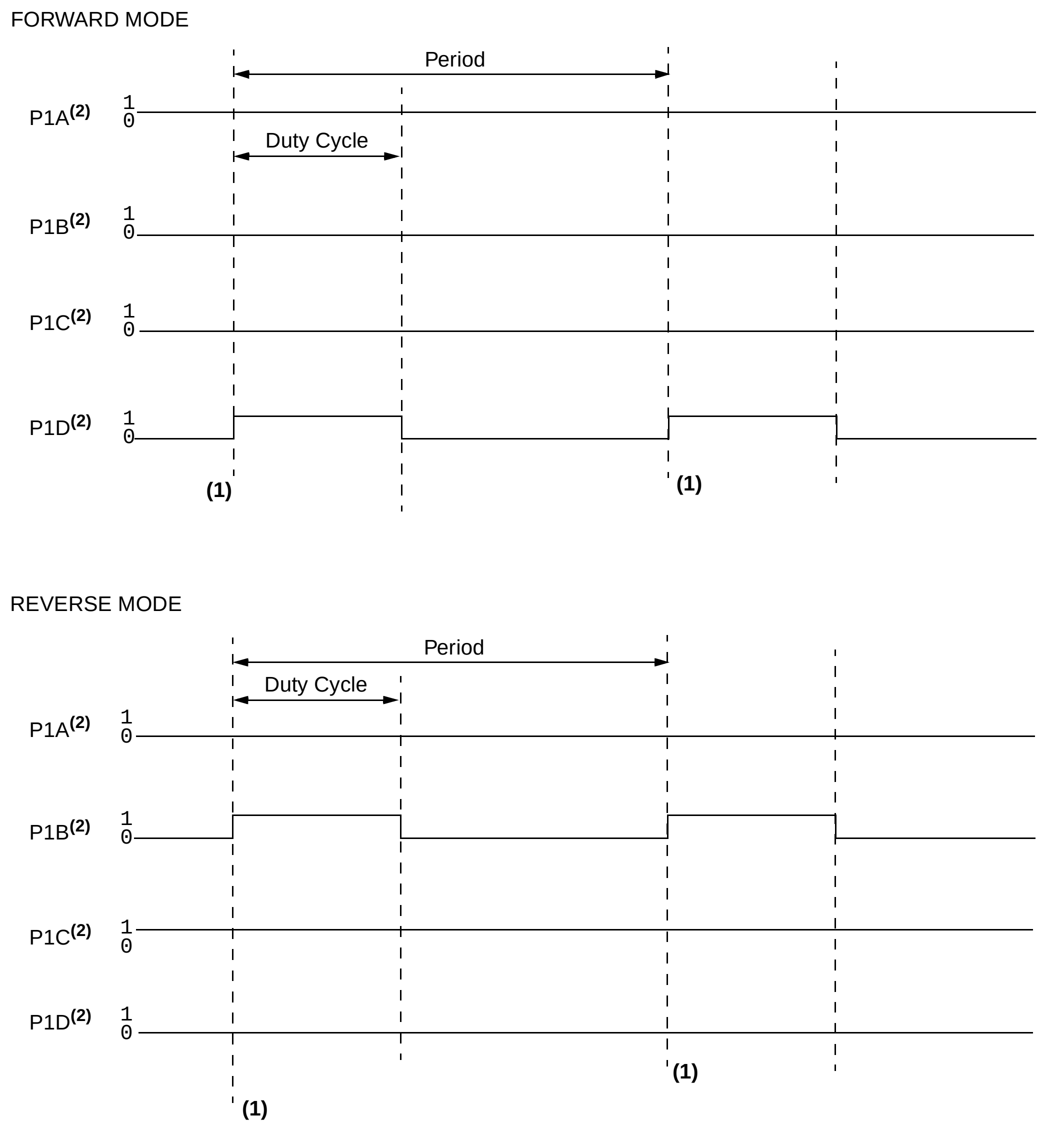
FIGURE 8-7: EXAMPLE OF HALF-BRIDGE OUTPUT MODE APPLICATIONS



In Full-Bridge output mode, four pins are used as outputs; however, only two outputs are active at a time. In the Forward mode, RB3/CCP1/P1A pin is continuously active, and RB7/T1OSI/P1D pin is modulated. In the Reverse mode, RB6/T1OSO/T1CKI/P1C pin is continuously active, and RB5/SDO/P1B pin is modulated.

P1A, P1B, P1C and P1D outputs are multiplexed with PORTB<3> and PORTB<5:7> data latches. TRISB<3> and TRISB<5:7> bits must be cleared to make the P1A, P1B, P1C, and P1D pins output.

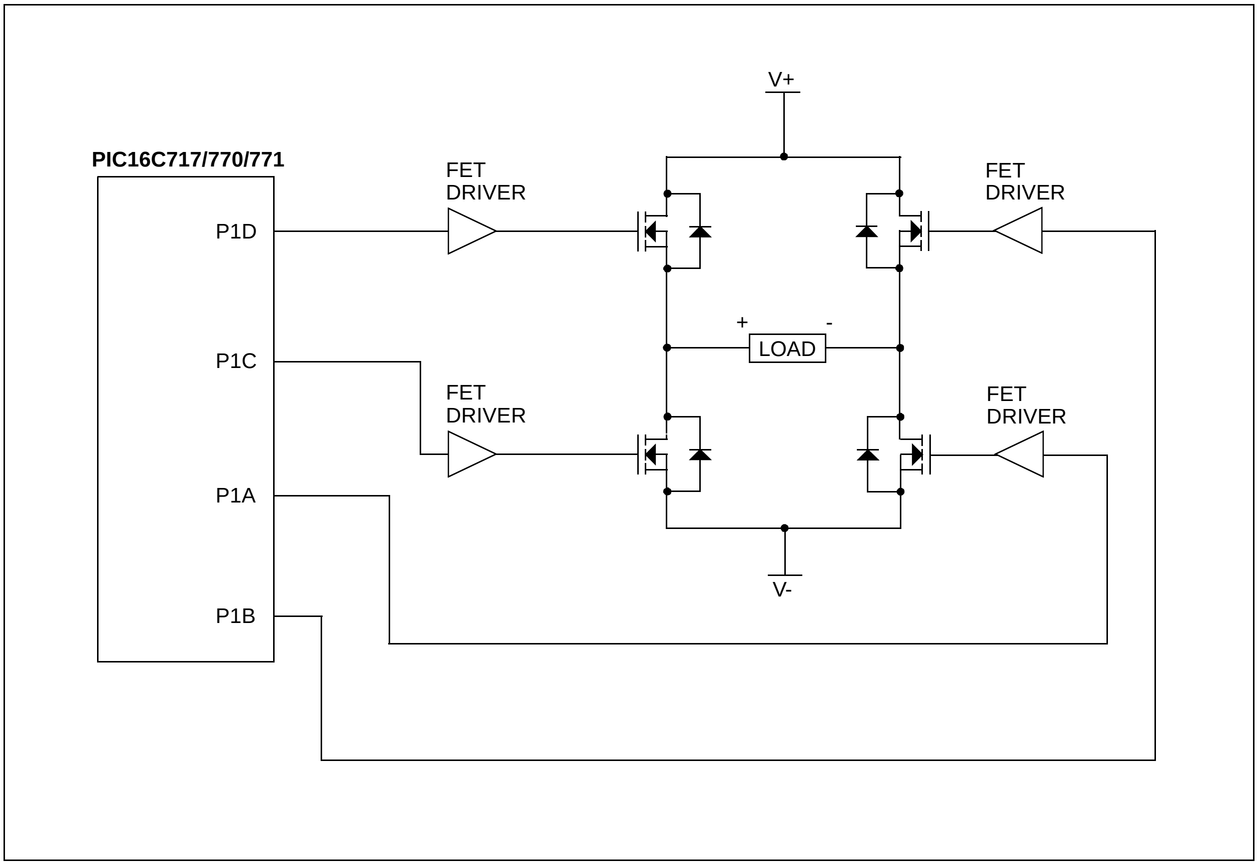
FIGURE 8-8: FULL-BRIDGE PWM OUTPUT



Note 1: At this time, the TMR2 register is equal to the PR2 register.
2: Output signal is shown as asserted high.

 Advanced Information

FIGURE 8-9: EXAMPLE OF FULL-BRIDGE APPLICATION



8.3.5 PROGRAMMABLE DEADBAND DELAY

In half-bridge or full-bridge applications, where all power switches are modulated at the PWM frequency at all time, the power switches normally require longer 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 will be on for a short period of time, until one switch completely turns off. During this time, a very high current, called 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 the power switch is normally delayed to allow the other switch to completely turn off.

In the Half-Bridge Output mode, a digitally programmable deadband 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. See Figure 8-6 for illustration. The P1DEL register sets the amount of delay.

REGISTER 8-2: PWM DELAY REGISTER (P1DEL: 97H)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
bit7							bit0

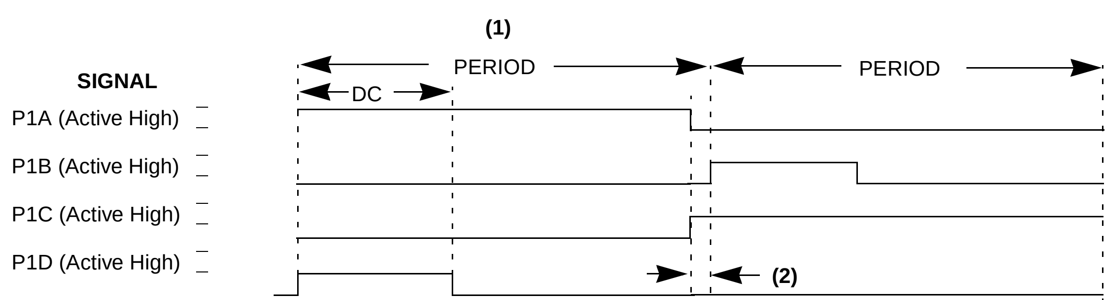
R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7-0: **P1DEL<7:0>: PWM Delay count for Half-Bridge output mode:** Number of FOSC/4 (Tosc•4) cycles between the P1A transition and the P1B transition.

8.3.6 DIRECTION CHANGE IN FULL-BRIDGE OUTPUT MODE

In the Full-Bridge Output mode, the PWM1M1 bit in the CCP1CON register allows user to control the Forward/Reverse direction. When the application firmware changes this direction control bit, the ECCP module will assume the new direction on the next PWM cycle. The current PWM cycle still continues, however, the non-modulated outputs, P1A and P1C signals, will transition to the new direction TOSC, 4•TOSC or 16•TOSC (for Timer2 presale T2CKRS<1:0> = 00, 01 and 1x respectively) earlier, before the end of the period. During this transition cycle, the modulated outputs, P1B and P1D, will go to the inactive state. See Figure 8-10 for illustration.

FIGURE 8-10: PWM DIRECTION CHANGE



Note 1: The Direction bit in the ECCP Control Register (CCP1CON.PWM1M1) is written anytime during the PWM cycle.
2: The P1A and P1C signals switch TOSC, 4*Tosc or 16*TOSC depending on the Timer2 prescaler value earlier when changing direction. The modulated P1B and P1D signals are inactive at this time.

Note that in the Full-Bridge output mode, the ECCP module does not provide any deadband delay. In general, since only one output is modulated at all time, deadband delay is not required. However, there is a situation where a deadband delay might be required. This situation occurs when all of the following conditions are true:

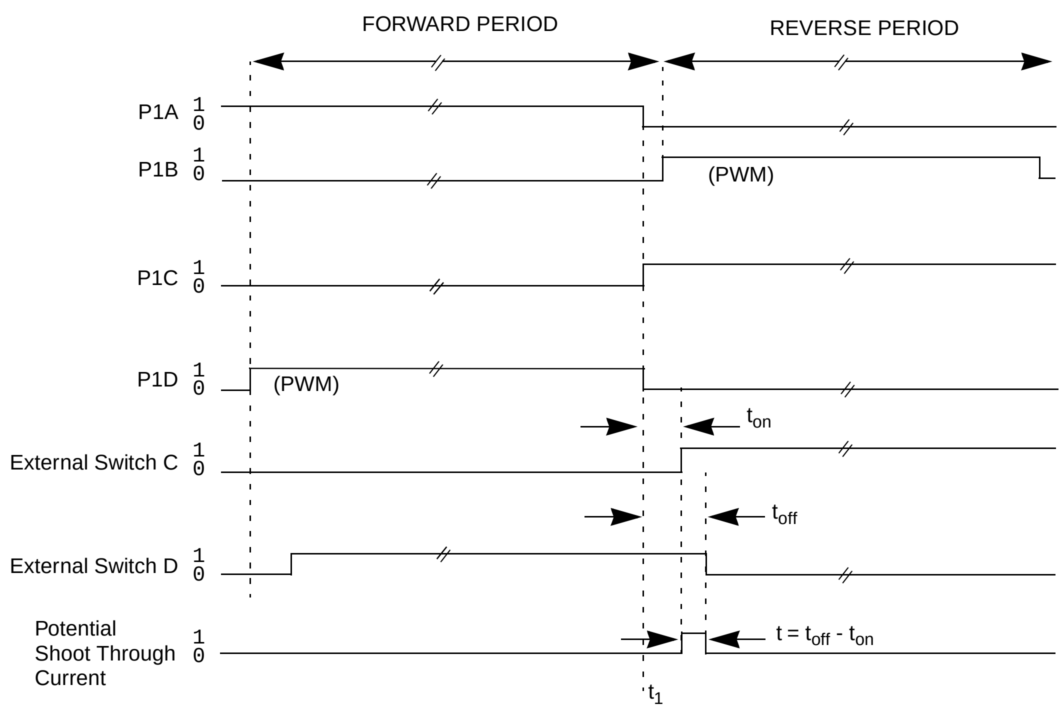
1. The direction of the PWM output changes when the duty cycle of the output is at or near 100%.
2. The turn off time of the power switch, including the power device and driver circuit, is greater than turn on time.

Figure 8-11 shows an example, where the PWM direction changes from forward to reverse at a near 100% duty cycle. At time t1, the output P1A and P1D become inactive, while output P1C becomes active. In this example, since the turn off time of the power devices is longer than the turn on time, a shoot-through current flows through the power devices, QB and QD, for the duration of $t = t_{off} - t_{on}$. The same phenomenon will occur to power devices, QC and QB, for PWM direction change from reverse to forward.

If changing PWM direction at high duty cycle is required for the user's application, one of the following requirements must be met:

1. Avoid changing PWM output direction at or near 100% duty cycle.
2. Use switch drivers that compensate the slow turn off of the power devices. The total turn off time (t_{off}) of the power device and the driver must be less than the turn on time (t_{on}).

FIGURE 8-11: PWM DIRECTION CHANGE AT NEAR 100% DUTY CYCLE



Note 1: All signals are shown as active high.

2: t_{on} is the turn on delay of power switch and driver.

3: t_{off} is the turn off delay of power switch and driver.

8.3.7 SYSTEM IMPLEMENTATION

When the ECCP module is used in the PWM mode, the application hardware must use the proper external pull-up and/or pull-down resistors on the PWM output pins. When the microcontroller powers up, all of the I/O pins are in the high-impedance state. The external pull-up and pull-down resistors must keep the power switch devices in the off state until the microcontroller drives the I/O pins with the proper signal levels, or activates the PWM output(s).

8.3.8 START-UP CONSIDERATIONS

Prior to enabling the PWM outputs, the P1A, P1B, P1C and P1D latches may not be in the proper states. Enabling the TRISB bits for output at the same time with the CCP module may cause damage to the power switch devices. The CCP1 module must be enabled in the proper output mode with the TRISB bits enabled as inputs. Once the CCP1 completes a full PWM cycle, the P1A, P1B, P1C and P1D output latches are properly initialized. At this time, the TRISB bits can be enabled for outputs to start driving the power switch devices. The completion of a full PWM cycle is indicated by the TMR2IF bit going from a '0' to a '1'.

8.3.9 SET UP FOR PWM OPERATION

The following steps should be taken when configuring the ECCP module for PWM operation:

1. Configure the PWM module:
   a) Disable the CCP1/P1A, P1B, P1C and/or P1D outputs by setting the respective TRISB bits.
   b) Set the PWM period by loading the PR2 register.
   c) Set the PWM duty cycle by loading the CCPR1L register and CCP1CON<5:4> bits.
   d) Configure the ECCP module for the desired PWM operation by loading the CCP1CON register. With the CCP1M<3:0> bits select the active high/low levels for each PWM output. With the PWM1M<1:0> bits select one of the available output modes: Single, Half-Bridge, Full-Bridge, Forward or Full-Bridge Reverse.
   e) For Half-Bridge output mode, set the dead-band delay by loading the P1DEL register.
2. Configure and start TMR2:
   a) Clear the TMR2 interrupt flag bit by clearing the TMR2IF bit in the PIR1 register.
   b) Set the TMR2 prescale value by loading the T2CKPS<1:0> bits in the T2CON register.
   c) Enable Timer2 by setting the TMR2ON bit in the T2CON register.
3. Enable PWM outputs after a new cycle has started:
   a) Wait until TMR2 overflows (TMR2IF bit becomes a '1'). The new PWM cycle begins here.
   b) Enable the CCP1/P1A, P1B, P1C and/or P1D pin outputs by clearing the respective TRISB bits.

TABLE 8-3: REGISTERS ASSOCIATED WITH PWM

Address	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
0Bh, 8Bh, 10Bh, 18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF	-0-- 0000	-0-- 0000
8Ch	PIE1	—	ADIE	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE	-0-- 0000	-0-- 0000
86h, 186h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
11h	TMR2	Timer2 register								0000 0000	0000 0000
92h	PR2	Timer2 period register								1111 1111	1111 1111
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
15h	CCPR1L	Capture/Compare/PWM register1 (LSB)								xxxx xxxx	uuuu uuuu
17h	CCP1CON	PWM1M1	PWM1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	0000 0000
97h	P1DEL	PWM1 Delay value								0000 0000	0000 0000

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

9.0 MASTER SYNCHRONOUS SERIAL PORT (MSSP) MODULE

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, etc. The MSSP module can operate in one of two modes:

- Serial Peripheral Interface (SPI™)
- Inter-Integrated Circuit (I^2C™)

REGISTER 9-1: SYNC SERIAL PORT STATUS REGISTER (SSPSTAT: 94h)

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
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

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^{2}C 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 (Figure 9-3, Figure 9-5, and Figure 9-6)
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/A:** Data/$\overline{\text{Address}}$ bit (I^{2}C 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^{2}C 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^{2}C 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/W:** Read/Write bit information (I^{2}C 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 $\overline{\text{ACK}}$ bit.
In I^{2}C slave mode:
1 = Read
0 = Write
In I^{2}C master mode:
1 = Transmit is in progress
0 = Transmit is not in progress.
Or'ing this bit with SEN, RSEN, PEN, RCEN, or AKEN will indicate if the MSSP is in IDLE mode

bit 1: **UA:** Update Address (10-bit I^{2}C 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^{2}C modes)
1 = Receive complete, SSPBUF is full
0 = Receive not complete, SSPBUF is empty
Transmit (I^{2}C mode only)
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

REGISTER 9-2: SYNC SERIAL PORT CONTROL REGISTER (SSPCON: 14h)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0
bit7							bit0

R = Readable bit
W = Writable bit
- n = Value at POR reset

bit 7: **WCOL:** Write Collision Detect bit
Master Mode:
1 = A write to the SSPBUF register was attempted while the I^{2}C conditions were not valid for a transmission to be started
0 = No collision
Slave Mode:
1 = The SSPBUF register is written while it is still transmitting the previous word (must be cleared in software)
0 = No collision

bit 6: **SSPOV:** Receive Overflow Indicator bit
In SPI mode
1 = A new byte is received while the SSPBUF register is still holding the previous data. In case of overflow, the data in SSPSR is lost. Overflow can only occur in slave mode. In slave mode, the user must read the SSPBUF, even if only transmitting data, to avoid setting overflow. In master mode, the overflow bit is not set since each new reception (and transmission) is initiated by writing to the SSPBUF register. (Must be cleared in software).
0 = No overflow
In I^{2}C mode
1 = A byte is received while the SSPBUF register is still holding the previous byte. SSPOV is a "don't care" in transmit mode. (Must be cleared in software).
0 = No overflow

bit 5: **SSPEN:** Synchronous Serial Port Enable bit
In both modes, when enabled, these pins must be properly configured as input or output.
In SPI mode
1 = Enables serial port and configures SCK, SDO, SDI, and $\overline{SS}$ as the source of the serial port pins
0 = Disables serial port and configures these pins as I/O port pins
In I^{2}C mode
1 = Enables the serial port and configures the SDA and SCL pins as the source of the serial port pins
0 = Disables serial port and configures these pins as I/O port pins

bit 4: **CKP:** Clock Polarity Select bit
In SPI mode
1 = Idle state for clock is a high level
0 = Idle state for clock is a low level
In I^{2}C slave mode
SCK release control
1 = Enable clock
0 = Holds clock low (clock stretch) (Used to ensure data setup time)
In I^{2}C master mode
Unused in this mode

bit 3-0: **SSPM<3:0>:** Synchronous Serial Port Mode Select bits
0000 = SPI master mode, clock = FOSC/4
0001 = SPI master mode, clock = FOSC/16
0010 = SPI master mode, clock = FOSC/64
0011 = SPI master mode, clock = TMR2 output/2
0100 = SPI slave mode, clock = SCK pin. $\overline{SS}$ pin control enabled.
0101 = SPI slave mode, clock = SCK pin. $\overline{SS}$ pin control disabled. $\overline{SS}$ can be used as I/O pin
0110 = I^{2}C slave mode, 7-bit address
0111 = I^{2}C slave mode, 10-bit address
1000 = I^{2}C master mode, clock = FOSC / (4 • (SSPADD+1))
1xx1 = Reserved
1x1x = Reserved

REGISTER 9-3: SYNC SERIAL PORT CONTROL REGISTER2 (SSPCON2: 91h)

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
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, Read as ‘0’
- n = Value at POR reset

bit 7: **GCEN:** General Call Enable bit (In I^{2}C 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^{2}C 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^{2}C master mode only)
In master receive mode:
Value that will be 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^{2}C 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^{2}C master mode only).
1 = Enables Receive mode for I^{2}C
0 = Receive idle

bit 2: **PEN:** Stop Condition Enable bit (In I^{2}C 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^{2}C 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^{2}C 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^{2}C 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).

9.1 SPI Mode

The SPI mode allows 8 bits of data to be synchronously transmitted and received simultaneously. All four modes of SPI are supported. To accomplish communication, typically three pins are used:

- Serial Data Out (SDO)
- Serial Data In (SDI)
- Serial Clock (SCK)

Additionally, a fourth pin may be used when in a slave mode of operation:

- Slave Select ($\overline{SS}$)

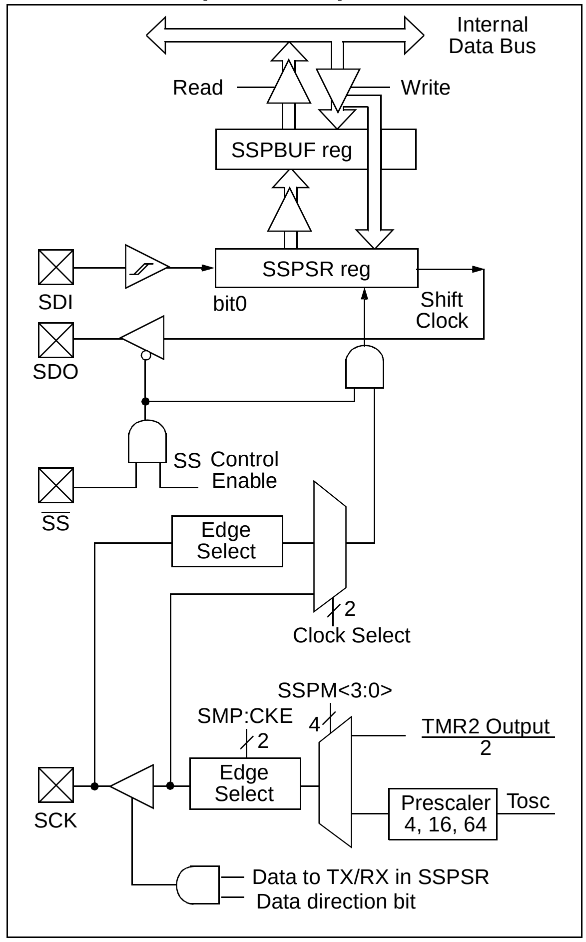
9.1.1 OPERATION

When initializing the SPI, several options need to be specified. This is done by programming the appropriate control bits (SSPCON<5:0> and SSPSTAT<7:6>). These control bits allow the following to be specified:

- Master Mode (SCK is the clock output)
- Slave Mode (SCK is the clock input)
- Clock Polarity (Idle state of SCK)
- Data input sample phase
  (middle or end of data output time)
- Clock edge
  (output data on rising/falling edge of SCK)
- Clock Rate (Master mode only)
- Slave Select Mode (Slave mode only)

Figure 9-1 shows the block diagram of the MSSP module when in SPI mode.

FIGURE 9-1: MSSP BLOCK DIAGRAM (SPI MODE)



The MSSP consists of a transmit/receive Shift Register (SSPSR) and a Buffer Register (SSPBUF). The SSPSR shifts the data in and out of the device, MSb first. The SSPBUF holds the data that was written to the SSPSR, until the received data is ready. Once the 8 bits of data have been received, that byte is moved to the SSPBUF register. Then the buffer full detect bit, BF (SSPSTAT<0>), and the interrupt flag bit, SSPIF (PIR1<3>), are set. This double buffering of the received data (SSPBUF) allows the next byte to start reception before reading the data that was just received. Any write to the SSPBUF register during transmission/reception of data will be ignored, and the write collision detect bit WCOL (SSPCON<7>) will be set. User software must clear the WCOL bit so that it can be determined if the following write(s) to the SSPBUF register completed successfully.

When the application software is expecting to receive valid data, the SSPBUF should be read before the next byte of data to transfer is written to the SSPBUF. Buffer full bit, BF (SSPSTAT<0>), indicates when the SSPBUF has been loaded with the received data (transmission is complete). When the SSPBUF is read, bit BF is cleared. This data may be irrelevant if the SPI is only a transmitter. Generally the MSSP Interrupt is used to determine when the transmission/reception has completed. The SSPBUF must be read and/or written. If the interrupt method is not going to be used, then software polling can be done to ensure that a write collision does not occur. Example 9-1 shows the loading of the SSPBUF (SSPSR) for data transmission.

EXAMPLE 9-1: LOADING THE SSPBUF (SSPSR) REGISTER

```
     BSF   STATUS, RP0    ;Specify Bank 1
LOOP BTFSS SSPSTAT, BF    ;Has data been
                          ;received
                          ;(transmit
                          ;complete)?
     GOTO  LOOP           ;No
     BCF   STATUS, RP0    ;Specify Bank 0
     MOVF  SSPBUF, W      ;W reg = contents
                          ;of SSPBUF
     MOVWF RXDATA         ;Save in user RAM
     MOVF  TXDATA, W      ;W reg = contents
                          ; of TXDATA
     MOVWF SSPBUF         ;New data to xmit
```

The SSPSR is not directly readable or writable, and can only be accessed by addressing the SSPBUF register. Additionally, the MSSP status register (SSPSTAT) indicates the various status conditions.

9.1.2 ENABLING SPI I/O

To enable the serial port, MSSP Enable bit, SSPEN (SSPCON<5>) must be set. To reset or reconfigure SPI mode, clear bit SSPEN, re-initialize the SSPCON registers, and then set bit SSPEN. 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 is automatically controlled by the SPI module
- SDO must have TRISB<5> cleared
- SCK (Master mode) must have TRISB<2> cleared
- SCK (Slave mode) must have TRISB<2> set
- $\overline{SS}$ must have TRISB<1> set, and ANSEL<5> cleared

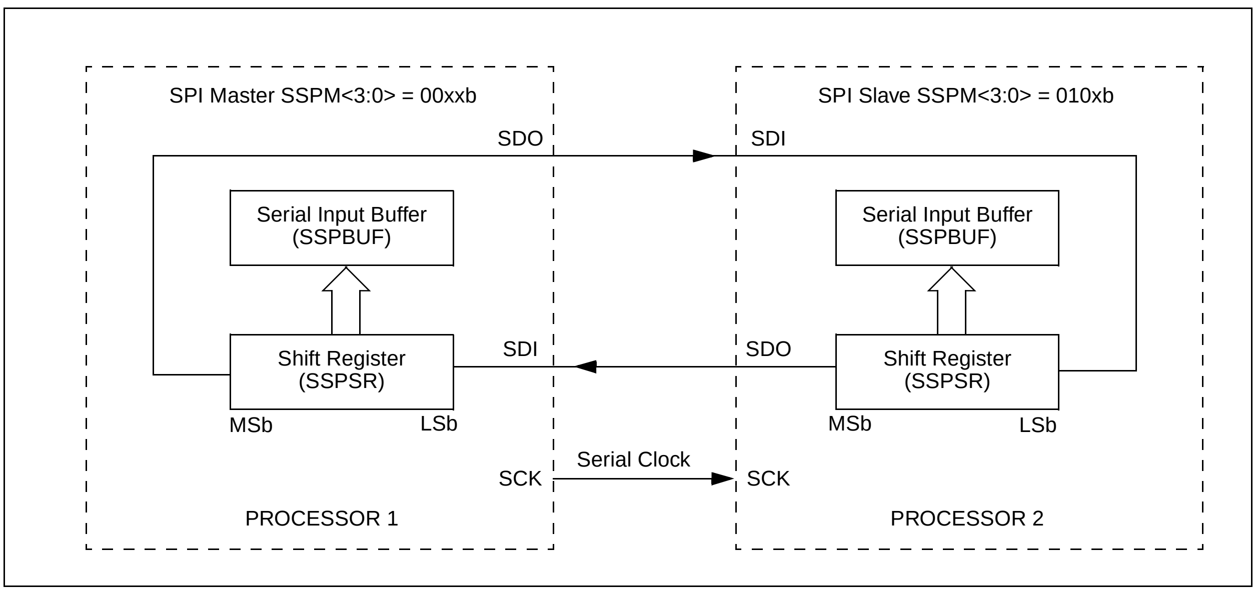
Any serial port function that is not desired may be overridden by programming the corresponding data direction (TRIS) register to the opposite value.

9.1.3 TYPICAL CONNECTION

Figure 9-2 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 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 9-2: SPI MASTER/SLAVE CONNECTION



 Advanced Information

9.1.4 MASTER MODE

The master can initiate the data transfer at any time because it controls the SCK. The master determines when the slave (Processor 2, Figure 9-2) is to broadcast data by the software protocol.

In master mode, the data is transmitted/received as soon as the SSPBUF register is written to. If the SPI module is only going to receive, the SDO output could be disabled (programmed as an input). The SSPSR register will continue to shift in the signal present on the SDI pin at the programmed clock rate. As each byte is received, it will be loaded into the SSPBUF register as if a normal received byte (interrupts and status bits appropriately set). This could be useful in receiver applications as a “line activity monitor”.

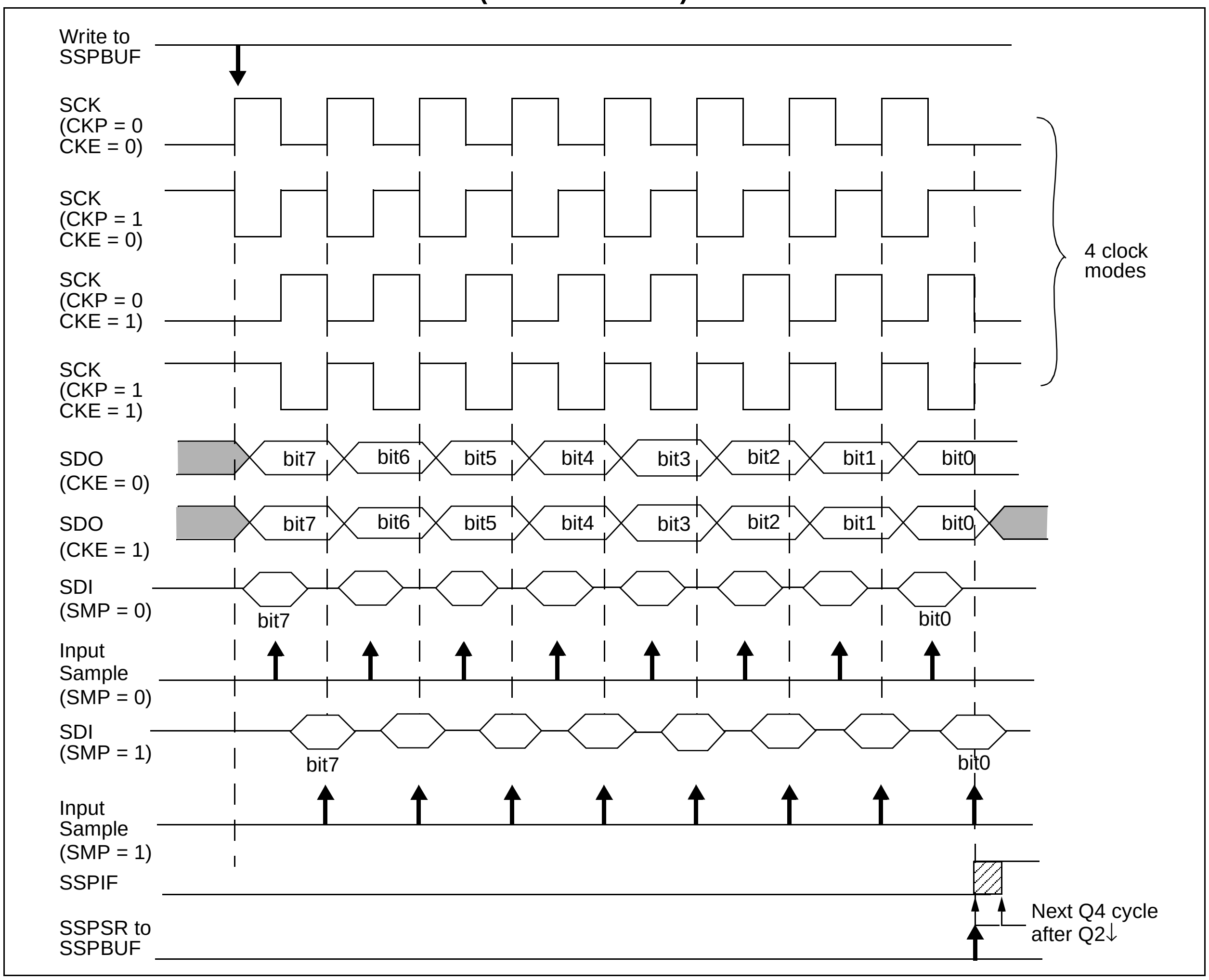
The clock polarity is selected by appropriately programming bit CKP (SSPCON<4>). This then would give waveforms for SPI communication as shown in Figure 9-3, Figure 9-5 and Figure 9-6, where the MSb is transmitted first. In master mode, the SPI clock rate (bit rate) is user programmable to be one of the following:

- FOSC/4 (or TCY)
- FOSC/16 (or 4 • TCY)
- FOSC/64 (or 16 • TCY)
- Timer2 output/2

This allows a maximum bit clock frequency (at 20 MHz) of 8.25 MHz.

Figure 9-3 shows the waveforms for Master mode. When CKE = 1, the SDO data is valid before there is a clock edge on SCK. The change of the input sample is shown based on the state of the SMP bit. The time when the SSPBUF is loaded with the received data is shown.

FIGURE 9-3: SPI MODE WAVEFORM (MASTER MODE)



9.1.5 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 interrupt flag bit SSPIF (PIR1<3>) 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 sleep mode, the slave can transmit/receive data. When a byte is received, the device will wake-up from sleep.

9.1.6 SLAVE SELECT SYNCHRONIZATION

The $\overline{SS}$ pin allows a synchronous slave mode. The SPI must be in slave mode with $\overline{SS}$ pin control enabled (SSPCON<3:0> = 0100). The pin must not be driven low for the $\overline{SS}$ pin to function as an input. TRISB<1> must be set. 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 no longer driven, even if in the middle of a transmitted byte, and becomes a floating output. External pull-up/ pull-down resistors may be desirable, depending on the application.

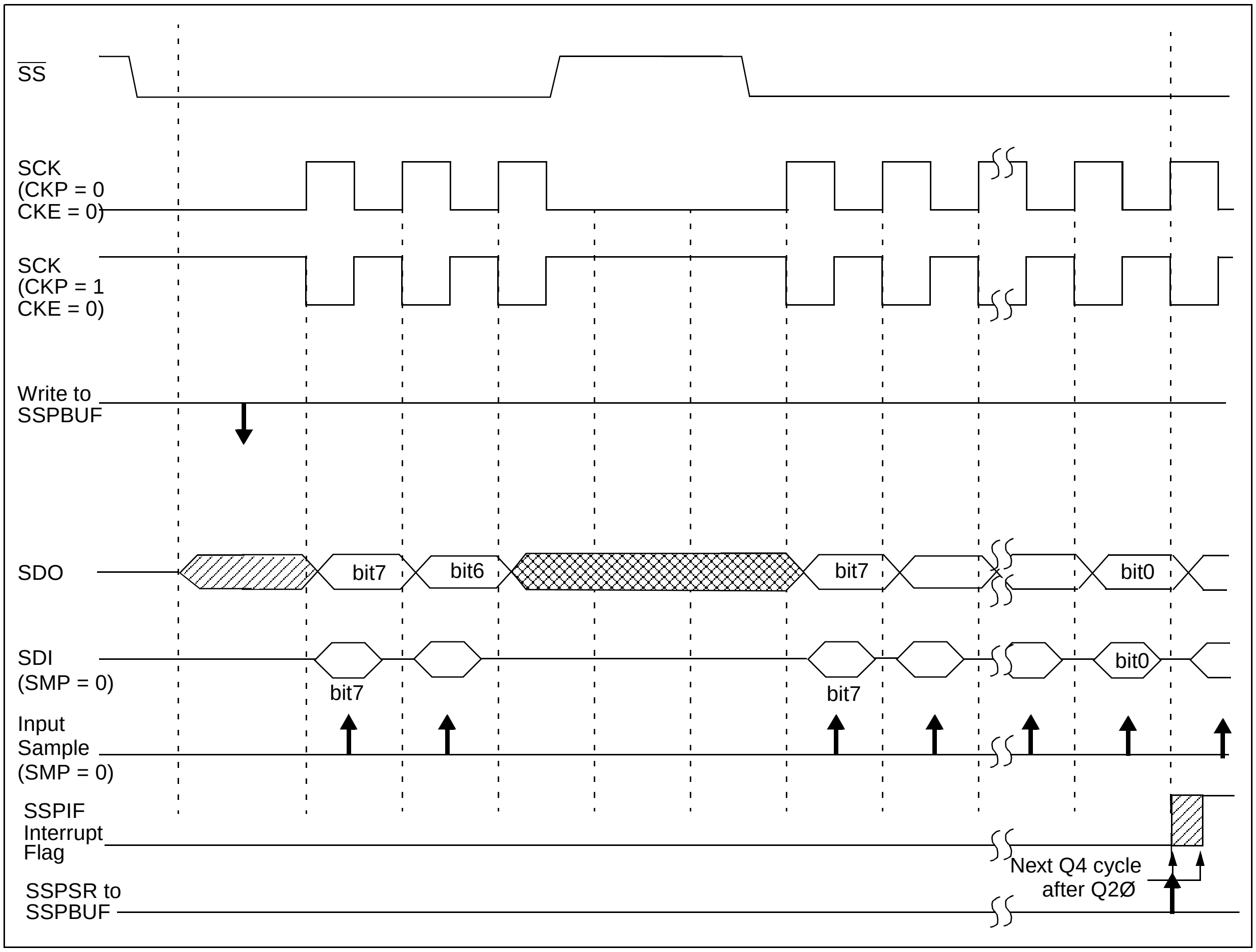
Note 1: When the SPI module is in Slave Mode with $\overline{SS}$ pin control enabled, (SSPCON<3:0> = 0100) the SPI module will reset if the $\overline{SS}$ pin is set to VDD.

2: If the SPI is used in Slave Mode with CKE = '1', then $\overline{SS}$ pin control must be enabled.

When the SPI module resets, the bit counter is forced to 0. This can be done by either forcing 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 9-4: SLAVE SYNCHRONIZATION WAVEFORM



 Advanced Information

FIGURE 9-5: SPI SLAVE MODE WAVEFORM (CKE = 0)

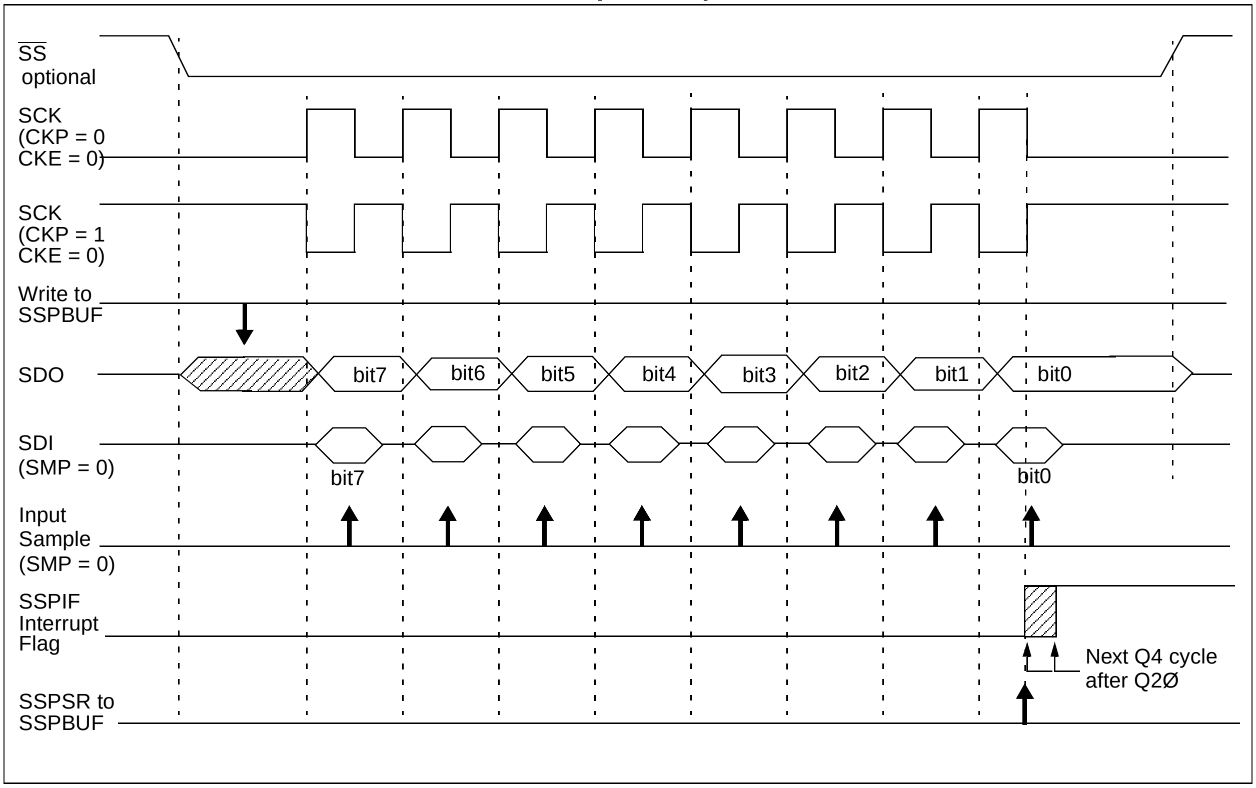
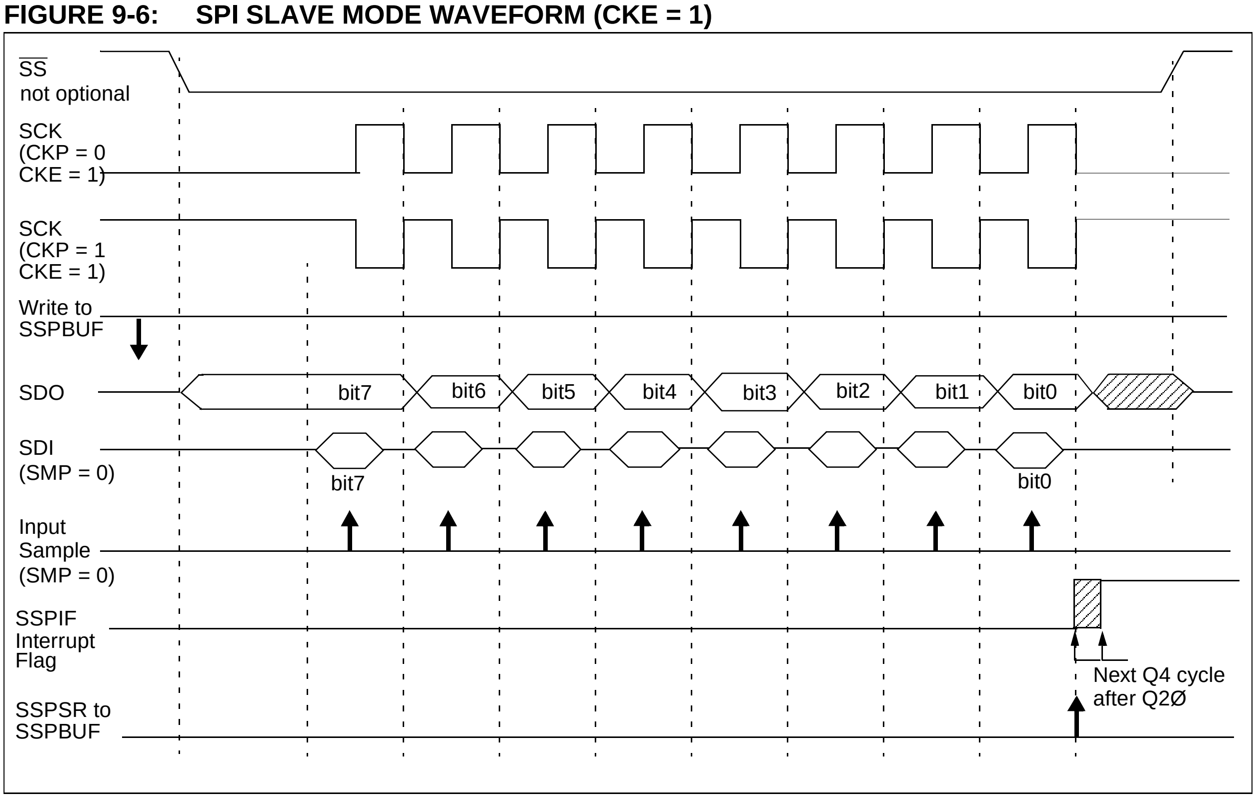


FIGURE 9-6: SPI SLAVE MODE WAVEFORM (CKE = 1)



9.1.7 SLEEP OPERATION

In master mode, all module clocks are halted and the transmission/reception will remain in that state until the device wakes from sleep. After the device returns to normal mode, the module will continue to transmit/ receive data.

In slave mode, the SPI transmit/receive shift register operates asynchronously to the device. This allows the device to be placed in sleep mode and data to be shifted into the SPI transmit/receive shift register. When all 8 bits have been received, the MSSP interrupt flag bit will be set and if enabled will wake the device from sleep.

9.1.8 EFFECTS OF A RESET

A reset disables the MSSP module and terminates the current transfer.

TABLE 9-1: REGISTERS ASSOCIATED WITH SPI OPERATION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR, BOR	$\overline{MCLR}$, WDT
0Bh, 8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF	-0-- 0000	-0-- 0000
8Ch	PIE1	—	ADIE	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE	-0-- 0000	-0-- 0000
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
94h	SSPSTAT	SMP	CKE	$D/\overline{A}$	P	S	$R/\overline{W}$	UA	BF	0000 0000	0000 0000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by the MSSP in SPI mode.

9.2 MSSP I²C Operation

The MSSP module in I²C mode fully implements all master and slave functions (including general call support) and provides interrupts on start and stop bits in hardware to determine a free bus (multi-master function). The MSSP module implements the standard mode specifications, as well as 7-bit and 10-bit addressing.

Refer to Application Note AN578, *"Use of the SSP Module in the I²C Multi-Master Environment."*

A "glitch" filter is on the SCL and SDA pins when the pin is an input. This filter operates in both the 100 kHz and 400 kHz modes. In the 100 kHz mode, when these pins are an output, there is a slew rate control of the pin that is independent of device frequency.

FIGURE 9-7: I²C SLAVE MODE BLOCK DIAGRAM

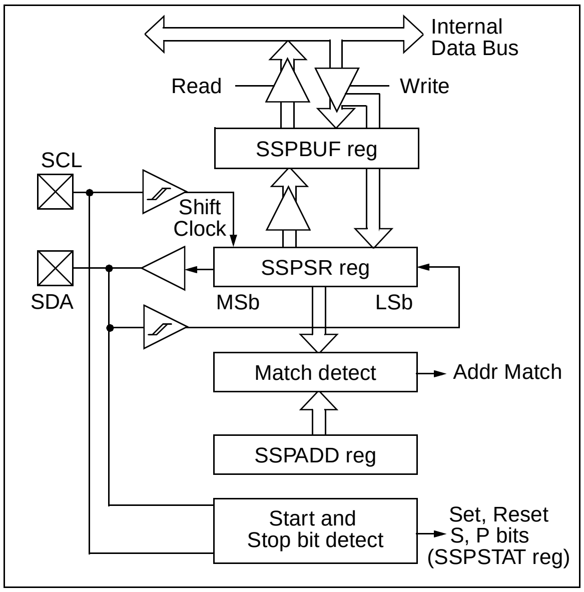
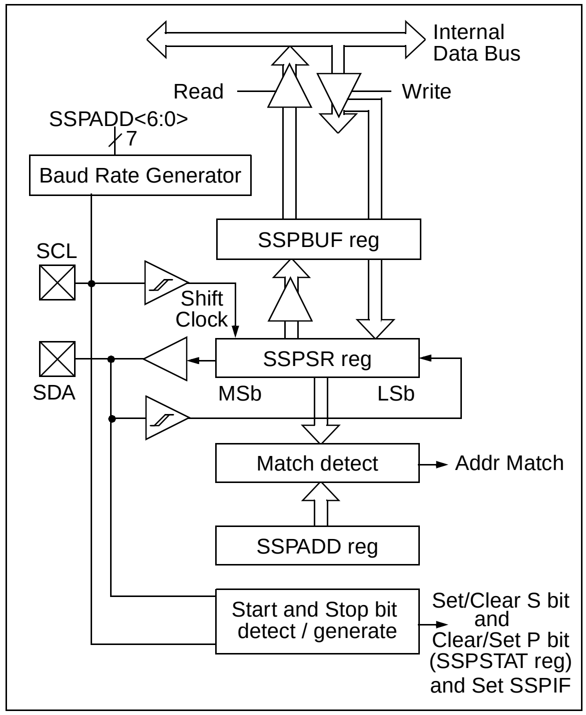


FIGURE 9-8: I²C MASTER MODE BLOCK DIAGRAM



Two pins are used for data transfer. These are the SCL pin, which is the clock, and the SDA pin, which is the data. The MSSP module functions are enabled by setting SSP Enable bit SSPEN (SSPCON<5>).

The MSSP module has six registers for I²C operation. They are the:

- SSP Control Register (SSPCON)
- SSP Control Register2 (SSPCON2)
- SSP Status Register (SSPSTAT)
- Serial Receive/Transmit Buffer (SSPBUF)
- SSP Shift Register (SSPSR) - Not directly accessible
- SSP Address Register (SSPADD)

The SSPCON register allows control of the I²C operation. Four mode selection bits (SSPCON<3:0>) allow one of the following I²C modes to be selected:

- I²C Slave mode (7-bit address)
- I²C Slave mode (10-bit address)
- I²C Master mode, clock = OSC/4 (SSPADD +1)

Before selecting any I²C mode, the SCL and SDA pins must be programmed to inputs by setting the appropriate TRIS bits. Selecting an I²C mode, by setting the SSPEN bit, enables the SCL and SDA pins to be used as the clock and data lines in I²C mode.

The SSPSTAT register gives the status of the data transfer. This information includes detection of a START (S) or STOP (P) bit, specifies if the received byte was data or address if the next byte is the completion of 10-bit address, and if this will be a read or write data transfer.

SSPBUF is the register to which the transfer data is written to or read from. The SSPSR register shifts the data in or out of the device. In receive operations, the SSPBUF and SSPSR create a doubled buffered receiver. This allows reception of the next byte to begin before reading the last byte of received data. When the complete byte is received, it is transferred to the SSPBUF register and flag bit SSPIF is set. If another complete byte is received before the SSPBUF register is read, a receiver overflow has occurred and bit SSPOV (SSPCON<6>) is set and the byte in the SSPSR is lost.

The SSPADD register holds the slave address. In 10-bit mode, the user needs to write the high byte of the address (`1111 0 A9 A8 0`). Following the high byte address match, the low byte of the address needs to be loaded (A7:A0).

9.2.1 SLAVE MODE

In slave mode, the SCL and SDA pins must be configured as inputs. The MSSP module will override the input state with the output data when required (slave-transmitter).

When an address is matched or the data transfer after an address match is received, the hardware automatically will generate the acknowledge ($\overline{\text{ACK}}$) pulse, and then load the SSPBUF register with the received value currently in the SSPSR register.

There are certain conditions that will cause the MSSP module not to give this $\overline{\text{ACK}}$ pulse. These are if either (or both):

a) The buffer full bit BF (SSPSTAT<0>) was set before the transfer was received.
b) The overflow bit SSPOV (SSPCON<6>) was set before the transfer was received.

If the BF bit is set, the SSPSR register value is not loaded into the SSPBUF, but bit SSPIF and SSPOV are set. Table 9-2 shows what happens when a data transfer byte is received, given the status of bits BF and SSPOV. The shaded cells show the condition where user software did not properly clear the overflow condition. Flag bit BF is cleared by reading the SSPBUF register while bit SSPOV is cleared through software.

The SCL clock input must have a minimum high and low time for proper operation. The high and low times of the I^2C specification as well as the requirement of the MSSP module is shown in timing parameter #100 and parameter #101 of the Electrical Specifications.

9.2.1.1 ADDRESSING

Once the MSSP module has been enabled, it waits for a START condition to occur. Following the START condition, the 8-bits are shifted into the SSPSR register. All incoming bits are sampled with the rising edge of the clock (SCL) line. The value of register SSPSR<7:1> is compared to the value of the SSPADD register. The address is compared on the falling edge of the eighth clock (SCL) pulse. If the addresses match, and the BF and SSPOV bits are clear, the following events occur:

a) The SSPSR register value is loaded into the SSPBUF register on the falling edge of the 8th SCL pulse.
b) The buffer full bit, BF is set on the falling edge of the 8th SCL pulse.
c) An $\overline{\text{ACK}}$ pulse is generated.
d) SSP interrupt flag bit, SSPIF (PIR1<3>) is set (interrupt is generated if enabled) - on the falling edge of the 9th SCL pulse.

In 10-bit address mode, two address bytes need to be received by the slave. The five Most Significant bits (MSbs) of the first address byte specify if this is a 10-bit address. Bit R/$\overline{\text{W}}$ (SSPSTAT<2>) must specify a write so the slave device will receive the second address byte. For a 10-bit address the first byte would equal '`1111 0 A9 A8 0`', where A9 and A8 are the two MSbs of the address. The sequence of events for a 10-bit address is as follows, with steps 7- 9 for slave-transmitter:

1. Receive first (high) byte of Address (bits SSPIF, BF, and bit UA (SSPSTAT<1>) are set).
2. Update the SSPADD register with second (low) byte of Address (clears bit UA and releases the SCL line).
3. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.
4. Receive second (low) byte of Address (bits SSPIF, BF, and UA are set).
5. Update the SSPADD register with the first (high) byte of Address. This will clear bit UA and release the SCL line.
6. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.
7. Receive Repeated Start condition.
8. Receive first (high) byte of Address (bits SSPIF and BF are set).
9. Read the SSPBUF register (clears bit BF) and clear flag bit SSPIF.

Note:	Following the Repeated Start condition (step 7) in 10-bit mode, the user only needs to match the first 7-bit address. The user does not update the SSPADD for the second half of the address.

9.2.1.2 SLAVE RECEPTION

When the R/$\overline{W}$ bit of the address byte is clear and an address match occurs, the R/$\overline{W}$ bit of the SSPSTAT register is cleared. The received address is loaded into the SSPBUF register.

When the address byte overflow condition exists, then no acknowledge ($\overline{ACK}$) pulse is given. An overflow condition is defined as either bit BF (SSPSTAT<0>) or bit SSPOV (SSPCON<6>) and is set.

A MSSP interrupt is generated for each data transfer byte. Flag bit SSPIF (PIR1<3>) must be cleared in software. The SSPSTAT register is used to determine the status of the received byte.

Note:	The SSPBUF will be loaded if the SSPOV bit is set and the BF flag is cleared. If a read of the SSPBUF was performed, but the user did not clear the state of the SSPOV bit before the next receive occurred, the $\overline{ACK}$ is not sent and the SSPBUF is updated.

TABLE 9-2: DATA TRANSFER RECEIVED BYTE ACTIONS

Status Bits as Data Transfer is Received		SSPSR → SSPBUF	Generate $\overline{ACK}$ Pulse	Set bit SSPIF (SSP Interrupt occurs if enabled)
BF	SSPOV			
0	0	Yes	Yes	Yes
1	0	No	No	Yes
1	1	No	No	Yes
0	1	Yes	No	Yes

Note 1: Shaded cells show the conditions where the user software did not properly clear the overflow condition.

9.2.1.3 SLAVE TRANSMISSION

When the R/$\overline{W}$ bit of the incoming address byte is set and an address match occurs, the R/$\overline{W}$ bit of the SSPSTAT register is set. The received address is loaded into the SSPBUF register. The $\overline{ACK}$ pulse will be sent on the ninth bit, and the SCL pin is held low. The transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. Then the SCL pin should be enabled by setting bit CKP (SSPCON<4>). The master must monitor the SCL pin prior to asserting another clock pulse. The slave devices may be holding off the master by stretching the clock. The eight data bits are shifted out on the falling edge of the SCL input. This ensures that the SDA signal is valid during the SCL high time (Figure 9-10).

A MSSP interrupt is generated for each data transfer byte. The SSPIF flag bit must be cleared in software, and the SSPSTAT register is used to determine the status of the byte transfer. The SSPIF flag bit is set on the falling edge of the ninth clock pulse.

As a slave-transmitter, the $\overline{ACK}$ pulse from the master-receiver is latched on the rising edge of the ninth SCL input pulse. If the SDA line was high (not $\overline{ACK}$), then the data transfer is complete. When the not $\overline{ACK}$ is latched by the slave, the slave logic is reset and the slave then monitors for another occurrence of the START bit. If the SDA line was low ($\overline{ACK}$), the transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. Then the SCL pin should be enabled by setting the CKP bit.

FIGURE 9-9: I²C WAVEFORMS FOR RECEPTION (7-BIT ADDRESS)

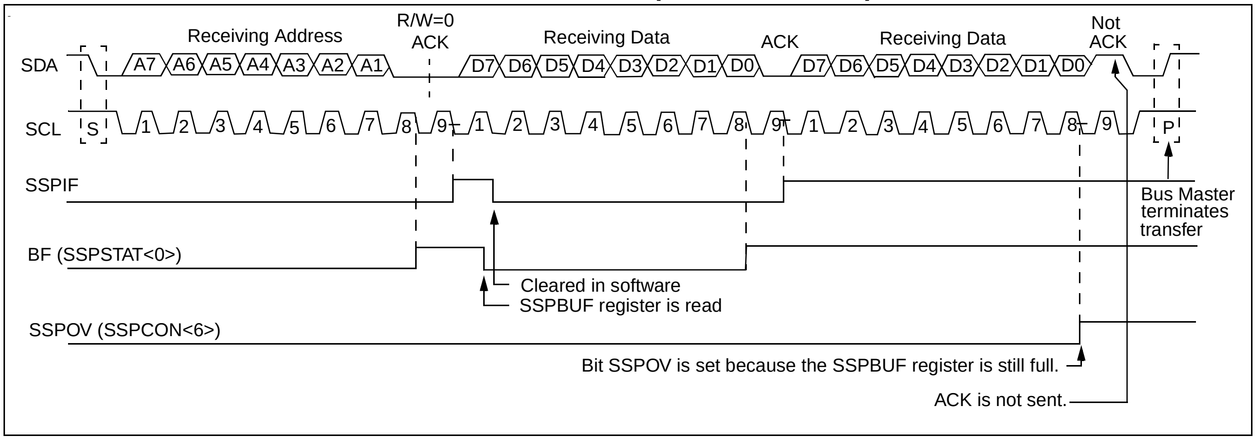


FIGURE 9-10: I²C WAVEFORMS FOR TRANSMISSION (7-BIT ADDRESS)

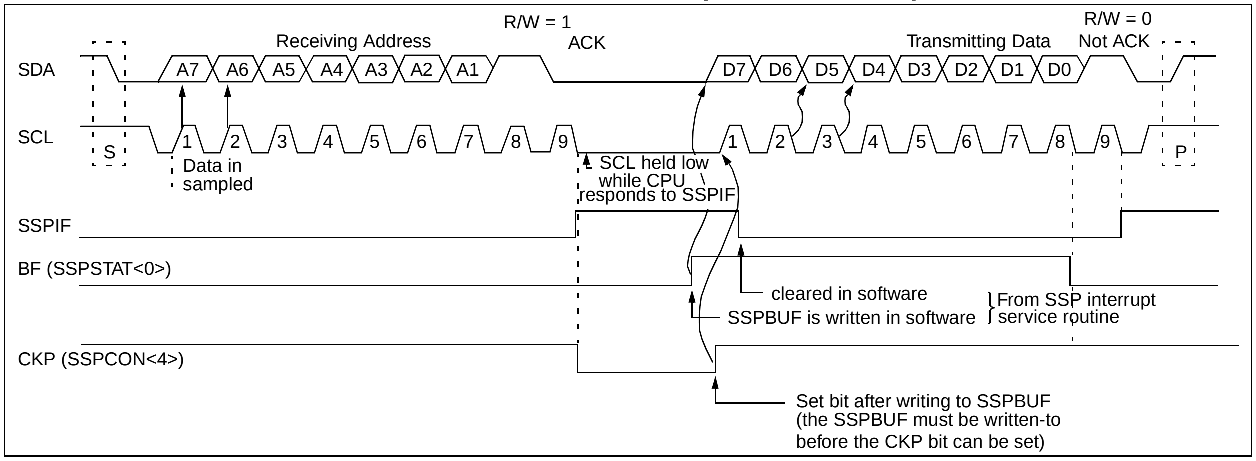


FIGURE 9-11: I[2]C SLAVE-TRANSMITTER (10-BIT ADDRESS)

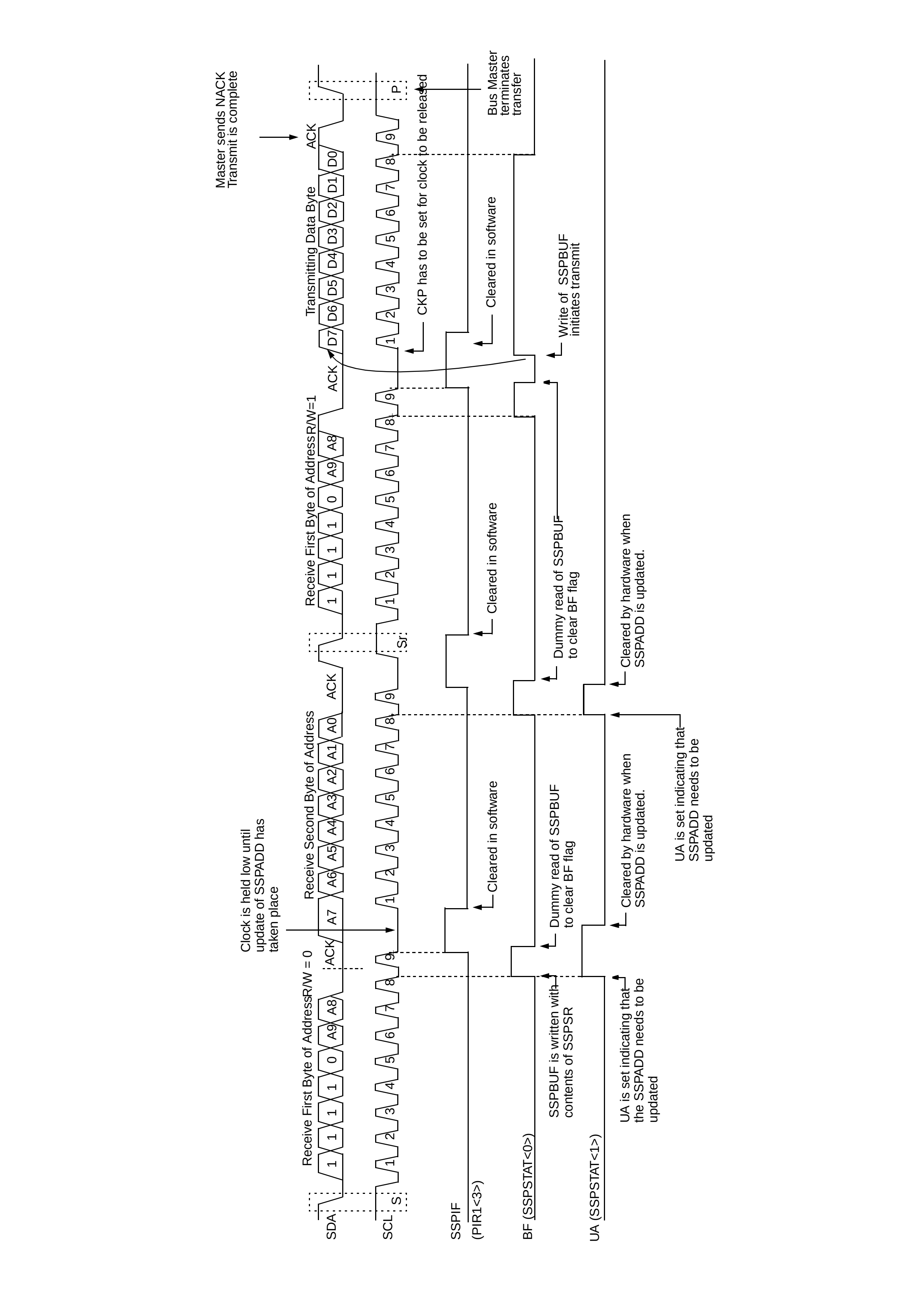
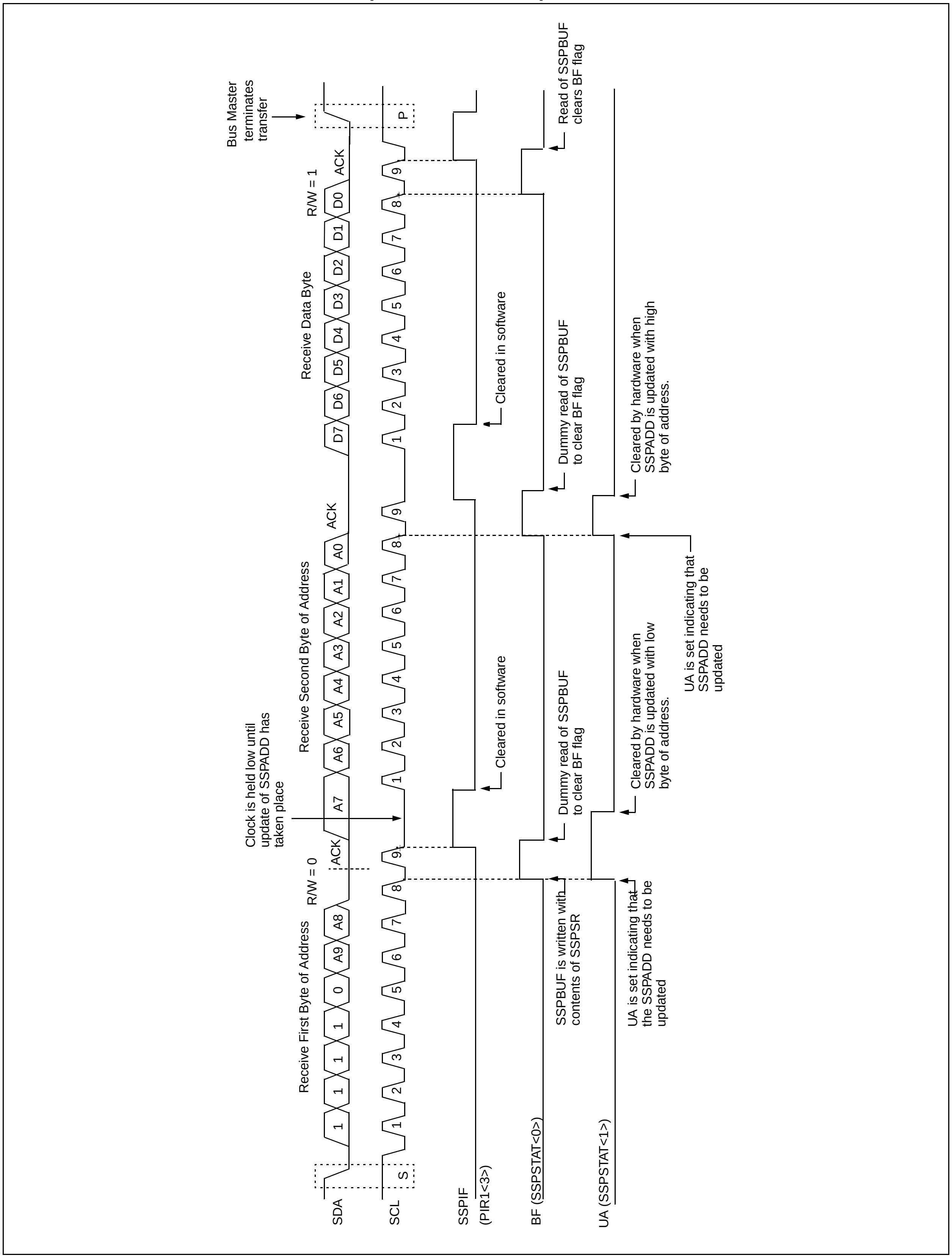


FIGURE 9-12: I^2C SLAVE-RECEIVER (10-BIT ADDRESS)



 Advanced Information

9.2.2 GENERAL CALL ADDRESS SUPPORT

The addressing procedure for the I^2C bus is such that the first byte after the START condition usually determines which device will be the slave addressed by the master. The exception is the general call address, which can address all devices. When this address is used, all devices should, in theory, respond with an acknowledge.

The general call address is one of eight addresses reserved for specific purposes by the I^2C protocol. It consists of all 0's with $R/\overline{W}$ = 0

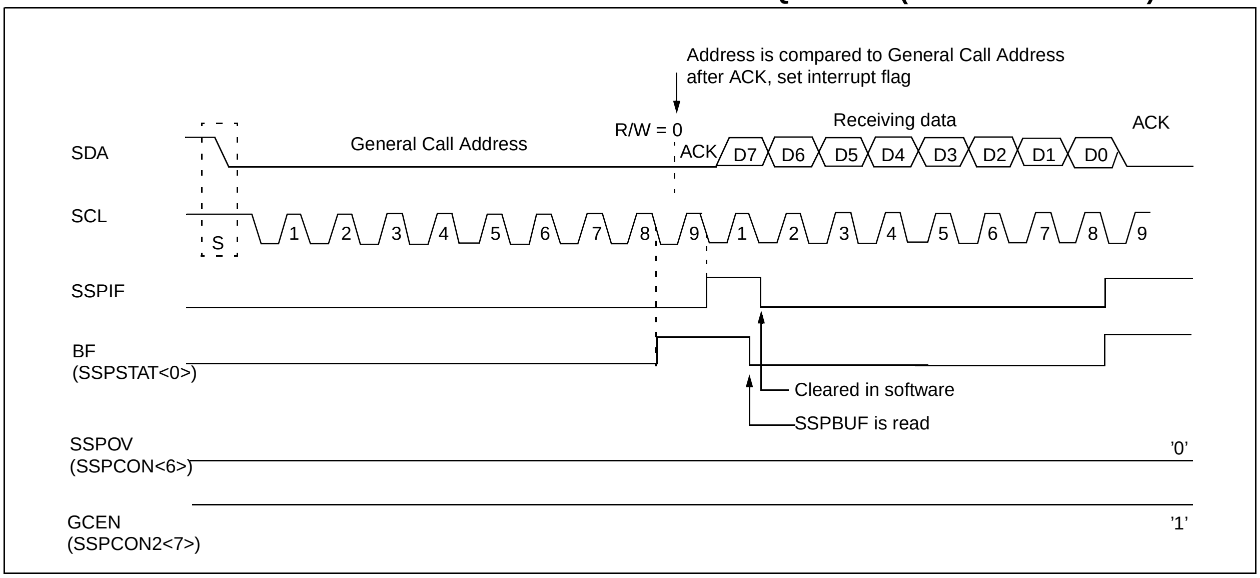
The general call address is recognized when the General Call Enable bit (GCEN) is enabled (SSPCON2<7> is set). Following a start-bit detect, 8 bits are shifted into SSPSR and the address is compared against SSPADD. It is also compared to the general call address, fixed in hardware.

If the general call address matches, the SSPSR is transferred to the SSPBUF, the BF flag is set (eighth bit), and on the falling edge of the ninth bit ($\overline{ACK}$ bit), the SSPIF flag is set.

When the interrupt is serviced, the source for the interrupt can be checked by reading the contents of the SSPBUF to determine if the address was device specific or a general call address.

In 10-bit mode, the SSPADD is required to be updated for the second half of the address to match, and the UA bit is set (SSPSTAT<1>). If the general call address is sampled when GCEN is set while the slave is configured in 10-bit address mode, then the second half of the address is not necessary, the UA bit will not be set, and the slave will begin receiving data after the acknowledge (Figure 9-13).

FIGURE 9-13: SLAVE MODE GENERAL CALL ADDRESS SEQUENCE (7 OR 10-BIT MODE)



9.2.3 SLEEP OPERATION

While in sleep mode, the I^2C module can receive addresses or data, and when an address match or complete byte transfer occurs, wake the processor from sleep (if the SSP interrupt bit is enabled).

9.2.4 EFFECTS OF A RESET

A reset disables the MSSP module and terminates the current transfer.

9.2.5 MASTER MODE

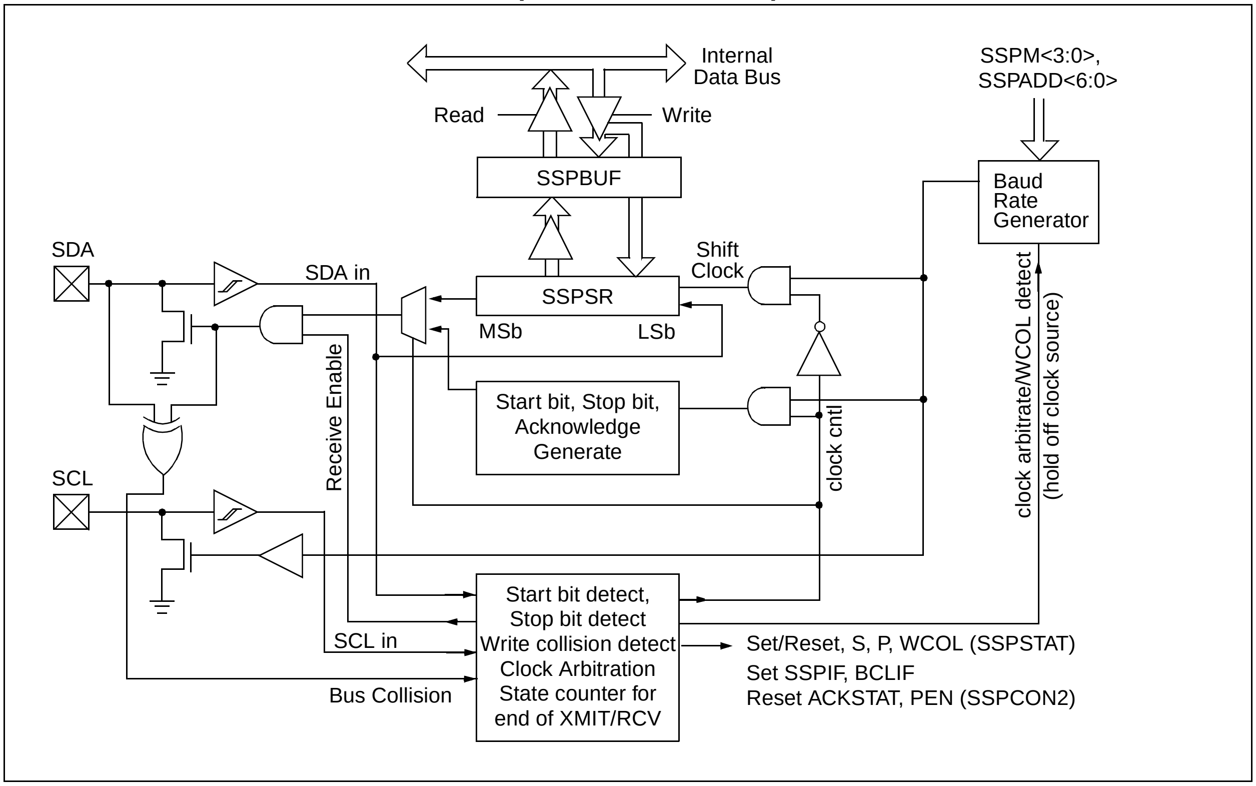
Master mode operation is supported by interrupt generation on the detection of the START and STOP conditions. The STOP (P) and START (S) bits are cleared from a reset or when the MSSP module is disabled. Control of the I^2C bus may be taken when the P bit is set, or the bus is idle with both the S and P bits clear.

In master mode, the SCL and SDA lines are manipulated by the MSSP hardware.

The following events will cause SSP Interrupt Flag bit, SSPIF, to be set (SSP Interrupt if enabled):

- START condition
- STOP condition
- Data transfer byte transmitted/received
- Acknowledge transmit
- Repeated Start

FIGURE 9-14: MSSP BLOCK DIAGRAM (I^2C MASTER MODE)



9.2.6 MULTI-MASTER OPERATION

In multi-master mode, the interrupt generation on the detection of the START and STOP conditions allows the determination of when the bus is free. The STOP (P) and START (S) bits are cleared from a reset or when the MSSP module is disabled. Control of the I^{2}C bus may be taken when bit P (SSPSTAT<4>) is set, or the bus is idle with both the S and P bits clear. When the bus is busy, enabling the SSP Interrupt will generate the interrupt when the STOP condition occurs.

In multi-master operation, the SDA line must be monitored for arbitration to see if the signal level is the expected output level. This check is performed in hardware, with the result placed in the BCLIF bit.

The states where arbitration can be lost are:

- Address Transfer
- Data Transfer
- A Start Condition
- A Repeated Start Condition
- An Acknowledge Condition

9.2.7 I^{2}C MASTER OPERATION SUPPORT

Master Mode is enabled by setting and clearing the appropriate SSPM bits in SSPCON and by setting the SSPEN bit. Once master mode is enabled, the user has six options.

- Assert a start condition on SDA and SCL.
- Assert a Repeated Start condition on SDA and SCL.
- Write to the SSPBUF register initiating transmission of data/address.
- Generate a stop condition on SDA and SCL.
- Configure the I^{2}C port to receive data.
- Generate an Acknowledge condition at the end of a received byte of data.

> **Note:** The MSSP Module, when configured in I^{2}C Master Mode, does not allow queueing of events. For instance, the user is not allowed to initiate a start condition and immediately write the SSPBUF register to initiate transmission before the START condition is complete. In this case, the SSPBUF will not be written to, and the WCOL bit will be set, indicating that a write to the SSPBUF did not occur.

9.2.7.1 I^{2}C MASTER MODE OPERATION

The master device generates all of the serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a Repeated Start condition. Since the Repeated Start condition is also the beginning of the next serial transfer, the I^{2}C bus will not be released.

In Master Transmitter mode, serial data is output through SDA, while SCL outputs the serial clock. The first byte transmitted contains the slave address of the receiving device (7 bits) and the Read/Write (R/$\overline{W}$) bit. In this case, the R/$\overline{W}$ bit will be logic '0'. Serial data is transmitted 8 bits at a time. After each byte is transmitted, an acknowledge bit is received. START and STOP conditions are output to indicate the beginning and the end of a serial transfer.

In Master receive mode, the first byte transmitted contains the slave address of the transmitting device (7 bits) and the R/$\overline{W}$ bit. In this case the R/$\overline{W}$ bit will be logic '1'. Thus the first byte transmitted is a 7-bit slave address followed by a '1' to indicate receive bit. Serial data is received via SDA while SCL outputs the serial clock. Serial data is received 8 bits at a time. After each byte is received, an acknowledge bit is transmitted. START and STOP conditions indicate the beginning and end of transmission.

The baud rate generator used for SPI mode operation is now used to set the SCL clock frequency for either 100 kHz, 400 kHz, or 1 MHz I^{2}C operation. The baud rate generator reload value is contained in the lower 7 bits of the SSPADD register. The baud rate generator will automatically begin counting on a write to the SSPBUF. Once the given operation is complete (i.e. transmission of the last data bit is followed by ACK), the internal clock will automatically stop counting and the SCL pin will remain in its last state

A typical transmit sequence would go as follows:

a) The user generates a Start Condition by setting the START enable bit (SEN) in SSPCON2.

b) SSPIF is set. The module will wait the required start time before any other operation takes place.

c) The user loads the SSPBUF with address to transmit.

d) Address is shifted out the SDA pin until all 8 bits are transmitted.

e) The MSSP Module shifts in the ACK bit from the slave device, and writes its value into the SSPCON2 register (SSPCON2<6>).

f) The module generates an interrupt at the end of the ninth clock cycle by setting SSPIF.

g) The user loads the SSPBUF with eight bits of data.

h) DATA is shifted out the SDA pin until all 8 bits are transmitted.

i) The MSSP Module shifts in the ACK bit from the slave device and writes its value into the SSPCON2 register (SSPCON2<6>).

j) The MSSP module generates an interrupt at the end of the ninth clock cycle by setting the SSPIF bit.

k) The user generates a STOP condition by setting the STOP enable bit PEN in SSPCON2.

l) Interrupt is generated once the STOP condition is complete.

9.2.8 BAUD RATE GENERATOR

In I^2C master mode, the reload value for the BRG is located in the lower 7 bits of the SSPADD register (Figure 9-15). When the BRG is loaded with this value, the BRG counts down to 0 and stops until another reload has taken place. The BRG count is decremented twice per instruction cycle (TCY) on the Q2 and Q4 clock.

In I^2C master mode, the BRG is reloaded automatically. If Clock Arbitration is taking place for instance, the BRG will be reloaded when the SCL pin is sampled high (Figure 9-16).

FIGURE 9-15: BAUD RATE GENERATOR BLOCK DIAGRAM

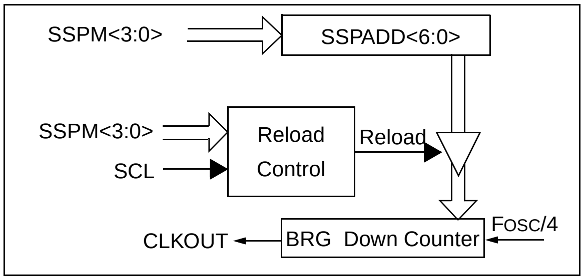
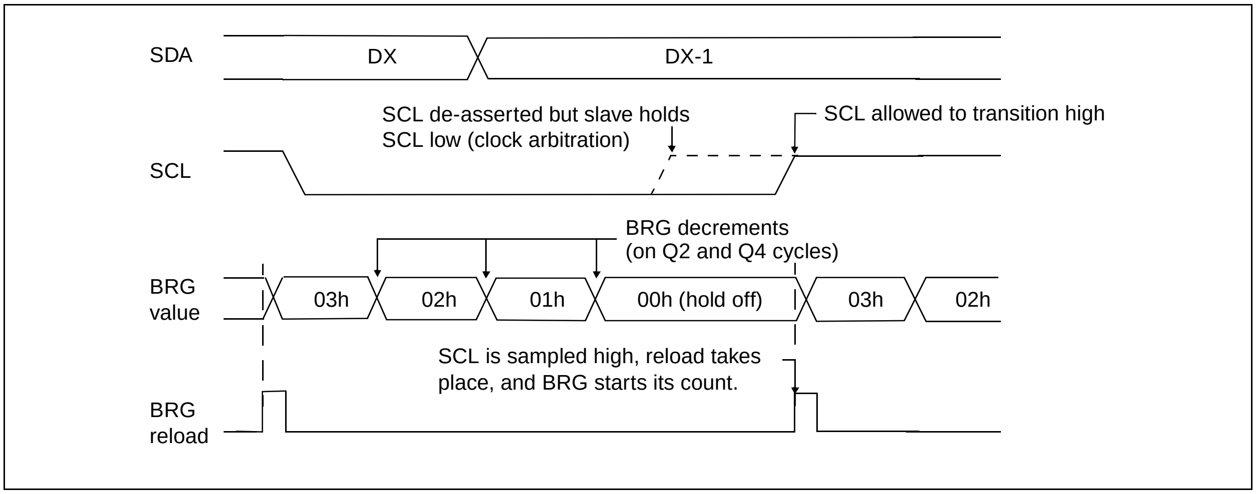


FIGURE 9-16: BAUD RATE GENERATOR TIMING WITH CLOCK ARBITRATION



 Advanced Information

9.2.9 I²C MASTER MODE START CONDITION TIMING

To initiate a START condition, the user sets the start condition enable bit, SEN (SSPCON2<0>). If the SDA and SCL pins are sampled high, the baud rate generator is re-loaded with the contents of SSPADD<6:0>, and starts its count. If SCL and SDA are both sampled high when the baud rate generator times out (T_{BRG}), the SDA pin is driven low. The action of the SDA being driven low while SCL is high is the START condition, and causes the S bit (SSPSTAT<3>) to be set. Following this, the baud rate generator is reloaded with the contents of SSPADD<6:0> and resumes its count. When the baud rate generator times out (T_{BRG}), the SEN bit (SSPCON2<0>) will be automatically cleared by hardware, the baud rate generator is suspended leaving the SDA line held low, and the START condition is complete.

Note:	If at the beginning of START condition, the SDA and SCL pins are already sampled low, or if during the START condition, the SCL line is sampled low before the SDA line is driven low, a bus collision occurs, the Bus Collision Interrupt Flag (BCLIF) is set, the START condition is aborted, and the I²C module is reset into its IDLE state.

9.2.9.1 WCOL STATUS FLAG

If the user writes the SSPBUF when an START sequence is in progress, then WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

Note:	Because queueing of events is not allowed, writing to the lower 5 bits of SSPCON2 is disabled until the START condition is complete.

FIGURE 9-17: FIRST START BIT TIMING

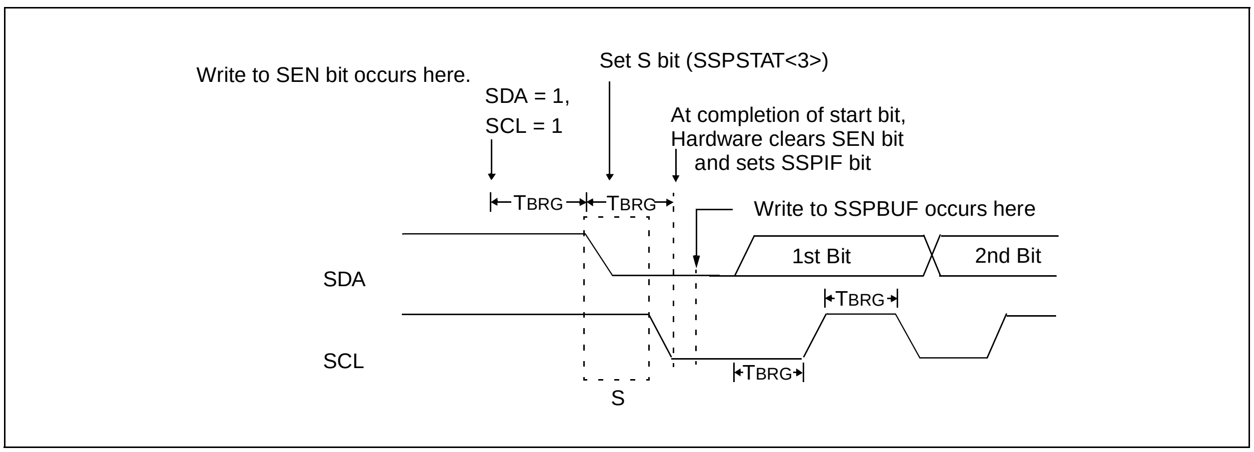
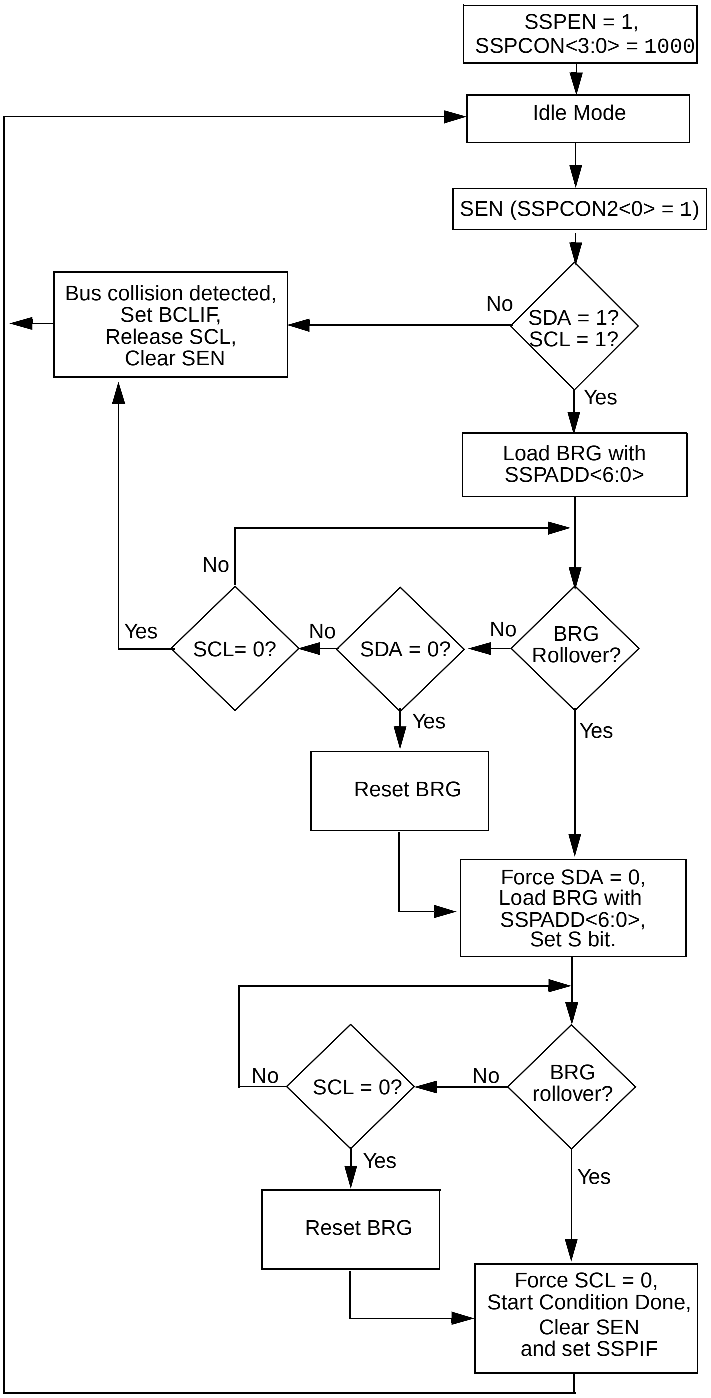


FIGURE 9-18: START CONDITION FLOWCHART



 Advanced Information

9.2.10 I^2C MASTER MODE REPEATED START CONDITION TIMING

A Repeated Start condition occurs when the RSEN bit (SSPCON2<1>) is set high and the I^2C module is in the idle state. When the RSEN bit is set, the SCL pin is asserted low. When the SCL pin is sampled low, the baud rate generator is loaded with the contents of SSPADD<6:0>, and begins counting. The SDA pin is released (brought high) for one baud rate generator count (T_{BRG}). When the baud rate generator times out, if SDA is sampled high, the SCL pin will be de-asserted (brought high). When SCL is sampled high the baud rate generator is re-loaded with the contents of SSPADD<6:0> and begins counting. SDA and SCL must be sampled high for one T_{BRG}. This action is then followed by assertion of the SDA pin (SDA is low) for one T_{BRG} while SCL is high. Following this, the RSEN bit in the SSPCON2 register will be automatically cleared, and the baud rate generator is not reloaded, leaving the SDA pin held low. As soon as a start condition is detected on the SDA and SCL pins, the S bit (SSPSTAT<3>) will be set. The SSPIF bit will not be set until the baud rate generator has timed-out.

Note 1: If RSEN is set while any other event is in progress, it will not take effect.

Note 2: A bus collision during the Repeated Start condition occurs if:

- SDA is sampled low when SCL goes from low to high.
- SCL goes low before SDA is asserted low. This may indicate that another master is attempting to transmit a data "1".

Immediately following the SSPIF bit getting set, the user may write the SSPBUF with the 7-bit address in 7-bit mode, or the default first address in 10-bit mode. After the first eight bits are transmitted and an ACK is received, the user may then transmit an additional eight bits of address (10-bit mode) or eight bits of data (7-bit mode).

9.2.10.1 WCOL STATUS FLAG

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

Note: Because queueing of events is not allowed, writing of the lower 5 bits of SSPCON2 is disabled until the Repeated Start condition is complete.

FIGURE 9-19: REPEAT START CONDITION WAVEFORM

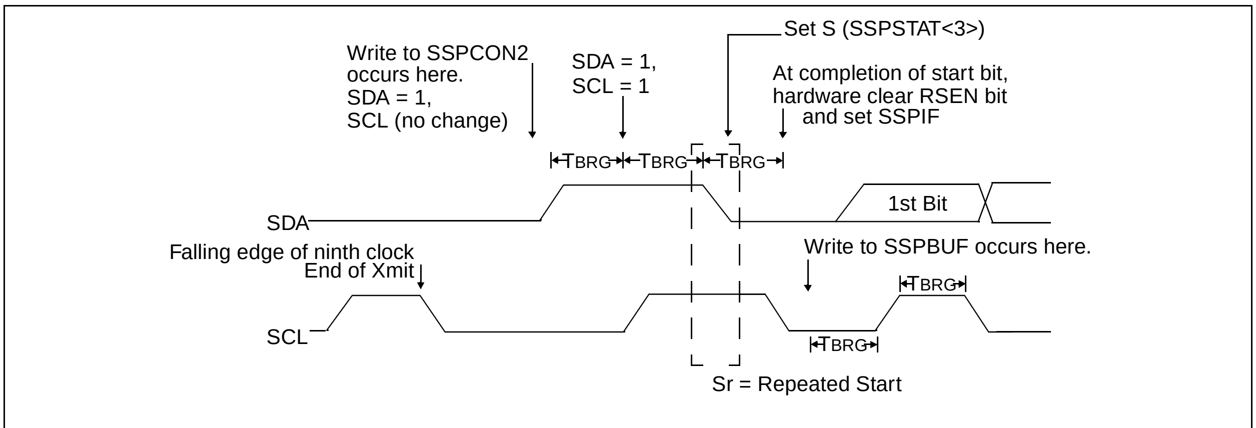
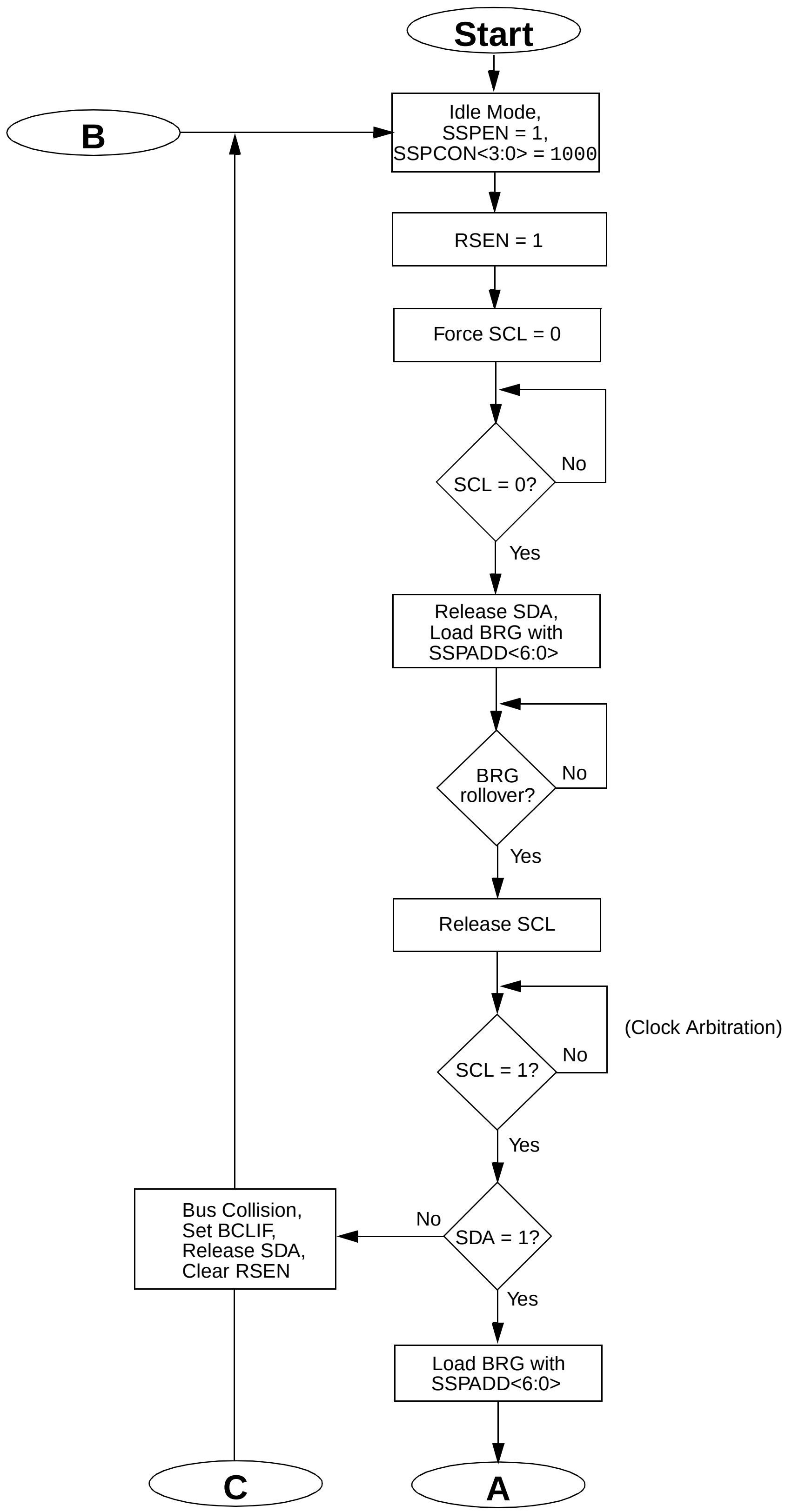
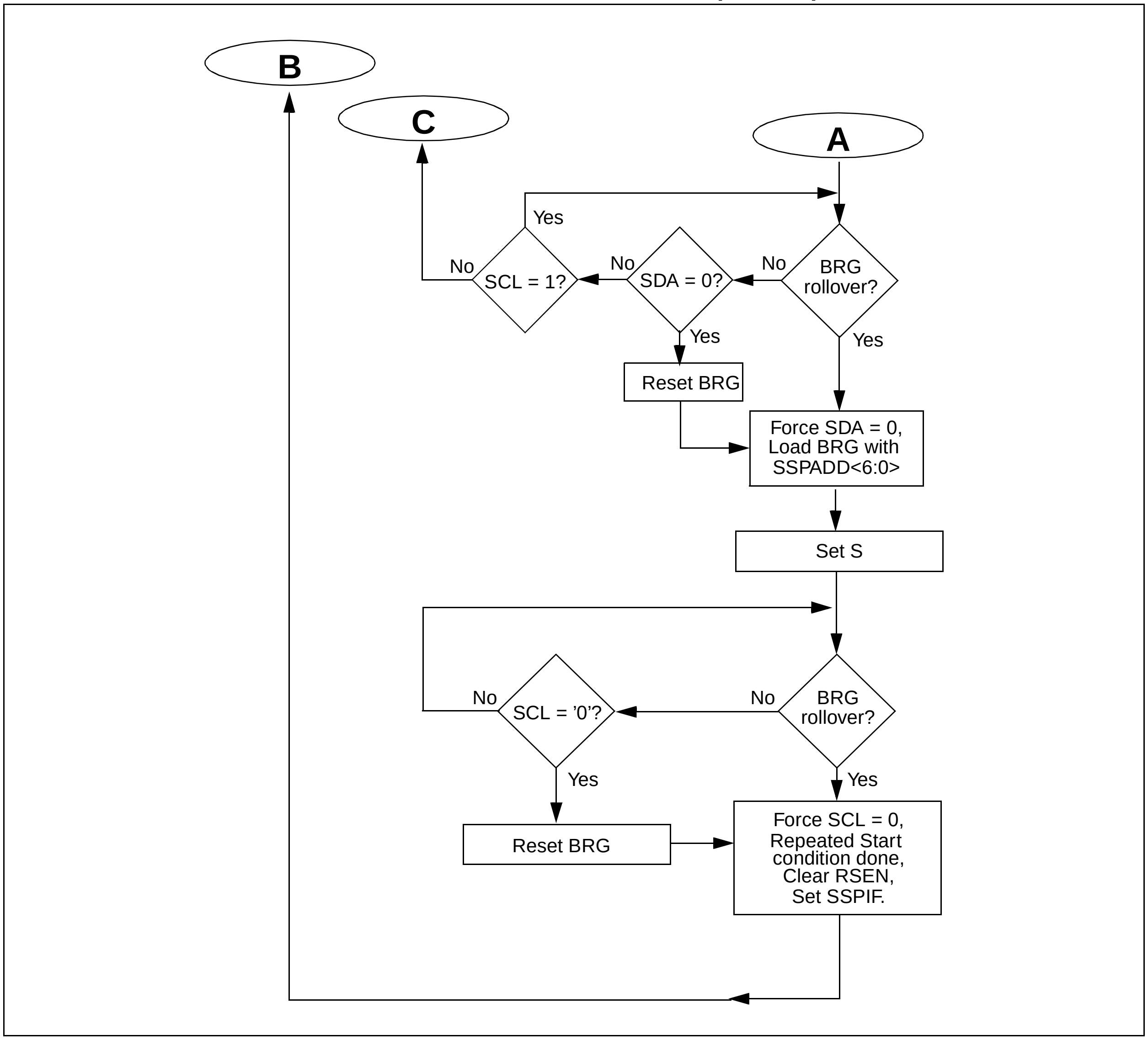


FIGURE 9-20: REPEATED START CONDITION FLOWCHART (PAGE 1)



 Advanced Information

FIGURE 9-21: REPEATED START CONDITION FLOWCHART (PAGE 2)



9.2.11 I^{2}C MASTER MODE TRANSMISSION

Transmission of a data byte, a 7-bit address, or either half of a 10-bit address is accomplished by simply writing a value to the SSPBUF register. This action will set the buffer full flag (BF) and allow the baud rate generator to begin counting and start the next transmission. Each bit of address/data will be shifted out onto the SDA pin after the falling edge of SCL is asserted (see data hold time spec). SCL is held low for one baud rate generator roll over count (T_{BRG}). Data should be valid before SCL is released high (see data setup time spec). When the SCL pin is released high, it is held that way for T_{BRG}, the data on the SDA pin must remain stable for that duration and some hold time after the next falling edge of SCL. After the eighth bit is shifted out (the falling edge of the eighth clock), the BF flag is cleared and the master releases SDA allowing the slave device being addressed to respond with an $\overline{ACK}$ bit during the ninth bit time, if an address match occurs or if data was received properly. The status of $\overline{ACK}$ is read into the ACKDT on the falling edge of the ninth clock. If the master receives an acknowledge, the acknowledge status bit (ACKSTAT) is cleared. If not, the bit is set. After the ninth clock the SSPIF is set, and the master clock (baud rate generator) is suspended until the next data byte is loaded into the SSPBUF leaving SCL low and SDA unchanged (Figure 9-23).

After the write to the SSPBUF, each bit of address will be shifted out on the falling edge of SCL until all seven address bits and the R/$\overline{W}$ bit are completed. On the falling edge of the eighth clock, the master will de-assert the SDA pin allowing the slave to respond with an acknowledge. On the falling edge of the ninth clock, the master will sample the SDA pin to see if the address was recognized by a slave. The status of the ACK bit is loaded into the ACKSTAT status bit (SSPCON2<6>). Following the falling edge of the ninth clock transmission of the address, the SSPIF is set, the BF flag is cleared, and the baud rate generator is turned off until another write to the SSPBUF takes place, holding SCL low and allowing SDA to float.

9.2.11.1 BF STATUS FLAG

In transmit mode, the BF bit (SSPSTAT<0>) is set when the CPU writes to SSPBUF and is cleared when all 8 bits are shifted out.

9.2.11.2 WCOL STATUS FLAG

If the user writes the SSPBUF when a transmit is already in progress (i.e. SSPSR is still shifting out a data byte), then WCOL is set and the contents of the buffer are unchanged (the write doesn’t occur).

WCOL must be cleared in software.

9.2.11.3 ACKSTAT STATUS FLAG

In transmit mode, the ACKSTAT bit (SSPCON2<6>) is cleared when the slave has sent an acknowledge ($\overline{ACK}$ = 0), and is set when the slave does not acknowledge ($\overline{ACK}$ = 1). A slave sends an acknowledge when it has recognized its address (including a general call), or when the slave has properly received its data.

FIGURE 9-22: MASTER TRANSMIT FLOWCHART

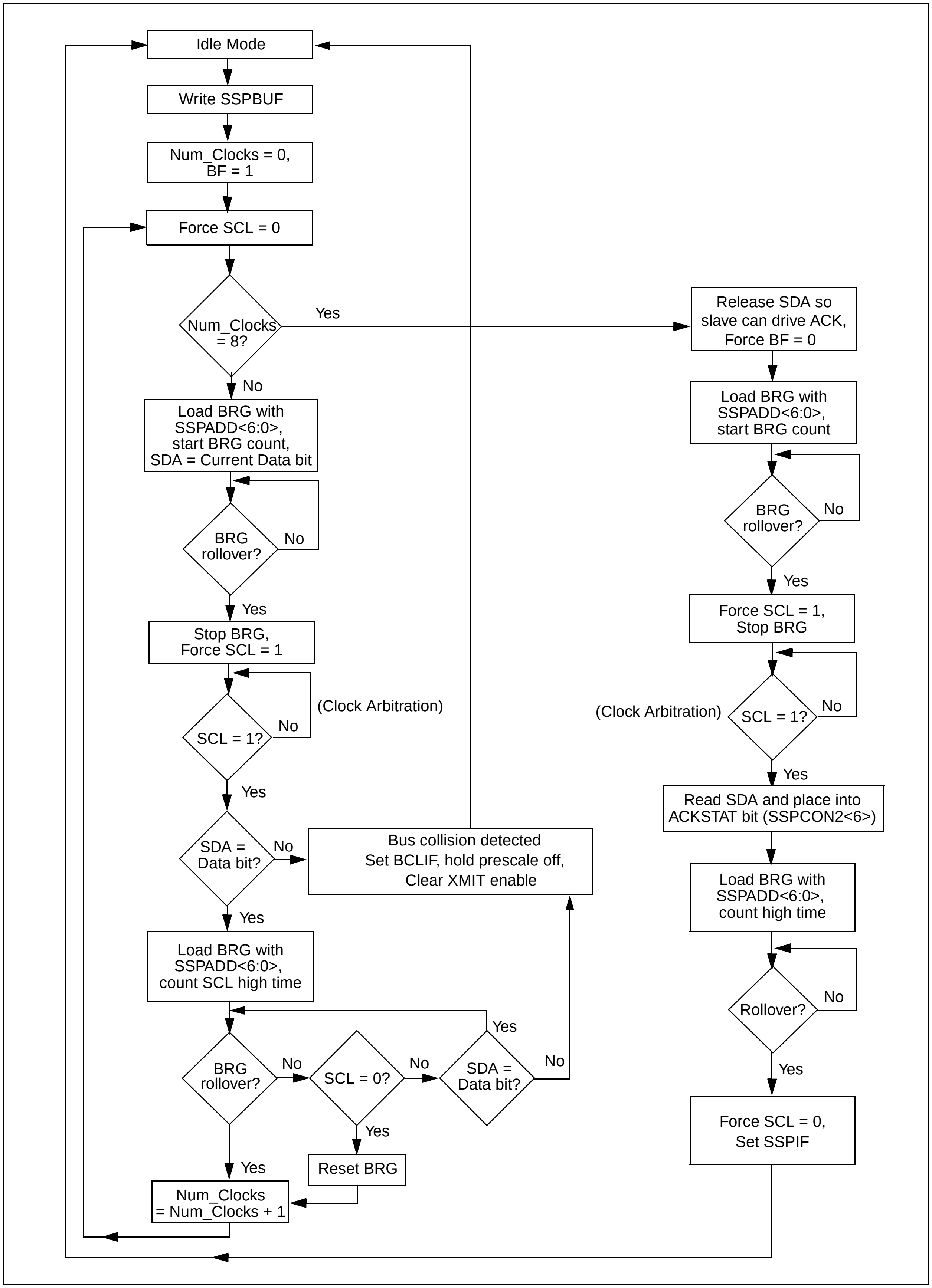
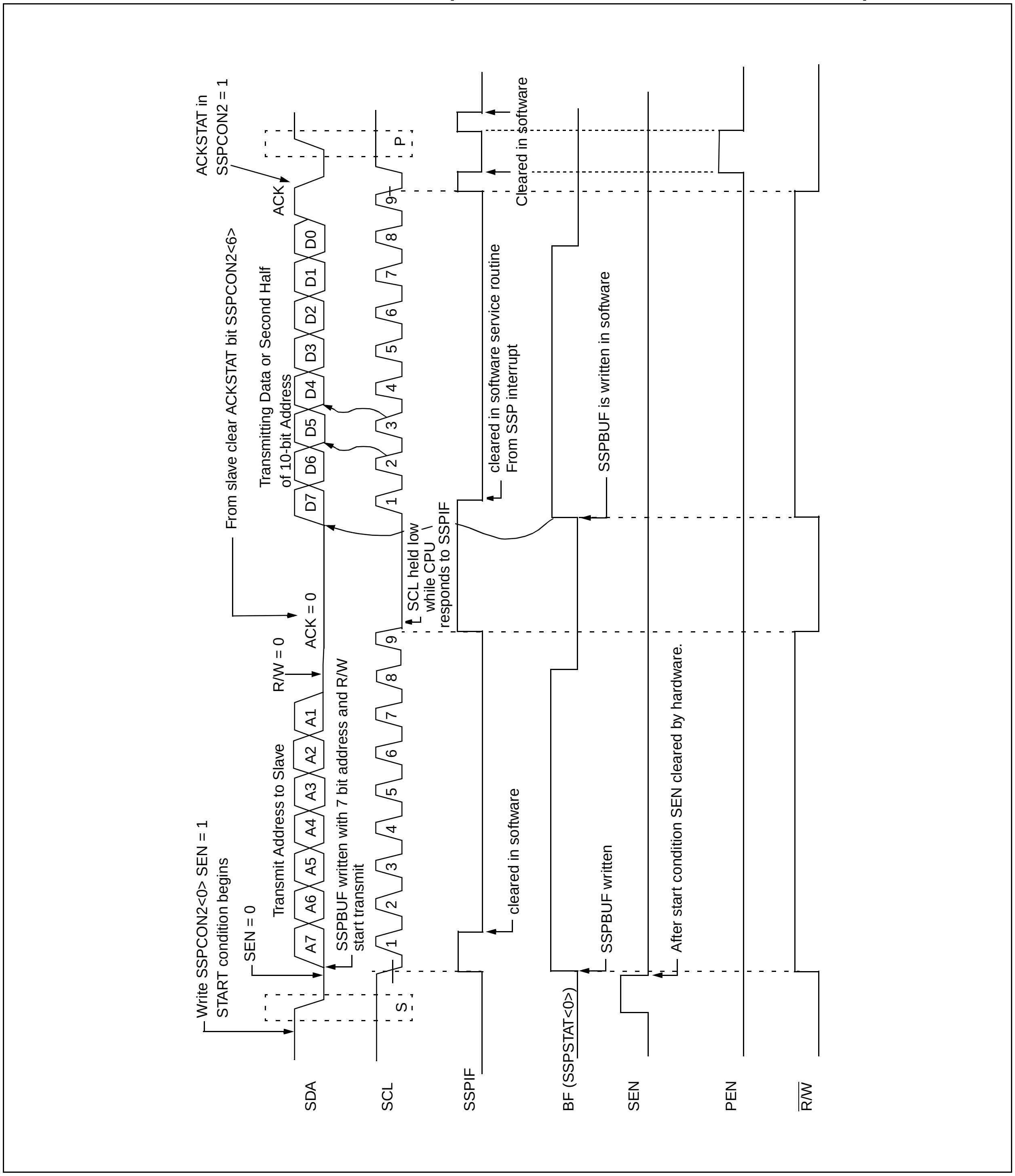


FIGURE 9-23: I²C MASTER MODE TIMING (TRANSMISSION, 7 OR 10-BIT ADDRESS)



 Advanced Information

9.2.12 I^2C MASTER MODE RECEPTION

Master mode reception is enabled by setting the receive enable bit, RCEN (SSPCON2<3>).

Note:	The MSSP Module must be in an IDLE STATE before the RCEN bit is set, or the RCEN bit will be disregarded.

The baud rate generator begins counting, and on each rollover, the state of the SCL pin changes (high to low/ low to high) and data is shifted into the SSPSR. After the falling edge of the eighth clock, the receive enable flag is automatically cleared, the contents of the SSPSR are loaded into the SSPBUF, the BF flag is set, the SSPIF is set, and the baud rate generator is suspended from counting, holding SCL low. The SSP is now in IDLE state, awaiting the next command. When the buffer is read by the CPU, the BF flag is automatically cleared. The user can then send an acknowledge bit at the end of reception, by setting the acknowledge sequence enable bit, ACKEN (SSPCON2<4>).

9.2.12.1 BF STATUS FLAG

In receive operation, BF is set when an address or data byte is loaded into SSPBUF from SSPSR. It is cleared when SSPBUF is read.

9.2.12.2 SSPOV STATUS FLAG

In receive operation, SSPOV is set when 8 bits are received into the SSPSR, and the BF flag is already set from a previous reception.

9.2.12.3 WCOL STATUS FLAG

If the user writes the SSPBUF when a receive is already in progress (i.e. SSPSR is still shifting in a data byte), then WCOL is set and the contents of the buffer are unchanged (the write doesn’t occur).

FIGURE 9-24: MASTER RECEIVER FLOWCHART

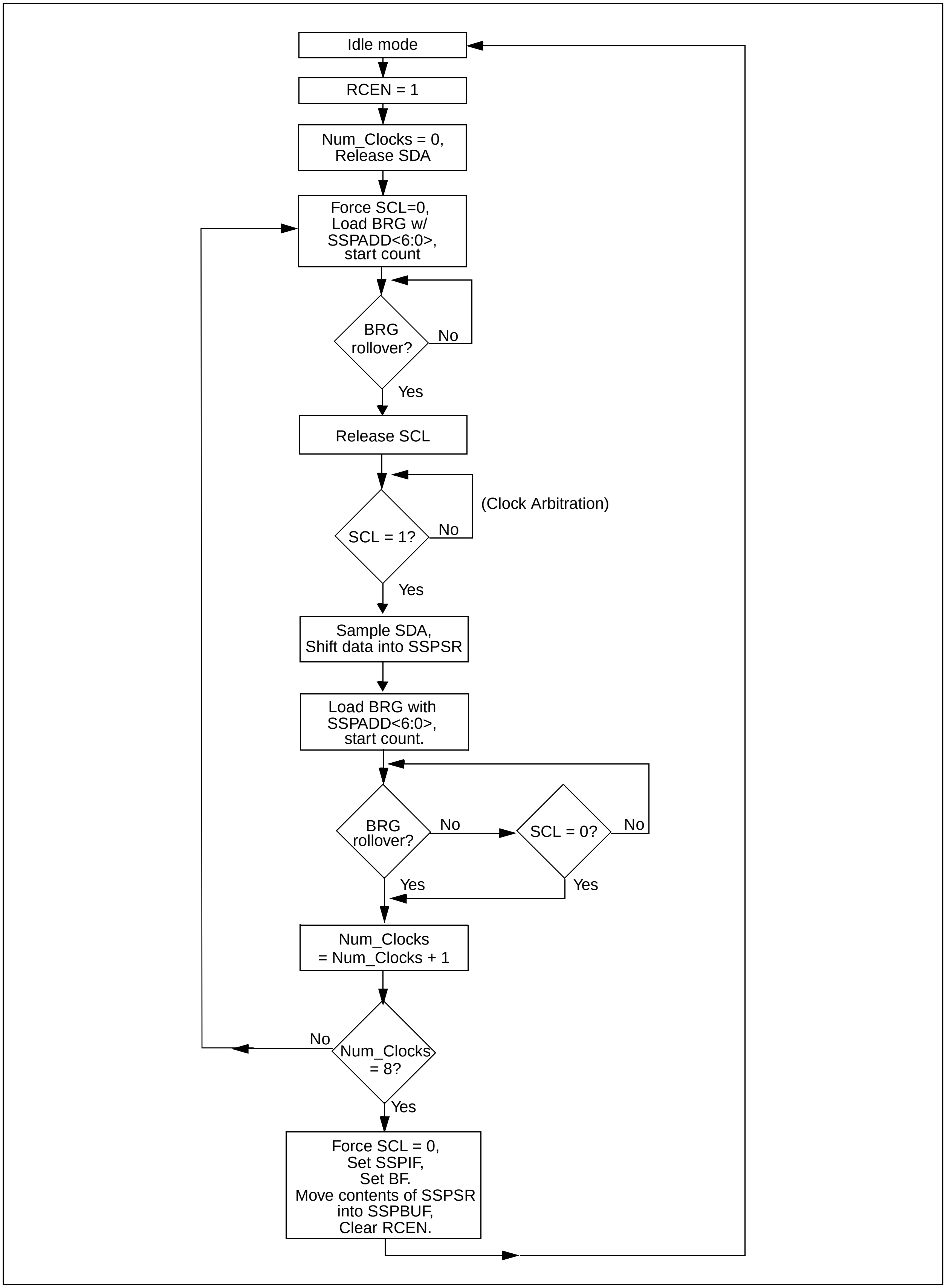
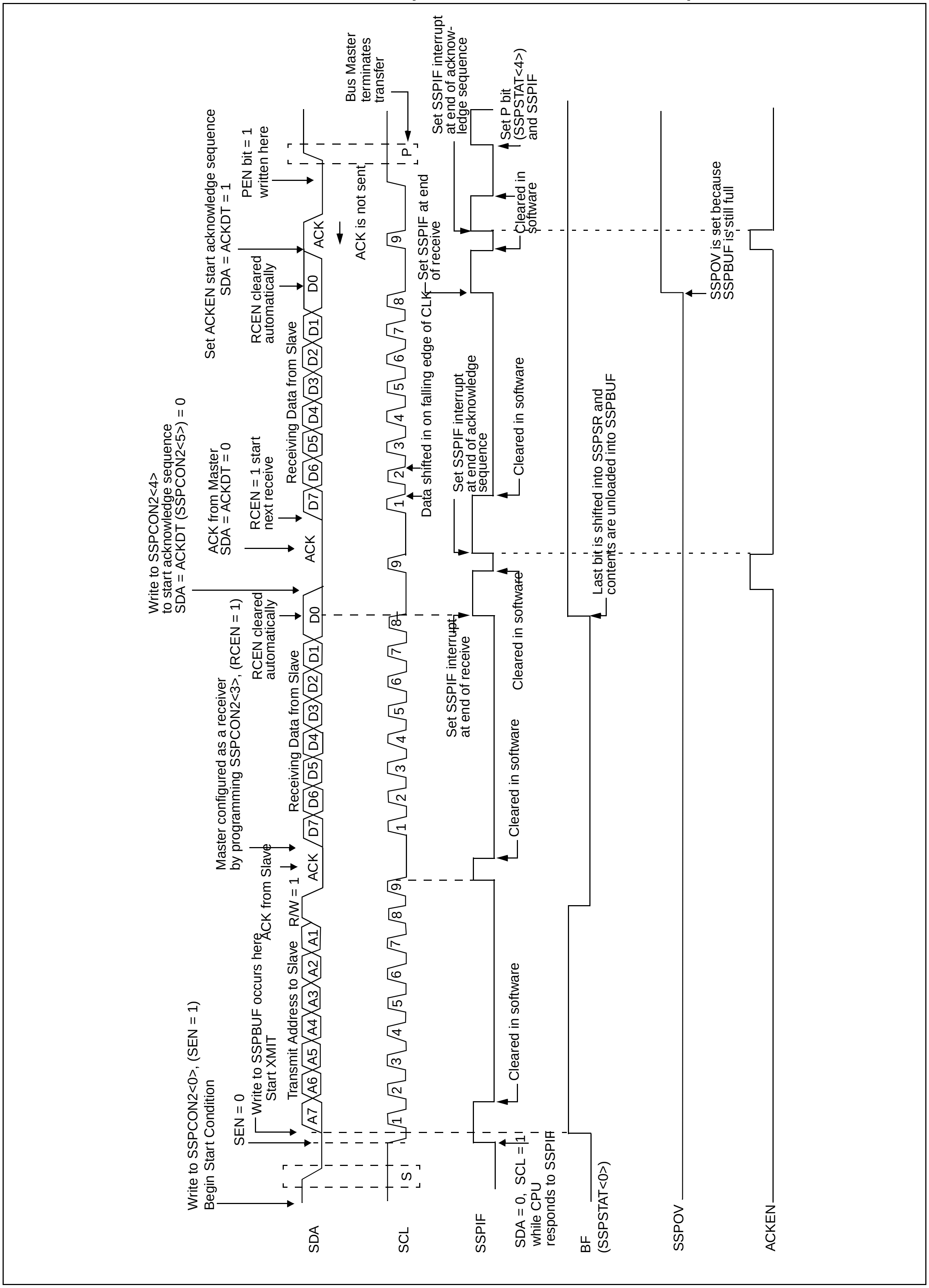


FIGURE 9-25: I²C MASTER MODE TIMING (RECEPTION 7-BIT ADDRESS)



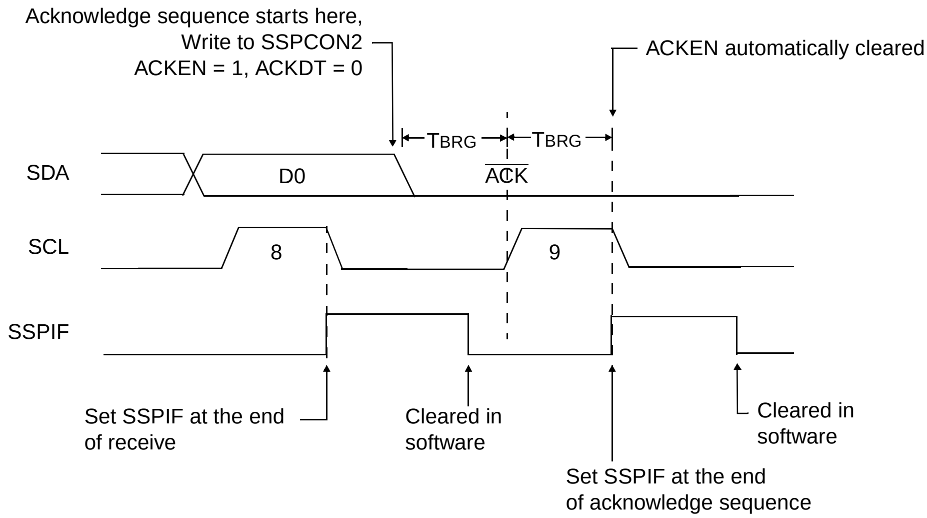
9.2.13 ACKNOWLEDGE SEQUENCE TIMING

An acknowledge sequence is enabled by setting the acknowledge sequence enable bit, ACKEN (SSPCON2<4>). When this bit is set, the SCL pin is pulled low and the contents of the acknowledge data bit is presented on the SDA pin. If the user wishes to generate an acknowledge, then the ACKDT bit should be cleared. If not, the user should set the ACKDT bit before starting an acknowledge sequence. The baud rate generator then counts for one rollover period (TBRG), and the SCL pin is de-asserted (pulled high). When the SCL pin is sampled high (clock arbitration), the baud rate generator counts for TBRG . The SCL pin is then pulled low. Following this, the ACKEN bit is automatically cleared, the baud rate generator is turned off, and the MSSP module then goes into IDLE mode. (Figure 9-26)

9.2.13.1 WCOL STATUS FLAG

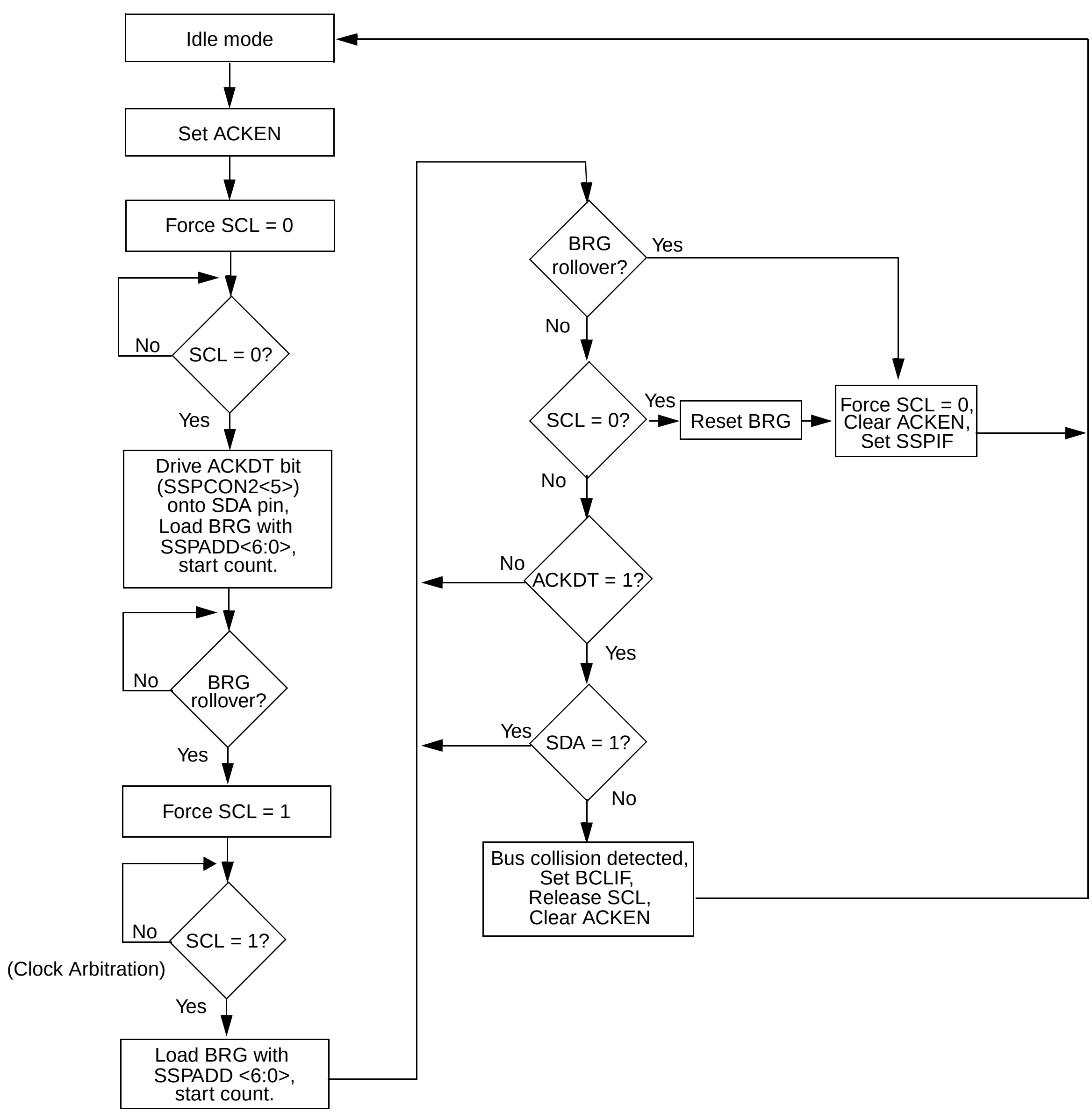
If the user writes the SSPBUF when an acknowledged sequence is in progress, then WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

FIGURE 9-26: ACKNOWLEDGE SEQUENCE WAVEFORM



Note: TBRG = one baud rate generator period.

FIGURE 9-27: ACKNOWLEDGE FLOWCHART



9.2.14 STOP CONDITION TIMING

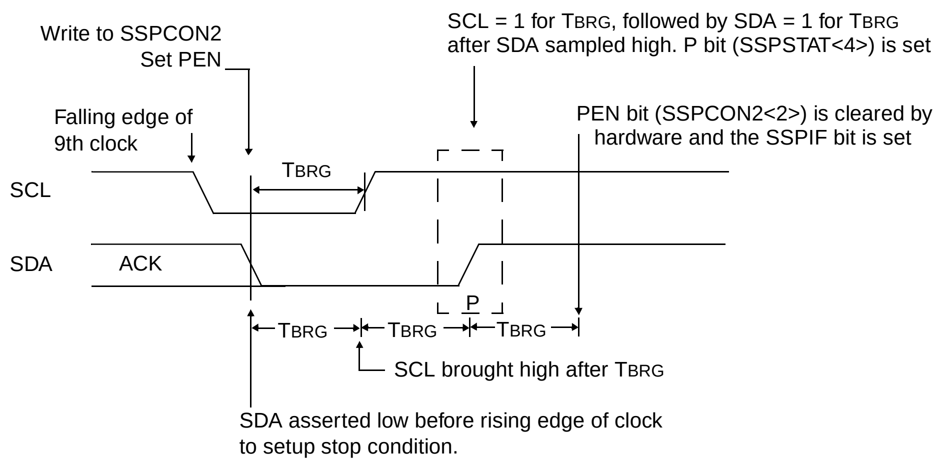
A stop bit is asserted on the SDA pin at the end of a receive/transmit by setting the Stop Sequence Enable bit PEN (SSPCON2<2>). At the end of a receive/transmit, the SCL line is held low after the falling edge of the ninth clock. When the PEN bit is set, the master will assert the SDA line low . When the SDA line is sampled low, the baud rate generator is reloaded and counts down to 0. When the baud rate generator times out, the SCL pin will be brought high and one T_{BRG} (baud rate generator rollover count) later, the SDA pin will be de-asserted. When the SDA pin is sampled high while SCL is high, the P bit (SSPSTAT<4>) is set. A T_{BRG} later the PEN bit is cleared and the SSPIF bit is set (Figure 9-28).

Whenever the firmware decides to take control of the bus, it will first determine if the bus is busy by checking the S and P bits in the SSPSTAT register. If the bus is busy, then the CPU can be interrupted (notified) when a Stop bit is detected (i.e. bus is free).

9.2.14.1 WCOL STATUS FLAG

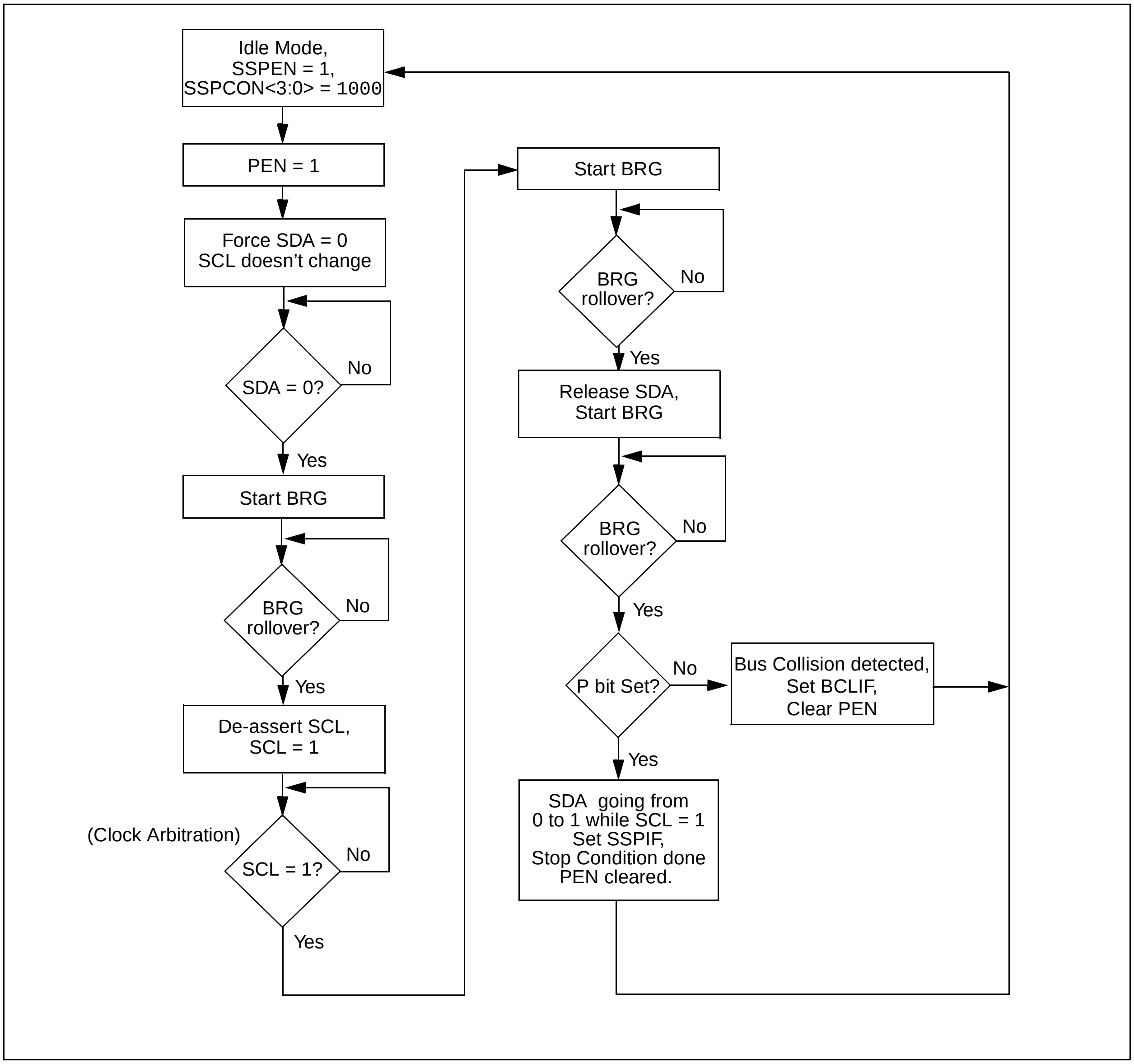
If the user writes the SSPBUF when a STOP sequence is in progress, then WCOL is set and the contents of the buffer are unchanged (the write doesn't occur).

FIGURE 9-28: STOP CONDITION RECEIVE OR TRANSMIT MODE



Note: T_{BRG} = one baud rate generator period.

FIGURE 9-29: STOP CONDITION FLOWCHART



9.2.15 CLOCK ARBITRATION

Clock arbitration occurs when the master, during any receive, transmit or repeated start/stop condition, de-asserts the SCL pin (SCL allowed to float high). When the SCL pin is allowed to float high, the baud rate generator (BRG) is suspended from counting until the SCL pin is actually sampled high. When the SCL pin is sampled high, the baud rate generator is reloaded with the contents of SSPADD<6:0> and begins counting. This ensures that the SCL high time will always be at least one BRG rollover count in the event that the clock is held low by an external device (Figure 9-30).

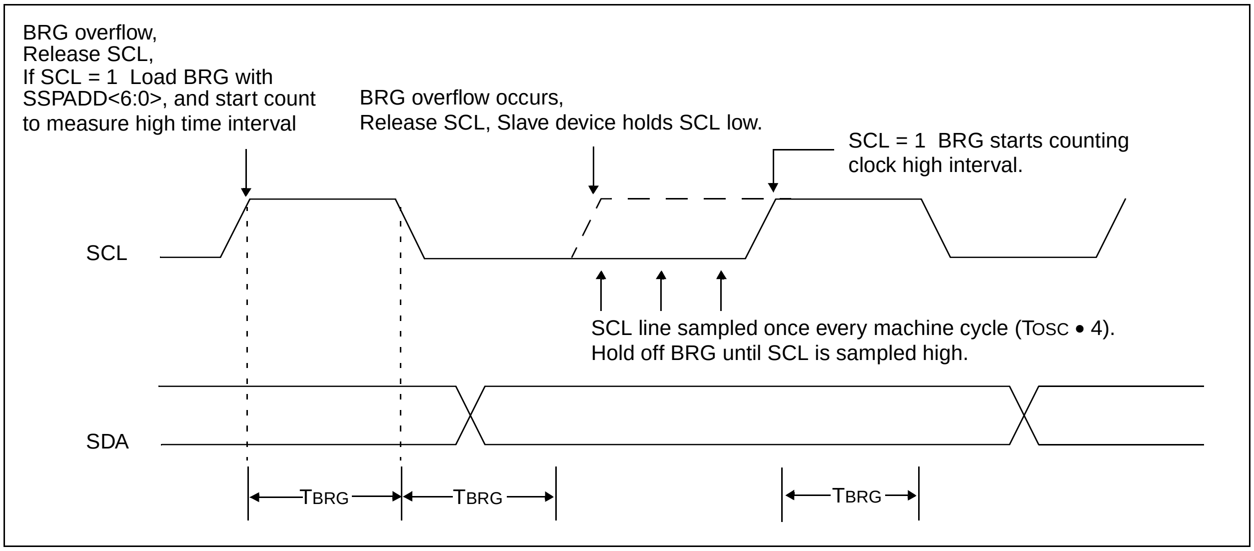
9.2.16 SLEEP OPERATION

While in sleep mode, the I^2C module can receive addresses or data, and when an address match or complete byte transfer occurs, wake the processor from sleep (if the SSP interrupt is enabled).

9.2.17 EFFECTS OF A RESET

A reset disables the MSSP module and terminates the current transfer.

FIGURE 9-30: CLOCK ARBITRATION TIMING IN MASTER TRANSMIT MODE



 Advanced Information

9.2.18 MULTI -MASTER COMMUNICATION, BUS COLLISION, AND BUS ARBITRATION

Multi-Master mode support is achieved by bus arbitration. When the master outputs address/data bits onto the SDA pin, arbitration takes place when the master outputs a '1' on SDA by letting SDA float high and another master asserts a '0'. When the SCL pin floats high, data should be stable. If the expected data on SDA is a '1' and the data sampled on the SDA pin = '0', then a bus collision has taken place. The master will set the Bus Collision Interrupt Flag, BCLIF, and reset the I^2C port to its IDLE state. (Figure 9-31).

If a transmit was in progress when the bus collision occurred, the transmission is halted, the BF flag is cleared, the SDA and SCL lines are de-asserted, and the SSPBUF can be written to. When the user services the bus collision interrupt service routine, and if the I^2C bus is free, the user can resume communication by asserting a START condition.

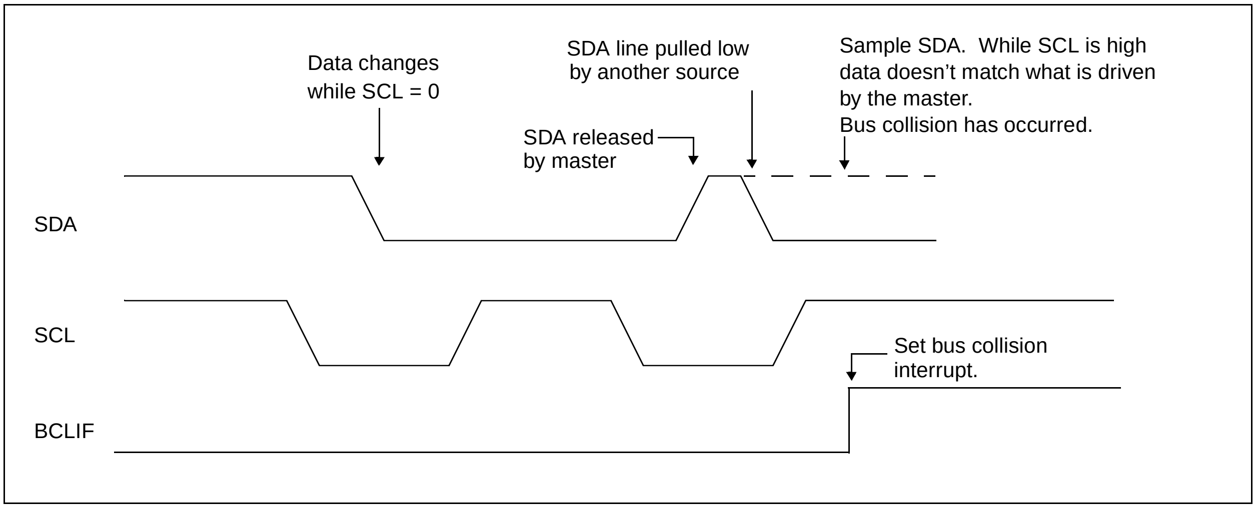
If a START, Repeated Start, STOP or Acknowledge condition was in progress when the bus collision occurred, the condition is aborted, the SDA and SCL lines are de-asserted, and the respective control bits in the SSPCON2 register are cleared. When the user services the bus collision interrupt service routine, and if the I^2C bus is free, the user can resume communication by asserting a START condition.

The Master will continue to monitor the SDA and SCL pins, and if a STOP condition occurs, the SSPIF bit will be set.

A write to the SSPBUF will start the transmission of data at the first data bit, regardless of where the transmitter left off when bus collision occurred.

In multi-master mode, the interrupt generation on the detection of start and stop conditions allows the determination of when the bus is free. Control of the I^2C bus can be taken when the P bit is set in the SSPSTAT register, or the bus is idle and the S and P bits are cleared.

FIGURE 9-31: BUS COLLISION TIMING FOR TRANSMIT AND ACKNOWLEDGE



9.2.18.1 BUS COLLISION DURING A START CONDITION

During a START condition, a bus collision occurs if:

a) SDA or SCL are sampled low at the beginning of the START condition (Figure 9-32).
b) SCL is sampled low before SDA is asserted low. (Figure 9-33).

During a START condition both the SDA and the SCL pins are monitored.

If:

the SDA pin is already low
or the SCL pin is already low,

then:

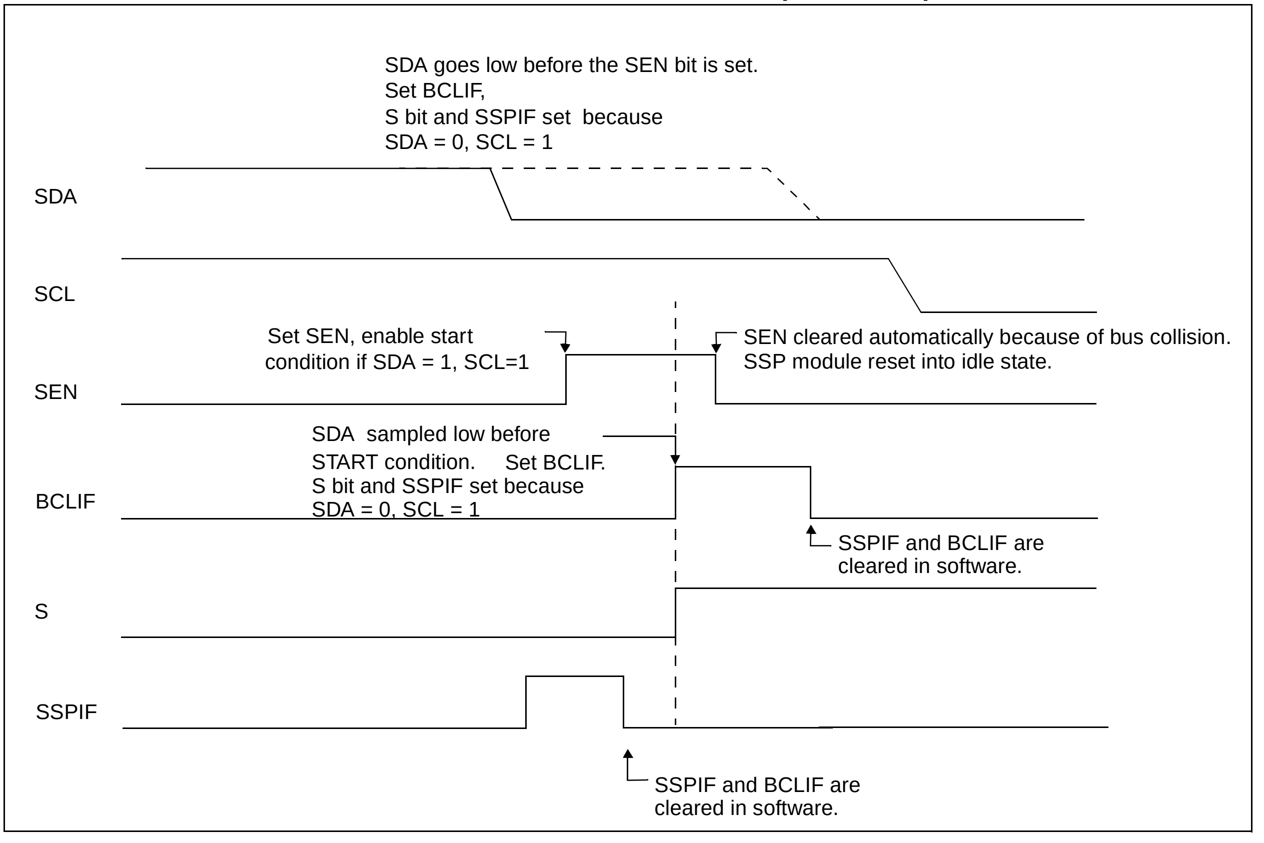
the START condition is aborted,
and the BCLIF flag is set,
and the SSP module is reset to its IDLE state (Figure 9-32).

The START condition begins with the SDA and SCL pins de-asserted. When the SDA pin is sampled high, the baud rate generator is loaded from SSPADD<6:0> and counts down to 0. If the SCL pin is sampled low while SDA is high, a bus collision occurs, because it is assumed that another master is attempting to drive a data ’1’ during the START condition.

If the SDA pin is sampled low during this count, the BRG is reset and the SDA line is asserted early (Figure 9-34). If however a ’1’ is sampled on the SDA pin, the SDA pin is asserted low at the end of the BRG count. The baud rate generator is then reloaded and counts down to 0, and during this time, if the SCL pins is sampled as ’0’, a bus collision does not occur. At the end of the BRG count the SCL pin is asserted low.

Note: The reason that bus collision is not a factor during a START condition is that no two bus masters can assert a START condition at the exact same time. Therefore, one master will always assert SDA before the other. This condition does not cause a bus collision, because the two masters must be allowed to arbitrate the first address following the START condition. If the address is the same, arbitration must be allowed to continue into the data portion, REPEATED START or STOP conditions.

FIGURE 9-32: BUS COLLISION DURING START CONDITION (SDA ONLY)



 Advanced Information

FIGURE 9-33: BUS COLLISION DURING START CONDITION (SCL = 0)

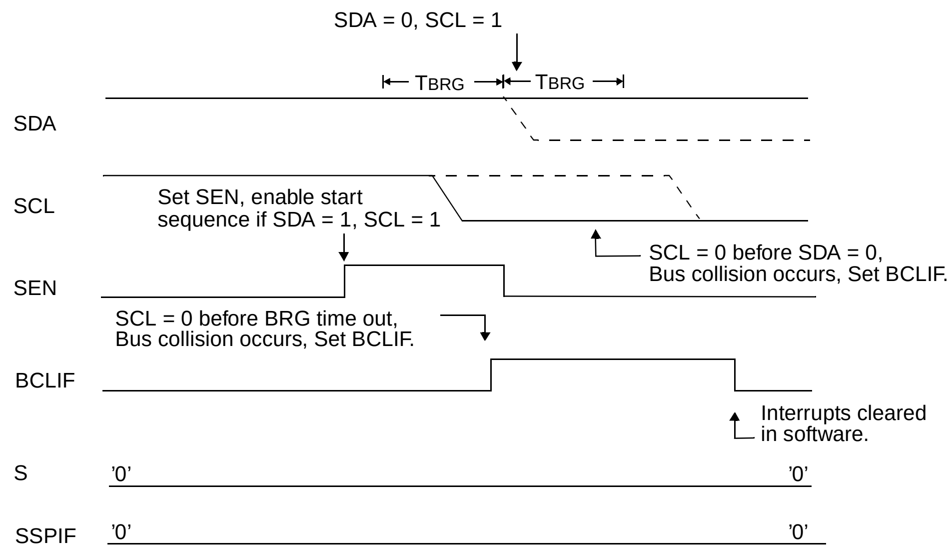
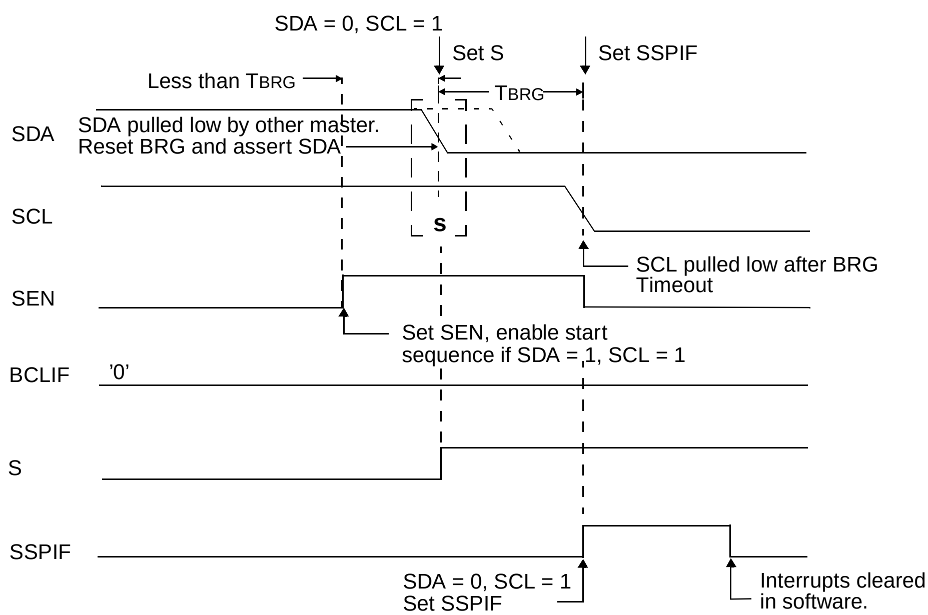


FIGURE 9-34: BRG RESET DUE TO SDA COLLISION DURING START CONDITION



9.2.18.2 BUS COLLISION DURING A REPEATED START CONDITION

During a Repeated Start condition, a bus collision occurs if:

a) A low level is sampled on SDA when SCL goes from low level to high level.
b) SCL goes low before SDA is asserted low, indicating that another master is attempting to transmit a data '1'.

When the user de-asserts SDA and the pin is allowed to float high, the BRG is loaded with SSPADD<6:0>, and counts down to 0. The SCL pin is then de-asserted, and when sampled high, the SDA pin is sampled. If SDA is low, a bus collision has occurred (i.e. another master is attempting to transmit a data '0'). If however SDA is sampled high, then the BRG is reloaded and begins counting. If SDA goes from high to low before the BRG times out, no bus collision occurs, because no two masters can assert SDA at exactly the same time.

If, however, SCL goes from high to low before the BRG times out and SDA has not already been asserted, then a bus collision occurs. In this case, another master is attempting to transmit a data '1' during the Repeated Start condition.

If at the end of the BRG time out both SCL and SDA are still high, the SDA pin is driven low, the BRG is reloaded, and begins counting. At the end of the count, regardless of the status of the SCL pin, the SCL pin is driven low and the Repeated Start condition is complete (Figure 9-35).

FIGURE 9-35: BUS COLLISION DURING A REPEATED START CONDITION (CASE 1)

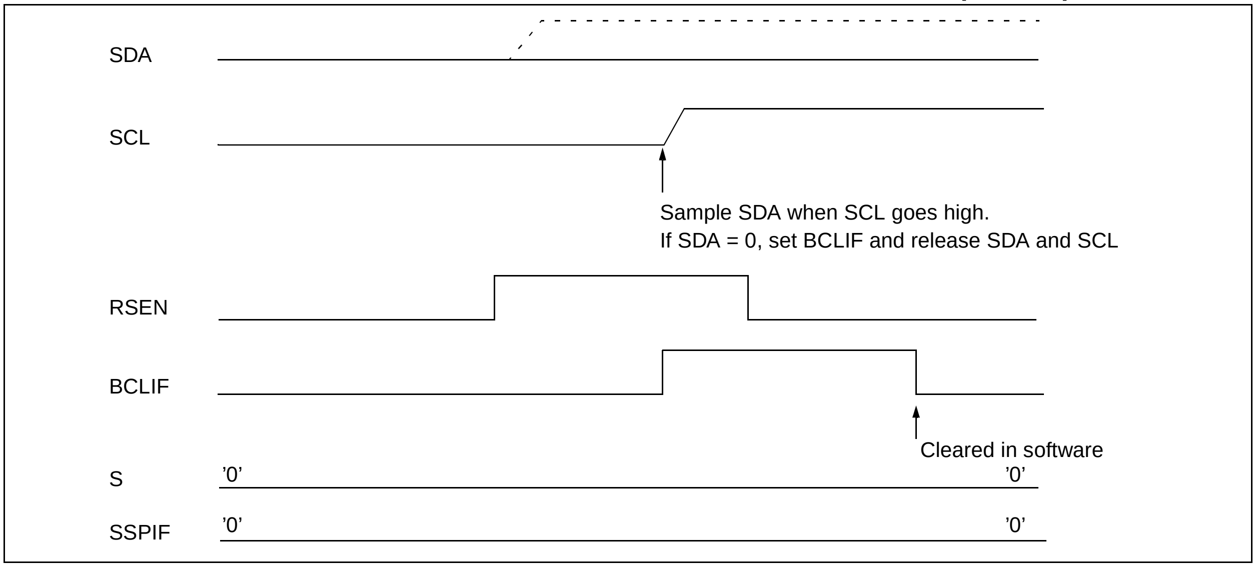
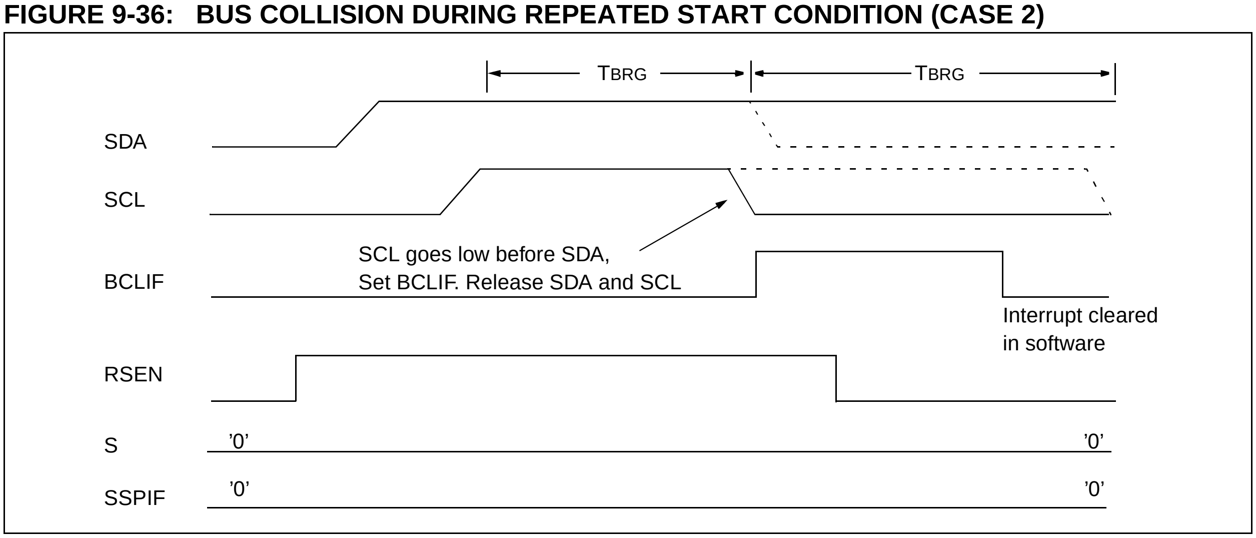


FIGURE 9-36: BUS COLLISION DURING REPEATED START CONDITION (CASE 2)



9.2.18.3 BUS COLLISION DURING A STOP CONDITION

Bus collision occurs during a STOP condition if:

a) After the SDA pin has been de-asserted and allowed to float high, SDA is sampled low after the BRG has timed out.
b) After the SCL pin is de-asserted, SCL is sampled low before SDA goes high.

The STOP condition begins with SDA asserted low. When SDA is sampled low, the SCL pin is allow to float. When the pin is sampled high (clock arbitration), the baud rate generator is loaded with SSPADD<6:0> and counts down to 0. After the BRG times out SDA is sampled. If SDA is sampled low, a bus collision has occurred. This is due to another master attempting to drive a data '0'. If the SCL pin is sampled low before SDA is allowed to float high, a bus collision occurs. This is another case of another master attempting to drive a data '0' (Figure 9-37).

FIGURE 9-37: BUS COLLISION DURING A STOP CONDITION (CASE 1)

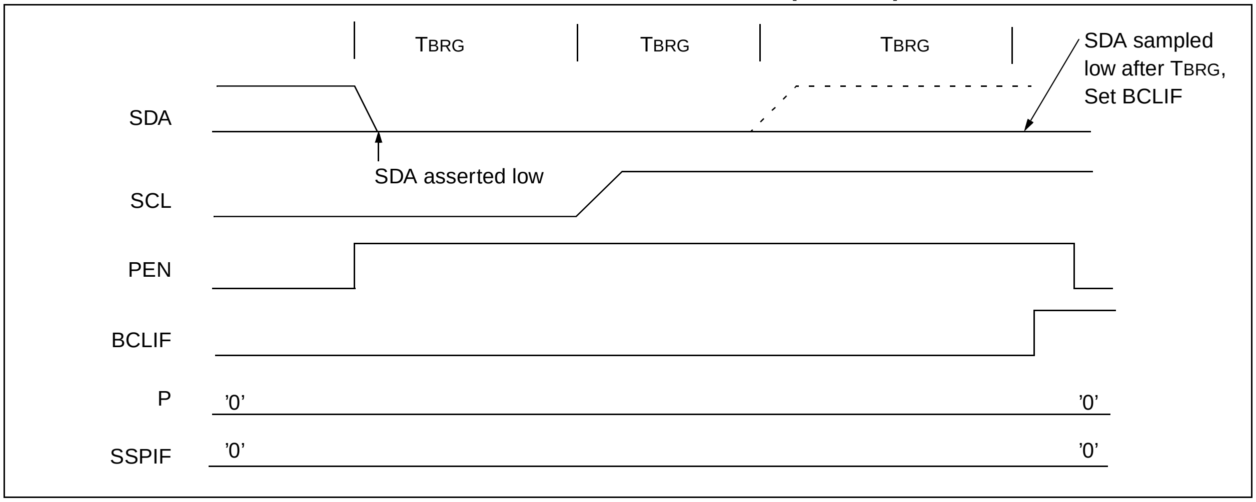
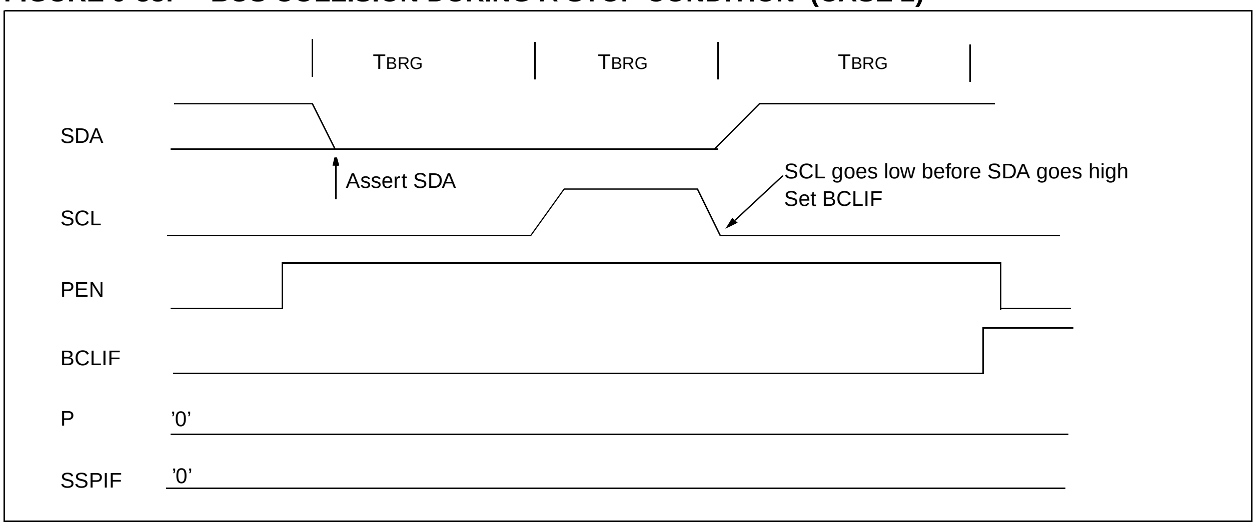


FIGURE 9-38: BUS COLLISION DURING A STOP CONDITION (CASE 2)



9.2.19 CONNECTION CONSIDERATIONS FOR I2C BUS

For standard-mode I2C bus devices, the values of resistors R_p and R_s in Figure 9-39 depends on the following parameters

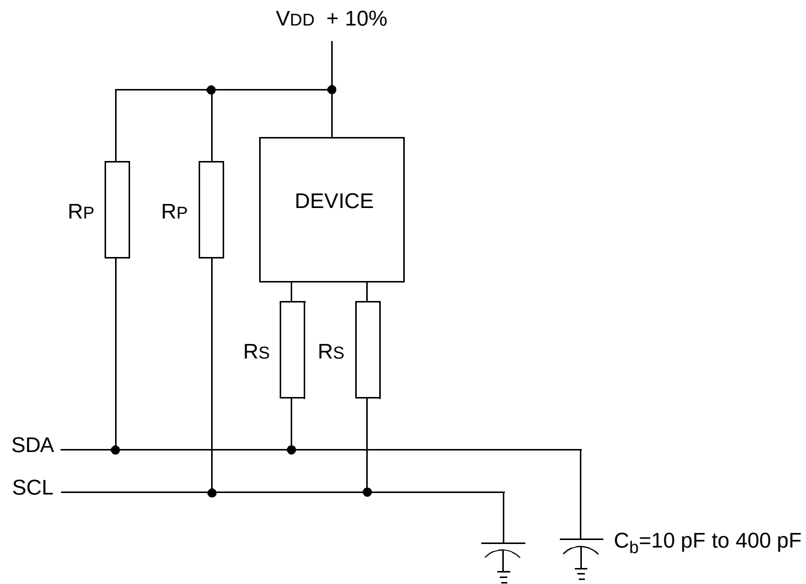
- Supply voltage
- Bus capacitance
- Number of connected devices (input current + leakage current).

The supply voltage limits the minimum value of resistor R_p due to the specified minimum sink current of 3 mA at VOL max = 0.4V for the specified output stages. For example, with a supply voltage of VDD = 5V±10% and VOL max = 0.4V at 3 mA, $R_{p\ min}$ = (5.5-0.4)/0.003 = 1.7 kΩ. VDD as a function of R_p is shown in Figure 9-39. The desired noise margin of 0.1VDD for the low level limits the maximum value of R_s. Series resistors are optional and used to improve ESD susceptibility.

The bus capacitance is the total capacitance of wire, connections, and pins. This capacitance limits the maximum value of R_p due to the specified rise time (Figure 9-39).

The SMP bit is the slew rate control enabled bit. This bit is in the SSPSTAT register, and controls the slew rate of the I/O pins when in I2C mode (master or slave).

FIGURE 9-39: SAMPLE DEVICE CONFIGURATION FOR I2C BUS



Note: I2C devices with input levels related to VDD must have one common supply line to which the pull-up resistor is also connected.

TABLE 9-3: REGISTERS ASSOCIATED WITH I2C OPERATION

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR, BOR	MCLR, WDT
0Bh, 8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF	-0-- 0000	-0-- 0000
8Ch	PIE1	—	ADIE	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE	-0-- 0000	-0-- 0000
0Dh	PIR2	LVDIF	—	—	—	BCLIF	—	—	CCP2IF	0--- 0--0	0--- 0--0
8Dh	PIE2	LVDIE	—	—	—	BCLIE	—	—	CCP2IE	0--- 0--0	0--- 0--0
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	uuuu uuuu
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
91h	SSPCON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
94h	SSPSTAT	SMP	CKE	D/A	P	S	R/W	UA	BF	0000 0000	0000 0000

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by the MSSP in I2C mode.

10.0 VOLTAGE REFERENCE MODULE AND LOW-VOLTAGE DETECT

The Voltage Reference module provides reference voltages for the Brown-out Reset circuitry, the Low-voltage Detect circuitry and the A/D converter.

The source for the reference voltages comes from the bandgap reference circuit. The bandgap circuit is energized anytime the reference voltage is required by the other sub-modules, and is powered down when not in use. The control registers for this module are LVDCON and REFCON, as shown in Register 10-1 and Figure 10-2.

REGISTER 10-1: LOW-VOLTAGE DETECT CONTROL REGISTER (LVDCON: 9Ch)

U-0	U-0	R-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-1
—	—	BGST	LVDEN	LV3	LV2	LV1	LV0
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7-6: **Unimplemented:** Read as '0'

bit 5: **BGST:** Bandgap Stable Status Flag bit
1 = Indicates that the bandgap voltage is stable, and LVD interrupt is reliable
0 = Indicates that the bandgap voltage is not stable, and LVD interrupt should not be enabled

bit 4: **LVDEN:** Low-voltage Detect Power Enable bit
1 = Enables LVD, powers up bandgap circuit and reference generator
0 = Disables LVD, powers down bandgap circuit if unused by BOR or VRH/VRL

bit 3-0: **LV<3:0>:** Low Voltage Detection Limit bits (1)
1111 = External analog input is used
1110 = 4.5V
1101 = 4.2V
1100 = 4.0V
1011 = 3.8V
1010 = 3.6V
1001 = 3.5V
1000 = 3.3V
0111 = 3.0V
0110 = 2.8V
0101 = 2.7V
0100 = 2.5V
0011 = Reserved. Do not use.
0010 = Reserved. Do not use.
0001 = Reserved. Do not use.
0000 = Reserved. Do not use.

Note 1: These are the minimum trip points for the LVD. See Table 15-3 for the trip point tolerances. Selection of reserved setting may result in an inadvertent interrupt.

REGISTER 10-2: VOLTAGE REFERENCE CONTROL REGISTER (REFCON: 9BH)

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
VRHEN	VRLEN	VRHOEN	VRLOEN	—	—	—	—
bit7							bit0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as ‘0’
- n = Value at POR reset

bit 7: **VRHEN:** Voltage Reference High Enable bit (VRH = 4.096V nominal)
1 = Enabled, powers up reference generator
0 = Disabled, powers down reference generator if unused by LVD, BOR, or VRL

bit 6: **VRLEN:** Voltage Reference Low Enable bit (VRL = 2.048V nominal)
1 = Enabled, powers up reference generator
0 = Disabled, powers down reference generator if unused by LVD, BOR, or VRH

bit 5: **VRHOEN:** High Voltage Reference Output Enable bit
1 = Enabled, VRH analog reference is output on RA3 if enabled (VRHEN = 1)
0 = Disabled, analog reference is used internally only

bit 4: **VRLOEN:** Low Voltage Reference Output Enable bit
1 = Enabled, VRL analog reference is output on RA2 if enabled (VRLEN = 1)
0 = Disabled, analog reference is used internally only

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

10.1 Bandgap Voltage Reference

The bandgap module generates a stable voltage reference of over a range of temperatures and device supply voltages. This module is enabled anytime any of the following are enabled:

- Brown-out Reset
- Low-voltage Detect
- Either of the internal analog references (VRH, VRL)

Whenever the above are all disabled, the bandgap module is disabled and draws no current.

10.2 Internal VREF for A/D Converter

The bandgap output voltage is used to generate two stable references for the A/D converter module. These references are enabled in software to provide the user with the means to turn them on and off in order to minimize current consumption. Each reference can be individually enabled.

The VRH reference is enabled with control bit VRHEN (REFCON<7>). When this bit is set, the gain amplifier is enabled. After a specified start-up time a stable reference of 4.096V nominal is generated and can be used by the A/D converter as a reference input.

The VRL reference is enabled by setting control bit VRLEN (REFCON<6>). When this bit is set, the gain amplifier is enabled. After a specified start up time a stable reference of 2.048V nominal is generated and can be used by the A/D converter as a reference input.

Each voltage reference is available for external use via VRL and VRH pins.

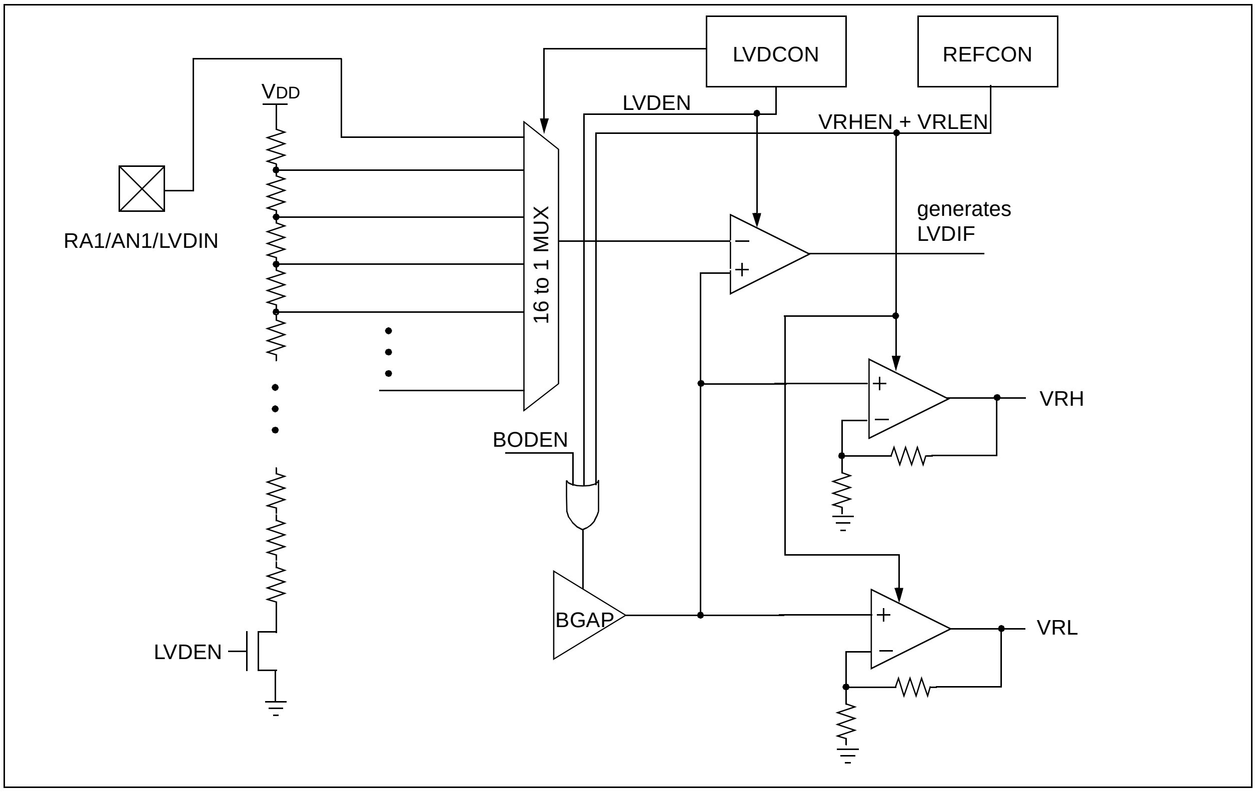
Each reference, if enabled, can be output on an external pin by setting the VRHOEN (high reference output enable) or VRLOEN (low reference output enable) control bit. If the reference is not enabled, the VRHOEN and VRLOEN bits will have no effect on the corresponding pin. The device specific pin can then be used as general purpose I/O.

Note:	If VRH or VRL is enabled and the other reference (VRL or VRH), the BOR, and the LVD modules are not enabled, the bandgap will require a start-up time before the bandgap reference is stable. Before using the internal VRH or VRL reference, ensure that the bandgap reference voltage is stable by monitoring the BGST bit in the LVDCON register. The voltage references will not be reliable until the bandgap is stable as shown by BGST being set.

10.3 Low-voltage Detect (LVD)

This module is used to generate an interrupt when the supply voltage falls below a specified "trip" voltage. This module operates completely under software control. This allows a user to power the module on and off to periodically monitor the supply voltage, and thus minimize total current consumption.

FIGURE 10-1: BLOCK DIAGRAM OF LVD AND VOLTAGE REFERENCE CIRCUIT



The LVD module is enabled by setting the LVDEN bit in the LVDCON register. The "trip point" voltage is the minimum supply voltage level at which the device can operate before the LVD module asserts an interrupt. When the supply voltage is equal to or less than the trip point, the module will generate an interrupt signal setting interrupt flag bit LVDIF. If interrupt enable bit LVDIE was set, then an interrupt is generated. The LVD interrupt can wake the device from sleep. The "trip point" voltage is software programmable to any one of 16 values, five of which are reserved (See Figure 10-1). The trip point is selected by programming the LV<3:0> bits (LVDCON<3:0>).

Note:	The LVDIF bit can not be cleared until the supply voltage rises above the LVD trip point. If interrupts are enabled, clear the LVDIE bit once the first LVD interrupt occurs to prevent reentering the interrupt service routine immediately after exiting the ISR.

Once the LV bits have been programmed for the specified trip voltage, the low-voltage detect circuitry is then enabled by setting the LVDEN (LVDCON<4>) bit.

If the bandgap reference voltage is previously unused by either the brown-out circuitry or the voltage reference circuitry, then the bandgap circuit requires a time to start-up and become stable before a low voltage condition can be reliably detected. The low-voltage interrupt flag is prevented from being set until the bandgap has reached a stable reference voltage.

When the bandgap is stable the BGST (LVDCON<5>) bit is set indicating that the low-voltage interrupt flag bit is released to be set if VDD is equal to or less than the LVD trip point.

10.3.1 EXTERNAL ANALOG VOLTAGE INPUT

The LVD module has an additional feature that allows the user to supply the trip voltage to the module from an external source. This mode is enabled when LV<3:0> = 1111. When these bits are set the comparator input is multiplexed from an external input pin (RA1/AN1/LVDIN).

NOTES:

11.0 ANALOG-TO-DIGITAL CONVERTER (A/D) MODULE

The analog-to-digital (A/D) converter module has six inputs for the PIC16C717/770/771.

The PIC16C717 analog-to-digital converter (A/D) allows conversion of an analog input signal to a corresponding 10-bit digital value, while the A/D converter in the PIC16C770/771 allows conversion to a corresponding 12-bit digital value. The A/D module has up to 6 analog inputs, which are multiplexed into one sample and hold. The output of the sample and hold is the input into the converter, which generates the result via successive approximation. The analog reference voltages are software selectable to either the device's analog positive and negative supply voltages (AV_{DD}/ AV_{SS}), the voltage level on the $V_{REF}+$ and $V_{REF}-$ pins, or internal voltage references if enabled (VRH, VRL).

The A/D converter can be triggered by setting the GO/$\overline{\text{DONE}}$ bit, or by the special event compare mode of the ECCP1 module. When conversion is complete, the GO/$\overline{\text{DONE}}$ bit returns to '0', the ADIF bit in the PIR1 register is set, and an A/D interrupt will occur, if enabled.

The A/D converter has a unique feature of being able to operate while the device is in SLEEP mode. To operate in sleep, the A/D conversion clock must be derived from the A/D's internal RC oscillator.

The A/D module has four registers. These registers are:

- A/D Result Register Low ADRESL
- A/D Result Register High ADRESH
- A/D Control Register 0 (ADCON0)
- A/D Control Register 1 (ADCON1)

A device reset forces all registers to their reset state. This forces the A/D module to be turned off and any conversion is aborted.

11.1 Control Registers

The ADCON0 register, shown in Register 11-1, controls the operation of the A/D module. The ADCON1 register, shown in Register 11-2, configures the functions of the port pins, the voltage reference configuration and the result format. The port pins can be configured as analog inputs or as digital I/O.

The combination of the ADRESH and ADRESL registers contain the result of the A/D conversion. The register pair is referred to as the ADRES register. When the A/D conversion is complete, the result is loaded into ADRES, the GO/$\overline{\text{DONE}}$ bit (ADCON0<2>) is cleared, and the A/D interrupt flag ADIF is set. The block diagram of the A/D module is shown in Figure 11-3.

REGISTER 11-1: A/D CONTROL REGISTER 0 (ADCON0: 1Fh).

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{\text{DONE}}$	CHS3	ADON
bit7							bit 0

R =	Readable bit
W =	Writable bit
- n =	Value at POR reset

bit 7-6: **ADCS<1:0>:** A/D Conversion Clock Select bits
If internal VRL and/or VRH are not used for A/D reference (VCFG<2:0> = 000, 001, 011 or 101):
00 = FOSC/2
01 = FOSC/8
10 = FOSC/32
11 = FRC (clock derived from a dedicated RC oscillator = 1 MHz max)
If internal VRL and/or VRH are used for A/D reference (VCFG<2:0> = 010, 100, 110 or 111):
00 = FOSC/16
01 = FOSC/64
10 = FOSC/256
11 = FRC (clock derived from a dedicated RC oscillator = 125 kHz max)

bit 1,5-3: **CHS:<3:0>:** Analog Channel Select bits
0000 = channel 00 (AN0)
0001 = channel 01 (AN1)
0010 = channel 02 (AN2)
0011 = channel 03 (AN3)
0100 = channel 04 (AN4)
0101 = channel 05 (AN5)
0110 = reserved, do not select
0111 = reserved, do not select
1000 = reserved, do not select
1001 = reserved, do not select
1010 = reserved, do not select
1011 = reserved, do not select
1100 = reserved, do not select
1101 = reserved, do not select
1110 = reserved, do not select
1111 = reserved, do not select

bit 2: **GO/$\overline{\text{DONE}}$:** A/D Conversion Status bit
1 = A/D conversion cycle in progress. Setting this bit starts an A/D conversion cycle.
This bit is automatically cleared by hardware when the A/D conversion has completed.
0 = A/D conversion completed/not in progress

bit 0: **ADON:** A/D On bit
1 = A/D converter module is operating
0 = A/D converter is shutoff and consumes no operating current

REGISTER 11-2: A/D CONTROL REGISTER 1 (ADCON1: 9Fh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	VCFG2	VCFG1	VCFG0	Reserved			
bit7							bit 0

R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

bit 7: **ADFM:** A/D Result Format Select bit
1 = Right justified
0 = Left justified

bit 6-4: **VCFG<2:0>:** Voltage reference configuration bits

	A/D VREF+	A/D VREF-
000	AVDD	AVSS
001	External VREF+	External VREF-
010	Internal VRH	Internal VRL
011	External VREF+	AVSS
100	Internal VRH	AVSS
101	AVDD	External VREF-
110	AVDD	Internal VRL
111	Internal VRL	AVSS

bit 3-0: **Reserved:** Do not use.

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

The A/D conversion results can be left justified (ADFM bit cleared), or right justified (ADFM bit set). Figure 11-1 through Figure 11-2 show the A/D result data format of the PIC16C717/770/771.

FIGURE 11-1: PIC16C770/771 12-BIT A/D RESULT FORMATS

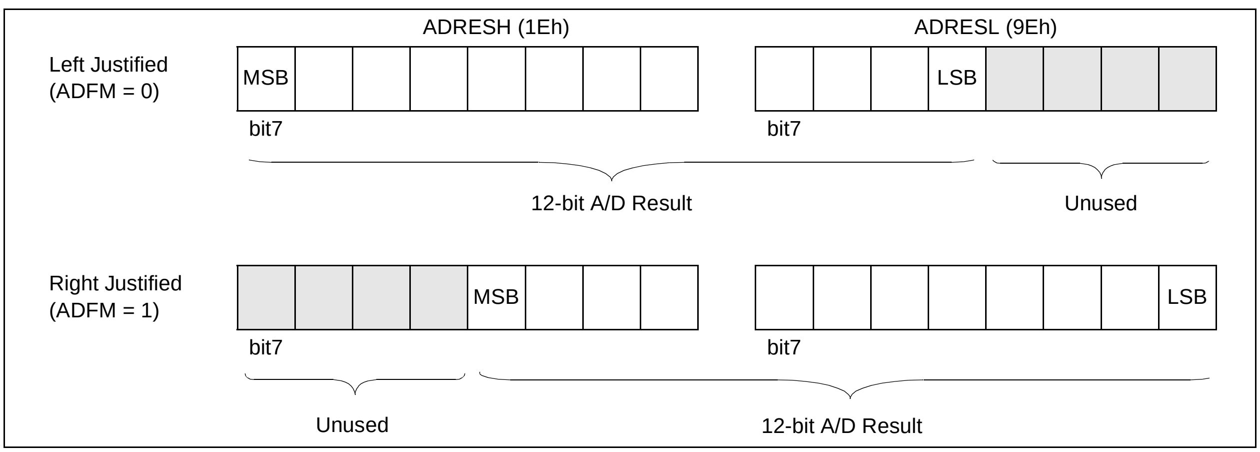
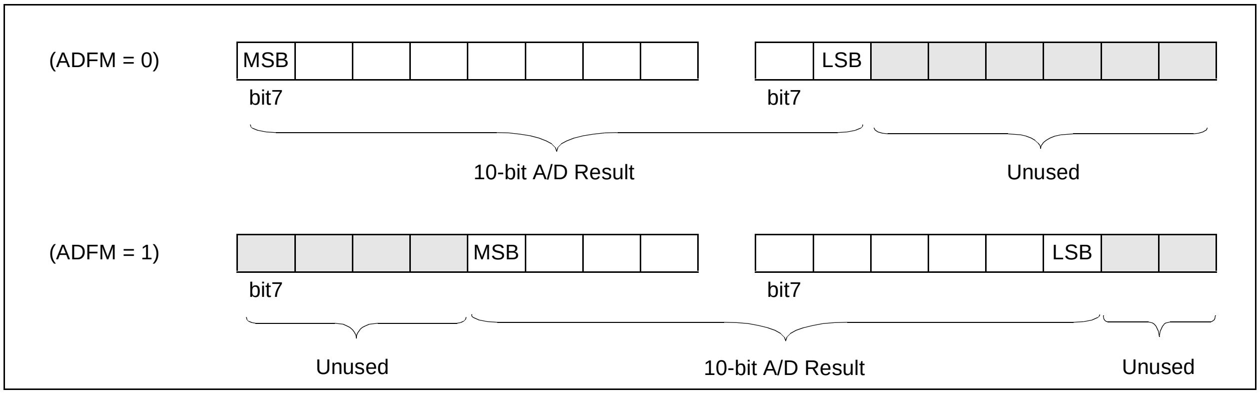


FIGURE 11-2: PIC16C717 10-BIT A/D RESULT FORMAT



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 and ANSEL bits selected as an input. To determine acquisition time, see Section 11.6. After this acquisition time has elapsed, the A/D conversion can be started. The following steps should be followed for doing an A/D conversion:

11.2 Configuring the A/D Module

11.2.1 CONFIGURING ANALOG PORT PINS

The ANSEL and TRIS registers control the operation of the A/D port pins. The port pins that are desired as analog inputs must have their corresponding TRIS bit set (input). If the TRIS bit is cleared (output), the digital output level (V_{OH} or V_{OL}) will be converted. The proper ANSEL bits must be set (analog input) to disable the digital input buffer.

The A/D operation is independent of the state of the TRIS bits and the ANSEL bits.

> **Note 1:** When reading the PORTA or PORTB register, all pins configured as analog input channels will read as '0'.
>
> **2:** Analog levels on any pin that is defined as a digital input, including the ANx pins, may cause the input buffer to consume current that is out of the devices specification.

11.2.2 CONFIGURING THE REFERENCE VOLTAGES

The VCFG bits in the ADCON1 register configure the A/D module reference inputs. The reference high input can come from an internal reference (VRH) or (VRL), an external reference (V_{REF+}), or AV_{DD}. The low reference input can come from an internal reference (VRL), an external reference (V_{REF-}), or AV_{SS}. If an external reference is chosen for the reference high or reference low inputs, the port pin that multiplexes the incoming external references is configured as an analog input, regardless of the values contained in the A/D port configuration bits (PCFG<3:0>).

After the A/D module has been configured as desired and the analog input channels have their corresponding TRIS bits selected for port inputs, the selected channel must be acquired before conversion is started. The A/D conversion cycle can be initiated by setting the GO/$\overline{\text{DONE}}$ bit. The A/D conversion begins and lasts for 13TAD. The following steps should be followed for performing an A/D conversion:

1. Configure port pins:
   - Configure analog input mode (ANSEL)
   - Configure pin as input (TRISA or TRISB)
2. Configure the A/D module
   - Configure A/D Result Format / voltage reference (ADCON1)
   - Select A/D input channel (ADCON0)
   - Select A/D conversion clock (ADCON0)
   - Turn on A/D module (ADCON0)
3. Configure A/D interrupt (if required)
   - Clear ADIF bit
   - Set ADIE bit
   - Set PEIE bit
   - Set GIE bit
4. Wait the required acquisition time (3TAD)
5. Start conversion
   - Set GO/$\overline{\text{DONE}}$ bit (ADCON0)
6. Wait 13TAD until A/D conversion is complete, by either:
   - Polling for the GO/$\overline{\text{DONE}}$ bit to be cleared

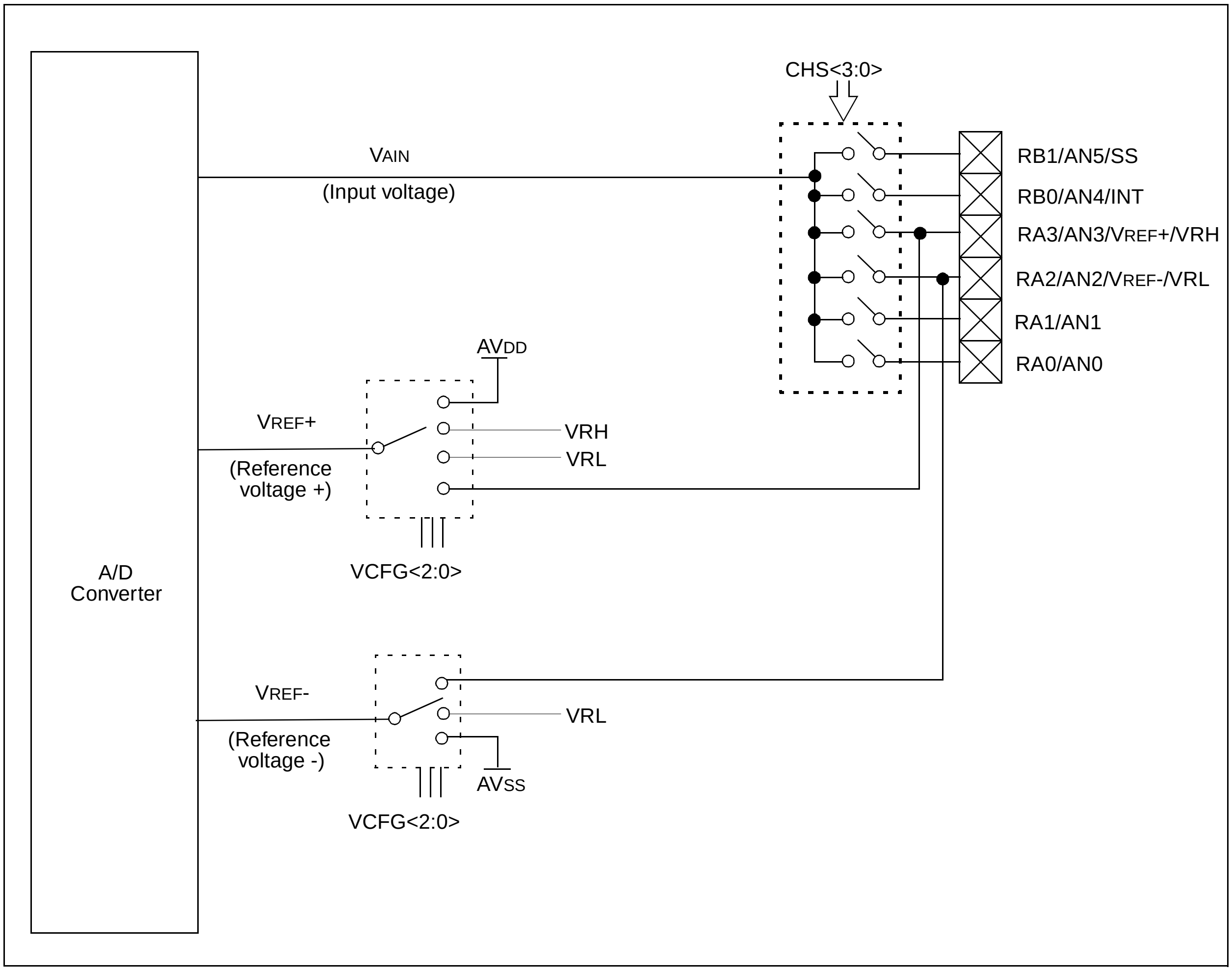
   OR

   - Waiting for the A/D interrupt
7. Read A/D Result registers (ADRESH and ADRESL), clear ADIF if required.
8. For next conversion, go to step 1, step 2 or step 3 as required.

Clearing the GO/$\overline{\text{DONE}}$ bit during a conversion will abort the current conversion. The ADRESH and ADRESL registers will be updated with the partially completed A/D conversion value. That is, the ADRESH and ADRESL registers will contain the value of the current incomplete conversion.

Note:	Do not set the ADON bit and the GO/$\overline{\text{DONE}}$ bit in the same instruction. Doing so will cause the GO/$\overline{\text{DONE}}$ bit to be automatically cleared.

FIGURE 11-3: A/D BLOCK DIAGRAM



11.3 Selecting the A/D Conversion Clock

The A/D conversion cycle requires 13TAD: 1 TAD for settling time, and 12 TAD for conversion. The source of the A/D conversion clock is software selected. If neither the internal VRH nor VRL are used for the A/D converter, the four possible options for TAD are:

- 2 TOSC
- 8 TOSC
- 32 TOSC
- A/D RC oscillator

If the VRH or VRL are used for the A/D converter reference, then the TAD requirement is automatically increased by a factor of 8.

For correct A/D conversions, the A/D conversion clock (TAD) must be selected to ensure a minimum TAD time of 1.6 µs. Table 11-1 shows the resultant TAD times derived from the device operating frequencies and the A/D clock source selected.

The ADIF bit is set on the rising edge of the 14th TAD. The GO/DONE bit is cleared on the falling edge of the 14th TAD.

TABLE 11-1: TAD vs. DEVICE OPERATING FREQUENCIES

A/D Reference Source	A/D Clock Source (TAD)		Device Frequency			
	Operation	ADCS<1:0>	20 MHz	5 MHz	4 MHz	1.25 MHz
External VREF or Analog Supply	2 TOSC	00	100 ns(2)	400 ns(2)	500 ns(2)	1.6 µs
	8 TOSC	01	800 ns(2)	1.6 µs	2.0 µs	6.4 µs
	32 TOSC	10	1.6 µs	6.4 µs	8.0 µs(3)	24 µs(3)
	A/D RC	11	2 - 6 µs(1,4)	2 - 6 µs(1,4)	2 - 6 µs(1,4)	2 - 6 µs(1,4)
Internal VRH or VRL	16 TOSC	00	800 ns(2)	3.2 µs(2)	4 µs(2)	12.8 µs
	64 TOSC	01	6.4 µs(2)	12.8 µs	16 µs	51.2 µs
	256 TOSC	10	12.8 µs	51.2 µs	64 µs(3)	192 µs(3)
	A/D RC	11	16 - 48 µs(4,5)	16 - 48 µs(4,5)	16 - 48 µs(4,5)	16 - 48 µs(4,5)

Legend: Shaded cells are outside of recommended range.

Note 1: The A/D RC source has a typical TAD time of 4 µs for VDD > 3.0V.

2: These values violate the minimum required TAD time.

3: For faster conversion times, the selection of another clock source is recommended.

4: When the device frequency is greater than 1 MHz, the A/D RC clock source is only recommended if the conversion will be performed during sleep.

5: The resource has a typical TAD time of 32 µs for VDD > 3.0V.

11.4 A/D Conversions

Example 11-1 shows an example that performs an A/D conversion. The port pins are configured as analog inputs. The analog reference V_{REF}+ is the device AV_{DD} and the analog reference V_{REF}- is the device AV_{SS}. The A/D interrupt is enabled and the A/D conversion clock is T_{RC}. The conversion is performed on the AN0 channel.

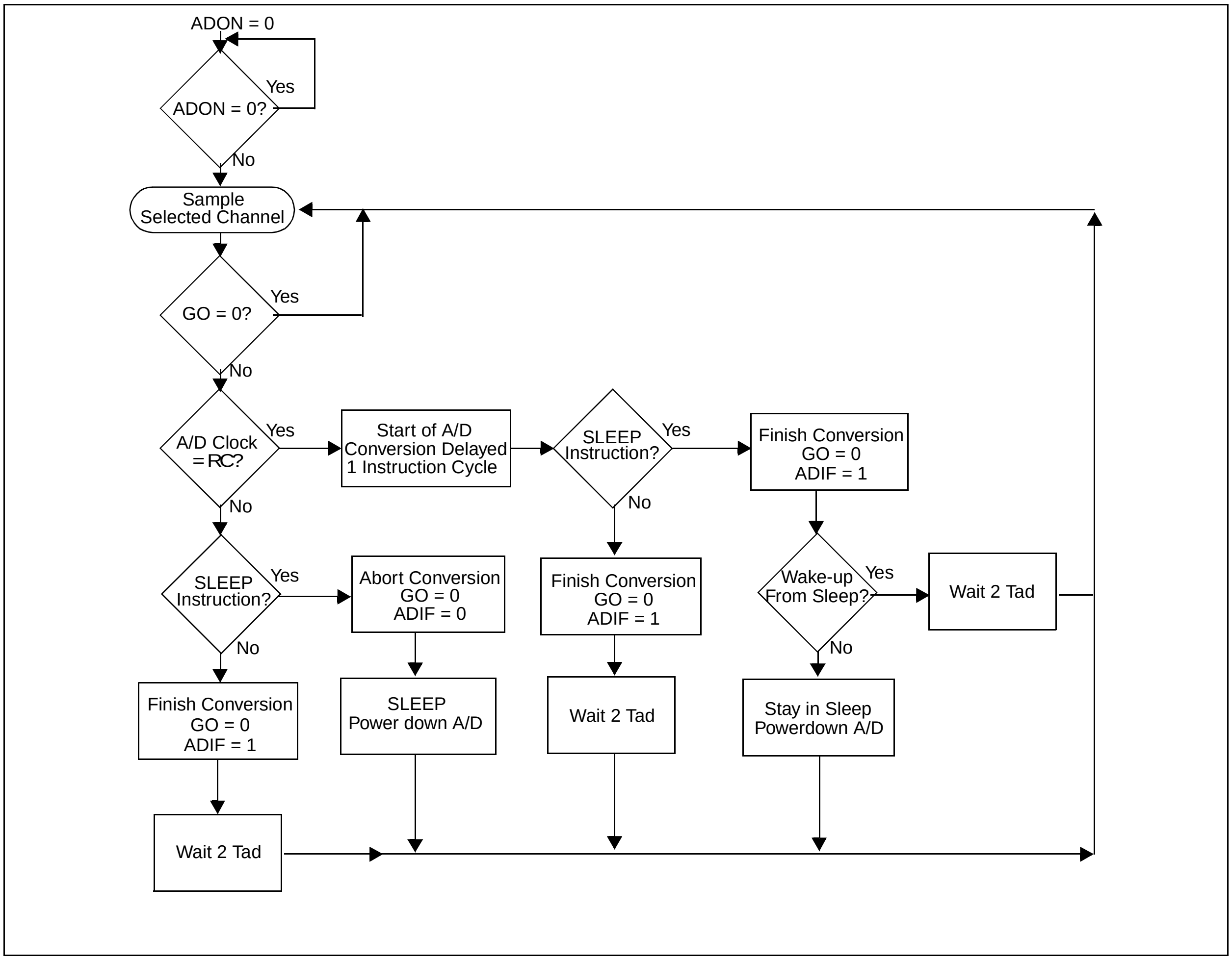
EXAMPLE 11-1: PERFORMING AN A/D CONVERSION

```
    BCF     PIR1, ADIF      ;Clear A/D Int Flag
    BSF     STATUS, RP0     ;Select Bank 1
    CLRF    ADCON1          ;Configure A/D Voltage Reference
    MOVLW   0x01
    MOVWF   ANSEL           ;disable AN0 digital input buffer
    MOVWF   TRISA           ;RA0 is input mode
    BSF     PIE1, ADIE      ;Enable A/D interrupt
    BCF     STATUS, RP0     ;Select Bank 0
    MOVLW   0xC1            ;RC clock, A/D is on,
                            ;Ch 0 is selected
    MOVWF   ADCON0          ;
    BSF     INTCON, PEIE    ;Enable Peripheral
    BSF     INTCON, GIE     ;Enable All Interrupts
;
; Ensure that the required sampling time for the
; selected input channel has lapsed. Then the
; conversion may be started.
    BSF     ADCON0, GO      ;Start A/D Conversion
            :               ;The ADIF bit will be
                            ;set and the GO/DONE bit
            :               ;cleared upon completion-
                            ;of the A/D conversion.
; Wait for A/D completion and read ADRESH:ADRESL for result.
```

11.5 A/D Converter Module Operation

Figure 11-4 shows the flowchart of the A/D converter module.

FIGURE 11-4: FLOWCHART OF A/D OPERATION



 Advanced Information

11.6 A/D Sample Requirements

11.6.1 RECOMMENDED SOURCE IMPEDANCE

The maximum recommended impedance for analog sources is 2.5 kΩ. This value is calculated based on the maximum leakage current of the input pin. The leakage current is 100 nA max., and the analog input voltage cannot be varied by more than 1/4 LSb or 250 μV due to leakage. This places a requirement on the input impedance of 250 μV/100 nA = 2.5 kΩ.

11.6.2 SAMPLING TIME CALCULATION

For the A/D converter to meet its specified accuracy, the charge holding capacitor (C_{HOLD}) must be allowed to fully charge to the input channel voltage level. The analog input model is shown in Figure 11-5. The source impedance (R_S) and the internal sampling switch (R_{SS}) impedance directly affect the time required to charge the capacitor C_{HOLD}. The sampling switch (R_{SS}) impedance varies over the device voltage (V_{DD}), see Figure 11-5. **The maximum recommended impedance for analog sources is 2.5 kΩ** . After the analog input channel is selected (changed) this sampling must be done before the conversion can be started.

To calculate the minimum sampling time, Equation 11-2 may be used. This equation assumes that 1/4 LSb error is used (16384 steps for the A/D). The 1/4 LSb error is the maximum error allowed for the A/D to meet its specified resolution.

The C_{HOLD} is assumed to be 25 pF for the 12-bit A/D.

EXAMPLE 11-2: A/D SAMPLING TIME EQUATION

$$V_{HOLD} = (V_{REF} - V_{REF}/16384) = (V_{REF}) \bullet (1 - e^{(-T_C/C(R_{IC} + R_{SS} + R_S))}) \quad V_{REF}(1 - 1/16384) = V_{REF} \bullet (1 - e^{(-T_C/C(R_{IC} + R_{SS} + R_S))})$$

$$T_C = -C_{HOLD}(1k\Omega + R_{SS} + R_S) \ln(1/16384)$$

Figure 11-3 shows the calculation of the minimum time required to charge C_{HOLD}. This calculation is based on the following system assumptions:

C_{HOLD} = 25 pF

R_S = 2.5 kΩ

1/4 LSb error

V_{DD} = 5V → R_{SS} = 10 kΩ (worst case)

Temp (system Max.) = 50°C

Note 1: The reference voltage (V_{REF}) has no effect on the equation, since it cancels itself out.

2: The charge holding capacitor (C_{HOLD}) is not discharged after each conversion.

3: The maximum recommended impedance for analog sources is 2.5 kΩ . This is required to meet the pin leakage specification.

4: After a conversion has completed, you must wait 2 T_{AD} time before sampling can begin again. During this time, the holding capacitor is not connected to the selected A/D input channel.

EXAMPLE 11-3: CALCULATING THE MINIMUM REQUIRED SAMPLE TIME

T_{ACQ} = Amplifier Settling Time
+ Holding Capacitor Charging Time
+Temperature offset †

T_{ACQ} = 5 µs
+ T_C
+ [(Temp - 25°C)(0.05 µs/°C)] †

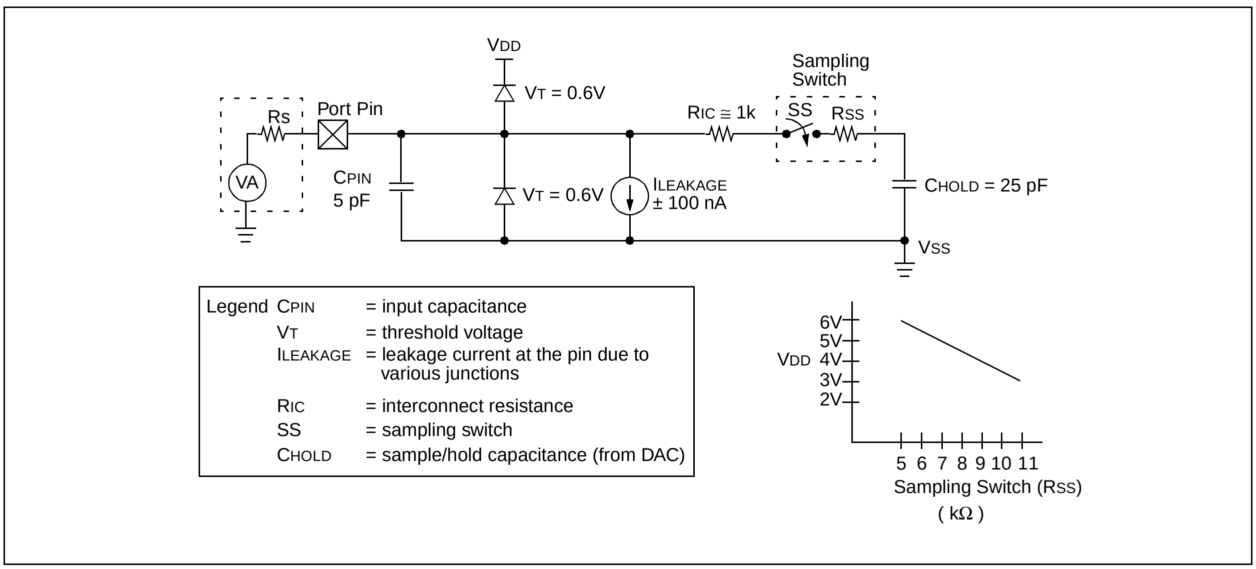
T_C = Holding Capacitor Charging Time
T_C = (C_{HOLD}) (R_{IC} + R_{SS} + R_S) In (1/16384)
T_C = -25 pF (1 kΩ +10 kΩ + 2.5 kΩ) In (1/16384)
T_C = -25 pF (13.5 kΩ) In (1/16384)
T_C = -0.338 (-9.704)µs
T_C = 3.3µs

T_{ACQ} = 5 µs
+ 3.3 µs
+ [(50°C - 25°C)(0.05 µs / °C)]

T_{ACQ} = 8.3 µs + 1.25 µs
T_{ACQ} = 9.55 µs

† The temperature coefficient is only required for temperatures > 25°C.

FIGURE 11-5: ANALOG INPUT MODEL



11.7 Use of the ECCP1 Trigger

An A/D conversion can be started by the “special event trigger” of the CCP module. This requires that the CCP1M<3:0> bits be programmed as `1011b` and that the A/D module is enabled (ADON is set). When the trigger occurs, the GO/$\overline{DONE}$ bit will be set on Q2 to start the A/D conversion and the Timer1 counter will be reset to zero. Timer1 is reset to automatically repeat the A/D conversion cycle, with minimal software overhead (moving the ADRESH and ADRESL to the desired location). The appropriate analog input channel must be selected before the “special event trigger” sets the GO/$\overline{DONE}$ bit (starts a conversion cycle).

If the A/D module is not enabled (ADON is cleared), then the “special event trigger” will be ignored by the A/D module, but will still reset the Timer1 counter.

11.8 Effects of a RESET

A device reset forces all registers to their reset state. This forces the A/D module to be turned off, and any conversion is aborted. The value that is in the ADRESH and ADRESL registers are not modified. The ADRESH and ADRESL registers will contain unknown data after a Power-on Reset.

11.9 Faster Conversion - Lower Resolution Trade-off

Not all applications require a result with 12-bits of resolution, but may instead require a faster conversion time. The A/D module allows users to make the trade-off of conversion speed to resolution. Regardless of the resolution required, the acquisition time is the same. To speed up the conversion, the A/D module may be halted by clearing the GO/$\overline{DONE}$ bit after the desired number of bits in the result have been converted. Once the GO/$\overline{DONE}$ bit has been cleared, all of the remaining A/D result bits are ‘0’. The equation to determine the time before the GO/$\overline{DONE}$ bit can be switched is as follows:

Conversion time = $(N+1)T_{AD}$

Where: N = number of bits of resolution required, and $1T_{AD}$ is the amplifier settling time.

Since T_{AD} is based from the device oscillator, the user must use some method (a timer, software loop, etc.) to determine when the A/D GO/$\overline{DONE}$ bit may be cleared. Table 11-4 shows a comparison of time required for a conversion with 4-bits of resolution, versus the normal 12-bit resolution conversion. The example is for devices operating at 20 MHz. The A/D clock is programmed for 32 T_{OSC}.

EXAMPLE 11-4: 4-BIT vs. 12-BIT CONVERSION TIME EXAMPLE

4 Bit Example:

Conversion Time = $(N + 1)\ T_{AD}$

= $(4 + 1)\ T_{AD}$

= $(5)(1.6\ \mu S)$

= $8\ \mu S$

12 Bit Example:

Conversion Time = $(N + 1)\ T_{AD}$

= $(12 + 1)\ T_{AD}$

= $(13)(1.6\ \mu S)$

= $20.8\ \mu S$

11.10 A/D Operation During Sleep

The A/D module can operate during SLEEP mode. This requires that the A/D clock source be configured for RC (ADCS<1:0> = 11b). With the RC clock source selected, when the GO/$\overline{DONE}$ bit is set the A/D module waits one instruction cycle before starting the conversion cycle. This allows the SLEEP instruction to be executed, which eliminates all digital switching noise during the sample and conversion. When the conversion cycle is completed the GO/$\overline{DONE}$ bit is cleared, and the result loaded into the ADRESH and ADRESL registers. If the A/D interrupt is enabled, the device will wake-up from SLEEP. If the A/D interrupt is not enabled, the A/D module will then be turned off, although the ADON bit will remain set.

When the A/D clock source is another clock option (not RC), a SLEEP instruction causes the present conversion to be aborted and the A/D module is turned off, though the ADON bit will remain set.

Turning off the A/D places the A/D module in its lowest current consumption state.

Note: For the A/D module to operate in SLEEP, the A/D clock source must be configured to RC (ADCS<1:0> = 11b).

11.11 Connection Considerations

Since the analog inputs employ ESD protection, they have diodes to V_{DD} and V_{SS}. This requires that the analog input must be between V_{DD} and V_{SS}. If the input voltage exceeds this range by greater than 0.3V (either direction), one of the diodes becomes forward biased and it may damage the device if the input current specification is exceeded.

An external RC filter is sometimes added for anti-aliasing of the input signal. The R component should be selected to ensure that the total source impedance is kept under the 2.5 kΩ recommended specification. Any external components connected (via hi-impedance) to an analog input pin (capacitor, zener diode, etc.) should have very little leakage current at the pin.

TABLE 11-2: SUMMARY OF A/D REGISTERS

Address	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
0Bh,8Bh, 10Bh,18Bh	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
0Ch	PIR1	—	ADIF	—	—	SSPIF	CCP1IF	TMR2IF	TMR1IF	-0-- 0000	-0-- 0000
8Ch	PIE1	—	ADIE	—	—	SSPIE	CCP1IE	TMR2IE	TMR1IE	-0-- 0000	-0-- 0000
1Eh	ADRESH	A/D High Byte Result Register								xxxx xxxx	uuuu uuuu
9Eh	ADRESL	A/D Low Byte Result Register								xxxx xxxx	uuuu uuuu
9Bh	REFCON	VRHEN	VRLEN	VRHOEN	VRLOEN	—	—	—	—	0000 ----	0000 ----
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{DONE}$	CHS3	ADON	0000 0000	0000 0000
9Fh	ADCON1	ADFM	VCFG2	VCFG1	VCFG0	—	—	—	—	0000 ----	0000 ----
05h	PORTA	PORTA Data Latch when written: PORTA pins when read								000x 0000	000u 0000
06h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xx00	uuuu uu00
85h	TRISA	PORTA Data Direction Register								1111 1111	1111 1111
86h	TRISB	PORTB Data Direction Register								1111 1111	1111 1111
9Dh	ANSEL			ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111

Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used for A/D conversion.

12.0 SPECIAL FEATURES OF THE CPU

These devices have a host of features intended to maximize system reliability, minimize cost through elimination of external components, provide power saving operating modes and offer code protection. These are:

- Oscillator Selection
- Reset
  - Power-on Reset (POR)
  - Power-up Timer (PWRT)
  - Oscillator Start-up Timer (OST)
  - Brown-out Reset (BOR)
- Interrupts
- Watchdog Timer (WDT)
- Low-voltage detection
- SLEEP
- Code protection
- ID locations
- In-circuit serial programming (ICSP)

These devices have a Watchdog Timer, which can be shut off only through configuration bits. It runs off its own RC oscillator for added reliability. There are two timers that offer necessary delays on power-up. One is the Oscillator Start-up Timer (OST), intended to keep the chip in reset until the crystal oscillator is stable. The other is the Power-up Timer (PWRT), which provides a fixed delay of 72 ms (nominal) on power-up type resets only (POR, BOR), designed to keep the part in reset while the power supply stabilizes. With these two timers on-chip, most applications need no external reset circuitry.

SLEEP mode is designed to offer a very low current power-down mode. The user can wake-up from SLEEP through external reset, Watchdog Timer Wake-up, or through an interrupt. Several oscillator options are also made available to allow the part to fit the application. The INTRC and ER oscillator options save system cost while the LP crystal option saves power. A set of configuration bits are used to select various options.

Additional information on special features is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

12.1 Configuration Bits

The configuration bits can be programmed (read as '0') or left unprogrammed (read as '1') to select various device configurations. These bits are mapped in program memory location 2007h.

The user will note that address 2007h is beyond the user program memory space. In fact, it belongs to the special test/configuration memory space (2000h - 3FFFh), which can be accessed only during programming.

Some of the core features provided may not be necessary to each application that a device may be used for. The configuration word bits allow these features to be configured/enabled/disabled as necessary. These features include code protection, brown-out reset and its trippoint, the power-up timer, the watchdog timer and the devices oscillator mode. As can be seen in Figure 12-1, some additional configuration word bits have been provided for brown-out reset trippoint selection.

FIGURE 12-1: CONFIGURATION WORD FOR 16C717/770/771 DEVICE

$\overline{CP}$	$\overline{CP}$	BORV1	BORV0	$\overline{CP}$	$\overline{CP}$	—	BODEN	MCLRE	$\overline{PWRTE}$	WDTE	FOSC2	FOSC1	FOSC0
bit13	12	11	10	9	8	7	6	5	4	3	2	1	bit0

Register: CONFIG
Address 2007h

bit 13,12: **$\overline{CP}$: Program Memory Code Protection**
bit 9,8: `1` = Code protection off
`0` = All program memory is protected(2)

bit 11-10: **BORV<1:0>:** Brown-out Reset Voltage bits
`00` = V_{BOR} set to 4.5V
`01` = V_{BOR} set to 4.2V
`10` = V_{BOR} set to 2.7V
`11` = V_{BOR} set to 2.5V

bit 7: **Unimplemented:** Read as '1'

bit 6: **BODEN:** Brown-out Detect Reset Enable bit (1)
`1` = Brown-out Detect Reset enabled
`0` = Brown-out Detect Reset disabled

bit 5: **MCLRE**: RA5/$\overline{MCLR}$ pin function select
`1` = RA5/$\overline{MCLR}$ pin function is $\overline{MCLR}$
`0` = RA5/$\overline{MCLR}$ pin function is digital input, $\overline{MCLR}$ internally tied to V_{DD}

bit 4: **$\overline{PWRTE}$:** Power-up Timer Enable bit (1)
`1` = PWRT disabled
`0` = PWRT enabled

bit 3: **WDTE:** Watchdog Timer Enable bit
`1` = WDT enabled
`0` = WDT disabled

bit 2-0: **FOSC<2:0>:** Oscillator Selection bits
`000` = LP oscillator: Ceramic resonator on RA6/OSC2/CLKOUT and RA7/OSC1/CLKIN
`001` = XT oscillator: Crystal on RA6/OSC2/CLKOUT and RA7/OSC1/CLKIN
`010` = HS oscillator: High frequency crystal on RA6/OSC2/CLKOUT and RA7/OSC1/CLKIN
`011` = EC: I/O function on RA6/OSC2/CLKOUT pin, CLKIN function on RA7/OSC1/CLKIN
`100` = INTRC oscillator: I/O function on RA6/OSC2/CLKOUT pin, I/O function on RA7/OSC1/CLKIN
`101` = INTRC oscillator: CLKOUT function on RA6/OSC2/CLKOUT pin, I/O function on RA7/OSC1/CLKIN
`110` = ER oscillator: I/O function on RA6/OSC2/CLKOUT pin, Resistor on RA7/OSC1/CLKIN
`111` = ER oscillator: CLKOUT function on RA6/OSC2/CLKOUT pin, Resistor on RA7/OSC1/CLKIN

Note 1: Enabling Brown-out Reset automatically enables the Power-up Timer (PWRT), regardless of the value of bit $\overline{PWRTE}$. Ensure the Power-up Timer is enabled anytime Brown-out Reset is enabled.
2: All of the CP bits must be given the same value to enable code protection.

12.2 Oscillator Configurations

12.2.1 OSCILLATOR TYPES

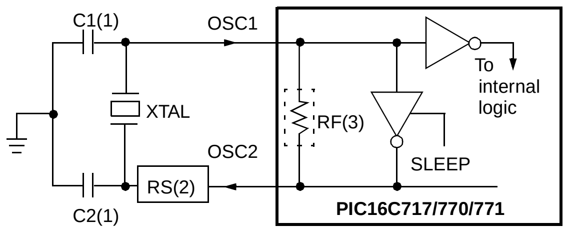
The PIC16C717/770/771 can be operated in four different oscillator modes. The user can program three configuration bits (FOSC<2:0>) to select one of these eight modes:

- LP Low Power Crystal
- XT Crystal/Resonator
- HS High Speed Crystal/Resonator
- ER External Resistor (with and without CLKOUT)
- INTRC Internal 4 MHz (with and without CLKOUT)
- EC External Clock

12.2.2 LP, XT AND HS MODES

In LP, XT or HS modes, a crystal or ceramic resonator is connected to the OSC1/CLKIN and OSC2/CLKOUT pins to establish oscillation (Figure 12-2). The PIC16C717/770/771 oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may give a frequency out of the crystal manufacturers specifications.

FIGURE 12-2: CRYSTAL/CERAMIC RESONATOR OPERATION (HS, XT OR LP OSC CONFIGURATION)



Note1: See Table 12-1 and Table 12-2 for recommended values of C1 and C2.
2: A series resistor (RS) may be required for AT strip cut crystals.
3: RF varies with the crystal chosen.

TABLE 12-1: CERAMIC RESONATORS

Ranges Tested:			
Mode	**Freq**	**OSC1**	**OSC2**
XT	455 kHz 2.0 MHz 4.0 MHz	68 - 100 pF 15 - 68 pF 15 - 68 pF	68 - 100 pF 15 - 68 pF 15 - 68 pF
HS	8.0 MHz 16.0 MHz	10 - 68 pF 10 - 22 pF	10 - 68 pF 10 - 22 pF
These values are for design guidance only. See notes at bottom of page.			
All resonators used did not have built-in capacitors.			

TABLE 12-2: CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR

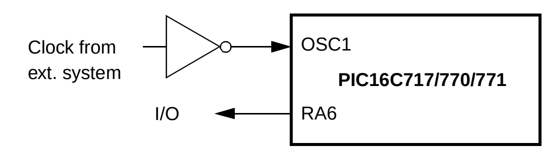
Osc Type	Crystal Freq	Cap. Range C1	Cap. Range C2
LP	32 kHz	33 pF	33 pF
	200 kHz	15 pF	15 pF
XT	200 kHz	47-68 pF	47-68 pF
	1 MHz	15 pF	15 pF
	4 MHz	15 pF	15 pF
HS	4 MHz	15 pF	15 pF
	8 MHz	15-33 pF	15-33 pF
	20 MHz	15-33 pF	15-33 pF
These values are for design guidance only. See notes at bottom of page.			

Note 1: Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components.
2: Higher capacitance increases the stability of oscillator but also increases the start-up time.

12.2.3 EC MODE

In applications where the clock source is external, the PIC16C717/770/771 should be programmed to select the EC (External Clock) mode. In this mode, the RA6/OSC2/CLKOUT pin is available as an I/O pin. See Figure 12-3 for illustration.

FIGURE 12-3: EXTERNAL CLOCK INPUT OPERATION (EC OSC CONFIGURATION)

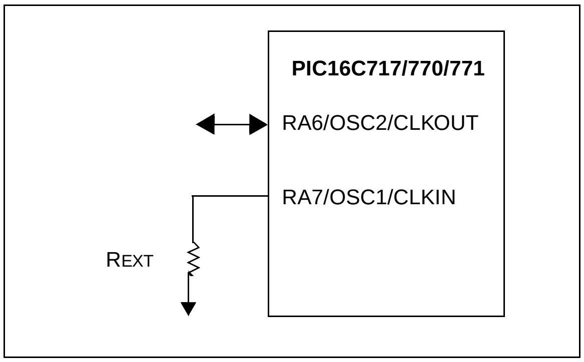


12.2.4 ER MODE

For timing insensitive applications, the ER (External Resistor) clock mode offers additional cost savings. Only one external component, a resistor connected to the OSC1 pin and VSS, is needed to set the operating frequency of the internal oscillator. The resistor draws a DC bias current which controls the oscillation frequency. In addition to the resistance value, the oscillator frequency will vary from unit to unit, and as a function of supply voltage and temperature. Since the controlling parameter is a DC current and not a capacitance, the particular package type and lead frame will not have a significant effect on the resultant frequency.

Figure 12-4 shows how the controlling resistor is connected to the PIC16C717/770/771. For Rext values below 38k ohms, the oscillator operation may become unstable, or stop completely. For very high Rext values (e.g. 1M), the oscillator becomes sensitive to noise, humidity and leakage. Thus, we recommend keeping Rext between 38k and 1M ohms.

FIGURE 12-4: EXTERNAL RESISTOR



The Electrical Specification section shows the relationship between the Rext resistance value and the operating frequency as well as frequency variations due to operating temperature for given Rext and VDD values.

The ER oscillator mode has two options that control the OSC2 pin. The first allows it to be used as a general purpose I/O port. The other configures the pin as CLKOUT. The ER oscillator does not run during reset.

12.2.5 INTRC MODE

The internal RC oscillator provides a fixed 4 MHz (nominal) system clock at VDD = 5V and 25°C, see “Electrical Specifications” section for information on variation over voltage and temperature. The INTRC oscillator does not run during reset.

12.2.6 CLKOUT

In the INTRC and ER modes, the PIC16C717/770/771 can be configured to provide a clock out signal by programming the configuration word. The oscillator frequency, divided by 4, can be used for test purposes or to synchronize other logic.

In the INTRC and ER modes, if the CLKOUT output is enabled, CLKOUT is held low during reset.

12.2.7 DUAL SPEED OPERATION FOR ER AND INTRC MODES

A software programmable dual speed oscillator is available in either ER or INTRC oscillator modes. This feature allows the applications to dynamically toggle the oscillator speed between normal and slow frequencies. The nominal slow frequency is 37KHz. In ER mode, the slow speed operation is fixed and does not vary with resistor size. Applications that require low current power savings, but cannot tolerate putting the part into sleep, may use this mode.

The OSCF bit in the PCON register is used to control dual speed mode. See the PCON Register, Register 2-8, for details.

When changing the INTRC or ER internal oscillator speed, there is a period of time when the processor is inactive. When the speed changes from fast to slow, the processor inactive period is in the range of 100 µS to 300 µS. For speed change from slow to fast, the processor is in active for 1.25 µS to 3.25 µS.

 Advanced Information

12.3 Reset

The PIC16C717/770/771 devices have several different resets. These resets are grouped into two classifications; power-up and non-power-up. The power-up type resets are the power-on and brown-out resets which assume the device VDD was below its normal operating range for the device's configuration. The non-power up type resets assume normal operating limits were maintained before/during and after the reset.

- Power-on Reset (POR)
- Programmable Brown-out Reset (PBOR)
- MCLR reset during normal operation
- MCLR reset during SLEEP
- WDT Reset (during normal operation)

Some registers are not affected in any reset condition. Their status is unknown on a power-up reset and unchanged in any other reset. Most other registers are placed into an initialized state upon reset, however they are not affected by a WDT reset during sleep, because this is considered a WDT Wakeup, which is viewed as the resumption of normal operation.

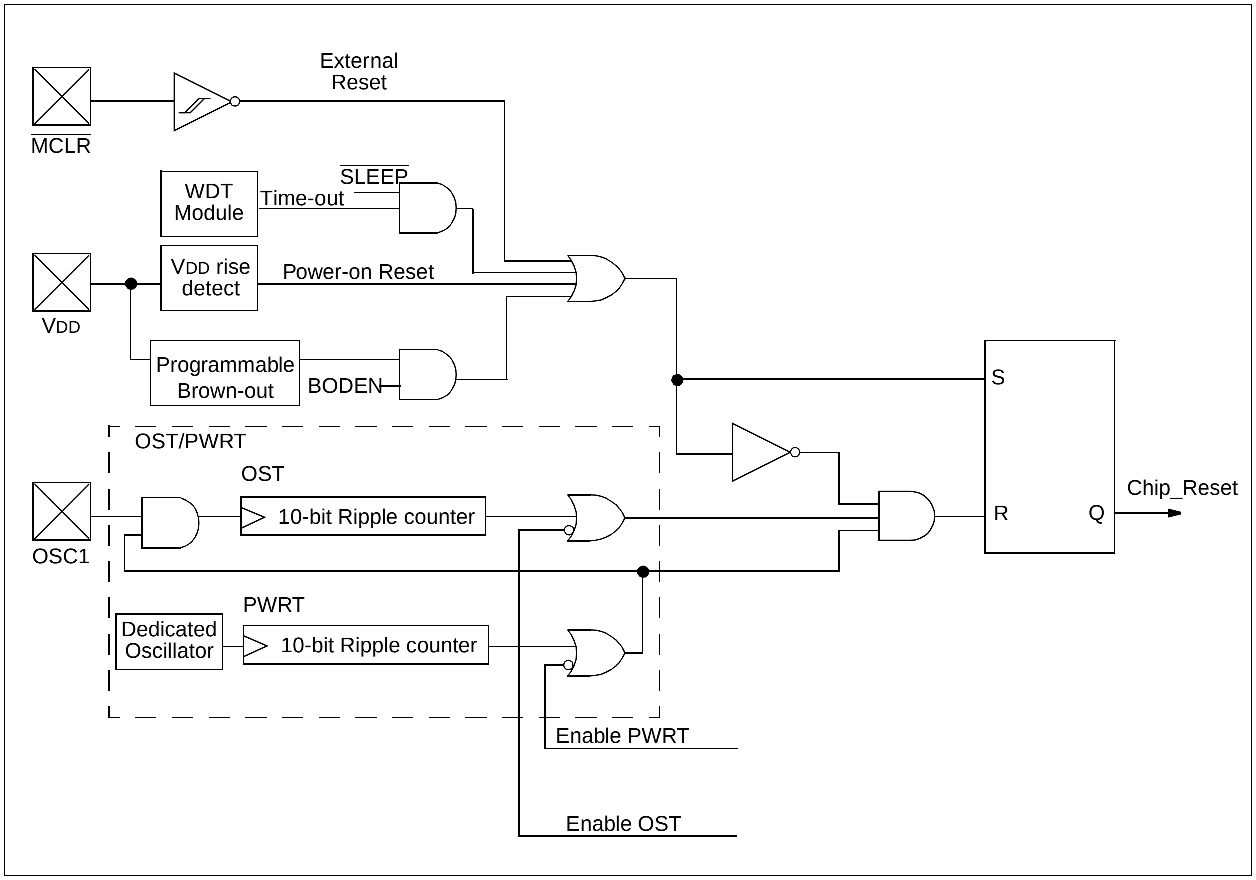
Several status bits have been provided to indicate which reset occurred (see Table 12-4). See Table 12-6 for a full description of reset states of all registers.

A simplified block diagram of the on-chip reset circuit is shown in Figure 12-5.

These devices have a MCLR noise filter in the MCLR reset path. The filter will detect and ignore small pulses.

It should be noted that a WDT Reset does not drive MCLR pin low.

FIGURE 12-5: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



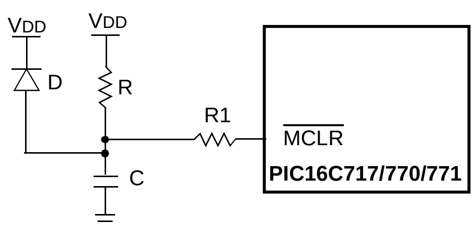
12.4 Power-On Reset (POR)

A Power-on Reset pulse is generated on-chip when VDD rise is detected (in the range of 1.5V - 2.1V). To take advantage of the POR, just enable the internal MCLR feature. This will eliminate external RC components usually needed to create a Power-on Reset. A maximum rise time for VDD is specified. See Electrical Specifications for details. For a slow rise time, see Figure 12-6.

Two delay timers, (PWRT on OST), have been provided which hold the device in reset after a POR (dependent upon device configuration) so that all operational parameters have been met prior to releasing the device to resume/begin normal operation.

When the device starts normal operation (exits the reset condition), device operating parameters (voltage, frequency, temperature,...) must be met to ensure operation. If these conditions are not met, the device must be held in reset until the operating conditions are met. Brown-out Reset may be used to meet the startup conditions, or if necessary an external POR circuit may be implemented to delay end of reset for as long as needed.

FIGURE 12-6: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW VDD RAMP)



Note 1: External Power-on Reset circuit is required only if VDD power-up slope is too slow. The diode D helps discharge the capacitor quickly when VDD powers down.

2: R < 40 kΩ is recommended to make sure that voltage drop across R does not violate the device's electrical specification.

3: R1 = 100Ω to 1 kΩ will limit any current flowing into MCLR from external capacitor C in the event of MCLR/VPP pin breakdown due to Electrostatic Discharge (ESD) or Electrical Overstress (EOS).

12.5 Power-up Timer (PWRT)

The Power-up Timer provides a fixed TPWRT time-out on power-up type resets only. For a POR, the PWRT is invoked when the POR pulse is generated. For a BOR, the PWRT is invoked when the device exits the reset condition (VDD rises above BOR trippoint). The Power-up Timer operates on an internal RC oscillator. The chip is kept in reset as long as the PWRT is active. The PWRT's time delay is designed to allow VDD to rise to an acceptable level. A configuration bit is provided to enable/disable the PWRT for the POR only. For a BOR the PWRT is always available regardless of the configuration bit setting.

The power-up time delay will vary from chip-to-chip due to VDD, temperature and process variation. See DC parameters for details.

12.6 Oscillator Start-up Timer (OST)

The Oscillator Start-up Timer (OST) provides 1024 oscillator cycle (from OSC1 input) delay after the PWRT delay is over. This ensures that the crystal oscillator or resonator has started and stabilized.

The OST time-out is invoked only for XT, LP and HS modes and only on a power-up type reset or a wake-up from SLEEP.

12.7 Programmable Brown-Out Reset (PBOR)

The Programmable Brown-out Reset module is used to generate a reset when the supply voltage falls below a specified trip voltage. The trip voltage is configurable to any one of four voltages provided by the BORV<1:0> configuration word bits.

Configuration bit, BODEN, can disable (if clear/programmed) or enable (if set) the Brown-out Reset circuitry. If VDD falls below the specified trippoint for longer than TBOR, (parameter #35), the brown-out situation will reset the chip. A reset may not occur if VDD falls below the trippoint for less than TBOR. The chip will remain in Brown-out Reset until VDD rises above VBOR. The Power-up Timer will be invoked at that point and will keep the chip in RESET an additional TPWRT. If VDD drops below VBOR while the Power-up Timer is running, the chip will go back into a Brown-out Reset and the Power-up Timer will be re-initialized. Once VDD rises above VBOR, the Power-up Timer will again begin a TPWRT time delay. Even though the PWRT is always enabled when brown-out is enabled, the PWRT configuration word bit should be cleared (enabled) when brown-out is enabled.

12.8 Time-out Sequence

On power-up, the time-out sequence is as follows: First PWRT time-out is invoked by the POR pulse. When the PWRT delay expires, the Oscillator Start-up Timer is activated. The total time-out will vary based on oscillator configuration and the status of the PWRT. For example, in RC mode with the PWRT disabled, there will be no time-out at all. Figure 12-7, Figure 12-8, Figure 12-9 and Figure 12-10 depict time-out sequences on power-up.

Since the time-outs occur from the POR pulse, if $\overline{MCLR}$ is kept low long enough, the time-outs will expire. Then bringing $\overline{MCLR}$ high will begin execution immediately (Figure 12-9). This is useful for testing purposes or to synchronize more than one PICmicro microcontroller operating in parallel.

Table 12-5 shows the reset conditions for some special function registers, while Table 12-6 shows the reset conditions for all the registers.

12.9 Power Control/Status Register (PCON)

The Power Control/Status Register, PCON, has two status bits that provide indication of which power-up type reset occurred.

Bit0 is Brown-out Reset Status bit, $\overline{BOR}$. Bit $\overline{BOR}$ is set on a Power-on Reset. It must then be set by the user and checked on subsequent resets to see if bit $\overline{BOR}$ cleared, indicating a BOR occurred. However, if the brown-out circuitry is disabled, the $\overline{BOR}$ bit is a "Don't Care" bit and is considered unknown upon a POR.

Bit1 is $\overline{POR}$ (Power-on Reset Status bit). It is cleared on a Power-on Reset and unaffected otherwise. The user must set this bit following a Power-on Reset.

TABLE 12-3: TIME-OUT IN VARIOUS SITUATIONS

Oscillator Configuration	Power-up		Brown-out	Wake-up from SLEEP
	$\overline{PWRTE}$ = 0	$\overline{PWRTE}$ = 1		
XT, HS, LP	TPWRT + 1024TOSC	1024TOSC	TPWRT + 1024TOSC	1024TOSC
EC, ER, INTRC	TPWRT	—	TPWRT	—

TABLE 12-4: STATUS BITS AND THEIR SIGNIFICANCE

$\overline{POR}$	$\overline{BOR}$	$\overline{TO}$	$\overline{PD}$	
0	1	1	1	Power-on Reset
0	x	0	x	Illegal, $\overline{TO}$ is set on $\overline{POR}$
0	x	x	0	Illegal, $\overline{PD}$ is set on $\overline{POR}$
1	0	1	1	Brown-out Reset
1	1	0	1	WDT Reset
1	1	0	0	WDT Wake-up
1	1	u	u	$\overline{MCLR}$ Reset during normal operation
1	1	1	0	$\overline{MCLR}$ Reset during SLEEP or interrupt wake-up from SLEEP

TABLE 12-5: RESET CONDITION FOR SPECIAL REGISTERS

Condition	Program Counter	STATUS Register	PCON Register
Power-on Reset	000h	0001 1xxx	---- 1-01
$\overline{MCLR}$ Reset during normal operation	000h	000u uuuu	---- 1-uu
$\overline{MCLR}$ Reset during SLEEP	000h	0001 0uuu	---- 1-uu
WDT Reset	000h	0000 1uuu	---- 1-uu
WDT Wake-up	PC + 1	uuu0 0uuu	---- u-uu
Brown-out Reset	000h	0001 1uuu	---- 1-u0
Interrupt wake-up from SLEEP, GIE = 0	PC + 1	uuu1 0uuu	---- u-uu
Interrupt wake-up from SLEEP, GIE = 1	0004h	uuu1 0uuu	---- u-uu

Legend: u = unchanged, x = unknown, - = unimplemented bit read as '0'.

TABLE 12-6: INITIALIZATION CONDITIONS FOR ALL REGISTERS

Register	Power-on Reset or Brown-out Reset	$\overline{MCLR}$ Reset or WDT Reset	Wake-up via WDT or Interrupt
W	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	0000 0000	uuuu uuuu	uuuu uuuu
TMR0	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	0000h	0000h	PC + 1(1)
STATUS	0001 1xxx	000q quuu(2)	uuuq quuu(2)
FSR	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA	xxxx 0000	uuuu 0000	uuuu uuuu
PORTB	xxxx xx00	uuuu uu00	uuuu uu00
PCLATH	---0 0000	---0 0000	---u uuuu
INTCON	0000 000x	0000 000u	uuuu uuqq
PIR1	-0-- 0000	-0-- 0000	-0-- uuuu
PIR2	0--- 0---	0--- 0---	q--- q---
TMR1L	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	--00 0000	--uu uuuu	--uu uuuu
TMR2	0000 0000	0000 0000	uuuu uuuu
T2CON	-000 0000	-000 0000	-uuu uuuu
SSPBUF	xxxx xxxx	uuuu uuuu	uuuu uuuu
SSPCON	0000 0000	0000 0000	uuuu uuuu
CCPR1L	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR1H	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP1CON	0000 0000	0000 0000	uuuu uuuu
ADRESH	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON0	0000 0000	0000 0000	uuuu uuuu
OPTION_REG	1111 1111	1111 1111	uuuu uuuu
TRISA	1111 1111	1111 1111	uuuu uuuu
TRISB	1111 1111	1111 1111	uuuu uuuu
PIE1	-0-- 0000	-0-- 0000	-u-- uuuu
PIE2	0--- 0---	0--- 0---	u--- u---
PCON	---- 1-qq	---- 1-uu	---- u-uu
PR2	1111 1111	1111 1111	1111 1111
SSPADD	0000 0000	0000 0000	uuuu uuuu
SSPSTAT	0000 0000	0000 0000	uuuu uuuu
WPUB	1111 1111	1111 1111	uuuu uuuu
IOCB	1111 0000	1111 0000	uuuu uuuu
P1DEL	0000 0000	0000 0000	uuuu uuuu
REFCON	0000 ----	0000 ----	uuuu ----

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition

Note 1: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

2: See Table 12-5 for reset value for specific condition.

TABLE 12-6: INITIALIZATION CONDITIONS FOR ALL REGISTERS (CONTINUED)

Register	Power-on Reset or Brown-out Reset	$\overline{MCLR}$ Reset or WDT Reset	Wake-up via WDT or Interrupt
LVDCON	--00 0101	--00 0101	--uu uuuu
ANSEL	1111 1111	1111 1111	uuuu uuuu
ADRESL	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON1	0000 000	0000 0000	uuuu uuuu
PMDATL	xxxx xxxx	uuuu uuuu	uuuu uuuu
PMADRL	xxxx xxxx	uuuu uuuu	uuuu uuuu
PMDATH	--xx xxxx	--uu uuuu	--uu uuuu
PMADRH	---- xxxx	---- uuuu	---- uuuu
PMCON1	1--- ---0	1--- ---0	1--- ---0

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition

Note 1: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

2: See Table 12-5 for reset value for specific condition.

FIGURE 12-7: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ TIED TO V_{DD})

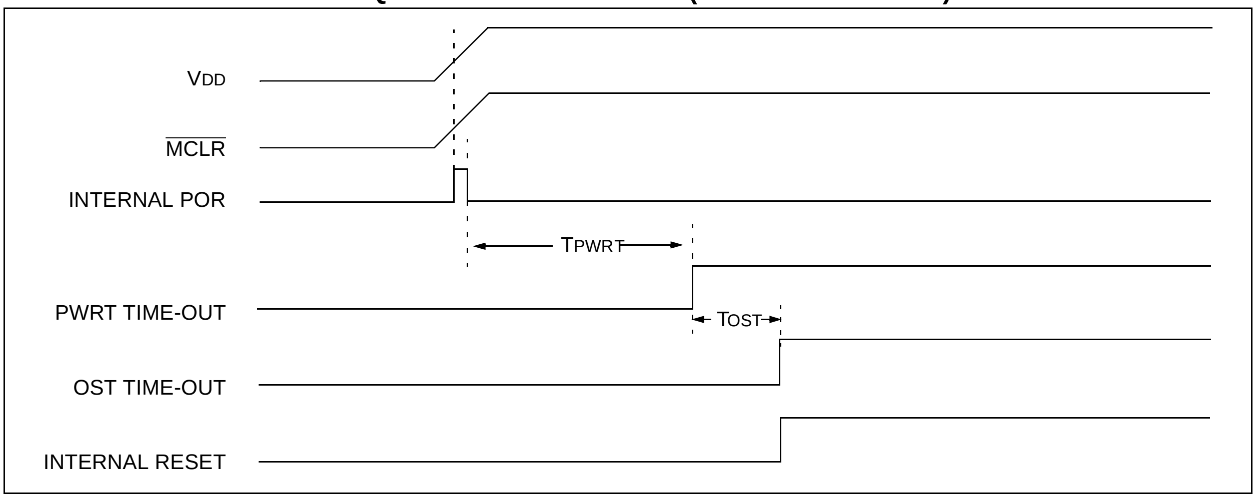


FIGURE 12-8: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ NOT TIED TO V_{DD}): CASE 1

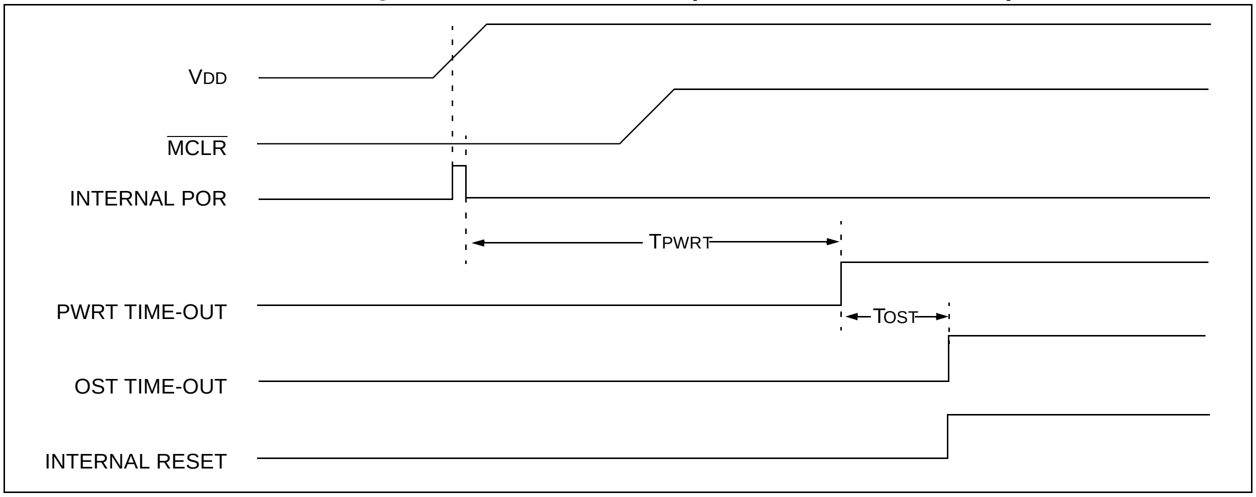


FIGURE 12-9: TIME-OUT SEQUENCE ON POWER-UP ($\overline{MCLR}$ NOT TIED TO V_{DD}): CASE 2

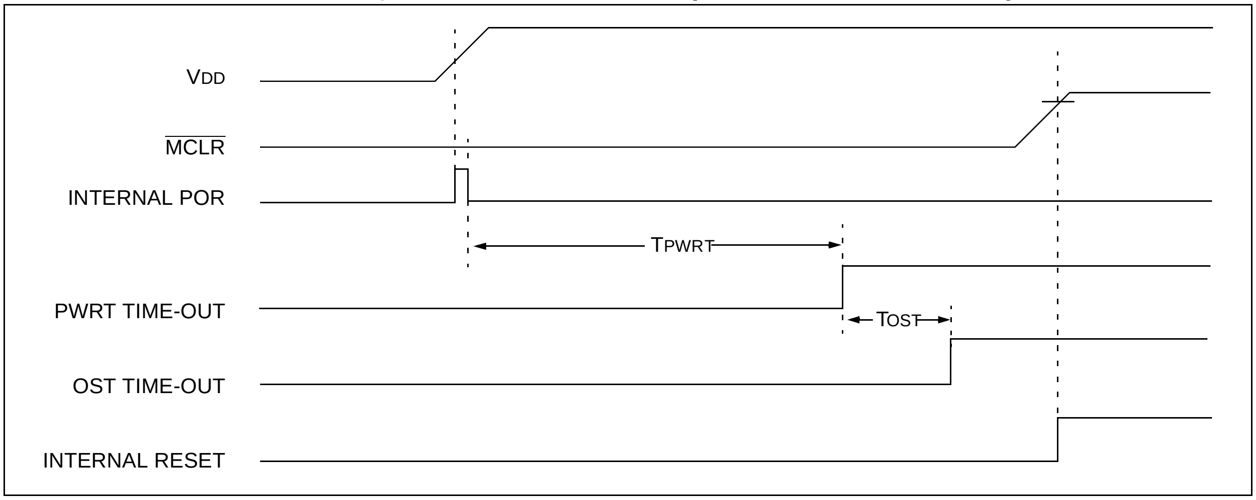
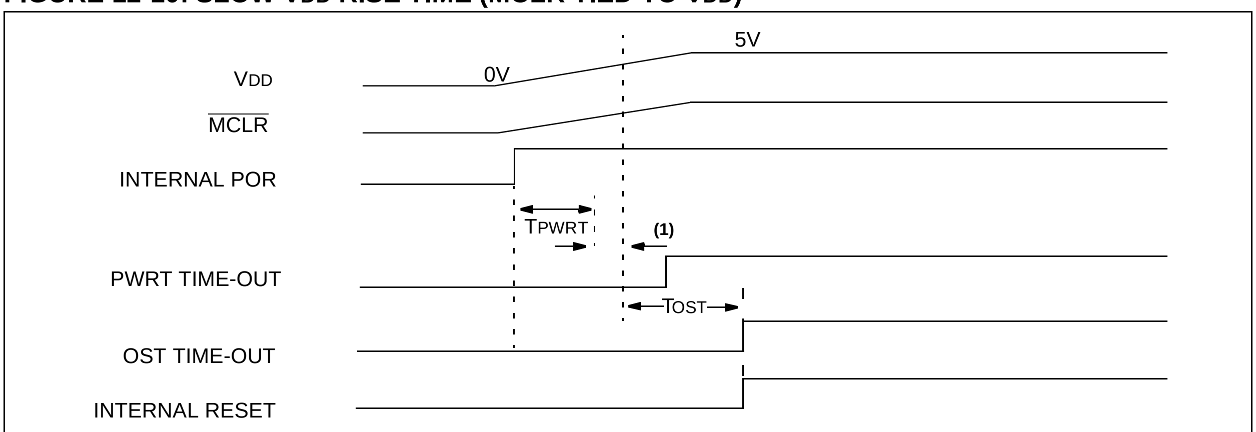


FIGURE 12-10: SLOW V_{DD} RISE TIME ($\overline{MCLR}$ TIED TO V_{DD})



Note 1: Time dependent on oscillator circuit

12.10 Interrupts

The devices have up to 11 sources of interrupt. The interrupt control register (INTCON) records individual interrupt requests in flag bits. It also has individual and global interrupt enable bits.

Note:	Individual interrupt flag bits are set regardless of the status of their corresponding mask bit or the GIE bit.

A global interrupt enable bit, GIE (INTCON<7>), enables (if set) all un-masked interrupts or disables (if cleared) all interrupts. When bit GIE is enabled and an interrupt's flag bit and mask bit are set, the interrupt will vector immediately. Individual interrupts can be disabled through their corresponding enable bits in various registers. Individual interrupt bits are set, regardless of the status of the GIE bit. The GIE bit is cleared on reset.

The "return from interrupt" instruction, `RETFIE`, exits the interrupt routine as well as sets the GIE bit, which re-enables interrupts.

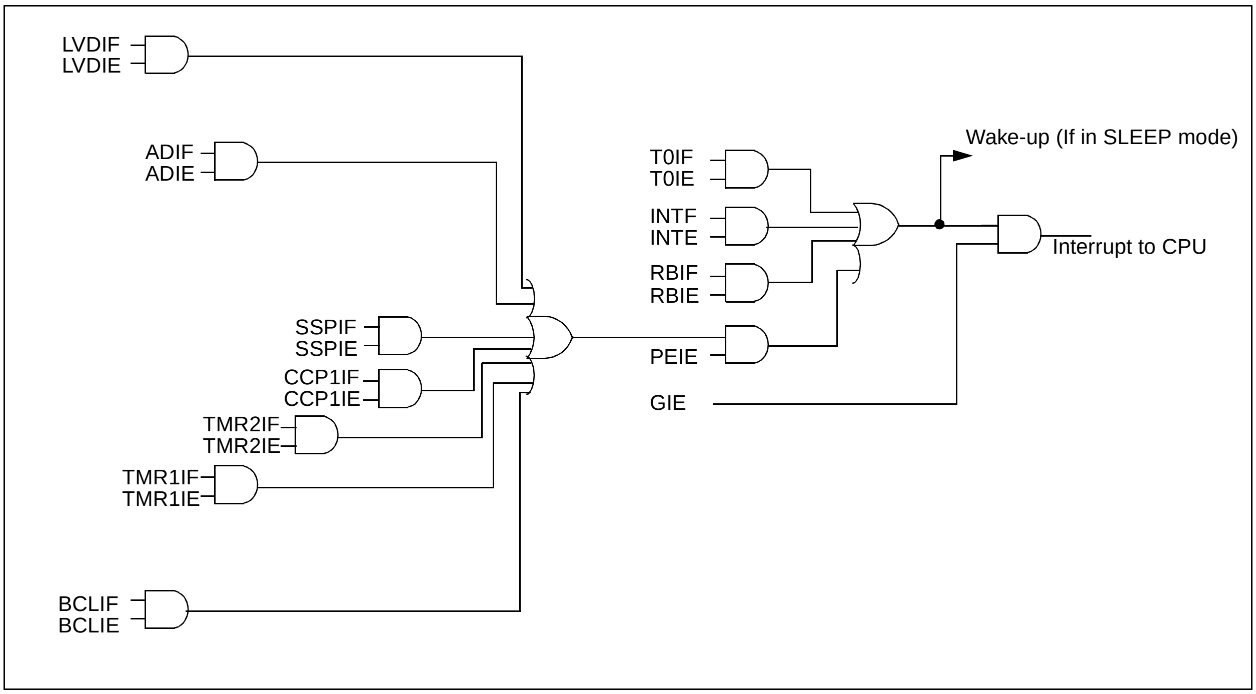
The RB0/INT pin interrupt, the RB port change interrupt and the TMR0 overflow interrupt flags are contained in the INTCON register.

The peripheral interrupt flags are contained in the special function registers PIR1 and PIR2. The corresponding interrupt enable bits are contained in special function registers PIE1 and PIE2, and the peripheral interrupt enable bit is contained in special function register INTCON.

When an interrupt is responded to, the GIE bit is cleared to disable any further interrupt, the return address is pushed onto the stack and the PC is loaded with 0004h. Once in the interrupt service routine the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid recursive interrupts.

For external interrupt events, such as the INT pin or PORTB change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends when the interrupt event occurs. The latency is the same for one or two cycle instructions. Individual interrupt flag bits are set regardless of the status of their corresponding mask bit or the GIE bit

FIGURE 12-11: INTERRUPT LOGIC



12.10.1 INT INTERRUPT

External interrupt on RB0/INT pin is edge triggered: either rising if bit INTEDG (OPTION_REG<6>) is set, or falling, if the INTEDG bit is clear. When a valid edge appears on the RB0/INT pin, flag bit INTF (INTCON<1>) is set. This interrupt can be disabled by clearing enable bit INTE (INTCON<4>). Flag bit INTF must be cleared in software in the interrupt service routine before re-enabling this interrupt. The INT interrupt can wake-up the processor from SLEEP, if bit INTE was set prior to going into SLEEP. The status of global interrupt enable bit GIE decides whether or not the processor branches to the interrupt vector following wake-up. See Section 12.13 for details on SLEEP mode.

12.10.2 TMR0 INTERRUPT

An overflow (FFh $\rightarrow$ 00h) in the TMR0 register will set flag bit T0IF (INTCON<2>). The interrupt can be enabled/disabled by setting/clearing enable bit T0IE (INTCON<5>). (Section 2.2.2.3)

12.10.3 PORTB INTCON CHANGE

An input change on PORTB<7:0> sets flag bit RBIF (INTCON<0>). The PORTB pin(s) which can individually generate interrupt is selectable in the IOCB register. The interrupt can be enabled/disabled by setting/clearing enable bit RBIE (INTCON<4>). (Section 2.2.2.3)

12.11 Context Saving During Interrupts

During an interrupt, only the PC is saved on the stack. At the very least, W and STATUS should be saved to preserve the context for the interrupted program. All registers that may be corrupted by the ISR, such as PCLATH or FSR, should be saved.

Example 12-1 stores and restores the STATUS, W and PCLATH registers. The register, W_TEMP, is defined in Common RAM, the last 16 bytes of each bank that may be accessed from any bank. The STATUS_TEMP and PCLATH_TEMP are defined in bank 0.

The example:

a) Stores the W register.
b) Stores the STATUS register in bank 0.
c) Stores the PCLATH register in bank 0.
d) Executes the ISR code.
e) Restores the PCLATH register.
f) Restores the STATUS register
g) Restores W.

Note that W_TEMP, STATUS_TEMP and PCLATH_TEMP are defined in the common RAM area (70h - 7Fh) to avoid register bank switching during context save and restore.

EXAMPLE 12-1: SAVING STATUS, W, AND PCLATH REGISTERS IN RAM

```
#define    W_TEMP            0x70
#define    STATUS_TEMP       0x71
#define    PCLATH_TEMP       0x72
    org    0x04              ; start at Interrupt Vector
    MOVWF   W_TEMP           ; Save W register
    MOVF   STATUS,w
    MOVWF  STATUS_TEMP       ; save STATUS
    MOVF   PCLATH,w
    MOVWF  PCLATH_TEMP       ; save PCLATH
    :
    (Interrupt Service Routine)
    :
    MOVF   PCLATH_TEMP,w
    MOVWF  PCLATH
    MOVF   STATUS_TEMP,w
    MOVWF  STATUS
    SWAPF  W_TEMP,f          ;
    SWAPF  W_TEMP,w          ; swapf loads W without affecting STATUS flags
    RETFIE
```

 Advanced Information

12.12 Watchdog Timer (WDT)

The Watchdog Timer is a free running on-chip RC oscillator, which does not require any external components. This oscillator is in dependent from the processor clock. The WDT will run, even if the main clock of the device has been stopped, for example, by execution of a SLEEP instruction.

During normal operation, a WDT time-out generates a device RESET (Watchdog Timer Reset). If the device is in SLEEP mode, a WDT time-out causes the device to wake-up and continue with normal operation (Watchdog Timer Wake-up). The $\overline{\text{TO}}$ bit in the STATUS register will be cleared upon a Watchdog Timer time-out.

The WDT can be permanently disabled by programming the configuration bit WDTE (Section 12.1)'0'.

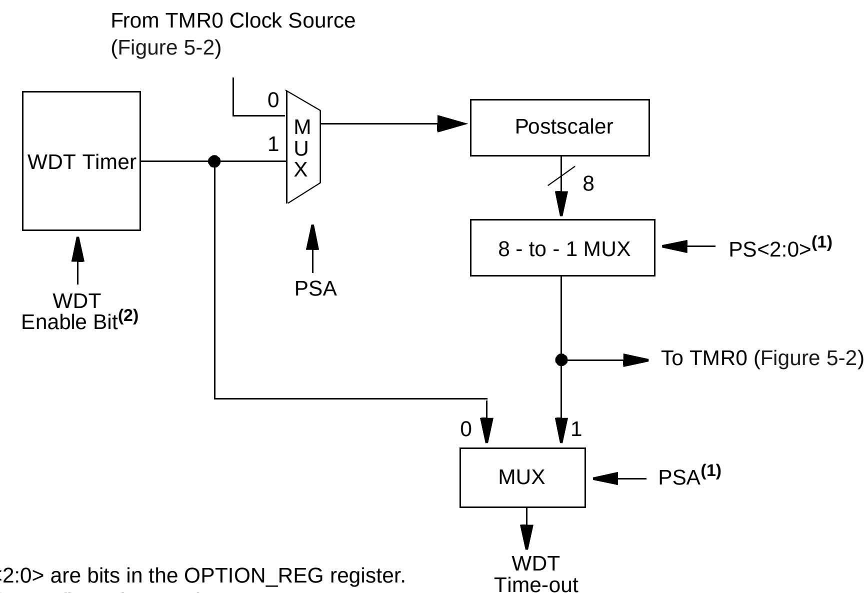
WDT time-out period values may be found in the Electrical Specifications. Values for the WDT prescaler may be assigned using the OPTION_REG register.

Note:	The `CLRWDT` and `SLEEP` instructions clear the WDT and the postscaler, if assigned to the WDT, and prevent it from timing out and generating a device RESET condition.

.

Note:	When a `CLRWDT` instruction is executed and the prescaler is assigned to the WDT, the prescaler count will be cleared, but the prescaler assignment is not changed.

FIGURE 12-12: WATCHDOG TIMER BLOCK DIAGRAM



Note 1: PSA and PS<2:0> are bits in the OPTION_REG register.
2: WDTE bit in the configuration word.

TABLE 12-7: SUMMARY OF WATCHDOG TIMER REGISTERS

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2007h	Config. bits(1)	—	BODEN	$\overline{\text{MCLRE}}$	$\overline{\text{PWRTE}}$	WDTE	FOSC2	FOSC1	FOSC0
81h,181h	OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0

Legend: Shaded cells are not used by the Watchdog Timer.
Note 1: See Figure 12-1 for the full description of the configuration word bits.

12.13 Power-down Mode (SLEEP)

Power-down mode is entered by executing a SLEEP instruction.

If enabled, the Watchdog Timer will be cleared but keeps running, the $\overline{PD}$ bit (STATUS<3>) is cleared, the $\overline{TO}$ (STATUS<4>) bit is set, and the oscillator driver is turned off. The I/O ports maintain the status they had, before the SLEEP instruction was executed (driving high, low, or hi-impedance).

For lowest current consumption in this mode, place all I/O pins at either V_{DD}, or V_{SS}, ensure no external circuitry is drawing current from the I/O pin, power-down the A/D, disable external clocks. Pull all I/O pins, that are hi-impedance inputs, high or low externally to avoid switching currents caused by floating inputs. The T0CKI input should also be at V_{DD} or V_{SS} for lowest current consumption. The contribution from on-chip pull-ups on PORTB should be considered.

12.13.1 WAKE-UP FROM SLEEP

The device can wake up from SLEEP through one of the following events:

1. External reset input on $\overline{MCLR}$ pin.
2. Watchdog Timer Wake-up (if WDT was enabled).
3. Interrupt from INT pin, RB port change, or some Peripheral Interrupts.

External $\overline{MCLR}$ Reset will cause a device reset. All other events are considered a continuation of program execution and cause a "wake-up". The $\overline{TO}$ and $\overline{PD}$ bits in the STATUS register can be used to determine the cause of device reset. The $\overline{PD}$ bit, which is set on power-up, is cleared when SLEEP is invoked. The $\overline{TO}$ bit is cleared if a WDT time-out occurred (and caused wake-up).

The following peripheral interrupts can wake the device from SLEEP:

1. TMR1 interrupt. Timer1 must be operating as an asynchronous counter.
2. CCP capture mode interrupt.
3. Special event trigger (Timer1 in asynchronous mode using an external clock).
4. SSP (Start/Stop) bit detect interrupt.
5. SSP transmit or receive in slave mode (SPI/I^2C).
6. A/D conversion (when A/D clock source is RC).
7. Low-voltage detect.

Other peripherals cannot generate interrupts since during SLEEP, no on-chip clocks are present.

When the SLEEP instruction is being executed, the next instruction (PC + 1) is pre-fetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be set (enabled). Wake-up is regardless of the state of the GIE bit. If the GIE bit is clear (disabled), the device continues execution at the instruction after the SLEEP instruction. If the GIE bit is set (enabled), the device executes the instruction after the SLEEP instruction and then branches to the interrupt address (0004h). In cases where the execution of the instruction following SLEEP is not desirable, the user should have a NOP after the SLEEP instruction.

12.13.2 WAKE-UP USING INTERRUPTS

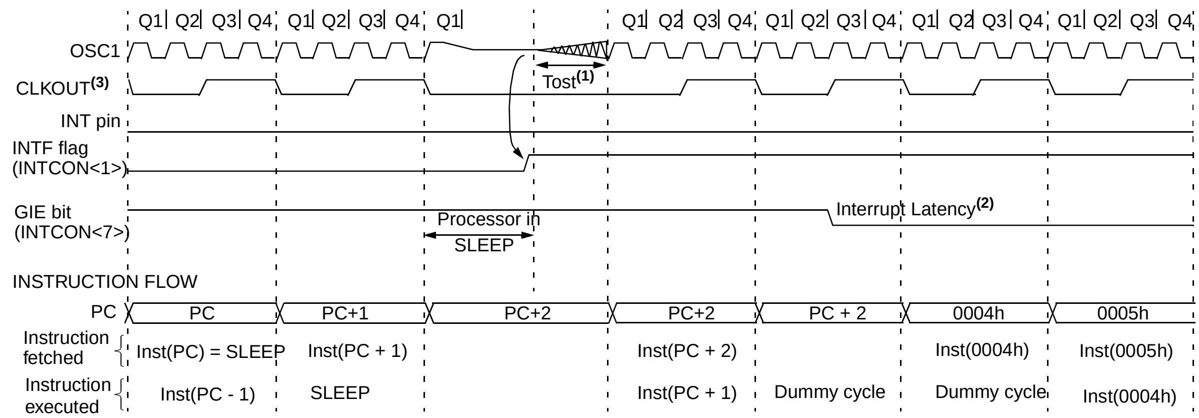
When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the execution of a SLEEP instruction, the SLEEP instruction will complete as a NOP. Therefore, the WDT and WDT postscaler will not be cleared, the $\overline{TO}$ bit will not be set and $\overline{PD}$ bits will not be cleared.
- If the interrupt occurs **during or after** the execution of a SLEEP instruction, the device will immediately wake up from sleep. The SLEEP instruction will be completely executed before the wake-up. Therefore, the WDT and WDT postscaler will be cleared, the $\overline{TO}$ bit will be set and the $\overline{PD}$ bit will be cleared.

Even if the flag bits were checked before executing a SLEEP instruction, it may be possible for flag bits to become set before the SLEEP instruction completes. To determine whether a SLEEP instruction executed, test the $\overline{PD}$ bit. If the $\overline{PD}$ bit is set, the SLEEP instruction was executed as a NOP.

To ensure that the WDT is cleared, a CLRWDT instruction should be executed before a SLEEP instruction.

FIGURE 12-13: WAKE-UP FROM SLEEP THROUGH INTERRUPT



Note 1: T_{OST} = 1024T_{OSC} (drawing not to scale) This delay applies to LP, XT and HS modes only.
2: GIE = '1' assumed. In this case after wake- up, the processor jumps to the interrupt routine. If GIE = '0', execution will continue in-line.
3: CLKOUT is not available in these osc modes, but shown here for timing reference.

12.14 Program Verification/Code Protection

If the code protection bit(s) have not been programmed, the on-chip program memory can be read out for verification purposes.

Note:	Microchip does not recommend code protecting windowed devices.

12.15 ID Locations

Four memory locations (2000h - 2003h) are designated as ID locations where the user can store checksum or other code-identification numbers. These locations are not accessible during normal execution but are readable and writable during program/verify. It is recommended that only the 4 least significant bits of the ID location are used.

For ROM devices, these values are submitted along with the ROM code.

12.16 In-Circuit Serial Programming (ICSP™)

PIC16CXXX microcontrollers can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data, and three other lines for power, ground, and the programming voltage. This allows customers to manufacture boards with unprogrammed devices, and then program the microcontroller just before shipping the product. This also allows the most recent firmware or a custom firmware to be programmed.

For complete details of serial programming, please refer to the In-Circuit Serial Programming (ICSP™) Guide, (DS30277).

NOTES:

13.0 INSTRUCTION SET SUMMARY

Each PIC16CXXX instruction is a 14-bit word divided into an OPCODE which specifies the instruction type and one or more operands which further specify the operation of the instruction. The PIC16CXX instruction set summary in Table 13-2 lists **byte-oriented**, **bit-oriented**, and **literal and control** operations. Table 13-1 shows the opcode field descriptions.

For **byte-oriented** instructions, 'f' represents a file register designator and 'd' represents a destination designator. The file register designator specifies which file register is to be used by the instruction.

The destination designator specifies where the result of the operation is to be placed. If 'd' is zero, the result is placed in the W register. If 'd' is one, the result is placed in the file register specified in the instruction.

For **bit-oriented** instructions, 'b' represents a bit field designator which selects the number of the bit affected by the operation, while 'f' represents the number of the file in which the bit is located.

For **literal and control** operations, 'k' represents an eight or eleven bit constant or literal value.

TABLE 13-1: OPCODE FIELD DESCRIPTIONS

Field	Description
f	Register file address (0x00 to 0x7F)
W	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1) The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1
PC	Program Counter
$\overline{TO}$	Time-out bit
$\overline{PD}$	Power-down bit

The instruction set is highly orthogonal and is grouped into three basic categories:

- **Byte-oriented** operations
- **Bit-oriented** operations
- **Literal and control** operations

All instructions are executed within one single instruction cycle, unless a conditional test is true or the program counter is changed as a result of an instruction. In this case, the execution takes two instruction cycles with the second cycle executed as a NOP. One instruction cycle consists of four oscillator periods. Thus, for an oscillator frequency of 4 MHz, the normal instruction execution time is 1 µs. If a conditional test is true or the program counter is changed as a result of an instruction, the instruction execution time is 2 µs.

Table 13-2 lists the instructions recognized by the MPASM assembler.

Figure 13-1 shows the general formats that the instructions can have.

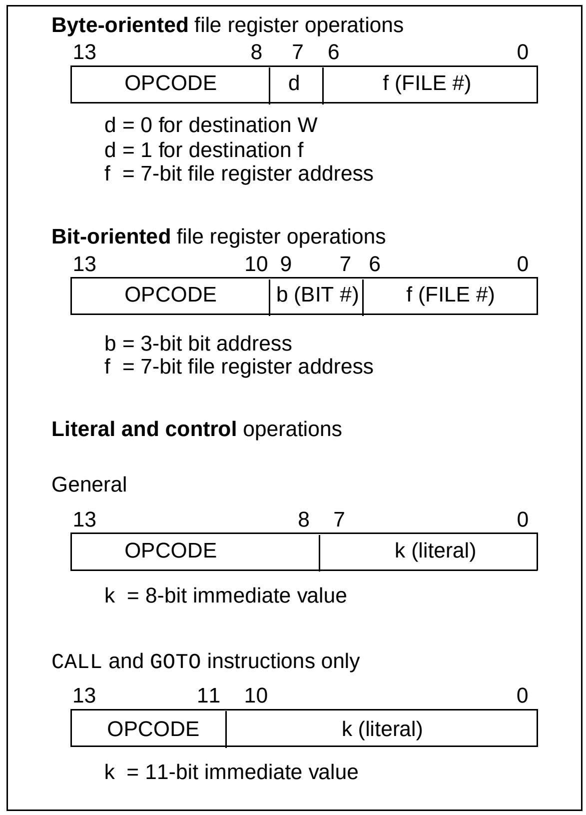
Note:	To maintain upward compatibility with future PIC16CXXX products, do not use the `OPTION` and `TRIS` instructions.

All examples use the following format to represent a hexadecimal number:

0xhh

where h signifies a hexadecimal digit.

FIGURE 13-1: GENERAL FORMAT FOR INSTRUCTIONS



A description of each instruction is available in the PICmicro™ Mid-Range Reference Manual, (DS33023).

TABLE 13-2: PIC16CXXX INSTRUCTION SET

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb			LSb	Status Affected	Notes
BYTE-ORIENTED FILE REGISTER OPERATIONS									
ADDWF	**f, d**	Add W and f	1	00	0111	dfff	ffff	C,DC,Z	1,2
ANDWF	**f, d**	AND W with f	1	00	0101	dfff	ffff	Z	1,2
CLRF	**f**	Clear f	1	00	0001	lfff	ffff	Z	2
CLRW	**-**	Clear W	1	00	0001	0000	0011	Z	
COMF	**f, d**	Complement f	1	00	1001	dfff	ffff	Z	1,2
DECF	**f, d**	Decrement f	1	00	0011	dfff	ffff	Z	1,2
DECFSZ	**f, d**	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1,2,3
INCF	**f, d**	Increment f	1	00	1010	dfff	ffff	Z	1,2
INCFSZ	**f, d**	Increment f, Skip if 0	1(2)	00	1111	dfff	ffff		1,2,3
IORWF	**f, d**	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	1,2
MOVF	**f, d**	Move f	1	00	1000	dfff	ffff	Z	1,2
MOVWF	**f**	Move W to f	1	00	0000	lfff	ffff		
NOP	**-**	No Operation	1	00	0000	0xx0	0000		
RLF	**f, d**	Rotate Left f through Carry	1	00	1101	dfff	ffff	C	1,2
RRF	**f, d**	Rotate Right f through Carry	1	00	1100	dfff	ffff	C	1,2
SUBWF	**f, d**	Subtract W from f	1	00	0010	dfff	ffff	C,DC,Z	1,2
SWAPF	**f, d**	Swap nibbles in f	1	00	1110	dfff	ffff		1,2
XORWF	**f, d**	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	1,2
BIT-ORIENTED FILE REGISTER OPERATIONS									
BCF	**f, b**	Bit Clear f	1	01	00bb	bfff	ffff		1,2
BSF	**f, b**	Bit Set f	1	01	01bb	bfff	ffff		1,2
BTFSC	**f, b**	Bit Test f, Skip if Clear	1 (2)	01	10bb	bfff	ffff		3
BTFSS	**f, b**	Bit Test f, Skip if Set	1 (2)	01	11bb	bfff	ffff		3
LITERAL AND CONTROL OPERATIONS									
ADDLW	**k**	Add literal and W	1	11	111x	kkkk	kkkk	C,DC,Z	
ANDLW	**k**	AND literal with W	1	11	1001	kkkk	kkkk	Z	
CALL	**k**	Call subroutine	2	10	0kkk	kkkk	kkkk		
CLRWDT	**-**	Clear Watchdog Timer	1	00	0000	0110	0100	$\overline{TO}$,$\overline{PD}$	
GOTO	**k**	Go to address	2	10	1kkk	kkkk	kkkk		
IORLW	**k**	Inclusive OR literal with W	1	11	1000	kkkk	kkkk	Z	
MOVLW	**k**	Move literal to W	1	11	00xx	kkkk	kkkk		
RETFIE	**-**	Return from interrupt	2	00	0000	0000	1001		
RETLW	**k**	Return with literal in W	2	11	01xx	kkkk	kkkk		
RETURN	**-**	Return from Subroutine	2	00	0000	0000	1000		
SLEEP	**-**	Go into standby mode	1	00	0000	0110	0011	$\overline{TO}$,$\overline{PD}$	
SUBLW	**k**	Subtract W from literal	1	11	110x	kkkk	kkkk	C,DC,Z	
XORLW	**k**	Exclusive OR literal with W	1	11	1010	kkkk	kkkk	Z	

Note 1: When an I/O register is modified as a function of itself (e.g., `MOVF PORTB, 1`), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

2: If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 Module.

3: If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

13.1 Instruction Descriptions

ADDLW	Add Literal and W
Syntax:	[*label*] ADDLW k
Operands:	$0 \le k \le 255$
Operation:	(W) + k → (W)
Status Affected:	C, DC, Z
Description:	The contents of the W register are added to the eight bit literal 'k' and the result is placed in the W register.

ADDWF	Add W and f
Syntax:	[*label*] ADDWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(W) + (f) → (destination)
Status Affected:	C, DC, Z
Description:	Add the contents of the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

ANDLW	AND Literal with W
Syntax:	[*label*] ANDLW k
Operands:	$0 \le k \le 255$
Operation:	(W) .AND. (k) → (W)
Status Affected:	Z
Description:	The contents of W register are AND'ed with the eight bit literal 'k'. The result is placed in the W register.

ANDWF	AND W with f
Syntax:	[*label*] ANDWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(W) .AND. (f) → (destination)
Status Affected:	Z
Description:	AND the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

BCF	Bit Clear f
Syntax:	[*label*] BCF f,b
Operands:	$0 \le f \le 127$ $0 \le b \le 7$
Operation:	0 → (f<b>)
Status Affected:	None
Description:	Bit 'b' in register 'f' is cleared.

BSF	Bit Set f
Syntax:	[*label*] BSF f,b
Operands:	$0 \le f \le 127$ $0 \le b \le 7$
Operation:	1 → (f<b>)
Status Affected:	None
Description:	Bit 'b' in register 'f' is set.

BTFSS	Bit Test f, Skip if Set
Syntax:	[*label*] BTFSS f,b
Operands:	$0 \le f \le 127$ $0 \le b < 7$
Operation:	skip if (f<b>) = 1
Status Affected:	None
Description:	If bit 'b' in register 'f' is '0', the next instruction is executed. If bit 'b' is '1', then the next instruction is discarded and a `NOP` is executed instead making this a 2TCY instruction.

BTFSC	Bit Test, Skip if Clear
Syntax:	[*label*] BTFSC f,b
Operands:	$0 \le f \le 127$ $0 \le b \le 7$
Operation:	skip if (f<b>) = 0
Status Affected:	None
Description:	If bit 'b' in register 'f' is '1', the next instruction is executed. If bit 'b', in register 'f', is '0', the next instruction is discarded, and a `NOP` is executed instead, making this a 2TCY instruction.

CALL	Call Subroutine
Syntax:	[*label*] CALL k
Operands:	$0 \le k \le 2047$
Operation:	(PC)+ 1 $\rightarrow$ TOS, k $\rightarrow$ PC<10:0>, (PCLATH<4:3>) $\rightarrow$ PC<12:11>
Status Affected:	None
Description:	Call Subroutine. First, return address (PC+1) is pushed onto the stack. The eleven bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. `CALL` is a two cycle instruction.

CLRF	Clear f
Syntax:	[*label*] CLRF f
Operands:	$0 \le f \le 127$
Operation:	00h $\rightarrow$ (f) 1 $\rightarrow$ Z
Status Affected:	Z
Description:	The contents of register 'f' are cleared and the Z bit is set.

CLRW	Clear W
Syntax:	[*label*] CLRW
Operands:	None
Operation:	00h $\rightarrow$ (W) 1 $\rightarrow$ Z
Status Affected:	Z
Description:	W register is cleared. Zero bit (Z) is set.

CLRWDT	Clear Watchdog Timer
Syntax:	[*label*] CLRWDT
Operands:	None
Operation:	00h $\rightarrow$ WDT 0 $\rightarrow$ WDT prescaler, 1 $\rightarrow$ $\overline{TO}$ 1 $\rightarrow$ $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	`CLRWDT` instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits $\overline{TO}$ and $\overline{PD}$ are set.

COMF	**Complement f**
Syntax:	[*label*] COMF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	$(\bar{f})$ → (destination)
Status Affected:	Z
Description:	The contents of register 'f' are complemented. If 'd' is 0, the result is stored in W. If 'd' is 1, the result is stored back in register 'f'.

DECF	**Decrement f**
Syntax:	[*label*] DECF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f) - 1 → (destination)
Status Affected:	Z
Description:	Decrement register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

DECFSZ	**Decrement f, Skip if 0**
Syntax:	[*label*] DECFSZ f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f) - 1 → (destination); skip if result = 0
Status Affected:	None
Description:	The contents of register 'f' are decremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'. If the result is 1, the next instruction is executed. If the result is 0, then a `NOP` is executed instead making it a 2TCY instruction.

GOTO	**Unconditional Branch**
Syntax:	[*label*] GOTO k
Operands:	0 ≤ k ≤ 2047
Operation:	k → PC<10:0> PCLATH<4:3> → PC<12:11>
Status Affected:	None
Description:	`GOTO` is an unconditional branch. The eleven bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH<4:3>. `GOTO` is a two cycle instruction.

INCF	**Increment f**
Syntax:	[*label*] INCF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f) + 1 → (destination)
Status Affected:	Z
Description:	The contents of register 'f' are incremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'.

INCFSZ	**Increment f, Skip if 0**
Syntax:	[*label*] INCFSZ f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f) + 1 → (destination), skip if result = 0
Status Affected:	None
Description:	The contents of register 'f' are incremented. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'. If the result is 1, the next instruction is executed. If the result is 0, a `NOP` is executed instead making it a 2TCY instruction.

IORLW	Inclusive OR Literal with W
Syntax:	[*label*] IORLW k
Operands:	$0 \le k \le 255$
Operation:	(W) .OR. k → (W)
Status Affected:	Z
Description:	The contents of the W register are OR'ed with the eight bit literal 'k'. The result is placed in the W register.

IORWF	Inclusive OR W with f
Syntax:	[*label*] IORWF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(W) .OR. (f) → (destination)
Status Affected:	Z
Description:	Inclusive OR the W register with register 'f'. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'.

MOVF	Move f
Syntax:	[*label*] MOVF f,d
Operands:	$0 \le f \le 127$ $d \in [0,1]$
Operation:	(f) → (destination)
Status Affected:	Z
Description:	The contents of register f are moved to a destination dependant upon the status of d. If d = 0, destination is W register. If d = 1, the destination is file register f itself. d = 1 is useful to test a file register since status flag Z is affected.

MOVLW	Move Literal to W
Syntax:	[*label*] MOVLW k
Operands:	$0 \le k \le 255$
Operation:	k → (W)
Status Affected:	None
Description:	The eight bit literal 'k' is loaded into W register. The don't cares will assemble as 0's.

MOVWF	Move W to f
Syntax:	[*label*] MOVWF f
Operands:	$0 \le f \le 127$
Operation:	(W) → (f)
Status Affected:	None
Description:	Move data from W register to register 'f'.

NOP	No Operation
Syntax:	[*label*] NOP
Operands:	None
Operation:	No operation
Status Affected:	None
Description:	No operation.

RETFIE — **Return from Interrupt**

Syntax:	[*label*] RETFIE
Operands:	None
Operation:	TOS → PC, 1 → GIE
Status Affected:	None

RETLW — **Return with Literal in W**

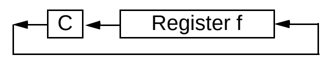
Syntax:	[*label*] RETLW k
Operands:	0 ≤ k ≤ 255
Operation:	k → (W); TOS → PC
Status Affected:	None
Description:	The W register is loaded with the eight bit literal 'k'. The program counter is loaded from the top of the stack (the return address). This is a two cycle instruction.

RETURN — **Return from Subroutine**

Syntax:	[*label*] RETURN
Operands:	None
Operation:	TOS → PC
Status Affected:	None
Description:	Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two cycle instruction.

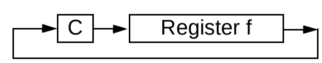
RLF — **Rotate Left f through Carry**

Syntax:	[*label*] RLF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	See description below
Status Affected:	C
Description:	The contents of register 'f' are rotated one bit to the left through the Carry Flag. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is stored back in register 'f'.



RRF — **Rotate Right f through Carry**

Syntax:	[*label*] RRF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	See description below
Status Affected:	C
Description:	The contents of register 'f' are rotated one bit to the right through the Carry Flag. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'.



SLEEP

Syntax:	[*label*] SLEEP
Operands:	None
Operation:	00h → WDT, 0 → WDT prescaler, 1 → $\overline{TO}$, 0 → $\overline{PD}$
Status Affected:	$\overline{TO}$, $\overline{PD}$
Description:	The power-down status bit, $\overline{PD}$ is cleared. Time-out status bit, $\overline{TO}$ is set. Watchdog Timer and its prescaler are cleared. The processor is put into SLEEP mode with the oscillator stopped. See Section 13.8 for more details.

SUBLW	Subtract W from Literal
Syntax:	[*label*] SUBLW k
Operands:	0 ≤ k ≤ 255
Operation:	k - (W) → (W)
Status Affected:	C, DC, Z
Description:	The W register is subtracted (2's complement method) from the eight bit literal 'k'. The result is placed in the W register.

SUBWF	Subtract W from f
Syntax:	[*label*] SUBWF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f) - (W) → (destination)
Status Affected:	C, DC, Z
Description:	Subtract (2's complement method) W register from register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

SWAPF	Swap Nibbles in f
Syntax:	[*label*] SWAPF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(f<3:0>) → (destination<7:4>), (f<7:4>) → (destination<3:0>)
Status Affected:	None
Description:	The upper and lower nibbles of register 'f' are exchanged. If 'd' is 0, the result is placed in W register. If 'd' is 1, the result is placed in register 'f'.

XORLW	Exclusive OR Literal with W
Syntax:	[*label*] XORLW k
Operands:	0 ≤ k ≤ 255
Operation:	(W) .XOR. k → (W)
Status Affected:	Z
Description:	The contents of the W register are XOR'ed with the eight bit literal 'k'. The result is placed in the W register.

XORWF	Exclusive OR W with f
Syntax:	[*label*] XORWF f,d
Operands:	0 ≤ f ≤ 127 d ∈ [0,1]
Operation:	(W) .XOR. (f) → (destination)
Status Affected:	Z
Description:	Exclusive OR the contents of the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

14.0 DEVELOPMENT SUPPORT

The PICmicro® microcontrollers are supported with a full range of hardware and software development tools:

- Integrated Development Environment
  - MPLAB™ IDE Software
- Assemblers/Compilers/Linkers
  - MPASM Assembler
  - MPLAB-C17 and MPLAB-C18 C Compilers
  - MPLINK/MPLIB Linker/Librarian
- Simulators
  - MPLAB-SIM Software Simulator
- Emulators
  - MPLAB-ICE Real-Time In-Circuit Emulator
  - PICMASTER®/PICMASTER-CE In-Circuit Emulator
  - ICEPIC™
- In-Circuit Debugger
  - MPLAB-ICD for PIC16F877
- Device Programmers
  - PRO MATE® II Universal Programmer
  - PICSTART® Plus Entry-Level Prototype Programmer
- Low-Cost Demonstration Boards
  - SIMICE
  - PICDEM-1
  - PICDEM-2
  - PICDEM-3
  - PICDEM-17
  - SEEVAL®
  - KEELOQ®

14.1 MPLAB Integrated Development Environment Software

The MPLAB IDE software brings an ease of software development previously unseen in the 8-bit microcontroller market. MPLAB is a Windows®-based application which contains:

- Multiple functionality
  - editor
  - simulator
  - programmer (sold separately)
  - emulator (sold separately)
- A full featured editor
- A project manager
- Customizable tool bar and key mapping
- A status bar
- On-line help

MPLAB allows you to:

- Edit your source files (either assembly or 'C')
- One touch assemble (or compile) and download to PICmicro tools (automatically updates all project information)
- Debug using:
  - source files
  - absolute listing file
  - object code

The ability to use MPLAB with Microchip's simulator, MPLAB-SIM, allows a consistent platform and the ability to easily switch from the cost-effective simulator to the full featured emulator with minimal retraining.

14.2 MPASM Assembler

MPASM is a full featured universal macro assembler for all PICmicro MCU's. It can produce absolute code directly in the form of HEX files for device programmers, or it can generate relocatable objects for MPLINK.

MPASM has a command line interface and a Windows shell and can be used as a standalone application on a Windows 3.x or greater system. MPASM generates relocatable object files, Intel standard HEX files, MAP files to detail memory usage and symbol reference, an absolute LST file which contains source lines and generated machine code, and a COD file for MPLAB debugging.

MPASM features include:

- MPASM and MPLINK are integrated into MPLAB projects.
- MPASM allows user defined macros to be created for streamlined assembly.
- MPASM allows conditional assembly for multi purpose source files.
- MPASM directives allow complete control over the assembly process.

14.3 MPLAB-C17 and MPLAB-C18 C Compilers

The MPLAB-C17 and MPLAB-C18 Code Development Systems are complete ANSI 'C' compilers and integrated development environments for Microchip's PIC17CXXX and PIC18CXXX family of microcontrollers, respectively. These compilers provide powerful integration capabilities and ease of use not found with other compilers.

For easier source level debugging, the compilers provide symbol information that is compatible with the MPLAB IDE memory display.

14.4 MPLINK/MPLIB Linker/Librarian

MPLINK is a relocatable linker for MPASM and MPLAB-C17 and MPLAB-C18. It can link relocatable objects from assembly or C source files along with pre-compiled libraries using directives from a linker script.

MPLIB is a librarian for pre-compiled code to be used with MPLINK. When a routine from a library is called from another source file, only the modules that contains that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications. MPLIB manages the creation and modification of library files.

MPLINK features include:

- MPLINK works with MPASM and MPLAB-C17 and MPLAB-C18.
- MPLINK allows all memory areas to be defined as sections to provide link-time flexibility.

MPLIB features include:

- MPLIB makes linking easier because single libraries can be included instead of many smaller files.
- MPLIB helps keep code maintainable by grouping related modules together.
- MPLIB commands allow libraries to be created and modules to be added, listed, replaced, deleted, or extracted.

14.5 MPLAB-SIM Software Simulator

The MPLAB-SIM Software Simulator allows code development in a PC host environment by simulating the PICmicro series microcontrollers on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a file or user-defined key press to any of the pins. The execution can be performed in single step, execute until break, or trace mode.

MPLAB-SIM fully supports symbolic debugging using MPLAB-C17 and MPLAB-C18 and MPASM. The Software Simulator offers the flexibility to develop and debug code outside of the laboratory environment making it an excellent multi-project software development tool.

14.6 MPLAB-ICE High Performance Universal In-Circuit Emulator with MPLAB IDE

The MPLAB-ICE Universal In-Circuit Emulator is intended to provide the product development engineer with a complete microcontroller design tool set for PICmicro microcontrollers (MCUs). Software control of MPLAB-ICE is provided by the MPLAB Integrated Development Environment (IDE), which allows editing, “make” and download, and source debugging from a single environment.

Interchangeable processor modules allow the system to be easily reconfigured for emulation of different processors. The universal architecture of the MPLAB-ICE allows expansion to support new PICmicro microcontrollers.

The MPLAB-ICE Emulator System has been designed as a real-time emulation system with advanced features that are generally found on more expensive development tools. The PC platform and Microsoft® Windows 3.x/95/98 environment were chosen to best make these features available to you, the end user.

MPLAB-ICE 2000 is a full-featured emulator system with enhanced trace, trigger, and data monitoring features. Both systems use the same processor modules and will operate across the full operating speed range of the PICmicro MCU.

14.7 PICMASTER/PICMASTER CE

The PICMASTER system from Microchip Technology is a full-featured, professional quality emulator system. This flexible in-circuit emulator provides a high-quality, universal platform for emulating Microchip 8-bit PICmicro microcontrollers (MCUs). PICMASTER systems are sold worldwide, with a CE compliant model available for European Union (EU) countries.

14.8 ICEPIC

ICEPIC is a low-cost in-circuit emulation solution for the Microchip Technology PIC16C5X, PIC16C6X, PIC16C7X, and PIC16CXXX families of 8-bit one-time-programmable (OTP) microcontrollers. The modular system can support different subsets of PIC16C5X or PIC16CXXX products through the use of interchangeable personality modules or daughter boards. The emulator is capable of emulating without target application circuitry being present.

14.9 MPLAB-ICD In-Circuit Debugger

Microchip’s In-Circuit Debugger, MPLAB-ICD, is a powerful, low-cost run-time development tool. This tool is based on the flash PIC16F877 and can be used to develop for this and other PICmicro microcontrollers from the PIC16CXXX family. MPLAB-ICD utilizes the In-Circuit Debugging capability built into the PIC16F87X. This feature, along with Microchip’s In-Circuit Serial Programming protocol, offers cost-effective in-circuit flash programming and debugging from the graphical user interface of the MPLAB Integrated Development Environment. This enables a designer to develop and debug source code by watching variables, single-stepping and setting break points. Running at full speed enables testing hardware in real-time. The MPLAB-ICD is also a programmer for the flash PIC16F87X family.

14.10 PRO MATE II Universal Programmer

The PRO MATE II Universal Programmer is a full-featured programmer capable of operating in stand-alone mode as well as PC-hosted mode. PRO MATE II is CE compliant.

The PRO MATE II has programmable V_{DD} and V_{PP} supplies which allows it to verify programmed memory at V_{DD} min and V_{DD} max for maximum reliability. It has an LCD display for instructions and error messages, keys to enter commands and a modular detachable socket assembly to support various package types. In stand-alone mode the PRO MATE II can read, verify or program PICmicro devices. It can also set code-protect bits in this mode.

14.11 PICSTART Plus Entry Level Development System

The PICSTART programmer is an easy-to-use, low-cost prototype programmer. It connects to the PC via one of the COM (RS-232) ports. MPLAB Integrated Development Environment software makes using the programmer simple and efficient.

PICSTART Plus supports all PICmicro devices with up to 40 pins. Larger pin count devices such as the PIC16C92X, and PIC17C76X may be supported with an adapter socket. PICSTART Plus is CE compliant.

14.12 SIMICE Entry-Level Hardware Simulator

SIMICE is an entry-level hardware development system designed to operate in a PC-based environment with Microchip's simulator MPLAB-SIM. Both SIMICE and MPLAB-SIM run under Microchip Technology's MPLAB Integrated Development Environment (IDE) software. Specifically, SIMICE provides hardware simulation for Microchip's PIC12C5XX, PIC12CE5XX, and PIC16C5X families of PICmicro 8-bit microcontrollers. SIMICE works in conjunction with MPLAB-SIM to provide non-real-time I/O port emulation. SIMICE enables a developer to run simulator code for driving the target system. In addition, the target system can provide input to the simulator code. This capability allows for simple and interactive debugging without having to manually generate MPLAB-SIM stimulus files. SIMICE is a valuable debugging tool for entry-level system development.

14.13 PICDEM-1 Low-Cost PICmicro Demonstration Board

The PICDEM-1 is a simple board which demonstrates the capabilities of several of Microchip's microcontrollers. The microcontrollers supported are: PIC16C5X (PIC16C54 to PIC16C58A), PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 and PIC17C44. All necessary hardware and software is included to run basic demo programs. The users can program the sample microcontrollers provided with the PICDEM-1 board, on a PRO MATE II or PICSTART-Plus programmer, and easily test firmware. The user can also connect the PICDEM-1 board to the MPLAB-ICE emulator and download the firmware to the emulator for testing. Additional prototype area is available for the user to build some additional hardware and connect it to the microcontroller socket(s). Some of the features include an RS-232 interface, a potentiometer for simulated analog input, push-button switches and eight LEDs connected to PORTB.

14.14 PICDEM-2 Low-Cost PIC16CXX Demonstration Board

The PICDEM-2 is a simple demonstration board that supports the PIC16C62, PIC16C64, PIC16C65, PIC16C73 and PIC16C74 microcontrollers. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM-2 board, on a PRO MATE II programmer or PICSTART-Plus, and easily test firmware. The MPLAB-ICE emulator may also be used with the PICDEM-2 board to test firmware. Additional prototype area has been provided to the user for adding additional hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push-button switches, a potentiometer for simulated analog input, a Serial EEPROM to demonstrate usage of the I^2C bus and separate headers for connection to an LCD module and a keypad.

14.15 PICDEM-3 Low-Cost PIC16CXXX Demonstration Board

The PICDEM-3 is a simple demonstration board that supports the PIC16C923 and PIC16C924 in the PLCC package. It will also support future 44-pin PLCC microcontrollers with a LCD Module. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM-3 board, on a PRO MATE II programmer or PICSTART Plus with an adapter socket, and easily test firmware. The MPLAB-ICE emulator may also be used with the PICDEM-3 board to test firmware. Additional prototype area has been provided to the user for adding hardware and connecting it to the microcontroller socket(s). Some of the features include an RS-232 interface, push-button switches, a potentiometer for simulated analog input, a thermistor and separate headers for connection to an external LCD module and a keypad. Also provided on the PICDEM-3 board is an LCD panel, with 4 commons and 12 segments, that is capable of displaying time, temperature and day of the week. The PICDEM-3 provides an additional RS-232 interface and Windows 3.1 software for showing the demultiplexed LCD signals on a PC. A simple serial interface allows the user to construct a hardware demultiplexer for the LCD signals.

14.16 PICDEM-17

The PICDEM-17 is an evaluation board that demonstrates the capabilities of several Microchip microcontrollers, including PIC17C752, PIC17C756, PIC17C762, and PIC17C766. All necessary hardware is included to run basic demo programs, which are supplied on a 3.5-inch disk. A programmed sample is included, and the user may erase it and program it with the other sample programs using the PRO MATE II or PICSTART Plus device programmers and easily debug and test the sample code. In addition, PICDEM-17 supports down-loading of programs to and executing out of external FLASH memory on board. The PICDEM-17 is also usable with the MPLAB-ICE or PICMASTER emulator, and all of the sample programs can be run and modified using either emulator. Additionally, a generous prototype area is available for user hardware.

14.17 SEEVAL Evaluation and Programming System

The SEEVAL SEEPROM Designer's Kit supports all Microchip 2-wire and 3-wire Serial EEPROMs. The kit includes everything necessary to read, write, erase or program special features of any Microchip SEEPROM product including Smart Serials™ and secure serials. The Total Endurance™ Disk is included to aid in trade-off analysis and reliability calculations. The total kit can significantly reduce time-to-market and result in an optimized system.

14.18 KEELOQ Evaluation and Programming Tools

KEELOQ evaluation and programming tools support Microchips HCS Secure Data Products. The HCS evaluation kit includes an LCD display to show changing codes, a decoder to decode transmissions, and a programming interface to program test transmitters.

TABLE 14-1: DEVELOPMENT TOOLS FROM MICROCHIP

		PIC12CXXX	PIC14000	PIC16C5X	PIC16C6X	PIC16CXXX	PIC16F62X	PIC16C7X	PIC16C7XX	PIC16C8X	PIC16F8XX	PIC16C9XX	PIC17C4X	PIC17C7XX	PIC18CXX2	24CXX/ 25CXX/ 93CXX	HCSXXX	MCRFXXX	MCP2510
Software Tools	**MPLAB™ Integrated Development Environment**	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
	MPLAB™ C17 Compiler												✓	✓					
	MPLAB™ C18 Compiler														✓				
	MPASM/MPLINK	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Emulators	**MPLAB™-ICE**	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓				
	PICMASTER/PICMASTER-CE	✓	✓	✓	✓	✓		✓	✓	✓		✓	✓	✓					
	ICEPIC™ Low-Cost In-Circuit Emulator	✓		✓	✓	✓		✓	✓	✓		✓							
Debugger	**MPLAB-ICD In-Circuit Debugger**				✓*			✓*			✓								
Programmers	**PICSTART®Plus Low-Cost Universal Dev. Kit**	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓				
	PRO MATE® II Universal Programmer	✓	✓	✓	✓	✓	✓**	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Demo Boards and Eval Kits	**SIMICE**	✓		✓															
	PICDEM-1			✓		✓		✓†		✓			✓						
	PICDEM-2				✓†			✓†							✓				
	PICDEM-3											✓							
	PICDEM-14A		✓																
	PICDEM-17													✓					
	KEELOQ® Evaluation Kit																✓		
	KEELOQ Transponder Kit																✓		
	microID™ Programmer's Kit																	✓	
	125 kHz microID Developer's Kit																	✓	
	125 kHz Anticollision microID Developer's Kit																	✓	
	13.56 MHz Anticollision microID Developer's Kit																	✓	
	MCP2510 CAN Developer's Kit																		✓

* Contact the Microchip Technology Inc. web site at www.microchip.com for information on how to use the MPLAB-ICD In-Circuit Debugger (DV164001) with PIC16C62, 63, 64, 65, 72, 73, 74, 76, 77

** Contact Microchip Technology Inc. for availability date.

† Development tool is available on select devices.

NOTES:

15.0 ELECTRICAL CHARACTERISTICS

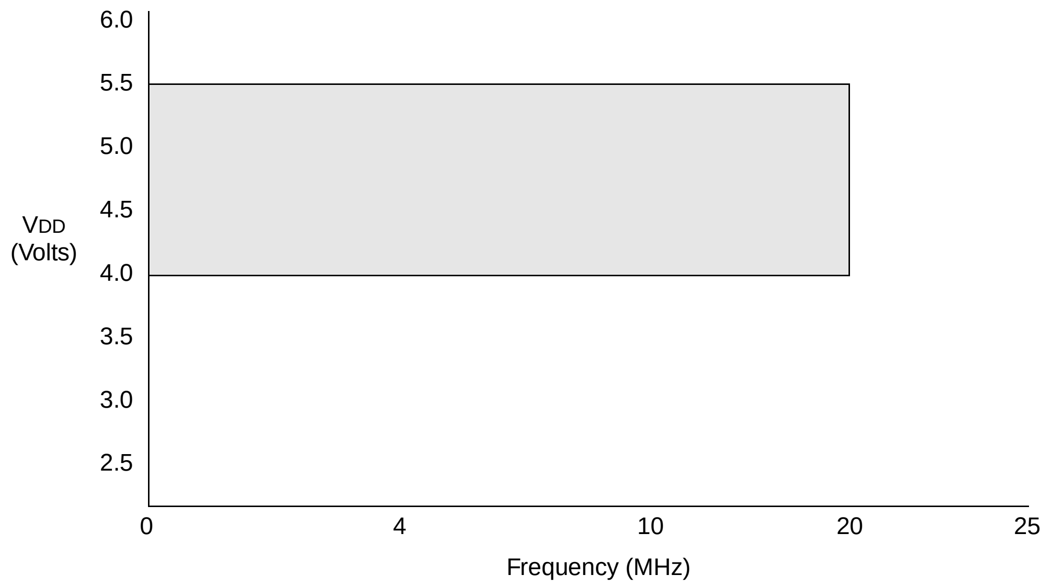
Absolute Maximum Ratings †

Ambient temperature under bias	-55 to +125°C
Storage temperature	-65°C to +150°C
Voltage on any pin with respect to V_{SS} (except V_{DD}, $\overline{MCLR}$ and RA4)	-0.3V to (V_{DD} + 0.3V)
Voltage on V_{DD} with respect to V_{SS}	-0.3 to +7.5V
Maximum voltage between AV_{DD} and V_{DD} pins	± 0.3V
Maximum voltage between AV_{SS} and V_{SS} pins	± 0.3V
Voltage on $\overline{MCLR}$ with respect to V_{SS}	-0.3V to +8.5V
Voltage on RA4 with respect to Vss	-0.3V to +10.5V
Total power dissipation (Note 1)	1.0W
Maximum current out of V_{SS} pin	300 mA
Maximum current into V_{DD} pin	250 mA
Input clamp current, I_{IK} (V_I < 0 or V_I > V_{DD})	± 20 mA
Output clamp current, I_{OK} (V_O < 0 or V_O > V_{DD})	± 20 mA
Maximum output current sunk by any I/O pin	25 mA
Maximum output current sourced by any I/O pin	25 mA
Maximum current sunk by PORTA and PORTB (combined)	200 mA
Maximum current sourced by PORTA and PORTB (combined)	200 mA

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

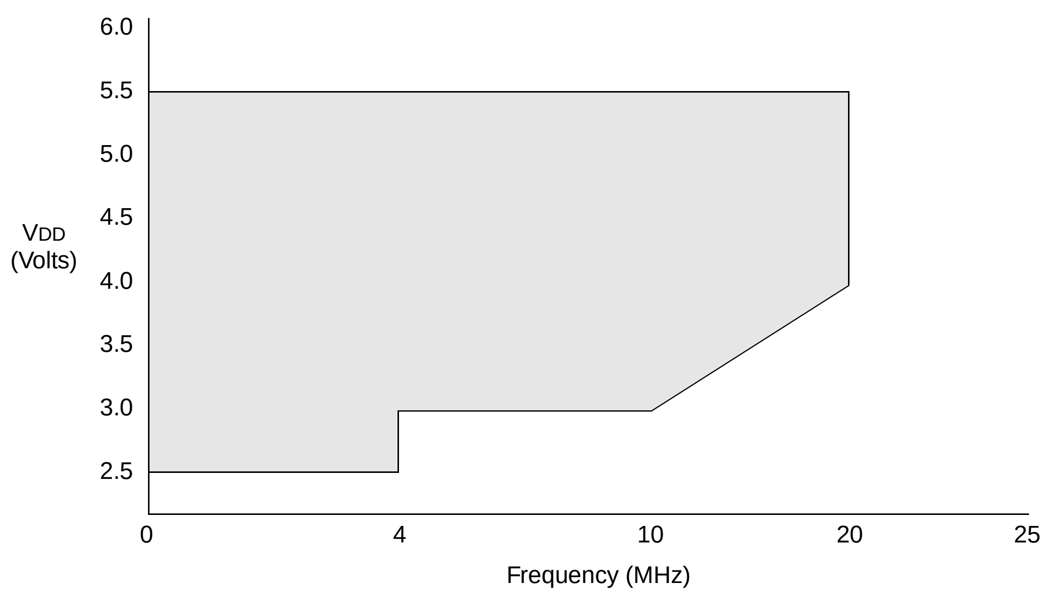
† NOTICE: Stresses above those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure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

FIGURE 15-1: PIC16C717/770/771 VOLTAGE-FREQUENCY GRAPH, -40°C ≤ T_A ≤ +85°C



Note 1: The shaded region indicates the permissible combinations of voltage and frequency.

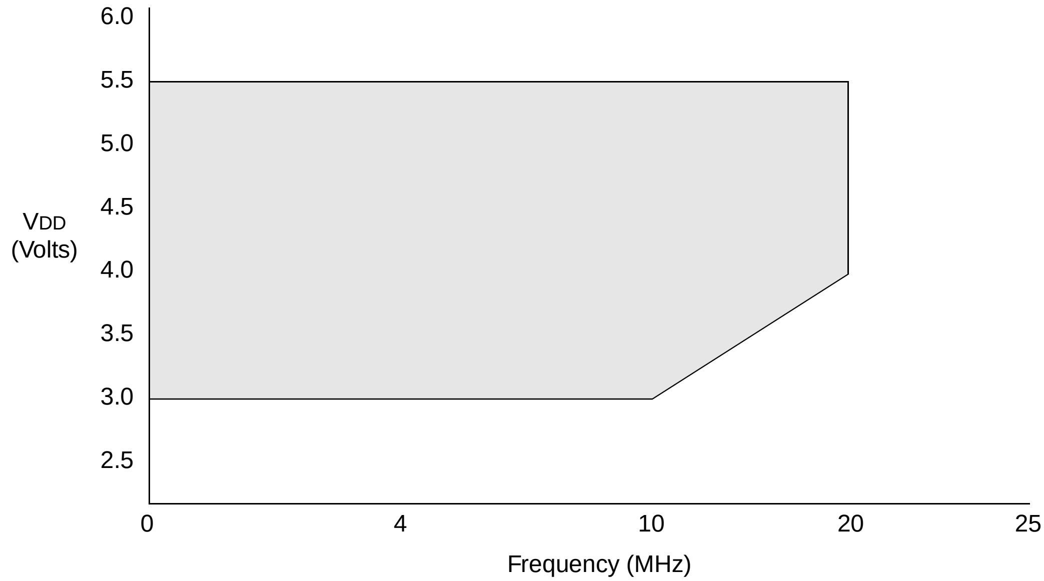
FIGURE 15-2: PIC16LC717/770/771 VOLTAGE-FREQUENCY GRAPH, 0°C ≤ T_A ≤ +70°C



Note 1: The shaded region indicates the permissible combinations of voltage and frequency.

 Advanced Information

FIGURE 15-3: PIC16LC717/770/771 VOLTAGE-FREQUENCY GRAPH, -40°C ≤ TA ≤ 0°C, +70°C ≤ TA ≤ +85°C



Note 1: The shaded region indicates the permissible combinations of voltage and frequency.

15.1 DC Characteristics: PIC16C717/770/771 (Commercial, Industrial)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial and 0°C ≤ TA ≤ +70°C for commercial				
Param No.	**Sym**	**Characteristic**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
D001	VDD	Supply Voltage	4.0	—	5.5	V	
D002*	VDR	RAM Data Retention Voltage(1)	—	1.5	—	V	
D003	VPOR	VDD start voltage to ensure internal Power-on Reset signal	—	VSS	—	V	See section on Power-on Reset for details
D004*	SVDD	VDD rise rate to ensure internal Power-on Reset signal	0.05	—	—	V/ms	See section on Power-on Reset for details. PWRT enabled
D010	IDD	Supply Current(2)			TBD TBD TBD TBD	mA mA mA mA	FOSC = 20 MHz, VDD = 5.5V* FOSC = 20 MHz, VDD = 4.0V FOSC = 4 MHz, VDD = 4.0V* FOSC = 32 KHz, VDD = 4.0V
D020 D020A	IPD	Power-down Current(3)	— — —	 1.5 1.5	TBD 16 19	mA µA µA	VDD = 5.5V, 0°C to +70°C* VDD = 4.0V, 0°C to +70°C VDD = 4.0V, -40°C to +85°C
		Module Differential Current(5)					
D021	ΔIWDT	Watchdog Timer	—	6.0	20	µA	VDD = 4.0V
D023B*	ΔIBG(6)	Bandgap voltage generator	—	40µA	TBD	µA	VDD = 4.0V
D025*	ΔIT1OSC	Timer1 oscillator	—	5	9	µA	VDD = 4.0V
D026*	ΔIAD	A/D Converter	—	300	—	µA	VDD = 5.5V, A/D on, not converting
	ΔILVD	Low Voltage Detect		10	TBD	µA	VDD = 4.0V*
	ΔIPBOR	Programmable Brown-Out Reset		10	TBD	µA	PBOR enabled, VDD = 5.0V*
	ΔIVRH	Voltage Reference High		70	TBD	µA	VDD = 5.0V, no load on VRH*
	ΔIVRL	Voltage Reference Low		70	TBD	µA	VDD = 4.0V, no load on VRL*
1A	Fosc	LP oscillator, operating freq. INTRC oscillator operating freq. ER oscillator operating freq. XT oscillator operating freq. HS oscillator operating freq.	9 — — TBD — 0 0	— 4 37 — 37 — —	200 — — TBD — 4 20	KHz MHz MHz MHz MHz MHz MHz	All temperatures All temperatures, OSCF = 1 All temperatures, OSCF = 0 All temperatures, OSCF = 1 All temperatures, OSCF = 0 All temperatures All temperatures

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: This is the limit to which VDD can be lowered without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption.
The test conditions for all IDD measurements in active operation mode are:
OSC1 = external square wave, from rail to rail; all I/O pins tristated, pulled to VDD
$\overline{\text{MCLR}}$ = VDD; WDT enabled/disabled as specified.

3: The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD or VSS.

4: For ER osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula Ir = TBD with Rext in kOhm.

5: The Δ current is the additional current consumed when the peripheral is enabled. This current should be added to the base (IPD or IDD) current.

6: The bandgap voltage reference provides 1.22V nominal to the VRL, VRH, LVD and BOR circuits. When calculating current consumption use the following formula: ΔIVRL + ΔIVRH + ΔILVD + ΔIBOR + ΔIBG. Any of the ΔIVRL, ΔIVRH, ΔILVD or ΔIBOR can be 0.

15.2 DC Characteristics: PIC16LC717/770/771 (Commercial, Industrial)

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ T_A ≤ +85°C for industrial and 0°C ≤ T_A ≤ +70°C for commercial				
Param No.	**Sym**	**Characteristic**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
D001	V_{DD}	Supply Voltage	2.5	—	5.5	V	
D002*	V_{DR}	RAM Data Retention Voltage(1)	—	1.5	—	V	
D003	V_{POR}	V_{DD} start voltage to ensure internal Power-on Reset signal	—	V_{SS}	—	V	See section on Power-on Reset for details
D004*	S_{VDD}	V_{DD} rise rate to ensure internal Power-on Reset signal	0.05	—	—	V/ms	See section on Power-on Reset for details. PWRT enabled
D010	Supply Current(2)	I_{DD}			TBD	mA	F_{OSC} = 20 MHz, V_{DD} = 5.5V*
					TBD	mA	F_{OSC} = 20 MHz, V_{DD} = 4.0V
					TBD	mA	F_{OSC} = 10 MHz, V_{DD} = 3.0V
					TBD	mA	F_{OSC} = 4 MHz, V_{DD} = 2.5V
					TBD	mA	F_{OSC} = 32 KHz, V_{DD} = 2.5V
D020 D020A	I_{PD}	Power-down Current(3)	—	TBD	TBD	μA	V_{DD} = 5.5V, 0°C to +70°C
			—	1.5	16	μA	V_{DD} = 4.0V, 0°C to +70°C
			—	1.5	19	μA	V_{DD} = 4.0V, -40°C to +85°C
			—	0.9	5	μA	V_{DD} = 2.5V, 0°C to +70°C
			—	0.9	5	μA	V_{DD} = 3.0V, -40°C to +85°C
		Module Differential Current(5)					
D021	ΔI_{WDT}	Watchdog Timer	—	6	20	μA	V_{DD} = 3.0V
D023B*	ΔI_{BG}	Bandgap voltage generator	—	40	TBD	μA	V_{DD} = 3.0V
D025*	ΔI_{T1OSC}	Timer1 oscillator	—	1.5	3	μA	V_{DD} = 3.0V
D026*	ΔI_{AD}	A/D Converter	—	300	—	μA	V_{DD} = 5.5V, A/D on, not converting
	ΔI_{LVD}	Low Voltage Detect		10	TBD	μA	V_{DD} = 4.0V*
	ΔI_{PBOR}	Programmable Brown-Out Reset		10	TBD	μA	PBOR enabled, V_{DD} = 5.0V*
	ΔI_{VRH}	Voltage Reference High		70	TBD	μA	V_{DD} = 5.0V, no load on VRH*
	ΔI_{VRL}	Voltage Reference Low		70	TBD	μA	V_{DD} = 4.0V, no load on VRL*
1A	Fosc	LP oscillator, operating freq.	9	—	200	KHz	All temperatures
		INTRC oscillator operating freq.	—	4	—	MHz	All temperatures, OSCF = 1
			—	37	—	MHz	All temperatures, OSCF = 0
		ER oscillator operating freq.	TBD	—	TBD	MHz	All temperatures, OSCF = 1
			—	37	—	MHz	All temperatures, OSCF = 0
		XT oscillator operating freq.	0	—	4	MHz	All temperatures
		HS oscillator operating freq.	0	—	20	MHz	All temperatures

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: This is the limit to which V_{DD} can be lowered without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as I/O pin loading and switching rate, oscillator type, internal code execution pattern and temperature also have an impact on the current consumption.
The test conditions for all I_{DD} measurements in active operation mode are:
$\overline{\text{OSC1}}$ = external square wave, from rail to rail; all I/O pins tristated, pulled to V_{DD}
$\overline{\text{MCLR}}$ = V_{DD}; WDT enabled/disabled as specified.

3: The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to V_{DD} or V_{SS}.

4: For ER osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula Ir = V_{DD}/2Rext (mA) with Rext in kOhm.

5: The Δ current is the additional current consumed when the peripheral is enabled. This current should be added to the base (I_{PD} or I_{DD}) current.

15.3 DC Characteristics: PIC16C717/770/771 & PIC16LC717/770/771 (Commercial, Industrial)

DC CHARACTERISTICS	Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ T_A ≤ +85°C for industrial and 0°C ≤ T_A ≤ +70°C for commercial Operating voltage V_{DD} range as described in DC spec Section 15.1 and Section 15.2.

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
		Input Low Voltage					
	V_{IL}	I/O ports					
D030		with TTL buffer	V_{SS}	—	0.15V_{DD}	V	For entire V_{DD} range
D030A			V_{SS}	—	0.8V	V	4.5V ≤ V_{DD} ≤ 5.5V
D031		with Schmitt Trigger buffer	V_{SS}	—	0.2V_{DD}	V	For entire V_{DD} range
D032		$\overline{MCLR}$	V_{SS}	—	0.2V_{DD}	V	
D033		OSC1 (in XT, HS, LP and EC)	V_{SS}	—	0.3V_{DD}	V	
		Input High Voltage					
	V_{IH}	I/O ports		—			
		with TTL buffer					
D040			2.0	—	V_{DD}	V	4.5V ≤ V_{DD} ≤ 5.5V
D040A			(0.25V_{DD} + 0.8V)	—	V_{DD}	V	For entire V_{DD} range
D041		with Schmitt Trigger buffer	0.8V_{DD}	—	V_{DD}	V	For entire V_{DD} range
D042		$\overline{MCLR}$	0.8V_{DD}	—	V_{DD}	V	
D042A		OSC1 (XT, HS, LP and EC)	0.7V_{DD}	—	V_{DD}	V	
D070	I_{PURB}	PORTB weak pull-up current per pin	50	250	400	μA	V_{DD} = 5V, V_{PIN} = V_{SS}
		Input Leakage Current (1,2)					
D060	I_{IL}	I/O ports (with digital functions)	—	—	±1	μA	Vss ≤ V_{PIN} ≤ V_{DD}, Pin at hi-impedance
D060A	I_{IL}	I/O ports (with analog functions)	—	—	±100	nA	Vss ≤ V_{PIN} ≤ V_{DD}, Pin at hi-impedance
D061		RA5/$\overline{MCLR}$/V_{PP}	—	—	±5	μA	Vss ≤ V_{PIN} ≤ V_{DD}
D063		OSC1	—	—	±5	μA	Vss ≤ V_{PIN} ≤ V_{DD}, XT, HS, LP and EC osc configuration
		Output Low Voltage					
D080	V_{OL}	I/O ports	—	—	0.6	V	I_{OL} = 8.5 mA, V_{DD} = 4.5V
		Output High Voltage					
D090	V_{OH}	I/O ports(2)	V_{DD} - 0.7	—	—	V	I_{OH} = -3.0 mA, V_{DD} = 4.5V
D150*	V_{OD}	**Open-Drain High Voltage**	—	—	10.5	V	RA4 pin
		Capacitive Loading Specs on Output Pins*					
D100	COSC2	OSC2 pin	—	—	15	pF	In XT, HS and LP modes when external clock is used to drive OSC1.
D101	C_{IO}	All I/O pins and OSC2 (in RC mode)	—	—	50	pF	
D102	C_B	SCL, SDA in I^2C mode	—	—	400	pF	
	C_{VRH}	V_{RH} pin	—	—	200	pF	V_{RH} output enabled
	C_{VRL}	V_{RL} pin	—	—	200	pF	V_{RL} output enabled

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: The leakage current on the $\overline{MCLR}$ pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

2: Negative current is defined as current sourced by the pin.

15.4 AC Characteristics: PIC16C717/770/771 & PIC16LC717/770/771 (Commercial, Industrial)

15.4.1 TIMING PARAMETER SYMBOLOGY

The timing parameter symbols have been created following one of the following formats:

1. TppS2ppS		3. TCC:ST	(I^2C specifications only)
2. TppS		4. Ts	(I^2C specifications only)

T			
F	Frequency	T	Time

Lowercase letters (pp) and their meanings:

pp			
cc	CCP1	osc	OSC1
ck	CLKOUT	rd	$\overline{\text{RD}}$
cs	$\overline{\text{CS}}$	rw	$\overline{\text{RD}}$ or $\overline{\text{WR}}$
di	SDI	sc	SCK
do	SDO	ss	$\overline{\text{SS}}$
dt	Data in	t0	T0CKI
io	I/O port	t1	T1CKI
mc	$\overline{\text{MCLR}}$	wr	$\overline{\text{WR}}$

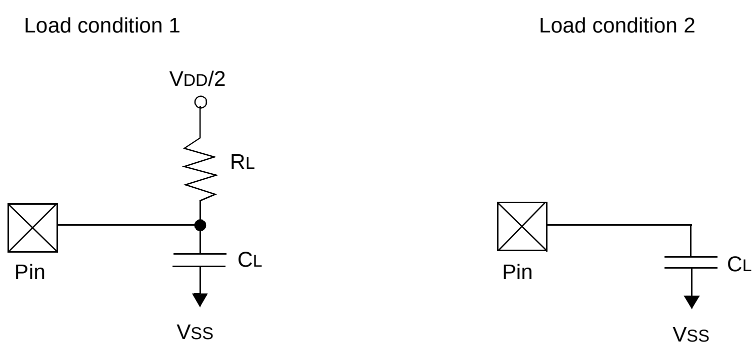
Uppercase letters and their meanings:

S			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (Hi-impedance)	V	Valid
L	Low	Z	Hi-impedance
I^2C only			
AA	output access	High	High
BUF	Bus free	Low	Low

TCC:ST (I^2C specifications only)

CC			
HD	Hold	SU	Setup
ST			
DAT	DATA input hold	STO	STOP condition
STA	START condition		

FIGURE 15-4: LOAD CONDITIONS



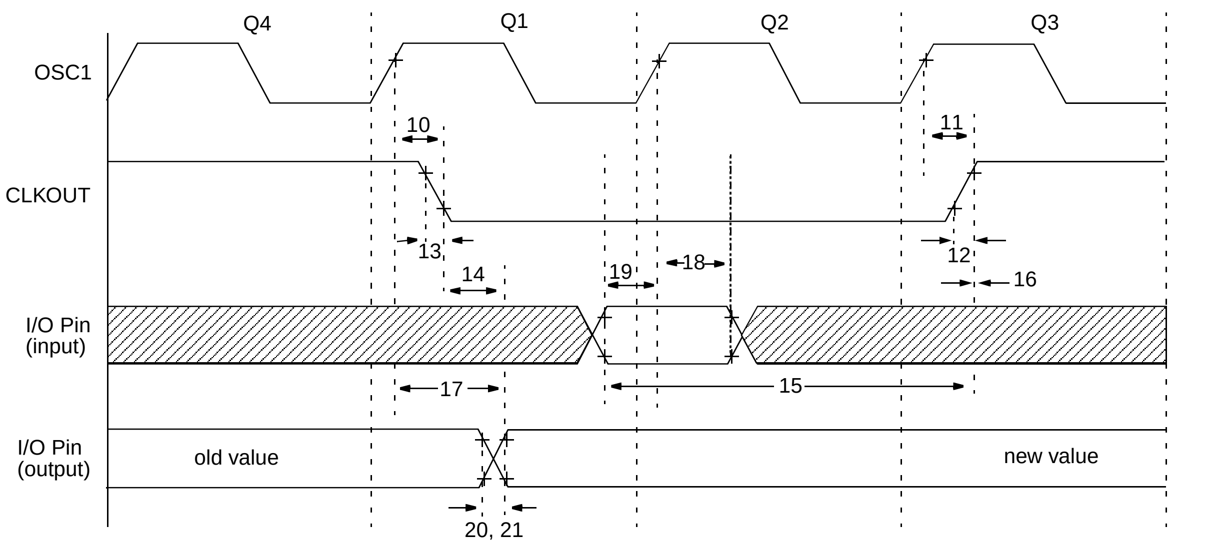
R_L = 464Ω

C_L = 50 pF for all pins except OSC2

15 pF for OSC2 output

15.4.2 TIMING DIAGRAMS AND SPECIFICATIONS

FIGURE 15-5: CLKOUT AND I/O TIMING



Note: Refer to Figure 15-4 for load conditions.

TABLE 15-1: CLKOUT AND I/O TIMING REQUIREMENTS

Parameter No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
10*	TosH2ckL	OSC1↑ to CLKOUT↓		—	75	200	ns	Note 1
11*	TosH2ckH	OSC1↑ to CLKOUT↑		—	75	200	ns	Note 1
12*	TckR	CLKOUT rise time		—	35	100	ns	Note 1
13*	TckF	CLKOUT fall time		—	35	100	ns	Note 1
14*	TckL2ioV	CLKOUT ↓ to Port out valid		—	—	0.5TCY + 20	ns	Note 1
15*	TioV2ckH	Port in valid before CLKOUT ↑		0.25TCY + 25	—	—	ns	Note 1
16*	TckH2ioI	Port in hold after CLKOUT ↑		0	—	—	ns	Note 1
17*	TosH2ioV	OSC1↑ (Q1 cycle) to Port out valid		—	50	150	ns	
18*	TosH2ioI	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	PIC16**C**717/770/771	100	—	—	ns	
			PIC16**LC**717/770/771	200	—	—	ns	
19*	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)		0	—	—	ns	
20*	TioR	Port output rise time	PIC16**C**717/770/771	—	10	25	ns	
			PIC16**LC**717/770/771	—	—	60	ns	
21*	TioF	Port output fall time	PIC16**C**717/770/771	—	10	25	ns	
			PIC16**LC**717/770/771	—	—	60	ns	
22††*	Tinp	INT pin high or low time		TCY	—	—	ns	
23††*	Trbp	RB7:RB0 change INT high or low time		TCY	—	—	ns	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

†† These parameters are asynchronous events not related to any internal clock edges.

Note 1: Measurements are taken in RC Mode where CLKOUT output is 4 x TOSC.

 Advanced Information

FIGURE 15-6: EXTERNAL CLOCK TIMING

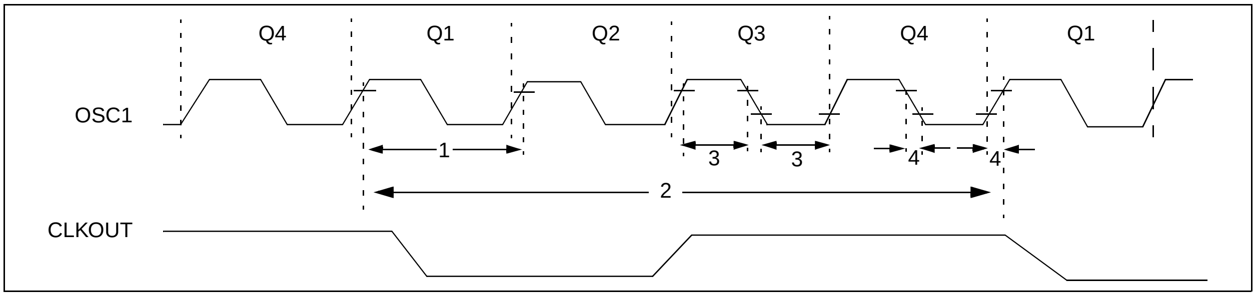


TABLE 15-2: EXTERNAL CLOCK TIMING REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
1A	FOSC	**External CLKIN Frequency (Note 1)**	DC	—	4	MHz	XT osc mode
			DC	—	20	MHz	EC osc mode
			DC	—	20	MHz	HS osc mode
			DC	—	200	kHz	LP osc mode
		Oscillator Frequency (Note 1)	0.1	—	4	MHz	XT osc mode
			4	—	20	MHz	HS osc mode
			5	—	200	kHz	LP osc mode
1	TOSC	**External CLKIN Period (Note 1)**	250	—	—	ns	XT and RC osc mode
			50	—	—	ns	EC osc mode
			50	—	—	ns	HS osc mode
			5	—	—	µs	LP osc mode
		Oscillator Period (Note 1)	250	—	10,000	ns	XT osc mode
			50	—	250	ns	HS osc mode
			5	—	—	µs	LP osc mode
2	TCY	**Instruction Cycle Time (Note 1)**	200	TCY	DC	ns	TCY = 4/FOSC
3*	TosL, TosH	**External Clock in (OSC1) High or Low Time**	100	—	—	ns	XT oscillator
			2.5	—	—	µs	LP oscillator
			15	—	—	ns	HS oscillator
							EC oscillator
4*	TosR, TosF	**External Clock in (OSC1) Rise or Fall Time**	—	—	25	ns	XT oscillator
			—	—	50	ns	LP oscillator
			—	—	15	ns	HS oscillator
							EC oscillator

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: Instruction cycle period (TCY) equals four times the input oscillator time-base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min." values with an external clock applied to the OSC1/CLKIN pin. When an external clock input is used, the "Max." cycle time limit is "DC" (no clock) for all devices.

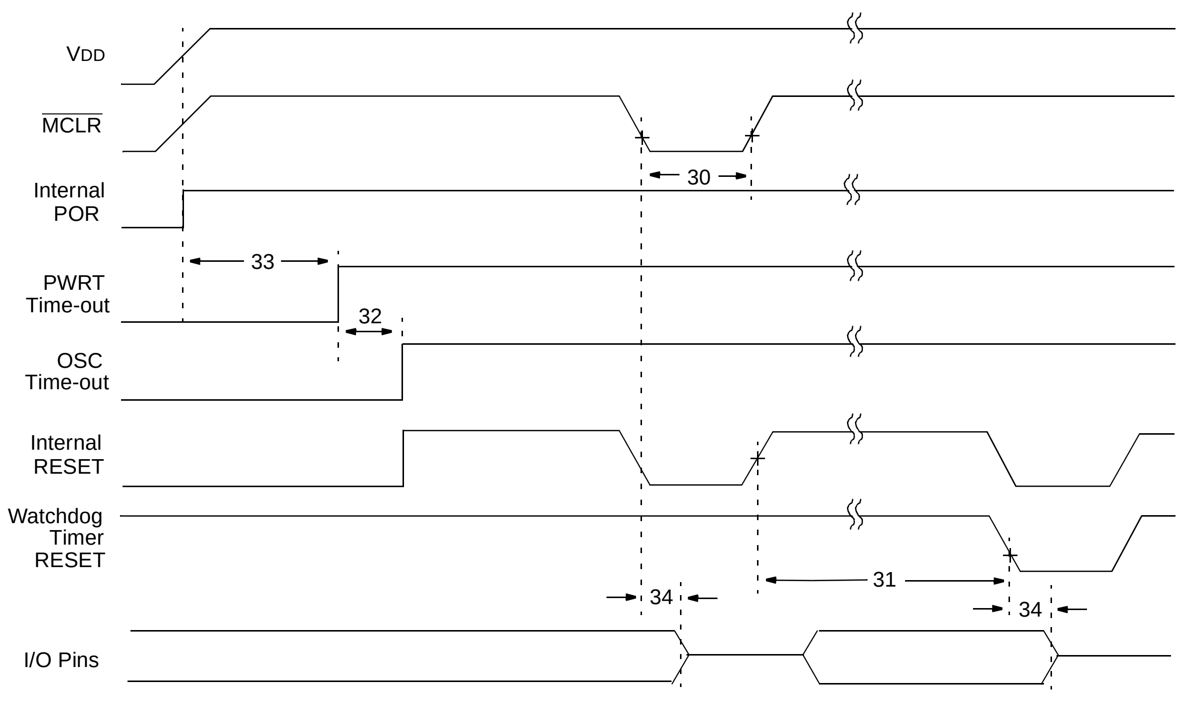
TABLE 15-3: CALIBRATED INTERNAL RC FREQUENCIES - PIC16C717/770/771 AND PIC16LC717/770/771

AC Characteristics	Standard Operating Conditions (unless otherwise specified) Operating Temperature 0°C ≤ TA ≤ +70°C (commercial), –40°C ≤ TA ≤ +85°C (industrial), Operating Voltage VDD range is described in Section 15.1 and Section 15.2					
Parameter No.	**Sym**	**Characteristic**	**Min***	**Typ(1)**	**Max***	**Units**	**Conditions**
		Internal Calibrated RC Frequency	3.65	4.00	4.28	MHz	VDD = 5.0V
		Internal Calibrated RC Frequency	3.55	4.00	4.31	MHz	VDD = 2.5V

* These parameters are characterized but not tested.

Note 1: Data in the Typical ("Typ") column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 15-7: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING



Note: Refer to Figure 15-4 for load conditions.

FIGURE 15-8: BROWN-OUT RESET TIMING

TABLE 15-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER, AND BROWN-OUT RESET REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30*	TMCL	$\overline{MCLR}$ Pulse Width (low)	2	—	—	µs	VDD = 5V, -40°C to +85°C
31*	TWDT	Watchdog Timer Time-out Period (No Prescaler)	7	18	33	ms	VDD = 5V, -40°C to +85°C
32*	TOST	Oscillation Start-up Timer Period	—	1024 TOSC	—	—	TOSC = OSC1 period
33*	TPWRT	Power up Timer Period	28	72	132	ms	VDD = 5V, -40°C to +85°C
34*	TIOZ	I/O Hi-impedance from $\overline{MCLR}$ Low or Watchdog Timer Reset	—	—	2.1	µs	
35*	TBOR	Brown-out Reset pulse width	100	—	—	µs	VDD ≤ VBOR (D005)

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 15-9: BROWN-OUT RESET CHARACTERISTICS

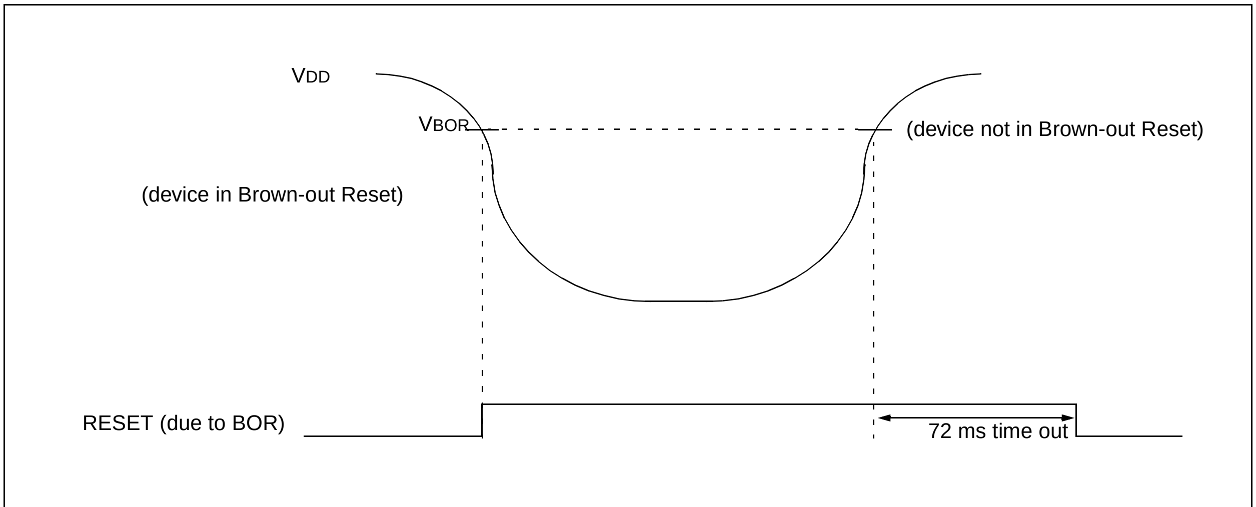
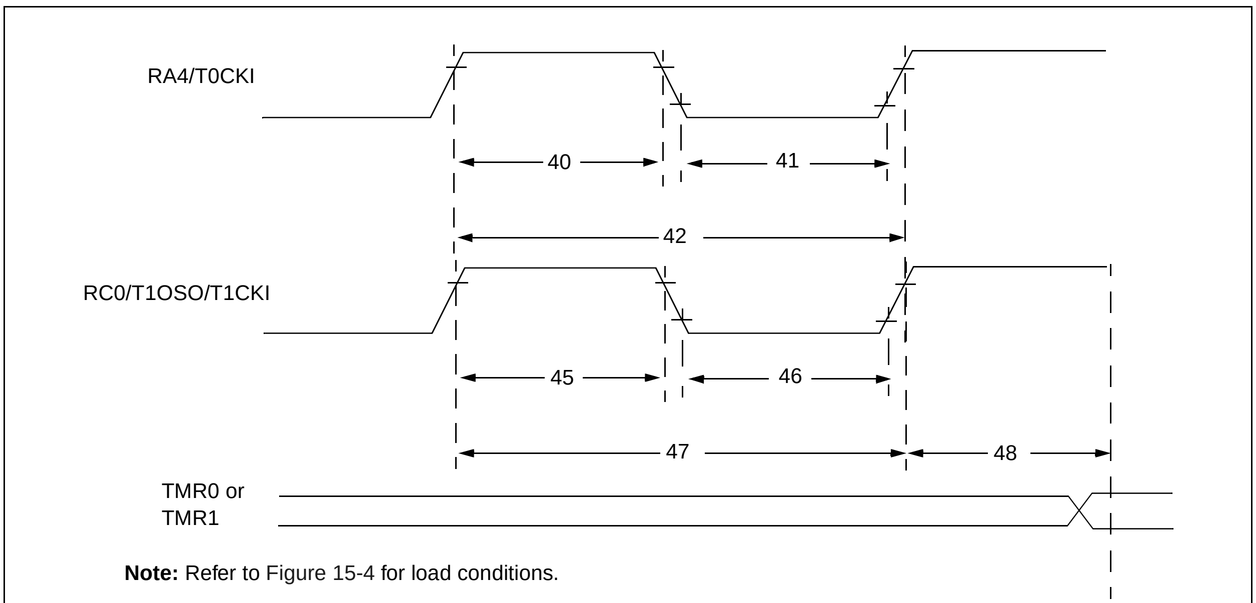


FIGURE 15-10: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS



Note: Refer to Figure 15-4 for load conditions.

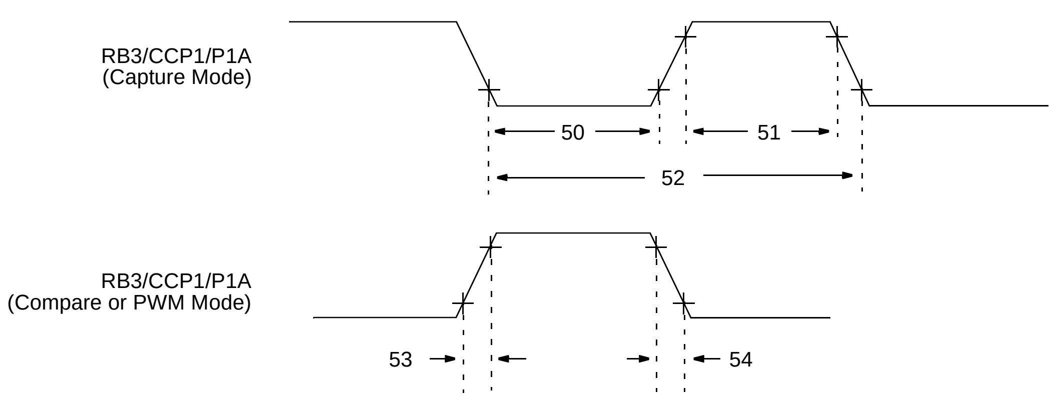
TABLE 15-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

Param No.	Sym	Characteristic			Min	Typ†	Max	Units	Conditions
40*	Tt0H	T0CKI High Pulse Width	No Prescaler		0.5TCY + 20	—	—	ns	Must also meet parameter 42
			With Prescaler		10	—	—	ns	
41*	Tt0L	T0CKI Low Pulse Width	No Prescaler		0.5TCY + 20	—	—	ns	Must also meet parameter 42
			With Prescaler		10	—	—	ns	
42*	Tt0P	T0CKI Period	No Prescaler		TCY + 40	—	—	ns	
			With Prescaler		Greater of: 20 or $\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (2, 4, ..., 256)
45*	Tt1H	T1CKI High Time	Synchronous, Prescaler = 1		0.5TCY + 20	—	—	ns	Must also meet parameter 47
			Synchronous, Prescaler = 2,4,8	PIC16**C**717/770/771	15	—	—	ns	
				PIC16**LC**717/770/771	25	—	—	ns	
			Asynchronous	PIC16**C**717/770/771	30	—	—	ns	
				PIC16**LC**717/770/771	50	—	—	ns	
46*	Tt1L	T1CKI Low Time	Synchronous, Prescaler = 1		0.5TCY + 20	—	—	ns	Must also meet parameter 47
			Synchronous, Prescaler = 2,4,8	PIC16**C**717/770/771	15	—	—	ns	
				PIC16**LC**717/770/771	25	—	—	ns	
			Asynchronous	PIC16**C**717/770/771	30	—	—	ns	
				PIC16**LC**717/770/771	50	—	—	ns	
47*	Tt1P	T1CKI input period	Synchronous	PIC16**C**717/770/771	Greater of: 30 OR $\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1, 2, 4, 8)
				PIC16**LC**717/770/771	Greater of: 50 OR $\frac{T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1, 2, 4, 8)
			Asynchronous	PIC16**C**717/770/771	60	—	—	ns	
				PIC16**LC**717/770/771	100	—	—	ns	
	Ft1	Timer1 oscillator input frequency range (oscillator enabled by setting bit T1OSCEN)			DC	—	50	kHz	
48	Tcke2tmr1	Delay from external clock edge to timer increment			2Tosc	—	7Tosc	—	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

FIGURE 15-11: ENHANCED CAPTURE/COMPARE/PWM TIMINGS (ECCP1)



Note: Refer to Figure 15-4 for load conditions.

TABLE 15-6: ENHANCED CAPTURE/COMPARE/PWM REQUIREMENTS (ECCP1)

Param No.	Sym	Characteristic			Min	Typ†	Max	Units	Conditions
50*	TccL	CCP1 input low time	No Prescaler		0.5TCY + 20	—	—	ns	
			With Prescaler	PIC16**C**717/770/771	10	—	—	ns	
				PIC16**LC**717/770/771	20	—	—	ns	
51*	TccH	CCP1 input high time	No Prescaler		0.5TCY + 20	—	—	ns	
			With Prescaler	PIC16**C**717/770/771	10	—	—	ns	
				PIC16**LC**717/770/771	20	—	—	ns	
52*	TccP	CCP1 input period			$\frac{3T_{CY} + 40}{N}$	—	—	ns	N = prescale value (1,4 or 16)
53*	TccR	CCP1 output fall time		PIC16**C**717/770/771	—	10	25	ns	
				PIC16**LC**717/770/771	—	25	45	ns	
54*	TccF	CCP1 output fall time		PIC16**C**717/770/771	—	10	25	ns	
				PIC16**LC**717/770/771	—	25	45	ns	

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

15.5 Analog Peripherals Characteristics: PIC16C717/770/771 & PIC16LC717/770/771 (Commercial, Industrial)

15.5.1 BANDGAP MODULE

FIGURE 15-12: BANDGAP START-UP TIME

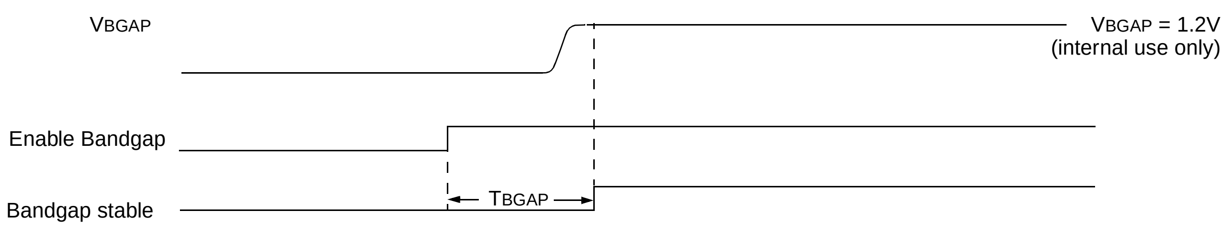


TABLE 15-7: BANDGAP START-UP TIME

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
36*	TBGAP	Bandgap start-up time	—	30	TBD	μS	Defined as the time between the instant that the bandgap is enabled and the moment that the bandgap reference voltage is stable.

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

 Advanced Information

15.5.2 LOW VOLTAGE DETECT MODULE (LVD)

TABLE 15-8: LOW-VOLTAGE DETECT CHARACTERISTICS

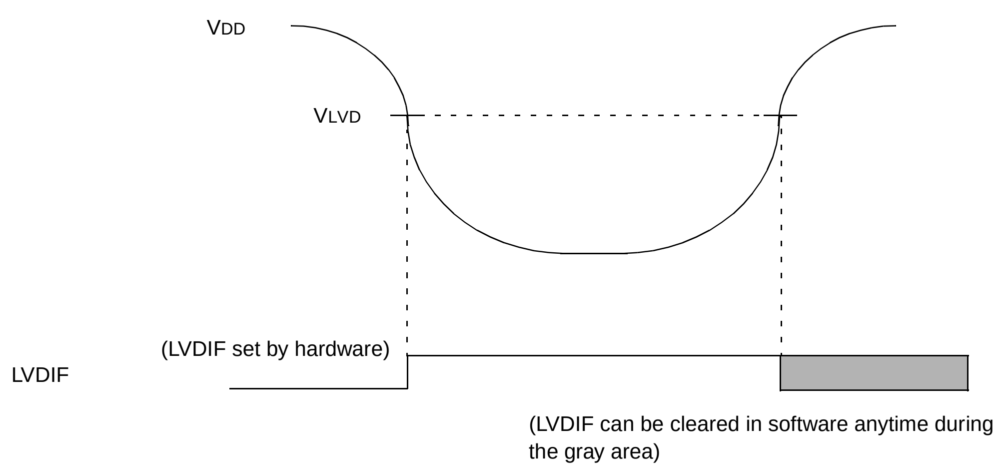


TABLE 15-9: ELECTRICAL CHARACTERISTICS: LVD

DC CHARACTERISTICS		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ T_A ≤ +85°C for industrial and 0°C ≤ T_A ≤ +70°C for commercial Operating voltage V_{DD} range as described in DC spec Section 15.1 and Section 15.2.						
Param No.	**Characteristic**		**Symbol**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
D420*	LVD Voltage	LVV = 0100		2.5	2.58	2.66	V	
		LVV = 0101		2.7	2.78	2.86	V	
		LVV = 0110		2.8	2.89	2.98	V	
		LVV = 0111		3.0	3.1	3.2	V	
		LVV = 1000		3.3	3.41	3.52	V	
		LVV = 1001		3.5	3.61	3.72	V	
		LVV = 1010		3.6	3.72	3.84	V	
		LVV = 1011		3.8	3.92	4.04	V	
		LVV = 1100		4.0	4.13	4.26	V	
		LVV = 1101		4.2	4.33	4.46	V	
		LVV = 1110		4.5	4.64	4.78	V	
D422*	LVD Voltage Temperature coefficient		TCV_{OUT}	—	15	50	ppm/°C	
D423*	LVD Voltage Supply Regulation		$\Delta V_{LVD}/\Delta V_{DD}$	—	—	50	µV/V	

* These parameters are characterized but not tested.

Note 1: Production tested at Tamb = 25°C. Specifications over temperature limits ensured by characterization.

15.5.3 PROGRAMMABLE BROWN-OUT RESET MODULE (PBOR)

TABLE 15-10: DC CHARACTERISTICS: PBOR

DC CHARACTERISTICS		Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial and 0°C ≤ TA ≤ +70°C for commercial Operating voltage VDD range as described in DC spec Section 15.1 and Section 15.2.						
Param No.	**Characteristic**		**Symbol**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
D005*	BOR Voltage	BORV<1:0> = 11	VBOR	2.5	2.58	2.66	V	
		BORV<1:0> = 10		2.7	2.78	2.86		
		BORV<1:0> = 01		4.2	4.33	4.46		
		BORV<1:0> = 00		4.5	4.64	4.78		
D006*	BOR Voltage Temperature coefficient		TCVOUT	—	15	50	ppm/°C	
D006A*	BOR Voltage Supply Regulation		ΔVBOR/ ΔVDD	—	—	50	μV/V	

* These parameters are characterized but not tested.

15.5.4 VREF MODULE

TABLE 15-11: DC CHARACTERISTICS: VREF

DC CHARACTERISTICS			Standard Operating Conditions (unless otherwise stated) Operating temperature -40°C ≤ TA ≤ +85°C for industrial and 0°C ≤ TA ≤ +70°C for commercial Operating voltage VDD range as described in DC spec Section 15.1 and Section 15.2.				
Param No.	**Symbol**	**Characteristic**	**Min**	**Typ†**	**Max**	**Units**	**Conditions**
D400	VRL	Output Voltage	2.0	2.048	2.1	V	VDD ≥ 2.5V
	VRH		4.0	4.096	4.2	V	VDD ≥ 4.5V
D402*	TCVOUT	Output Voltage Temperature coefficient	—	15	50	ppm/°C	
D404*	IVREFSO	External Load Source	—	—	5	mA	
D405*	IVREFSI	External Load Sink	—	—	-5	mA	
*	CL	External capacitor load	—	—	200	pF	
D406*	ΔVOUT/ ΔIOUT	Load Regulation	—	1	TBD	mV/mA	Isource = 0 mA to 5 mA
			—	1	TBD		Isink = 0 mA to 5 mA
D407*	ΔVOUT/ ΔVDD	Supply Regulation	—	—	50	μV/V	

* These parameters are characterized but not tested.

† Data in “Typ” column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

15.5.5 A/D CONVERTER MODULE

TABLE 15-12: PIC16C770/771 AND PIC16LC770/771 A/D CONVERTER CHARACTERISTICS:

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
A01	NR	Resolution	—	—	12 bits	bit	Min. resolution for A/D is 1 mV, VREF+ = AVDD = 4.096V, VREF- = AVSS = 0V, VREF- ≤ VAIN ≤ VREF+
A03	EIL	Integral error	—	—	TBD	—	VREF+ = AVDD = 4.096V, VREF- = AVSS = 0V, VREF- ≤ VAIN ≤ VREF+
A04	EDL	Differential error	—	—	TBD	—	No missing codes to 10 bits VREF+ = AVDD = 4.096V, VREF- = AVSS = 0V, VREF- ≤ VAIN ≤ VREF+
A06	EOFF	Offset error	—	—	TBD	—	VREF+ = AVDD = 4.096V, VREF- = AVSS = 0V, VREF- ≤ VAIN ≤ VREF+
A07	EGN	Gain Error	—	—	TBD	LSb	VREF+ = AVDD = 4.096V, VREF- = AVSS = 0V, VREF- ≤ VAIN ≤ VREF+
A10	—	Monotonicity	—	guaranteed(3)	—	—	AVSS ≤ VAIN ≤ VREF+
A20	VREF	Reference voltage (VREF+ VREF-)	4.096	—	VDD +0.3V	V	Absolute minimum electrical spec to ensure 12-bit accuracy.
A21	VREF+	Reference V High (AVDD or VREF+)	VREF-	—	AVDD	V	Min. resolution for A/D is 1 mV
A22	VREF-	Reference V Low (AVSS or VREF-)	AVSS	—	VREF+	V	Min. resolution for A/D is 1 mV
A25	VAIN	Analog input voltage	VREFL	—	VREFH	V	
A30	ZAIN	Recommended impedance of analog voltage source	—	—	2.5	kΩ	
A50	IREF	VREF input current (Note 2)	—	—	10	μA	During VAIN acquisition. Based on differential of VHOLD to VAIN. To charge CHOLD see Section 11.0. During A/D conversion cycle.

* These parameters are characterized but not tested.

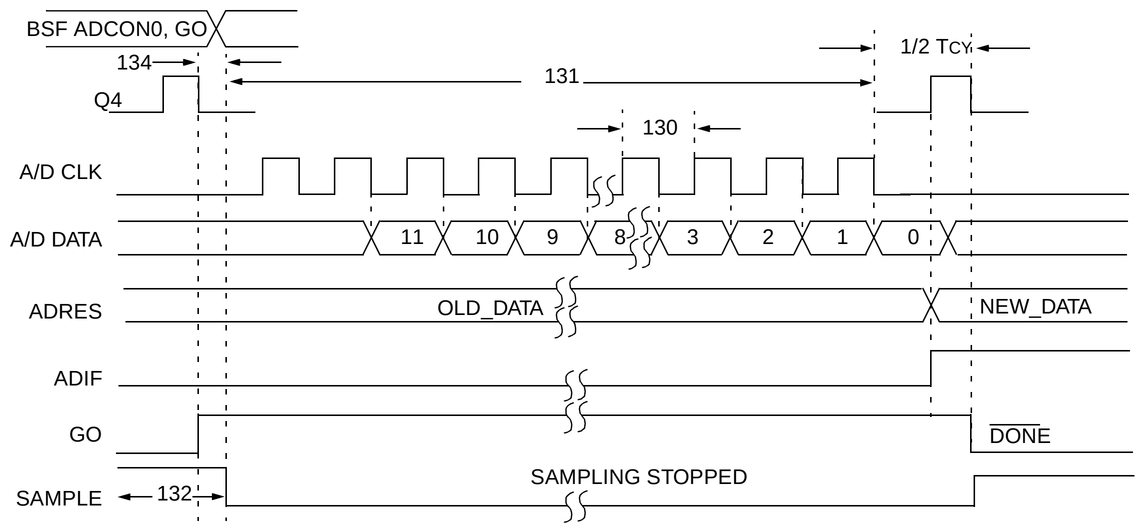
† Data in “Typ” column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: When A/D is off, it will not consume any current other than minor leakage current. The power down current spec includes any such leakage from the A/D module.

2: VREF input current is from External VREF+, or VREF-, or AVSS, or AVDD pin, whichever is selected as reference input.

3: The A/D conversion result never decreases with an increase in the input voltage and has no missing codes.

FIGURE 15-13: PIC16C770/771 AND PIC16LC770/771 A/D CONVERSION TIMING (NORMAL MODE)



Note 1: If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

TABLE 15-13: PIC16C770/771 AND PIC16LC770/771 A/D CONVERSION REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
130*	TAD	A/D clock period	1.6	—	—	µs	Tosc based, VREF ≥ 2.5V
			3.0	—	—	µs	Tosc based, VREF full range
130*	TAD	A/D Internal RC oscillator period					ADCS<1:0> = 11 (RC mode)
			3.0	6.0	9.0	µs	At VDD = 2.5V
			2.0	4.0	6.0	µs	At VDD = 5.0V
131*	TCNV	Conversion time (not including acquisition time) (Note 1)	—	13TAD	—	TAD	Set GO bit to new data in A/D result register
132*	TACQ	Acquisition Time	Note 2	11.5	—	µs	
			5*	—	—	µs	The minimum time is the amplifier settling time. This may be used if the “new” input voltage has not changed by more than 1LSb (i.e 1mV @ 4.096V) from the last sampled voltage (as stated on CHOLD).
134*	TGO	Q4 to A/D clock start	—	TOSC/2	—	—	If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

* These parameters are characterized but not tested.

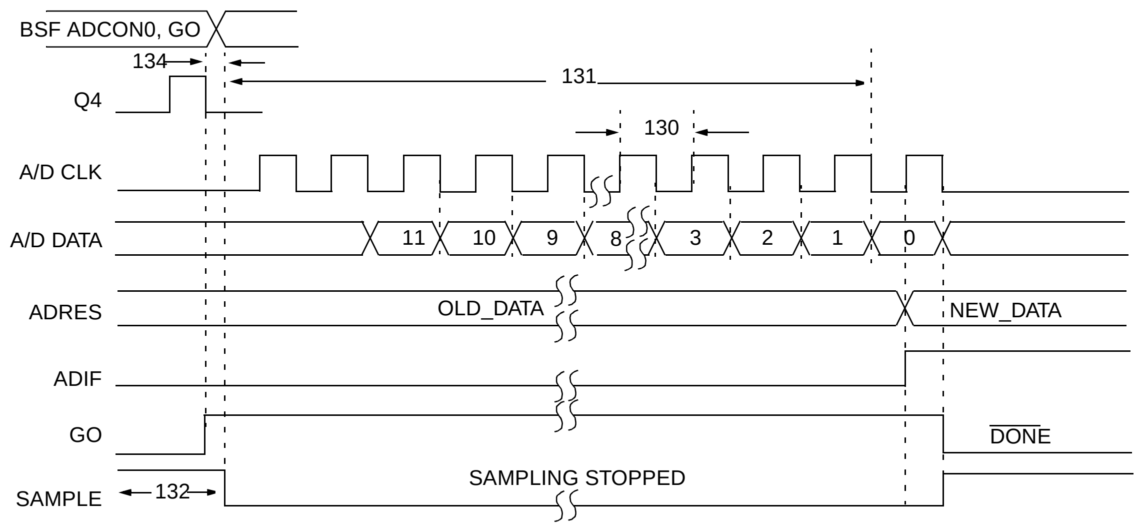
† Data in “Typ” column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: ADRES register may be read on the following TCY cycle.

2: See Section 11.6 for minimum conditions.

 Advanced Information

FIGURE 15-14: PIC16C770/771 AND PIC16LC770/771 A/D CONVERSION TIMING (SLEEP MODE)



Note 1: If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

TABLE 15-14: PIC16C770/771 AND PIC16LC770/771 A/D CONVERSION REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
130*	TAD	A/D clock period	1.6	—	—	µs	VREF ≥ 2.5V
			TBD	—	—	µs	VREF full range
130*	TAD	A/D Internal RC oscillator period					ADCS<1:0> = 11 (RC mode)
			3.0	6.0	9.0	µs	At VDD = 3.0V
			2.0	4.0	6.0	µs	At VDD = 5.0V
131*	TCNV	Conversion time (not including acquisition time) (Note 1)	—	13TAD	—	—	
132*	TACQ	Acquisition Time	Note 2	11.5	—	µs	
			5*	—	—	µs	The minimum time is the amplifier settling time. This may be used if the “new” input voltage has not changed by more than 1LSb (i.e 1mV @ 4.096V) from the last sampled voltage (as stated on CHOLD).
134*	TGO	Q4 to A/D clock start	—	TOSC/2 + TCY	—	—	If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

* These parameters are characterized but not tested.

† Data in “Typ” column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: ADRES register may be read on the following TCY cycle.

2: See Section 11.6 for minimum conditions.

TABLE 15-15: PIC16C717 AND PIC16LC717 A/D CONVERTER CHARACTERISTICS:

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
A01	N_R	Resolution	—	—	10 bits	bit	Min. resolution for A/D is 4.1 mV, $V_{REF+} = AV_{DD} = 4.096V$, $V_{REF-} = AV_{SS} = 0V$, $V_{REF-} \leq V_{AIN} \leq V_{REF+}$
A03	E_{IL}	Integral error	—	—	TBD	—	$V_{REF+} = AV_{DD} = 4.096V$, $V_{REF-} = AV_{SS} = 0V$, $V_{REF-} \leq V_{AIN} \leq V_{REF+}$
A04	E_{DL}	Differential error	—	—	TBD	—	No missing codes to 10 bits $V_{REF+} = AV_{DD} = 4.096V$, $V_{REF-} = AV_{SS} = 0V$, $V_{REF-} \leq V_{AIN} \leq V_{REF+}$
A06	E_{OFF}	Offset error	—	—	TBD	—	$V_{REF+} = AV_{DD} = 4.096V$, $V_{REF-} = AV_{SS} = 0V$, $V_{REF-} \leq V_{AIN} \leq V_{REF+}$
A07	E_{GN}	Gain Error	—	—	TBD	LSb	$V_{REF+} = AV_{DD} = 4.096V$, $V_{REF-} = AV_{SS} = 0V$, $V_{REF-} \leq V_{AIN} \leq V_{REF+}$
A10	—	Monotonicity	—	guaranteed(3)	—	—	$AV_{SS} \leq V_{AIN} \leq V_{REF+}$
A20	V_{REF}	Reference voltage (V_{REF+} V_{REF-})	4.096	—	V_{DD} +0.3V	V	Absolute minimum electrical spec to ensure 10-bit accuracy.
A21	V_{REF+}	Reference V High (AV_{DD} or V_{REF+})	V_{REF-}	—	AV_{DD}	V	Min. resolution for A/D is 4.1 mV
A22	V_{REF-}	Reference V Low (AV_{SS} or V_{REF-})	AV_{SS}	—	V_{REF+}	V	Min. resolution for A/D is 4.1 mV
A25	V_{AIN}	Analog input voltage	V_{REFL}	—	V_{REFH}	V	
A30	Z_{AIN}	Recommended impedance of analog voltage source	—	—	2.5	kΩ	
A50	I_{REF}	V_{REF} input current (Note 2)	—	—	10	μA	During V_{AIN} acquisition. Based on differential of V_{HOLD} to V_{AIN}. To charge C_{HOLD} see Section 11.0. During A/D conversion cycle.

* These parameters are characterized but not tested.

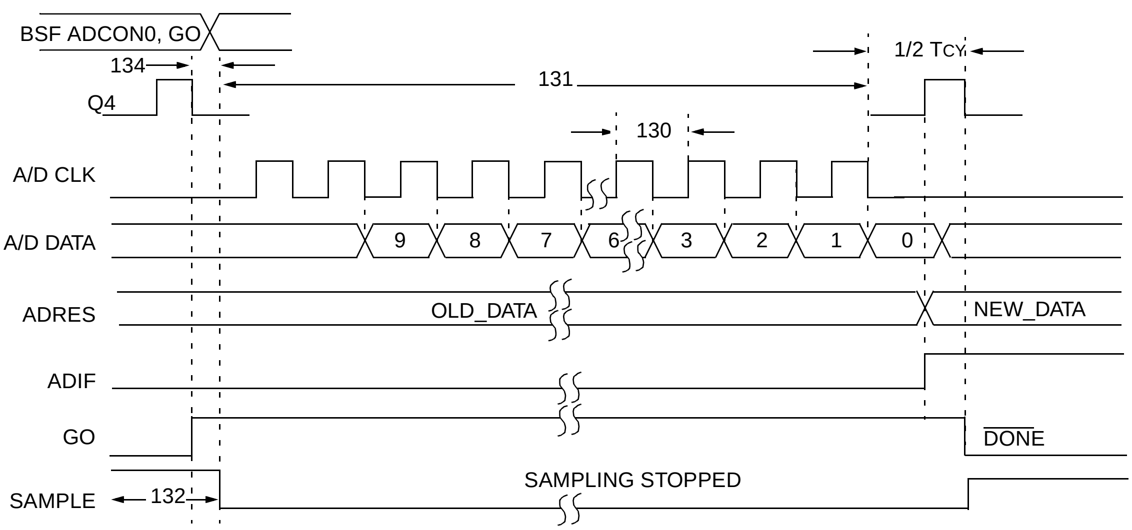
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: When A/D is off, it will not consume any current other than leakage current. The power down current spec includes any such leakage from the A/D module.

2: V_{REF} current is from External V_{REF+}, or V_{REF-}, or AV_{SS}, or AV_{DD} pin, whichever is selected as reference input.

3: The A/D conversion result never decreases with an increase in the input voltage and has no missing codes.

FIGURE 15-15: PIC16C717 A/D CONVERSION TIMING (NORMAL MODE)



Note 1: If the A/D clock source is selected as RC, a time of T_{CY} is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

TABLE 15-16: PIC16C717 AND PIC16LC717 A/D CONVERSION REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
130*	T_{AD}	A/D clock period	1.6	—	—	µs	Tosc based, $V_{REF} \geq 2.5V$
			3.0	—	—	µs	Tosc based, V_{REF} full range
130*	T_{AD}	A/D Internal RC oscillator period					ADCS<1:0> = 11 (RC mode)
			3.0	6.0	9.0	µs	At V_{DD} = 2.5V
			2.0	4.0	6.0	µs	At V_{DD} = 5.0V
131*	T_{CNV}	Conversion time (not including acquisition time) (Note 1)	—	11T_{AD}	—	T_{AD}	Set GO bit to new data in A/D result register
132*	T_{ACQ}	Acquisition Time	Note 2	11.5	—	µs	
			5*	—	—	µs	The minimum time is the amplifier settling time. This may be used if the “new” input voltage has not changed by more than 1LSb (i.e 1mV @ 4.096V) from the last sampled voltage (as stated on C_{HOLD}).
134*	T_{GO}	Q4 to A/D clock start	—	$T_{OSC}/2$	—	—	If the A/D clock source is selected as RC, a time of T_{CY} is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

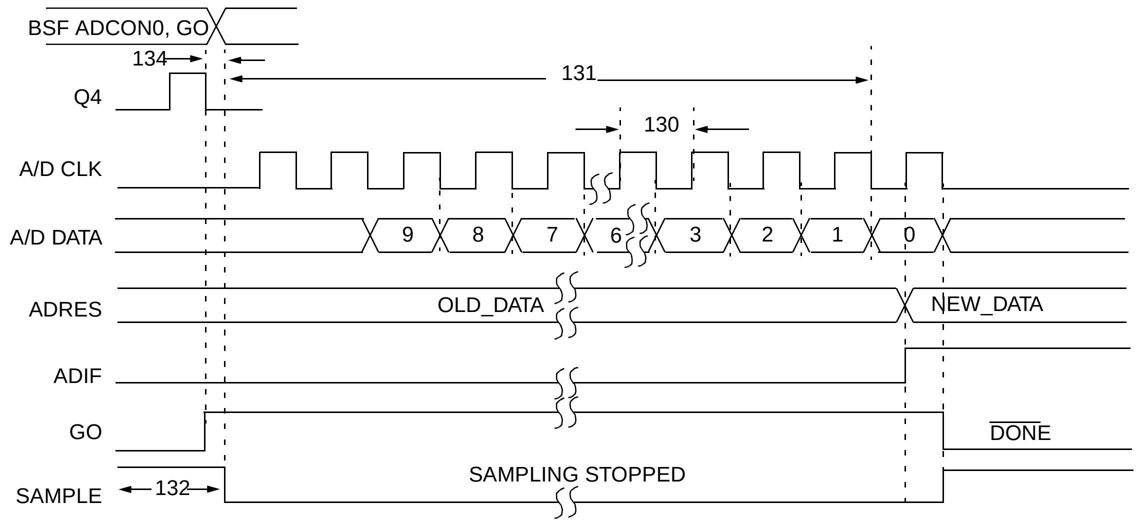
* These parameters are characterized but not tested.

† Data in “Typ” column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: ADRES register may be read on the following T_{CY} cycle.

2: See Section 11.6 for minimum conditions.

FIGURE 15-16: PIC16C717 A/D CONVERSION TIMING (SLEEP MODE)



Note 1: If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

TABLE 15-17: PIC16C717 AND PIC16LC717 A/D CONVERSION REQUIREMENTS

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
130*	TAD	A/D clock period	1.6	—	—	µs	VREF ≥ 2.5V
			TBD	—	—	µs	VREF full range
130*	TAD	A/D Internal RC oscillator period					ADCS<1:0> = 11 (RC mode)
			3.0	6.0	9.0	µs	At VDD = 3.0V
			2.0	4.0	6.0	µs	At VDD = 5.0V
131*	TCNV	Conversion time (not including acquisition time) (Note 1)	—	11TAD	—	—	
132*	TACQ	Acquisition Time	Note 2	11.5	—	µs	
			5*	—	—	µs	The minimum time is the amplifier settling time. This may be used if the “new” input voltage has not changed by more than 1LSb (i.e 1mV @ 4.096V) from the last sampled voltage (as stated on CHOLD).
134*	TGO	Q4 to A/D clock start	—	TOSC/2 + TCY	—	—	If the A/D clock source is selected as RC, a time of TCY is added before the A/D clock starts. This allows the SLEEP instruction to be executed.

* These parameters are characterized but not tested.

† Data in “Typ” column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: ADRES register may be read on the following TCY cycle.

2: See Section 11.6 for minimum conditions.

 Advanced Information

16.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES

The graphs and tables provided in this section are for **design guidance** and are **not tested**.

In some graphs or tables, the data presented are **outside specified operating range** (i.e., outside specified V_{DD} range). This is for **information only**.

The data presented in this section is a **statistical summary** of data collected on units from different lots over a period of time and matrix samples. 'Typical' represents the mean of the distribution at 25°C. 'Max' or 'min' represents (mean + 3σ) or (mean - 3σ) respectively, where σ is standard deviation, over the whole temperature range.

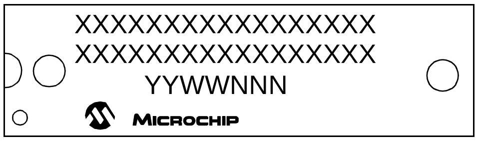
Graphs and Tables not available at this time.

NOTES:

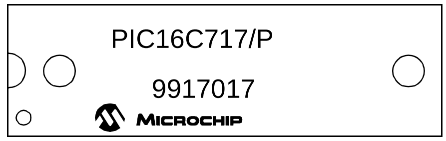
17.0 PACKAGING INFORMATION

17.1 Package Marking Information

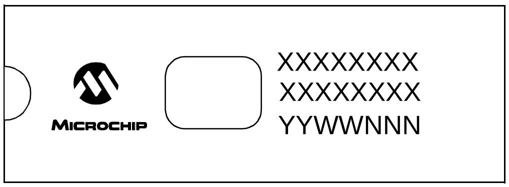
18-Lead PDIP



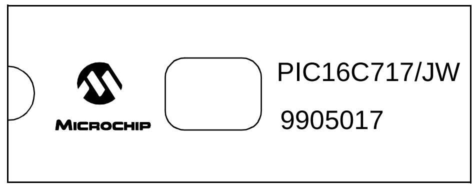
Example



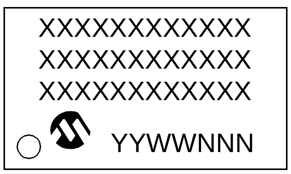
18-Lead CERDIP Windowed



Example

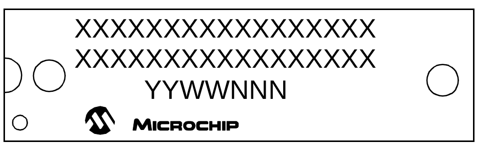


18-Lead SOIC

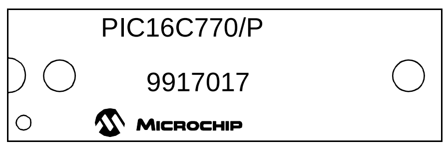


Example

20-Lead PDIP



Example



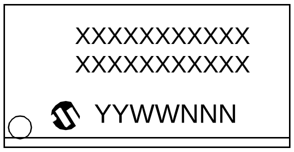
Legend:	MM...M	Microchip part number information
	XX...X	Customer specific information*
	YY	Year code (last 2 digits of calendar year)
	WW	Week code (week of January 1 is week '01')
	NNN	Alphanumeric traceability code

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.

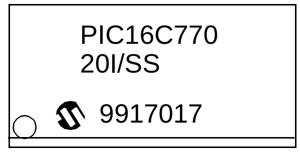
* Standard OTP marking consists of Microchip part number, year code, week code, facility code, mask rev#, and assembly code. For OTP marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

Package Marking Information (Cont'd)

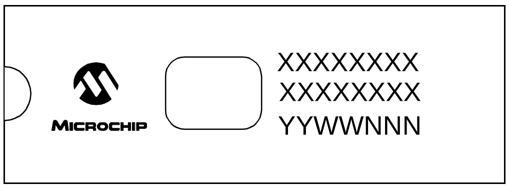
20-Lead SSOP



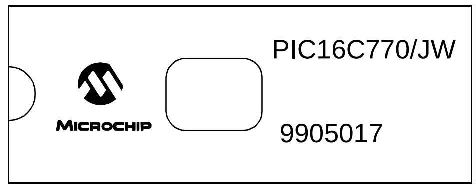
Example



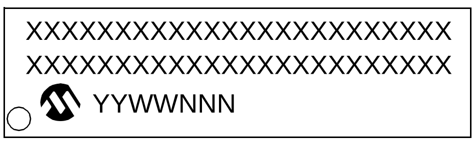
20-Lead CERDIP Windowed



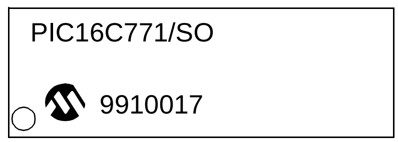
Example



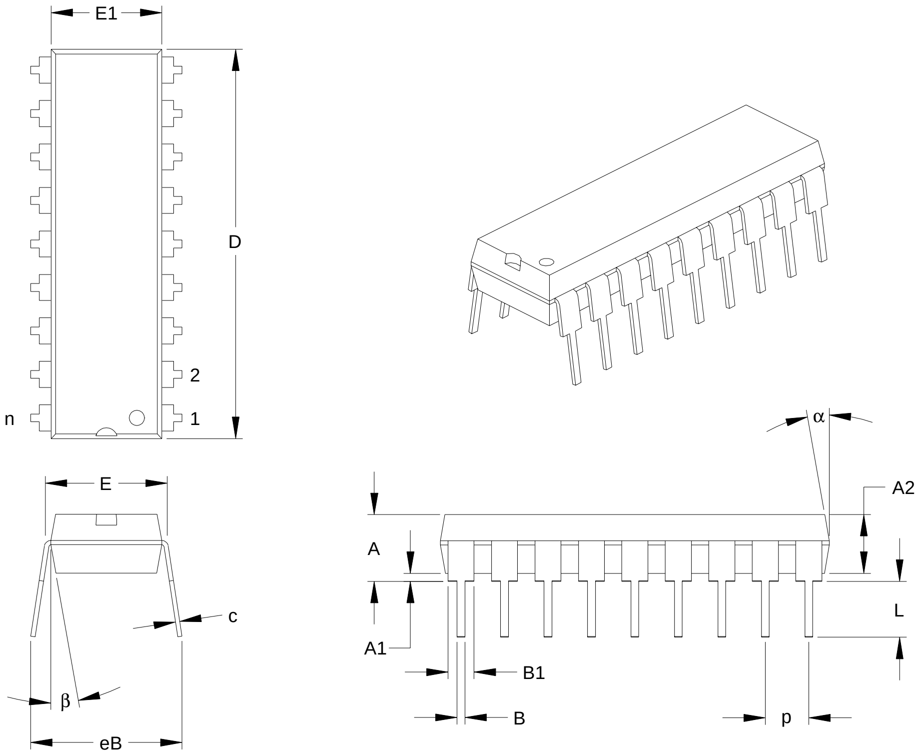
20-Lead SOIC



Example



17.2 18-Lead Plastic Dual In-line (P) – 300 mil (PDIP)



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		18			18	
Pitch	p		.100			2.54	
Top to Seating Plane	A	.140	.155	.170	3.56	3.94	4.32
Molded Package Thickness	A2	.115	.130	.145	2.92	3.30	3.68
Base to Seating Plane	A1	.015			0.38		
Shoulder to Shoulder Width	E	.300	.313	.325	7.62	7.94	8.26
Molded Package Width	E1	.240	.250	.260	6.10	6.35	6.60
Overall Length	D	.890	.898	.905	22.61	22.80	22.99
Tip to Seating Plane	L	.125	.130	.135	3.18	3.30	3.43
Lead Thickness	c	.008	.012	.015	0.20	0.29	0.38
Upper Lead Width	B1	.045	.058	.070	1.14	1.46	1.78
Lower Lead Width	B	.014	.018	.022	0.36	0.46	0.56
Overall Row Spacing	eB	.310	.370	.430	7.87	9.40	10.92
Mold Draft Angle Top	α	5	10	15	5	10	15
Mold Draft Angle Bottom	β	5	10	15	5	10	15

*Controlling Parameter

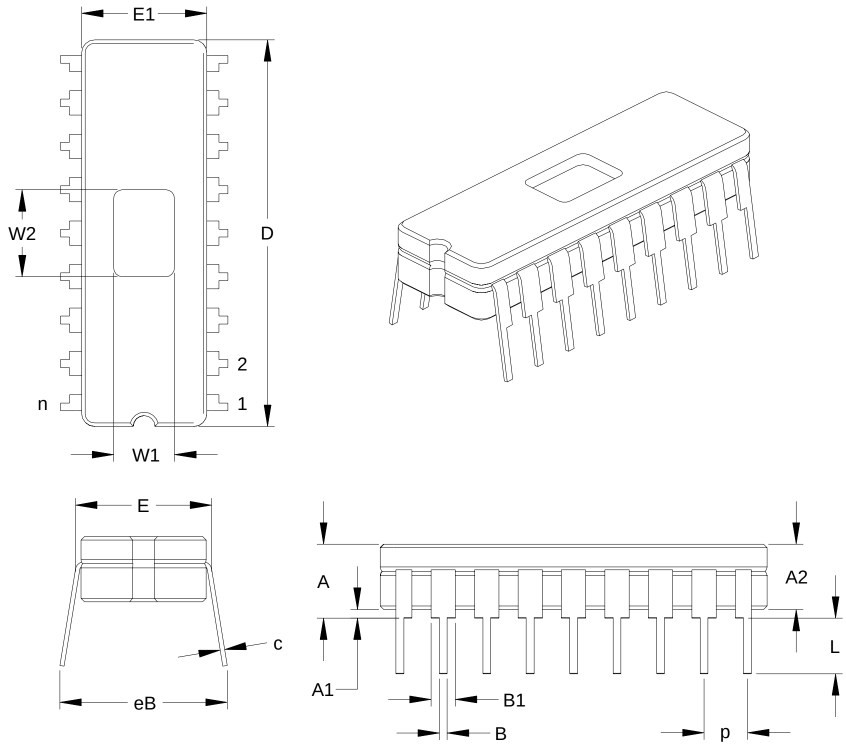
Notes:

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: MS-001

Drawing No. C04-007

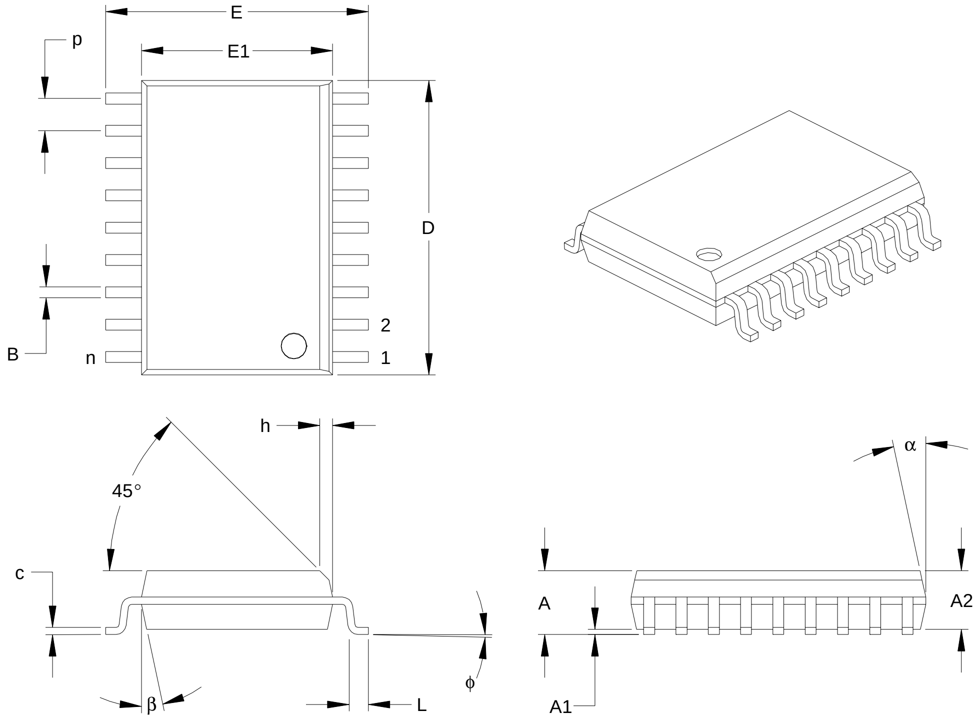
17.3 18-Lead Ceramic Dual In-line with Window (JW) – 300 mil (CERDIP)



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		18			18	
Pitch	p		.100			2.54	
Top to Seating Plane	A	.170	.183	.195	4.32	4.64	4.95
Ceramic Package Height	A2	.155	.160	.165	3.94	4.06	4.19
Standoff	A1	.015	.023	.030	0.38	0.57	0.76
Shoulder to Shoulder Width	E	.300	.313	.325	7.62	7.94	8.26
Ceramic Pkg. Width	E1	.285	.290	.295	7.24	7.37	7.49
Overall Length	D	.880	.900	.920	22.35	22.86	23.37
Tip to Seating Plane	L	.125	.138	.150	3.18	3.49	3.81
Lead Thickness	c	.008	.010	.012	0.20	0.25	0.30
Upper Lead Width	B1	.050	.055	.060	1.27	1.40	1.52
Lower Lead Width	B	.016	.019	.021	0.41	0.47	0.53
Overall Row Spacing	eB	.345	.385	.425	8.76	9.78	10.80
Window Width	W1	.130	.140	.150	3.30	3.56	3.81
Window Length	W2	.190	.200	.210	4.83	5.08	5.33

*Controlling Parameter
JEDEC Equivalent: MO-036
Drawing No. C04-010

17.4 18-Lead Plastic Small Outline (SO) – Wide, 300 mil (SOIC)



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		18			18	
Pitch	p		.050			1.27	
Overall Height	A	.093	.099	.104	2.36	2.50	2.64
Molded Package Thickness	A2	.088	.091	.094	2.24	2.31	2.39
Standoff	A1	.004	.008	.012	0.10	0.20	0.30
Overall Width	E	.394	.407	.420	10.01	10.34	10.67
Molded Package Width	E1	.291	.295	.299	7.39	7.49	7.59
Overall Length	D	.446	.454	.462	11.33	11.53	11.73
Chamfer Distance	h	.010	.020	.029	0.25	0.50	0.74
Foot Length	L	.016	.033	.050	0.41	0.84	1.27
Foot Angle	φ	0	4	8	0	4	8
Lead Thickness	c	.009	.011	.012	0.23	0.27	0.30
Lead Width	B	.014	.017	.020	0.36	0.42	0.51
Mold Draft Angle Top	α	0	12	15	0	12	15
Mold Draft Angle Bottom	β	0	12	15	0	12	15

*Controlling Parameter

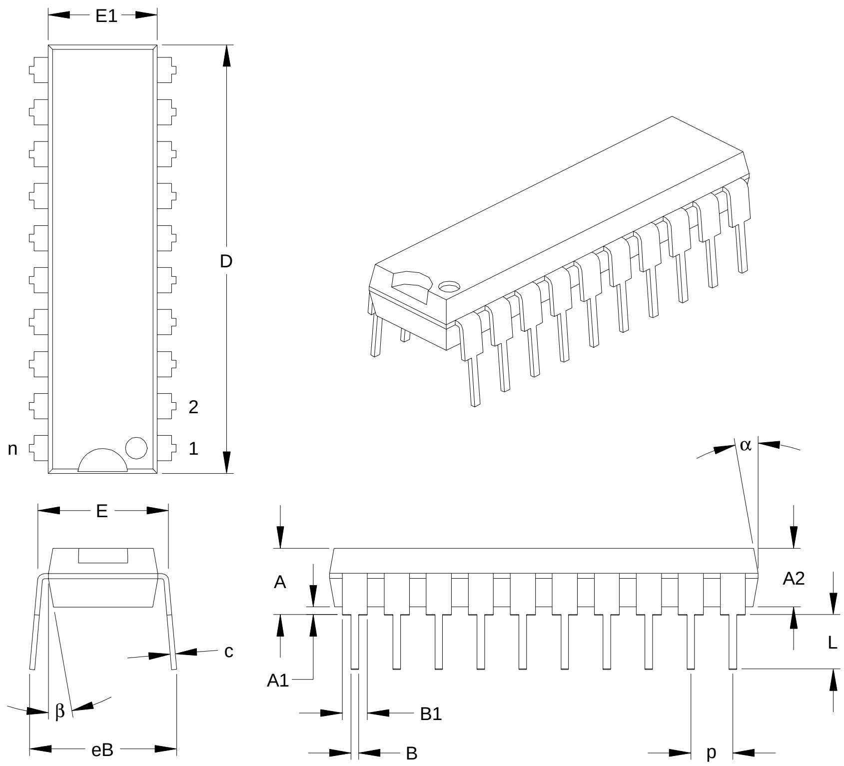
Notes:

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: MS-013

Drawing No. C04-051

17.5 20-Lead Plastic Dual In-line (P) – 300 mil (PDIP)



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		20			20	
Pitch	p		.100			2.54	
Top to Seating Plane	A	.140	.155	.170	3.56	3.94	4.32
Molded Package Thickness	A2	.115	.130	.145	2.92	3.30	3.68
Base to Seating Plane	A1	.015			0.38		
Shoulder to Shoulder Width	E	.295	.310	.325	7.49	7.87	8.26
Molded Package Width	E1	.240	.250	.260	6.10	6.35	6.60
Overall Length	D	1.025	1.033	1.040	26.04	26.24	26.42
Tip to Seating Plane	L	.120	.130	.140	3.05	3.30	3.56
Lead Thickness	c	.008	.012	.015	0.20	0.29	0.38
Upper Lead Width	B1	.055	.060	.065	1.40	1.52	1.65
Lower Lead Width	B	.014	.018	.022	0.36	0.46	0.56
Overall Row Spacing	eB	.310	.370	.430	7.87	9.40	10.92
Mold Draft Angle Top	α	5	10	15	5	10	15
Mold Draft Angle Bottom	β	5	10	15	5	10	15

*Controlling Parameter

Notes:

Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010” (0.254mm) per side.

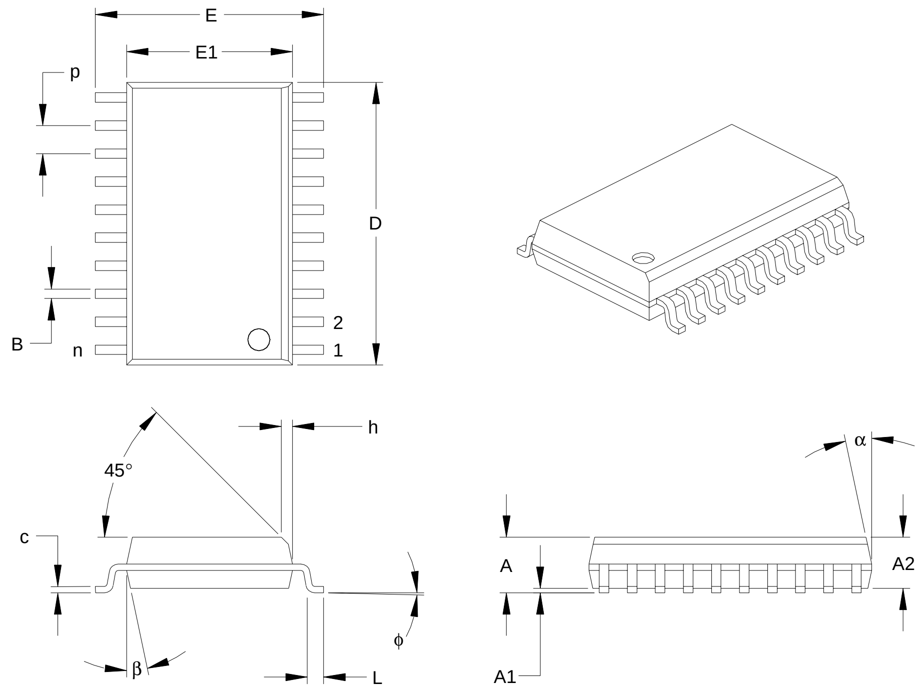
JEDEC Equivalent: MS-001

Drawing No. C04-019

17.6 20-Lead Ceramic Dual In-line with Window (JW) – 300 mil (CERDIP)

DRAWING NOT AVAILABLE

17.7 20-Lead Plastic Small Outline (SO) – Wide, 300 mi (SOIC)



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		20			20	
Pitch	p		.050			1.27	
Overall Height	A	.093	.099	.104	2.36	2.50	2.64
Molded Package Thickness	A2	.088	.091	.094	2.24	2.31	2.39
Standoff	A1	.004	.008	.012	0.10	0.20	0.30
Overall Width	E	.394	.407	.420	10.01	10.34	10.67
Molded Package Width	E1	.291	.295	.299	7.39	7.49	7.59
Overall Length	D	.496	.504	.512	12.60	12.80	13.00
Chamfer Distance	h	.010	.020	.029	0.25	0.50	0.74
Foot Length	L	.016	.033	.050	0.41	0.84	1.27
Foot Angle	φ	0	4	8	0	4	8
Lead Thickness	c	.009	.011	.013	0.23	0.28	0.33
Lead Width	B	.014	.017	.020	0.36	0.42	0.51
Mold Draft Angle Top	α	0	12	15	0	12	15
Mold Draft Angle Bottom	β	0	12	15	0	12	15

*Controlling Parameter

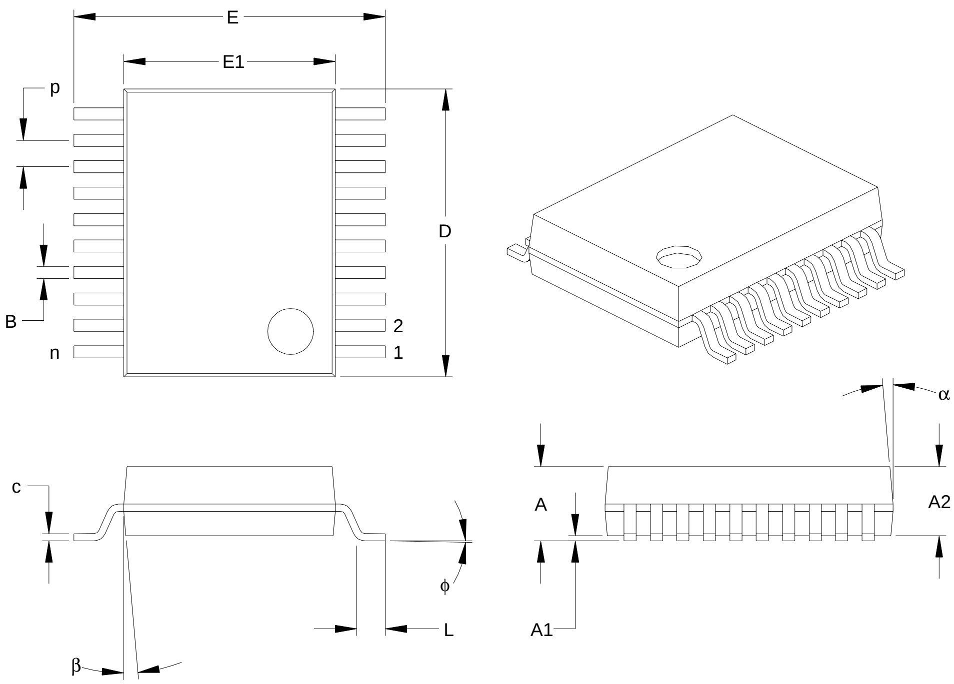
Notes:

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: MS-013

Drawing No. C04-094

17.8 20-Lead Plastic Shrink Small Outline (SS) – 209 mil, 5.30 mm (SSOP)



Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		20			20	
Pitch	p		.026			0.66	
Overall Height	A	.068	.073	.078	1.73	1.85	1.98
Molded Package Thickness	A2	.064	.068	.072	1.63	1.73	1.83
Standoff	A1	.002	.006	.010	0.05	0.15	0.25
Overall Width	E	.299	.309	.322	7.59	7.85	8.18
Molded Package Width	E1	.201	.207	.212	5.11	5.25	5.38
Overall Length	D	.278	.284	.289	7.06	7.20	7.34
Foot Length	L	.022	.030	.037	0.56	0.75	0.94
Lead Thickness	c	.004	.007	.010	0.10	0.18	0.25
Foot Angle	φ	0	4	8	0.00	101.60	203.20
Lead Width	B	.010	.013	.015	0.25	0.32	0.38
Mold Draft Angle Top	α	0	5	10	0	5	10
Mold Draft Angle Bottom	β	0	5	10	0	5	10

*Controlling Parameter

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-150

Drawing No. C04-072

NOTES:

APPENDIX A: REVISION HISTORY

Version	Date	Revision Description
A	9/16/99	This is a new data sheet. However, the devices described in this data sheet are the upgrades to the devices found in the *PIC16C7X Data Sheet*, DS30390E.

APPENDIX B: DEVICE DIFFERENCES

The differences between the devices in this data sheet are listed in Table B-1.

TABLE B-1: DEVICE DIFFERENCES

Difference	PIC16C717	PIC16C770	PIC16C771
Program Memory	2K	2K	4K
A/D	6 channels, 10 bits	6 channels, 12 bits	6 channels, 12 bits
Dedicated AV_{DD} and AV_{SS}	Not available	Available	Available
Packages	18-pin PDIP, 18-pin windowed CERDIP, 18-pin SOIC, 20-pin SSOP	20-pin PDIP, 20-pin windowed CERDIP, 20-pin SOIC, 20-pin SSOP	20-pin PDIP, 20-pin windowed CERDIP, 20-pin SOIC, 20-pin SSOP

NOTES:

INDEX

 Advanced Information

 Advanced Information

NOTES:

ON-LINE SUPPORT

Microchip provides on-line support on the Microchip World Wide Web (WWW) site.

The web site is used by Microchip as a means to make files and information easily available to customers. To view the site, the user must have access to the Internet and a web browser, such as Netscape or Microsoft Explorer. Files are also available for FTP download from our FTP site.

Connecting to the Microchip Internet Web Site

The Microchip web site is available by using your favorite Internet browser to attach to:

www.microchip.com

The file transfer site is available by using an FTP service to connect to:

ftp://ftp.microchip.com

The web site and file transfer site provide a variety of services. Users may download files for the latest Development Tools, Data Sheets, Application Notes, User's Guides, Articles and Sample Programs. A variety of Microchip specific business information is also available, including listings of Microchip sales offices, distributors and factory representatives. Other data available for consideration is:

- Latest Microchip Press Releases
- Technical Support Section with Frequently Asked Questions
- Design Tips
- Device Errata
- Job Postings
- Microchip Consultant Program Member Listing
- Links to other useful web sites related to Microchip Products
- Conferences for products, Development Systems, technical information and more
- Listing of seminars and events

Systems Information and Upgrade Hot Line

The Systems Information and Upgrade Line provides system users a listing of the latest versions of all of Microchip's development systems software products. Plus, this line provides information on how customers can receive any currently available upgrade kits.The Hot Line Numbers are:

1-800-755-2345 for U.S. and most of Canada, and

1-480-786-7302 for the rest of the world.

981103

READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 786-7578.

Please list the following information, and use this outline to provide us with your comments about this Data Sheet.

To: Technical Publications Manager   Total Pages Sent

RE: Reader Response

From: Name ____________________

Company ____________________

Address ____________________

City / State / ZIP / Country ____________________

Telephone: (_______) _________ - _________   FAX: (______) _________ - _________

Application (optional):

Would you like a reply? ___Y ___N

Device: **PIC16C717/770/771**   Literature Number: **DS41120A**

Questions:

1. What are the best features of this document?

2. How does this document meet your hardware and software development needs?

3. Do you find the organization of this data sheet easy to follow? If not, why?

4. What additions to the data sheet do you think would enhance the structure and subject?

5. What deletions from the data sheet could be made without affecting the overall usefulness?

6. Is there any incorrect or misleading information (what and where)?

7. How would you improve this document?

8. How would you improve our software, systems, and silicon products?

PIC16C717/770/771 PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery refer to the factory or the listed sales office.

PART NO. - X /XX XXX

Pattern:	QTP, SQTP, Code or Special Requirements
Package:	JW = Windowed CERDIP SO = SOIC P = PDIP SS = SSOP
Temperature Range:	- = 0°C to +70°C I = -40°C to +85°C
Device	PIC16C771 : V_{DD} range 4.0V to 5.5V PIC16C771T : V_{DD} range 4.0V to 5.5V (Tape/Reel) PIC16LC771 : V_{DD} range 2.5V to 5.5V PIC16LC771T: V_{DD} range 2.5V to 5.5V (Tape/Reel)

Examples

a) PIC16C771/P Commercial Temp., PDIP Package, normal V_{DD} limits.

* JW Devices are UV erasable and can be programmed to any device configuration. JW Devices meet the electrical requirement of each oscillator type (including LC devices).

Sales and Support

Data Sheets

Products supported by a preliminary Data Sheet may have an errata sheet describing minor operational differences and recommended workarounds. To determine if an errata sheet exists for a particular device, please contact one of the following:

1. Your local Microchip sales office
2. The Microchip Corporate Literature Center U.S. FAX: (480) 786-7277
3. The Microchip Worldwide Site (www.microchip.com)

Please specify which device, revision of silicon and Data Sheet (include Literature #) you are using.

New Customer Notification System

Register on our web site (www.microchip.com/cn) to receive the most current information on our products.

WORLDWIDE SALES AND SERVICE

AMERICAS

Corporate Office
Microchip Technology Inc.
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-786-7200 Fax: 480-786-7277
Technical Support: 480-786-7627
Web Address: http://www.microchip.com

Atlanta
Microchip Technology Inc.
500 Sugar Mill Road, Suite 200B
Atlanta, GA 30350
Tel: 770-640-0034 Fax: 770-640-0307

Boston
Microchip Technology Inc.
5 Mount Royal Avenue
Marlborough, MA 01752
Tel: 508-480-9990 Fax: 508-480-8575

Chicago
Microchip Technology Inc.
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 630-285-0071 Fax: 630-285-0075

Dallas
Microchip Technology Inc.
4570 Westgrove Drive, Suite 160
Addison, TX 75248
Tel: 972-818-7423 Fax: 972-818-2924

Dayton
Microchip Technology Inc.
Two Prestige Place, Suite 150
Miamisburg, OH 45342
Tel: 937-291-1654 Fax: 937-291-9175

Detroit
Microchip Technology Inc.
Tri-Atria Office Building
32255 Northwestern Highway, Suite 190
Farmington Hills, MI 48334
Tel: 248-538-2250 Fax: 248-538-2260

Los Angeles
Microchip Technology Inc.
18201 Von Karman, Suite 1090
Irvine, CA 92612
Tel: 949-263-1888 Fax: 949-263-1338

New York
Microchip Technology Inc.
150 Motor Parkway, Suite 202
Hauppauge, NY 11788
Tel: 631-273-5305 Fax: 631-273-5335

San Jose
Microchip Technology Inc.
2107 North First Street, Suite 590
San Jose, CA 95131
Tel: 408-436-7950 Fax: 408-436-7955

AMERICAS (continued)

Toronto
Microchip Technology Inc.
5925 Airport Road, Suite 200
Mississauga, Ontario L4V 1W1, Canada
Tel: 905-405-6279 Fax: 905-405-6253

ASIA/PACIFIC

Hong Kong
Microchip Asia Pacific
Unit 2101, Tower 2
Metroplaza
223 Hing Fong Road
Kwai Fong, N.T., Hong Kong
Tel: 852-2-401-1200 Fax: 852-2-401-3431

Beijing
Microchip Technology, Beijing
Unit 915, 6 Chaoyangmen Bei Dajie
Dong Erhuan Road, Dongcheng District
New China Hong Kong Manhattan Building
Beijing 100027 PRC
Tel: 86-10-85282100 Fax: 86-10-85282104

India
Microchip Technology Inc.
India Liaison Office
No. 6, Legacy, Convent Road
Bangalore 560 025, India
Tel: 91-80-229-0061 Fax: 91-80-229-0062

Japan
Microchip Technology Intl. Inc.
Benex S-1 6F
3-18-20, Shinyokohama
Kohoku-Ku, Yokohama-shi
Kanagawa 222-0033 Japan
Tel: 81-45-471- 6166 Fax: 81-45-471-6122

Korea
Microchip Technology Korea
168-1, Youngbo Bldg. 3 Floor
Samsung-Dong, Kangnam-Ku
Seoul, Korea
Tel: 82-2-554-7200 Fax: 82-2-558-5934

Shanghai
Microchip Technology
RM 406 Shanghai Golden Bridge Bldg.
2077 Yan'an Road West, Hong Qiao District
Shanghai, PRC 200335
Tel: 86-21-6275-5700 Fax: 86 21-6275-5060

ASIA/PACIFIC (continued)

Singapore
Microchip Technology Singapore Pte Ltd.
200 Middle Road
#07-02 Prime Centre
Singapore 188980
Tel: 65-334-8870 Fax: 65-334-8850

Taiwan, R.O.C
Microchip Technology Taiwan
10F-1C 207
Tung Hua North Road
Taipei, Taiwan, ROC
Tel: 886-2-2717-7175 Fax: 886-2-2545-0139

EUROPE

United Kingdom
Arizona Microchip Technology Ltd.
505 Eskdale Road
Winnersh Triangle
Wokingham
Berkshire, England RG41 5TU
Tel: 44 118 921 5858 Fax: 44-118 921-5835

Denmark
Microchip Technology Denmark ApS
Regus Business Centre
Lautrup hoj 1-3
Ballerup DK-2750 Denmark
Tel: 45 4420 9895 Fax: 45 4420 9910

France
Arizona Microchip Technology SARL
Parc d'Activite du Moulin de Massy
43 Rue du Saule Trapu
Batiment A - ler Etage
91300 Massy, France
Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79

Germany
Arizona Microchip Technology GmbH
Gustav-Heinemann-Ring 125
D-81739 München, Germany
Tel: 49-89-627-144 0 Fax: 49-89-627-144-44

Italy
Arizona Microchip Technology SRL
Centro Direzionale Colleoni
Palazzo Taurus 1 V. Le Colleoni 1
20041 Agrate Brianza
Milan, Italy
Tel: 39-039-65791-1 Fax: 39-039-6899883

11/15/99

DNV Certification, Inc.
USA
DNV MSC
The Netherlands
Accredited by the RvA
ISO 9001 / QS-9000
REGISTERED FIRM

Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999. The Company's quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs and microperipheral products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001 certified.