



MICROCHIP

PIC16C717/770/771

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

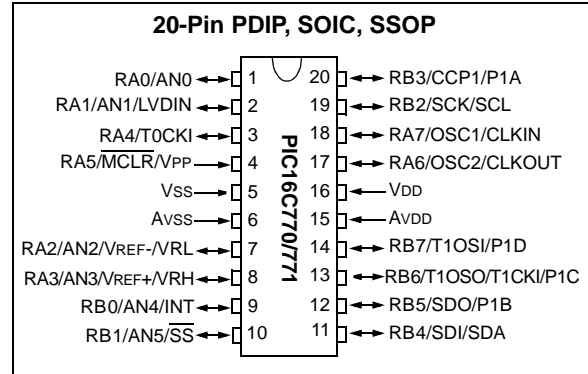
Microcontroller Core Features:

- 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 (4 MHz and 37 kHz nominal) dynamically switchable for power savings
 - ER - External resistor, dual speed (user selectable frequency and 37 kHz nominal) 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™ (ICSP™)
- 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 @ 4V, 4 MHz
 - 11 µA typical @ 2.5V, 37 kHz
 - < 1 µA typical standby current

Pin Diagram

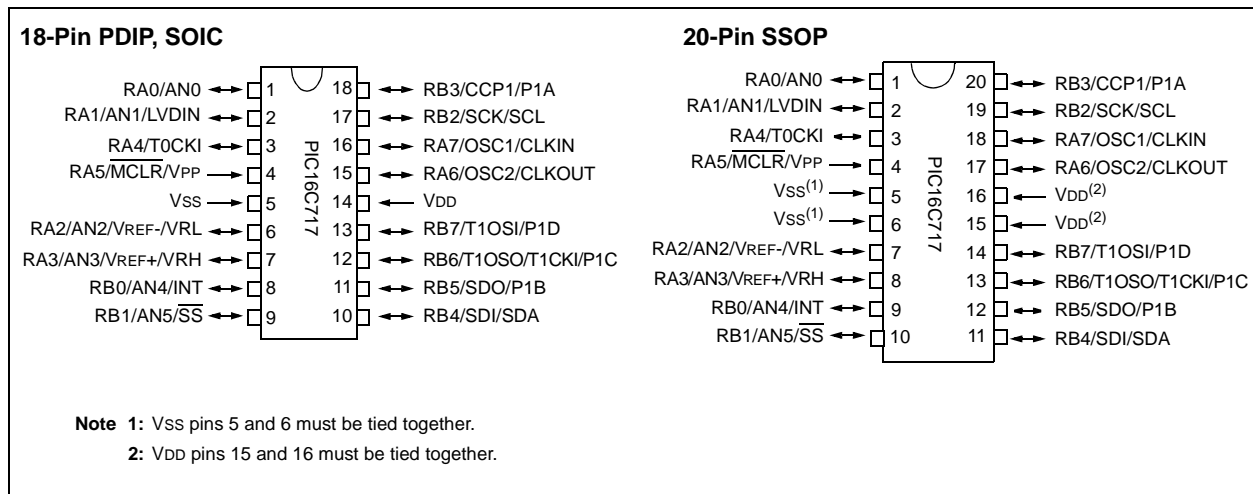


Peripheral Features:

- 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

PIC16C717/770/771

Pin Diagrams



| Key Features PICmicro™ Mid-Range MCU Family Reference Manual, (DS33023) | PIC16C717 | PIC16C770 | PIC16C771 |
|--|---------------------------------|---------------------------------|---------------------------------|
| Operating Frequency | DC - 20 MHz | DC - 20 MHz | DC - 20 MHz |
| RESETS (and Delays) | POR, BOR, MCLR, WDT (PWRT, OST) | POR, BOR, MCLR, WDT (PWRT, OST) | POR, BOR, 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

| | |
|---|-----|
| 1.0 Device Overview | 5 |
| 2.0 Memory Organization..... | 9 |
| 3.0 I/O Ports..... | 25 |
| 4.0 Program Memory Read (PMR) | 41 |
| 5.0 Timer0 Module | 45 |
| 6.0 Timer1 Module | 47 |
| 7.0 Timer2 Module | 51 |
| 8.0 Enhanced Capture/Compare/PWM (ECCP) Modules..... | 53 |
| 9.0 Master Synchronous Serial Port (MSSP) Module..... | 65 |
| 10.0 Voltage Reference Module and Low-voltage Detect..... | 101 |
| 11.0 Analog-to-Digital Converter (A/D) Module..... | 105 |
| 12.0 Special Features of the CPU | 117 |
| 13.0 Instruction Set Summary..... | 133 |
| 14.0 Development Support | 141 |
| 15.0 Electrical Characteristics..... | 147 |
| 16.0 DC and AC Characteristics Graphs and Tables..... | 179 |
| 17.0 Packaging Information | 197 |
| APPENDIX A: Revision History | 207 |
| APPENDIX B: Device Differences | 208 |
| Index | 209 |
| On-Line Support..... | 215 |
| Reader Response | 216 |
| PIC16C717/770/771 Product Identification System..... | 217 |

TO OUR VALUED CUSTOMERS

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

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

Most Current Data Sheet

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

<http://www.microchip.com>

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

Errata

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

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

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

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

Customer Notification System

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

PIC16C717/770/771

NOTES:

PIC16C717/770/771

1.0 DEVICE OVERVIEW

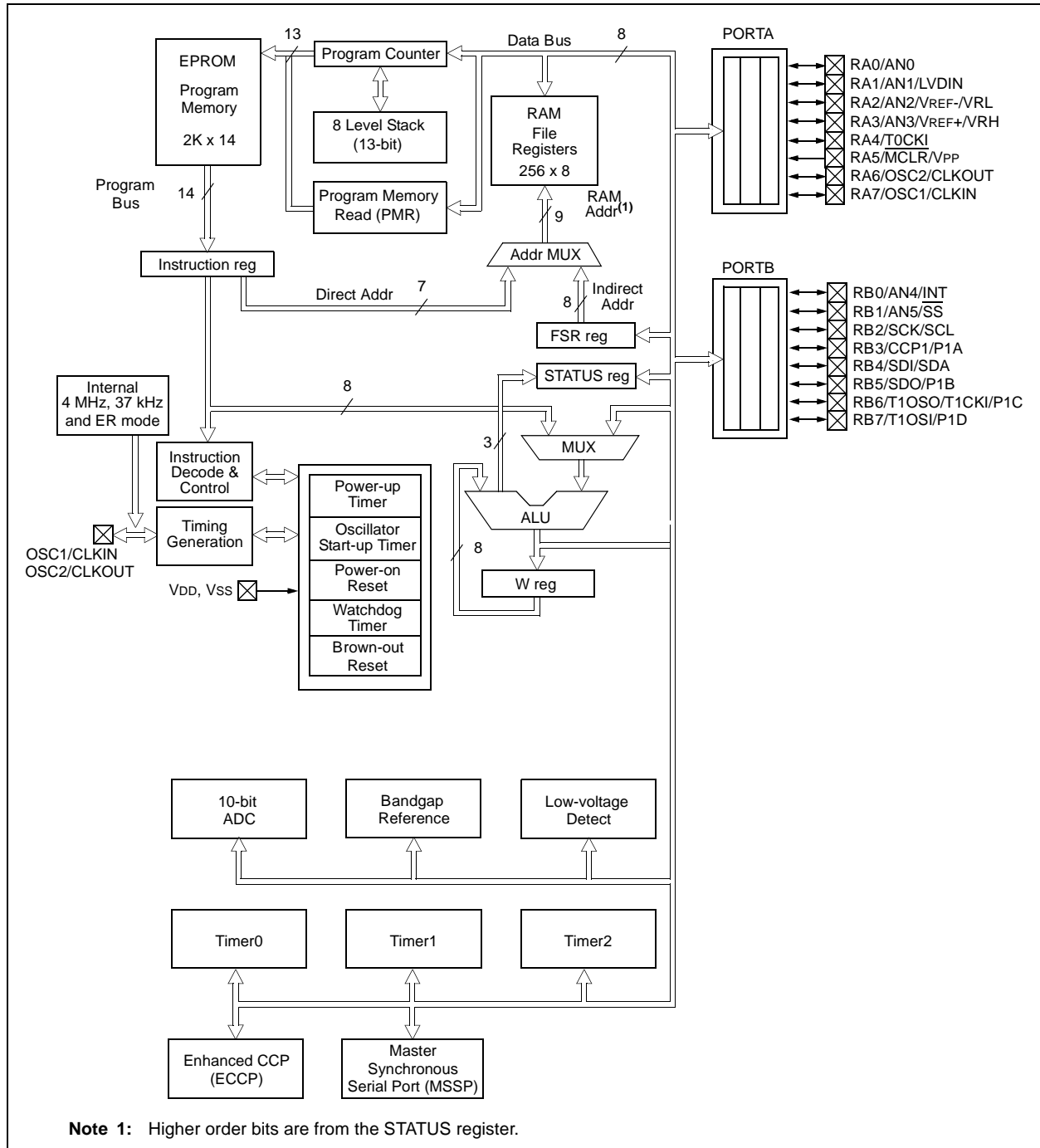
This document contains device-specific information. Additional information may be found in the PICmicro™ Mid-Range MCU Family Reference Manual, (DS33023), which may be obtained from your local Microchip Sales Representative or downloaded from the Microchip website. The Reference Manual should be considered 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 data sheet. The PIC16C717 device comes in 18/20-pin packages and the PIC16C770/771 devices come in 20-pin packages.

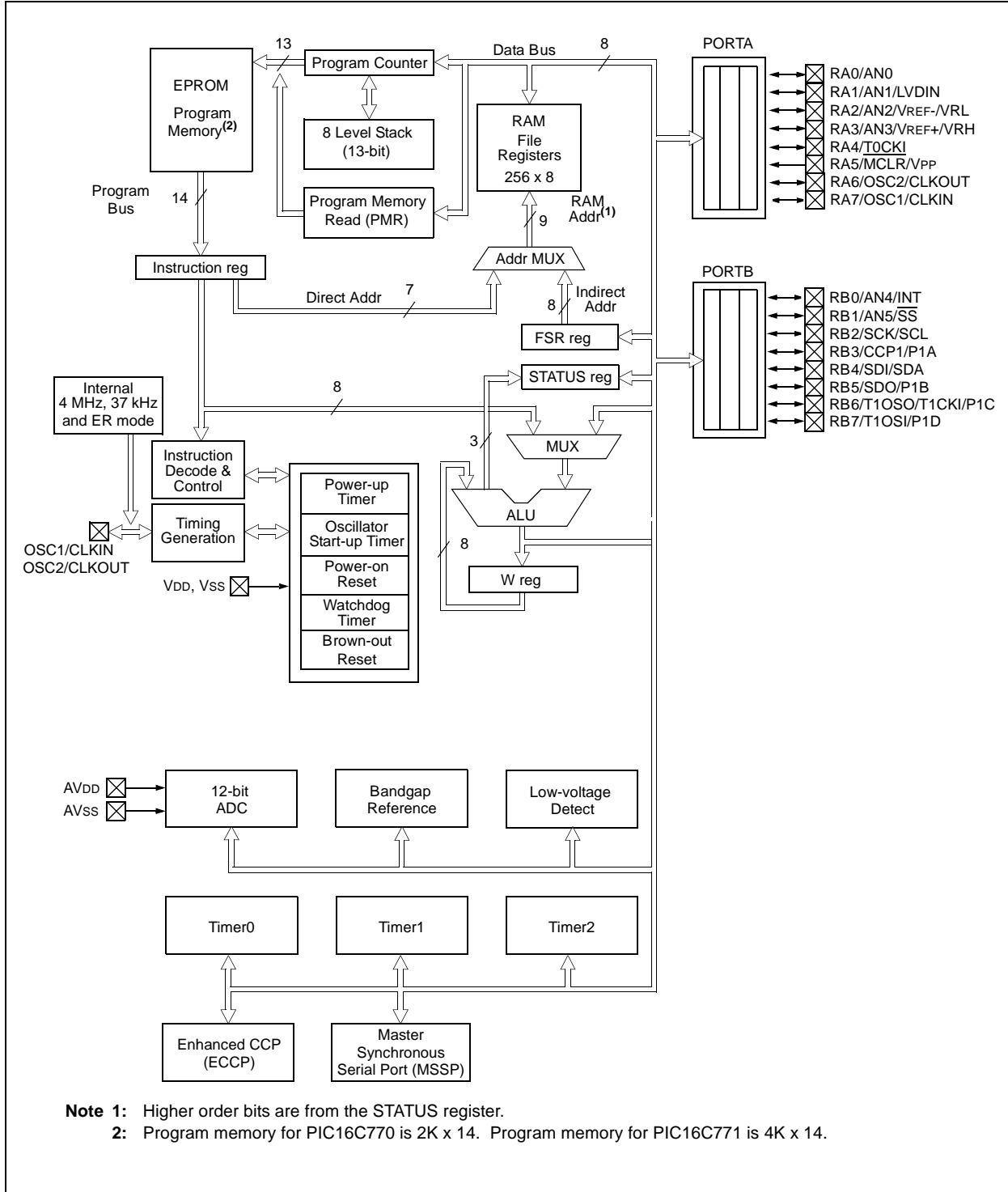
The following two figures are device block diagrams of the PIC16C717 and the PIC16C770/771.

FIGURE 1-1: PIC16C717 BLOCK DIAGRAM



PIC16C717/770/771

FIGURE 1-2: PIC16C770/771 BLOCK DIAGRAM



PIC16C717/770/771

TABLE 1-1: PIC16C717/770/771 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 I/O ⁽¹⁾ |
| | SCK | ST | CMOS | Serial clock I/O for SPI |
| | SCL | ST | OD | Serial clock I/O for I ² C |
| RB3/CCP1/P1A | RB3 | TTL | CMOS | Bi-directional I/O ⁽¹⁾ |
| | CCP1 | ST | CMOS | Capture 1 input/Compare 1 output |
| | P1A | | CMOS | PWM P1A output |
| RB4/SDI/SDA | RB4 | TTL | CMOS | Bi-directional I/O ⁽¹⁾ |
| | SDI | ST | | Serial data in for SPI |
| | SDA | ST | OD | Serial data I/O for I ² C |
| RB5/SDO/P1B | RB5 | TTL | CMOS | Bi-directional I/O ⁽¹⁾ |
| | SDO | | CMOS | Serial data out for SPI |
| | P1B | | CMOS | PWM P1B output |

Note 1: Bit programmable pull-ups.

Note 2: Only in PIC16C770/771 devices.

PIC16C717/770/771

TABLE 1-1: PIC16C717/770/771 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 | CMOS | | 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 |
| Vss | Vss | Power | | Ground reference for logic and I/O pins |
| VDD | VDD | Power | | Positive supply for logic and I/O pins |
| AVss ⁽²⁾ | AVss | Power | | Ground reference for analog |
| AVDD ⁽²⁾ | AVDD | Power | | Positive supply for analog |

Note 1: Bit programmable pull-ups.

Note 2: Only in PIC16C770/771 devices.

2.0 MEMORY ORGANIZATION

There are two memory blocks in each of these PIC[®] 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 MCU Family 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 wrap-around.

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

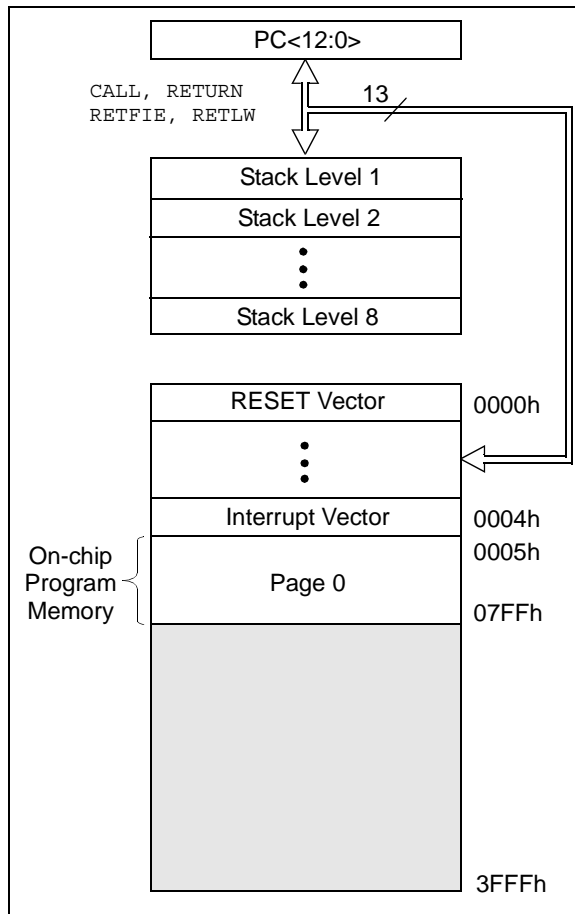
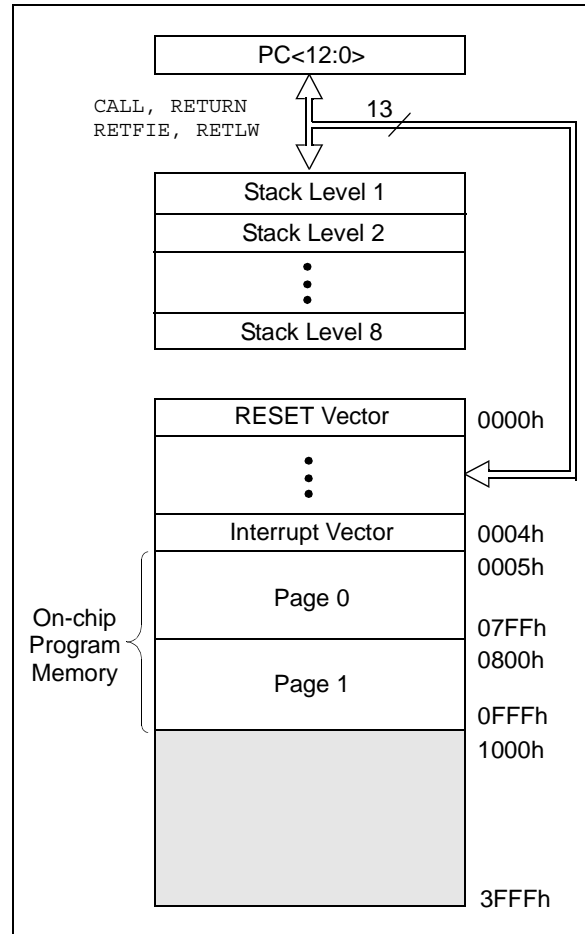


FIGURE 2-2: PROGRAM MEMORY MAP AND STACK OF THE PIC16C771



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.

PIC16C717/770/771

FIGURE 2-3: REGISTER FILE MAP

| File Address | | File Address | | File Address | | 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 |
| | 20h | | A0h | | 120h | | 1A0h |
| General Purpose Register | | General Purpose Register 80 Bytes | | General Purpose Register 80 Bytes | | | |
| 96 Bytes | | | | | | | |
| | | accesses 70h-7Fh | | accesses 70h - 7Fh | | accesses 70h - 7Fh | |
| | 7Fh | | EFh | | 16Fh | | 1EFh |
| | | | F0h | | 170h | | 1F0h |
| | | | FFh | | 17Fh | | 1FFh |
| Bank 0 | | Bank 1 | | Bank 2 | | Bank 3 | |

Unimplemented data memory locations, read as '0'.
 * Not a physical register.

PIC16C717/770/771

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 | Details on Page: |
|----------------------|---------|--|---------|---------|--|-----------------|----------------------|---------|---------|--------------------------|------------------------|
| Bank 0 | | | | | | | | | | | |
| 00h ⁽³⁾ | INDF | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | 0000 0000 | 23 |
| 01h | TMR0 | Timer0 module's register | | | | | | | | xxxx xxxx | 45 |
| 02h ⁽³⁾ | PCL | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 22 |
| 03h ⁽³⁾ | STATUS | IRP | RP1 | RP0 | \overline{TO} | \overline{PD} | Z | DC | C | 0001 1xxx | 14 |
| 04h ⁽³⁾ | FSR | Indirect data memory address pointer | | | | | | | | xxxx xxxx | 23 |
| 05h | PORTA | RA7 | RA6 | RA5 | RA4 | RA3 | RA2 | RA1 | RA0 | xxxx 0000 | 25 |
| 06h | PORTB | RB7 | RB6 | RB5 | RB4 | RB3 | RB2 | RB1 | RB0 | xxxx xx11 | 33 |
| 07h | — | Unimplemented | | | | | | | | — | — |
| 08h | — | Unimplemented | | | | | | | | — | — |
| 09h | — | Unimplemented | | | | | | | | — | — |
| 0Ah ^(1,3) | PCLATH | — | — | — | Write Buffer for the upper 5 bits of the Program Counter | | | | | ---0 0000 | 22 |
| 0Bh ⁽³⁾ | INTCON | GIE | PEIE | TOIE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 16 |
| 0Ch | PIR1 | — | ADIF | — | — | SSPIF | CCP1IF | TMR2IF | TMR1IF | 0---0000 | 18 |
| 0Dh | PIR2 | LVDIF | — | — | — | BCLIF | — | — | — | 0--- 0--- | 20 |
| 0Eh | TMR1L | Holding register for the Least Significant Byte of the 16-bit TMR1 register | | | | | | | | xxxx xxxx | 47 |
| 0Fh | TMR1H | Holding register for the Most Significant Byte of the 16-bit TMR1 register | | | | | | | | xxxx xxxx | 47 |
| 10h | T1CON | — | — | T1CKPS1 | T1CKPS0 | T1OSCEN | $\overline{T1SYNC}$ | TMR1CS | TMR1ON | --00 0000 | 47 |
| 11h | TMR2 | Timer2 module's register | | | | | | | | 0000 0000 | 51 |
| 12h | T2CON | — | TOUTPS3 | TOUTPS2 | TOUTPS1 | TOUTPS0 | TMR2ON | T2CKPS1 | T2CKPS0 | -000 0000 | 51 |
| 13h | SSPBUF | Synchronous Serial Port Receive Buffer/Transmit Register | | | | | | | | xxxx xxxx | 70 |
| 14h | SSPCON | WCOL | SSPOV | SSPEN | CKP | SSPM3 | SSPM2 | SSPM1 | SSPM0 | 0000 0000 | 67 |
| 15h | CCPR1L | Capture/Compare/PWM Register1 (LSB) | | | | | | | | xxxx xxxx | 54 |
| 16h | CCPR1H | Capture/Compare/PWM Register1 (MSB) | | | | | | | | xxxx xxxx | 54 |
| 17h | CCP1CON | PWM1M1 | PWM1M0 | DC1B1 | DC1B0 | CCP1M3 | CCP1M2 | CCP1M1 | CCP1M0 | 0000 0000 | 53 |
| 18h | — | Unimplemented | | | | | | | | — | — |
| 19h | — | Unimplemented | | | | | | | | — | — |
| 1Ah | — | Unimplemented | | | | | | | | — | — |
| 1Bh | — | Unimplemented | | | | | | | | — | — |
| 1Ch | — | Unimplemented | | | | | | | | — | — |
| 1Dh | — | Unimplemented | | | | | | | | — | — |
| 1Eh | ADRESH | A/D High Byte Result Register | | | | | | | | xxxx xxxx | 107 |
| 1Fh | ADCON0 | ADCS1 | ADCS0 | CHS2 | CHS1 | CHS0 | $\overline{GO/DONE}$ | CHS3 | ADON | 0000 0000 | 107 |

Legend: x = unknown, u = unchanged, c = 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 MCLR and Watchdog Timer Reset.

3: These registers can be addressed from any bank.

PIC16C717/770/771

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 | Details on Page: |
|----------------------|------------|--|---------|-----------------------|--|------------|--------------|-------------|-------------|--------------------|------------------|
| Bank 1 | | | | | | | | | | | |
| 80h ⁽³⁾ | INDF | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | 0000 0000 | 23 |
| 81h | OPTION_REG | RBP \bar{U} | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 15 |
| 82h ⁽³⁾ | PCL | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 22 |
| 83h ⁽³⁾ | STATUS | IRP | RP1 | RP0 | $\bar{T}O$ | $\bar{P}D$ | Z | DC | C | 0001 1xxx | 14 |
| 84h ⁽³⁾ | FSR | Indirect data memory address pointer | | | | | | | | xxxxx xxxxx | 23 |
| 85h | TRISA | PORTA Data Direction Register | | | | | | | | 1111 1111 | 25 |
| 86h | TRISB | PORTB Data Direction Register | | | | | | | | 1111 1111 | 33 |
| 87h | — | Unimplemented | | | | | | | | — | — |
| 88h | — | Unimplemented | | | | | | | | — | — |
| 89h | — | Unimplemented | | | | | | | | — | — |
| 8Ah ^(1,3) | PCLATH | — | — | — | Write Buffer for the upper 5 bits of the Program Counter | | | | | ---0 0000 | 22 |
| 8Bh ⁽³⁾ | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 16 |
| 8Ch | PIE1 | — | ADIE | — | — | SSPIE | CCP1IE | TMR2IE | TMR1IE | -0-- 0000 | 17 |
| 8Dh | PIE2 | LVDIE | — | — | — | BCLIE | — | — | — | 0--- 0--- | 19 |
| 8Eh | PCON | — | — | — | — | OSCF | — | $\bar{P}OR$ | $\bar{B}OR$ | ---- 1-gg | 21 |
| 8Fh | — | Unimplemented | | | | | | | | — | — |
| 90h | — | Unimplemented | | | | | | | | — | — |
| 91h | SSPCON2 | GCEN | ACKSTAT | ACKDT | ACKEN | RCEN | PEN | RSEN | SEN | 0000 0000 | 69 |
| 92h | PR2 | Timer2 Period Register | | | | | | | | 1111 1111 | 52 |
| 93h | SSPADD | Synchronous Serial Port (I ² C mode) Address Register | | | | | | | | 0000 0000 | 76 |
| 94h | SSPSTAT | SMP | CKE | D/ \bar{A} | P | S | R/ \bar{W} | UA | BF | 0000 0000 | 66 |
| 95h | WPUB | PORTB Weak Pull-up Control | | | | | | | | 1111 1111 | 34 |
| 96h | IOCB | PORTB Interrupt on Change Control | | | | | | | | 1111 0000 | 34 |
| 97h | P1DEL | PWM 1 Delay value | | | | | | | | 0000 0000 | 62 |
| 98h | — | Unimplemented | | | | | | | | — | — |
| 99h | — | Unimplemented | | | | | | | | — | — |
| 9Ah | — | Unimplemented | | | | | | | | — | — |
| 9Bh | REFCON | VRHEN | VRLEN | VRHOEN | VRLOEN | — | — | — | — | 0000 ---- | 102 |
| 9Ch | LVDCON | — | — | BGST | LV DEN | LVV3 | LVV2 | LVV1 | LVV0 | --00 0101 | 101 |
| 9Dh | ANSEL | — | — | Analog Channel Select | | | | | | --11 1111 | 25 |
| 9Eh | ADRESL | A/D Low Byte Result Register | | | | | | | | xxxxx xxxxx | 107 |
| 9Fh | ADCON1 | ADFM | VCFG2 | VCFG1 | VCFG0 | — | — | — | — | 0000 ---- | 107 |

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 $\bar{M}CLR$ and Watchdog Timer Reset.

3: These registers can be addressed from any bank.

PIC16C717/770/771

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 | Details on Page: |
|-----------------------|------------|--|--------|-------------------------------|--|----------------------------------|-------|-------|-----------|--------------------------|------------------------|
| Bank 2 | | | | | | | | | | | |
| 100h ⁽³⁾ | INDF | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | 0000 0000 | 23 |
| 101h | TMR0 | Timer0 module's register | | | | | | | | xxxx xxxx | 45 |
| 102h ⁽³⁾ | PCL | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 22 |
| 103h ⁽³⁾ | STATUS | IRP | RP1 | RP0 | \overline{TO} | \overline{PD} | Z | DC | C | 0001 1xxx | 14 |
| 104h ⁽³⁾ | FSR | Indirect data memory address pointer | | | | | | | | xxxx xxxx | 23 |
| 105h | — | Unimplemented | | | | | | | | — | — |
| 106h | PORTB | PORTB Data Latch when written: PORTB pins when read | | | | | | | | xxxx xx11 | 33 |
| 107h | — | Unimplemented | | | | | | | | — | — |
| 108h | — | Unimplemented | | | | | | | | — | — |
| 109h | — | Unimplemented | | | | | | | | — | — |
| 10Ah ^(1,3) | PCLATH | — | — | — | Write Buffer for the upper 5 bits of the Program Counter | | | | | ---0 0000 | 22 |
| 10Bh ⁽³⁾ | INTCON | GIE | PEIE | TOIE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 16 |
| 10Ch | PMDATL | Program memory read data low | | | | | | | | xxxx xxxx | |
| 10Dh | PMADRL | Program memory read address low | | | | | | | | xxxx xxxx | |
| 10Eh | PMDATH | — | — | Program memory read data high | | | | | --xx xxxx | | |
| 10Fh | PMADRH | — | — | — | — | Program memory read address high | | | | ---- xxxx | |
| 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 | 23 |
| 181h | OPTION_REG | \overline{RBPU} | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 15 |
| 182h ⁽³⁾ | PCL | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 22 |
| 183h ⁽³⁾ | STATUS | IRP | RP1 | RP0 | \overline{TO} | \overline{PD} | Z | DC | C | 0001 1xxx | 14 |
| 184h ⁽³⁾ | FSR | Indirect data memory address pointer | | | | | | | | xxxx xxxx | 23 |
| 185h | — | Unimplemented | | | | | | | | — | — |
| 186h | TRISB | PORTB Data Direction Register | | | | | | | | 1111 1111 | 33 |
| 187h | — | Unimplemented | | | | | | | | — | — |
| 188h | — | Unimplemented | | | | | | | | — | — |
| 189h | — | Unimplemented | | | | | | | | — | — |
| 18Ah ^(1,3) | PCLATH | — | — | — | Write Buffer for the upper 5 bits of the Program Counter | | | | | ---0 0000 | 22 |
| 18Bh ⁽³⁾ | INTCON | GIE | PEIE | TOIE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 16 |
| 18Ch | PMCON1 | Reserved | — | — | — | — | — | — | RD | 1--- ---0 | |
| 18Dh- 18Fh | — | Unimplemented | | | | | | | | — | — |

Legend: x = unknown, u = unchanged, c = 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 MCLR and Watchdog Timer Reset.

3: These registers can be addressed from any bank.

PIC16C717/770/771

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{TO} and \overline{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: The \overline{C} and \overline{DC} bits operate as a borrow and 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{TO} | \overline{PD} | Z | DC | C |
| bit 7 | | | | | | | | bit 0 |

- 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{TO} :** Time-out bit
 1 = After power-up, `CLRWDT` instruction, or `SLEEP` instruction
 0 = A WDT time-out occurred
- bit 3 **\overline{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/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/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.

Legend:

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

PIC16C717/770/771

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 |
| | RBP $\overline{\text{U}}$ | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 |
| bit 7 | | | | | | | | bit 0 |

- bit 7 **RBP $\overline{\text{U}}$:** 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).

Legend:

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

PIC16C717/770/771

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 | TOIE | INTE | RBIE | TOIF | INTF | RBIF |
| bit 7 | | | | | | bit 0 | |

- 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 **TOIE:** 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 **TOIF:** 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).

Legend:

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

PIC16C717/770/771

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 |
| bit 7 | | | | | | | bit 0 |

- 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

Legend:

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

PIC16C717/770/771

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 |
| bit 7 | | | | bit 0 | | | |

- 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.
 - I²C Slave / Master
A transmission/reception has taken place.
 - I²C 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 (Multi-master system).
A STOP condition occurred while the SSP module was IDLE (Multi-master 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

Legend:

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

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 ENABLE REGISTER 2 (PIE2: 8Dh)

| | | | | | | | |
|-------|-----|-----|-----|-------|-----|-----|-------|
| R/W-0 | U-0 | U-0 | U-0 | R/W-0 | U-0 | U-0 | U-0 |
| LVDIE | — | — | — | BCLIE | — | — | — |
| bit 7 | | | | | | | bit 0 |

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'

Legend:

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

PIC16C717/770/771

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 | — | — | — |
| bit 7 | | | | | | | bit 0 |

- 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 I²C Master was transmitting (must be cleared in software)
0 = No bus collision occurred
- bit 2-0 **Unimplemented:** Read as '0'

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

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 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: $\overline{\text{BOR}}$ is unknown on Power-on Reset. It must then be set by the user and checked on subsequent RESETS to see if $\overline{\text{BOR}}$ is clear, indicating a brown-out has occurred. The $\overline{\text{BOR}}$ 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 | — | $\overline{\text{POR}}$ | $\overline{\text{BOR}}$ |
| bit 7 | | | | | | | bit 0 | |

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

bit 3 **OSCF:** Oscillator Speed bit

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

bit 1 **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 **BOR:** Brown-out Reset Status bit (See Section 2.2.2.8 Note)

1 = No Brown-out Reset occurred

0 = A Brown-out Reset occurred (must be set in software after a Brown-out Reset occurs)

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

PIC16C717/770/771

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).

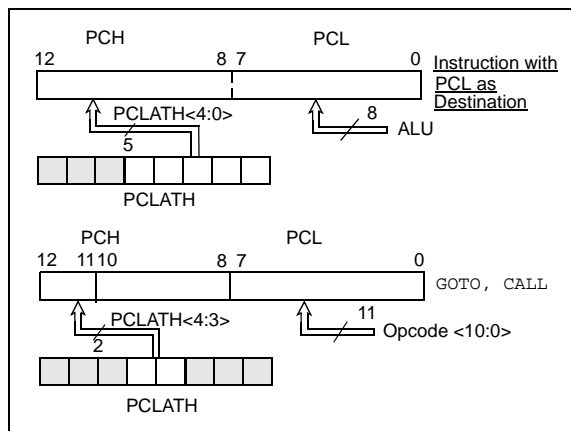
2.4 Stack

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).

FIGURE 2-4: LOADING OF PC IN DIFFERENT SITUATIONS



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.

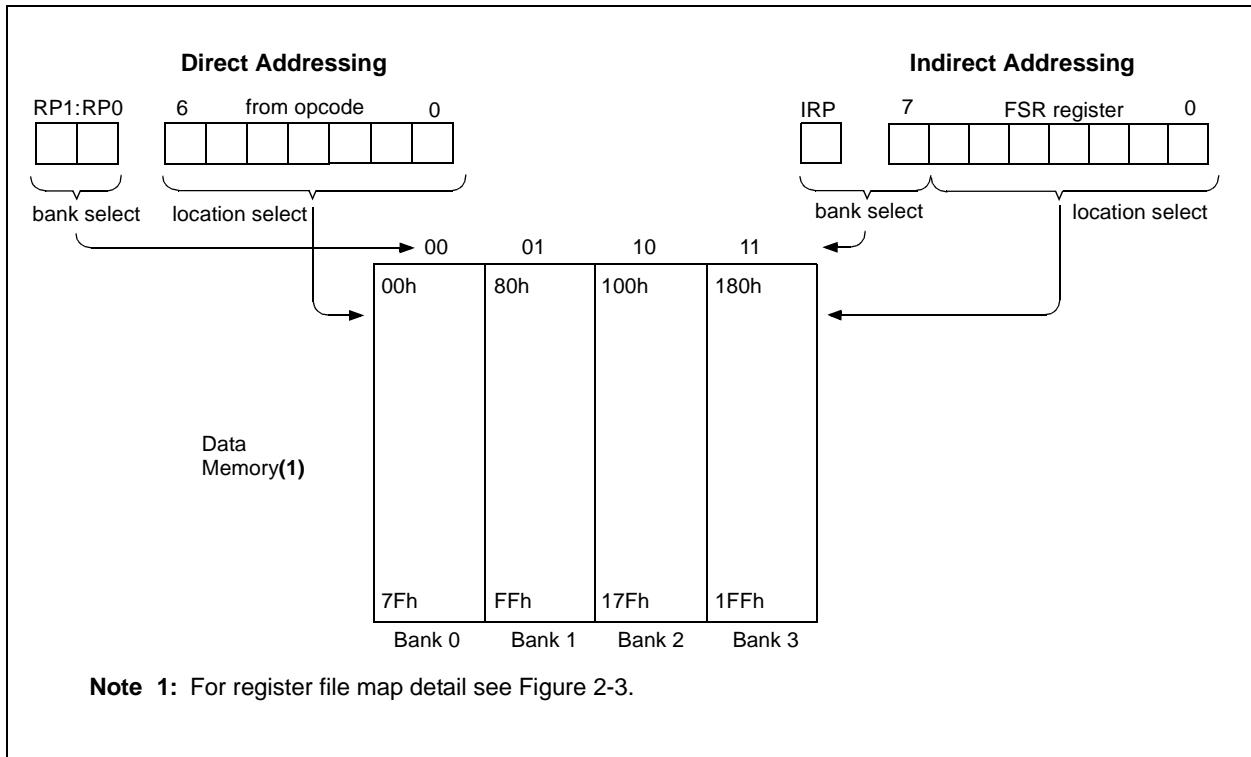
EXAMPLE 2-1: How to Clear RAM Using Indirect Addressing

```

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 2-5.

FIGURE 2-5: DIRECT/INDIRECT ADDRESSING



PIC16C717/770/771

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 PIC Mid-Range MCU Family 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 RA pin will always read '0' and RB pin will always read '1', 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 |
| bit 7 | | | | | | bit 0 | |

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 the digital input buffer on the pin. The corresponding TRIS bit should be set to Input mode when using pins as analog inputs.

Legend:

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

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 onboard 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 (MCLR) and programming input (VPP) functions. The RA5/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.

PIC16C717/770/771

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

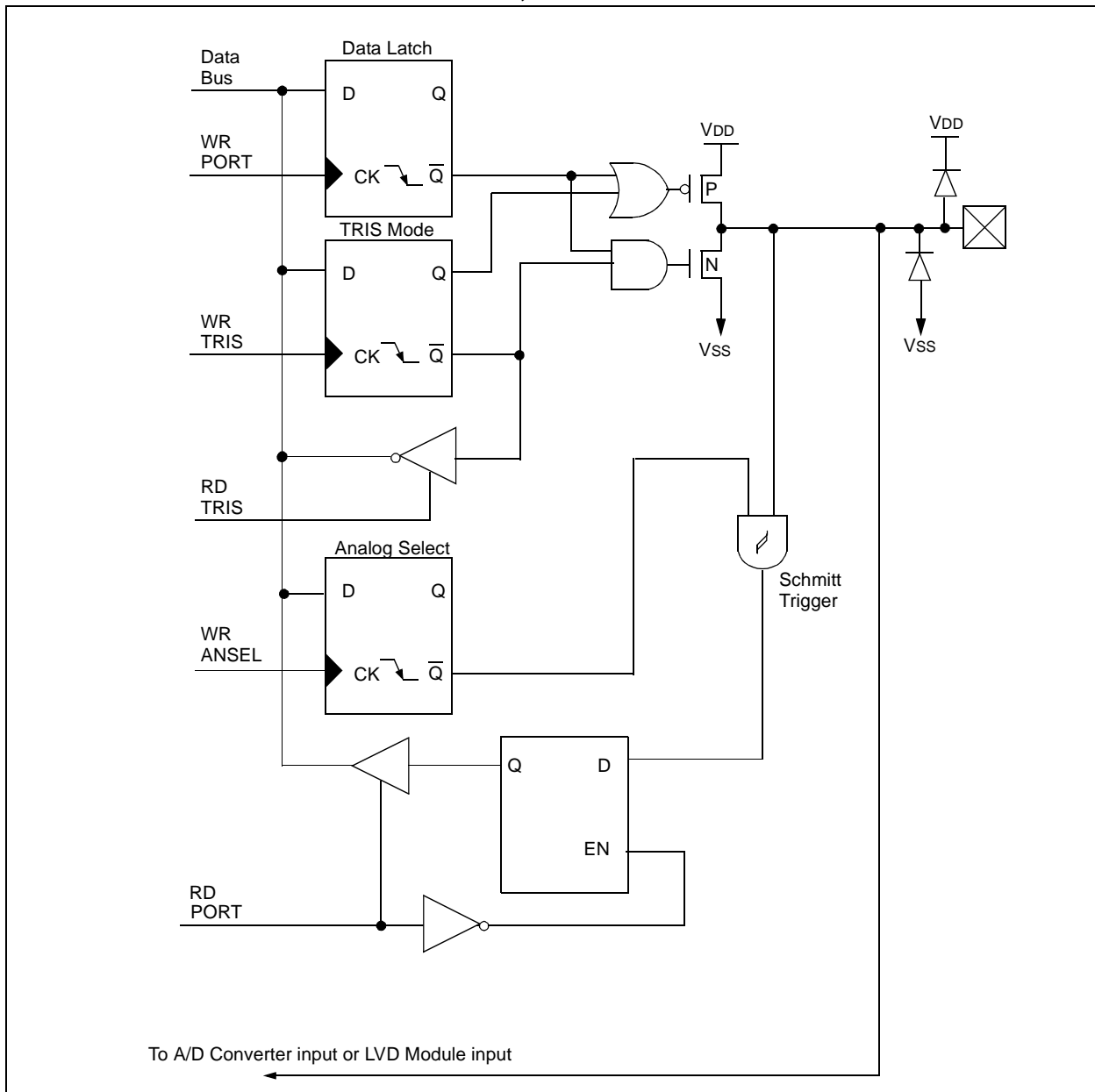
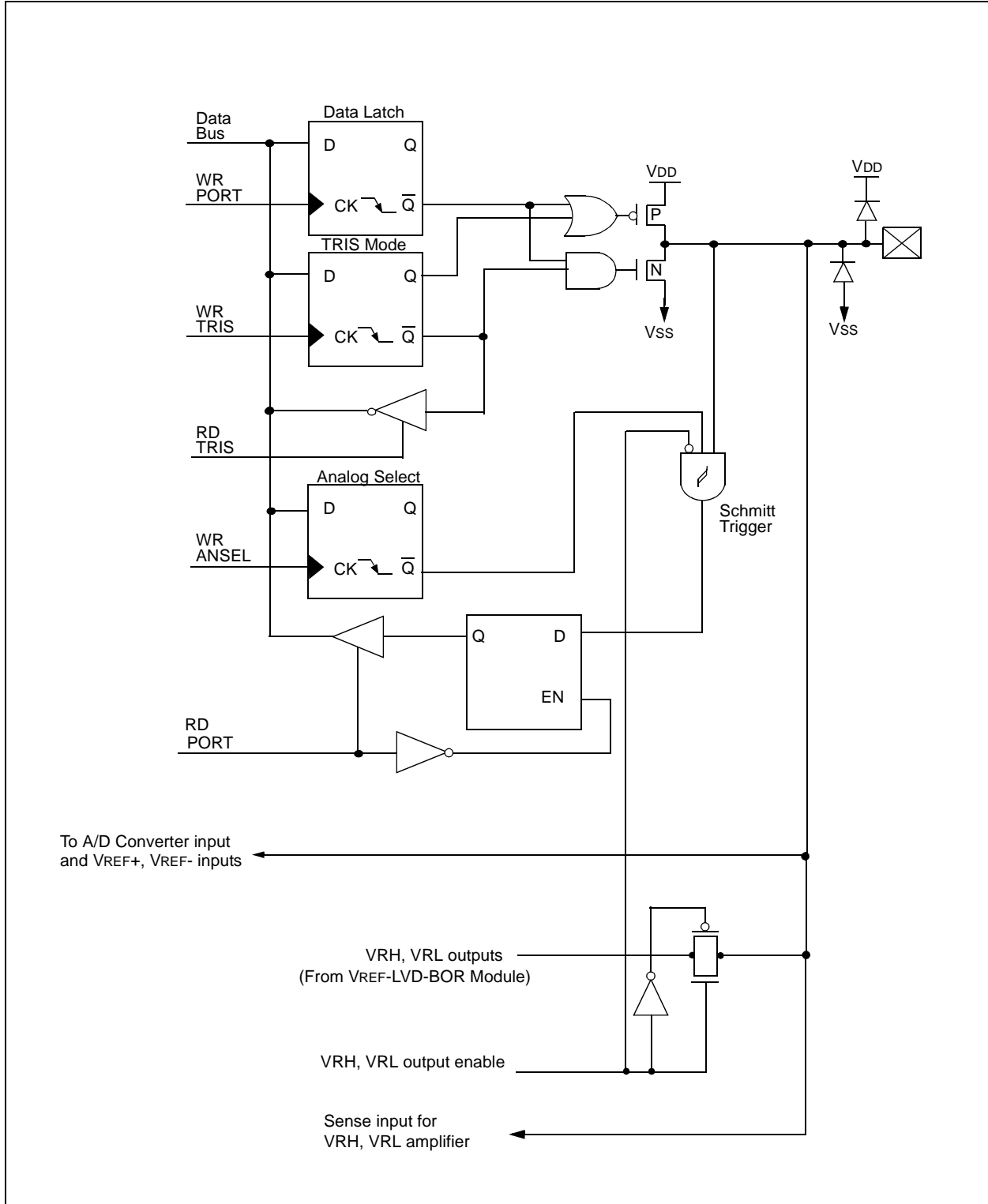


FIGURE 3-2: BLOCK DIAGRAM OF RA2/AN2/VREF-/VRL AND RA3/AN3/VREF+/VRH



PIC16C717/770/771

FIGURE 3-3: BLOCK DIAGRAM OF RA4/T0CKI

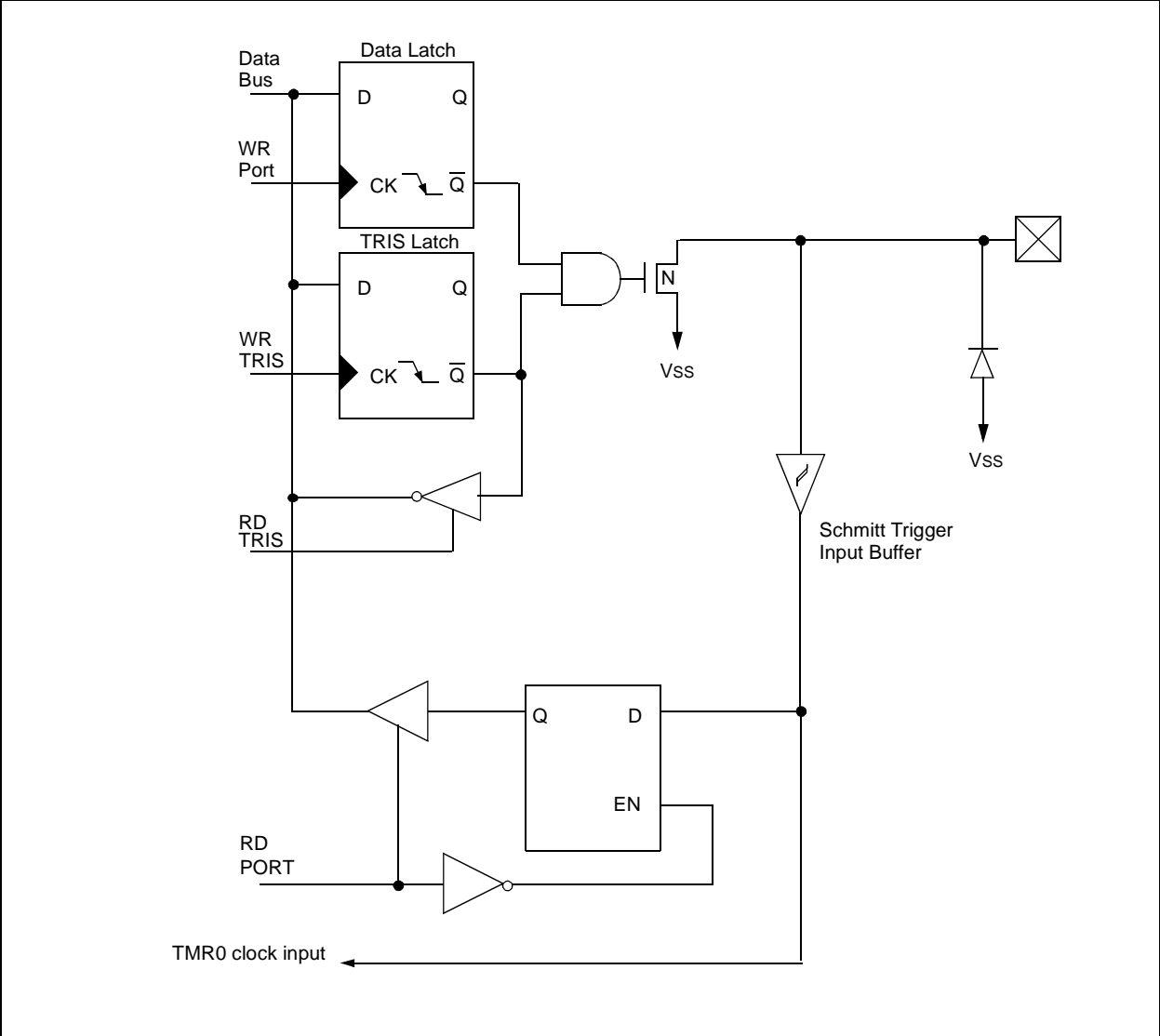
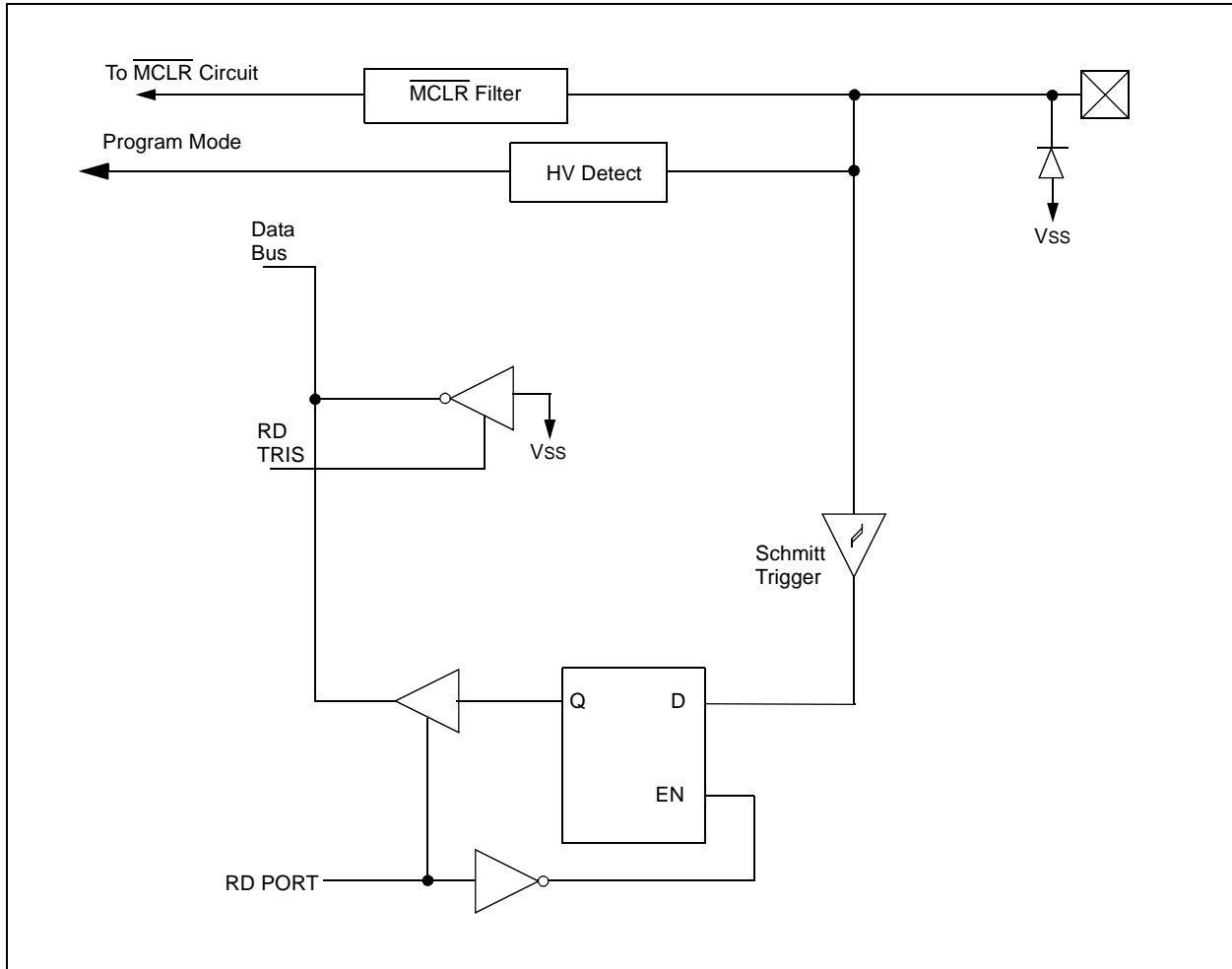


FIGURE 3-4: BLOCK DIAGRAM OF RA5/MCLR/VPP



PIC16C717/770/771

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

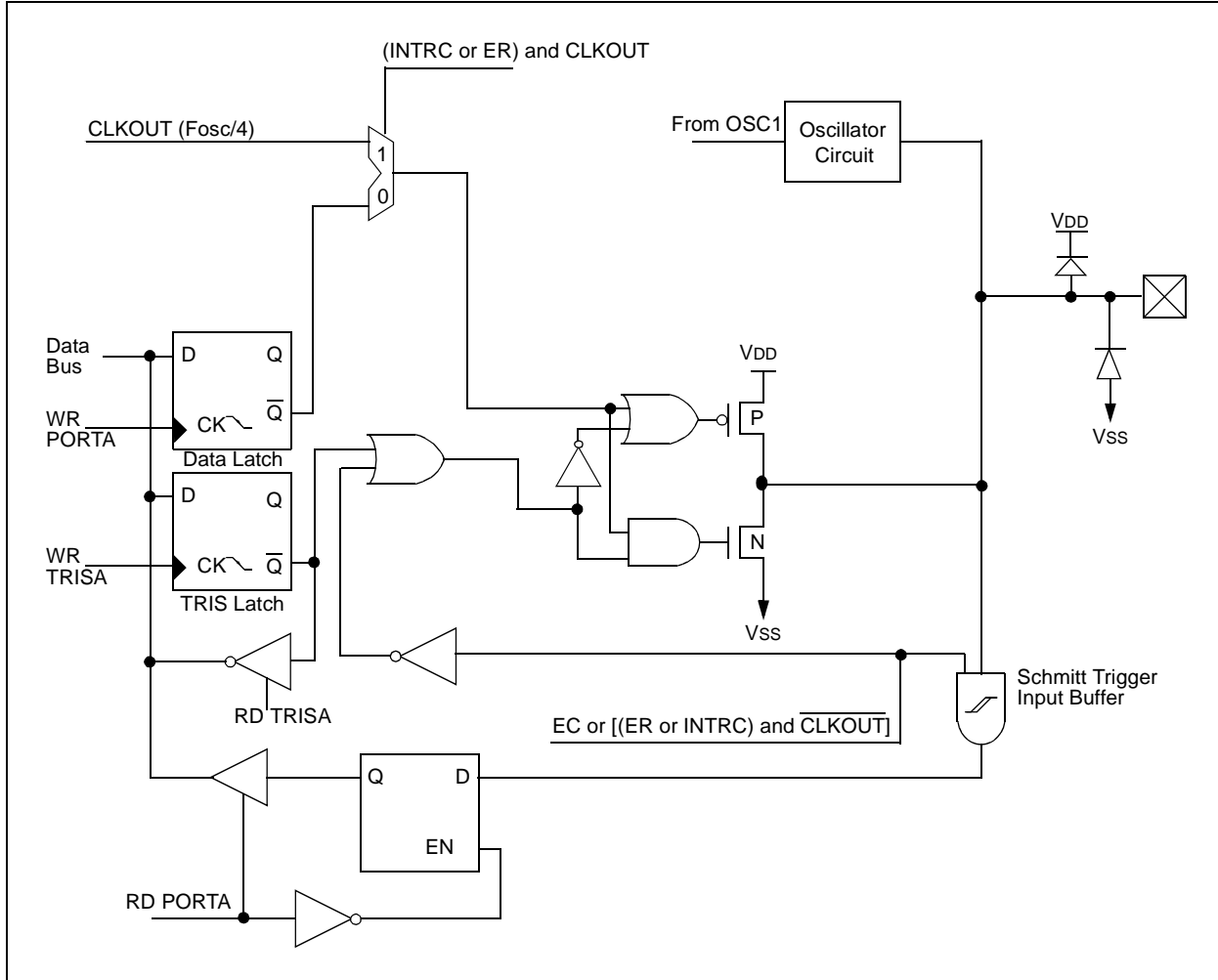
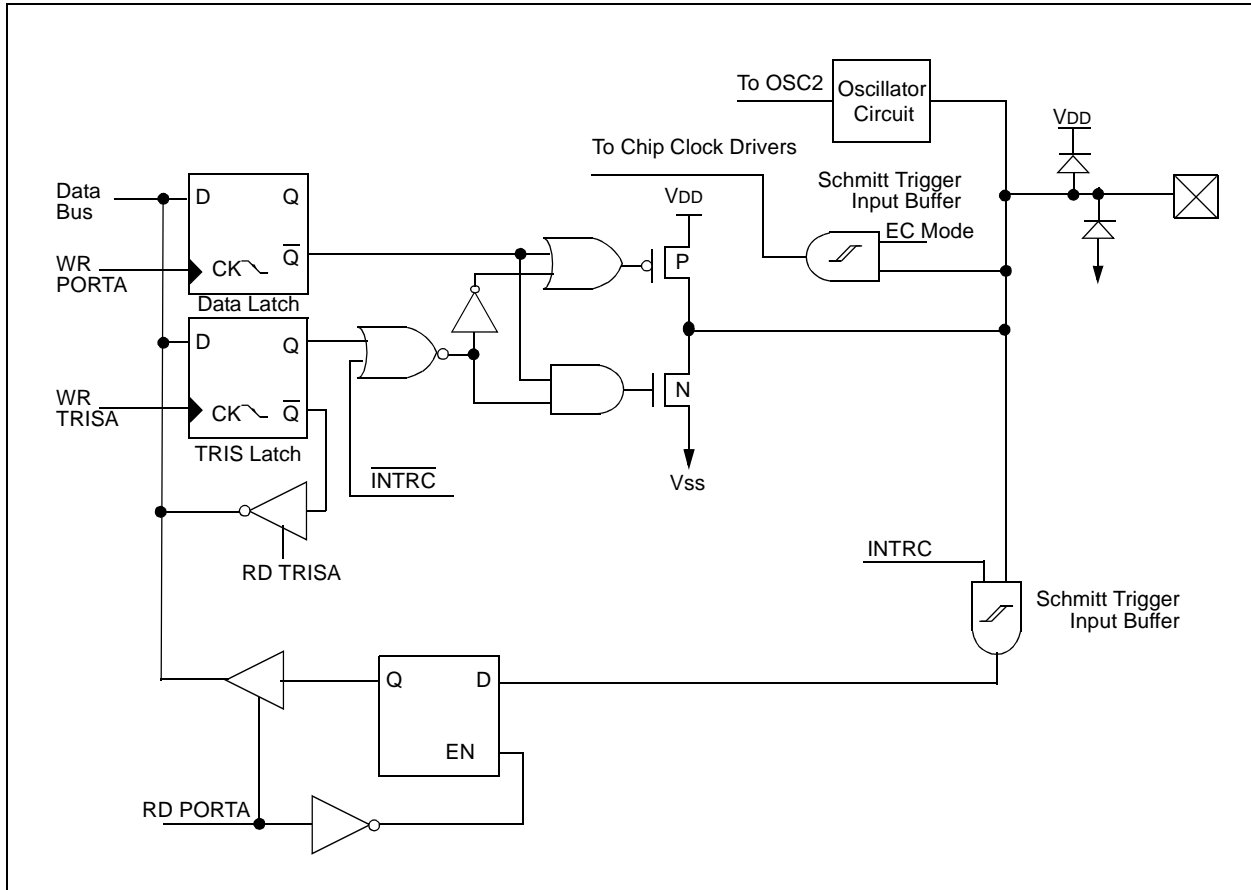


FIGURE 3-6: BLOCK DIAGRAM OF RA7/OSC1/CLKIN PIN



PIC16C717/770/771

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/MCLR/VPP | RA5 | ST | | Input port |
| | 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/AN | | 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 | --11 1111 | --11 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  0x30       ; 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 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.
- a) 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.

PIC16C717/770/771

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 |

bit 7 bit 0

bit 7-0 **WPUB<7:0>**: PORTB Weak Pull-Up Control bits

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).

Legend:

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

REGISTER 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 |

bit 7 bit 0

bit 7-0 **IOCB<7:0>**: Interrupt-on-Change PORTB Control bits

1 = Interrupt-on-change enabled
0 = Interrupt-on-change disabled

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

Legend:

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

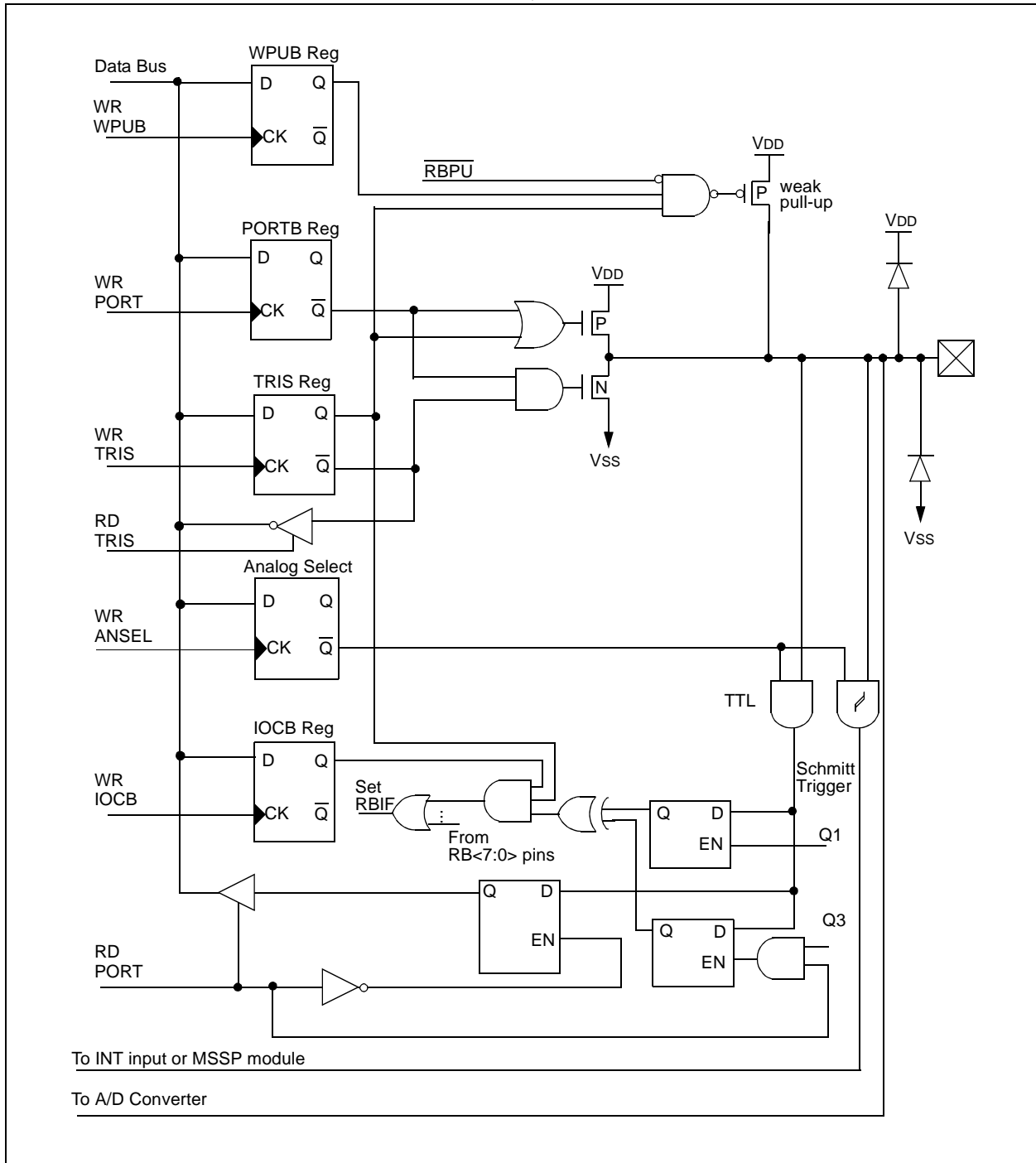
PIC16C717/770/771

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 '1'.

FIGURE 3-7: BLOCK DIAGRAM OF RB0/AN4/INT, RB1/AN5/SS PIN



PIC16C717/770/771

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

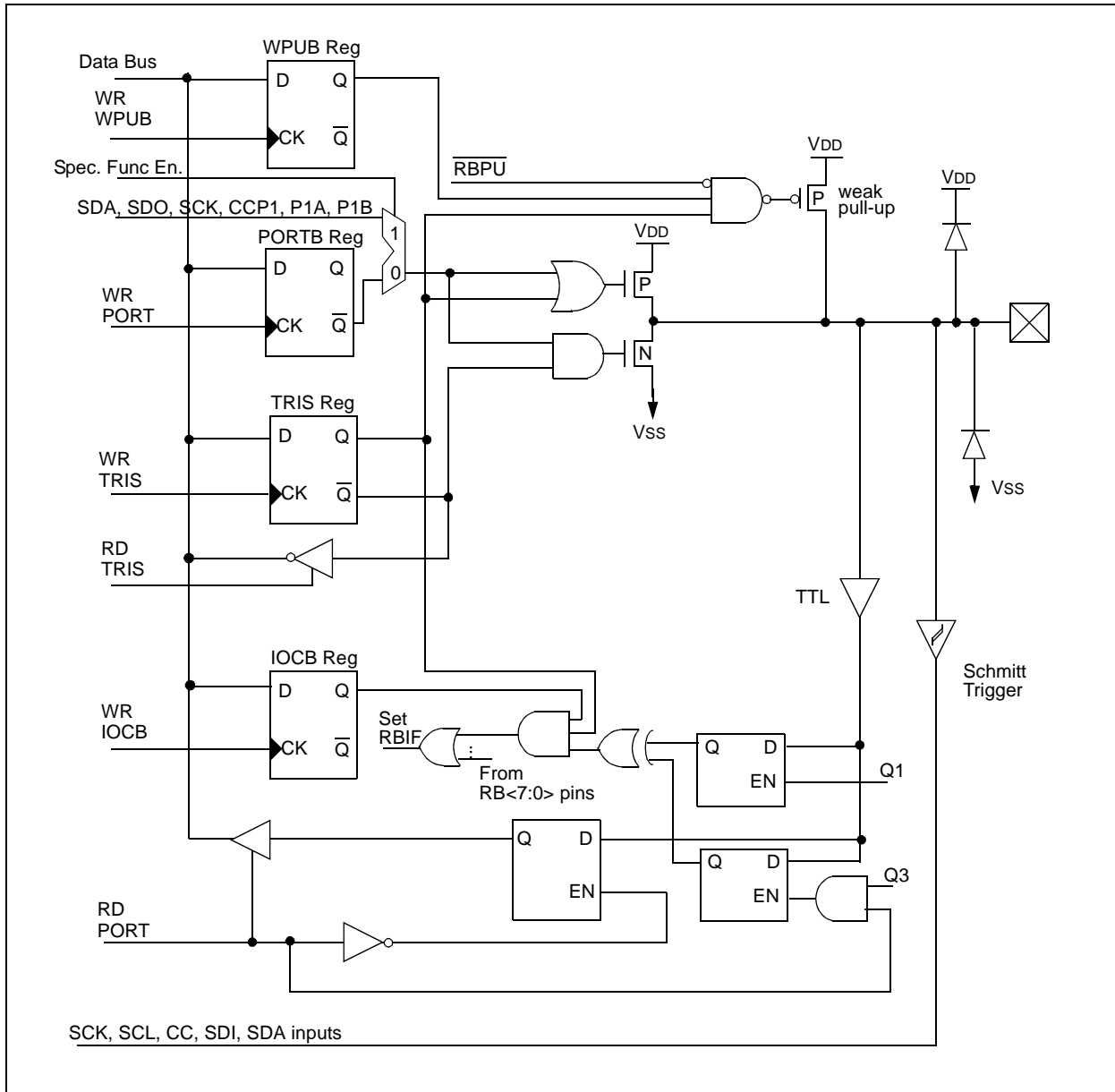
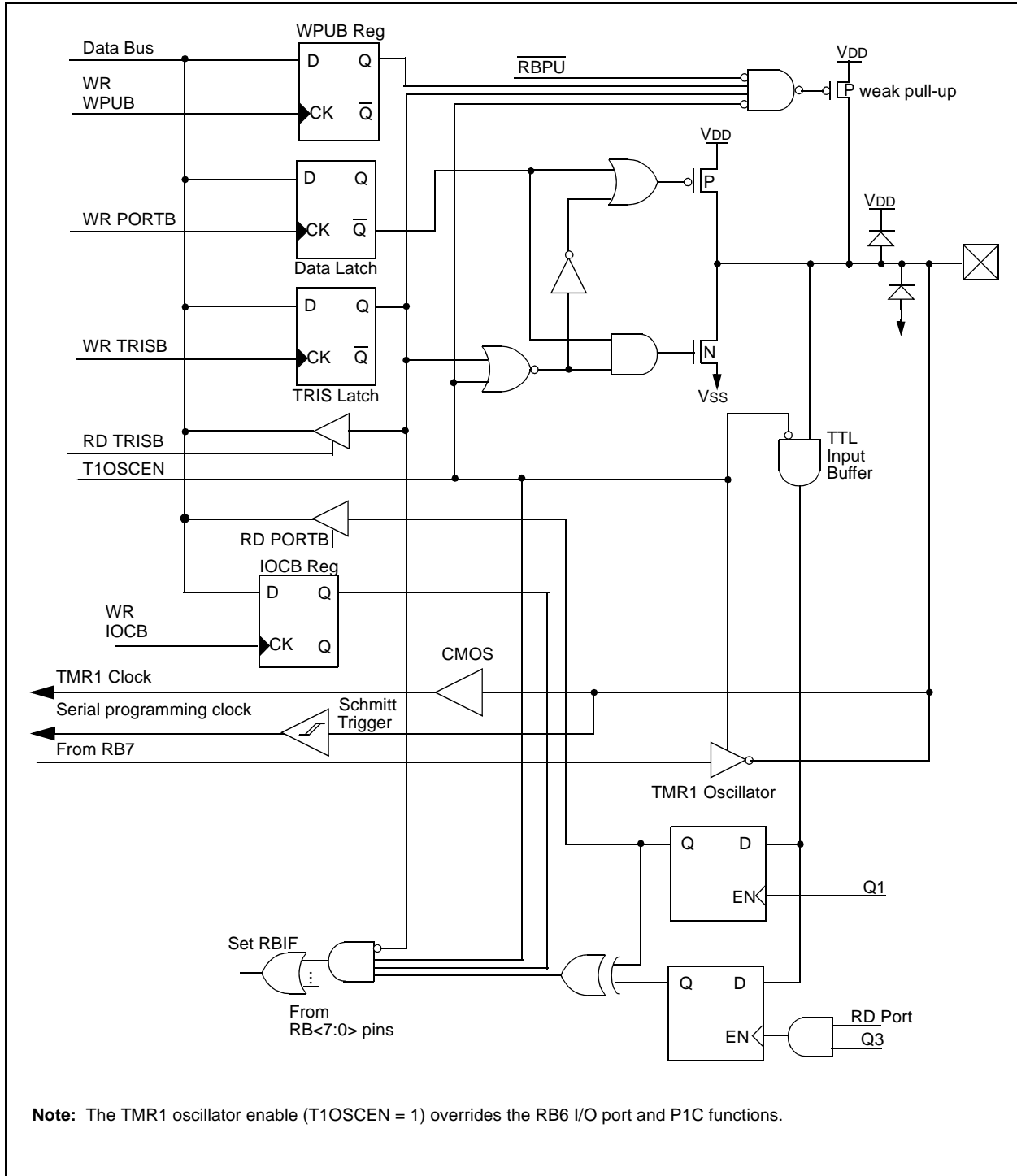


FIGURE 3-9: BLOCK DIAGRAM OF RB6/T1OSO/T1CKI/P1C



PIC16C717/770/771

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

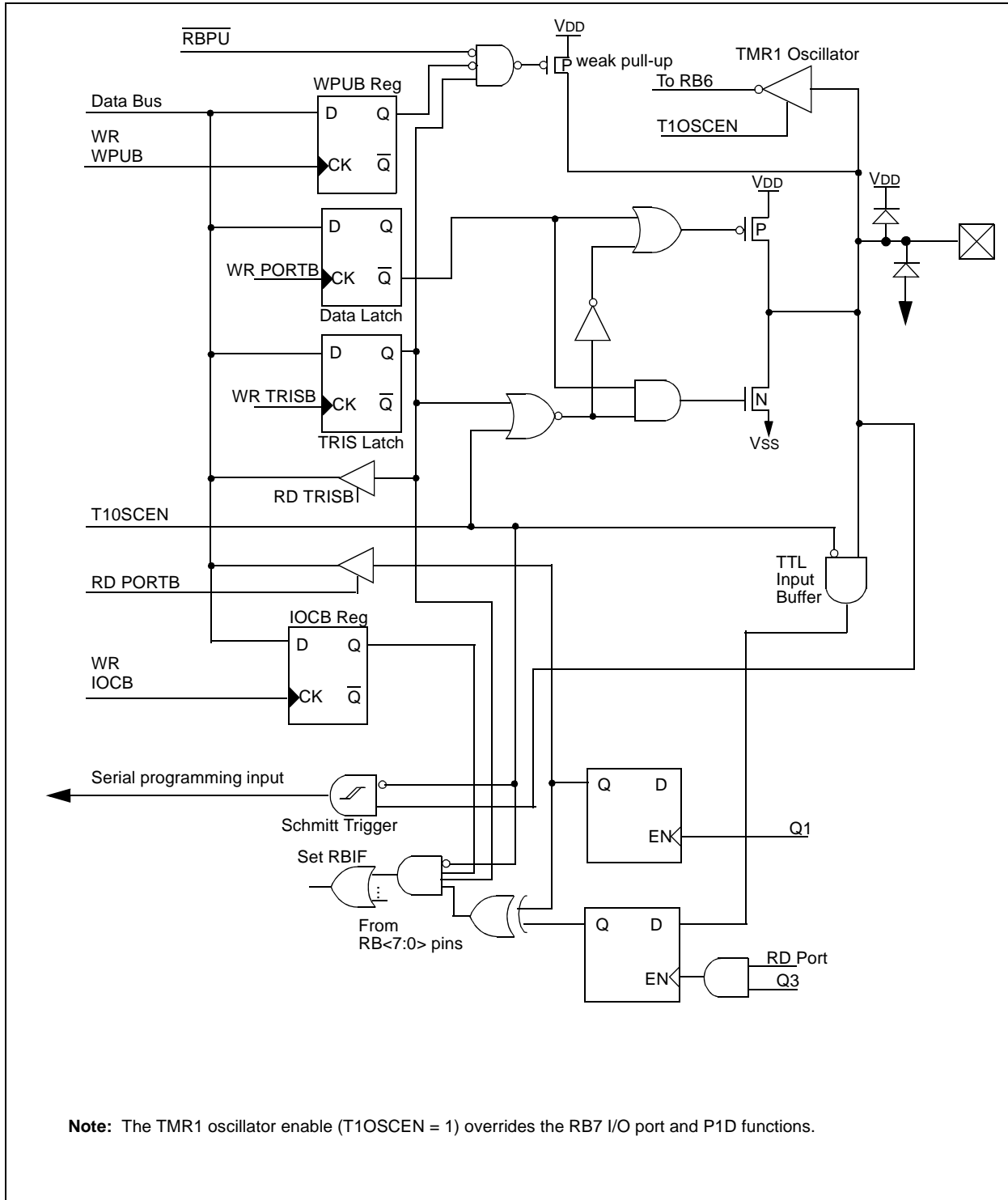


TABLE 3-3: PORTB FUNCTIONS

| 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 I/O ⁽¹⁾ |
| | SCK | ST | CMOS | Serial clock I/O for SPI |
| | SCL | ST | OD | Serial clock I/O for I ² C |
| RB3/CCP1/P1A | RB3 | TTL | CMOS | Bi-directional I/O ⁽¹⁾ |
| | CCP1 | ST | CMOS | Capture 1 input/Compare 1 output |
| | P1A | | CMOS | PWM P1A output |
| RB4/SDI/SDA | RB4 | TTL | CMOS | Bi-directional I/O ⁽¹⁾ |
| | SDI | ST | | Serial data in for SPI |
| | SDA | ST | OD | Serial data I/O for I ² C |
| RB5/SDO/P1B | RB5 | TTL | 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 | CMOS | | 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 xx11 | uuuu uu11 |
| 86h, 186h | TRISB | PORTB Data Direction Register | | | | | | | | 1111 1111 | 1111 1111 |
| 81h, 181h | OPTION_REG | 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 | --11 1111 | --11 1111 |

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

PIC16C717/770/771

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:PMADRL registers, the Most Significant bits of the PMADRH register are ignored.

4.1 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 |
| bit 7 | Reserved | — | — | — | — | — | — | RD |
| | | | | | | | | bit 0 |

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

| | | |
|--------------------|------------------|--|
| Legend: | | S = Settable (cleared in hardware) |
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' |
| - n = Value at POR | '1' = Bit is set | '0' = Bit is cleared x = Bit is unknown |

4.2 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.

PIC16C717/770/771

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 | |
| bit 7 | | | | | | | | bit 0 |

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.

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

REGISTER 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 | |
| bit 7 | | | | | | | | bit 0 |

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.

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

REGISTER 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 | |
| bit 7 | | | | | | | | bit 0 |

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

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

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

REGISTER 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 | |
| bit 7 | | | | | | | | bit 0 |

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

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

4.3 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 Program Memory Read or until it is written to by the user.

Note: The two instructions that follow setting the PMCON1 read bit must be NOPS.

EXAMPLE 4-1: OTP PROGRAM MEMORY Read

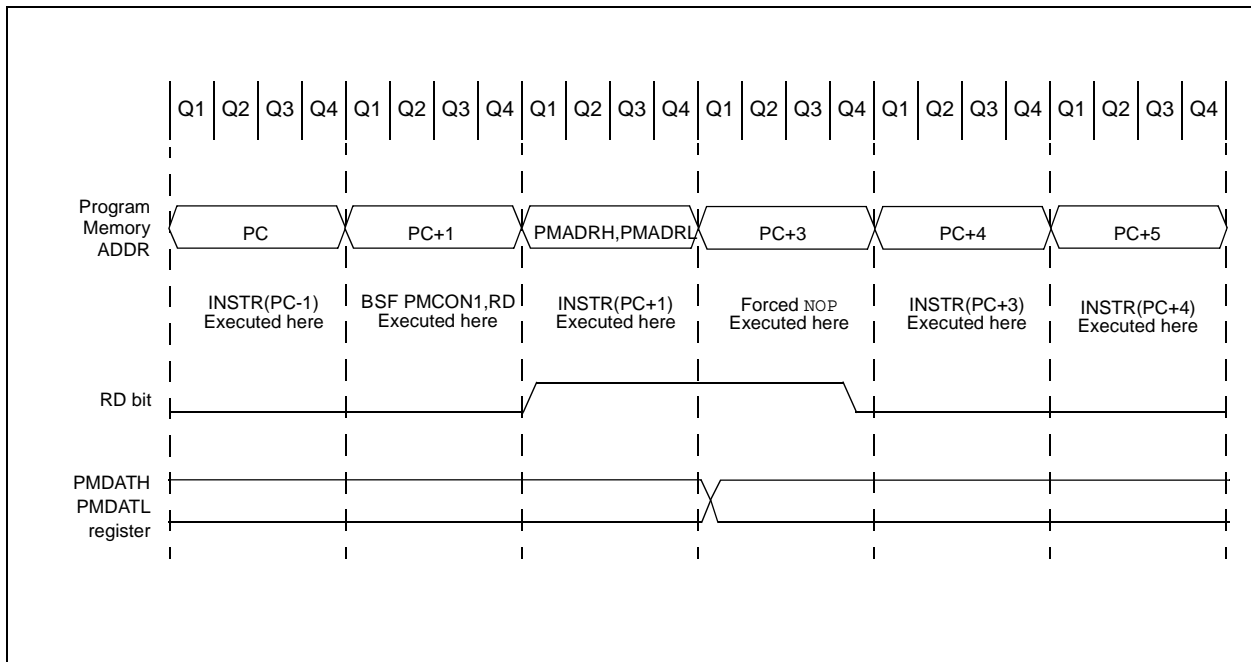
```

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 must be an NOP
NOP    ; This instruction must be an NOP
next instruction      ; PMDATH:PMDATL now has the data
    
```

4.4 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



PIC16C717/770/771

TABLE 4-1: PROGRAM MEMORY READ 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 |
|---------|--------|----------|-------|-------|-------|-------|-------|-------|-------|--------------------------|---------------------------------|
| 18Ch | PMCON1 | Reserved | — | — | — | — | — | — | RD | 1--- ---0 | 1--- ---0 |
| 10Eh | PMDATH | — | — | PMD13 | PMD12 | PMD11 | PMD10 | PMD9 | PMD8 | --xx xxxx | --uu uuuu |
| 10Ch | PMDATL | PMD7 | PMD6 | PMD5 | PMD4 | PMD3 | PMD2 | PMD1 | PMD0 | xxxx xxxx | uuuu uuuu |
| 10Fh | PMADRH | — | — | — | — | PMA11 | PMA10 | PMA9 | PMA8 | ---- xxxx | ---- uuuu |
| 10Dh | PMADRL | PMA7 | PMA6 | PMA5 | PMA4 | PMA3 | PMA2 | PMA1 | PMA0 | xxxx xxxx | uuuu uuuu |

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

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 PIC Mid-Range MCU Family 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 PIC Mid-Range MCU Family 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.

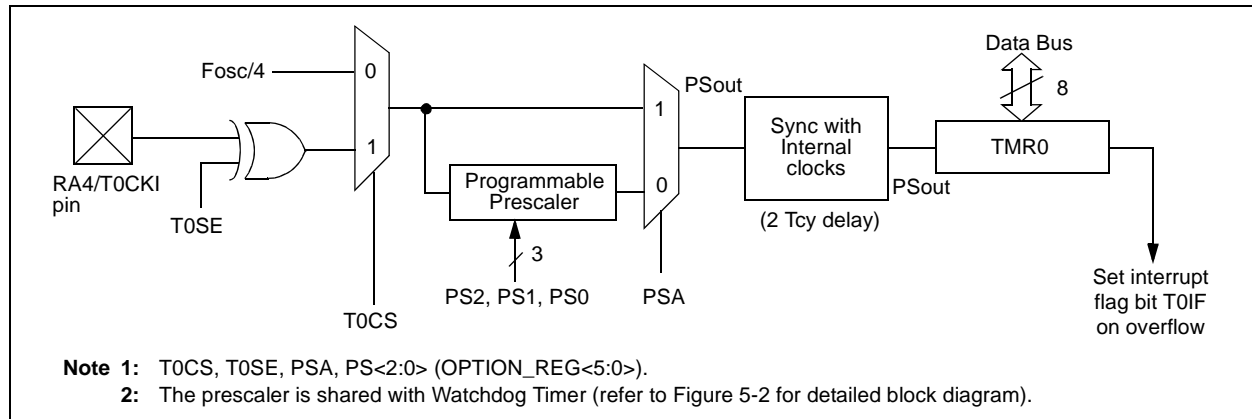
Clearing bit PSA will assign the prescaler to the Timer0 module. When the prescaler is assigned to the Timer0 module, prescale values of 1:2, 1:4, ..., 1:256 are selectable.

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



PIC16C717/770/771

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 PIC 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

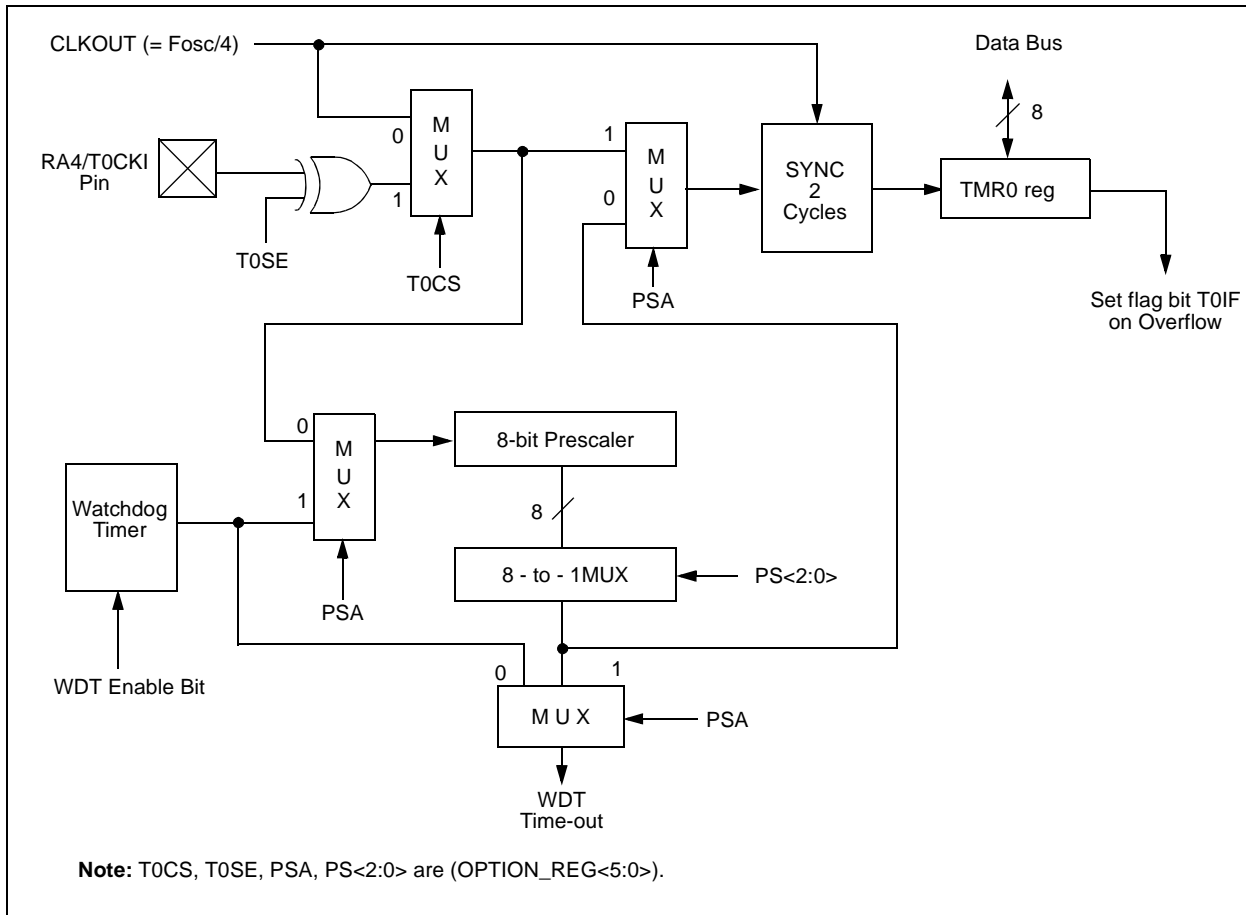


TABLE 5-1: REGISTERS ASSOCIATED WITH TIMER0

| 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 | RBP \bar{U} | 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 PIC Mid-Range MCU Family 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.

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 | T1SYNC | TMR1CS | TMR1ON |
| | bit 7 | | | | | | | bit 0 |
| 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 ⁽¹⁾ | | | | | | | |
| bit 2 | T1SYNC: Timer1 External Clock Input Synchronization Control bit <u>TMR1CS = 1:</u> 1 = Do not synchronize external clock input 0 = Synchronize external clock input <u>TMR1CS = 0:</u> 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 | | | | | | | |

Note 1: The oscillator inverter and feedback resistor are turned off to eliminate power drain.

Legend:

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

PIC16C717/770/771

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

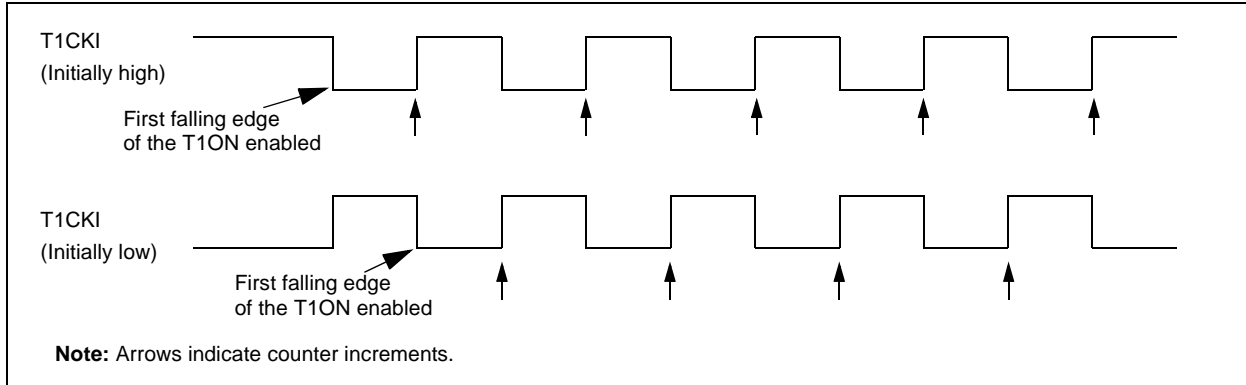
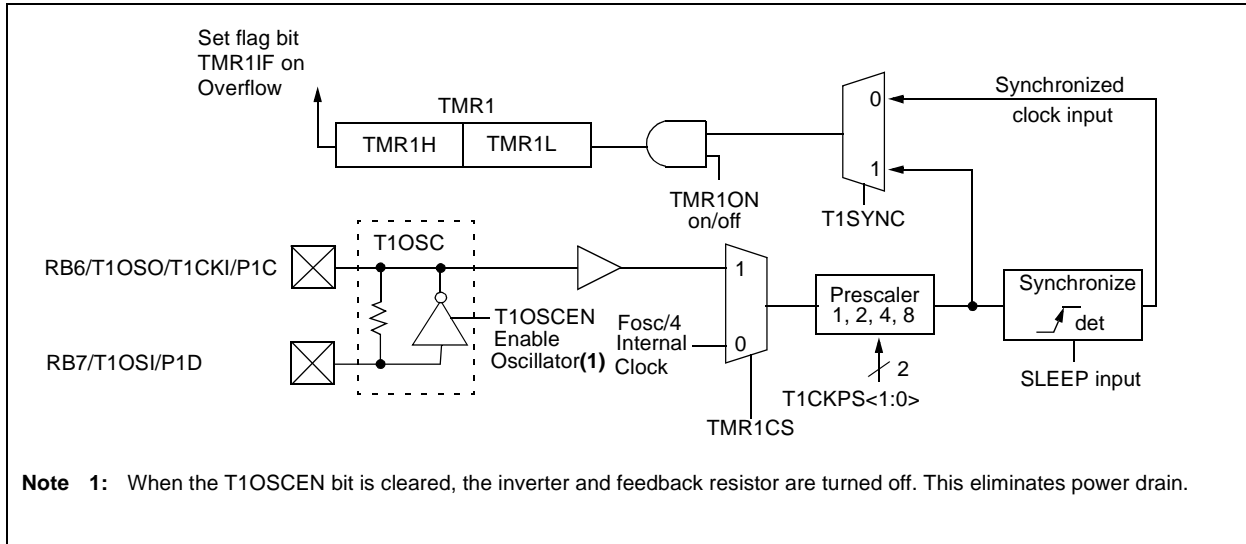


FIGURE 6-2: TIMER1 BLOCK DIAGRAM



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 ECCP, 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 | TOIE | INTE | RBIE | TOIF | 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 | T1SYNĀ | TMR1CS | TMR1ON | --00 0000 | --uu uuuu |

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

PIC16C717/770/771

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 PIC Mid-Range MCU Family 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 ($F_{osc}/4$) 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{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 |
| bit 7 | | | | | | | bit 0 |

- 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 **T2CKPS<1:0>:** Timer2 Clock Prescale Select bits
 00 = Prescaler is 1
 01 = Prescaler is 4
 1x = Prescaler is 16

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

PIC16C717/770/771

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

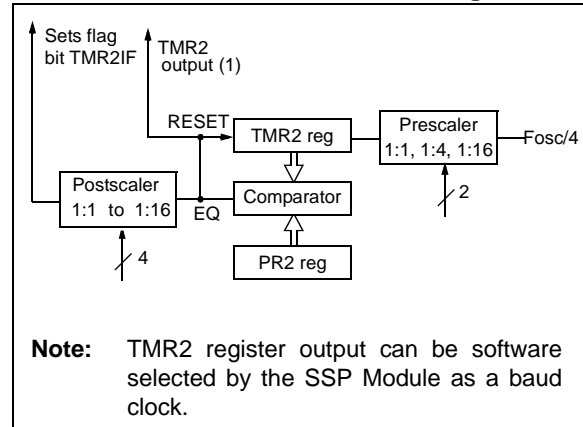


TABLE 7-1: REGISTERS ASSOCIATED WITH TIMER2 AS A TIMER/COUNTER

| 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 | TOIE | INTE | RBIE | TOIF | 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.

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 |
| bit 7 | | | | | | | bit 0 |

- bit 7-6 **PWM1M<1:0>**: PWM Output Configuration
 CCP1M<3:2> = 00, 01, 10
 xx = P1A assigned as Capture input, Compare output. P1B, P1C, P1D assigned as Port pins.
 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>**: ECCP 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.

Legend:

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

PIC16C717/770/771

TABLE 8-1: ECCP MODE - TIMER RESOURCE

| ECCP 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 three prescaler settings, specified by bits CCP1M<3:0>. Whenever the ECCP module is turned off or the ECCP module is not in Capture mode, the prescaler counter is cleared. This means that any RESET will clear the prescaler counter.

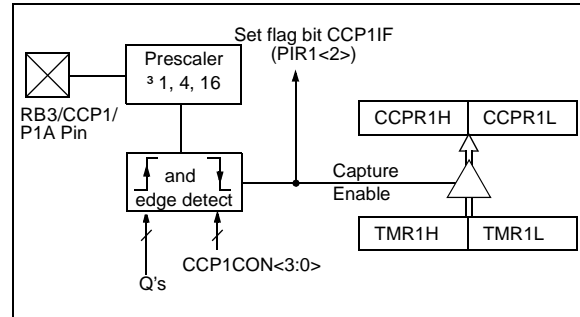
Switching from one capture prescaler to another may generate an interrupt. Also, the prescaler counter will not be cleared, therefore the first capture may be from a non-zero prescaler. Example 8-1 shows the recommended method for switching between capture prescalers. This example also clears the prescaler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

EXAMPLE 8-1: Changing Between Capture Prescalers

```

CLRF  CCP1CON    ; 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.

Changing the ECCP mode select bits to the clear output on Match mode (CCP1M<3:0> = “1000”) presets the CCP1 output latch to the logic 1 level. Changing the ECCP mode select bits to the clear output on Match mode (CCP1M<3:0> = “1001”) presets the CCP1 output latch to the logic 0 level.

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

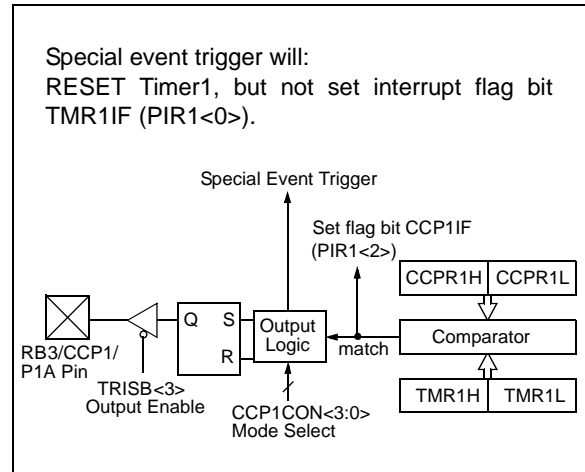


TABLE 8-2: REGISTERS ASSOCIATED WITH CAPTURE, COMPARE AND TIMER1

| 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 TMR1 register | | | | | | | | xxxx xxxx | uuuu uuuu |
| T1CON | — | — | T1CKPS 1 | T1CKP S0 | T1OSCEN | T1SYNC | TMR1CS | TMR1O N | --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 |

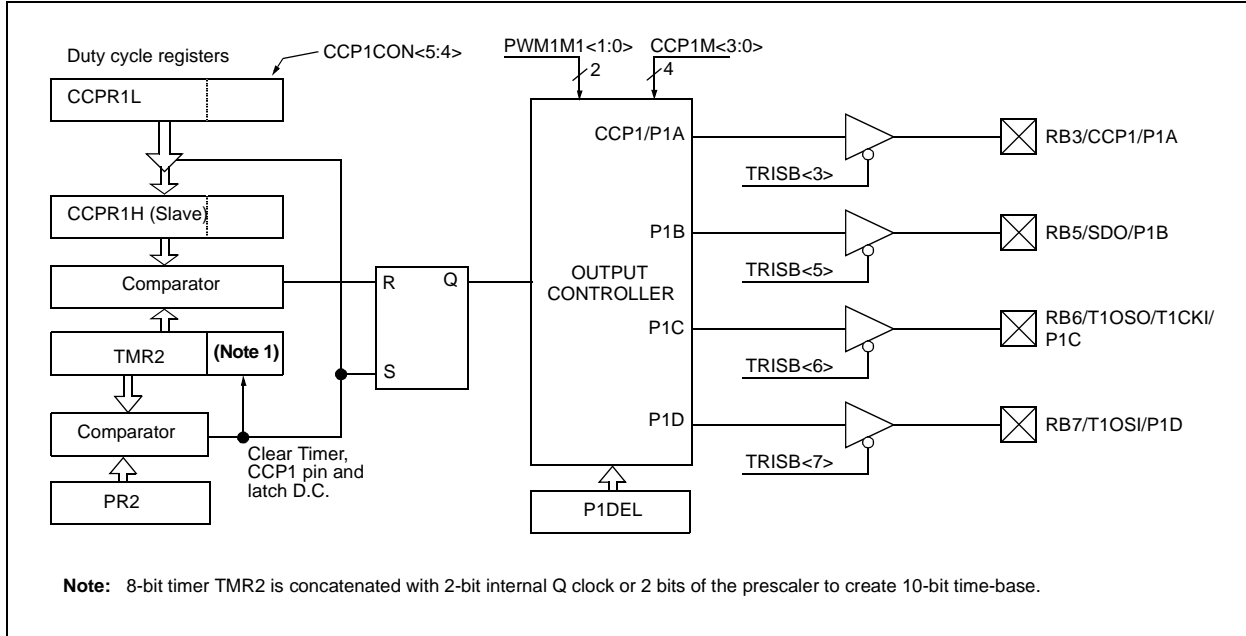
Legend: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by Capture and Timer1.

PIC16C717/770/771

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



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:

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

PWM frequency is defined as $1 / [\text{PWM period}]$.

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.

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 CCP1L:CCP1CON<5:4>. The following equation is used to calculate the PWM duty cycle in time:

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

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:

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

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

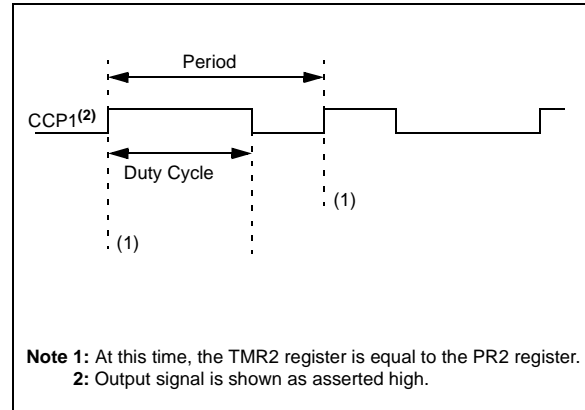
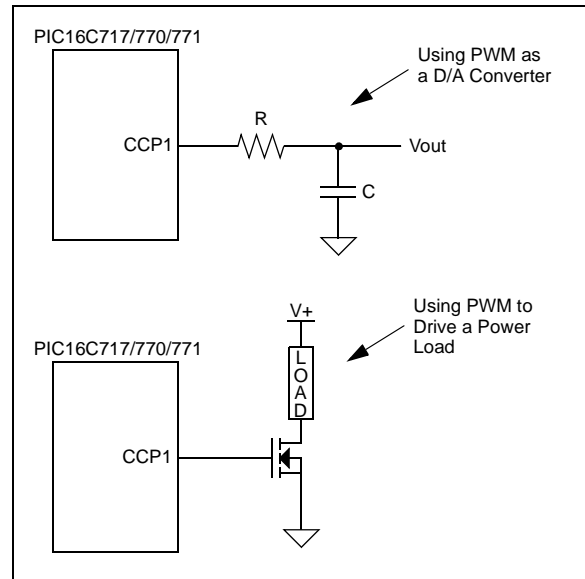


FIGURE 8-5: EXAMPLE OF SINGLE OUTPUT APPLICATION



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 dead-band 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.

PIC16C717/770/771

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. Changing 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

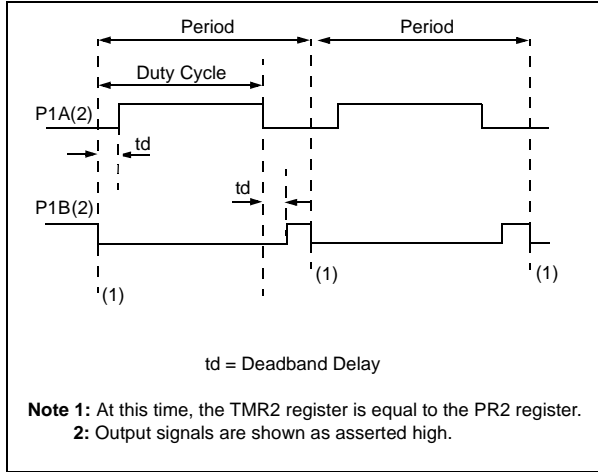
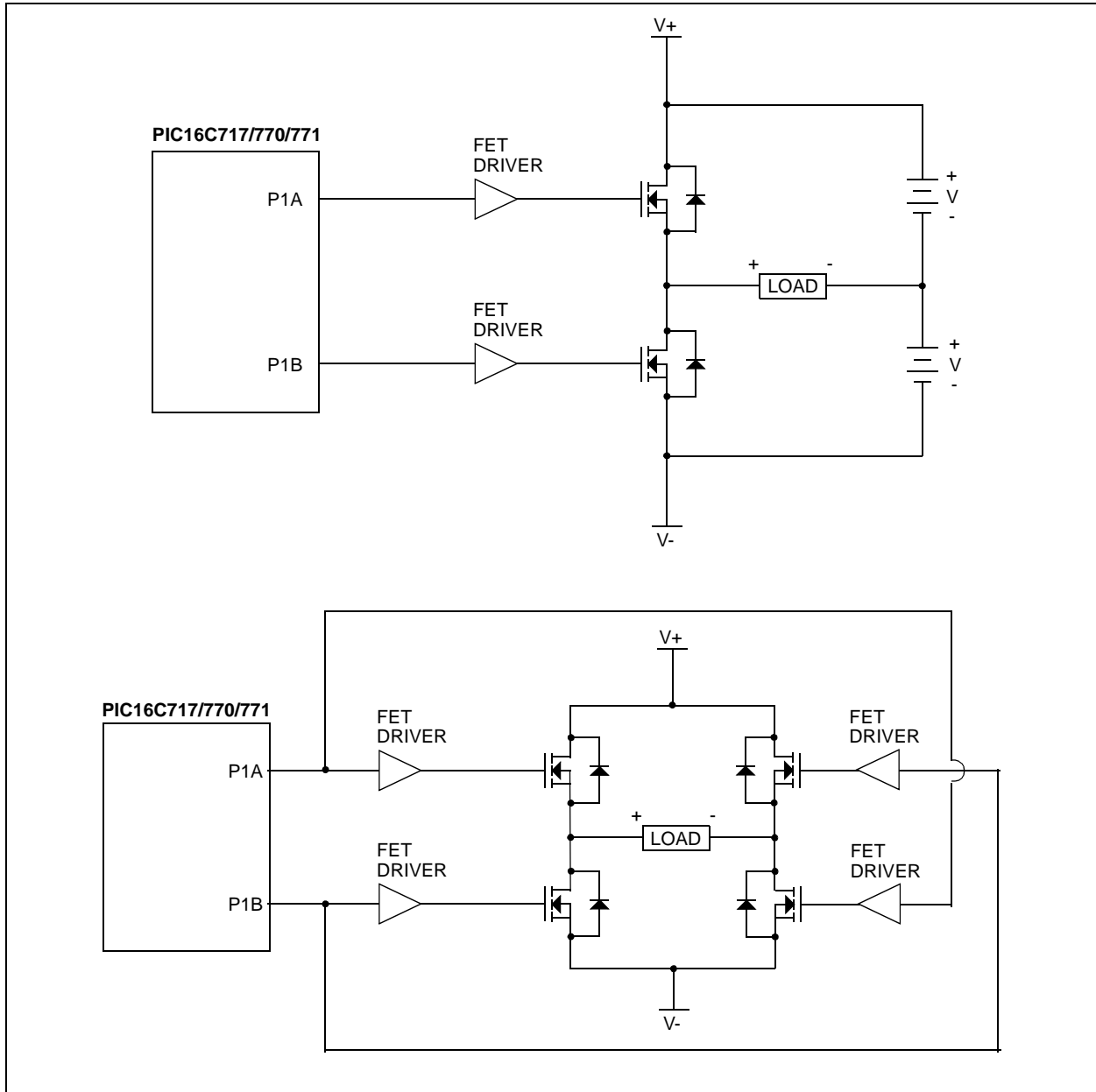


FIGURE 8-7: EXAMPLE OF HALF-BRIDGE OUTPUT MODE APPLICATIONS



PIC16C717/770/771

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

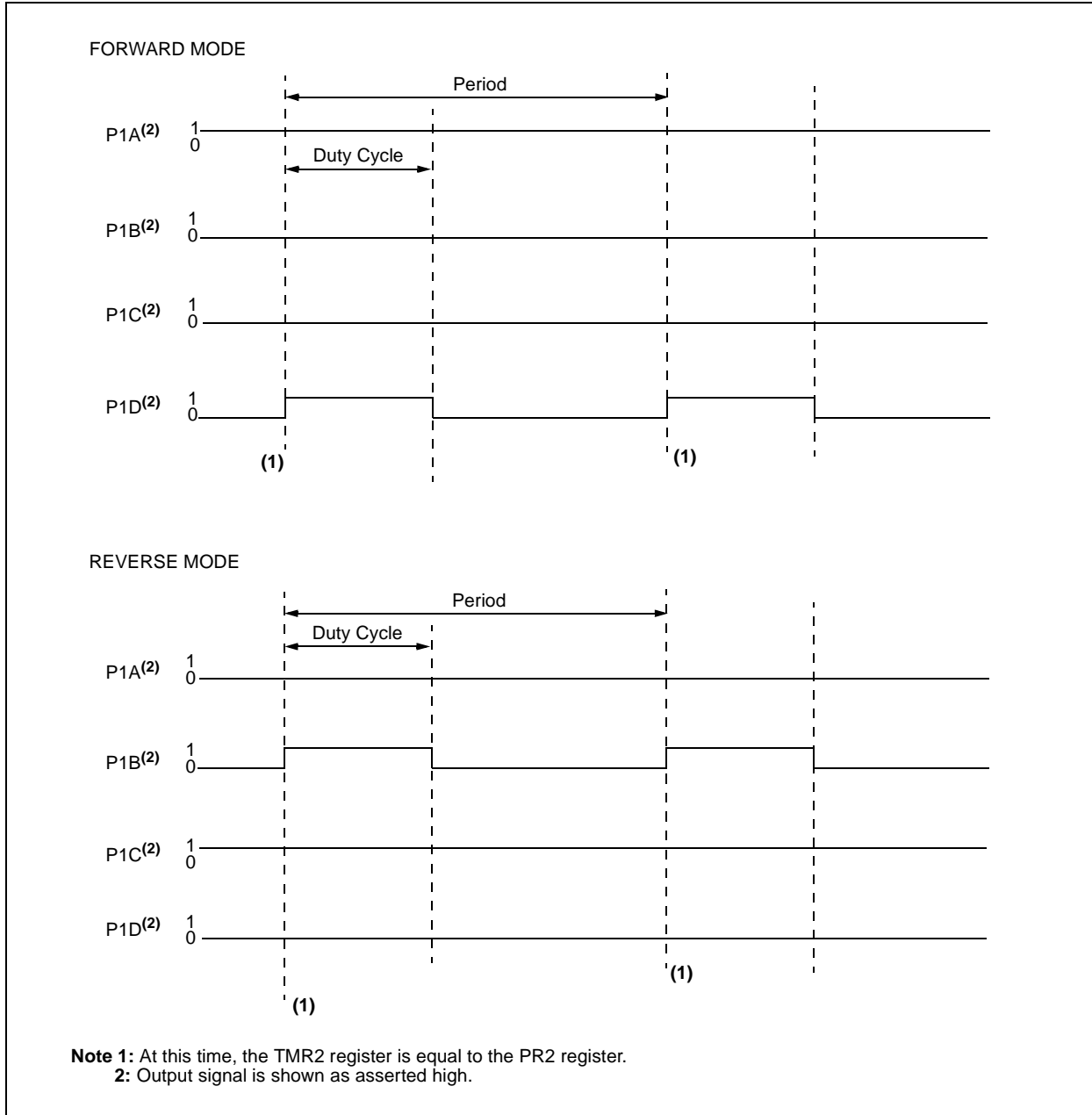
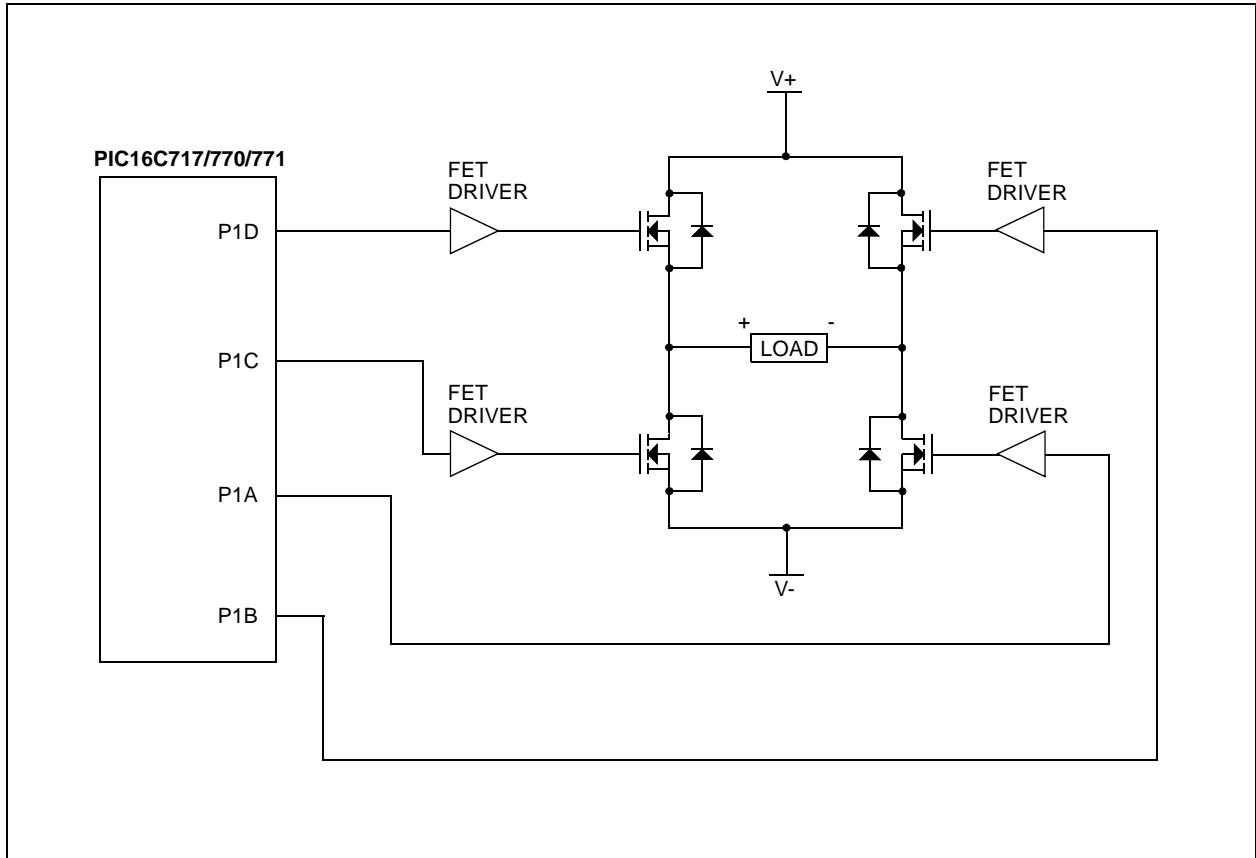


FIGURE 8-9: EXAMPLE OF FULL-BRIDGE APPLICATION



PIC16C717/770/771

8.3.5 PROGRAMMABLE DEADBAND DELAY

In half-bridge or full-bridge applications, driven by half-bridge outputs (see Figure 8-7), 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 |
| P1DEL7 | P1DEL6 | P1DEL5 | P1DEL4 | P1DEL3 | P1DEL2 | P1DEL1 | P1DEL0 |
| bit 7 | | | | | | | bit 0 |

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.

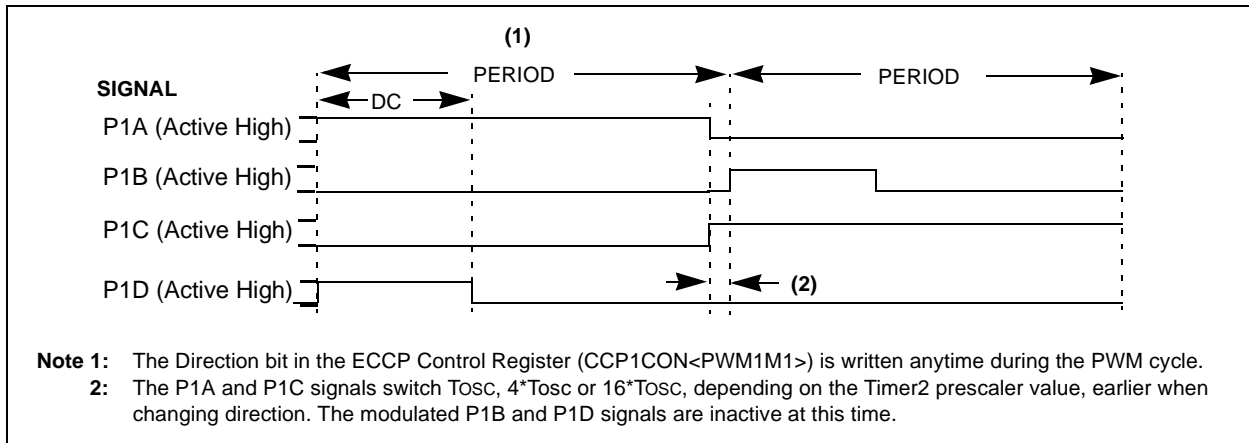
| | | | |
|--------------------|------------------|------------------------------------|--------------------|
| Legend: | | | |
| R = Readable bit | W = Writable bit | U = Unimplemented bit, read as '0' | |
| - n = Value at POR | '1' = Bit is set | '0' = Bit is cleared | x = Bit is unknown |

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 prescale 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 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 a 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.

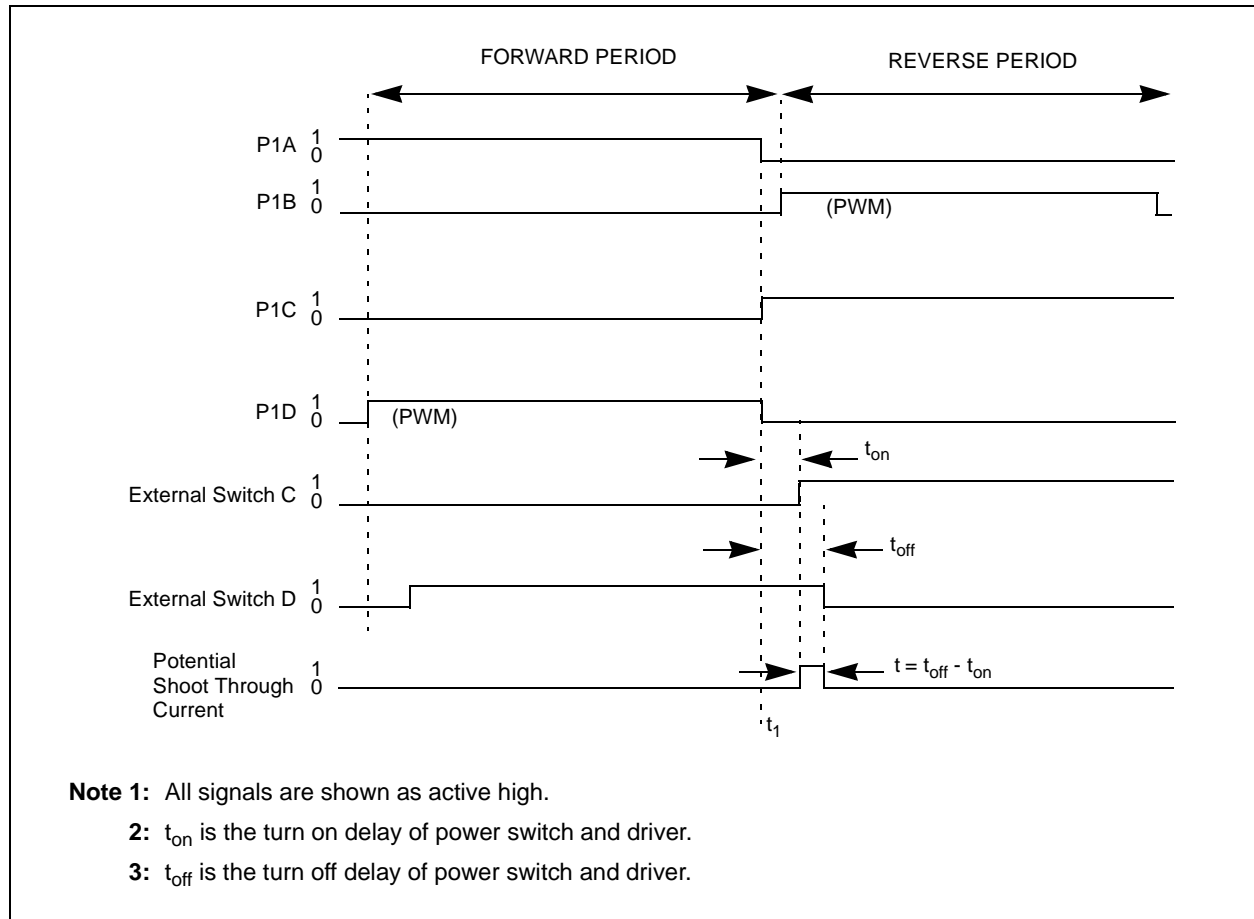
Figure 8-11 shows an example, where the PWM direction changes from forward to reverse at a near 100% duty cycle. At time t_1 , the output P1A and P1D become inactive, while output P1C becomes active. In this

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 for 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



PIC16C717/770/771

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 | TOIE | INTE | RBIE | TOIF | 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: x = unknown, u = unchanged, - = unimplemented read as '0'. Shaded cells are not used by ECCP module in PWM mode.

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 (I²C™)

PIC16C717/770/771

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/A | P | S | R/W | UA | BF |
| bit 7 | | | | | | bit 0 | |

| | |
|-------|--|
| bit 7 | <p>SMP: Sample bit <u>SPI Master Mode</u> 1 = Input data sampled at end of data output time 0 = Input data sampled at middle of data output time <u>SPI Slave Mode</u> SMP must be cleared when SPI is used in Slave mode In I²C 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)</p> |
| bit 6 | <p>CKE: SPI Clock Edge Select (Figure 9-3, Figure 9-5, and Figure 9-6) <u>CKP = 0</u> 1 = Data transmitted on rising edge of SCK 0 = Data transmitted on falling edge of SCK <u>CKP = 1</u> 1 = Data transmitted on falling edge of SCK 0 = Data transmitted on rising edge of SCK</p> |
| bit 5 | <p>D/A: Data/Address bit (I²C mode only) 1 = Indicates that the last byte received or transmitted was data 0 = Indicates that the last byte received or transmitted was address</p> |
| bit 4 | <p>P: STOP bit (I²C 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</p> |
| bit 3 | <p>S: START bit (I²C 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</p> |
| bit 2 | <p>R/W: Read/Write bit information (I²C 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 NACK bit. <u>In I²C Slave mode:</u> 1 = Read 0 = Write <u>In I²C Master mode:</u> 1 = Transmit is in progress 0 = Transmit is not in progress. ORing this bit with SEN, RSEN, PEN, RCEN, or AKEN will indicate if the MSSP is in IDLE mode</p> |
| bit 1 | <p>UA: Update Address (10-bit I²C mode only) 1 = Indicates that the user needs to update the address in the SSPADD register 0 = Address does not need to be updated</p> |
| bit 0 | <p>BF: Buffer Full Status bit <u>Receive (SPI and I²C modes)</u> 1 = Receive complete, SSPBUF is full 0 = Receive not complete, SSPBUF is empty <u>Transmit (I²C mode only)</u> 1 = Data Transmit in progress (does not include the ACK and STOP bits), SSPBUF is full 0 = Data Transmit complete (does not include the ACK and STOP bits), SSPBUF is empty</p> |

Legend:

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

PIC16C717/770/771

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 | |
| bit 7 | | | | | | | | bit 0 |

bit 7 **WCOL:** Write Collision Detect bit

Master Mode:

1 = A write to the SSPBUF register was attempted while the I²C 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 I²C 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, the I/O 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 I²C 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 I²C Slave mode SCK release control

1 = Enable clock

0 = Holds clock low (clock stretch) (used to ensure data setup time)

In I²C Master mode

Unused in this mode

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

- n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

PIC16C717/770/771

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

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 = I²C Slave mode, 7-bit address
- 0111 = I²C Slave mode, 10-bit address
- 1000 = I²C Master mode, clock = FOSC / (4 • (SSPADD+1))
- 1001 = Reserved
- 1010 = Reserved
- 1011 = Firmware controlled Master mode (slave idle)
- 1100 = Reserved
- 1101 = Reserved
- 1110 = 7-bit Slave mode with START and STOP condition interrupts
- 1111 = 10-bit Slave mode with START and STOP condition interrupts

Legend:

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

REGISTER 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 |

bit 7

bit 0

- bit 7 **GCEN:** General Call Enable bit (In I²C 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 I²C 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 I²C 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 (NACK)
 0 = Acknowledge (ACK)
- bit 4 **ACKEN:** Acknowledge Sequence Enable bit (In I²C 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 I²C Master mode only).
 1 = Enables Receive mode for I²C
 0 = Receive IDLE
- bit 2 **PEN:** STOP Condition Enable bit (In I²C 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 I²C 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 I²C 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 I²C module is not in the IDLE mode, this bit may not be set (no spooling) and the SSPBUF may not be written (or writes to the SSPBUF are disabled).

Legend:

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

PIC16C717/770/771

9.1 SPI Mode

The SPI mode allows eight 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})

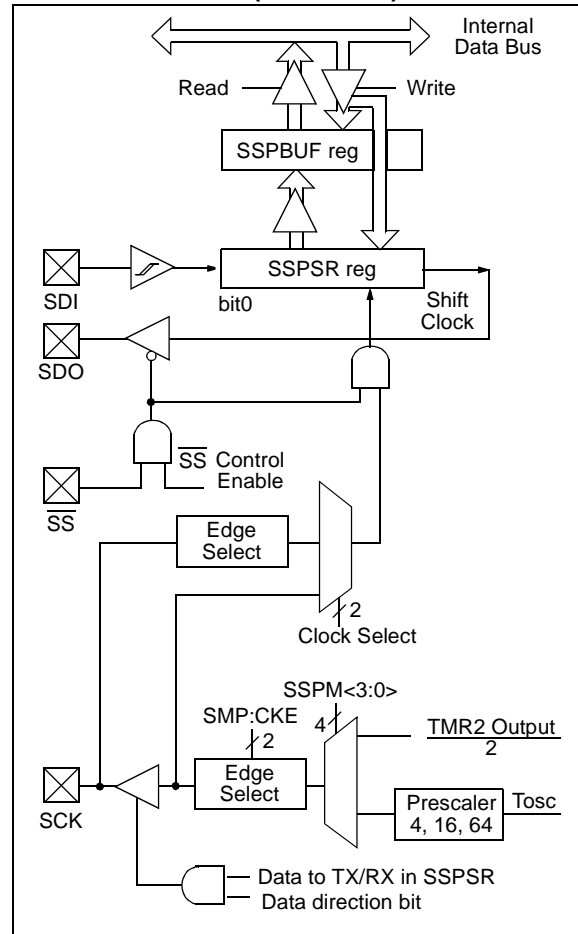
9.1.1 OPERATION

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 eight 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
                                ;(xmit complete)?
GOTO LOOP ;No
BCF STATUS, RP0 ;Specify Bank 0
MOVF SSPBUF, W ;Save SSPBUF...
MOVWF RXDATA ;...in user RAM
MOVF TXDATA, W ;Get next 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 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
- SS must have TRISB<1> set, and ANSEL<5> cleared

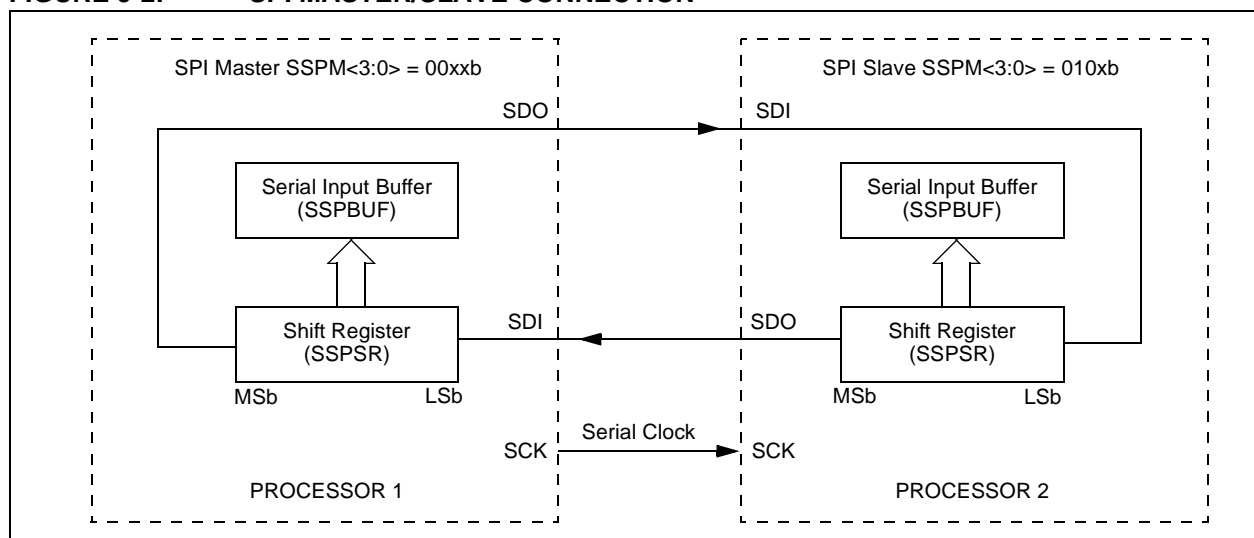
Any serial port function that is not desired may be overridden by programming the corresponding data direction (TRIS) register to the opposite value.

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 (SSPCON<4>), 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



PIC16C717/770/771

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

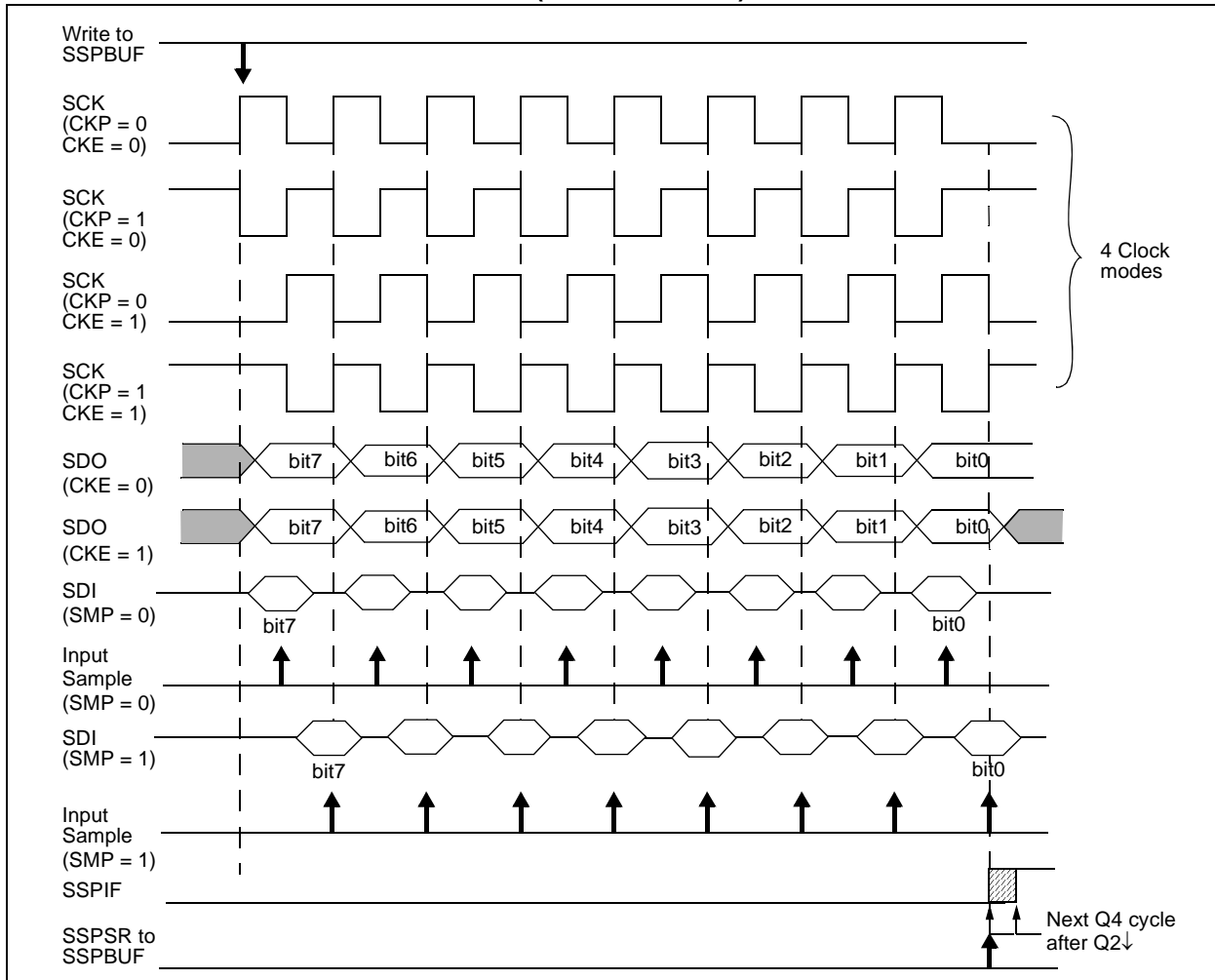
Figure 9-3, Figure 9-5 and Figure 9-6, where the MSb is transmitted first. In Master mode, the SPI clock rate (bit rate) is user programmable to be one of the following:

- $F_{OSC}/4$ (or T_{CY})
- $F_{OSC}/16$ (or $4 \cdot T_{CY}$)
- $F_{OSC}/64$ (or $16 \cdot T_{CY}$)
- $\text{Timer2 output}/2$

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 SSIPIF (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.

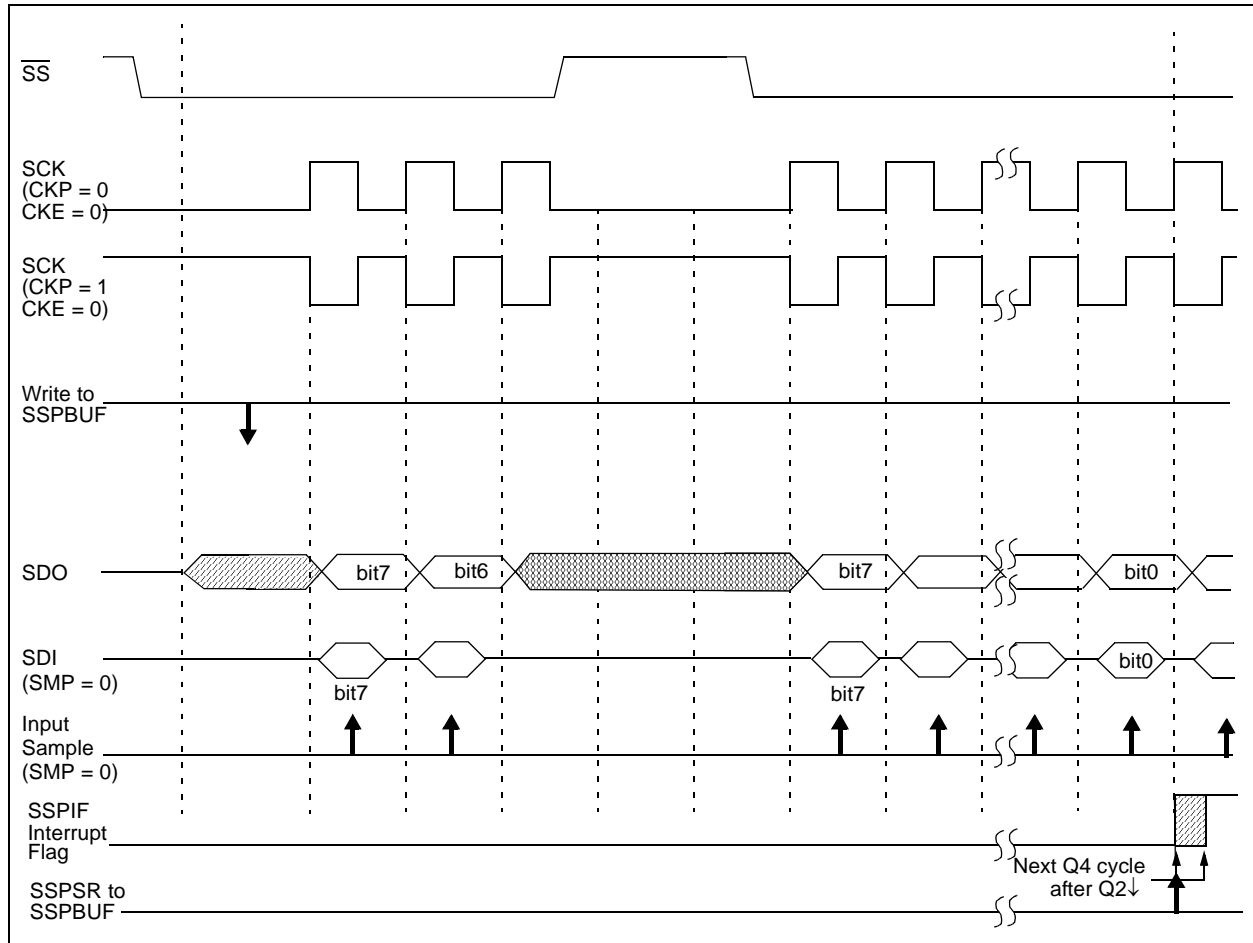
Note 1: When the SPI module is in Slave mode with \overline{SS} pin control enabled, (SSPCON<3:0> = 0100) the SPI module will RESET if the \overline{SS} pin is set to VDD.

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



PIC16C717/770/771

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

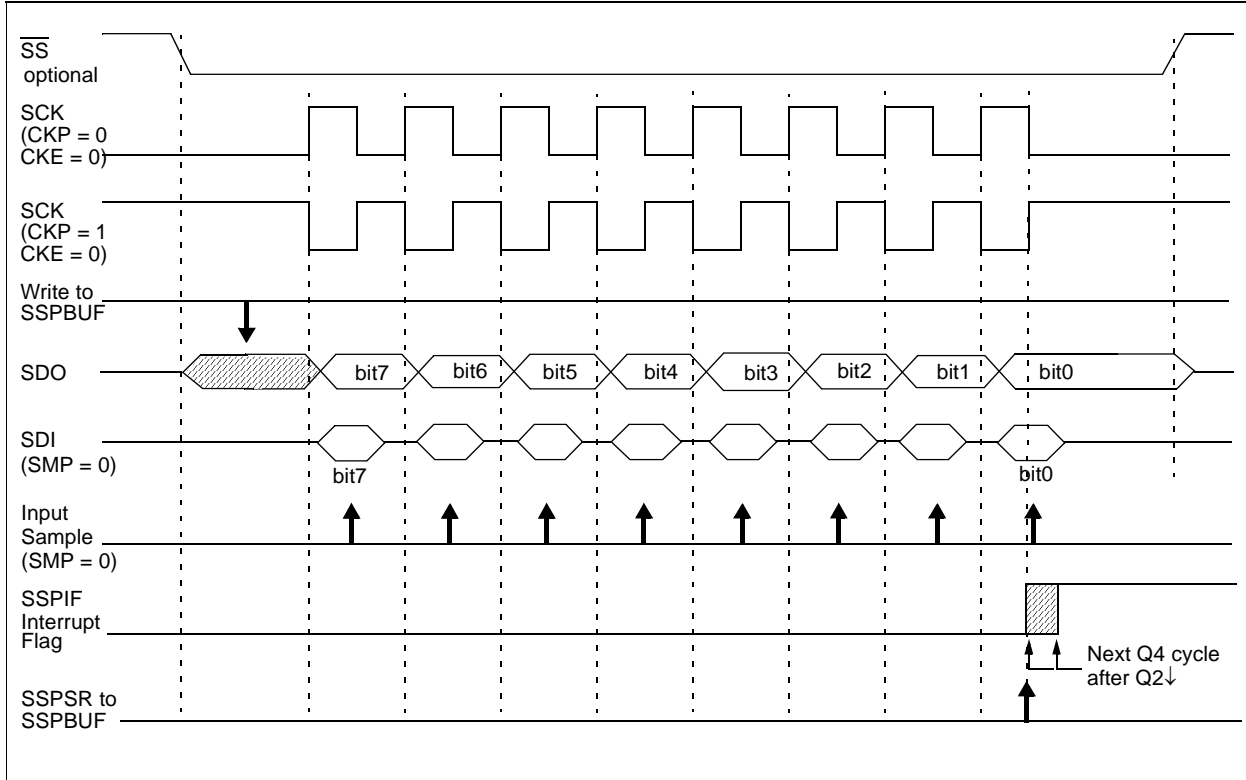
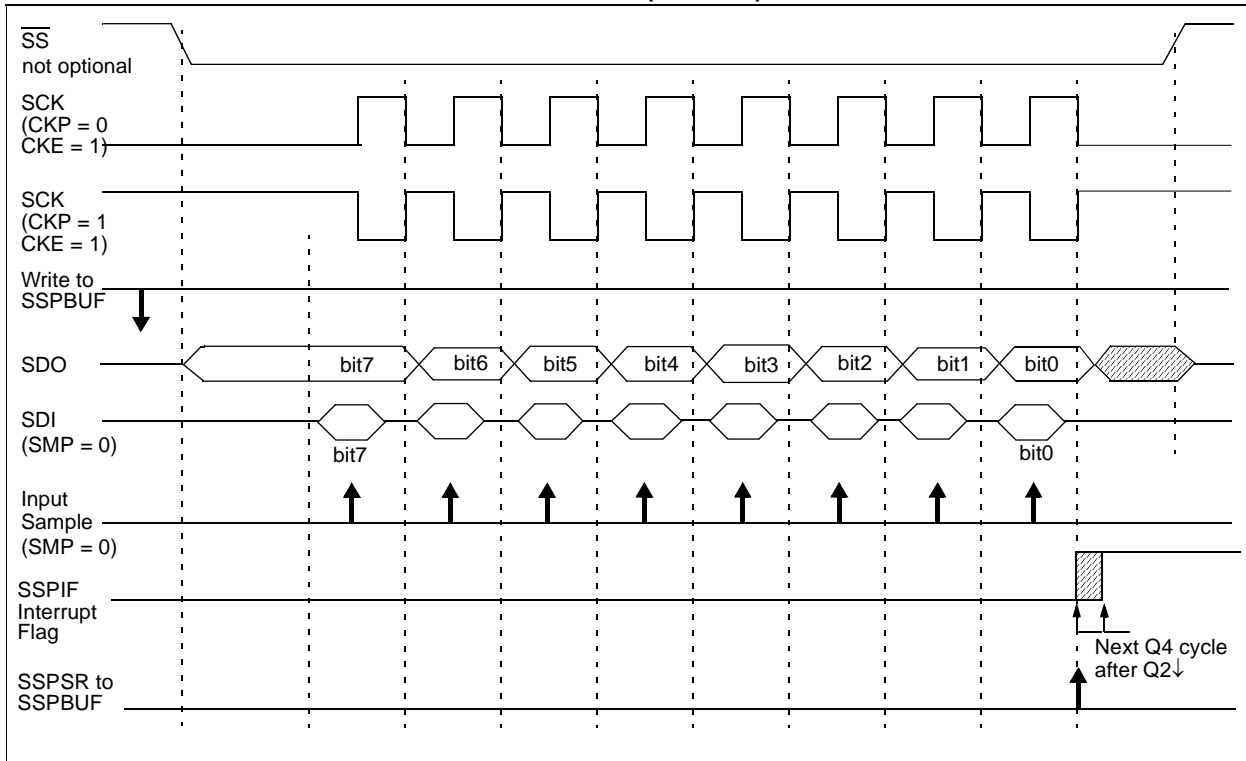


FIGURE 9-6: SPI SLAVE MODE WAVEFORM (CKE = 1)



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 eight bits have been received, the SSPIF 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 | MCLR, WDT |
|------------------------|---------|--|-------|-------|-------|-------|--------|--------|--------|------------|-----------|
| 0Bh, 8Bh, 10Bh,18Bh | INTCON | GIE | PEIE | TOIE | INTE | RBIE | TOIF | 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 xxxxx | uuuu uuuu |
| 14h | SSPCON | WCOL | SSPOV | SSPEN | CKP | SSPM3 | SSPM2 | SSPM1 | SSPM0 | 0000 0000 | 0000 0000 |
| 94h | SSPSTAT | SMP | CKE | D/A | P | S | R/W | UA | BF | 0000 0000 | 0000 0000 |
| 9Dh | ANSEL | | | | | | | | | --11 1111 | --11 1111 |
| 86h | TRISB | | | | | | | | | 1111 1111 | 1111 1111 |

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

PIC16C717/770/771

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 when the bus is free (multi-master function). The MSSP module implements the Standard mode specifications, as well as 7-bit and 10-bit addressing.

Two pins are used to transfer data. They are the SCL pin (clock) and the SDA pin (data). The MSSP module functions are enabled by setting SSP Enable bit SSPEN (SSPCON<5>). The SCL and SDA pins are "glitch" filtered when operating as inputs. This filter functions in both the 100 kHz and 400 kHz modes. When these pins operate as outputs in the 100 kHz mode, there is a slow rate control of the pin that is independent of device frequency.

Before selecting any I²C mode, the SCL and SDA pins must be programmed as inputs by setting the appropriate TRIS bits. This allows the MSSP module to configure and drive the I/O pins as required by the I²C protocol.

The MSSP module has six registers for I²C operation. They are listed below.

- 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 for control of the I²C operation. Four mode selection bits (SSPCON<3:0>) configure the MSSP as any one of the following I²C modes:

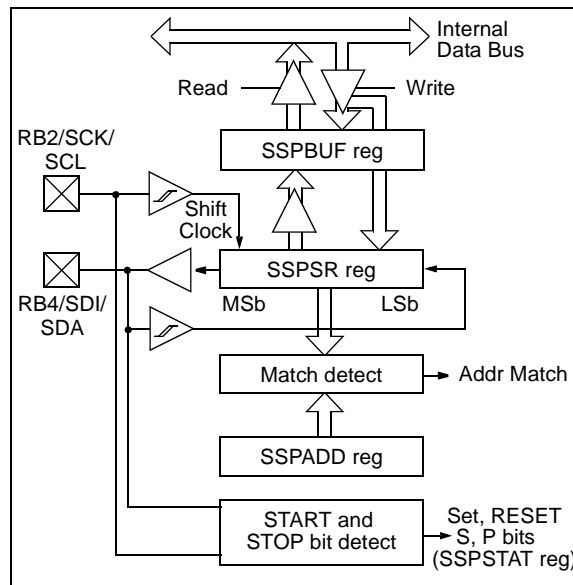
- I²C Slave mode (7-bit address)
- I²C Slave mode (10-bit address)
- I²C Master mode
SCL Freq = FOSC / [4 • (SSPADD + 1)]
- I²C Slave mode with START and STOP interrupts (7-bit address)
- I²C Slave mode with START and STOP interrupts (10-bit address)
- Firmware Controlled Master mode

The SSPSTAT register gives the status of the data transfer. This information includes detection of a START (S) or STOP (P) bit. It specifies whether 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, and from which the transfer data is read. 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 from the SSPSR register 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 occurs, in which case, the SSPOV bit (SSPCON<6>) is set and the byte in the SSPSR is lost.

FIGURE 9-7: I²C SLAVE MODE BLOCK DIAGRAM



9.2.1 UPWARD COMPATIBILITY WITH SSP MODULE

The MSSP module includes three SSP modes of operation to maintain upward compatibility with the SSP module. These modes are:

- Firmware controlled Master mode (slave idle)
- 7-bit Slave mode with START and STOP condition interrupts.
- 10-bit Slave mode with START and STOP condition interrupts.

The firmware controlled Master mode enables the START and STOP condition interrupts but all other I²C functions are generated through firmware including:

- Generating the START and STOP conditions
- Generating the SCL clock
- Supplying the SDA bits in the proper time and phase relationship to the SCL signal.

In firmware controlled Master mode, the SCL and SDA lines are manipulated by clearing and setting the corresponding TRIS bits. The output level is always low irrespective of the value(s) in the PORT register. A '1' is output by setting the TRIS bit and a '0' is output by clearing the TRIS bit

The 7-bit and 10-bit Slave modes with START and STOP condition interrupts operate identically to the MSSP Slave modes except that START and STOP conditions generate SSPIF interrupts.

For more information about these SSP modes see Section 15 of the *PIC Mid-Range MCU Family Reference Manual (DS33023)*.

9.2.2 SLAVE MODE

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. Then, it loads the SSPBUF register with the received value currently in the SSPSR register.

Any combination of the following conditions will cause the MSSP module to generate a NACK pulse in lieu of the $\overline{\text{ACK}}$ pulse:

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

If the BF bit is set, the SSPSR register value is not loaded into the SSPBUF. However, both the SSPIF and SSPOV bits 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. The BF flag bit is cleared by reading the SSPBUF register. The SSPOV flag bit 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 I²C specification as well as the requirements of the MSSP module are shown in timing parameters #100 and #101 of the Electrical Specifications.

9.2.2.1 7-BIT ADDRESSING

Once the MSSP module has been enabled (SSPEN=1), the slave module waits for a START condition to occur. Following the START condition, eight bits are shifted into the SSPSR register. All incoming bits are sampled on the rising edge of the clock (SCL) line. The received address (register SSPSR<7:1>) is compared to the stored address (register SSPADD<7:1>). SSPSR<0> is the R/W bit and is not considered in the comparison. Comparison is made 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 transferred to the SSPBUF register on the falling edge of the eighth SCL pulse.
- b) The buffer full bit; BF is set on the falling edge of the eighth SCL pulse.
- c) An $\overline{\text{ACK}}$ pulse is generated during the ninth clock cycle.
- d) SSP interrupt flag bit; SSPIF (PIR1<3>) is set (interrupt is generated if enabled) - on the falling edge of the ninth SCL pulse.

9.2.2.2 10-BIT ADDRESSING

In 10-bit mode, the basic receive and transmit operations are the same as in the 7-bit mode. However, the criteria for address match are more complex.

Two address bytes need to be received by the slave. The five Most Significant bits (MSBs) of the first address byte specify that this is a 10-bit address. The LSb of the first received address byte is the R/W bit, which must be zero, specifying a write so the slave device will receive the second address byte. For a 10-bit address, the first byte equals '11110 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 through 9 applicable only to the 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 with R/W bit set to 1 (bits SSPIF and BF are set). This also puts the MSSP module in the Slave-transmit mode.
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.

PIC16C717/770/771

9.2.2.3 SLAVE RECEPTION

When the R/\bar{W} bit of the address byte is clear ($\text{SSPSR}\langle 0 \rangle = 0$) and an address match occurs, the R/W bit of the SSPSTAT register is cleared. The received address is loaded into the SSPBUF register on the falling edge of the eighth SCL pulse.

When the address byte overflow condition exists, then no Acknowledge (\bar{ACK}) pulse is given. An overflow condition is defined as either bit BF ($\text{SSPSTAT}\langle 0 \rangle$) or bit SSPOV ($\text{SSPCON}\langle 6 \rangle$) is set.

An MSSP interrupt is generated for each data transfer byte. Flag bit SSPIF ($\text{PIR1}\langle 3 \rangle$) 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 \bar{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 \bar{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.

FIGURE 9-8: I²C SLAVE MODE WAVEFORMS FOR RECEPTION (7-BIT ADDRESS)

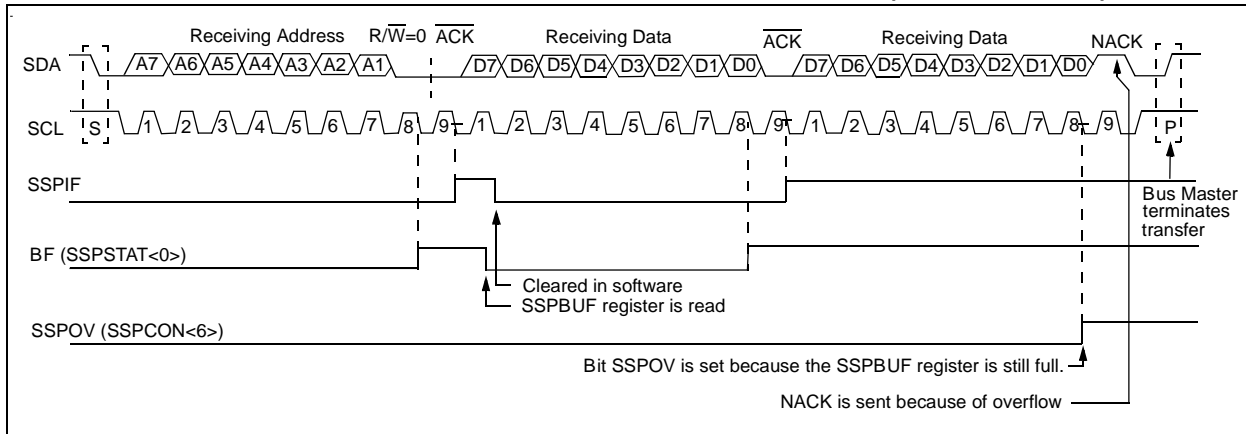
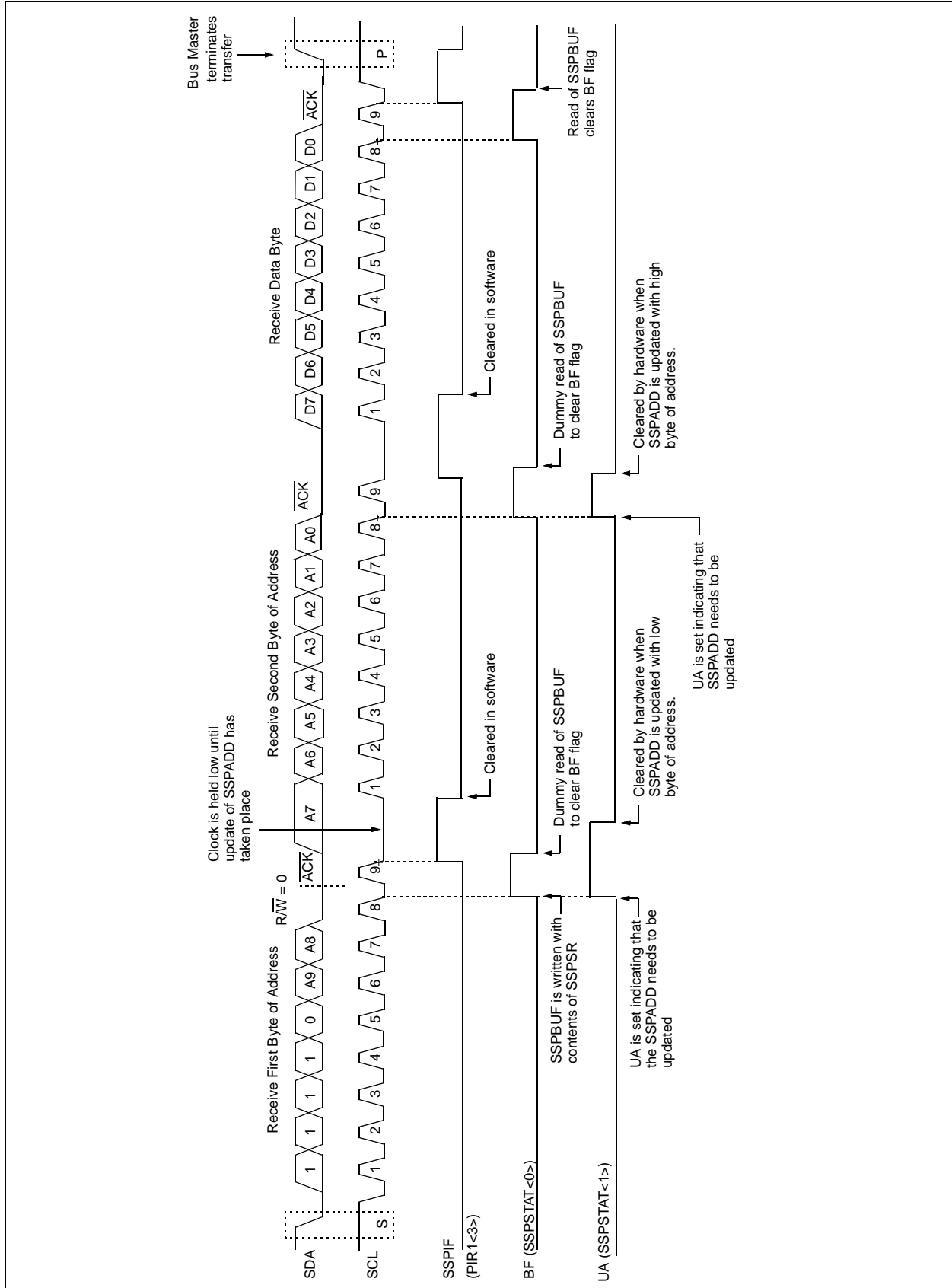


FIGURE 9-9: I²C SLAVE MODE FOR RECEPTION (10-BIT ADDRESS)



PIC16C717/770/771

9.2.2.4 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 on the falling edge of the eighth SCL pulse. The \overline{ACK} pulse will be sent on the ninth bit, and the SCL pin is held low. The slave module automatically stretches the clock by holding the SCL line low so that the master will be unable to assert another clock pulse until the slave is finished preparing the transmit data. The transmit data must be loaded into the SSPBUF register, which also loads the SSPSR register. The CKP bit (SSPCON<4>) must then be set to release the SCL pin from the forced low condition. The eight data bits are shifted out on the falling edges of the SCL input. This ensures that the SDA signal is valid during the SCL high time (Figure 9-10).

The \overline{ACK} or NACK signal from the master-receiver is latched on the rising edge of the ninth SCL input pulse. The master-receiver terminates slave transmission by

sending a NACK. If the SDA line is high (NACK), then the data transfer is complete. When the NACK is latched by the slave, the slave logic is RESET which also resets the R/\overline{W} bit to '0'. The slave module then monitors for another occurrence of the START bit. The slave firmware knows not to load another byte into the SSPBUF register by sensing that the buffer is empty ($BF = 0$) and the R/\overline{W} bit has gone low. If the SDA line is low (\overline{ACK}), the R/\overline{W} bit remains high indicating that the next transmit data must be loaded into the SSPBUF register.

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

For more information about the I²C Slave mode, refer to Application Note AN734, "Using the PIC[®] SSP for Slave I²C™ Communication".

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

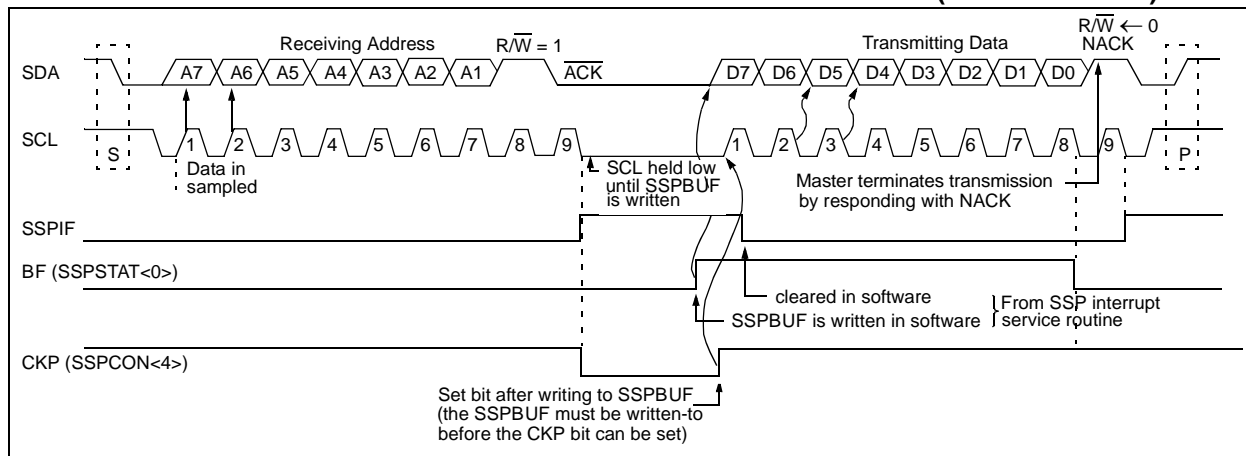
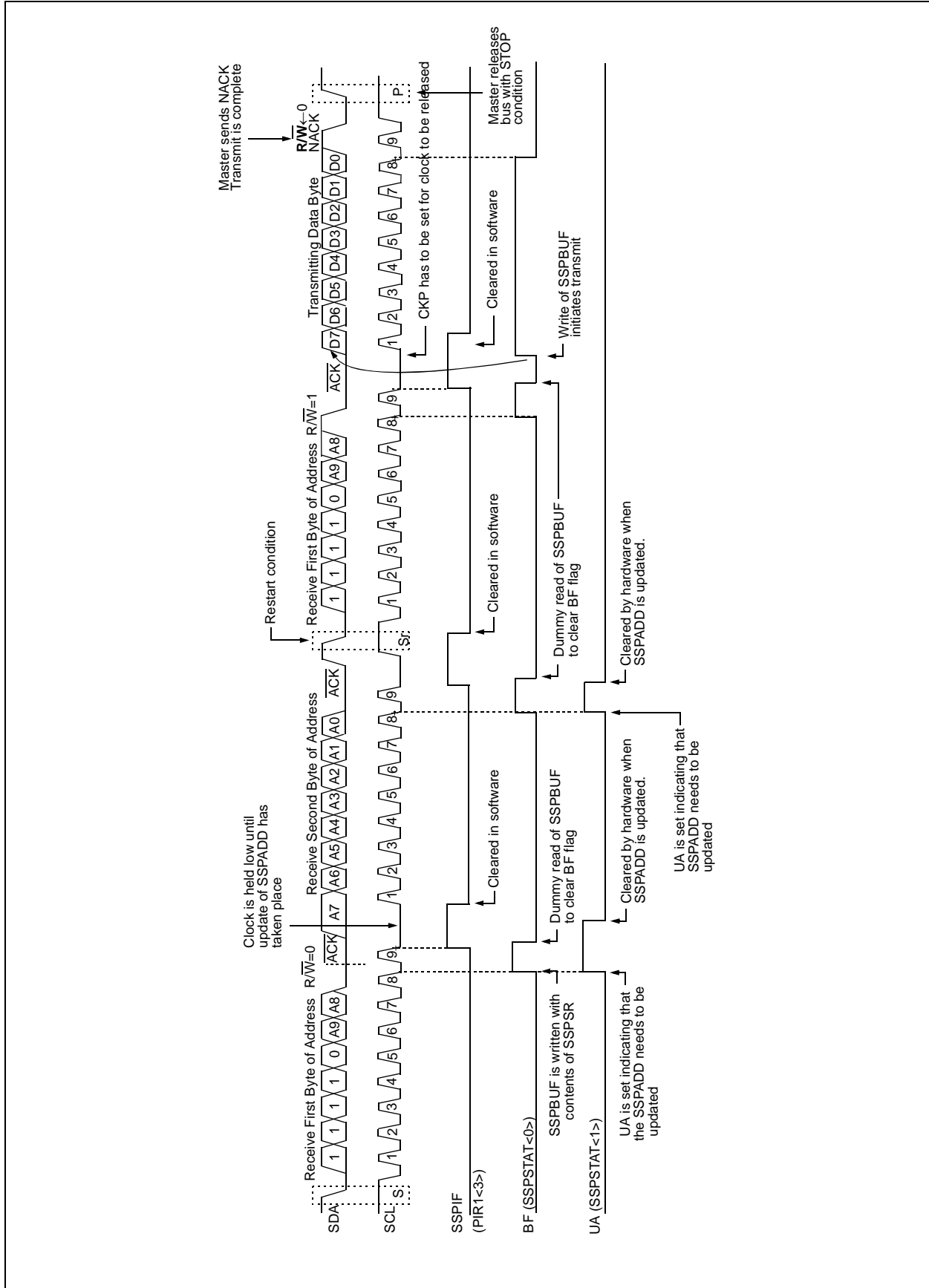


FIGURE 9-11: I²C SLAVE MODE WAVEFORMS FOR TRANSMISSION (10-BIT ADDRESS)



PIC16C717/770/771

9.2.3 GENERAL CALL ADDRESS SUPPORT

The addressing procedure for the I²C 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 I²C protocol. It consists of all 0's with R/W = 0

The general call address is recognized when the General Call Enable bit (GCEN) is set (SSPCON2<7> is set). Following a START bit detect, eight bits are shifted

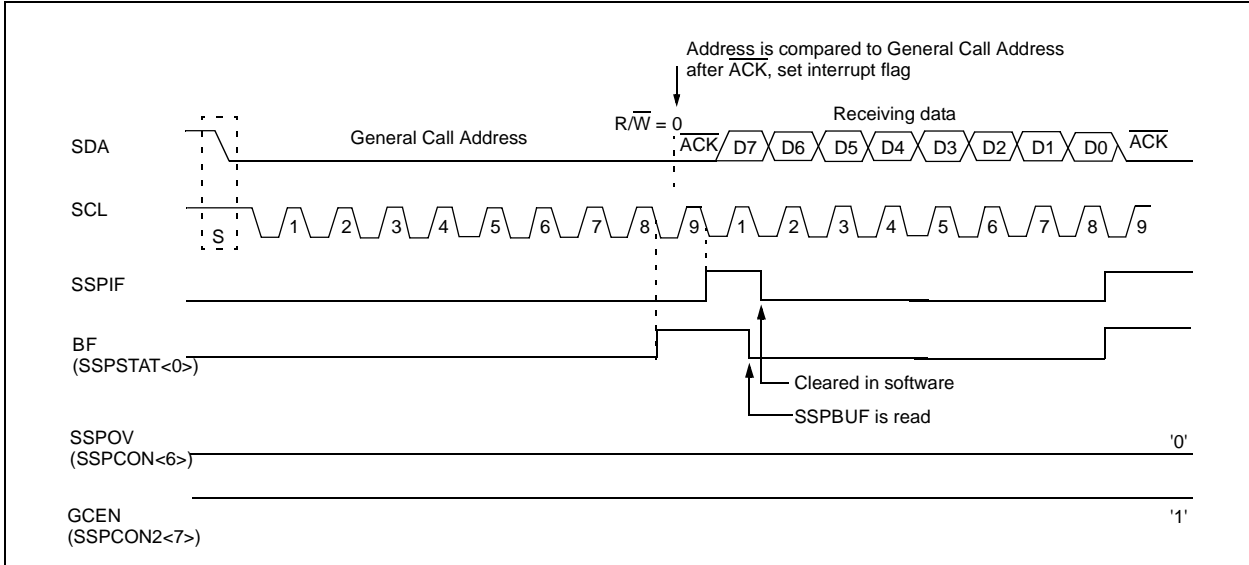
into the SSPSR, and the address is compared against SSPADD. It is also compared to the general call address, fixed in hardware.

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 ($\overline{\text{ACK}}$ 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.

If the general call address is sampled with GCEN set and the slave configured in 10-bit Address mode, 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 9-12).

FIGURE 9-12: SLAVE MODE GENERAL CALL ADDRESS SEQUENCE (7- OR 10-BIT MODE)



9.2.4 SLEEP OPERATION

While in SLEEP mode, the I²C slave module can receive addresses or data. When an address match or complete byte transfer occurs, it wakes the processor from SLEEP (if the SSP interrupt bit is enabled).

9.2.5 EFFECTS OF A RESET

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

9.2.6 MASTER MODE

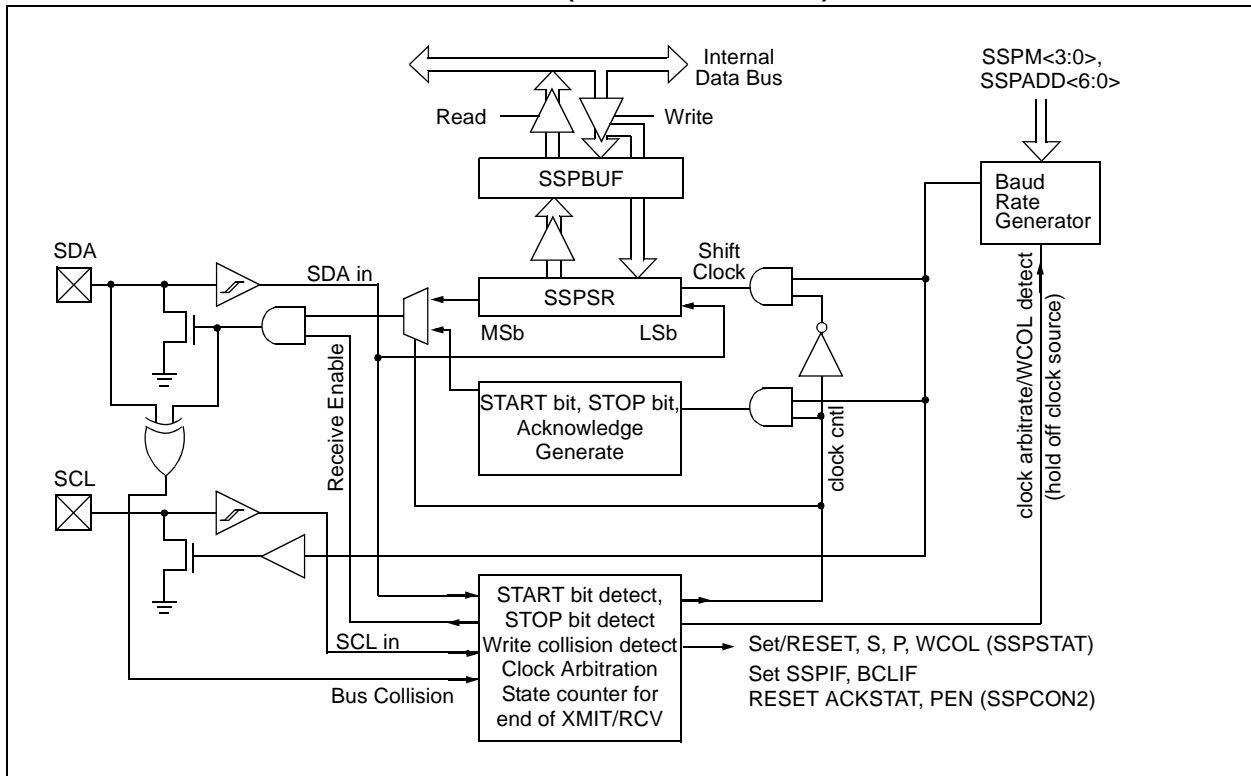
Master mode operation supports interrupt generation on the detection of the START and STOP conditions. The STOP (P) and START (S) bits are cleared from a RESET or when the MSSP module is disabled. Control of the I²C bus may be taken when the P bit is set or the bus is idle with both the S and P bits 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 9-13: MSSP BLOCK DIAGRAM (I²C MASTER MODE)



PIC16C717/770/771

9.2.7 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 I²C 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

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

9.2.8 I²C MASTER OPERATION

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.

1. Assert a START condition on SDA and SCL.
2. Assert a Repeated START condition on SDA and SCL.
3. Write to the SSPBUF register initiating transmission of data/address.
4. Generate a STOP condition on SDA and SCL.
5. Configure the I²C port to receive data.
6. Generate an Acknowledge condition at the end of a received byte of data.

The master device generates all 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 I²C bus will not be released.

Note: The MSSP Module, when configured in I²C Master mode, does not allow queuing 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.

9.2.9 BAUD RATE GENERATOR

The baud rate generator used for SPI mode operation is used in the I²C Master mode to set the SCL clock frequency. Standard SCL clock frequencies are 100 kHz, 400 kHz, and 1 MHz. One of these frequencies can be achieved by setting the SSPADD register to the appropriate number for the selected Fosc frequency. One half of the SCL period is equal to $[(SSPADD+1) \bullet 2]/Fosc$.

The baud rate generator reload value is contained in the lower seven bits of the SSPADD register (Figure 9-14). When the BRG is loaded with this value, the BRG counts down to 0 and stops until another reload occurs. The BRG count is decremented twice per instruction cycle (Tcy) on the Q2 and Q4 clock.

In I²C Master mode, the BRG is reloaded automatically provided that the SCL line is sampled high. For example, if Clock Arbitration is taking place, the BRG reload will be suppressed until the SCL line is released by the slave allowing the pin to float high (Figure 9-15).

FIGURE 9-14: BAUD RATE GENERATOR BLOCK DIAGRAM

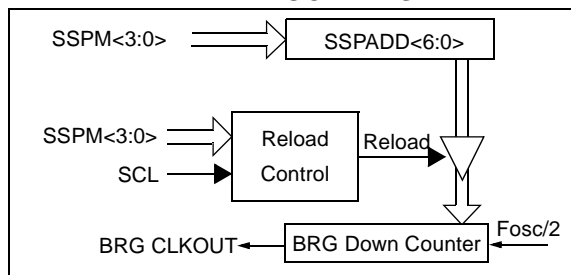
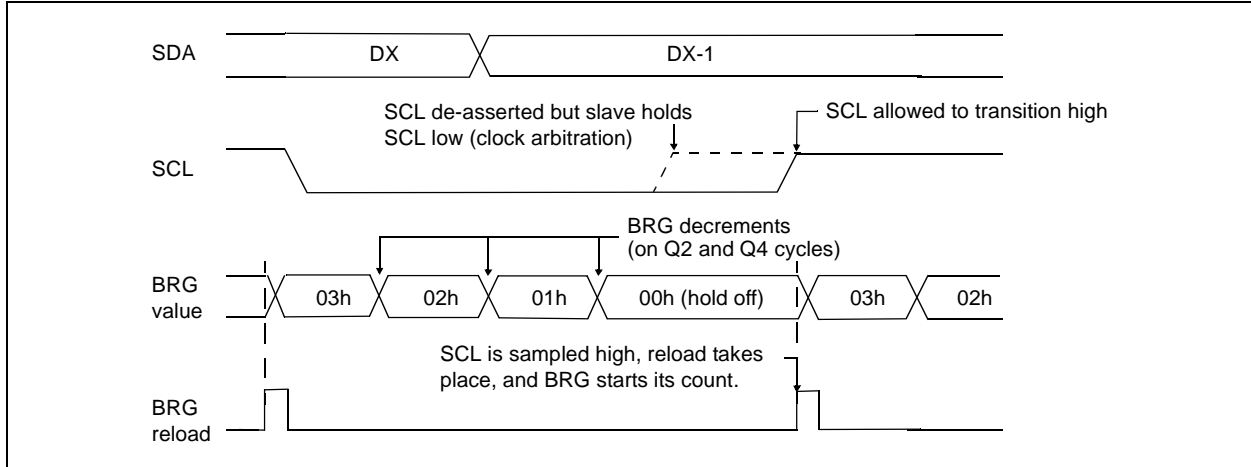


FIGURE 9-15: BAUD RATE GENERATOR TIMING WITH CLOCK ARBITRATION



9.2.10 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, indicating that the bus is available, the baud rate generator is 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 (TBRG) indicating the bus is still available, the SDA pin is driven low. The SDA transition from high to low while SCL is high is the START condition. This causes the S bit (SSPSTAT<3>) to be set. When the S bit is set, the baud rate generator is reloaded with the contents of SSPADD<6:0> and resumes its count. When the baud rate generator times out (TBRG) the START condition is complete, concurrent with the following events:

- The SEN bit (SSPCON2<0>) is automatically cleared by hardware,
- The baud rate generator is suspended leaving the SDA line held low.
- The SSPIF flag is set.

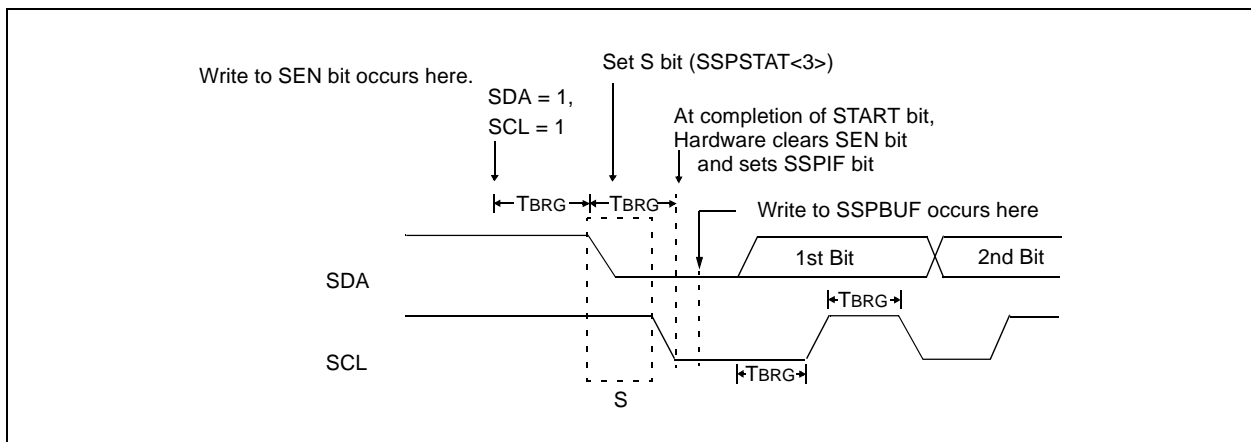
Note: If at the beginning of START condition, the SDA and SCL pins are already sampled low, or if during the START condition, the SCL line is sampled low before the SDA line is driven low, a bus collision occurs. Thus, the Bus Collision Interrupt Flag (BCLIF) is set, the START condition is aborted, and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

9.2.10.1 WCOL STATUS FLAG

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

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

FIGURE 9-16: FIRST START BIT TIMING



PIC16C717/770/771

9.2.11 I²C MASTER MODE REPEATED START CONDITION TIMING

A Repeated START condition occurs when the RSEN bit (SSPCON2<1>) is set high while the I²C 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 (TBRG). 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 reloaded with the contents of SSPADD<6:0> and begins counting. SDA and SCL must be sampled high for one TBRG period. This action is then followed by assertion of the SDA pin (SDA is low) for one TBRG period while SCL is high. As soon as a START condition is detected on the SDA and SCL pins, the S bit (SSPSTAT<3>) will be set. Following this, the baud rate generator is reloaded with the contents of SSPAD<6:0> and begins counting. When the BRG times out a third time, the RSEN bit in the SSPCON2 register is automatically cleared and SCL is pulled low. The SSPIF flag is set, which indicates the Restart sequence is complete.

Immediately following the SSPIF bit transition to true, 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 perform one of the following:

- Transmit an additional eight bits of address (if the user transmitted the first half of a 10-bit address with R/W = 0),
- Transmit eight bits of data (if the user transmitted a 7-bit address with R/W = 0), or
- Receive eight bits of data (if the user transmitted either the first half of a 10-bit address or a 7-bit address with R/W = 1).

9.2.11.1 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).

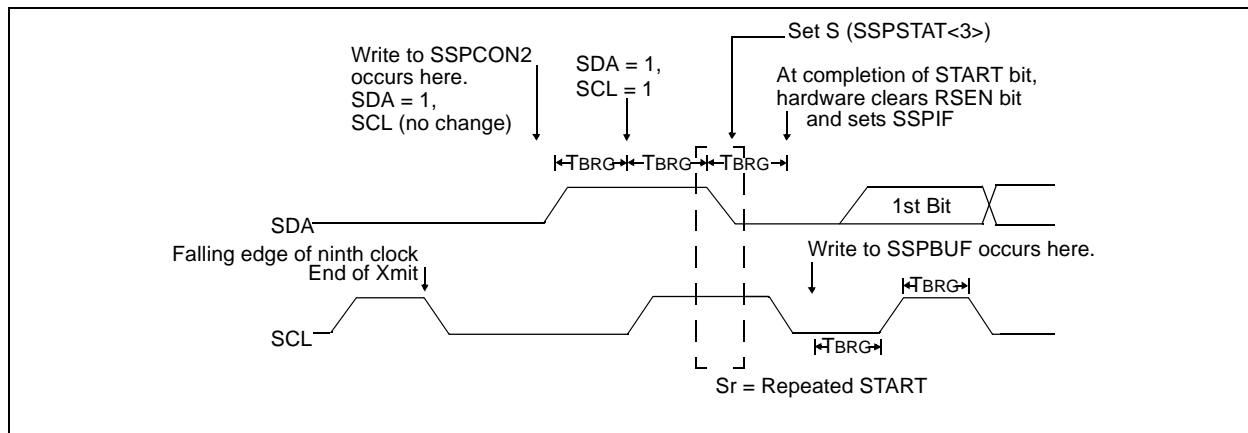
Note: Because queuing of events is not allowed, writing of the lower five bits of SSPCON2 is disabled until the Repeated START condition is complete.

Note 1: If RSEN is set while another event is in progress, it will not take effect. Queuing of events is not allowed.

2: A bus collision during the Repeated START condition occurs if either of the following is true:

- 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".

FIGURE 9-17: REPEAT START CONDITION WAVEFORM



9.2.12 I²C MASTER MODE TRANSMISSION

In Master-transmitter mode, serial data is output through SDA, while SCL outputs the serial clock. The first byte transmitted contains seven bits of address data and the Read/Write (R/W) bit. In this case, the R/W bit will be logic '0'. Subsequent serial data is transmitted eight 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.

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 (TBRG). 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 TBRG, 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. This allows the slave device being addressed to respond with an $\overline{\text{ACK}}$ bit during the ninth bit time. The status of $\overline{\text{ACK}}$ is read into the ACKDT on the rising edge of the ninth clock. If the master receives an Acknowledge, the Acknowledge status bit (ACKSTAT) is cleared. Otherwise, the bit is set. The SSPIF is set on the falling edge of the ninth clock, 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 9-18).

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 at the completion of the START sequence.
- c) The user resets the SSPIF bit and loads the SSPBUF with seven bits of address plus R/W bit to transmit.
- d) Address and R/W is shifted out the SDA pin until all eight bits are transmitted.
- e) The MSSP Module shifts in the $\overline{\text{ACK}}$ 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 resets the SSPIF bit and loads the SSPBUF with eight bits of data.
- h) DATA is shifted out the SDA pin until all eight bits are transmitted.
- i) The MSSP Module shifts in the $\overline{\text{ACK}}$ 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 resets the SSPIF bit and generates a STOP condition by setting the STOP enable bit PEN in SSPCON2.
- l) SSPIF is set when the STOP condition is complete.

9.2.12.1 BF STATUS FLAG

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

9.2.12.2 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).

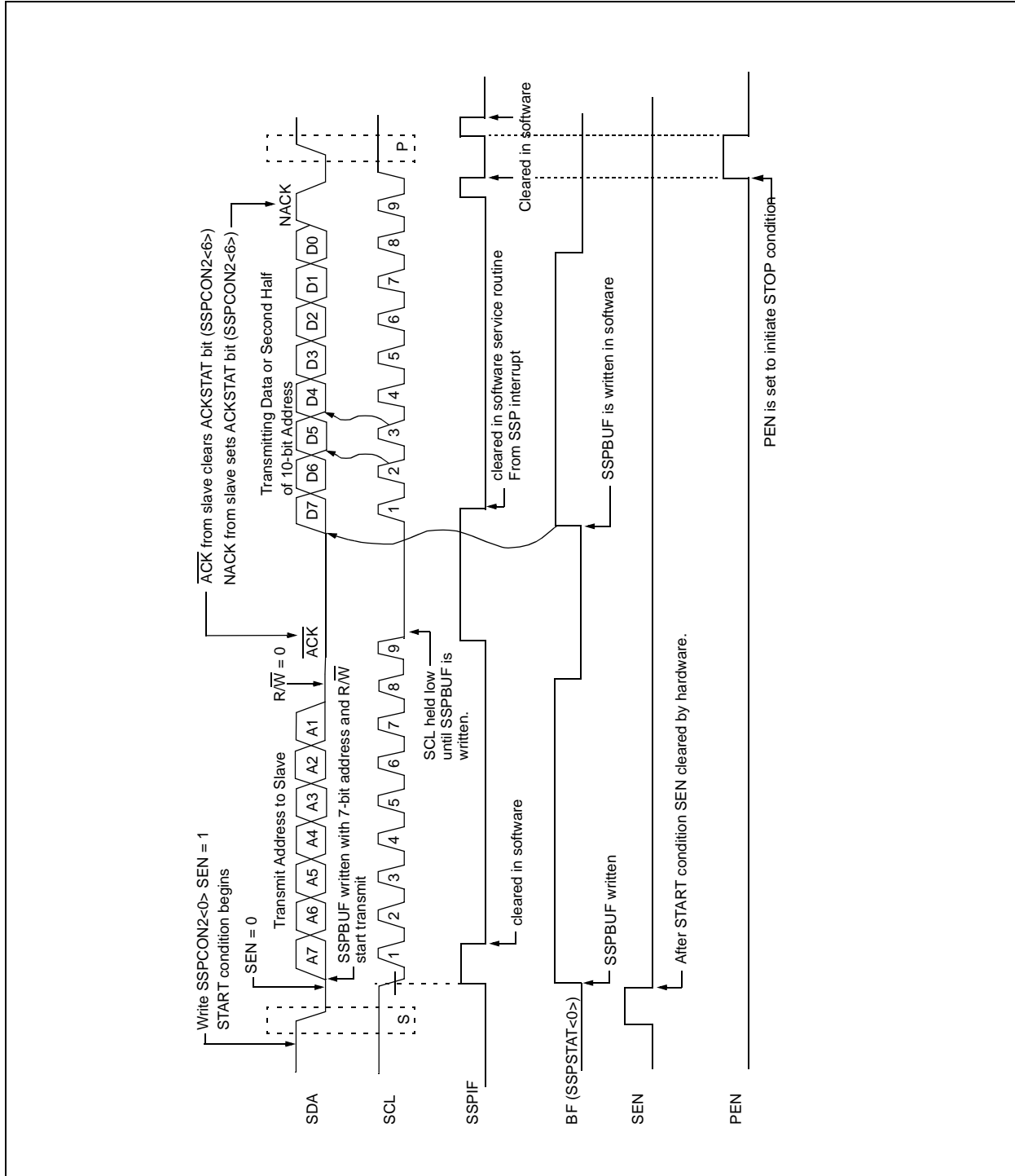
WCOL must be cleared in software.

9.2.12.3 ACKSTAT STATUS FLAG

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

PIC16C717/770/771

FIGURE 9-18: I²C MASTER MODE WAVEFORMS FOR TRANSMISSION (7 OR 10-BIT ADDRESS)



9.2.13 I²C MASTER MODE RECEPTION

In Master-receive mode, the first byte transmitted contains seven bits of address data and the R/W bit. In this case, the R/W bit will be logic '1'. Thus, the first byte transmitted is a 7-bit slave address followed by a '1' to indicate receive. Serial data is received via SDA, while SCL outputs the serial clock. Serial data is received eight bits at a time. After each byte is received, an Acknowledge bit is transmitted. The START condition indicates the beginning of a transmission. The master-receiver terminates slave transmission by responding to the last byte with a NACK Acknowledge and follows this with a STOP condition to indicate to other masters that the bus is free.

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

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 following events occur:

- The receive enable bit is automatically cleared.
- The contents of the SSPSR are loaded into the SSPBUF.
- The BF flag is set.
- The SSPIF is set.
- 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 clearing the ACKDT bit (SSPCON2<5>) and setting the Acknowledge sequence enable bit, ACKEN (SSPCON2<4>).

A typical receive 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 at the completion of the START sequence.
- c) The user resets the SSPIF bit and loads the SSPBUF with seven bits of address in the MSBs and the LSb (R/W bit) set to '1' for receive.
- d) Address and R/W is shifted out the SDA pin until all eight 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 resets the SSPIF bit and sets the RCEN bit to enable reception.
- h) DATA is shifted into the SDA pin until all eight bits are received.
- i) The MSSP module sets the SSPIF bit and clears the RCEN bit at the falling edge of the eighth clock.
- j) The user resets the SSPIF bit and sets the ACKDT bit to '0' (ACK), if another byte is anticipated. Otherwise, the ACKDT bit is set to '1' (NACK) to terminate reception. The user sets ADKEN to start the Acknowledge sequence.
- k) The MSSP module sets the SSPIF bit at the completion of the Acknowledge.
- l) If a NACK was sent in step (j), then the user proceeds with step (m). Otherwise, reception continues by repeating steps (g) through (j).
- m) The user generates a STOP condition by setting the STOP enable bit PEN in SSPCON2.
- n) SSPIF is set when the STOP condition is complete.

9.2.13.1 BF STATUS FLAG

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

9.2.13.2 SSPOV STATUS FLAG

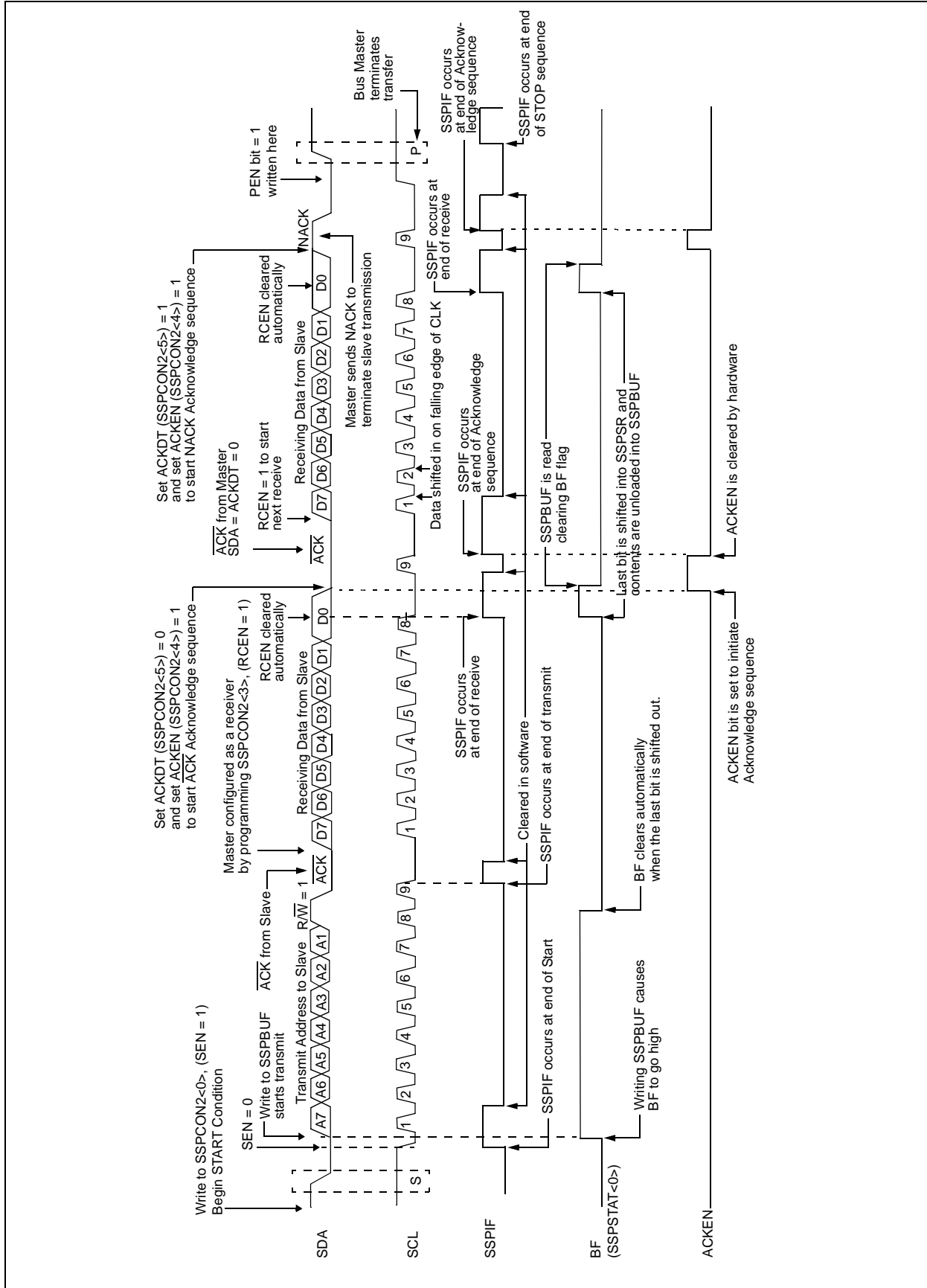
In receive operation, SSPOV is set when eight bits are received into the SSPSR and the BF flag is already set from a previous reception.

9.2.13.3 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).

PIC16C717/770/771

FIGURE 9-19: I²C MASTER WAVEFORMS FOR RECEPTION (7-BIT ADDRESS)



9.2.14 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 ACKDT (SSPCON2<5>) is presented on the SDA pin. If the user wishes to generate an Acknowledge (ACK), then the ACKDT bit should be cleared. Otherwise, the user should set the ACKDT bit (NACK) before starting an Acknowledge sequence. The baud rate generator is then loaded from SSPADD<6:0> and counts for one rollover period (TBRG). The SCL pin is then de-asserted (pulled high). When the SCL pin is sampled high (clock

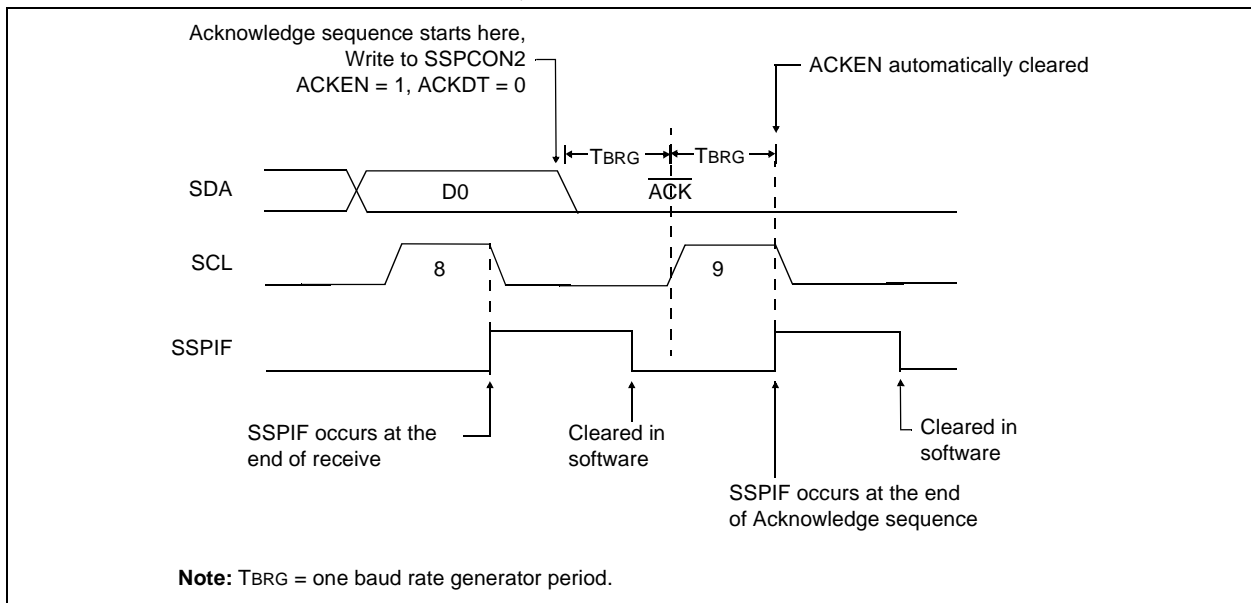
arbitration), the baud rate generator is reloaded and counts for another TBRG. At the completion of the TBRG period, the following events occur (see Figure 9-20):

- The SCL pin is pulled low.
- The ACKEN bit is automatically cleared.
- The baud rate generator is turned off.
- The MSSP module goes into IDLE mode.

9.2.14.1 WCOL STATUS FLAG

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

FIGURE 9-20: ACKNOWLEDGE SEQUENCE WAVEFORM



PIC16C717/770/771

9.2.15 STOP CONDITION TIMING

The master asserts a STOP condition on the SDA and SCL pins at the end of a receive/transmit by setting the Stop Sequence Enable bit PEN (SSPCON2<2>). At the end of a receive/transmit plus Acknowledge, the SCL line is held low immediately following the falling edge of the ninth SCL pulse. 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 loaded from SSPADD<6:0> and counts down to 0. When the baud rate generator times out, the SCL pin is brought high, the BRG is reloaded and one TBRG (baud rate generator rollover count) later, the SDA pin is de-asserted. The SDA pin transition from low to high while SCL is high is the STOP condition and causes the P bit (SSPSTAT<4>) 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

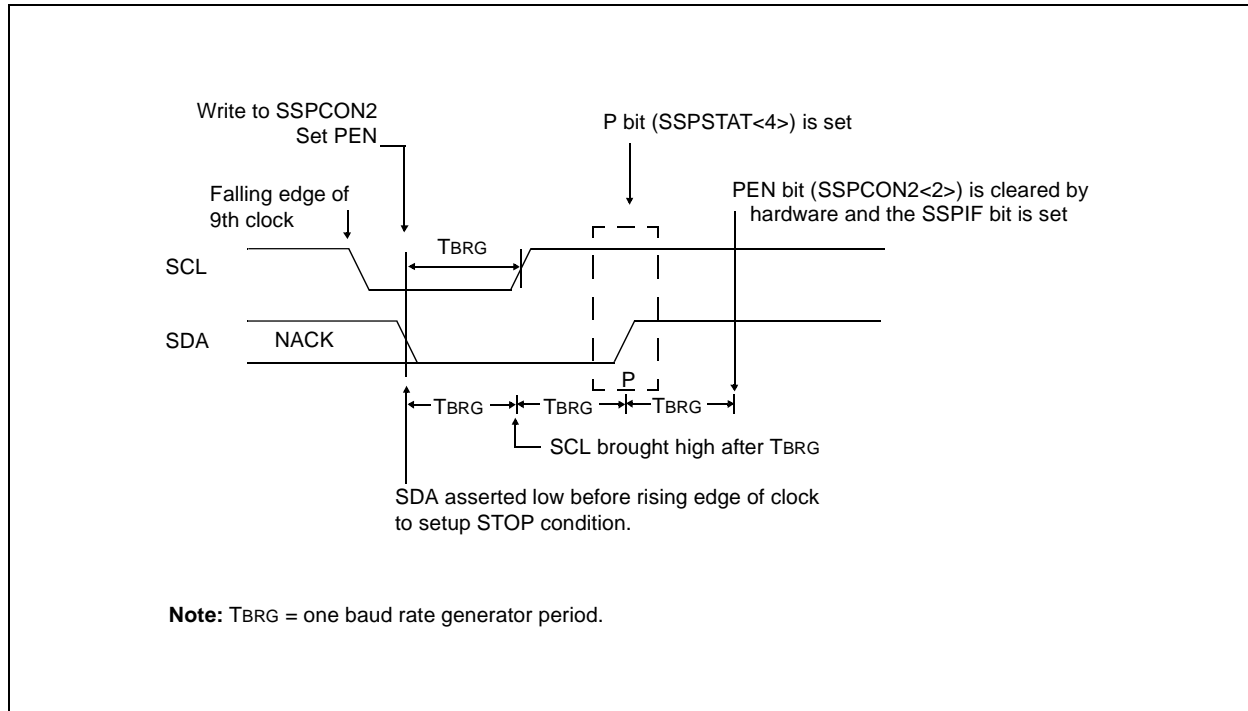
times out (TBRG) the STOP condition is complete and the PEN bit is cleared and the SSPIF bit is set (Figure 9-21).

Whenever the firmware decides to take control of the bus, it should first determine if the bus is busy by checking the S and P bits in the SSPSTAT register. When the MSSP module detects a START or STOP condition the SSPIF flag is set. If the bus is busy (S bit is set), then the CPU can be configured to be interrupted when the bus is free by enabling the SSPIF interrupt to detect the STOP bit.

9.2.15.1 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 9-21: STOP CONDITION RECEIVE OR TRANSMIT MODE

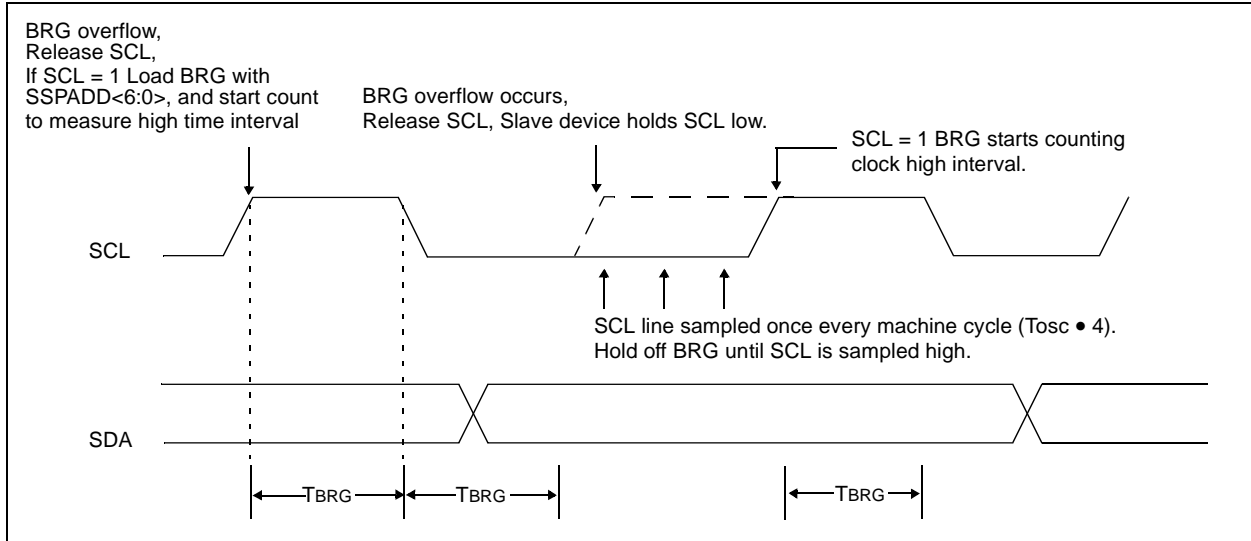


9.2.16 CLOCK ARBITRATION

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 9-22).

FIGURE 9-22: CLOCK ARBITRATION TIMING IN MASTER TRANSMIT MODE



PIC16C717/770/771

9.2.17 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, bus arbitration is initiated when one master outputs a '1' on SDA (by letting SDA float high) and another master asserts a '0'. 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 that expected a '1' will set the Bus Collision Interrupt Flag, BCLIF, and reset the I²C port to its IDLE state. (Figure 9-23).

A bus collision during transmit results in the following events:

- The transmission is halted.
- The BF flag is cleared
- The SDA and SCL lines are de-asserted
- The restriction on writing to the SSPBUF during transmission is lifted.

When the user services the bus collision interrupt service routine, and if the I²C bus is free, the user can resume communication by asserting a START condition.

A bus collision during a START, Repeated START, STOP or Acknowledge condition results in the following events:

- The condition is aborted.
- The SDA and SCL lines are de-asserted.
- The respective control bits in the SSPCON2 register are cleared.

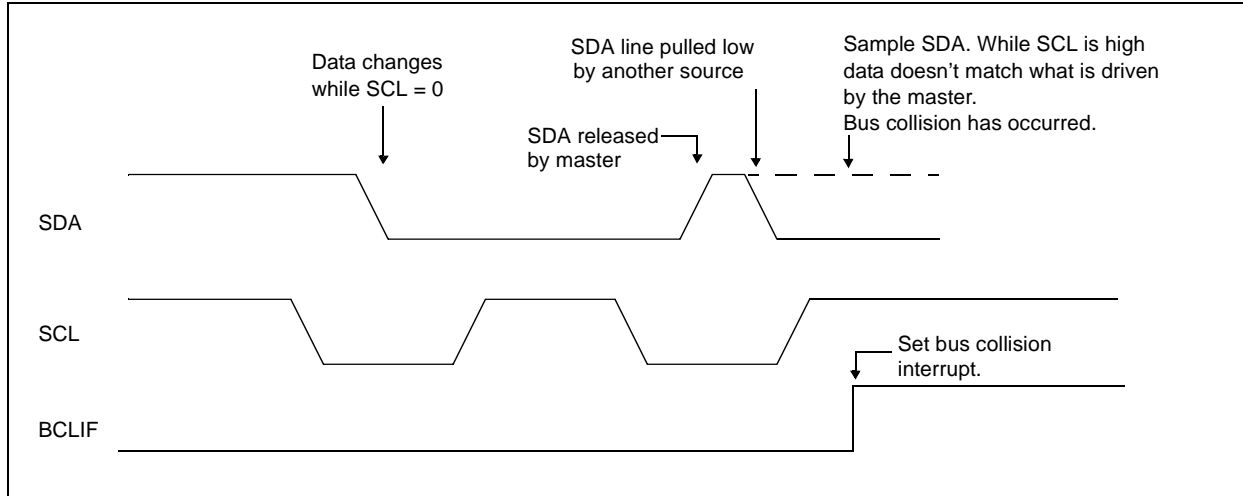
When the user services the bus collision interrupt service routine, and if the I²C bus is free, the user can resume communication by asserting a START condition.

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 I²C 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 9-23: BUS COLLISION TIMING FOR TRANSMIT AND ACKNOWLEDGE



9.2.17.1 BUS COLLISION DURING A START CONDITION

During a START condition, a bus collision occurs if:

- SDA or SCL are sampled low at the beginning of the START condition (Figure 9-24).
- SCL is sampled low before SDA is asserted low. (Figure 9-25).

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 9-24).

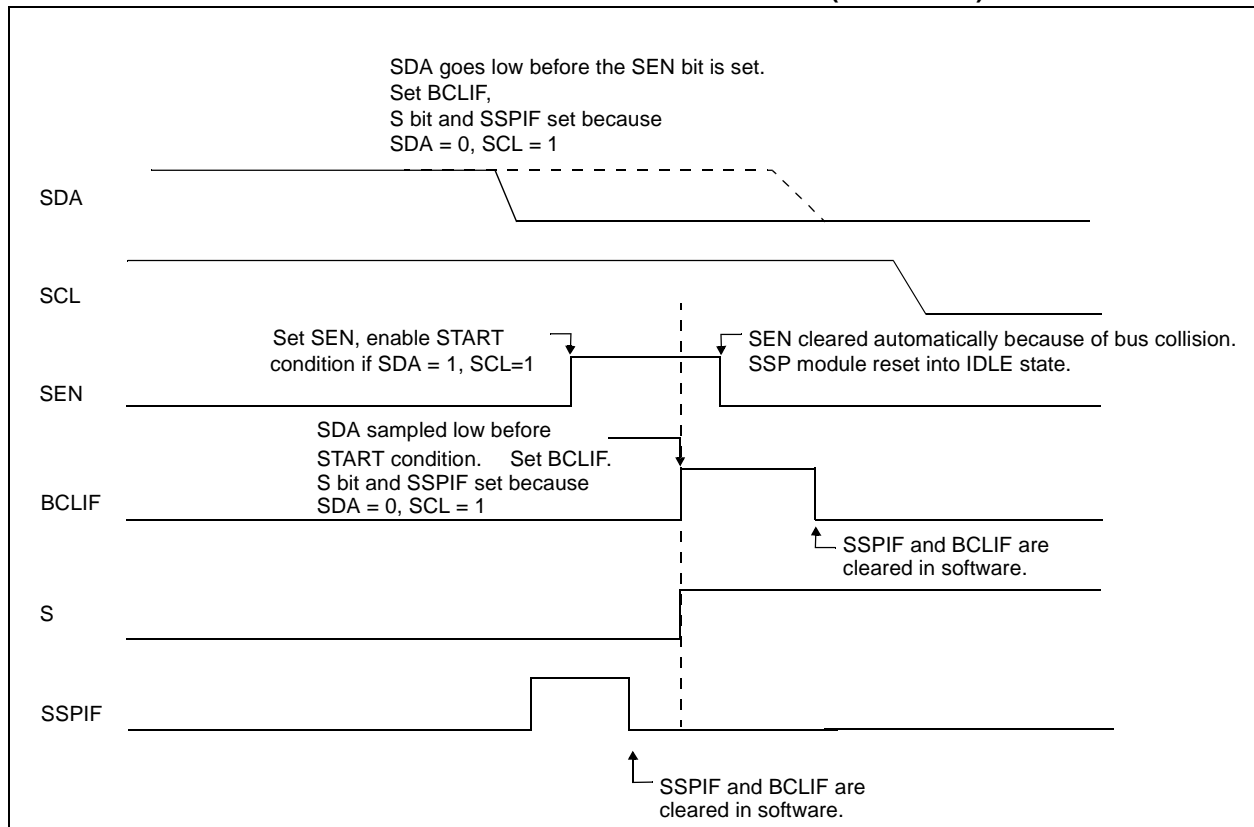
The START condition begins with the SDA and SCL pins de-asserted. When the SDA pin is sampled high, the baud rate generator is loaded from SSPADD<6:0> and counts down to 0. If the SCL pin is sampled low

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 9-26). 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 pin 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 9-24: BUS COLLISION DURING START CONDITION (SDA ONLY)



PIC16C717/770/771

FIGURE 9-25: BUS COLLISION DURING START CONDITION (SCL = 0)

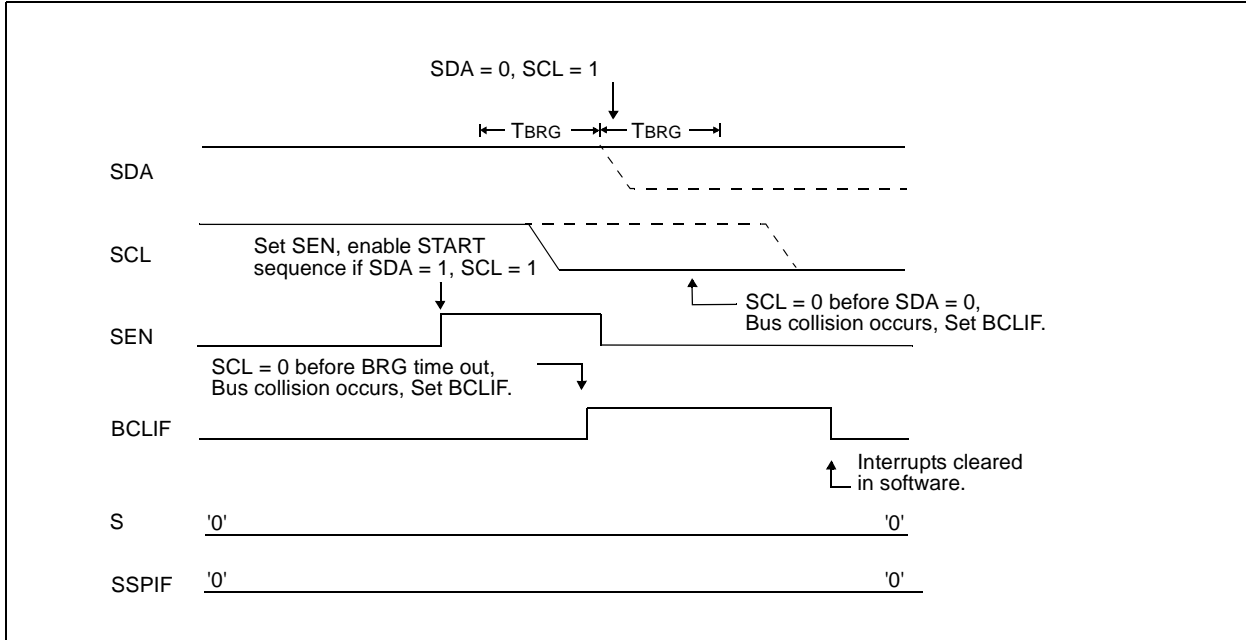
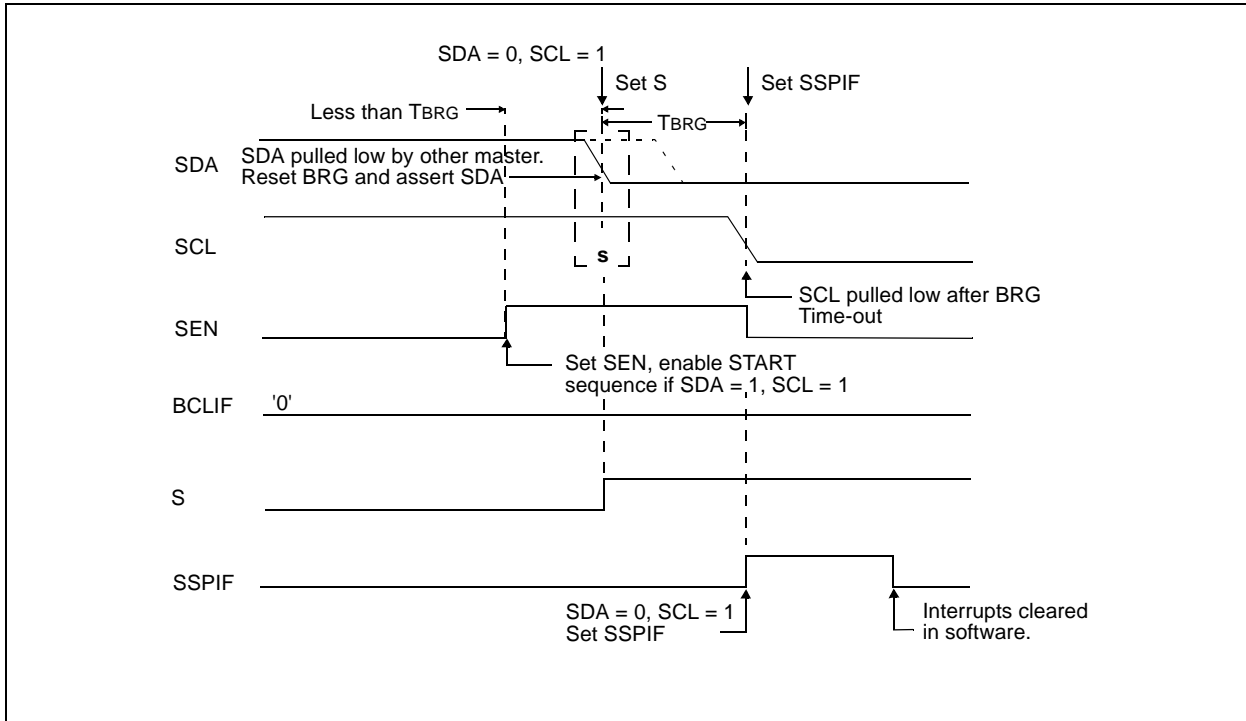


FIGURE 9-26: BRG RESET DUE TO SDA COLLISION DURING START CONDITION



9.2.17.2 BUS COLLISION DURING A REPEATED START CONDITION

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

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

When the master module 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 9-27).

FIGURE 9-27: BUS COLLISION DURING A REPEATED START CONDITION (CASE 1)

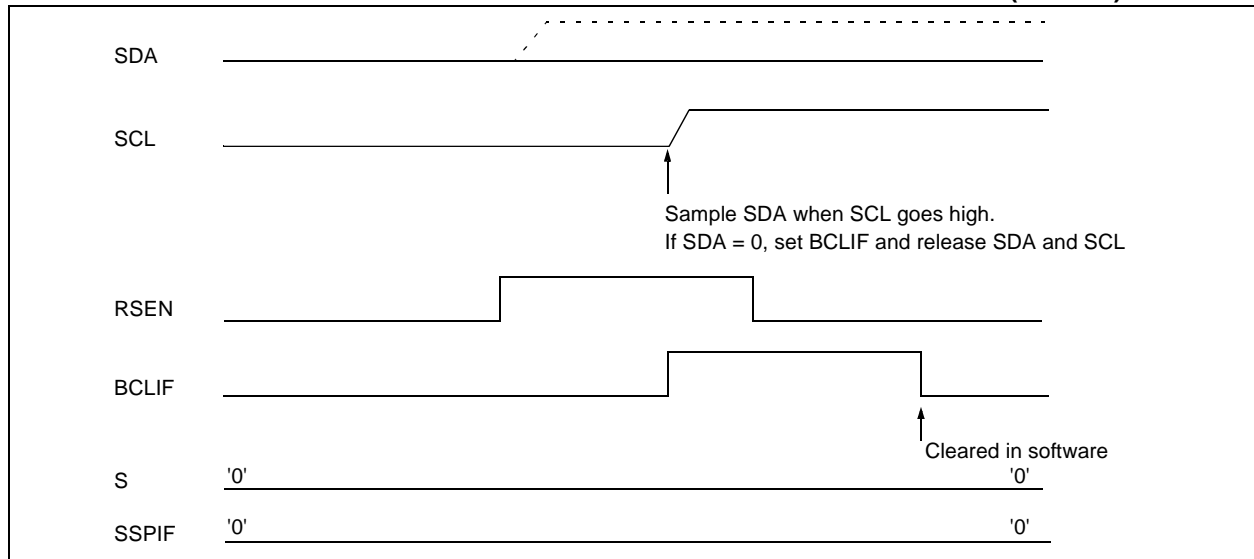
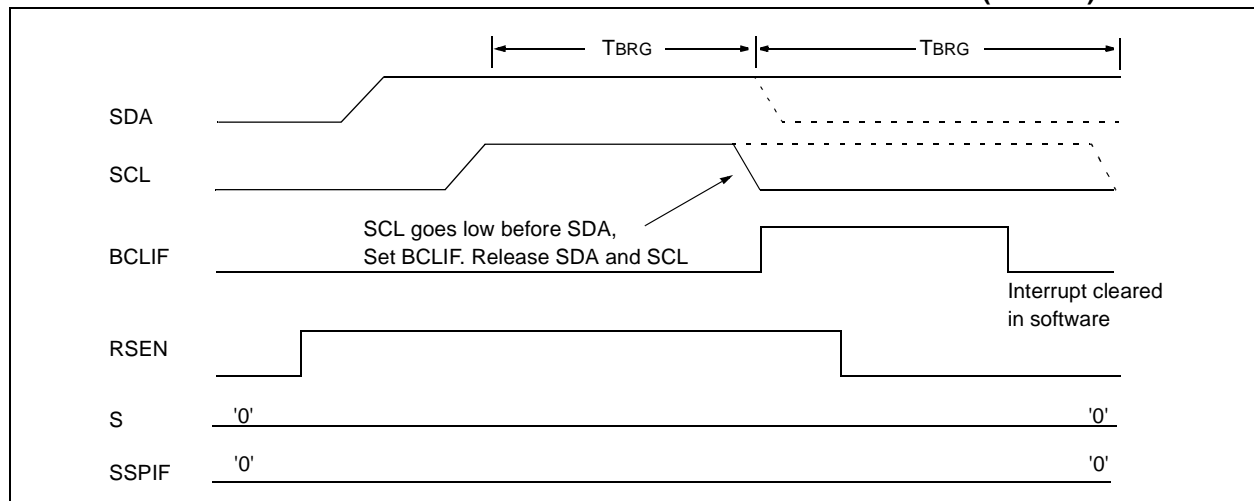


FIGURE 9-28: BUS COLLISION DURING REPEATED START CONDITION (CASE 2)



PIC16C717/770/771

9.2.17.3 BUS COLLISION DURING A STOP CONDITION

Bus collision occurs during a STOP condition if:

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

The STOP condition begins with SDA asserted low. When SDA is sampled low, the SCL pin is allowed 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' (Figure 9-29). 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 9-30).

FIGURE 9-29: BUS COLLISION DURING A STOP CONDITION (CASE 1)

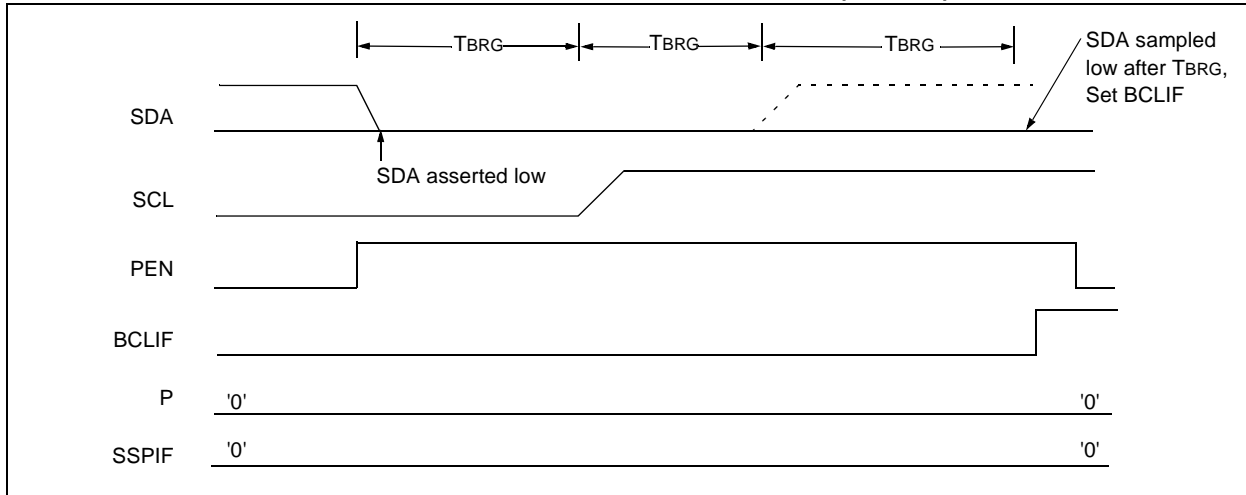
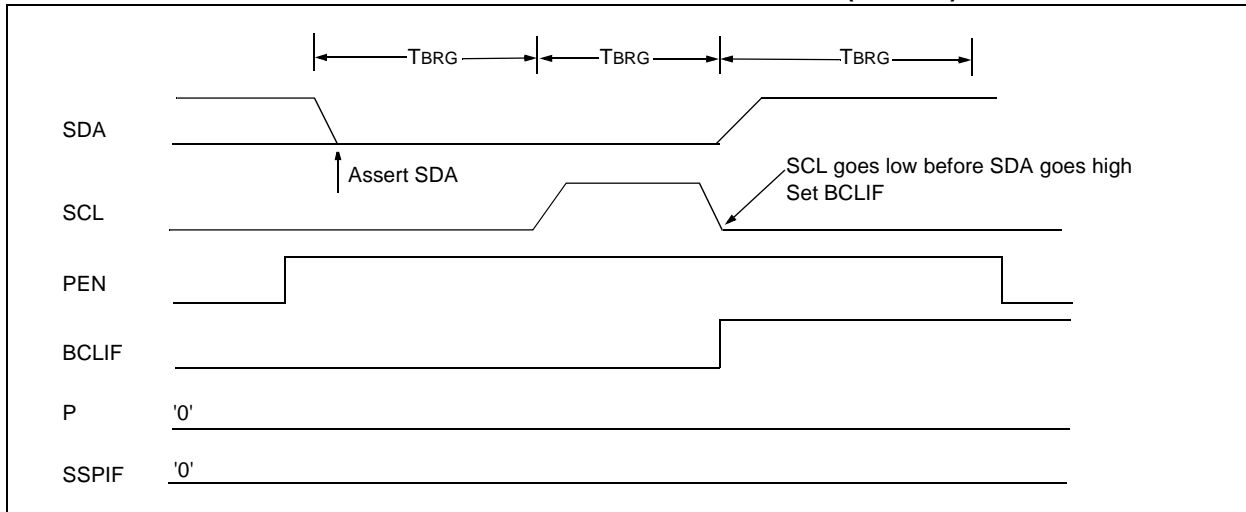


FIGURE 9-30: BUS COLLISION DURING A STOP CONDITION (CASE 2)



9.2.18 CONNECTION CONSIDERATIONS FOR I²C BUS

For Standard mode I²C bus devices, the values of resistors R_p and R_s in Figure 9-31 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 $V_{OL\ max} = 0.4V$ for the specified output stages. For

example, with a supply voltage of $V_{DD} = 5V \pm 10\%$ and $V_{OL\ max} = 0.4V$ at 3 mA, $R_{p\ min} = (5.5 - 0.4) / 0.003 = 1.7\ k\Omega$. V_{DD} as a function of R_p is shown in Figure 9-31. The desired noise margin of $0.1V_{DD}$ 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 9-31).

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 9-31: SAMPLE DEVICE CONFIGURATION FOR I²C BUS

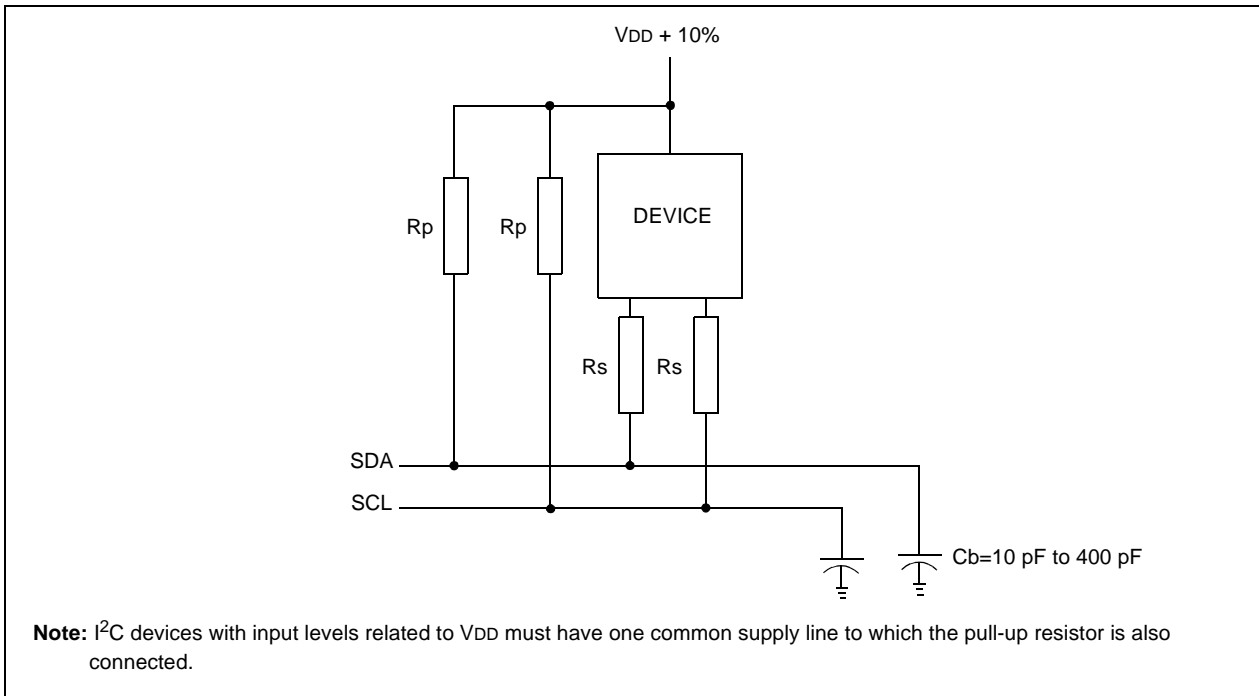


TABLE 9-3: REGISTERS ASSOCIATED WITH I²C OPERATION

| 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 |
| 93h | SSPADD | Synchronous Serial Port (I ² C Mode) Address Register | | | | | | | | 0000 0000 | 0000 0000 |

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

PIC16C717/770/771

NOTES:

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 | |
| bit 7 | | | | | | | | bit 0 |

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: These are the minimum trip points for the LVD. See Table 15-8 for the trip point tolerances. Selection of reserved setting may result in an inadvertent interrupt.

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

- n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

PIC16C717/770/771

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 | — | — | — | — |
| bit 7 | | | | bit 0 | | | |

- 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⁾
1 = Enabled, VRH analog reference is output on RA3 if enabled (VRHEN = 1)
0 = Disabled, analog reference is used internally only⁽¹⁾
- 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'

Note 1: RA2 and RA3 must be configured as analog inputs when the VREF output functions are enabled (See ANSEL on page 25).

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

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.

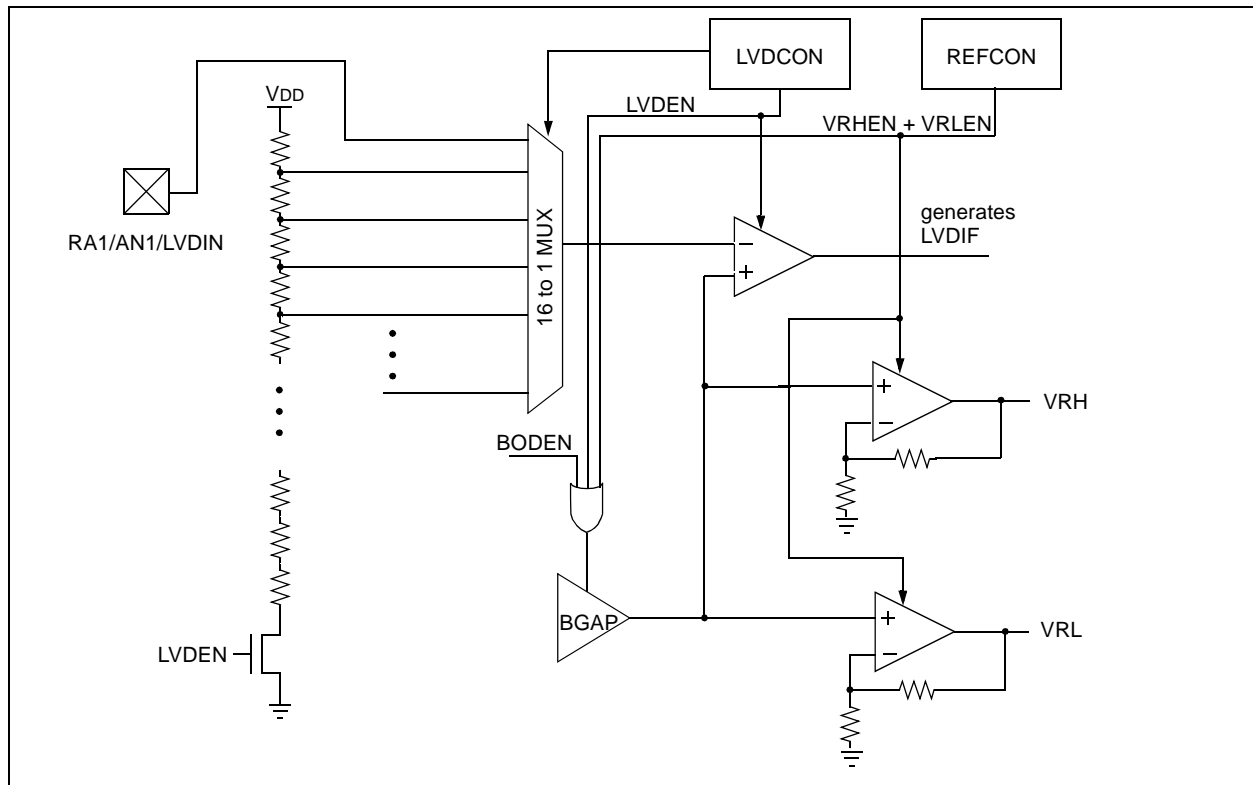
The VRL reference is enabled by setting control bit VRLEN (REFCON<6>). When this bit is set, the gain amplifier is enabled. After a specified start-up time a stable reference of 2.048V nominal is generated and can be used by the A/D converter as a reference input.

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 LVD-CON register. The voltage references will not be reliable until the bandgap is stable as shown by BGST being set.

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



PIC16C717/770/771

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.

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>).

| |
|---|
| <p>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.</p> |
|---|

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).

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 (AVDD/AVSS), the voltage level on the VREF+ and VREF- pins, or internal voltage references if enabled (VRH, VRL).

The A/D converter can be triggered by setting the GO/DONE bit, or by the special event Compare mode of the ECCP module. When conversion is complete, the GO/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 ANSEL register, shown in Register 3-1, selects between the Analog or Digital Port Pin modes. 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/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.

PIC16C717/770/771

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/DONE | CHS3 | ADON |
| bit 7 | | | | | | bit 0 | |

- 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)
 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/8
- bit 5-3,1 **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/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

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

REGISTER 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 | Reserved | Reserved | Reserved |
| bit 7 | | | | bit 0 | | | |

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 ⁽¹⁾ | AVSS ⁽²⁾ |
| 001 | External VREF+ | External VREF- |
| 010 | Internal VRH | Internal VRL |
| 011 | External VREF+ | AVSS ⁽²⁾ |
| 100 | Internal VRH | AVSS ⁽²⁾ |
| 101 | AVDD ⁽¹⁾ | External VREF- |
| 110 | AVDD ⁽¹⁾ | Internal VRL |
| 111 | Internal VRL | AVSS |

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

Note 1: This parameter is VDD for the PIC16C717.

2: This parameter is Vss for the PIC16C717.

Legend:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

- n = Value at POR

'1' = Bit is set

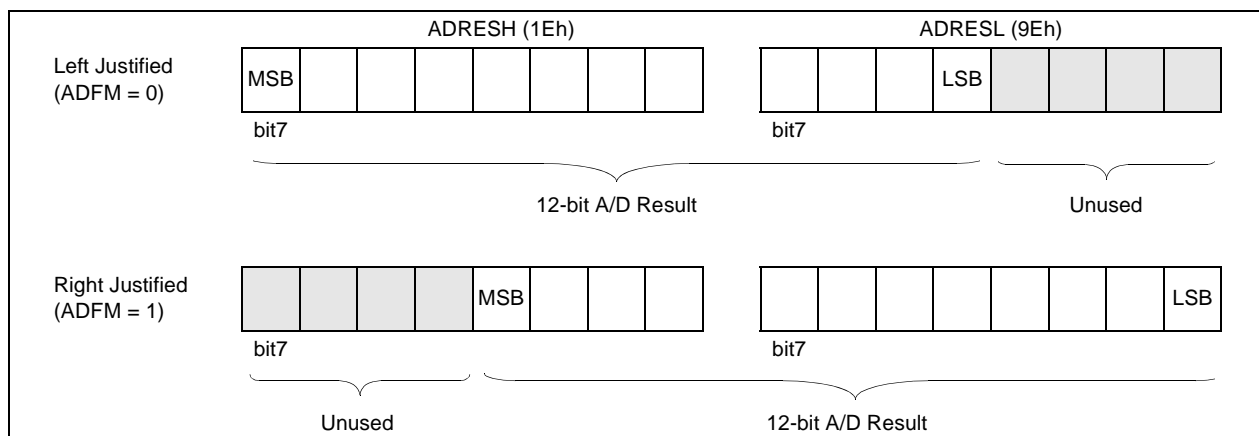
'0' = Bit is cleared

x = Bit is unknown

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.

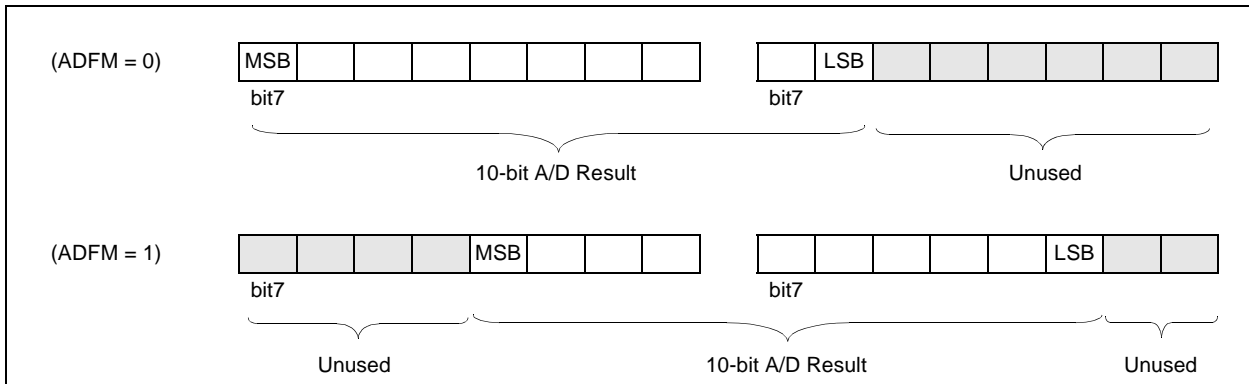
The A/D conversion results can be left justified (ADFM bit cleared), or right justified (ADFM bit set). Figure 11-1 through Figure 11-2 show the A/D result data format of the PIC16C717/770/771.

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



PIC16C717/770/771

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 (VOH or VOL) 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 register, all pins configured as analog input channels will read as '0'.
- 2:** When reading the PORTB register, all pins configured as analog pins on PORTB will be read as '1'.
- 3:** 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 (VREF+), or AVDD. The low reference input can come from an internal reference (VRL), an external reference (VREF-), or AVSS. 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 $\overline{\text{GO/DONE}}$ bit. The A/D conversion begins and lasts for 13TAD. The following steps should be followed for performing an A/D conversion:

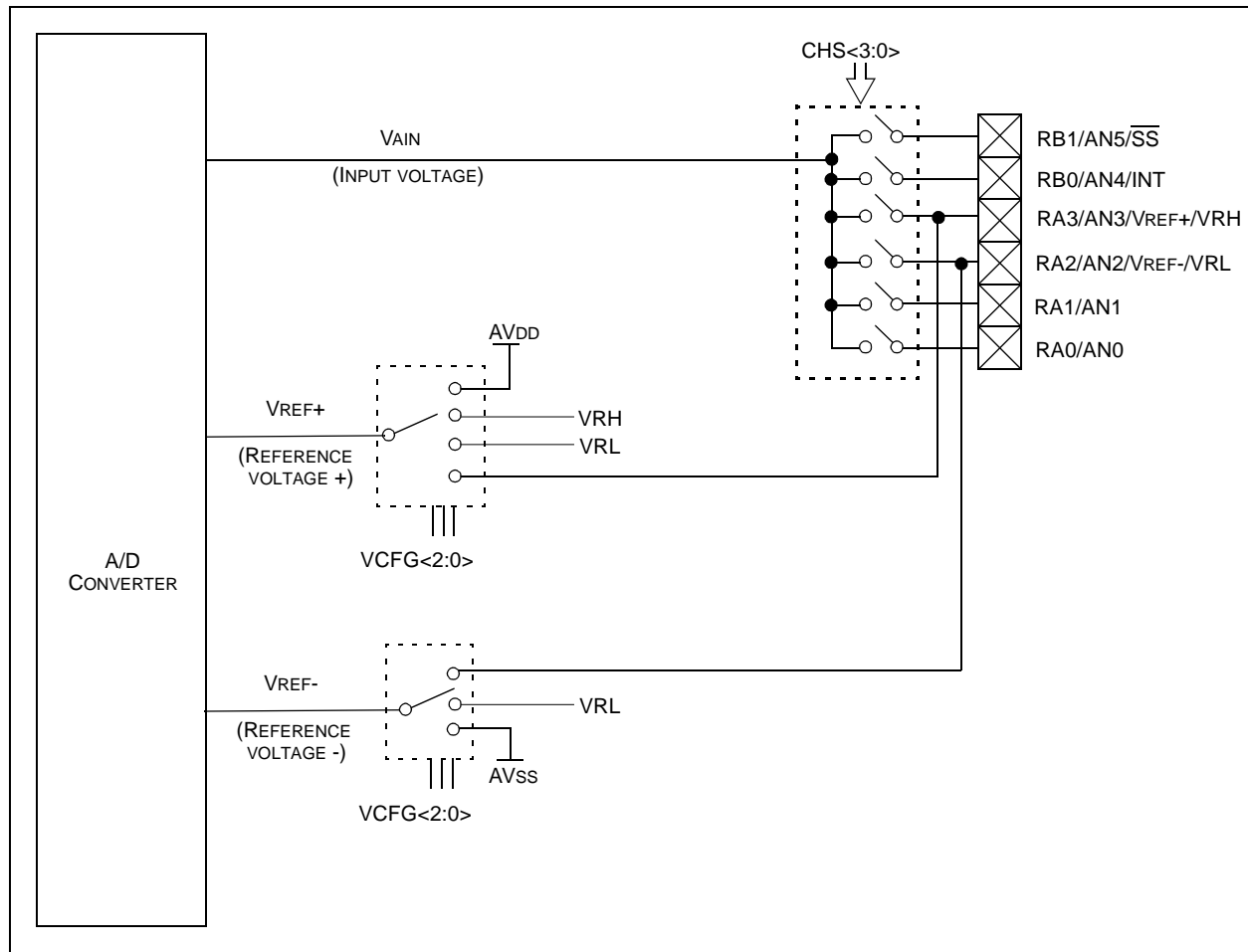
1. Configure port pins:
 - Configure Analog Input mode (ANSEL)
 - Configure pin as input (TRISA or TRISB)
2. Configure the A/D module
 - Configure A/D Result Format / voltage reference (ADCON1)
 - Select A/D input channel (ADCON0)
 - Select A/D conversion clock (ADCON0)
 - Turn on A/D module (ADCON0)
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.
5. START conversion
 - Set $\overline{\text{GO/DONE}}$ bit (ADCON0)
6. Wait 13TAD until A/D conversion is complete, by either:
 - Polling for the $\overline{\text{GO/DONE}}$ 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 $\overline{\text{GO/DONE}}$ 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 $\overline{\text{GO/DONE}}$ bit in the same instruction. Doing so will cause the $\overline{\text{GO/DONE}}$ bit to be automatically cleared.

FIGURE 11-3: A/D BLOCK DIAGRAM



PIC16C717/770/771

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 μ s. 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 μ s |
| | 8 TOSC | 01 | 400 ns ⁽²⁾ | 1.6 μ s | 2.0 μ s | 6.4 μ s |
| | 32 TOSC | 10 | 1.6 μ s | 6.4 μ s ⁽³⁾ | 8.0 μ s ⁽³⁾ | 25.6 μ s ⁽³⁾ |
| | A/D RC | 11 | 2 - 6 μ s ^(1,4) | 2 - 6 μ s ^(1,4) | 2 - 6 μ s ^(1,4) | 2 - 6 μ s ^(1,4) |
| Internal VRH or VRL | 16 TOSC | 00 | 800 ns ⁽²⁾ | 3.2 μ s ⁽²⁾ | 4 μ s ⁽²⁾ | 12.8 μ s |
| | 64 TOSC | 01 | 3.2 μ s ⁽²⁾ | 12.8 μ s | 16 μ s | 51.2 μ s ⁽³⁾ |
| | 256 TOSC | 10 | 12.8 μ s | 51.2 μ s ⁽³⁾ | 64 μ s ⁽³⁾ | 204.8 μ s ⁽³⁾ |
| | A/D RC | 11 | 16 - 48 μ s ^(4,5) | 16 - 48 μ s ^(4,5) | 16 - 48 μ s ^(4,5) | 16 - 48 μ s ^(4,5) |

Legend: Shaded cells are outside of recommended range.

Note 1: The A/D RC source has a typical TAD time of 4 μ s 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: A/D RC clock source has a typical TAD time of 32 μ s 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 VREF+ is the device AVDD and the analog reference VREF- is the device AVSS. The A/D interrupt is enabled and the A/D conversion clock is TRC. The conversion is performed on the AN0 channel.

EXAMPLE 11-1: PERFORMING AN A/D CONVERSION

```
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

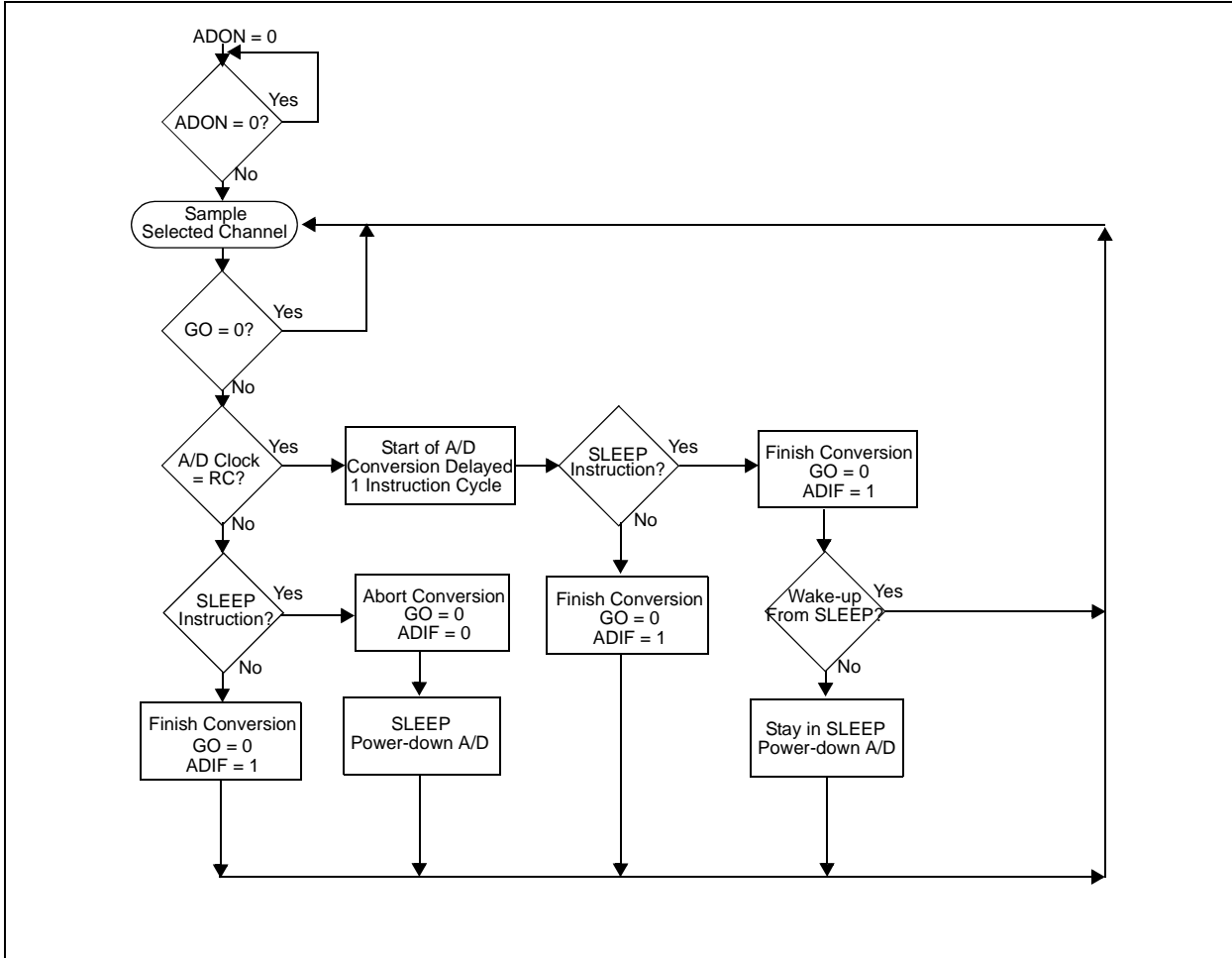
MOVWF  ADCON0        ;
BCF    PIR1, ADIF    ;Clear A/D Int Flag
BSF    INTCON, PEIE  ;Enable Peripheral
BSF    INTCON, GIE    ;Enable All Interrupts
;
; Ensure that the required sampling time for the
; selected input channel has lapsed. Then the
; conversion may be started.
BSF    ADCON0, GO     ;Start 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.
```

PIC16C717/770/771

11.5 A/D Converter Module Operation

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

FIGURE 11-4: FLOW CHART OF A/D OPERATION



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 (CHOLD) 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 (RS) and the internal sampling switch (RSS) impedance directly affect the time required to charge the capacitor CHOLD. The sampling switch (RSS) impedance varies over the device voltage (VDD), 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 CHOLD is assumed to be 25 pF for the 12-bit A/D.

EXAMPLE 11-2: A/D SAMPLING TIME EQUATION

$$V_{HOLD} = V_{REF} - \frac{V_{REF}}{16384}$$

$$V_{REF} - \frac{V_{REF}}{16384} = (V_{REF}) \left(1 - e^{\left(\frac{-Tc}{CHOLD(RIC + RSS + RS)} \right)} \right)$$

$$V_{REF} \left(1 - \frac{1}{16384} \right) = (V_{REF}) \left(1 - e^{\left(\frac{-Tc}{CHOLD(RIC + RSS + RS)} \right)} \right)$$

Solving for Tc:

$$Tc = -CHOLD(1k + RSS + RS) \ln \left(\frac{1}{16384} \right)$$

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

CHOLD = 25 pF

RS = 2.5 kΩ

1/4 LSB error

VDD = 5V → RSS = 10 kΩ (worst case)

Temp (system Max.) = 50°C

- Note 1:** The reference voltage (VREF) has no effect on the equation, since it cancels itself out.
- 2:** The charge holding capacitor (CHOLD) 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.

PIC16C717/770/771

EXAMPLE 11-3: CALCULATING THE MINIMUM REQUIRED SAMPLE TIME

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

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

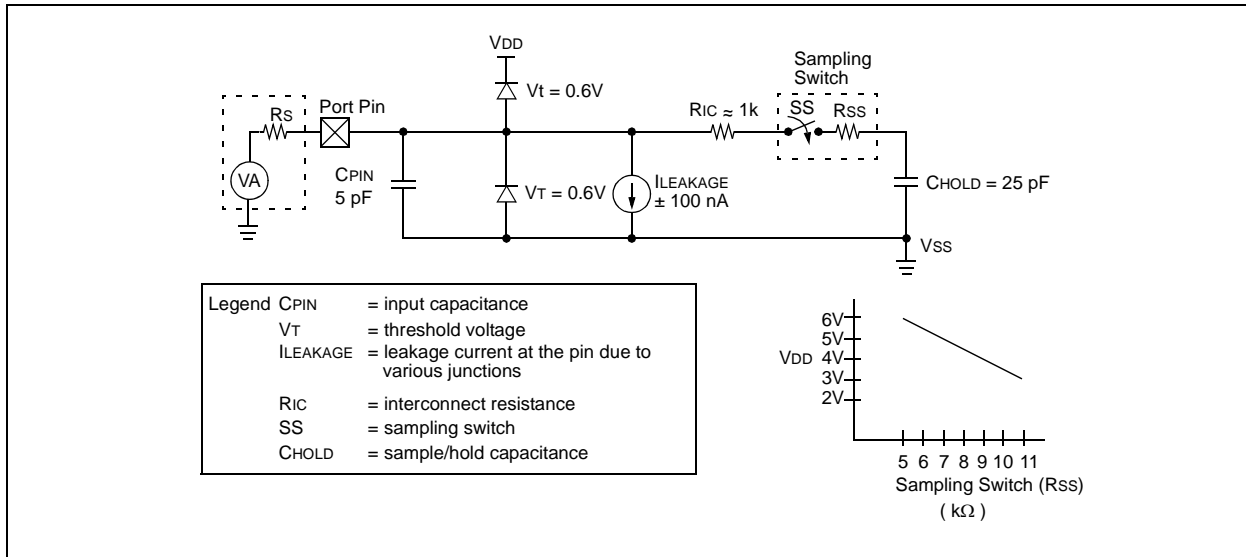
$T_C =$ Holding Capacitor Charging Time
 $T_C =$ (CHOLD) (RIC + RSS + RS) ln (1/16384)
 $T_C =$ -25 pF (1 k Ω + 10 k Ω + 2.5 k Ω) ln (1/16384)
 $T_C =$ -25 pF (13.5 k Ω) ln (1/16384)
 $T_C =$ -0.338 (-9.704) μ s
 $T_C =$ 3.3 μ s

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

$T_{ACQ} =$ 8.3 μ s + 1.25 μ s
 $T_{ACQ} =$ 9.55 μ s

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

FIGURE 11-5: ANALOG INPUT MODEL



11.7 Use of the ECCP 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/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/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/DONE bit after the desired number of bits in the result have been converted. Once the GO/DONE bit has been cleared, all of the remaining A/D result bits are '0'. The equation to determine the time before the GO/DONE bit can be switched is as follows:

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

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/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:

$$\begin{aligned}\text{Conversion Time} &= (N + 1) T_{AD} \\ &= (4 + 1) T_{AD} \\ &= (5)(1.6 \mu\text{S}) \\ &= 8 \mu\text{S}\end{aligned}$$

12-Bit Example:

$$\begin{aligned}\text{Conversion Time} &= (N + 1) T_{AD} \\ &= (12 + 1) T_{AD} \\ &= (13)(1.6 \mu\text{S}) \\ &= 20.8 \mu\text{S}\end{aligned}$$

PIC16C717/770/771

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/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/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> = 11).

11.11 Connection Considerations

Since the analog inputs employ ESD protection, they have diodes to VDD and VSS. This requires that the analog input must be between VDD and VSS. 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. It is recommended that any external components connected to an analog input pin (capacitor, zener diode, etc.) have very little leakage current.

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 | TOIE | INTE | RBIE | T0IF | INTF | RBFIF | 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/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 xx11 | uuuu uu11 |
| 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 |
| 17h | CCP1CON | — | — | — | — | — | — | — | — | 0000 0000 | 0000 0000 |

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 PIC Mid-Range MCU Family 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.

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 trip point, the Power-up Timer, the watchdog timer and the devices Oscillator mode. As can be seen in Register 12-1, some additional configuration word bits have been provided for Brown-out Reset trip point selection.

PIC16C717/770/771

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

| | | | | | | | | | | | | | |
|------------------------|------------------------|-------|-------|------------------------|------------------------|---|-------|-------|---------------------------|------|-------|-------|-------|
| $\overline{\text{CP}}$ | $\overline{\text{CP}}$ | BORV1 | BORV0 | $\overline{\text{CP}}$ | $\overline{\text{CP}}$ | — | BODEN | MCLRE | $\overline{\text{PWRTE}}$ | WDTE | FOSC2 | FOSC1 | FOSC0 |
| bit13 | | | | | | | | | | | | | bit0 |

bit 13-12, $\overline{\text{CP}}$: **Program Memory Code Protection**

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 = VBOR set to 4.5V
01 = VBOR set to 4.2V
10 = VBOR set to 2.7V
11 = VBOR 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/ $\overline{\text{MCLR}}$ pin function select

1 = RA5/ $\overline{\text{MCLR}}$ pin function is $\overline{\text{MCLR}}$
0 = RA5/ $\overline{\text{MCLR}}$ pin function is digital input, $\overline{\text{MCLR}}$ internally tied to VDD

bit 4: **$\overline{\text{PWRTE}}$** : 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: Crystal/Resonator on RA6/OSC2/CLKOUT and RA7/OSC1/CLKIN
001 = XT oscillator: Crystal/Resonator on RA6/OSC2/CLKOUT and RA7/OSC1/CLKIN
010 = HS oscillator: Crystal/Resonator 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 $\overline{\text{PWRTE}}$. Ensure the Power-up Timer is enabled anytime Brown-out Reset is enabled.

2: All of the $\overline{\text{CP}}$ bits must be given the same value to enable code protection.

Legend

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

1 = bit is set

0 = bit is cleared

x = bit is unknown

12.2 Oscillator Configurations

12.2.1 OSCILLATOR TYPES

The PIC16C717/770/771 can be operated in eight 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 12-1). 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 12-1: CRYSTAL/CERAMIC RESONATOR OPERATION (HS, XT OR LP OSC CONFIGURATION)

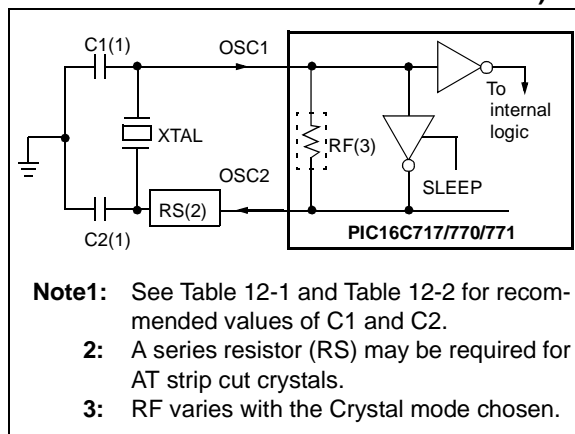


TABLE 12-1: CERAMIC RESONATORS

| Ranges Tested: | | | |
|----------------|----------|-------------|-------------|
| Mode | Freq | OSC1 | OSC2 |
| XT | 455 kHz | 68 - 100 pF | 68 - 100 pF |
| | 2.0 MHz | 15 - 68 pF | 15 - 68 pF |
| | 4.0 MHz | 15 - 68 pF | 15 - 68 pF |
| HS | 8.0 MHz | 10 - 68 pF | 10 - 68 pF |
| | 16.0 MHz | 10 - 22 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.

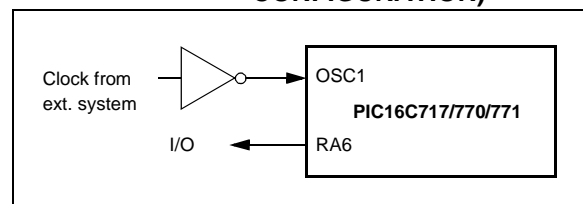
Note 1: Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components.

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 12-2 for illustration.

FIGURE 12-2: EXTERNAL CLOCK INPUT OPERATION (EC OSC CONFIGURATION)



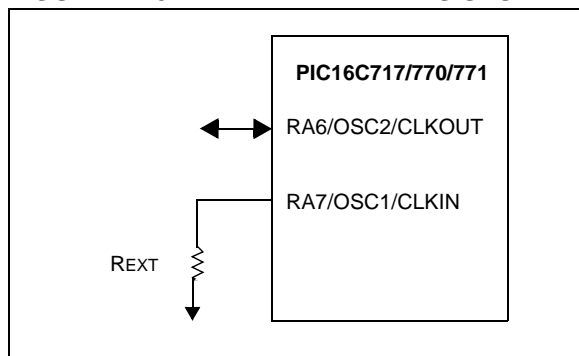
PIC16C717/770/771

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 12-3 shows how the controlling resistor is connected to the PIC16C717/770/771. For REXT values below 38 k Ω , 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 38 k Ω and 1 M Ω .

FIGURE 12-3: 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 37 kHz. 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 μ S to 300 μ S. For speed change from slow to fast, the processor is in active for 1.25 μ S to 3.25 μ S.

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)
- $\overline{\text{MCLR}}$ Reset during normal operation
- $\overline{\text{MCLR}}$ 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 Wake-up, which is viewed as the resumption of normal operation.

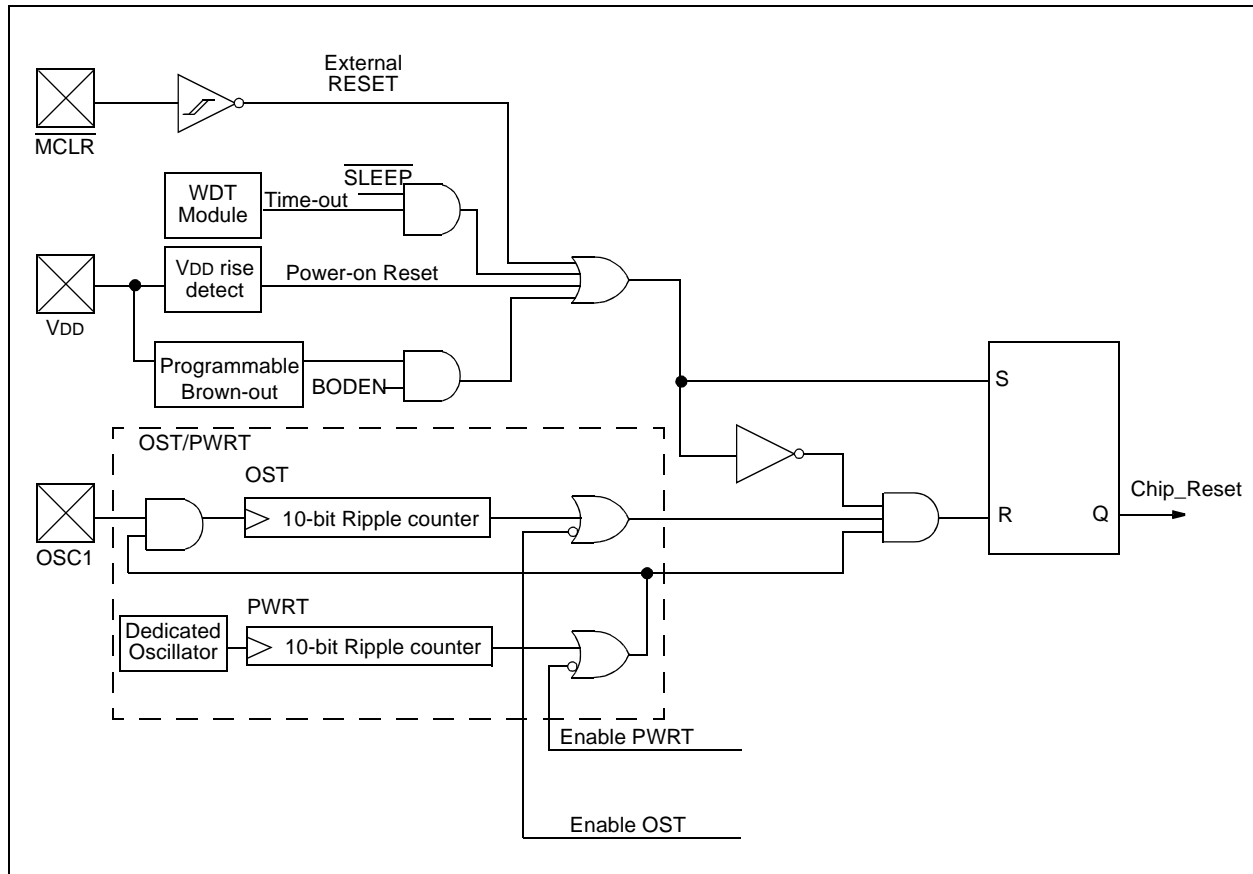
Several status bits have been provided to indicate which RESET occurred (see Table 12-4). See Table 12-6 for a full description of RESET states of all registers.

A simplified block diagram of the On-Chip Reset circuit is shown in Figure 12-4.

These devices have a $\overline{\text{MCLR}}$ noise filter in the $\overline{\text{MCLR}}$ Reset path. The filter will detect and ignore small pulses.

It should be noted that a WDT Reset does not drive $\overline{\text{MCLR}}$ pin low.

FIGURE 12-4: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



PIC16C717/770/771

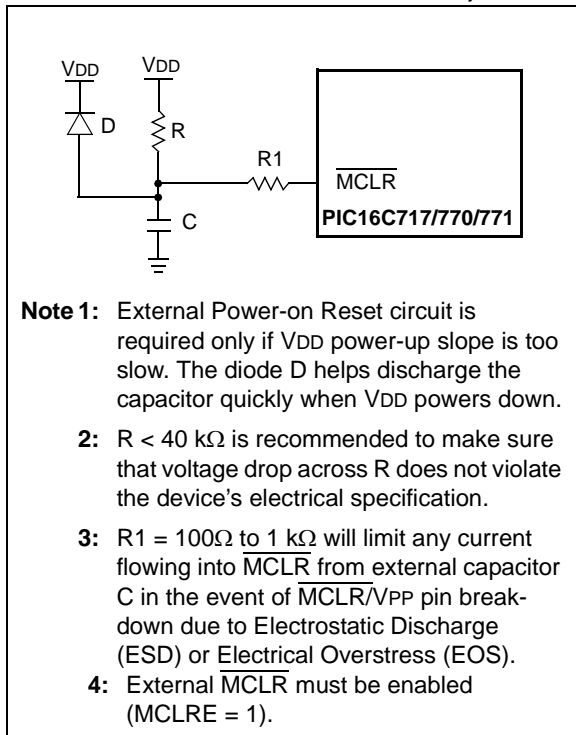
12.4 Power-On Reset (POR)

A Power-on Reset pulse is generated on-chip when a VDD rise is detected (in the range of 1.5V - 2.1V). Enable the internal MCLR feature to 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 long rise time, enable external MCLR function and use circuit as shown in Figure 12-5.

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 start-up conditions, or if necessary an external POR circuit may be implemented to delay end of RESET for as long as needed.

FIGURE 12-5: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW VDD RAMP)



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 trip point). 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 trip point for longer than TBOR, (parameter #35), the brown-out situation will RESET the chip. A RESET may not occur if VDD falls below the trip point 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 12-6, Figure 12-7, Figure 12-8 and Figure 12-9 depict time-out sequences on power-up.

Since the time-outs occur from the POR pulse, if $\overline{\text{MCLR}}$ is kept low long enough, the time-outs will expire. Then bringing $\overline{\text{MCLR}}$ high will begin execution immediately (Figure 12-8). This is useful for testing purposes or to synchronize more than one PIC[®] 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, $\overline{\text{BOR}}$. The $\overline{\text{BOR}}$ bit is unknown upon a POR. BOR must be set by the user and checked on subsequent RESETS to see if bit BOR cleared, indicating a BOR occurred.

Bit1 is $\overline{\text{POR}}$ (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 |
|--------------------------|------------------------------|------------------------------|------------------|--------------------|
| | $\overline{\text{PWRT}} = 0$ | $\overline{\text{PWRT}} = 1$ | | |
| XT, HS, LP | TPWRT + 1024TOSC | 1024TOSC | TPWRT + 1024TOSC | 1024TOSC |
| EC, ER, INTRC | TPWRT | — | TPWRT | — |

TABLE 12-4: STATUS BITS AND THEIR SIGNIFICANCE

| $\overline{\text{POR}}$ | $\overline{\text{BOR}}$ | $\overline{\text{TO}}$ | $\overline{\text{PD}}$ | |
|-------------------------|-------------------------|------------------------|------------------------|---|
| 0 | x | 1 | 1 | Power-on Reset |
| 0 | x | 0 | x | Illegal, $\overline{\text{TO}}$ is set on $\overline{\text{POR}}$ |
| 0 | x | x | 0 | Illegal, $\overline{\text{PD}}$ is set on $\overline{\text{POR}}$ |
| 1 | 0 | 1 | 1 | Brown-out Reset |
| 1 | 1 | 0 | 1 | WDT Reset |
| 1 | 1 | 0 | 0 | WDT Wake-up |
| 1 | 1 | u | u | $\overline{\text{MCLR}}$ Reset during normal operation |
| 1 | 1 | 1 | 0 | $\overline{\text{MCLR}}$ 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 | ---- 1-0x |
| $\overline{\text{MCLR}}$ Reset during normal operation | 000h | 000u uuuu | ---- 1-uu |
| $\overline{\text{MCLR}}$ 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'.

PIC16C717/770/771

TABLE 12-6: INITIALIZATION CONDITIONS FOR ALL REGISTERS

| Register | Power-on Reset or Brown-out Reset | 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 xx11 | uuuu uu11 | uuuu uuuu |
| PCLATH | ---0 0000 | ---0 0000 | ---u uuuu |
| INTCON | 0000 000x | 0000 000u | uuuu uuq q |
| 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-q q | ---- 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 |

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)

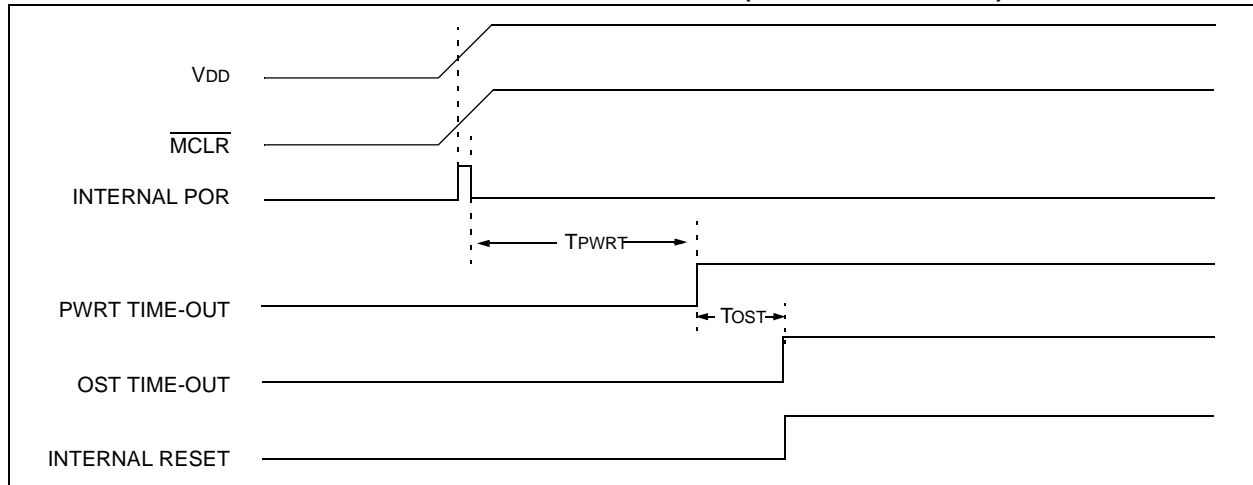
| Register | Power-on Reset or Brown-out Reset | $\overline{\text{MCLR}}$ Reset or WDT Reset | Wake-up via WDT or Interrupt |
|----------|-----------------------------------|---|------------------------------|
| P1DEL | 0000 0000 | 0000 0000 | uuuu uuuu |
| REFCON | 0000 ---- | 0000 ---- | uuuu ---- |
| LVDCON | --00 0101 | --00 0101 | --uu uuuu |
| ANSEL | --11 1111 | --11 1111 | --uu uuuu |
| ADRESL | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| ADCON1 | 0000 0000 | 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 12-6: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ TIED TO V_{DD})



PIC16C717/770/771

FIGURE 12-7: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ NOT TIED TO V_{DD}): CASE 1

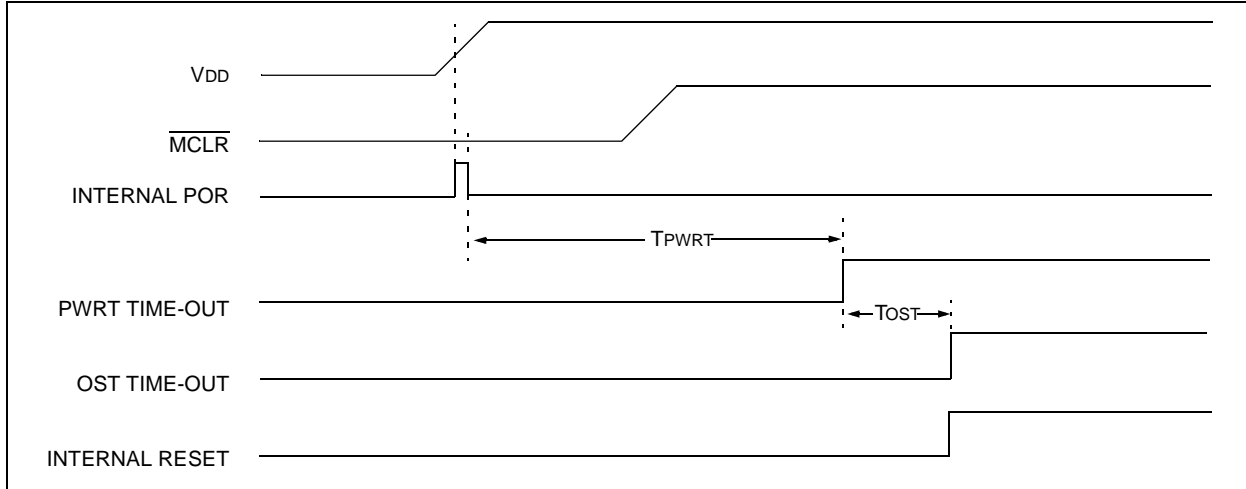


FIGURE 12-8: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ NOT TIED TO V_{DD}): CASE 2

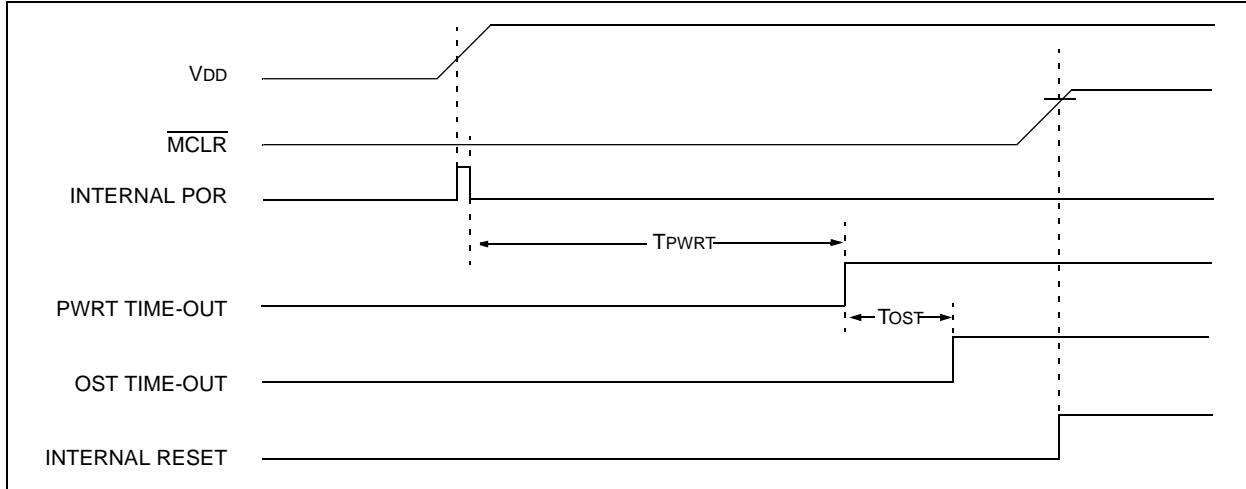
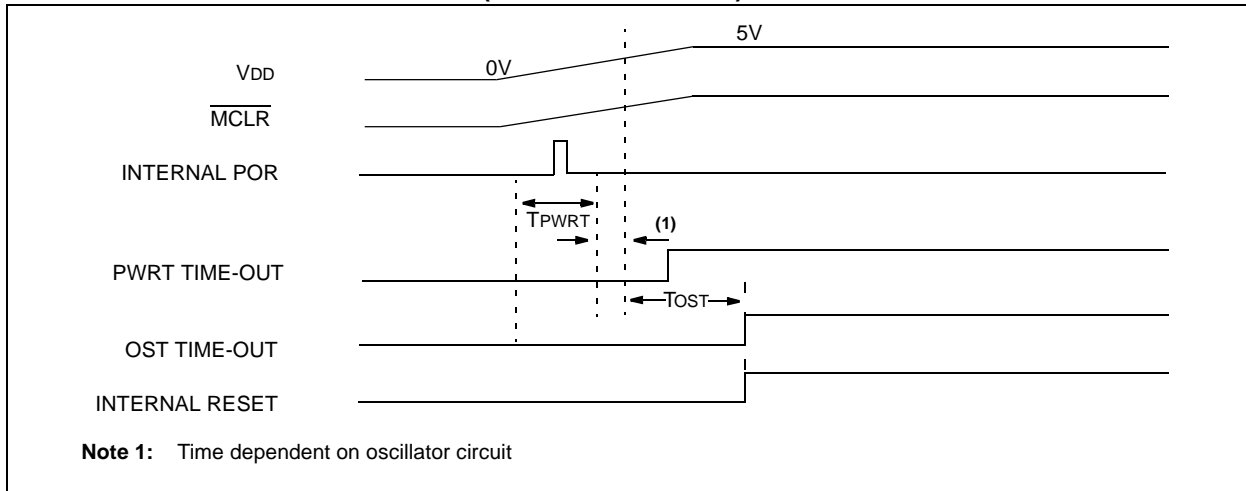


FIGURE 12-9: SLOW V_{DD} RISE TIME ($\overline{\text{MCLR}}$ TIED TO V_{DD})



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.

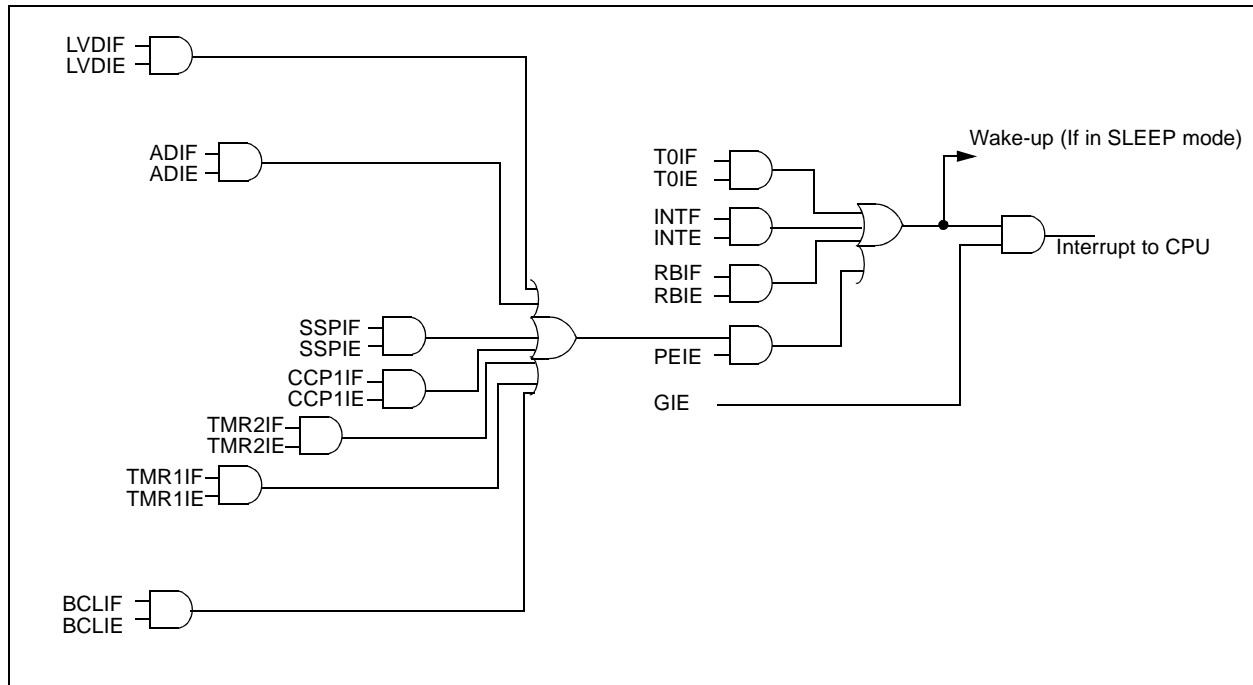
The RB0/INT pin interrupt, the RB port change interrupt and the TMR0 overflow interrupt flags are contained in the INTCON register.

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 12-10: INTERRUPT LOGIC



PIC16C717/770/771

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 → 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
```


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 independent from the processor clock. If enabled, 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{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* to '0' (Section 12.1).

WDT time-out period values may be found in Table 15-4. Values for the WDT prescaler may be assigned using the *OPTION_REG* register.

Note: The *SLEEP* instruction clears the WDT and the postscaler, if assigned to the WDT, restarting the WDT period.

FIGURE 12-11: WATCHDOG TIMER BLOCK DIAGRAM

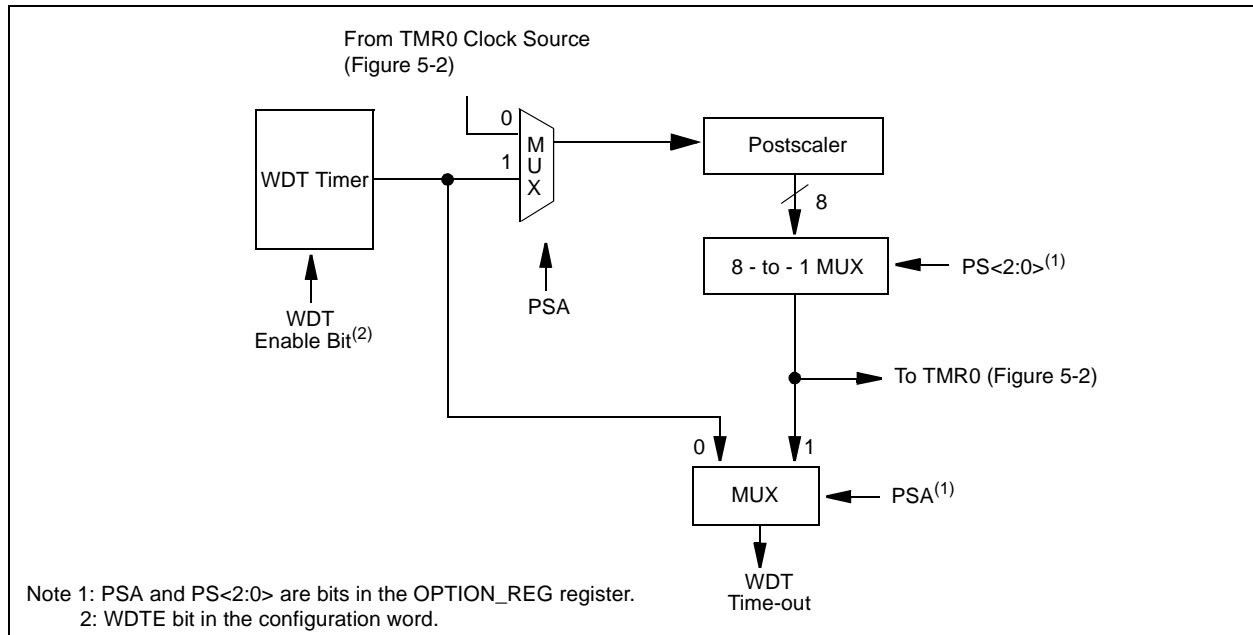


TABLE 12-7: SUMMARY OF WATCHDOG TIMER REGISTERS

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|----------|-----------------------------|-------------------|--------|-------|--------------------|-------|-------|-------|-------|
| 2007h | Config. bits ⁽¹⁾ | — | BODEN | MCLRE | \overline{PWRTE} | WDTE | FOSC2 | FOSC1 | FOSC0 |
| 81h,181h | OPTION_REG | \overline{RBPU} | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 |

Legend: Shaded cells are not used by the Watchdog Timer.

Note 1: See Register 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 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 VDD, or VSS, 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 VDD or VSS 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 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/I²C).
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

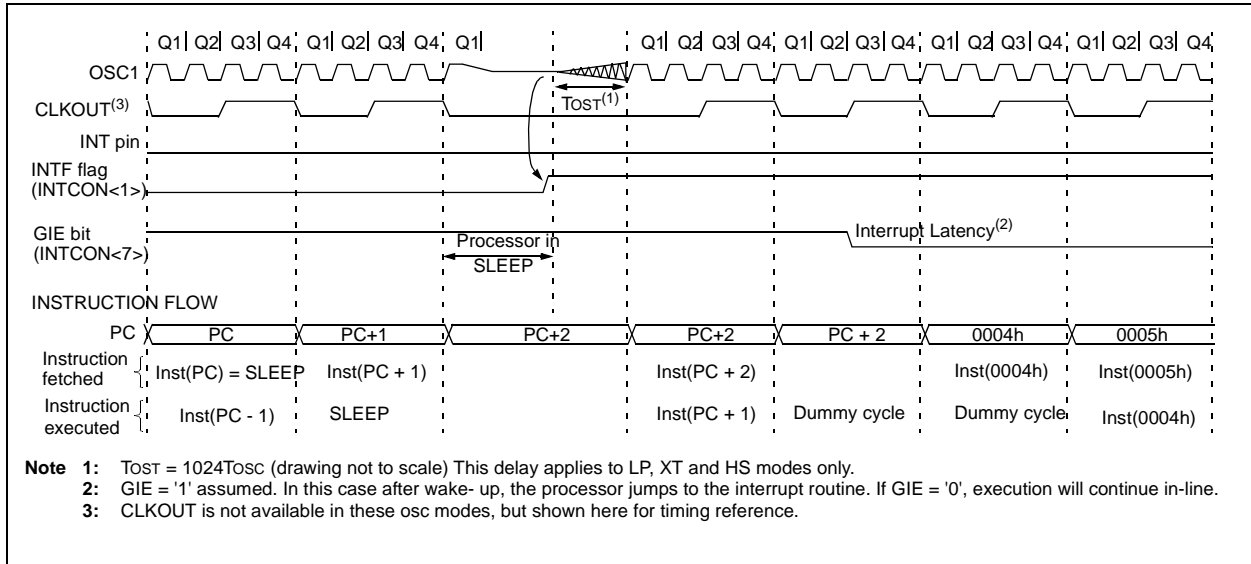
When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- 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`.

If a peripheral can wake the device from SLEEP, then to ensure that the WDT is cleared, a `CLRWDT` instruction should be executed before a `SLEEP` instruction.

FIGURE 12-12: WAKE-UP FROM SLEEP THROUGH INTERRUPT



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. Code protected devices are not reprogrammable.

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.

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).

PIC16C717/770/771

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 |
| TO | Time-out bit |
| 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.

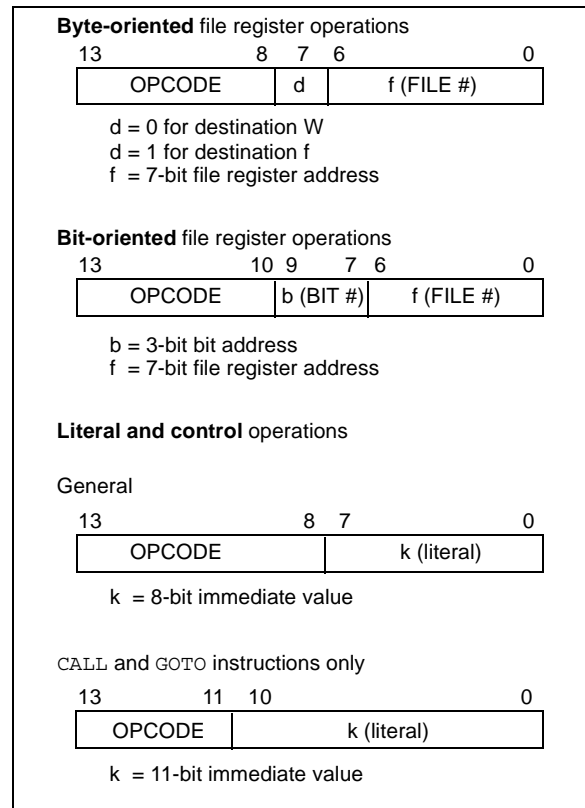
Note: To maintain upward compatibility with future PIC16CXXX products, do not use the OPTION and TRIS instructions.

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 PIC Mid-Range MCU Family Reference Manual, (DS33023).

PIC16C717/770/771

TABLE 13-2: PIC16CXXX INSTRUCTION SET

| Mnemonic, Operands | Description | Cycles | 14-Bit Opcode | | | Status Affected | Notes | | |
|---|-------------|------------------------------|---------------|-----|------|--------------------|-------|--------------------------------|-------|
| | | | MSb | LSb | | | | | |
| 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 | 1fff | ffff | Z | 2 |
| CLRWF | - | 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 | 1fff | 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 nybbles 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: $0 \leq k \leq 255$
Operation: $(W) + k \rightarrow (W)$
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.

ANDWF **AND W with f**

Syntax: *[label]* ANDWF f,d
Operands: $0 \leq f \leq 127$
 $d \in [0,1]$
Operation: $(W) \text{ .AND. } (f) \rightarrow (\text{destination})$
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'.

ADDWF **Add W and f**

Syntax: *[label]* ADDWF f,d
Operands: $0 \leq f \leq 127$
 $d \in [0,1]$
Operation: $(W) + (f) \rightarrow (\text{destination})$
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'.

BCF **Bit Clear f**

Syntax: *[label]* BCF f,b
Operands: $0 \leq f \leq 127$
 $0 \leq b \leq 7$
Operation: $0 \rightarrow (f)$
Status Affected: None
Description: Bit 'b' in register 'f' is cleared.

ANDLW **AND Literal with W**

Syntax: *[label]* ANDLW k
Operands: $0 \leq k \leq 255$
Operation: $(W) \text{ .AND. } (k) \rightarrow (W)$
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.

BSF **Bit Set f**

Syntax: *[label]* BSF f,b
Operands: $0 \leq f \leq 127$
 $0 \leq b \leq 7$
Operation: $1 \rightarrow (f)$
Status Affected: None
Description: Bit 'b' in register 'f' is set.

PIC16C717/770/771

| BTFSS | Bit Test f, Skip if Set |
|------------------|--|
| Syntax: | <code>[label] BTFSS f,b</code> |
| Operands: | $0 \leq f \leq 127$ $0 \leq b < 7$ |
| 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 <code>NOP</code> is executed instead making this a 2TCY instruction. |

| CLRF | Clear f |
|------------------|--|
| Syntax: | <code>[label] CLRF f</code> |
| Operands: | $0 \leq f \leq 127$ |
| Operation: | $00h \rightarrow (f)$ $1 \rightarrow Z$ |
| Status Affected: | Z |
| Description: | The contents of register 'f' are cleared and the Z bit is set. |

| BTFSC | Bit Test, Skip if Clear |
|------------------|---|
| Syntax: | <code>[label] BTFSC f,b</code> |
| Operands: | $0 \leq f \leq 127$ $0 \leq b \leq 7$ |
| 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 <code>NOP</code> is executed instead, making this a 2TCY instruction. |

| CLRW | Clear W |
|------------------|---|
| Syntax: | <code>[label] CLRW</code> |
| Operands: | None |
| Operation: | $00h \rightarrow (W)$ $1 \rightarrow Z$ |
| Status Affected: | Z |
| Description: | W register is cleared. Zero bit (Z) is set. |

| CALL | Call Subroutine |
|------------------|---|
| Syntax: | <code>[label] CALL k</code> |
| Operands: | $0 \leq k \leq 2047$ |
| 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. <code>CALL</code> is a two cycle instruction. |

| CLRWDT | Clear Watchdog Timer |
|------------------|--|
| Syntax: | <code>[label] CLRWDT</code> |
| Operands: | None |
| Operation: | $00h \rightarrow WDT$ $0 \rightarrow WDT \text{ prescaler},$ $1 \rightarrow \overline{TO}$ $1 \rightarrow \overline{PD}$ |
| Status Affected: | $\overline{TO}, \overline{PD}$ |
| Description: | <code>CLRWDT</code> 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: | [<i>label</i>] COMF f,d |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ |
| Operation: | $(\bar{f}) \rightarrow (\text{destination})$ |
| 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'. |

| GOTO | Unconditional Branch |
|------------------|---|
| Syntax: | [<i>label</i>] GOTO k |
| Operands: | $0 \leq k \leq 2047$ |
| Operation: | $k \rightarrow \text{PC}<10:0>$ $\text{PCLATH}<4:3> \rightarrow \text{PC}<12:11>$ |
| 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. |

| DECF | Decrement f |
|------------------|--|
| Syntax: | [<i>label</i>] DECF f,d |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ |
| Operation: | $(f) - 1 \rightarrow (\text{destination})$ |
| 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'. |

| INCF | Increment f |
|------------------|--|
| Syntax: | [<i>label</i>] INCF f,d |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ |
| Operation: | $(f) + 1 \rightarrow (\text{destination})$ |
| 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'. |

| DECFSZ | Decrement f, Skip if 0 |
|------------------|--|
| Syntax: | [<i>label</i>] DECFSZ f,d |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ |
| Operation: | $(f) - 1 \rightarrow (\text{destination});$ 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. |

| INCFSZ | Increment f, Skip if 0 |
|------------------|---|
| Syntax: | [<i>label</i>] INCFSZ f,d |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ |
| Operation: | $(f) + 1 \rightarrow (\text{destination}),$ 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. |

PIC16C717/770/771

IORLW **Inclusive OR Literal with W**

Syntax: [*label*] IORLW *k*

Operands: $0 \leq k \leq 255$

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.

MOVLW **Move Literal to W**

Syntax: [*label*] MOVLW *k*

Operands: $0 \leq k \leq 255$

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.

IORWF **Inclusive OR W with f**

Syntax: [*label*] IORWF *f,d*

Operands: $0 \leq f \leq 127$
d ∈ [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'.

MOVWF **Move W to f**

Syntax: [*label*] MOVWF *f*

Operands: $0 \leq f \leq 127$

Operation: (W) → (*f*)

Status Affected: None

Description: Move data from W register to register 'f'.

MOVF **Move f**

Syntax: [*label*] MOVF *f,d*

Operands: $0 \leq f \leq 127$
d ∈ [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.

NOP **No Operation**

Syntax: [*label*] NOP

Operands: None

Operation: No operation

Status Affected: None

Description: No operation.

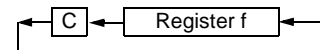
PIC16C717/770/771

RETFIE **Return from Interrupt**

Syntax: [*label*] RETFIE
Operands: None
Operation: TOS → PC,
 1 → GIE
Status Affected: None

RLF **Rotate Left f through Carry**

Syntax: [*label*] RLF f,d
Operands: $0 \leq f \leq 127$
 $d \in [0,1]$
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'.

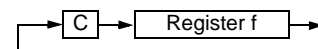


RETLW **Return with Literal in W**

Syntax: [*label*] RETLW k
Operands: $0 \leq k \leq 255$
Operation: $k \rightarrow (W)$;
 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.

RRF **Rotate Right f through Carry**

Syntax: [*label*] RRF f,d
Operands: $0 \leq f \leq 127$
 $d \in [0,1]$
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'.



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.

SLEEP

Syntax: [*label* SLEEP
]
Operands: None
Operation: 00h → WDT,
 0 → WDT prescaler,
 1 → \overline{TO} ,
 0 → 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 12.8 for more details.

PIC16C717/770/771

SUBLW **Subtract W from Literal**

Syntax: *[label]* SUBLW *k*

Operands: $0 \leq k \leq 255$

Operation: $k - (W) \rightarrow (W)$

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.

XORLW **Exclusive OR Literal with W**

Syntax: *[label]* XORLW *k*

Operands: $0 \leq k \leq 255$

Operation: $(W) .XOR. k \rightarrow (W)$

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.

SUBWF **Subtract W from f**

Syntax: *[label]* SUBWF *f,d*

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: $(f) - (W) \rightarrow (\text{destination})$

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'.

XORWF **Exclusive OR W with f**

Syntax: *[label]* XORWF *f,d*

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: $(W) .XOR. (f) \rightarrow (\text{destination})$

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'.

SWAPF **Swap Nybbles in f**

Syntax: *[label]* SWAPF *f,d*

Operands: $0 \leq f \leq 127$
 $d \in [0,1]$

Operation: $(f<3:0>) \rightarrow (\text{destination}<7:4>),$
 $(f<7:4>) \rightarrow (\text{destination}<3:0>)$

Status Affected: None

Description: The upper and lower nybbles 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'.

14.0 DEVELOPMENT SUPPORT

The PIC® 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™ Object Linker/
MPLIB™ Object Librarian
- Simulators
 - MPLAB SIM Software Simulator
- Emulators
 - MPLAB ICE 2000 In-Circuit Emulator
 - ICEPIC™ In-Circuit Emulator
- In-Circuit Debugger
 - MPLAB ICD
- Device Programmers
 - PRO MATE® II Universal Device Programmer
 - PICSTART® Plus Entry-Level Development Programmer
- Low Cost Demonstration Boards
 - PICDEM™ 1 Demonstration Board
 - PICDEM 2 Demonstration Board
 - PICDEM3 Demonstration Board
 - PICDEM 17 Demonstration Board
 - KEELOQ® Demonstration Board

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. The MPLAB IDE is a Windows®-based application that contains:

- An interface to debugging tools
 - simulator
 - programmer (sold separately)
 - emulator (sold separately)
 - in-circuit debugger (sold separately)
- A full-featured editor
- A project manager
- Customizable toolbar and key mapping
- A status bar
- On-line help

The MPLAB IDE allows you to:

- Edit your source files (either assembly or 'C')
- One touch assemble (or compile) and download to PIC MCU emulator and simulator tools (automatically updates all project information)
- Debug using:
 - source files
 - absolute listing file
 - machine code

The ability to use MPLAB IDE with multiple debugging tools allows users to easily switch from the cost-effective simulator to a full-featured emulator with minimal retraining.

14.2 MPASM Assembler

The MPASM assembler is a full-featured universal macro assembler for all PIC MCUs.

The MPASM assembler has a command line interface and a Windows shell. It can be used as a stand-alone application on a Windows 3.x or greater system, or it can be used through MPLAB IDE. The MPASM assembler generates relocatable object files for the MPLINK object linker, Intel® standard HEX files, MAP files to detail memory usage and symbol reference, an absolute LST file that contains source lines and generated machine code, and a COD file for debugging.

The MPASM assembler features include:

- Integration into MPLAB IDE projects.
- User-defined macros to streamline assembly code.
- Conditional assembly for multi-purpose source files.
- Directives that 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 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 Object Linker/ MPLIB Object Librarian

The MPLINK object linker combines relocatable objects created by the MPASM assembler and the MPLAB C17 and MPLAB C18 C compilers. It can also link relocatable objects from pre-compiled libraries, using directives from a linker script.

The MPLIB object librarian is a librarian for pre-compiled code to be used with the MPLINK object linker. When a routine from a library is called from another source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications. The MPLIB object librarian manages the creation and modification of library files.

The MPLINK object linker features include:

- Integration with MPASM assembler and MPLAB C17 and MPLAB C18 C compilers.
- Allows all memory areas to be defined as sections to provide link-time flexibility.

The MPLIB object librarian features include:

- Easier linking because single libraries can be included instead of many smaller files.
- Helps keep code maintainable by grouping related modules together.
- Allows 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-hosted environment by simulating the PIC 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.

The MPLAB SIM simulator fully supports symbolic debugging using the MPLAB C17 and the MPLAB C18 C compilers and the MPASM assembler. 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 PIC microcontrollers (MCUs). Software control of the MPLAB ICE in-circuit emulator is provided by the MPLAB Integrated Development Environment (IDE), which allows editing, building, downloading and source debugging from a single environment.

The MPLAB ICE 2000 is a full-featured emulator system with enhanced trace, trigger and data monitoring features. Interchangeable processor modules allow the system to be easily reconfigured for emulation of different processors. The universal architecture of the MPLAB ICE in-circuit emulator allows expansion to support new PIC microcontrollers.

The MPLAB ICE in-circuit 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 environment were chosen to best make these features available to you, the end user.

14.7 ICEPIC In-Circuit Emulator

The ICEPIC low cost, in-circuit emulator is a 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.

14.8 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 PIC MCUs and can be used to develop for this and other PIC microcontrollers. The MPLAB ICD utilizes the in-circuit debugging capability built into the FLASH devices. This feature, along with Microchip's In-Circuit Serial Programming™ protocol, offers cost-effective in-circuit FLASH 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.

14.9 PRO MATE II Universal Device Programmer

The PRO MATE II universal device programmer is a full-featured programmer, capable of operating in Stand-alone mode, as well as PC-hosted mode. The PRO MATE II device programmer is CE compliant.

The PRO MATE II device programmer has programmable VDD and VPP supplies, which allow it to verify programmed memory at VDD min and VDD 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 device programmer can read, verify, or program PIC devices. It can also set code protection in this mode.

14.10 PICSTART Plus Entry Level Development Programmer

The PICSTART Plus development programmer is an easy-to-use, low cost, prototype programmer. It connects to the PC via a COM (RS-232) port. MPLAB Integrated Development Environment software makes using the programmer simple and efficient.

The PICSTART Plus development programmer supports all PIC devices with up to 40 pins. Larger pin count devices, such as the PIC16C92X and PIC17C76X, may be supported with an adapter socket. The PICSTART Plus development programmer is CE compliant.

14.11 PICDEM 1 Low Cost PIC MCU Demonstration Board

The PICDEM 1 demonstration board 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 user can program the sample microcontrollers provided with the PICDEM 1 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The user can also connect the PICDEM 1 demonstration board to the MPLAB ICE in-circuit emulator and download the firmware to the emulator for testing. A 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.

14.12 PICDEM 2 Low Cost PIC16CXX Demonstration Board

The PICDEM 2 demonstration board 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 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The MPLAB ICE in-circuit emulator may also be used with the PICDEM 2 demonstration board to test firmware. A 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 I²C™ bus and separate headers for connection to an LCD module and a keypad.

PIC16C717/770/771

14.13 PICDEM 3 Low Cost PIC16CXXX Demonstration Board

The PICDEM 3 demonstration board 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 an 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 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer with an adapter socket, and easily test firmware. The MPLAB ICE in-circuit emulator may also be used with the PICDEM 3 demonstration board to test firmware. A prototype area has been provided to the user for adding 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 thermistor and separate headers for connection to an external LCD module and a keypad. Also provided on the PICDEM 3 demonstration board is a LCD panel, with 4 commons and 12 segments, that is capable of displaying time, temperature and day of the week. The PICDEM 3 demonstration board provides an additional RS-232 interface and Windows 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.

14.14 PICDEM 17 Demonstration Board

The PICDEM 17 demonstration board is an evaluation board that demonstrates the capabilities of several Microchip microcontrollers, including PIC17C752, PIC17C756A, 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 device programmer, or the PICSTART Plus development programmer, and easily debug and test the sample code. In addition, the PICDEM 17 demonstration board supports downloading of programs to and executing out of external FLASH memory on board. The PICDEM 17 demonstration board is also usable with the MPLAB ICE in-circuit emulator, or the 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.

14.15 KEELOQ Evaluation and Programming Tools

KEELOQ evaluation and programming tools support Microchip's HCS Secure Data Products. The HCS evaluation kit includes a 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

| Tool | PIC12CXX | PIC1400 | PIC16C5X | PIC16C6X | PIC16CXX | PIC16F62X | PIC16C7X | PIC16C7XX | PIC16C8X | PIC16F8XX | PIC16C9XX | PIC17C4X | PIC17C7XX | PIC18CXX2 | PIC18FXX | 24CXX/ 25CXX/ 93CXX | HCSXX | MCRFXXX | MCP2510 |
|---|----------|---------|----------|----------|----------|-----------|----------|-----------|----------|-----------|-----------|----------|-----------|-----------|----------|---------------------------|-------|---------|---------|
| Software Tools | | | | | | | | | | | | | | | | | | | |
| MPLAB® Integrated Development Environment | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| MPLAB® C17 C Compiler | | | | | | | | | | | | | | | | | | | |
| MPLAB® C18 C Compiler | | | | | | | | | | | | | | | | | | | |
| MPASM™ Assembler/ MPLINK™ Object Linker | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| MPLAB® ICE In-Circuit Emulator | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| ICEPIC™ In-Circuit Emulator | ✓ | | ✓ | ✓ | ✓ | | ✓ | ✓ | ✓ | ✓ | ✓ | | | | | | | | |
| Debugger | | | | | | | | | | | | | | | | | | | |
| MPLAB® ICD In-Circuit Debugger | | | | ✓* | | | ✓* | | | ✓ | | | | | ✓ | | | | |
| Programmers | | | | | | | | | | | | | | | | | | | |
| PICSTART® Plus Entry Level Development Programmer | ✓ | ✓ | ✓ | ✓ | ✓ | ✓** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PRO MATE® II Universal Device Programmer | ✓ | ✓ | ✓ | ✓ | ✓ | ✓** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Demo Boards and Eval Kits | | | | | | | | | | | | | | | | | | | |
| PICDEM™ 1 Demonstration Board | | | ✓ | | ✓ | | † | | ✓ | | | | | | | | | | |
| PICDEM™ 2 Demonstration Board | | | | ✓† | | | † | | | | | | | ✓ | | | | | |
| PICDEM™ 3 Demonstration Board | | | | | | | | | | ✓ | | | | | | | | | |
| PICDEM™ 14A Demonstration Board | | ✓ | | | | | | | | | | | ✓ | | | | | | |
| PICDEM™ 17 Demonstration Board | | | | | | | | | | | | | ✓ | | | | | | |
| 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.

PIC16C717/770/771

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 VSS (except VDD, $\overline{\text{MCLR}}$ and RA4) | -0.3V to (VDD + 0.3V) |
| Voltage on VDD with respect to VSS | -0.3 to +7.5V |
| Maximum voltage between AVDD and VDD pins..... | ± 0.3V |
| Maximum voltage between AVSS and VSS pins | ± 0.3V |
| Voltage on $\overline{\text{MCLR}}$ with respect to VSS..... | -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 VSS pin | 300 mA |
| Maximum current into VDD pin | 250 mA |
| Input clamp current, I _{IK} (V _I < 0 or V _I > VDD)..... | ± 20 mA |
| Output clamp current, I _{OK} (V _O < 0 or V _O > VDD) | ± 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: $P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$.

† 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.

PIC16C717/770/771

FIGURE 15-1: PIC16C717/770/771 VOLTAGE-FREQUENCY GRAPH, $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$

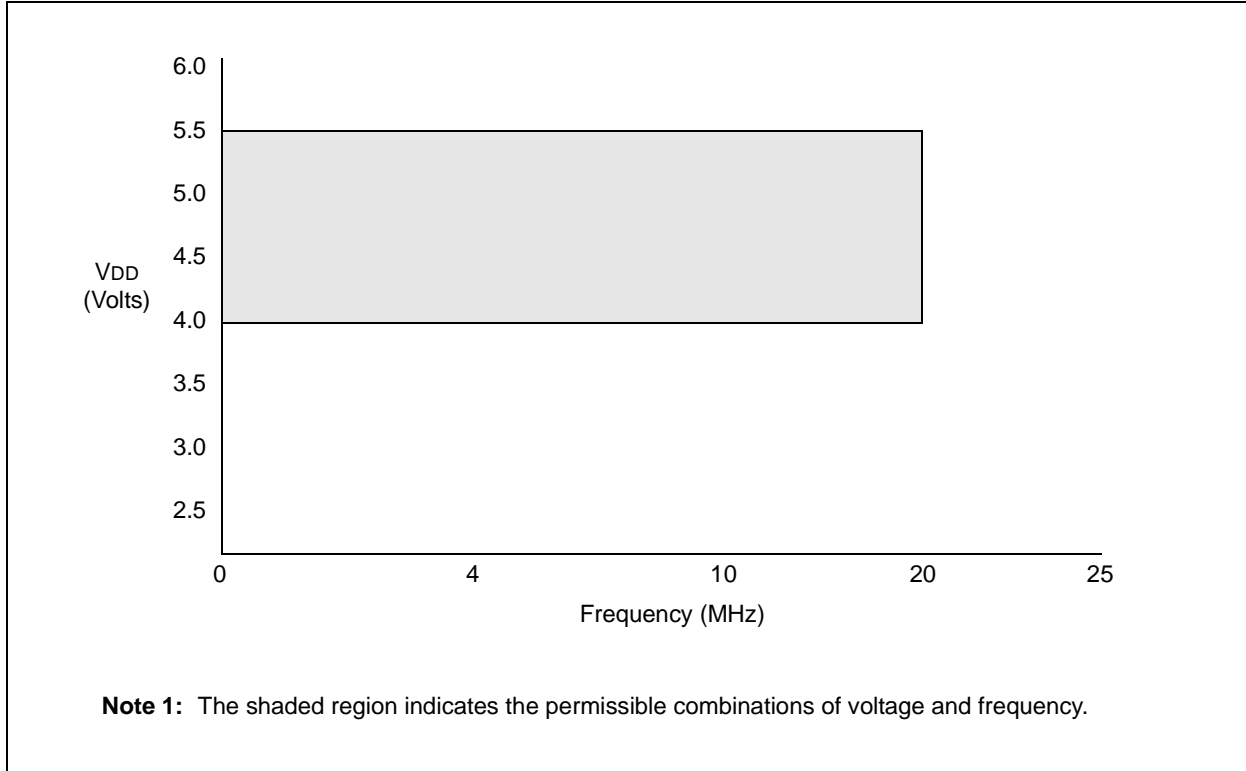
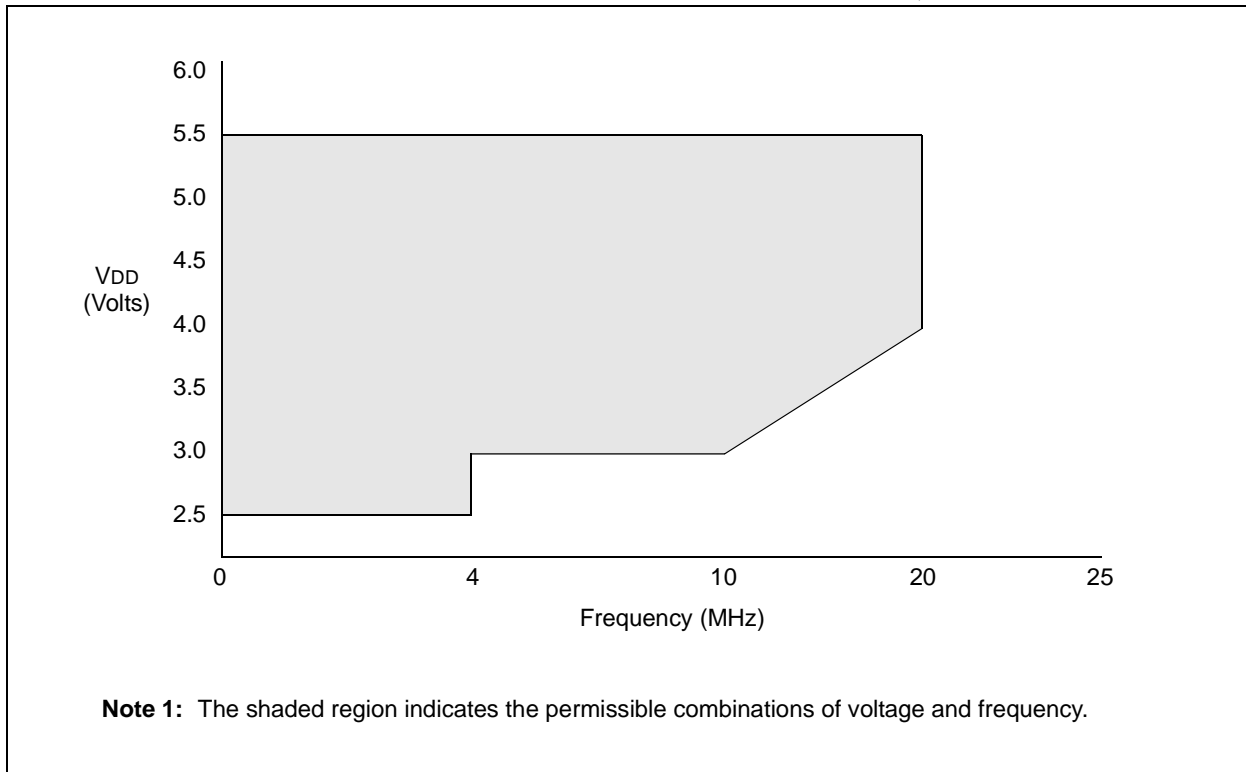
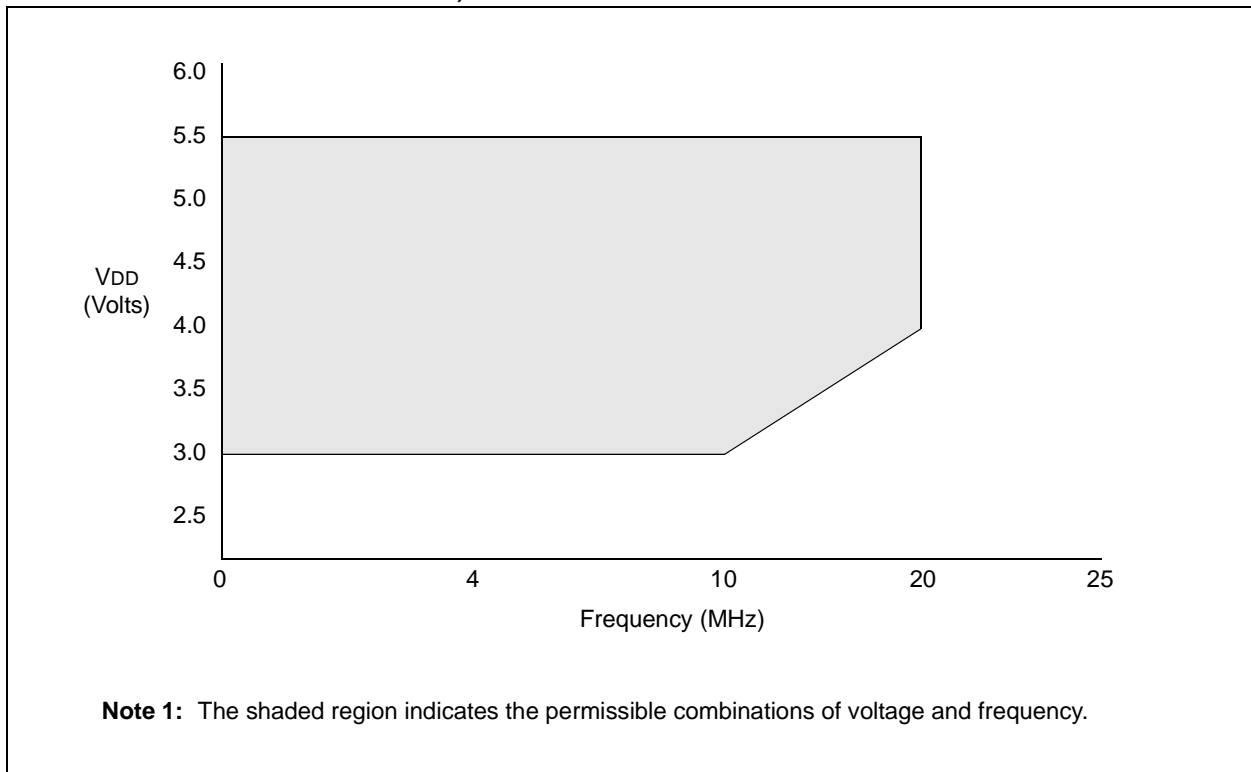


FIGURE 15-2: PIC16LC717/770/771 VOLTAGE-FREQUENCY GRAPH, $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$



**FIGURE 15-3: PIC16LC717/770/771 VOLTAGE-FREQUENCY GRAPH,
-40°C ≤ TA ≤ 0°C, +70°C ≤ TA ≤ +85°C**



PIC16C717/770/771

15.1 DC Characteristics: PIC16C717/770/771 (Commercial, Industrial, Extended) PIC16LC717/770/771 (Commercial, Industrial, Extended)

| PIC16LC717/770/771 | | Standard Operating Conditions (unless otherwise stated) Operating temperature 0°C ≤ TA ≤ +70°C for commercial -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended | | | | | |
|--------------------|------|---|------|------|-----|-------|---|
| PIC16C717/770/771 | | Standard Operating Conditions (unless otherwise stated) Operating temperature 0°C ≤ TA ≤ +70°C for commercial -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended | | | | | |
| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| D001 | VDD | Supply Voltage | 2.5 | — | 5.5 | V | |
| D001 | VDD | Supply Voltage | 4.0 | — | 5.5 | V | |
| D002* | VDR | RAM Data Retention Voltage ⁽¹⁾ | — | 1.5 | — | 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 |
| 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 |
| 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 |

* 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.

PIC16C717/770/771

15.1 DC Characteristics: PIC16C717/770/771 (Commercial, Industrial, Extended) PIC16LC717/770/771 (Commercial, Industrial, Extended) (Continued)

| Param. No. | | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|--------------------|--|-----|---|------|------|-----|-------|---|
| PIC16LC717/770/771 | | | Standard Operating Conditions (unless otherwise stated) Operating temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended | | | | | |
| PIC16C717/770/771 | | | Standard Operating Conditions (unless otherwise stated) Operating temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended | | | | | |
| | | IDD | Supply Current⁽²⁾ | | | | | |
| D010D D010E | | | PIC16LC7XX | | 1.0 | 2.0 | mA | FOSC = 10 MHz, VDD = 3V, -40°C to 85°C FOSC = 10 MHz, VDD = 3V, -40°C to 125°C |
| D010G | | | | 0.36 | 1.0 | | mA | FOSC = 4 MHz, VDD = 2.5V, -40°C to 125°C |
| D010K | | | | 11 | 45 | | μA | FOSC = 32 kHz, VDD = 2.5V, -40°C to 125°C |
| | | IDD | Supply Current⁽²⁾ | | | | | |
| D010 D010A | | | PIC16C7XX | | 4.0 | 7.5 | mA | FOSC = 20 MHz, VDD = 5.5V, -40°C to 85°C FOSC = 20 MHz, VDD = 5.5V, -40°C to 125°C |
| D010B D010C | | | | | 2.5 | 5.0 | mA | FOSC = 20 MHz, VDD = 4V, -40°C to 85°C FOSC = 20 MHz, VDD = 4V, -40°C to 125°C |
| D010F | | | | | 0.55 | 1.5 | mA | FOSC = 4 MHz, VDD = 4V, -40°C to 125°C |
| D010H D010J | | | | | 30 | 80 | μA | FOSC = 32 kHz, VDD = 4V, -40°C to 85°C FOSC = 32 kHz, VDD = 4V, -40°C to 125°C |

* 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

MCLR = VDD; WDT enabled/disabled as specified.

PIC16C717/770/771

| PIC16LC717/770/771 | | Standard Operating Conditions (unless otherwise stated) Operating temperature 0°C ≤ TA ≤ +70°C for commercial -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended | | | | | |
|---------------------------|-----|--|-----|------|-----|-------|----------------------------|
| PIC16C717/770/771 | | Standard Operating Conditions (unless otherwise stated) Operating temperature 0°C ≤ TA ≤ +70°C for commercial -40°C ≤ TA ≤ +85°C for industrial -40°C ≤ TA ≤ +125°C for extended | | | | | |
| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| | IPD | Power-down Current⁽³⁾ | | | | | |
| D020D D020E | | PIC16LC7XX | | 0.3 | 2.0 | μA | VDD = 3V, -40°C to 85°C |
| | | | | | 5.0 | | VDD = 3V, -40°C to 125°C |
| D020F D020G | | | | 0.1 | 1.5 | μA | VDD = 2.5V, -40°C to 85°C |
| | | | | | 3.0 | | VDD = 2.5V, -40°C to 125°C |
| D020 D020A | | PIC16C7XX | | 1.4 | 4.0 | μA | VDD = 5.5V, -40°C to 85°C |
| | | | | | 8.0 | | VDD = 5.5V, -40°C to 125°C |
| D020B D020C | | | | 1.0 | 3.5 | μA | VDD = 4V, -40°C to 85°C |
| | | | | | 6.0 | | VDD = 4V, -40°C to 125°C |

* 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

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.

PIC16C717/770/771

15.1 DC Characteristics: PIC16C717/770/771 (Commercial, Industrial, Extended) PIC16LC717/770/771 (Commercial, Industrial, Extended) (Continued)

| PIC16LC717/770/771 | | Standard Operating Conditions (unless otherwise stated) Operating temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended | | | | | |
|---------------------------------|--------|--|-----|------|-----|---------------|---|
| PIC16C717/770/771 | | Standard Operating Conditions (unless otherwise stated) Operating temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended | | | | | |
| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| Base plus Module current | | | | | | | |
| D021A | IWDT | Watchdog Timer | | 2 | 10 | μA | $V_{DD} = 3\text{V}$, -40°C to 125°C |
| D021 | IWDT | Watchdog Timer | | 5 | 20 | μA | $V_{DD} = 4\text{V}$, -40°C to 125°C |
| D021 | IWDT | Watchdog Timer | | 5 | 20 | μA | $V_{DD} = 4\text{V}$, -40°C to 125°C |
| D025 | IT1OSC | Timer1 Oscillator | | 3 | 9 | μA | $V_{DD} = 3\text{V}$, -40°C to 125°C |
| D025 | IT1OSC | Timer1 Oscillator | | 4 | 12 | μA | $V_{DD} = 4\text{V}$, -40°C to 125°C |
| D025 | IT1OSC | Timer1 Oscillator | | 4 | 12 | μA | $V_{DD} = 4\text{V}$, -40°C to 125°C |
| D026* | IAD | ADC Converter | | 300 | | μA | $V_{DD} = 5.5\text{V}$, A/D on, not converting |
| D026* | IAD | ADC Converter | | 300 | | μA | $V_{DD} = 5.5\text{V}$, A/D on, not converting |
| D027 | IPLVD | Programmable Low Voltage Detect | | 55 | 125 | μA | $V_{DD} = 4\text{V}$, -40°C to 85°C |
| D027A | | | | | 150 | | $V_{DD} = 4\text{V}$, -40°C to 125°C |
| D027 | IPLVD | Programmable Low Voltage Detect | | 55 | 125 | μA | $V_{DD} = 4\text{V}$, -40°C to 85°C |
| D027A | | | | | 150 | | $V_{DD} = 4\text{V}$, -40°C to 125°C |
| D028 | IPBOR | Programmable Brown-out Reset | | 55 | 125 | μA | $V_{DD} = 5\text{V}$, -40°C to 85°C |
| D028A | | | | | 150 | | $V_{DD} = 5\text{V}$, -40°C to 125°C |
| D028 | IPBOR | Programmable Brown-out Reset | | 55 | 125 | μA | $V_{DD} = 5\text{V}$, -40°C to 85°C |
| D028A | | | | | 150 | | $V_{DD} = 5\text{V}$, -40°C to 125°C |
| D029 | IVRH | Voltage reference High | | 200 | 750 | μA | $V_{DD} = 5\text{V}$, -40°C to 85°C |
| D029A | | | | | 1.0 | mA | $V_{DD} = 5\text{V}$, -40°C to 125°C |
| D029 | IVRH | Voltage reference High | | 200 | 750 | μA | $V_{DD} = 5\text{V}$, -40°C to 85°C |
| D029A | | | | | 1.0 | mA | $V_{DD} = 5\text{V}$, -40°C to 125°C |
| D030 | IVRL | Voltage reference Low | | 200 | 750 | μA | $V_{DD} = 4\text{V}$, -40°C to 85°C |
| D030A | | | | | 1.0 | mA | $V_{DD} = 4\text{V}$, -40°C to 125°C |
| D030 | IVRL | Voltage reference Low | | 200 | 750 | μA | $V_{DD} = 4\text{V}$, -40°C to 85°C |
| D030A | | | | | 1.0 | mA | $V_{DD} = 4\text{V}$, -40°C to 125°C |

* 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.

PIC16C717/770/771

15.2 DC Characteristics: PIC16C717/770/771 & PIC16LC717/770/771 (Commercial, Industrial, Extended)

| Standard Operating Conditions (unless otherwise stated) Operating temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended Operating voltage V_{DD} range as described in Section 15.1 and Section 15.2. | | | | | | | |
|--|----------------------|---|---|-----------------------|---|---|--|
| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
| DC CHARACTERISTICS | | | | | | | |
| D030 D030A D031 D032 D033 | V_{IL} | Input Low Voltage I/O ports with TTL buffer with Schmitt Trigger buffer $\overline{\text{MCLR}}$ OSC1 (in XT, HS, LP and EC) | V_{SS} V_{SS} V_{SS} V_{SS} V_{SS} | — — — — — | $0.15V_{DD}$ 0.8V $0.2V_{DD}$ $0.2V_{DD}$ $0.3V_{DD}$ | V V V V V | For entire V_{DD} range $4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$ For entire V_{DD} range |
| D040 D040A D041 D042 D042A | V_{IH} | Input High Voltage I/O ports with TTL buffer with Schmitt Trigger buffer $\overline{\text{MCLR}}$ OSC1 (XT, HS, LP and EC) | 2.0 ($0.25V_{DD}$ + 0.8V) $0.8V_{DD}$ $0.8V_{DD}$ $0.7V_{DD}$ | — — — — — | V_{DD} V_{DD} V_{DD} V_{DD} V_{DD} | V V V V V | $4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$ For entire V_{DD} range For entire V_{DD} range |
| D070 | IPURB | PORTB weak pull-up current per pin | 50 | 250 | 400 | μA | $V_{DD} = 5\text{V}$, $V_{PIN} = V_{SS}$ |
| D060 D060A D061 D063 | I_{IL} I_{IL} | Input Leakage Current ^(1,2) I/O ports (with digital functions) I/O ports (with analog functions) $\overline{\text{RA5}}/\overline{\text{MCLR}}/V_{PP}$ OSC1 | — — — — | — — — — | ± 1 ± 100 ± 5 ± 5 | μA nA μA μA | $V_{SS} \leq V_{PIN} \leq V_{DD}$, Pin at hi-impedance $V_{SS} \leq V_{PIN} \leq V_{DD}$, Pin at hi-impedance $V_{SS} \leq V_{PIN} \leq V_{DD}$ $V_{SS} \leq V_{PIN} \leq V_{DD}$, XT, HS, LP and EC osc configuration |
| D080 | V_{OL} | Output Low Voltage I/O ports | — | — | 0.6 | V | $I_{OL} = 8.5\text{ mA}$, $V_{DD} = 4.5\text{V}$ |
| D090 | V_{OH} | Output High Voltage I/O ports ⁽²⁾ | $V_{DD} - 0.7$ | — | — | V | $I_{OH} = -3.0\text{ mA}$, $V_{DD} = 4.5\text{V}$ |
| D150* | VOD | Open Drain High Voltage | — | — | 10.5 | V | RA4 pin |
| Capacitive Loading Specs on Output Pins* | | | | | | | |
| D100 | COS C2 | OSC2 pin | — | — | 15 | pF | In XT, HS and LP modes when external clock is used to drive OSC1. |
| D101 D102 | C_{IO} CB | All I/O pins and OSC2 (in RC mode) SCL, SDA in I ² C mode | — — | — — | 50 400 | pF pF | |
| | CVRH | VRH pin | — | — | 200 | pF | VRH output enabled |
| | CVRL | VRL pin | — | — | 200 | pF | VRL 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 $\overline{\text{MCLR}}$ 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.

15.3 AC Characteristics: PIC16C717/770/771 & PIC16LC717/770/771 (Commercial, Industrial, Extended)

15.3.1 TIMING PARAMETER SYMBOLOGY

The timing parameter symbols have been created using one of the following formats:

- | | | |
|-------------|-----------|--|
| 1. TppS2ppS | 3. TCC:ST | (I ² C specifications only) |
| 2. TppS | 4. Ts | (I ² C specifications only) |

| | | | |
|----------|-----------|---|------|
| T | | | |
| F | Frequency | T | Time |

Lowercase letters (pp) and their meanings:

| | | | |
|-----------|-------------------|-----|------------------------------------|
| pp | | | |
| cc | CCP1 | osc | OSC1 |
| ck | CLKOUT | rd | \overline{RD} |
| cs | \overline{CS} | rw | \overline{RD} or \overline{WR} |
| di | SDI | sc | SCK |
| do | SDO | ss | \overline{SS} |
| dt | Data in | t0 | T0CKI |
| io | I/O port | t1 | T1CKI |
| mc | \overline{MCLR} | wr | \overline{WR} |

Uppercase letters and their meanings:

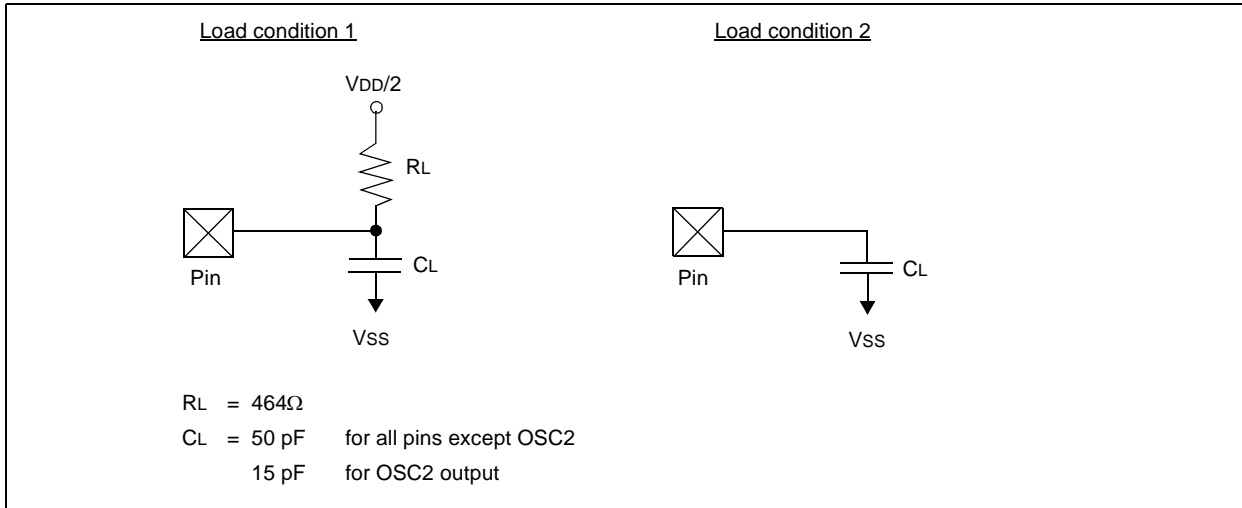
| | | | |
|--|------------------------|---|--------------|
| S | | | |
| F | Fall | P | Period |
| H | High | R | Rise |
| I | Invalid (Hi-impedance) | V | Valid |
| L | Low | Z | Hi-impedance |
| I²C (I ² C specifications only) | | | |
| AA | output access | | |
| BUF | Bus free | | |
| High | High | | |
| Low | Low | | |

TCC:ST (I²C specifications only)

| | | | |
|-----------|-----------------|-----|----------------|
| CC | | | |
| HD | Hold | SU | Setup |
| ST | | | |
| DAT | DATA input hold | STO | STOP condition |
| STA | START condition | | |

PIC16C717/770/771

FIGURE 15-4: LOAD CONDITIONS



15.3.2 TIMING DIAGRAMS AND SPECIFICATIONS

FIGURE 15-5: CLKOUT AND I/O TIMING

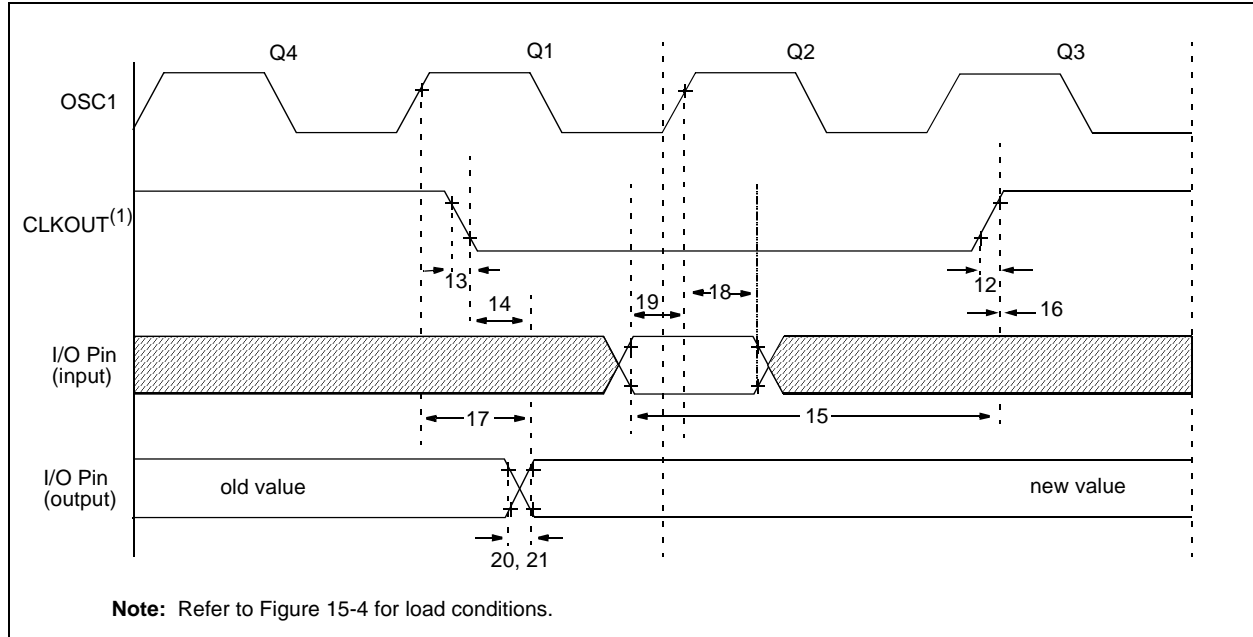


TABLE 15-1: CLKOUT AND I/O TIMING REQUIREMENTS

| Param. No. | Sym | Characteristic | Min | Typ† | Max | Unit | Conditions | |
|------------|----------|---|--------------------------|------|-------------------------|------|---------------|--|
| 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 | — | — | 0.5T _{CY} + 20 | ns | Note 1 | |
| 15* | TioV2ckH | Port in valid before CLKOUT ↑ | 0.25T _{CY} + 25 | — | — | ns | Note 1 | |
| 16* | TckH2iol | Port in hold after CLKOUT ↑ | 0 | — | — | ns | Note 1 | |
| 17* | TosH2ioV | OSC1↑ (Q1 cycle) to Port out valid | — | 50 | 150 | ns | | |
| 18* | TosH2iol | OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time) | PIC16C717/770/771 | 100 | — | — | ns | |
| | | | PIC16LC717/770/771 | 200 | — | — | ns | |
| 19* | TioV2osH | Port input valid to OSC1↑ (I/O in setup time) | 0 | — | — | ns | | |
| 20* | TioR | Port output rise time | PIC16C717/770/771 | — | 10 | 25 | ns | |
| | | | PIC16LC717/770/771 | — | — | 60 | ns | |
| 21* | TioF | Port output fall time | PIC16C717/770/771 | — | 10 | 25 | ns | |
| | | | PIC16LC717/770/771 | — | — | 60 | ns | |
| 22††* | Tinp | INT pin high or low time | T _{CY} | — | — | ns | | |
| 23††* | Trbp | RB<7:0> change INT high or low time | T _{CY} | — | — | 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 ER or INTRC w/CLKOUT mode where CLKOUT output is 4 x T_{OSC}.

PIC16C717/770/771

FIGURE 15-6: EXTERNAL CLOCK TIMING

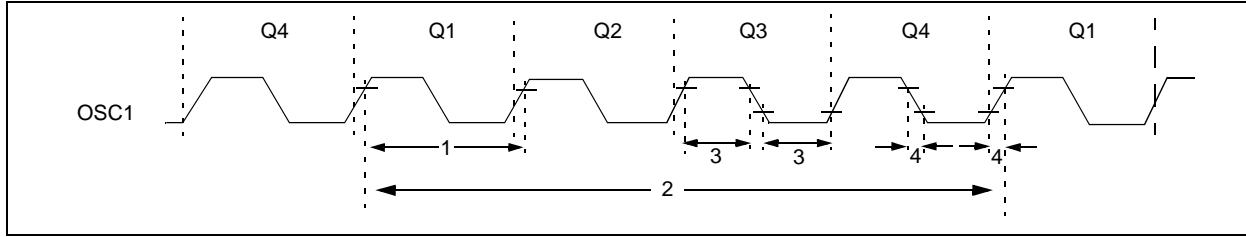


TABLE 15-2: EXTERNAL CLOCK TIMING REQUIREMENTS

| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|-----------|---------------|--|------|------|---------|-------|--------------|
| 1A | Fosc | External CLKIN Frequency (Note 1) | DC | — | 4 | MHz | XT mode |
| | | | DC | — | 20 | MHz | EC mode |
| | | | DC | — | 20 | MHz | HS mode |
| | | | DC | — | 200 | kHz | LP mode |
| | | Oscillator Frequency (Note 1) | 0.1* | — | 4 | MHz | XT mode |
| | | | 4* | — | 20 | MHz | HS mode |
| | | | 5* | — | 200 | kHz | LP mode |
| 1 | Tosc | External CLKIN Period (Note 1) | 250 | — | — | ns | XT mode |
| | | | 50 | — | — | ns | EC mode |
| | | | 50 | — | — | ns | HS mode |
| | | | 5 | — | — | μs | LP mode |
| | | Oscillator Period (Note 1) | 250 | — | 10,000* | ns | XT mode |
| | | | 50 | — | 250* | ns | HS mode |
| | | | 5 | — | — | μs | LP 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 mode |
| | | | 2.5 | — | — | μs | LP mode |
| | | | 15 | — | — | ns | HS mode |
| | | | — | — | — | ns | EC mode |
| 4* | TosR, TosF | External Clock in (OSC1) Rise or Fall Time | — | — | 25 | ns | XT mode |
| | | | — | — | 50 | ns | LP mode |
| | | | — | — | 15 | ns | HS mode |
| | | | — | — | — | ns | EC mode |

* 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 "Max. Frequency" values with a square wave applied to the OSC1/CLKIN pin.

When an external clock input is used, the "Min." frequency (or Max. Tcy) limit is "DC" (no clock) for all devices.

PIC16C717/770/771

TABLE 15-3: CALIBRATED INTERNAL RC FREQUENCIES - PIC16C717/770/771 AND PIC16LC717/770/771

| AC Characteristics | | Standard Operating Conditions (unless otherwise specified) | | | | | |
|--------------------|------|---|------|---------------------|------|-------|------------------------|
| | | Operating Temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended | | | | | |
| | | Operating Voltage V_{DD} range is described in Section and Section | | | | | |
| Parameter No. | Sym | Characteristic | Min | Typ ^{(1)*} | Max | Units | Conditions |
| | FIRC | Internal Calibrated RC Frequency | 3.65 | 4.00 | 4.28 | MHz | $V_{DD} = 5.0\text{V}$ |
| | | Internal RC Frequency* | 3.55 | 4.00 | 4.31 | MHz | $V_{DD} = 2.5\text{V}$ |

* 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

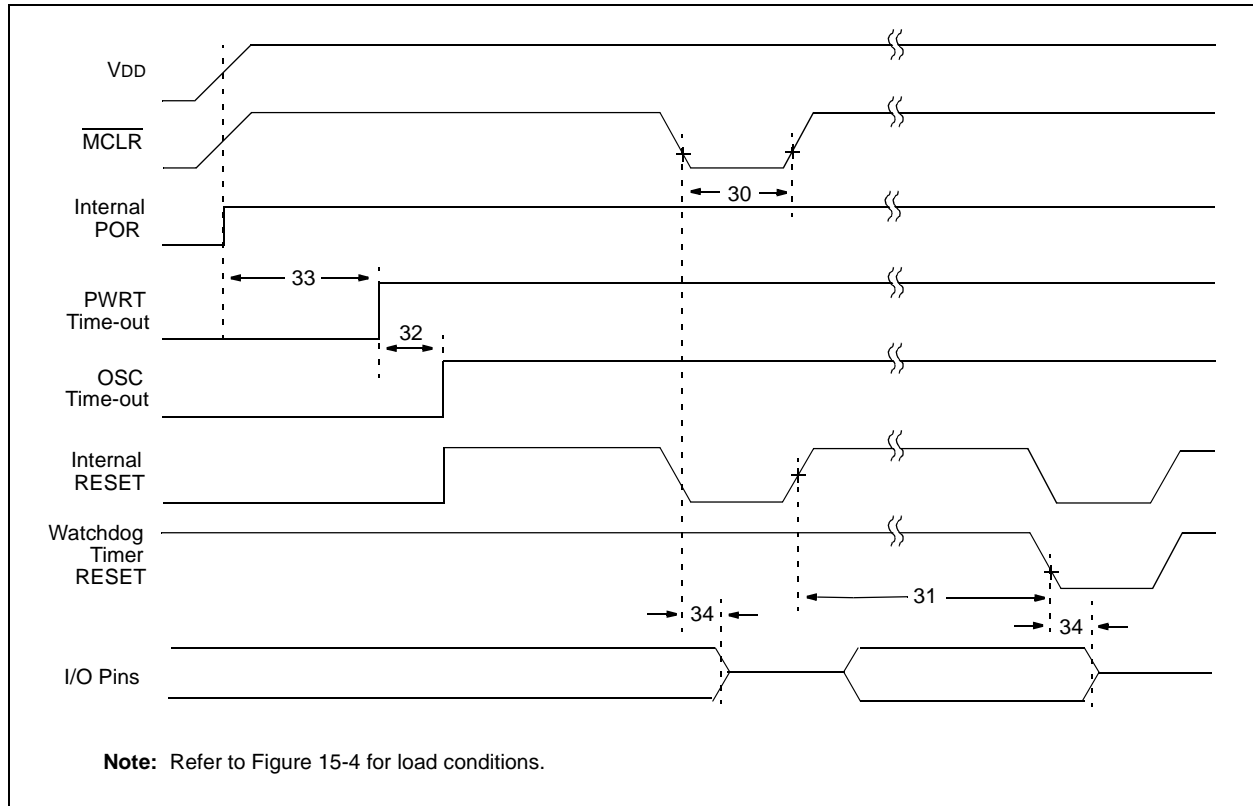
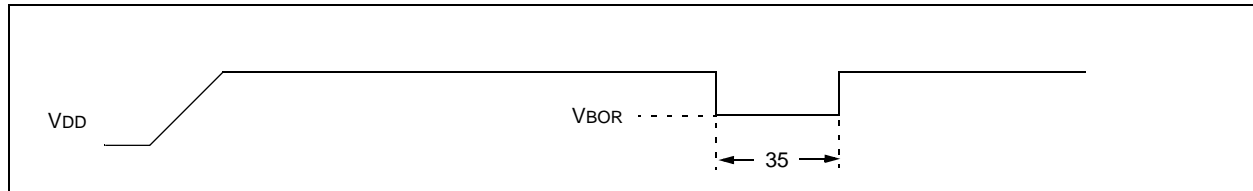


FIGURE 15-8: BROWN-OUT RESET TIMING



PIC16C717/770/771

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 | 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 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

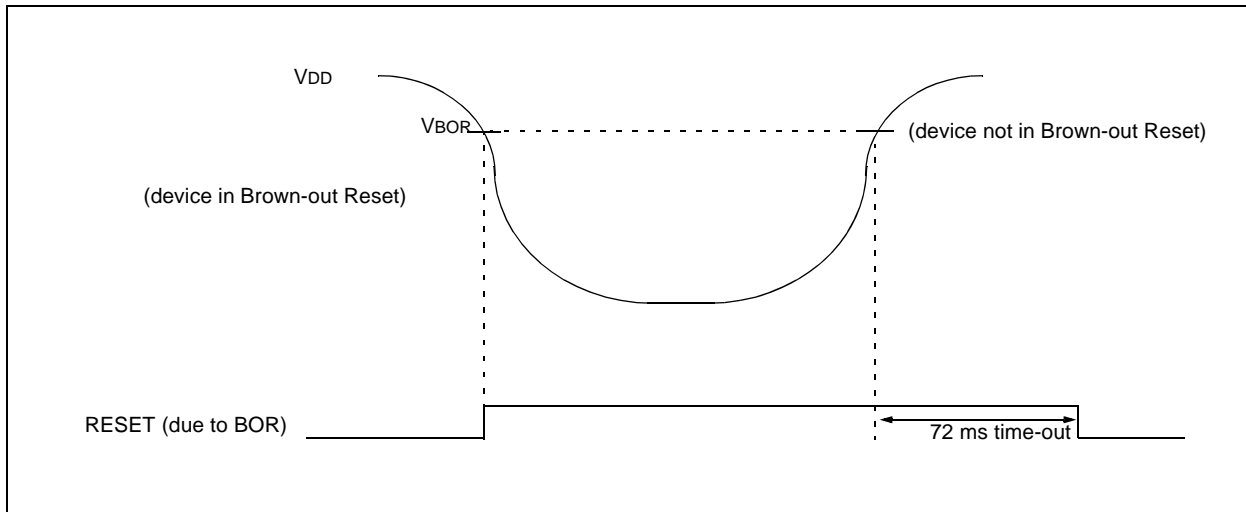
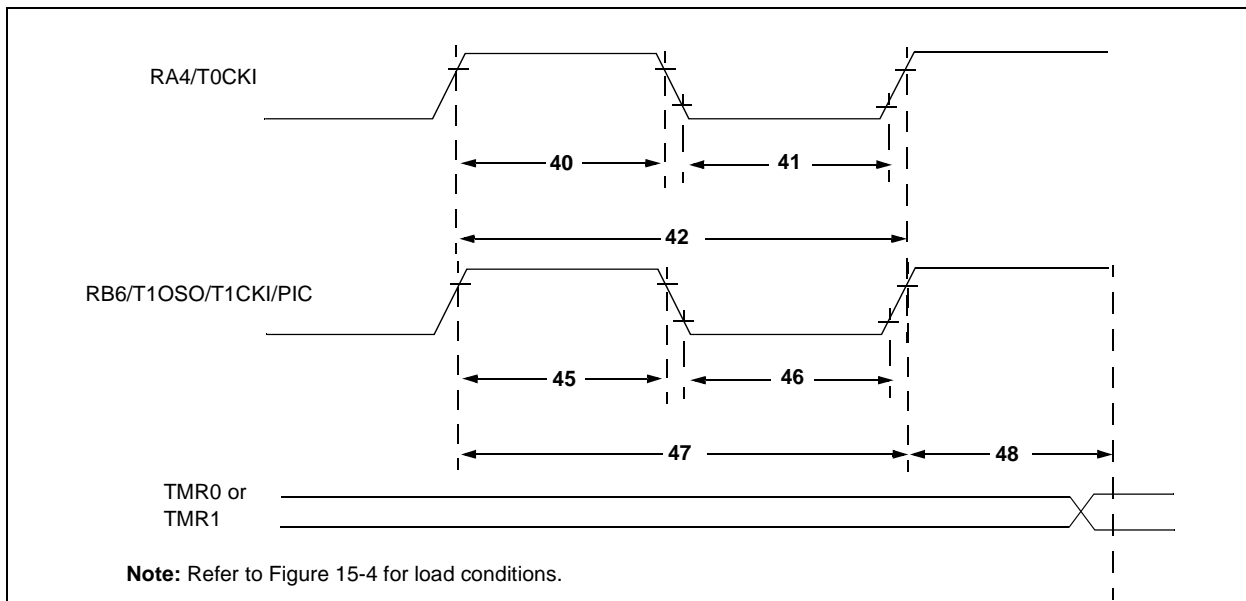


FIGURE 15-10: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS



Note: Refer to Figure 15-4 for load conditions.

PIC16C717/770/771

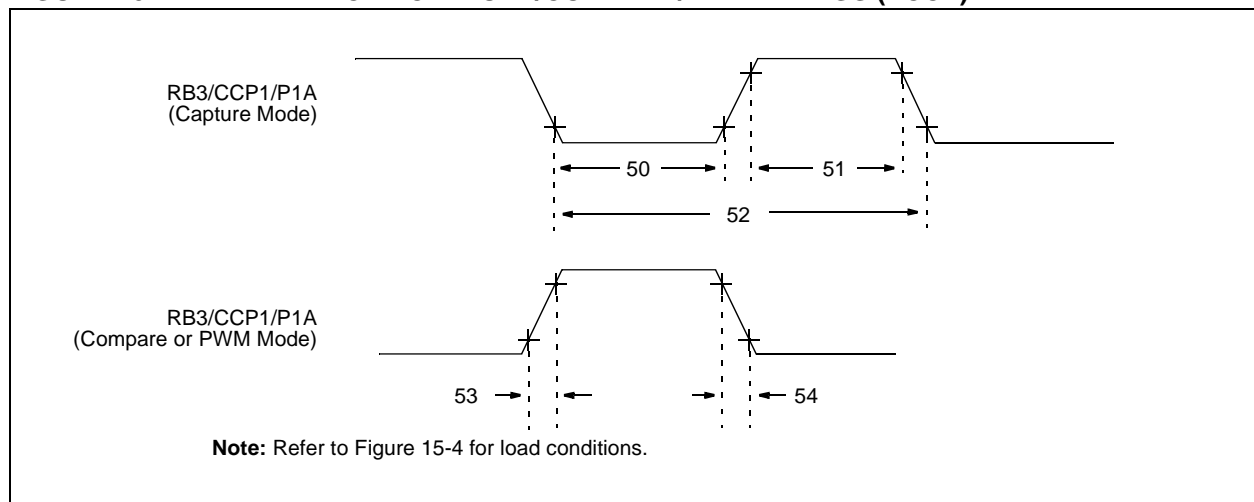
TABLE 15-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS

| Param. No. | Sym | Characteristic | | Min | Typ† | Max | Units | Conditions | |
|------------|-----------|---|--------------------------------|--|--|------------|-------|-------------------------------------|---------------------------------|
| 40* | Tt0H | T0CKI High Pulse Width | No Prescaler | $0.5T_{CY} + 20$ | — | — | ns | Must also meet parameter 42 | |
| | | | With Prescaler | 10 | — | — | ns | | |
| 41* | Tt0L | T0CKI Low Pulse Width | No Prescaler | $0.5T_{CY} + 20$ | — | — | ns | Must also meet parameter 42 | |
| | | | With Prescaler | 10 | — | — | ns | | |
| 42* | Tt0P | T0CKI Period | No Prescaler | $T_{CY} + 40$ | — | — | ns | N = prescale value (2, 4, ..., 256) | |
| | | | With Prescaler | Greater of: 20 or $\frac{T_{CY} + 40}{N}$ | — | — | ns | | |
| 45* | Tt1H | T1CKI High Time | Synchronous, Prescaler = 1 | $0.5T_{CY} + 20$ | — | — | ns | Must also meet parameter 47 | |
| | | | Synchronous, Prescaler = 2,4,8 | PIC16C717/770/771 PIC16LC717/770/771 | 15 25 | — | — | | ns ns |
| | | | Asynchronous | PIC16C717/770/771 | 30 | — | — | | ns |
| | | | | PIC16LC717/770/771 | 50 | — | — | | ns |
| 46* | Tt1L | T1CKI Low Time | Synchronous, Prescaler = 1 | $0.5T_{CY} + 20$ | — | — | ns | Must also meet parameter 47 | |
| | | | Synchronous, Prescaler = 2,4,8 | PIC16C717/770/771 PIC16LC717/770/771 | 15 25 | — | — | | ns ns |
| | | | Asynchronous | PIC16C717/770/771 | 30 | — | — | | ns |
| | | | | PIC16LC717/770/771 | 50 | — | — | | ns |
| 47* | Tt1P | T1CKI input period | Synchronous | PIC16C717/770/771 | Greater of: 30 OR $\frac{T_{CY} + 40}{N}$ | — | — | ns | N = prescale value (1, 2, 4, 8) |
| | | | | PIC16LC717/770/771 | Greater of: 50 OR $\frac{T_{CY} + 40}{N}$ | — | — | ns | N = prescale value (1, 2, 4, 8) |
| | | | Asynchronous | PIC16C717/770/771 | 60 | — | — | ns | |
| | | | | PIC16LC717/770/771 | 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 | | $2T_{osc}$ | — | $7T_{osc}$ | — | | |

* 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 (ECCP)



PIC16C717/770/771

TABLE 15-6: ENHANCED CAPTURE/COMPARE/PWM REQUIREMENTS (ECCP)

| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions | | |
|------------|------|-----------------------|-----------------------|--------------------|-----|-------|---------------------------------|--|----|
| 50* | TccL | CCP1 input low time | No Prescaler | 0.5Tcy + 20 | — | — | ns | | |
| | | | With Prescaler | PIC16C717/770/771 | 10 | — | — | | ns |
| | | | | PIC16LC717/770/771 | 20 | — | — | | ns |
| 51* | TccH | CCP1 input high time | No Prescaler | 0.5Tcy + 20 | — | — | ns | | |
| | | | With Prescaler | PIC16C717/770/771 | 10 | — | — | | ns |
| | | | | PIC16LC717/770/771 | 20 | — | — | | ns |
| 52* | TccP | CCP1 input period | $\frac{3Tcy + 40}{N}$ | — | — | ns | N = prescale value (1, 4 or 16) | | |
| 53* | TccR | CCP1 output fall time | PIC16C717/770/771 | — | 10 | 25 | ns | | |
| | | | PIC16LC717/770/771 | — | 25 | 45 | ns | | |
| 54* | TccF | CCP1 output fall time | PIC16C717/770/771 | — | 10 | 25 | ns | | |
| | | | PIC16LC717/770/771 | — | 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.

15.4 Analog Peripherals Characteristics: PIC16C717/770/771 & PIC16LC717/770/771 (Commercial, Industrial, Extended)

15.4.1 BANDGAP MODULE

FIGURE 15-12: BANDGAP START-UP TIME

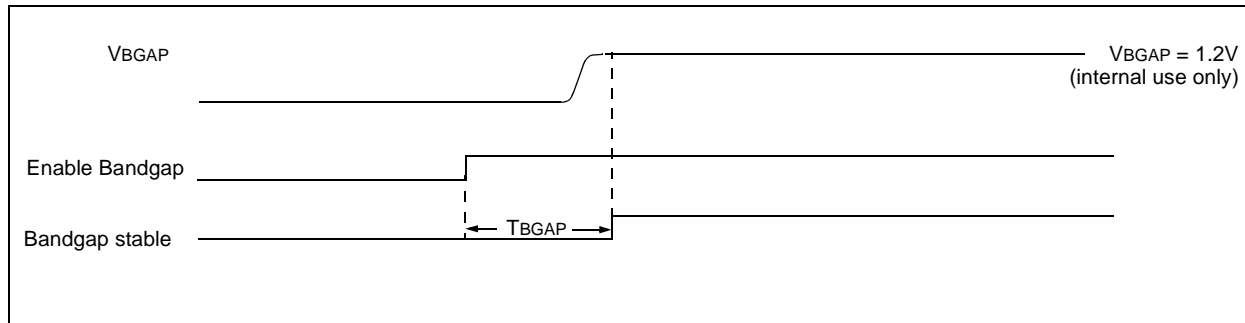


TABLE 15-7: BANDGAP START-UP TIME

| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|------------|-------|-----------------------|-----|------|-----|-------|--|
| 36* | TBGAP | Bandgap start-up time | — | 19 | 33 | 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.

PIC16C717/770/771

15.4.2 LOW VOLTAGE DETECT MODULE (LVD)

FIGURE 15-13: LOW VOLTAGE DETECT CHARACTERISTICS

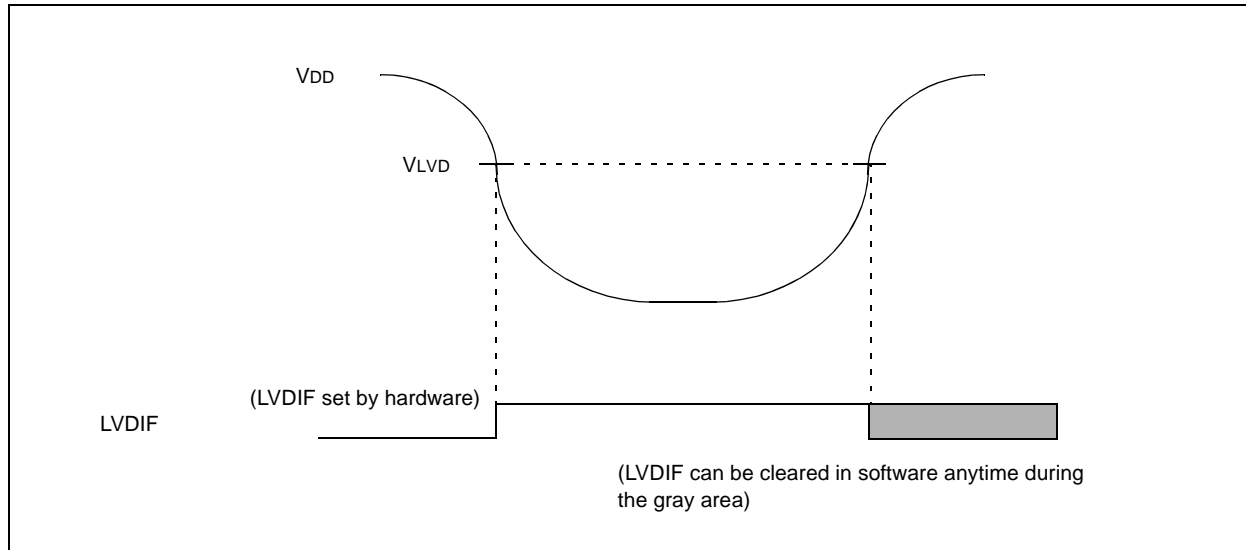


TABLE 15-8: ELECTRICAL CHARACTERISTICS: LVD

| DC CHARACTERISTICS | | Standard Operating Conditions (unless otherwise stated) | | | | | | Conditions |
|--------------------|----------------|---|-----|------|------|-------|------------|------------|
| | | Operating temperature $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial and $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial | | | | | | |
| | | Operating voltage VDD range as described in DC Characteristics Section 15.1. | | | | | | |
| 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 | | |

* These parameters are characterized but not tested.

Note 1: Production tested at Tamb = 25°C. Specifications over temperature limits ensured by characterization.

15.4.3 PROGRAMMABLE BROWN-OUT RESET MODULE (PBOR)

TABLE 15-9: DC CHARACTERISTICS: PBOR

| Standard Operating Conditions (unless otherwise stated) Operating temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended Operating voltage V_{DD} range as described in DC Characteristics Section 15.1. | | | | | | | | |
|--|----------------|----------------|------------------|-----|------|------|-------|------------|
| Param. No. | Characteristic | | Symbol | Min | Typ | Max | Units | Conditions |
| D005 | BOR Voltage | BORV<1:0> = 11 | V _{BOR} | 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 | | |

15.4.4 V_{REF} MODULE

TABLE 15-10: DC CHARACTERISTICS: V_{REF}

| Standard Operating Conditions (unless otherwise stated) Operating temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended Operating voltage V_{DD} range as described in DC Characteristics Section 15.1. | | | | | | | | |
|--|---------------------------------|-------------------------|-----|-------|-----|-------|--|------------------------|
| Param. No. | Symbol | Characteristic | Min | Typ† | Max | Units | Conditions | |
| D400 | VRL | Output Voltage | 2.0 | 2.048 | 2.1 | V | $V_{DD} \geq 2.7\text{V}$, $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ | |
| | VRH | | 4.0 | 4.096 | 4.2 | V | $V_{DD} \geq 4.5\text{V}$, $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ | |
| D400A | VRL | Output Voltage | 1.9 | 2.048 | 2.2 | V | $V_{DD} \geq 2.7\text{V}$, $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | |
| | VRH | | 4.0 | 4.096 | 4.3 | V | $V_{DD} \geq 4.5\text{V}$, $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ | |
| D404* | IVREFSO | External Load Source | — | — | 5 | mA | | |
| D405* | IVREFSI | External Load Sink | — | — | -5 | mA | | |
| * | CL | External Capacitor Load | — | — | 200 | pF | | |
| D406* | $\Delta V_{out}/\Delta I_{out}$ | VRH Load Regulation | — | 0.6 | 1 | mV/mA | $V_{DD} \geq 5\text{V}$ | ISOURCE = 0 mA to 5 mA |
| | | | — | 1 | 4 | | | ISINK = 0 mA to 5 mA |
| | | VRL Load Regulation | — | 0.6 | 1 | | $V_{DD} \geq 3\text{V}$ | ISOURCE = 0 mA to 5 mA |
| | | | — | 2 | 4 | | | ISINK = 0 mA to 5 mA |

* 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.

PIC16C717/770/771

15.4.5 A/D CONVERTER MODULE

TABLE 15-11: 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 | — | — | ±2 | LSb | VREF+ = AVDD = 4.096V, VREF- = AVSS = 0V, VREF- ≤ VAIN ≤ VREF+ |
| A04 | EDL | Differential error | — | — | +2 -1 | LSb | No missing codes to 12 bits VREF+ = AVDD = 4.096V, VREF- = AVSS = 0V, VREF- ≤ VAIN ≤ VREF+ |
| A06 | EOFF | Offset error | — | — | ±2 | LSb | VREF+ = AVDD = 4.096V, VREF- = AVSS = 0V, VREF- ≤ VAIN ≤ VREF+ |
| A07 | EGN | Gain Error | — | — | ±2 | LSb | VREF+ = AVDD = 4.096V, VREF- = AVSS = 0V, VREF- ≤ VAIN ≤ VREF+ |
| A10 | — | Monotonicity | — | Note 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.

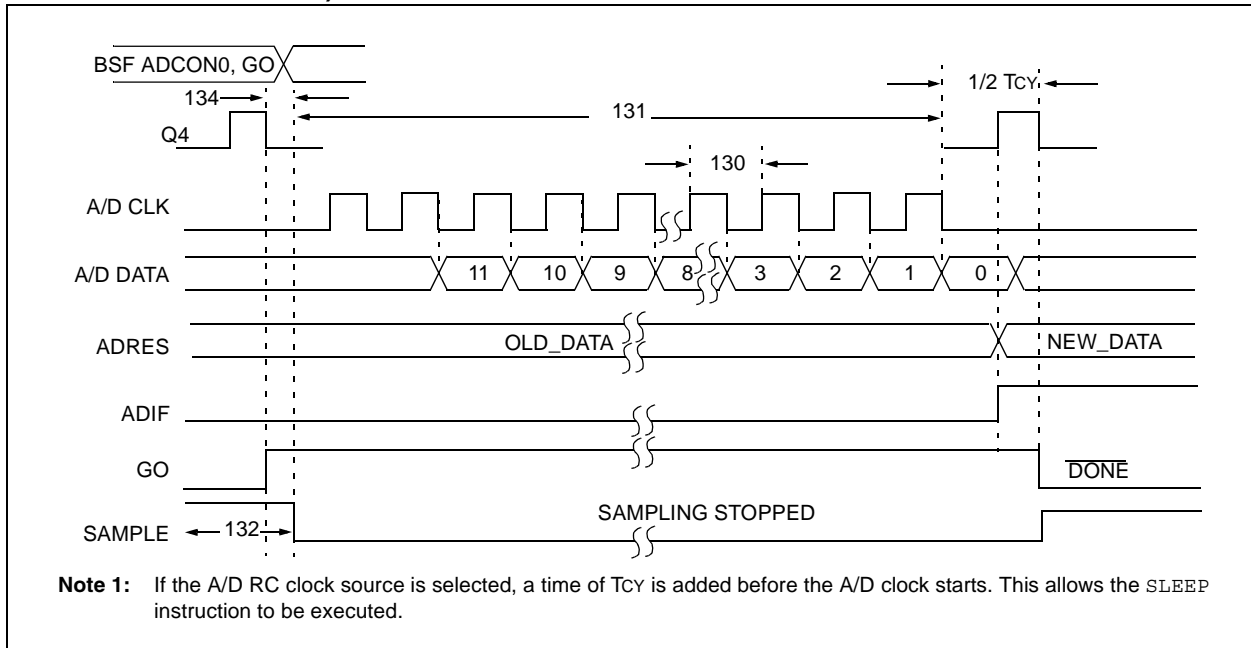
† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

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 15-14: PIC16C770/771 AND PIC16LC770/771 A/D CONVERSION TIMING (NORMAL MODE)



PIC16C717/770/771

TABLE 15-12: PIC16C770/771 AND PIC16LC770/771 A/D CONVERSION REQUIREMENTS (NORMAL MODE)

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|------|--|---------------|--------|-----|-------|---|
| 130*(3) | TAD | A/D clock period | 1.6 | — | — | μs | Tosc based, VREF ≥ 2.5V |
| | | | 3.0 | — | — | μs | Tosc based, VREF full range |
| | | | 3.0 | 6.0 | 9.0 | μs | ADCS<1:0> = 11 (A/D RC mode) 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 | |
| 132* | TACQ | Acquisition Time | Note 2 | 11.5 | — | 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). |
| | | | 5* | — | — | μs | |
| 134* | TGO | Q4 to A/D clock start | — | Tosc/2 | — | — | |

* 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.

3: These numbers multiplied by 8 if VRH or VRL is selected as A/D reference.

PIC16C717/770/771

FIGURE 15-15: PIC16C770/771 AND PIC16LC770/771 A/D CONVERSION TIMING (SLEEP MODE)

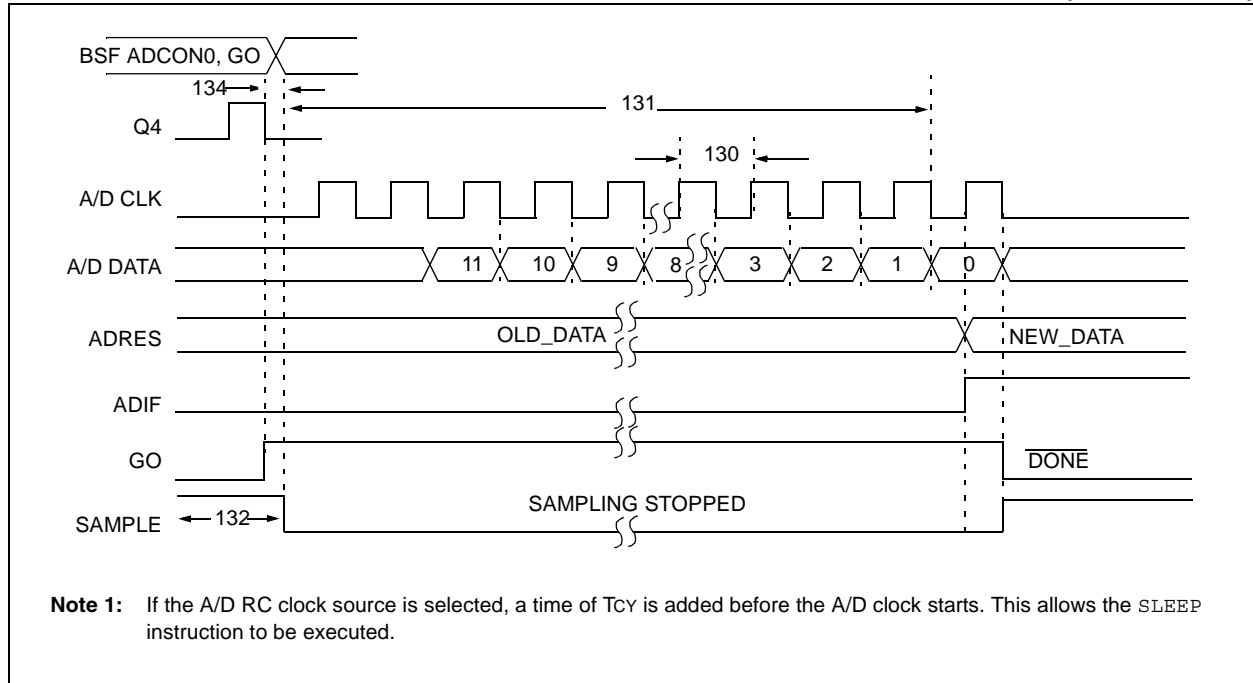


TABLE 15-13: PIC16C770/771 AND PIC16LC770/771 A/D CONVERSION REQUIREMENT (SLEEP MODE)

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|--------------------|------|--|-------------------------|----------------------|------------|--------------------------------|---|
| 130 ⁽³⁾ | TAD | A/D Internal RC oscillator period | 3.0 2.0 | 6.0 4.0 | 9.0 6.0 | μs μs | ADCS<1:0> = 11 (RC mode) At $V_{DD} = 3.0\text{V}$ At $V_{DD} = 5.0\text{V}$ |
| 131* | TCNV | Conversion time (not including acquisition time) (Note 1) | — | 13TAD | — | — | |
| 132* | TACQ | Acquisition Time | (Note 2) 5* | 11.5 — | — — | μs 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 | — | $T_{OSC}/2 + T_{CY}$ | — | — | 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 <code>SLEEP</code> 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.

3: These numbers multiplied by 8 if V_{RH} or V_{RL} is selected as A/D reference.

PIC16C717/770/771

TABLE 15-14: PIC16C717 AND PIC16LC717 A/D CONVERTER CHARACTERISTICS:

| Param. No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|------------|-------|--|-------|--------|-----------|-------|--|
| A01 | NR | Resolution | — | — | 10 bits | bit | Min. resolution for A/D is 4.1 mV, VREF+ = AVDD = 4.096V, VREF- = AVSS = 0V, VREF- ≤ VAIN ≤ VREF+ |
| A03 | EIL | Integral error | — | — | ±1 | LSb | VREF+ = AVDD = 4.096V, VREF- = AVSS = 0V, VREF- ≤ VAIN ≤ VREF+ |
| A04 | EDL | Differential error | — | — | ±1 | LSb | No missing codes to 10 bits VREF+ = AVDD = 4.096V, VREF- = AVSS = 0V, VREF- ≤ VAIN ≤ VREF+ |
| A06 | EOFF | Offset error | — | — | ±2 | LSb | VREF+ = AVDD = 4.096V, VREF- = AVSS = 0V, VREF- ≤ VAIN ≤ VREF+ |
| A07 | EGN | Gain Error | — | — | ±1 | LSb | VREF+ = AVDD = 4.096V, VREF- = AVSS = 0V, VREF- ≤ VAIN ≤ VREF+ |
| A10 | — | Monotonicity | — | Note 3 | — | — | AVSS ≤ VAIN ≤ VREF+ |
| A20* | VREF | Reference voltage (VREF+ - VREF-) | 4.096 | — | VDD +0.3V | V | Absolute minimum electrical spec to ensure 10-bit accuracy. |
| A21* | VREF+ | Reference V High (AVDD or VREF+) | VREF- | — | AVDD | V | Min. resolution for A/D is 4.1 mV |
| A22* | VREF- | Reference V Low (AVSS or VREF-) | AVSS | — | VREF+ | V | Min. resolution for A/D is 4.1 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 leakage current. The power-down current spec includes any such leakage from the A/D module.

2: VREF 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 15-16: PIC16C717 A/D CONVERSION TIMING (NORMAL MODE)

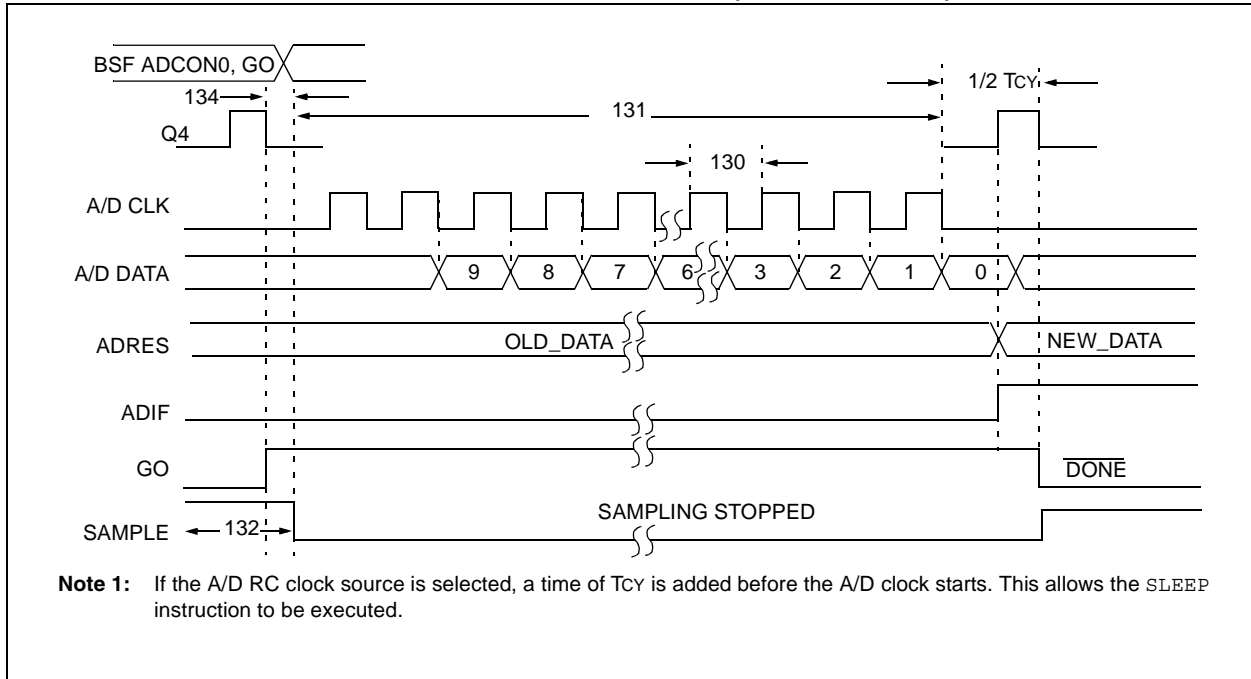


TABLE 15-15: PIC16C717 AND PIC16LC717 A/D CONVERSION REQUIREMENT (NORMAL MODE)

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|------|---|-----------------|-------------|-----|---------------|---|
| 130*(3) | TAD | A/D clock period | 1.6 | — | — | μs | Tosc based, $V_{REF} \geq 2.5\text{V}$ |
| | | | 3.0 | — | — | μs | Tosc based, V_{REF} full range |
| | | | 3.0 | 6.0 | 9.0 | μs | ADCS<1:0> = 11 (A/D RC mode) At $V_{DD} = 2.5\text{V}$ |
| | | | 2.0 | 4.0 | 6.0 | μs | At $V_{DD} = 5.0\text{V}$ |
| 131* | TCNV | Conversion time (not including acquisition time) (Note 1) | — | $11TAD$ | — | TAD | |
| 132* | TACQ | Acquisition Time | (Note 2) | 11.5 | — | 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). |
| | | | 5* | — | — | μs | |
| 134* | TGO | Q4 to A/D clock start | — | $T_{OSC}/2$ | — | — | |

* 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.

3: These numbers multiplied by 8 if VRH or VRL is selected as A/D reference.

PIC16C717/770/771

FIGURE 15-17: PIC16C717 A/D CONVERSION TIMING (SLEEP MODE)

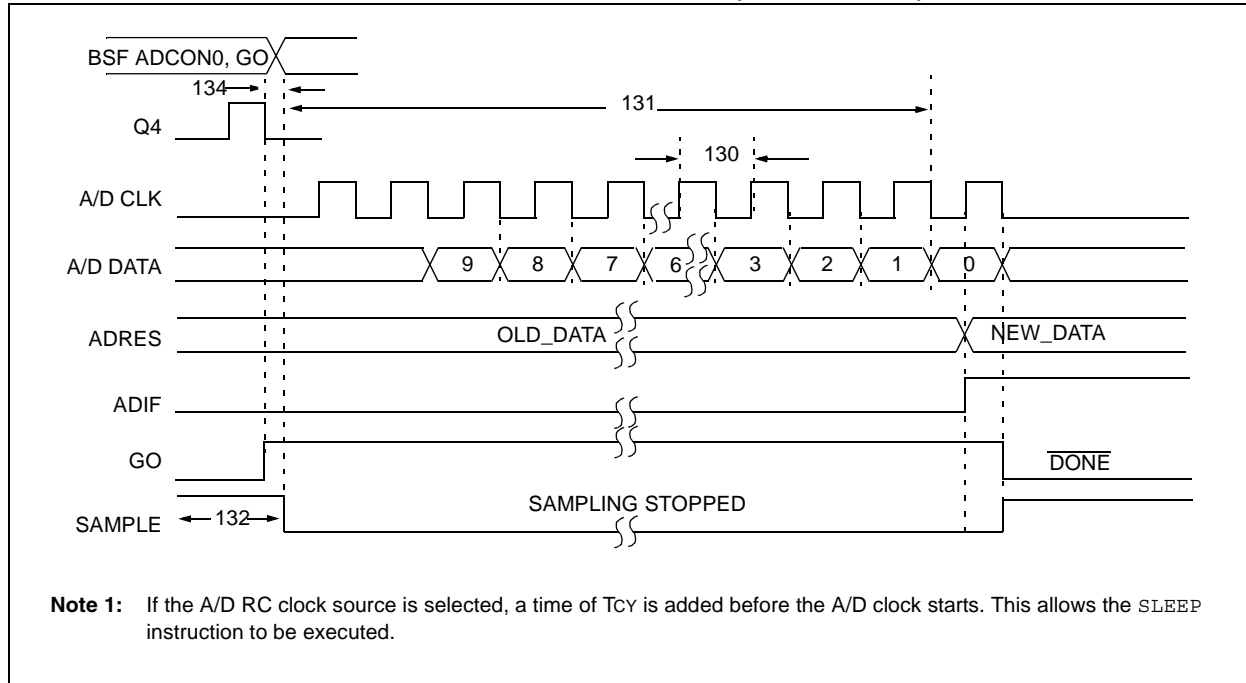


TABLE 15-16: PIC16C717 AND PIC16LC717 A/D CONVERSION REQUIREMENT (SLEEP MODE)

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|------|--|-------------------|----------------------|-----|---------------|---|
| 130*(3) | TAD | A/D clock period | 3.0 | 6.0 | 9.0 | μs | ADCS<1:0> = 11 (A/D RC mode) At $V_{DD} = 3.0\text{V}$ At $V_{DD} = 5.0\text{V}$ |
| | | | 2.0 | 4.0 | 6.0 | μs | |
| 131* | TCNV | Conversion time (not including acquisition time) (Note 1) | — | $11T_{AD}$ | — | — | |
| 132* | TACQ | Acquisition Time | (Note 2) | 11.5 | — | 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). |
| | | | 5* | — | — | μs | |
| 134* | TGO | Q4 to A/D clock start | — | $T_{OSC}/2 + T_{CY}$ | — | — | If the A/D RC clock source is selected, a time of T_{CY} is added before the A/D clock starts. This allows the <code>SLEEP</code> 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.

3: These numbers multiplied by 8 if V_{RH} or V_{RL} is selected as A/D reference.

15.5 Master SSP SPI Mode Timing Waveforms and Requirements

FIGURE 15-18: SPI MASTER MODE TIMING (CKE = 0)

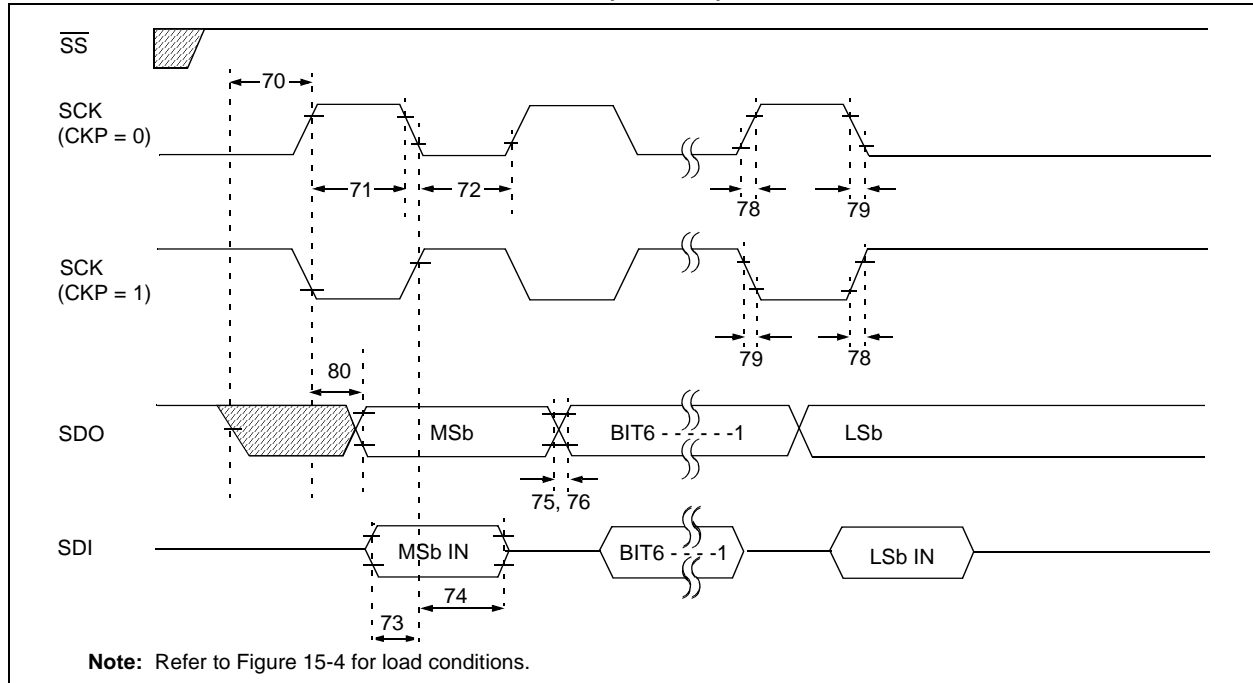


TABLE 15-17: SPI MODE REQUIREMENTS (MASTER MODE, CKE = 0)

| Param. No. | Symbol | Characteristic | Min | Typ† | Max | Units | Conditions |
|------------|--------------------|---|-------------------------|--------------------------|-----|-------|---------------|
| 70* | TssL2scH, TssL2scL | $\overline{SS}\downarrow$ to SCK \downarrow or SCK \uparrow input | T _{CY} | — | — | ns | |
| 71* | TscH | SCK input high time (Slave mode) | Continuous | 1.25T _{CY} + 30 | — | — | ns |
| 71A* | | | Single Byte | 40 | — | — | ns |
| 72* | TscL | SCK input low time (Slave mode) | Continuous | 1.25T _{CY} + 30 | — | — | ns |
| 72A* | | | Single Byte | 40 | — | — | ns |
| 73* | TdiV2scH, TdiV2scL | Setup time of SDI data input to SCK edge | 100 | — | — | ns | |
| 73A* | Tb2B | Last clock edge of Byte1 to the 1st clock edge of Byte2 | 1.5T _{CY} + 40 | — | — | ns | Note 1 |
| 74* | Tsch2diL, TscL2diL | Hold time of SDI data input to SCK edge | 100 | — | — | ns | |
| 75* | TdoR | SDO data output rise time | PIC16CXXX | — | 10 | 25 | ns |
| | | | PIC16LCXXX | — | 20 | 45 | ns |
| 76* | TdoF | SDO data output fall time | — | 10 | 25 | ns | |
| 78* | TscR | SCK output rise time (Master mode) | PIC16CXXX | — | 10 | 25 | ns |
| | | | PIC16LCXXX | — | 20 | 45 | ns |
| 79* | TscF | SCK output fall time (Master mode) | — | 10 | 25 | ns | |
| 80* | Tsch2doV, TscL2doV | SDO data output valid after SCK edge | PIC16CXXX | — | — | 50 | ns |
| | | | PIC16LCXXX | — | — | 100 | 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.

Note 1: Specification 73A is only required if specifications 71A and 72A are used.

PIC16C717/770/771

FIGURE 15-19: SPI MASTER MODE TIMING (CKE = 1)

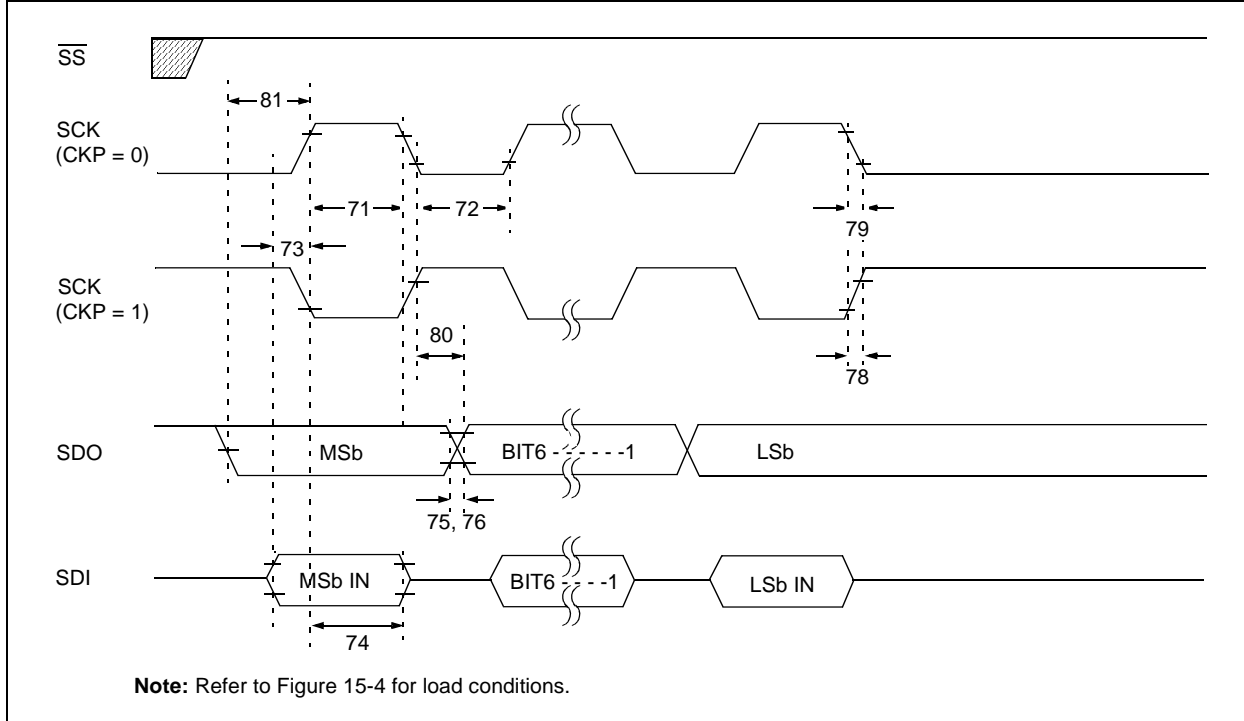


TABLE 15-18: SPI MODE REQUIREMENTS (MASTER MODE, CKE = 1)

| Param. No. | Symbol | Characteristic | Min | Typ† | Max | Units | Conditions |
|------------|--------------------|---|--------------------------|------|-----|-------|---------------|
| 71* | Tsch | SCK input high time | 1.25T _{CY} + 30 | — | — | ns | |
| 71A* | | (Slave mode) | | | | | |
| | | Continuous | 1.25T _{CY} + 30 | — | — | ns | |
| | | Single Byte | 40 | — | — | ns | Note 1 |
| 72* | TscL | SCK input low time | 1.25T _{CY} + 30 | — | — | ns | |
| 72A* | | (Slave mode) | | | | | |
| | | Continuous | 1.25T _{CY} + 30 | — | — | ns | |
| | | Single Byte | 40 | — | — | ns | Note 1 |
| 73* | TdiV2scH, TdiV2scL | Setup time of SDI data input to SCK edge | 100 | — | — | ns | |
| 73A* | TB2B | Last clock edge of Byte1 to the 1st clock edge of Byte2 | 1.5T _{CY} + 40 | — | — | ns | Note 1 |
| 74* | Tsch2diL, TscL2diL | Hold time of SDI data input to SCK edge | 100 | — | — | ns | |
| 75* | TdoR | SDO data output rise time | — | 10 | 25 | ns | |
| | | PIC16CXXX | — | 10 | 25 | ns | |
| | | PIC16LCXXX | — | 20 | 45 | ns | |
| 76* | TdoF | SDO data output fall time | — | 10 | 25 | ns | |
| 78* | TscR | SCK output rise time | — | 10 | 25 | ns | |
| | | (Master mode) | | | | | |
| | | PIC16CXXX | — | 10 | 25 | ns | |
| | | PIC16LCXXX | — | 20 | 45 | ns | |
| 79* | TscF | SCK output fall time (Master mode) | — | 10 | 25 | ns | |
| 80* | Tsch2doV, TscL2doV | SDO data output valid after SCK edge | — | — | 50 | ns | |
| | | PIC16CXXX | — | — | 50 | ns | |
| | | PIC16LCXXX | — | — | 100 | ns | |
| 81* | TdoV2scH, TdoV2scL | SDO data output setup to SCK edge | T _{CY} | — | — | 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.

Note 1: Specification 73A is only required if specifications 71A and 72A are used.

FIGURE 15-20: SPI SLAVE MODE TIMING (CKE = 0)

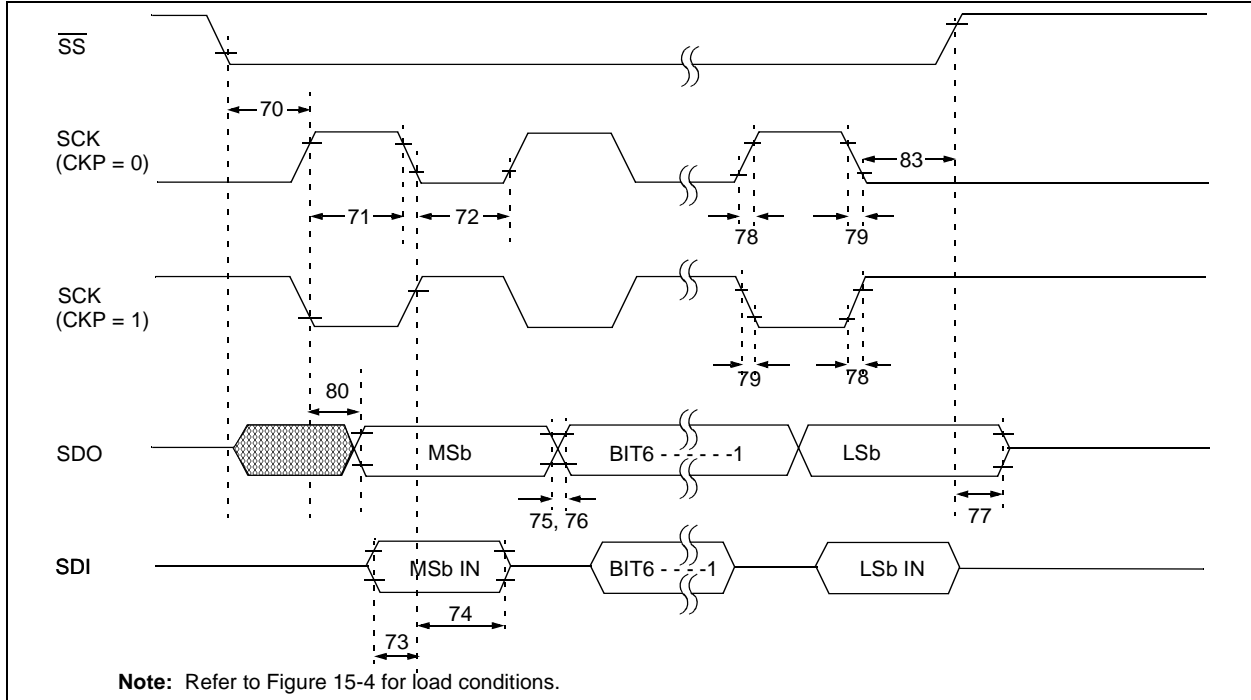


TABLE 15-19: SPI MODE REQUIREMENTS (SLAVE MODE TIMING (CKE = 0))

| Param. No. | Symbol | Characteristic | Min | Typ† | Max | Units | Conditions |
|------------|--------------------|--|-------------|--------------|-----|-------|---------------|
| 70* | Tssl2sch, Tssl2scl | $\overline{SS} \downarrow$ to SCK \downarrow or SCK \uparrow input | Tcy | — | — | ns | |
| 71* | Tsch | SCK input high time (Slave mode) | Continuous | 1.25Tcy + 30 | — | — | ns |
| 71A* | | | Single Byte | 40 | — | — | ns |
| 72* | TscL | SCK input low time (Slave mode) | Continuous | 1.25Tcy + 30 | — | — | ns |
| 72A* | | | Single Byte | 40 | — | — | ns |
| 73* | TdiV2sch, TdiV2scl | Setup time of SDI data input to SCK edge | 100 | — | — | ns | |
| 73A* | Tb2B | Last clock edge of Byte1 to the 1st clock edge of Byte2 | 1.5Tcy + 40 | — | — | ns | Note 1 |
| 74* | Tsch2diL, TscL2diL | Hold time of SDI data input to SCK edge | 100 | — | — | ns | |
| 75* | TdoR | SDO data output rise time | PIC16CXXX | — | 10 | 25 | ns |
| | | | PIC16LCXXX | — | 20 | 45 | ns |
| 76* | TdoF | SDO data output fall time | — | 10 | 25 | ns | |
| 77* | TssH2doZ | $\overline{SS} \uparrow$ to SDO output hi-impedance | 10 | — | 50 | ns | |
| 78* | TscR | SCK output rise time (Master mode) | PIC16CXXX | — | 10 | 25 | ns |
| | | | PIC16LCXXX | — | 20 | 45 | ns |
| 79* | TscF | SCK output fall time (Master mode) | — | 10 | 25 | ns | |
| 80* | Tsch2doV, TscL2doV | SDO data output valid after SCK edge | PIC16CXXX | — | — | 50 | ns |
| | | | PIC16LCXXX | — | — | 100 | ns |
| 83* | Tsch2ssH, TscL2ssH | $\overline{SS} \uparrow$ after SCK edge | 1.5Tcy + 40 | — | — | 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.

Note 1: Specification 73A is only required if specifications 71A and 72A are used.

PIC16C717/770/771

FIGURE 15-21: SPI SLAVE MODE TIMING (CKE = 1)

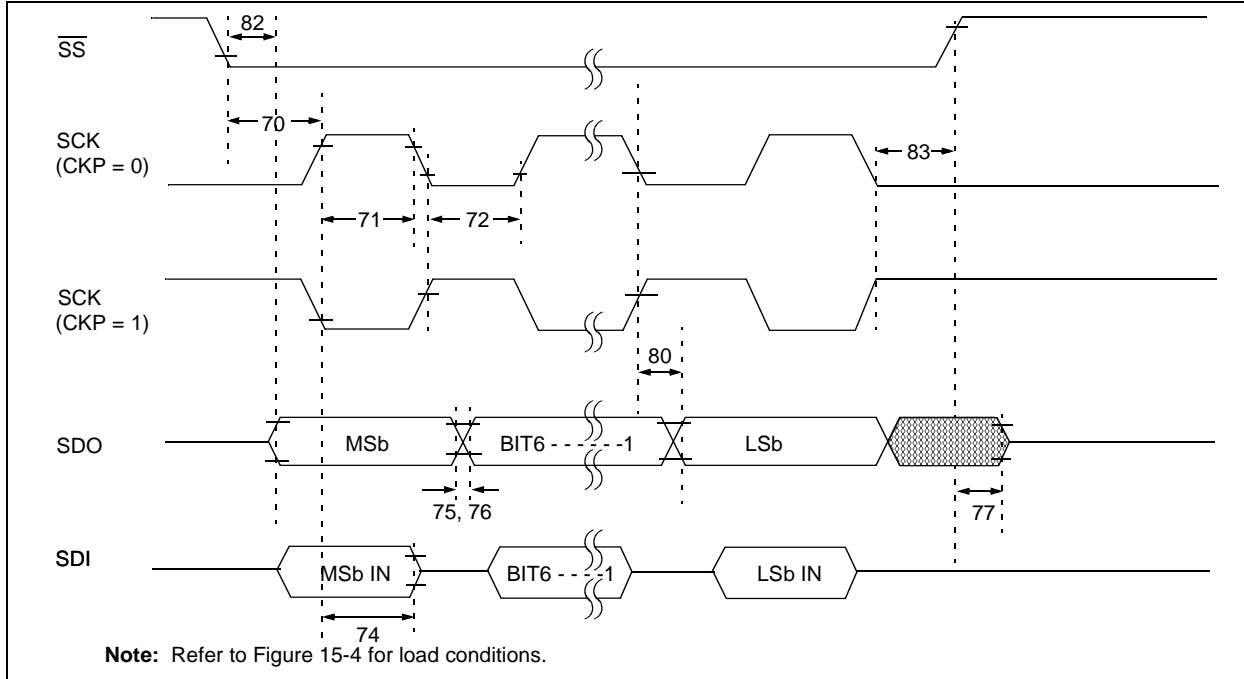


TABLE 15-20: SPI SLAVE MODE REQUIREMENTS (CKE = 1)

| Param. No. | Symbol | Characteristic | Min | Typ† | Max | Units | Conditions |
|------------|--------------------|--|-------------------------|--------------------------|-----|-------|---------------|
| 70* | TssL2scH, TssL2scL | $\overline{SS} \downarrow$ to SCK \downarrow or SCK \uparrow input | T _{CY} | — | — | ns | |
| 71* | TscH | SCK input high time (Slave mode) | Continuous | 1.25T _{CY} + 30 | — | — | ns |
| 71A* | | | Single Byte | 40 | — | — | ns |
| 72* | TscL | SCK input low time (Slave mode) | Continuous | 1.25T _{CY} + 30 | — | — | ns |
| 72A* | | | Single Byte | 40 | — | — | ns |
| 73A* | Tb2B | Last clock edge of Byte1 to the 1st clock edge of Byte2 | 1.5T _{CY} + 40 | — | — | ns | Note 1 |
| 74* | TscH2diL, TscL2diL | Hold time of SDI data input to SCK edge | 100 | — | — | ns | |
| 75* | TdoR | SDO data output rise time | PIC16CXXX | — | 10 | 25 | ns |
| 76* | | | PIC16LCXXX | — | 20 | 45 | ns |
| 76* | TdoF | SDO data output fall time | — | 10 | 25 | ns | |
| 77* | TssH2doZ | $\overline{SS} \uparrow$ to SDO output hi-impedance | 10 | — | 50 | ns | |
| 78* | TscR | SCK output rise time (Master mode) | PIC16CXXX | — | 10 | 25 | ns |
| 79* | | | PIC16LCXXX | — | 20 | 45 | ns |
| 79* | TscF | SCK output fall time (Master mode) | — | 10 | 25 | ns | |
| 80* | TscH2doV, TscL2doV | SDO data output valid after SCK edge | PIC16CXXX | — | — | 50 | ns |
| 82* | | | PIC16LCXXX | — | — | 100 | ns |
| 82* | TssL2doV | SDO data output valid after $\overline{SS} \downarrow$ edge | PIC16CXXX | — | — | 50 | ns |
| 83* | | | PIC16LCXXX | — | — | 100 | ns |
| 83* | TscH2ssH, TscL2ssH | $\overline{SS} \uparrow$ after SCK edge | 1.5T _{CY} + 40 | — | — | 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.

Note 1: Specification 73A is only required if specifications 71A and 72A are used.

15.6 Master SSP I²C Mode Timing Waveforms and Requirements

FIGURE 15-22: MASTER SSP I²C BUS START/STOP BITS TIMING WAVEFORMS

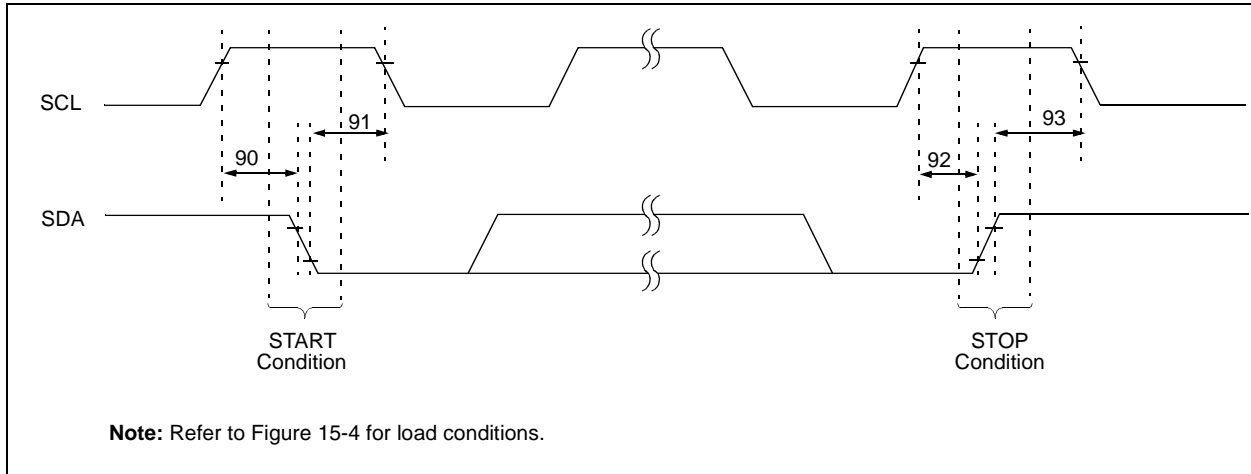


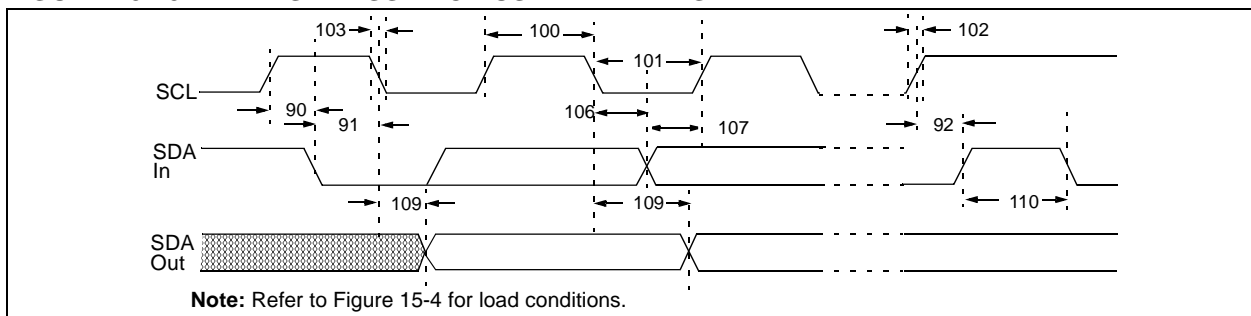
TABLE 15-21: MASTER SSP I²C BUS START/STOP BITS REQUIREMENTS

| Param. No. | Symbol | Characteristic | | Min | Typ | Max | Units | Conditions |
|------------|---------|----------------------------|---------------------------|-----------------------|-----|-----|-------|--|
| 90* | TSU:STA | START condition Setup time | 100 kHz mode | $2(T_{osc})(BRG + 1)$ | — | — | ns | Only relevant for a Repeated START condition |
| | | | 400 kHz mode | $2(T_{osc})(BRG + 1)$ | — | — | | |
| | | | 1 MHz mode ⁽¹⁾ | $2(T_{osc})(BRG + 1)$ | — | — | | |
| 91* | THD:STA | START condition Hold time | 100 kHz mode | $2(T_{osc})(BRG + 1)$ | — | — | ns | After this period the first clock pulse is generated |
| | | | 400 kHz mode | $2(T_{osc})(BRG + 1)$ | — | — | | |
| | | | 1 MHz mode ⁽¹⁾ | $2(T_{osc})(BRG + 1)$ | — | — | | |
| 92* | TSU:STO | STOP condition Setup time | 100 kHz mode | $2(T_{osc})(BRG + 1)$ | — | — | ns | |
| | | | 400 kHz mode | $2(T_{osc})(BRG + 1)$ | — | — | | |
| | | | 1 MHz mode ⁽¹⁾ | $2(T_{osc})(BRG + 1)$ | — | — | | |
| 93* | THD:STO | STOP condition Hold time | 100 kHz mode | $2(T_{osc})(BRG + 1)$ | — | — | ns | |
| | | | 400 kHz mode | $2(T_{osc})(BRG + 1)$ | — | — | | |
| | | | 1 MHz mode ⁽¹⁾ | $2(T_{osc})(BRG + 1)$ | — | — | | |

* These parameters are characterized but not tested. For the value required by the I²C specification, please refer to the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023).

Maximum pin capacitance = 10 pF for all I²C pins.

FIGURE 15-23: MASTER SSP I²C BUS DATA TIMING



PIC16C717/770/771

TABLE 15-22: MASTER SSP I²C BUS DATA REQUIREMENTS

| Param. No. | Symbol | Characteristic | Min | Max | Units | Conditions |
|------------|---------|----------------------------|---------------------------|------------------|-------|------------|
| 100* | THIGH | Clock high time | 100 kHz mode | 2(Tosc)(BRG + 1) | — | ms |
| | | | 400 kHz mode | 2(Tosc)(BRG + 1) | — | ms |
| | | | 1 MHz mode ⁽¹⁾ | 2(Tosc)(BRG + 1) | — | ms |
| 101* | TLOW | Clock low time | 100 kHz mode | 2(Tosc)(BRG + 1) | — | ms |
| | | | 400 kHz mode | 2(Tosc)(BRG + 1) | — | ms |
| | | | 1 MHz mode ⁽¹⁾ | 2(Tosc)(BRG + 1) | — | ms |
| 102* | TR | SDA and SCL rise time | 100 kHz mode | — | 1000 | ns |
| | | | 400 kHz mode | 20 + 0.1Cb | 300 | ns |
| | | | 1 MHz mode ⁽¹⁾ | — | 300 | ns |
| 103* | TF | SDA and SCL fall time | 100 kHz mode | — | 300 | ns |
| | | | 400 kHz mode | 20 + 0.1Cb | 300 | ns |
| | | | 1 MHz mode ⁽¹⁾ | — | 100 | ns |
| 90* | TSU:STA | START condition setup time | 100 kHz mode | 2(Tosc)(BRG + 1) | — | ms |
| | | | 400 kHz mode | 2(Tosc)(BRG + 1) | — | ms |
| | | | 1 MHz mode ⁽¹⁾ | 2(Tosc)(BRG + 1) | — | ms |
| 91* | THD:STA | START condition hold time | 100 kHz mode | 2(Tosc)(BRG + 1) | — | ms |
| | | | 400 kHz mode | 2(Tosc)(BRG + 1) | — | ms |
| | | | 1 MHz mode ⁽¹⁾ | 2(Tosc)(BRG + 1) | — | ms |
| 106* | THD:DAT | Data input hold time | 100 kHz mode | 0 | — | ns |
| | | | 400 kHz mode | 0 | 0.9 | ms |
| | | | 1 MHz mode ⁽¹⁾ | TBD | — | ns |
| 107* | TSU:DAT | Data input setup time | 100 kHz mode | 250 | — | ns |
| | | | 400 kHz mode | 100 | — | ns |
| | | | 1 MHz mode ⁽¹⁾ | TBD | — | ns |
| 92* | TSU:STO | STOP condition setup time | 100 kHz mode | 2(Tosc)(BRG + 1) | — | ms |
| | | | 400 kHz mode | 2(Tosc)(BRG + 1) | — | ms |
| | | | 1 MHz mode ⁽¹⁾ | 2(Tosc)(BRG + 1) | — | ms |
| 109* | TAA | Output valid from clock | 100 kHz mode | — | 3500 | ns |
| | | | 400 kHz mode | — | 1000 | ns |
| | | | 1 MHz mode ⁽¹⁾ | — | — | ns |
| 110 | TBUF | Bus free time | 100 kHz mode | 4.7 ‡ | — | ms |
| | | | 400 kHz mode | 1.3 ‡ | — | ms |
| | | | 1 MHz mode ⁽¹⁾ | TBD‡ | — | ms |
| D102 ‡ | Cb | Bus capacitive loading | — | 400 | pF | |

* These parameters are characterized but not tested. For the value required by the I²C specification, please refer to the PICmicro™ Mid-Range MCU Family Reference Manual (DS33023).

‡ These parameters are for design guidance only and are not tested, nor characterized.

Note 1: Maximum pin capacitance = 10 pF for all I²C pins.

Note 2: A Fast mode I²C bus device can be used in a Standard mode I²C bus system, but (TSU:DAT) ≥ 250 ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line.
[(TR) + (TSU:DAT) = 1000 + 250 = 1250 ns], for 100 kHz mode, before the SCL line is released.

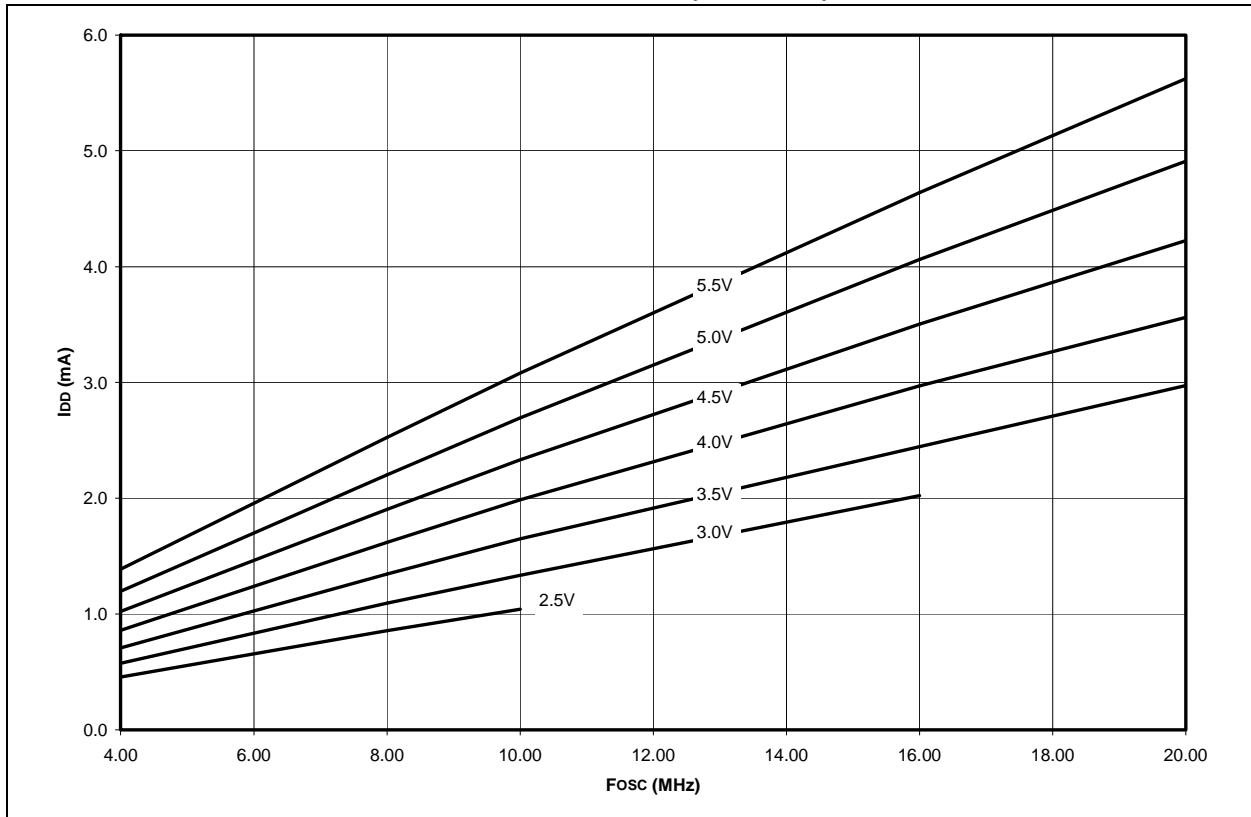
16.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES

The graphs and tables provided following this note are a statistical summary based on a limited number of samples and are provided for informational purposes only. The performance characteristics listed herein are not tested or guaranteed. In some graphs or tables, the data presented may be outside the specified operating range (e.g., outside specified power supply range) and therefore outside the warranted range.

“Typical” represents the mean of the distribution at 25°C. “Maximum” or “minimum” represents (mean + 3σ) or (mean - 3σ) respectively, where σ is a standard deviation, over the whole temperature range.

The Fosc IDD was determined using an external sinusoidal clock source with a peak amplitude ranging from Vss to VDD.

FIGURE 16-1: MAXIMUM IDD VS. Fosc OVER VDD (HS MODE)



PIC16C717/770/771

FIGURE 16-2: TYPICAL I_{DD} VS. F_{osc} OVER V_{DD} (HS MODE)

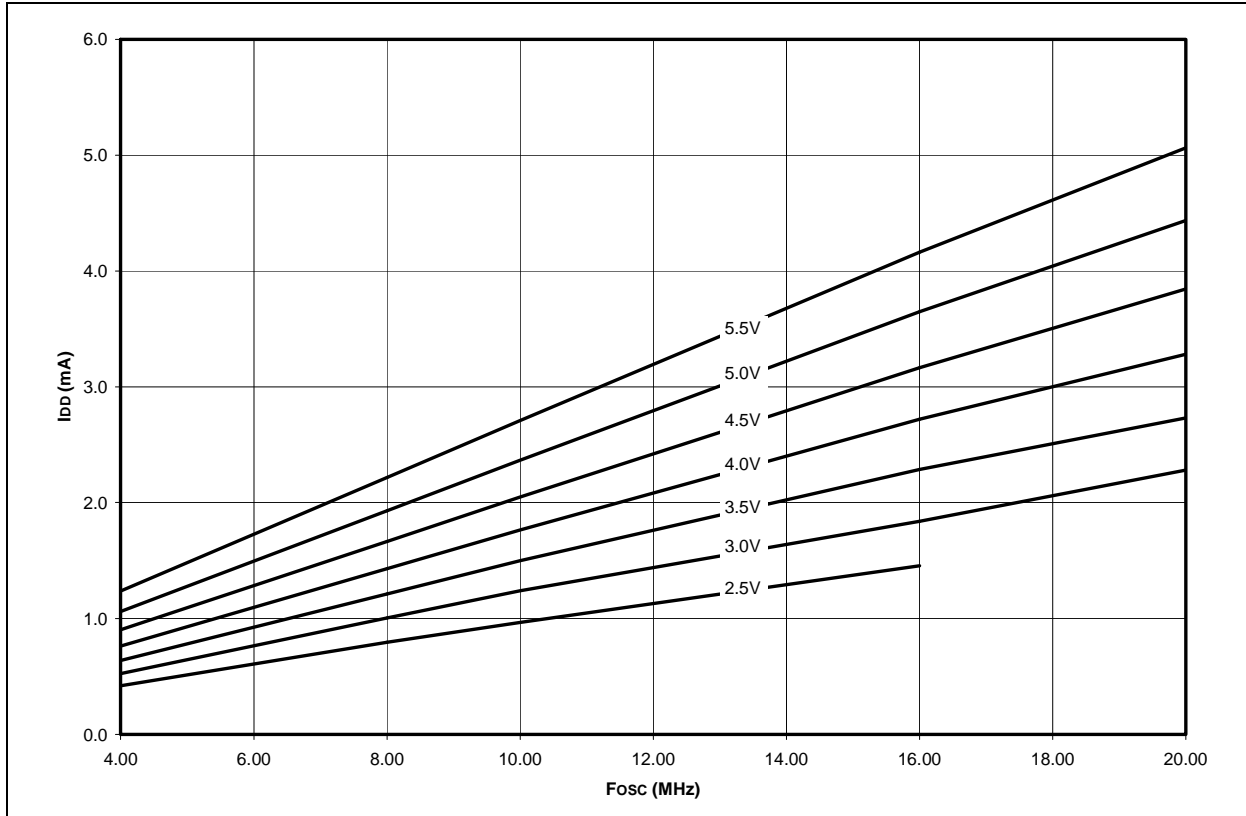


FIGURE 16-3: MAXIMUM I_{DD} VS. F_{osc} OVER V_{DD} (XT MODE)

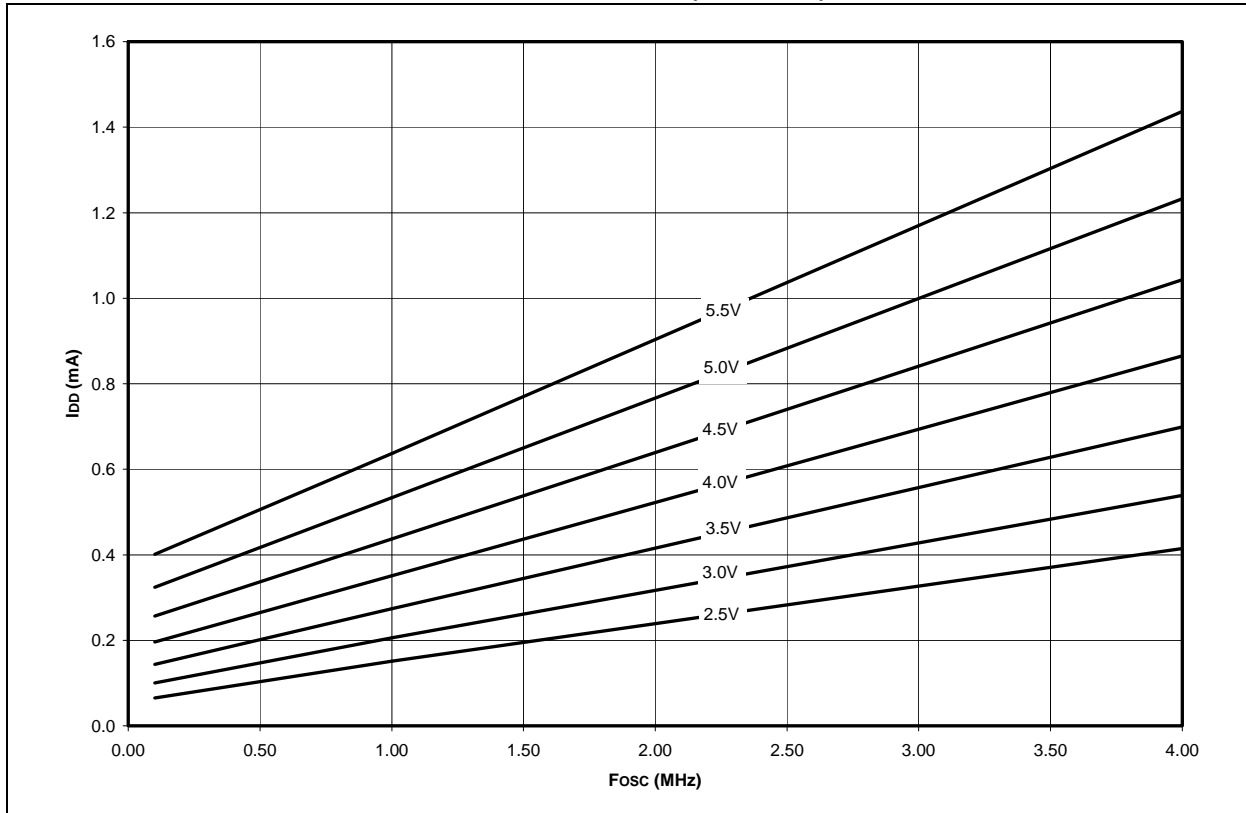


FIGURE 16-4: TYPICAL I_{DD} VS. F_{osc} OVER V_{DD} (XT MODE)

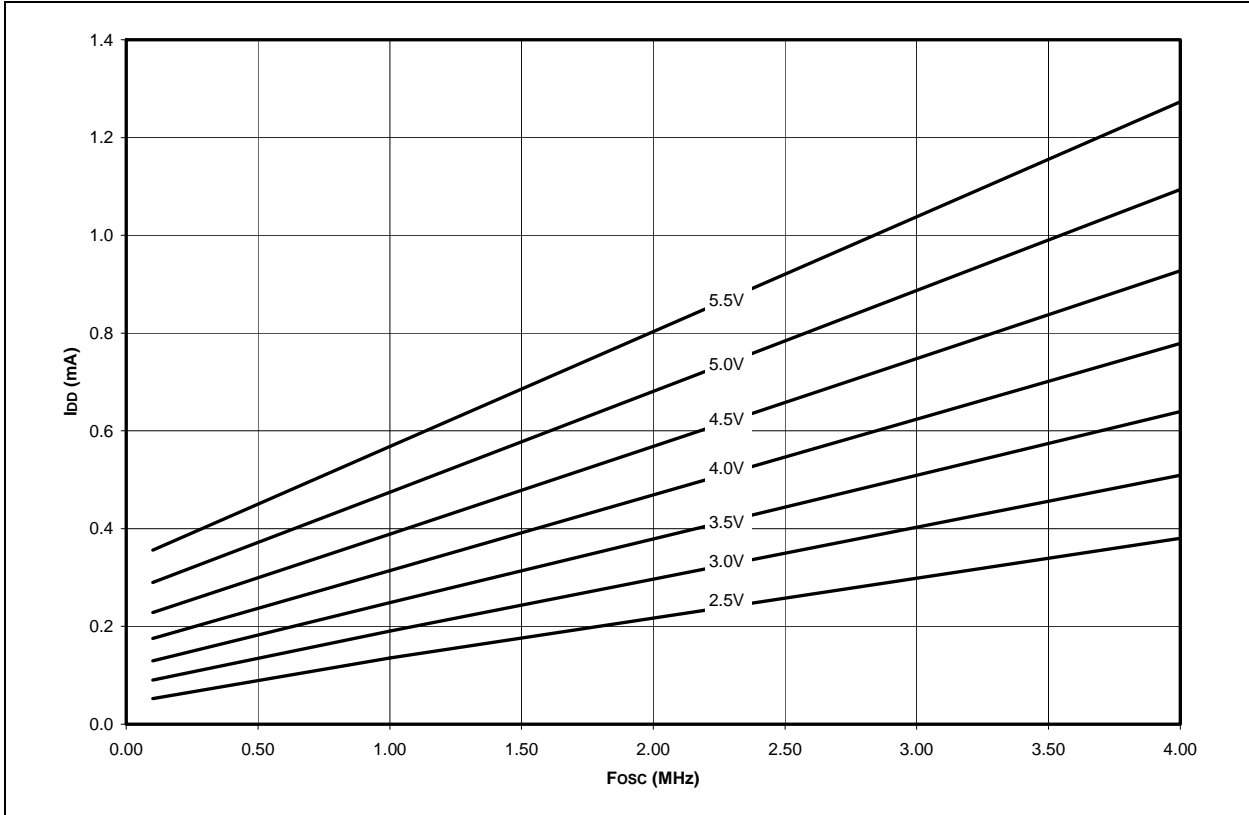
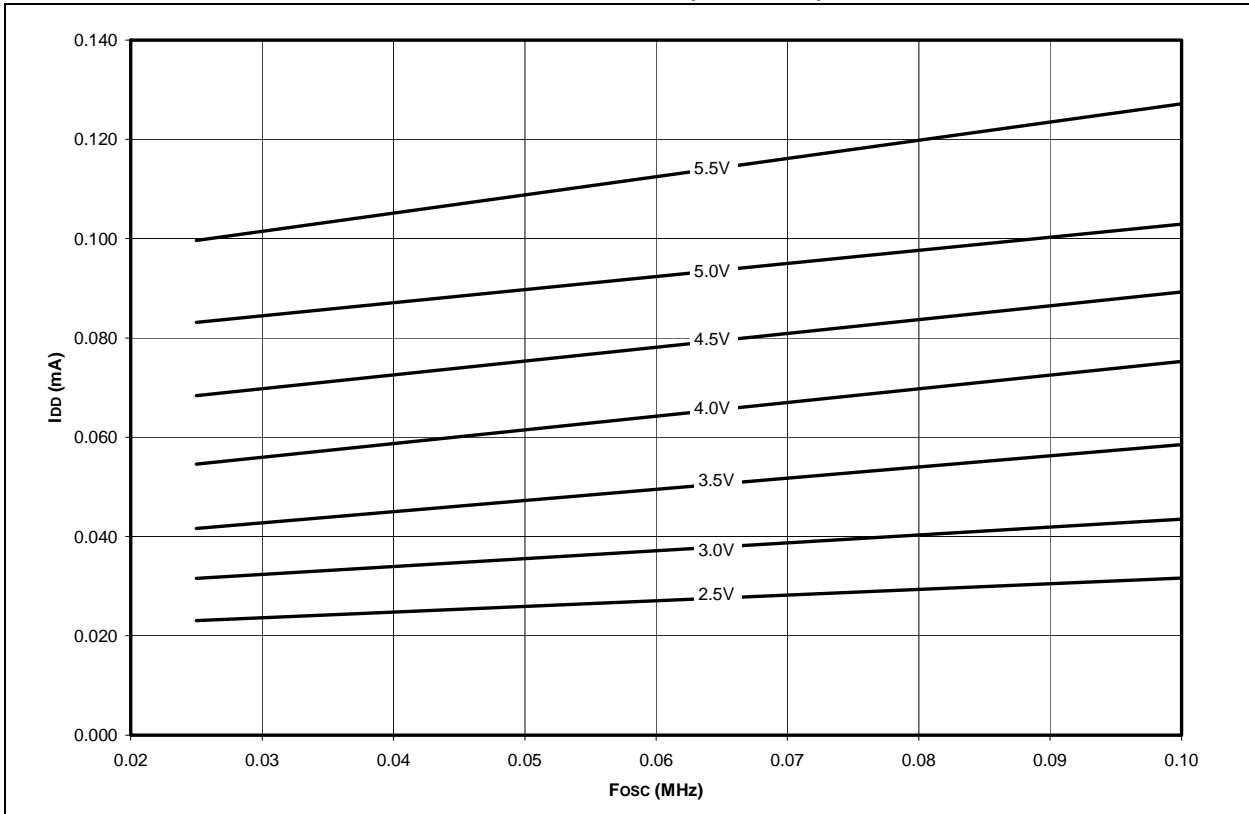


FIGURE 16-5: MAXIMUM I_{DD} VS. F_{osc} OVER V_{DD} (LP MODE)



PIC16C717/770/771

FIGURE 16-6: TYPICAL I_{DD} VS. F_{osc} OVER V_{DD} (LP MODE)

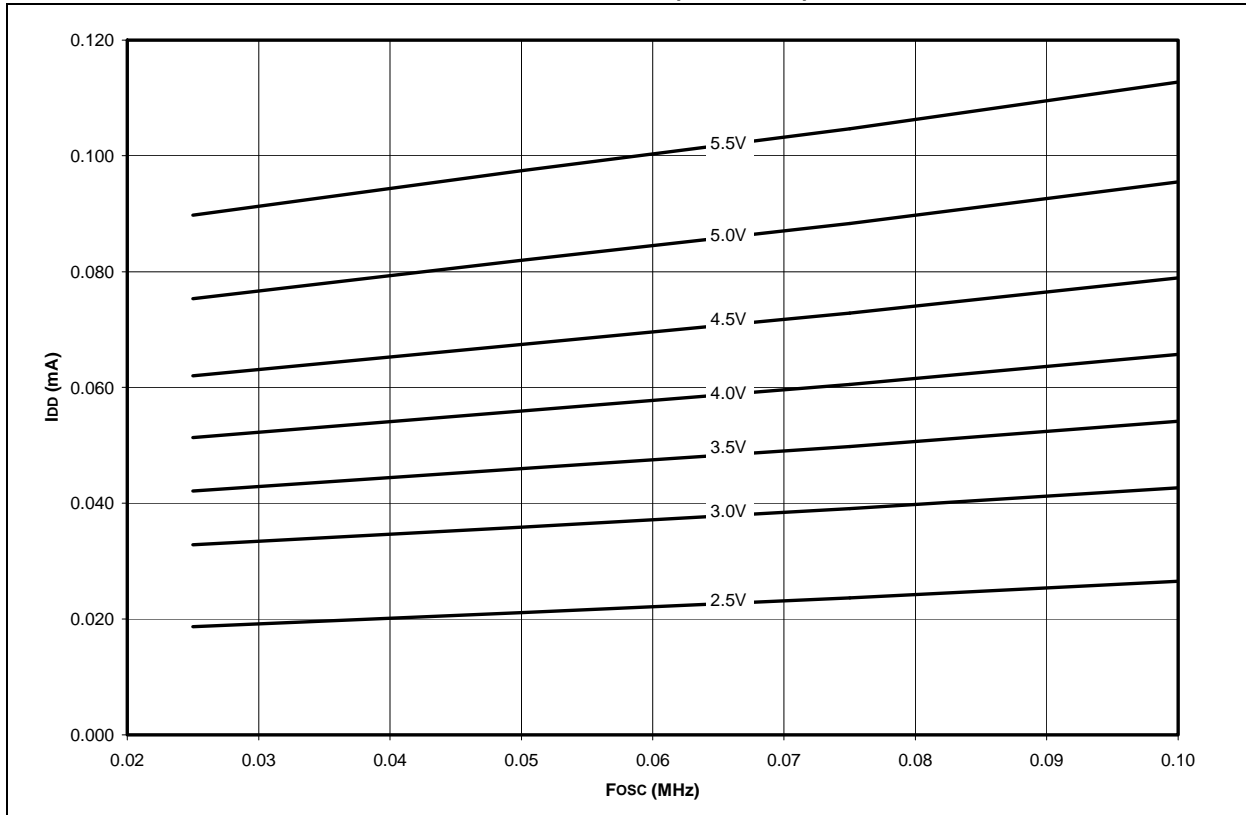


FIGURE 16-7: MAXIMUM I_{DD} VS. F_{osc} OVER V_{DD} (EC MODE)

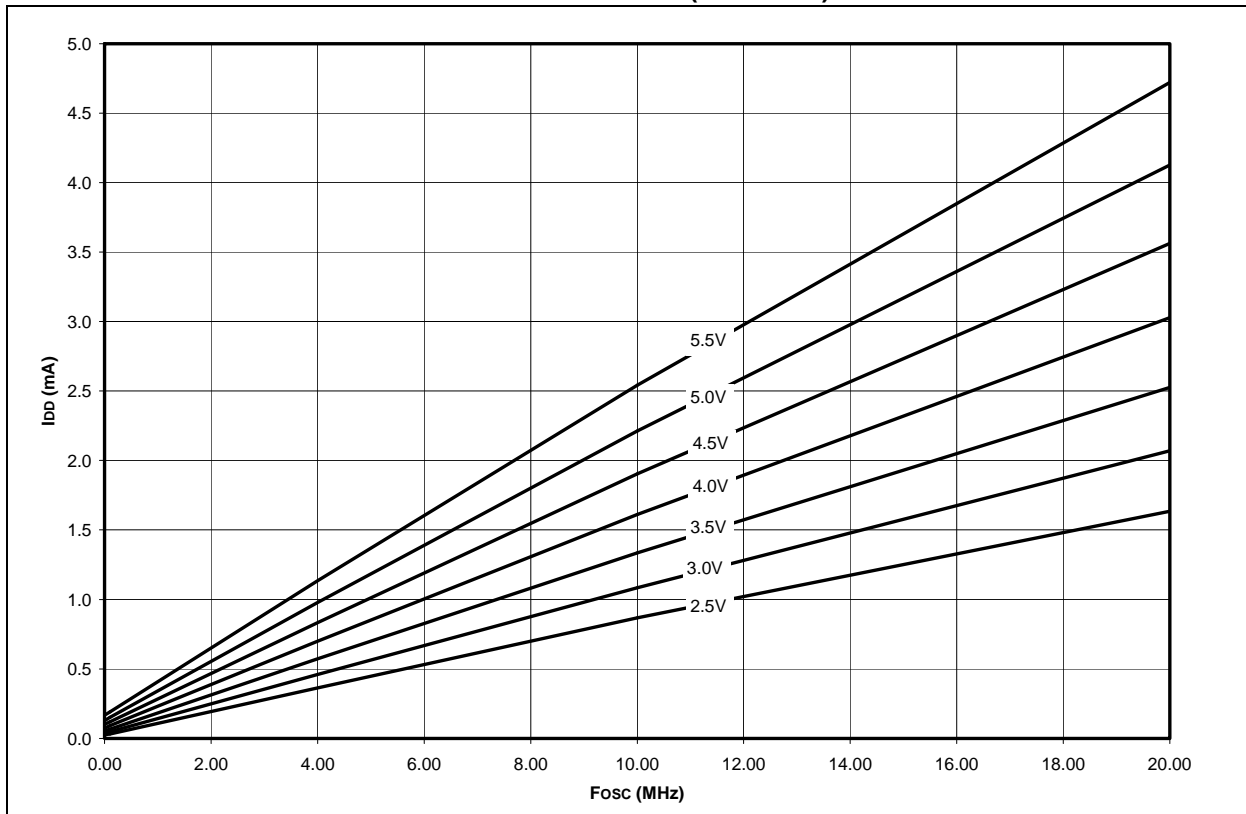


FIGURE 16-8: TYPICAL I_{DD} VS. F_{osc} OVER V_{DD} (EC MODE)

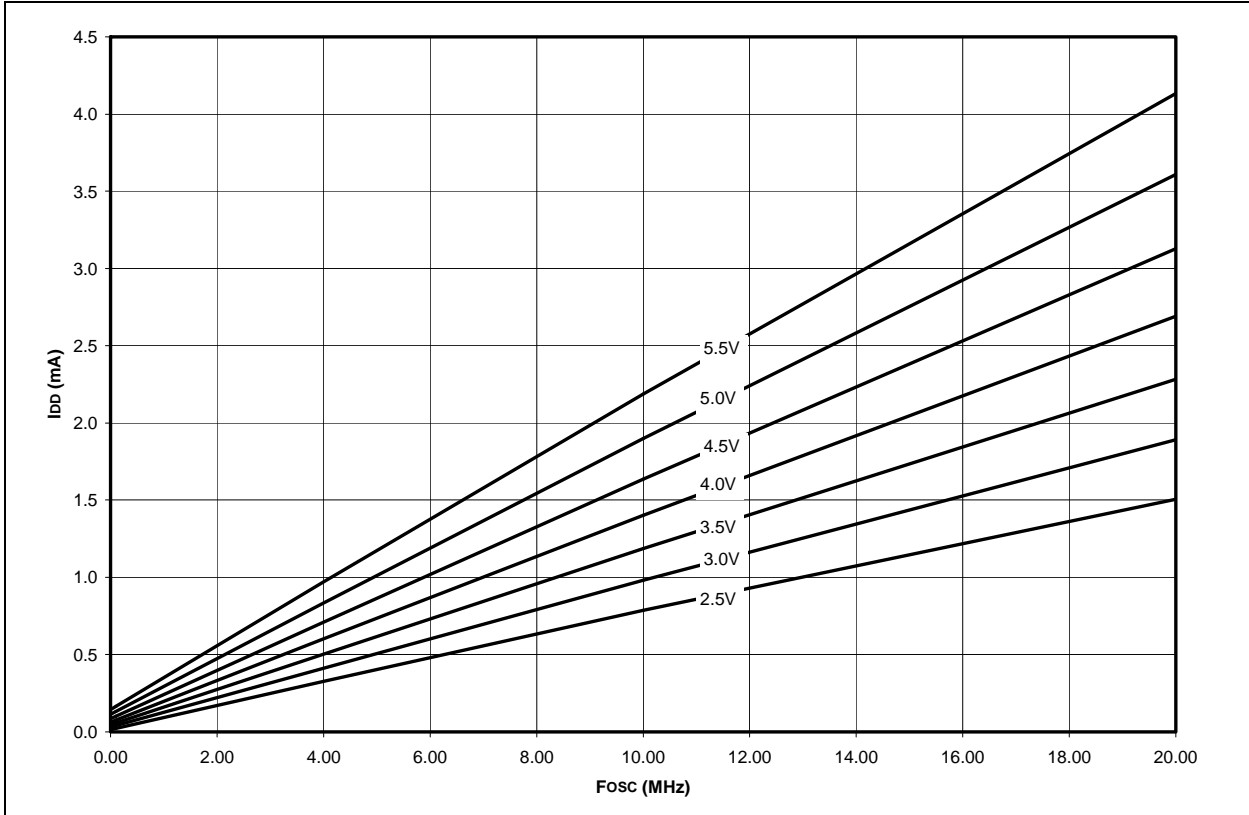
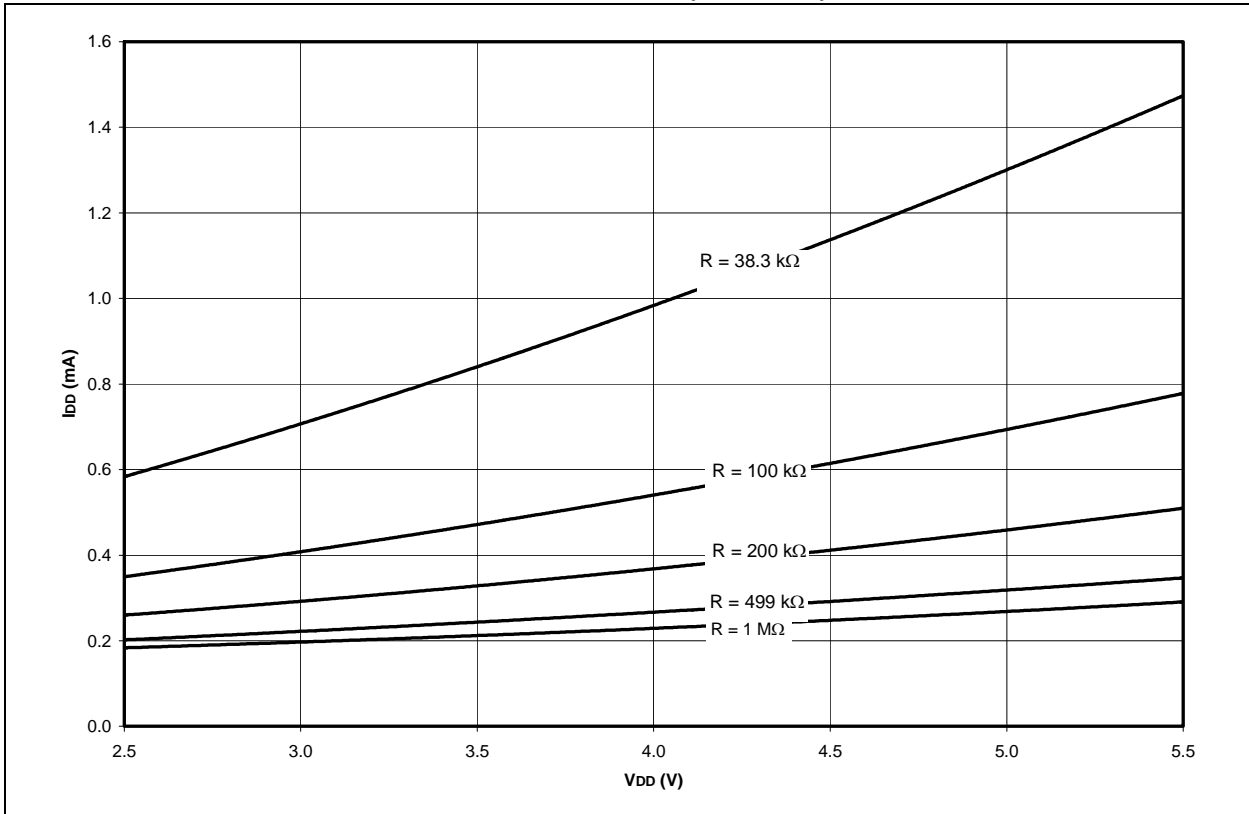


FIGURE 16-9: MAXIMUM I_{DD} VS. F_{osc} OVER V_{DD} (ER MODE)



PIC16C717/770/771

FIGURE 16-10: TYPICAL I_{DD} VS. F_{osc} OVER V_{DD} (ER MODE)

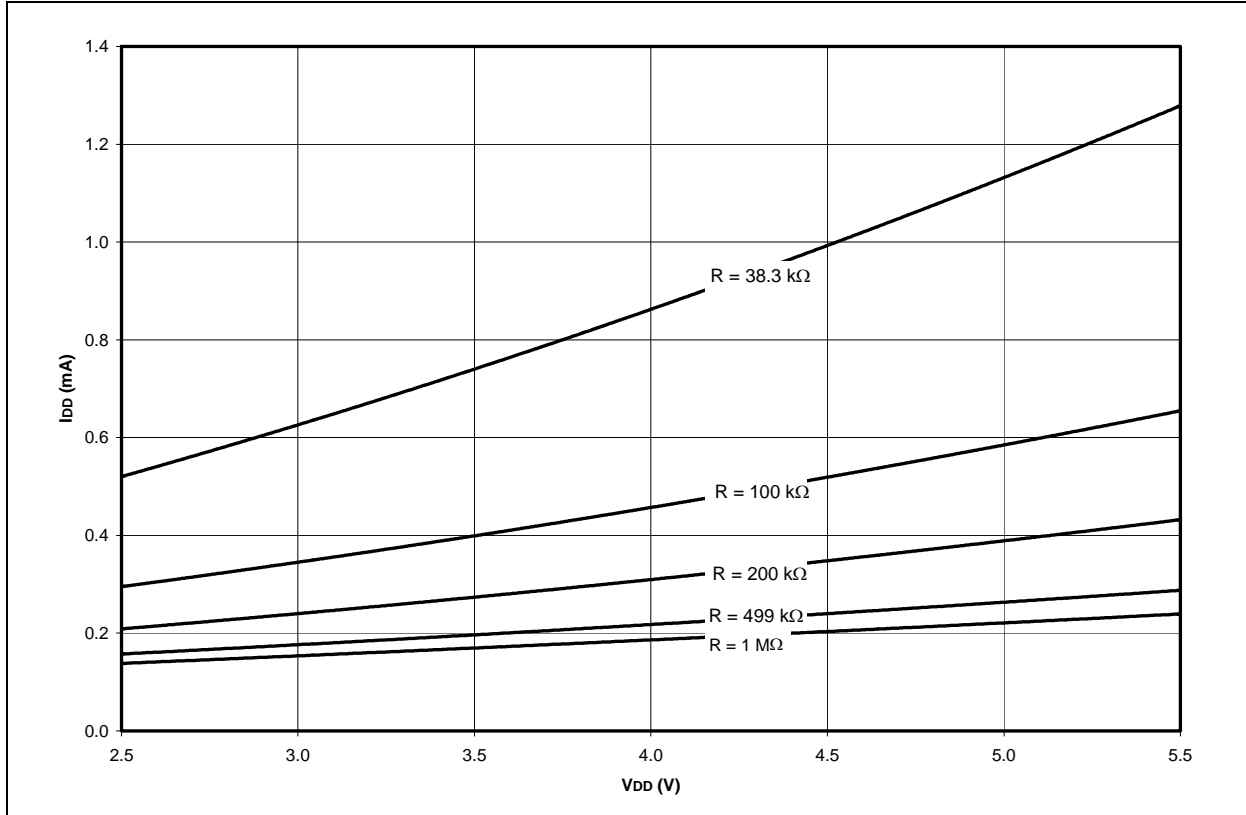


FIGURE 16-11: TYPICAL F_{osc} VS. V_{DD} (ER MODE)

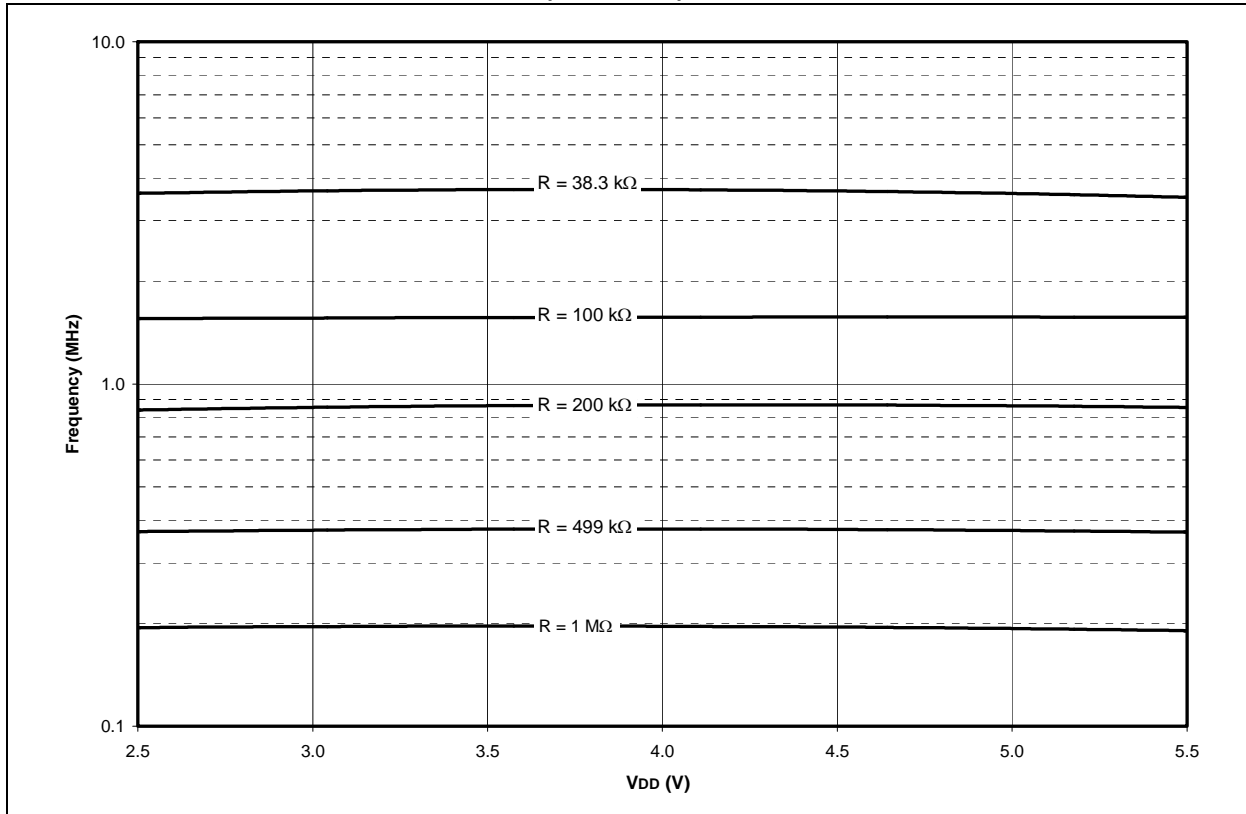


FIGURE 16-12: MAXIMUM I_{DD} VS. V_{DD} (INTRC 37 kHz MODE)

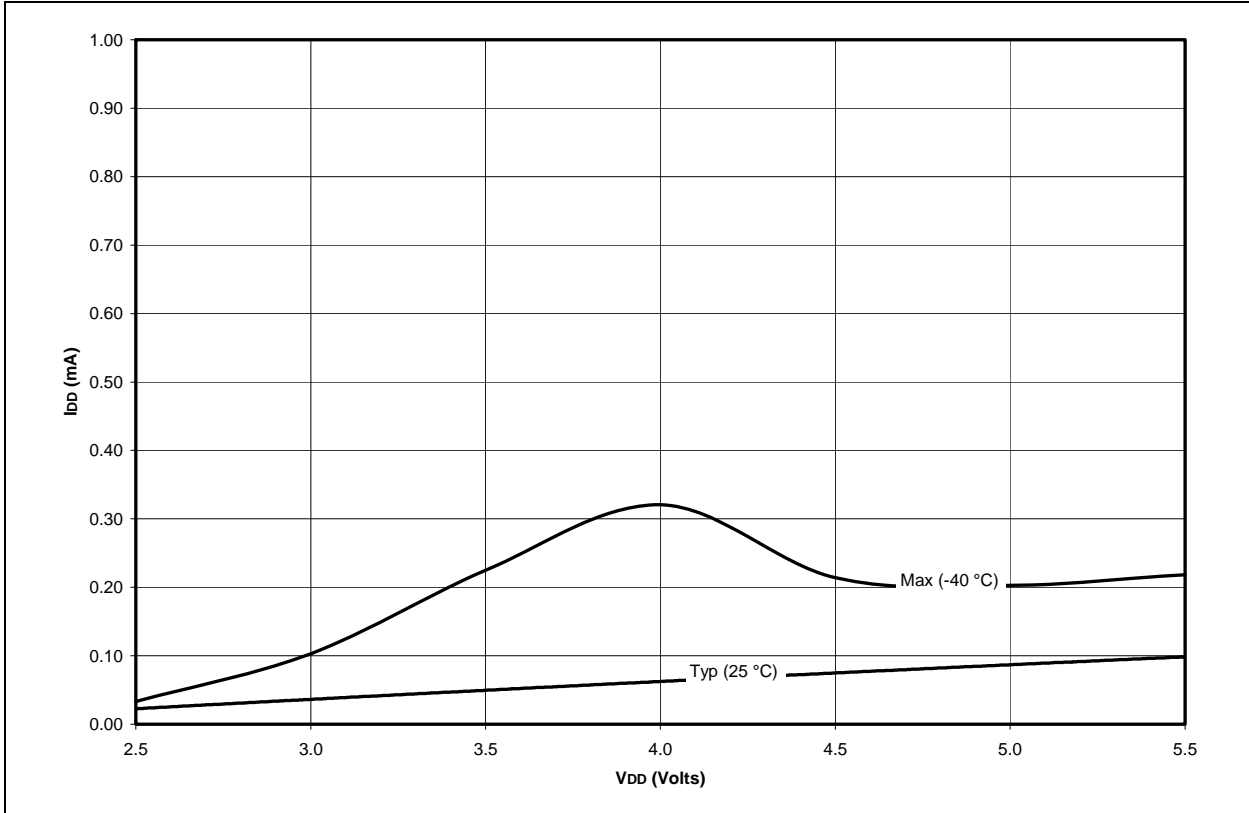
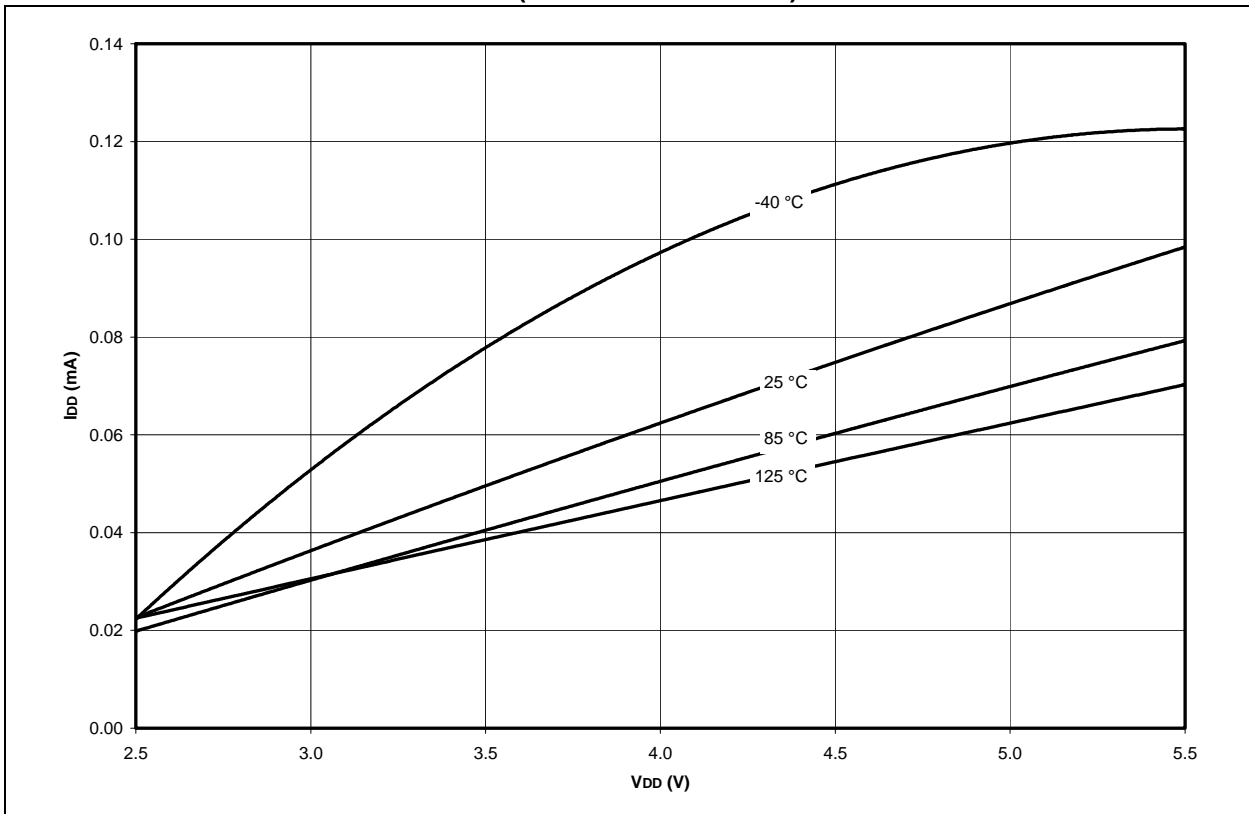


FIGURE 16-13: TYPICAL I_{DD} VS. V_{DD} (INTRC 37 kHz MODE)



PIC16C717/770/771

FIGURE 16-14: INTERNAL RC Fosc VS. VDD OVER TEMPERATURE (37 KHZ)

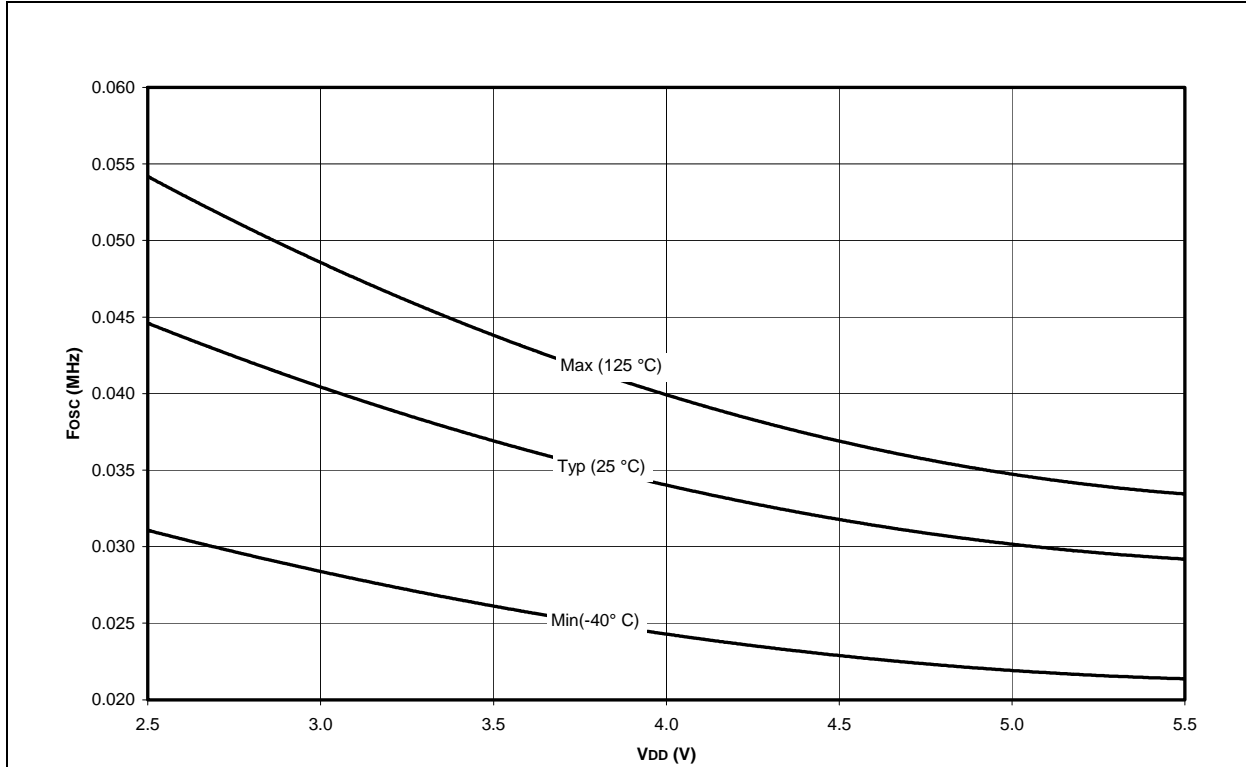


FIGURE 16-15: MAXIMUM AND TYPICAL IDD VS. VDD (INTRC 4 MHz MODE)

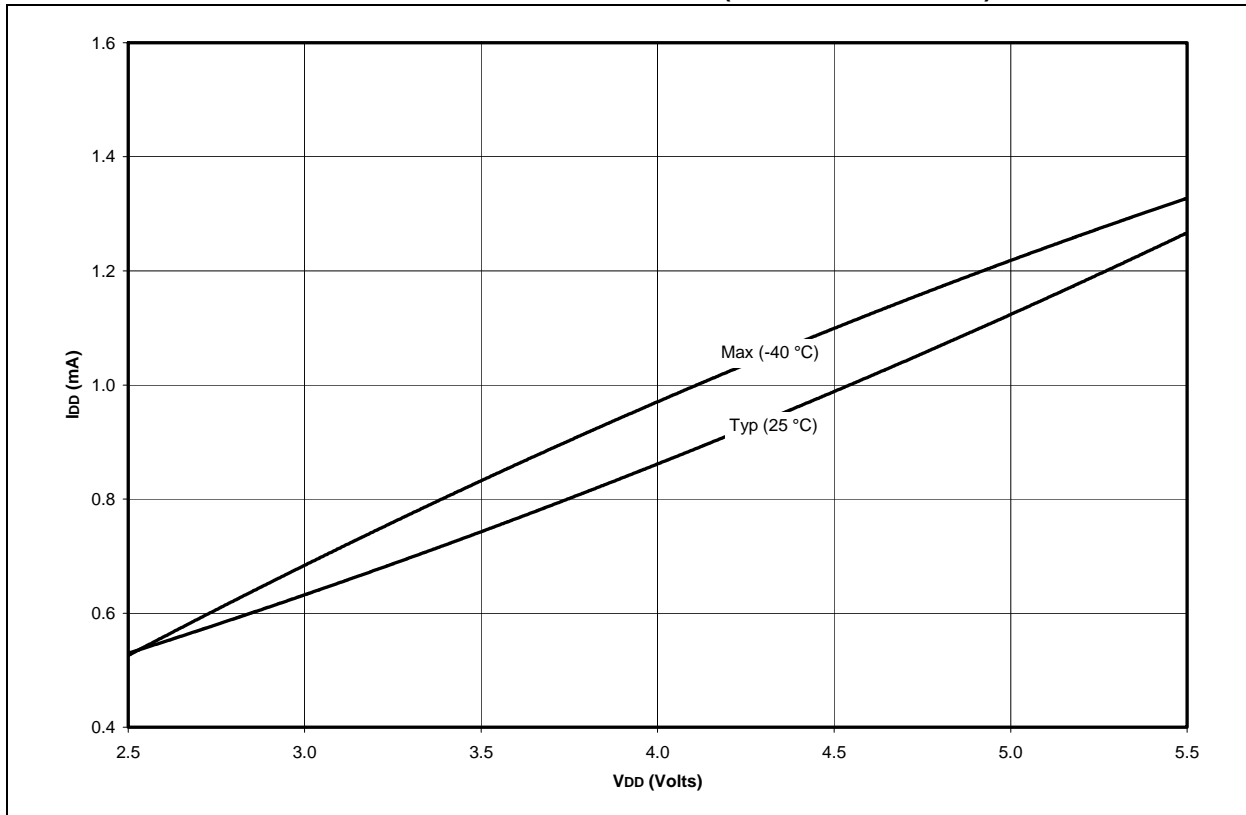


FIGURE 16-16: TYPICAL I_{DD} VS. V_{DD} (INTRC 4 MHz MODE)

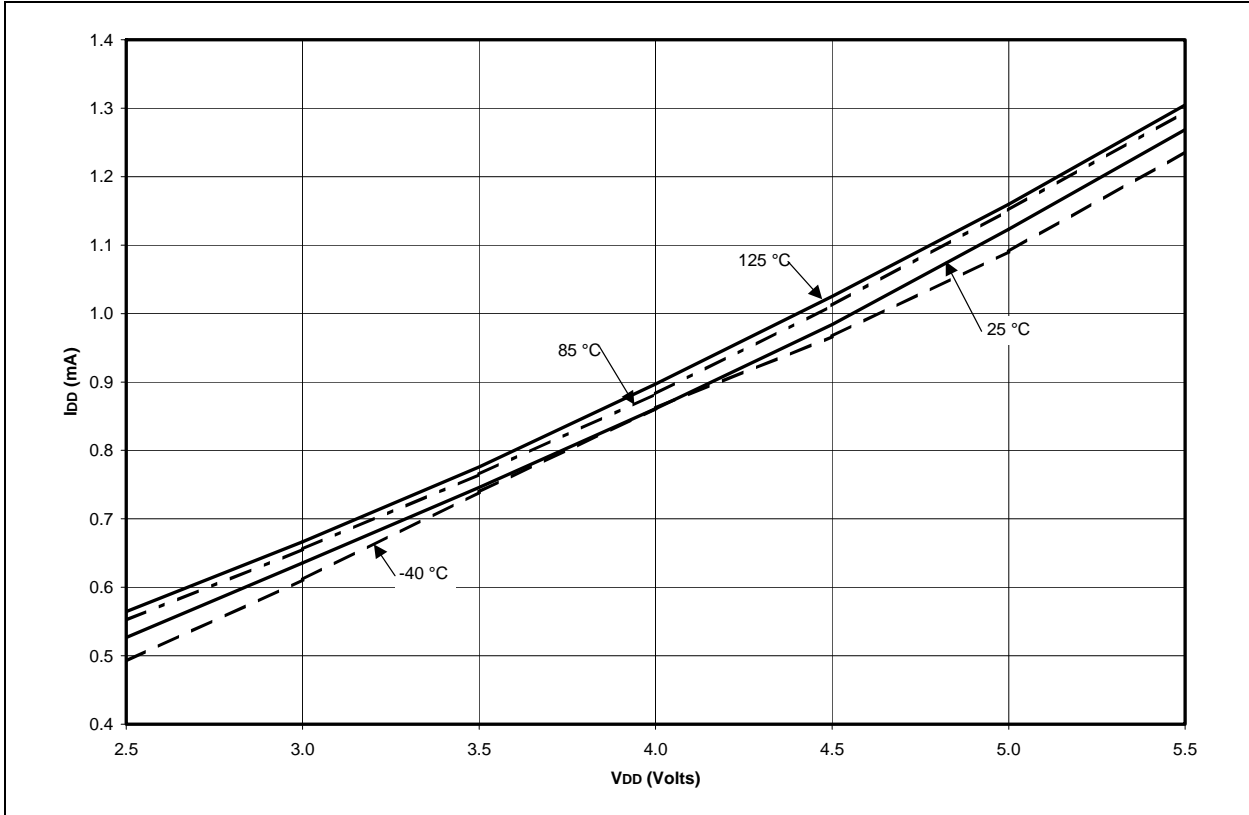
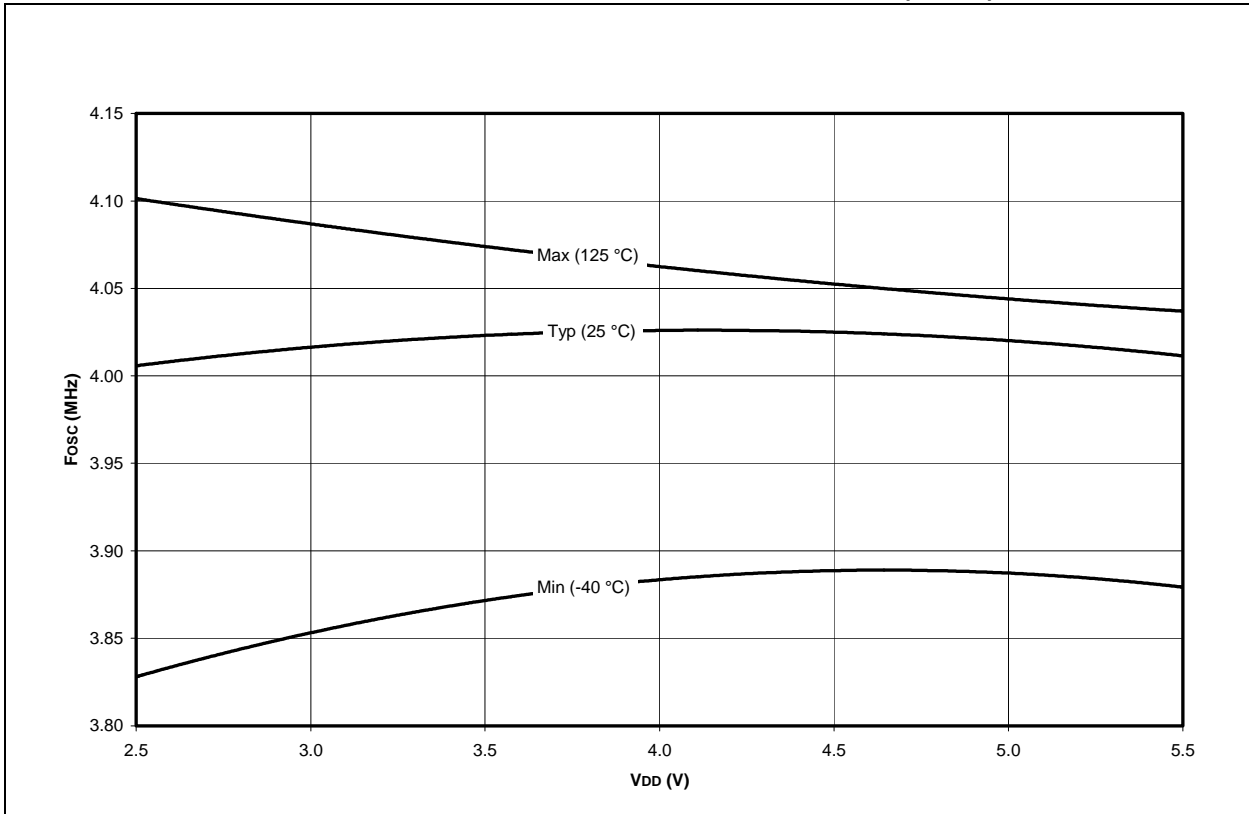


FIGURE 16-17: INTERNAL RC F_{osc} VS. V_{DD} OVER TEMPERATURE (4 MHz)



PIC16C717/770/771

FIGURE 16-18: MAXIMUM IPD VS. VDD (-40°C TO +125°C)

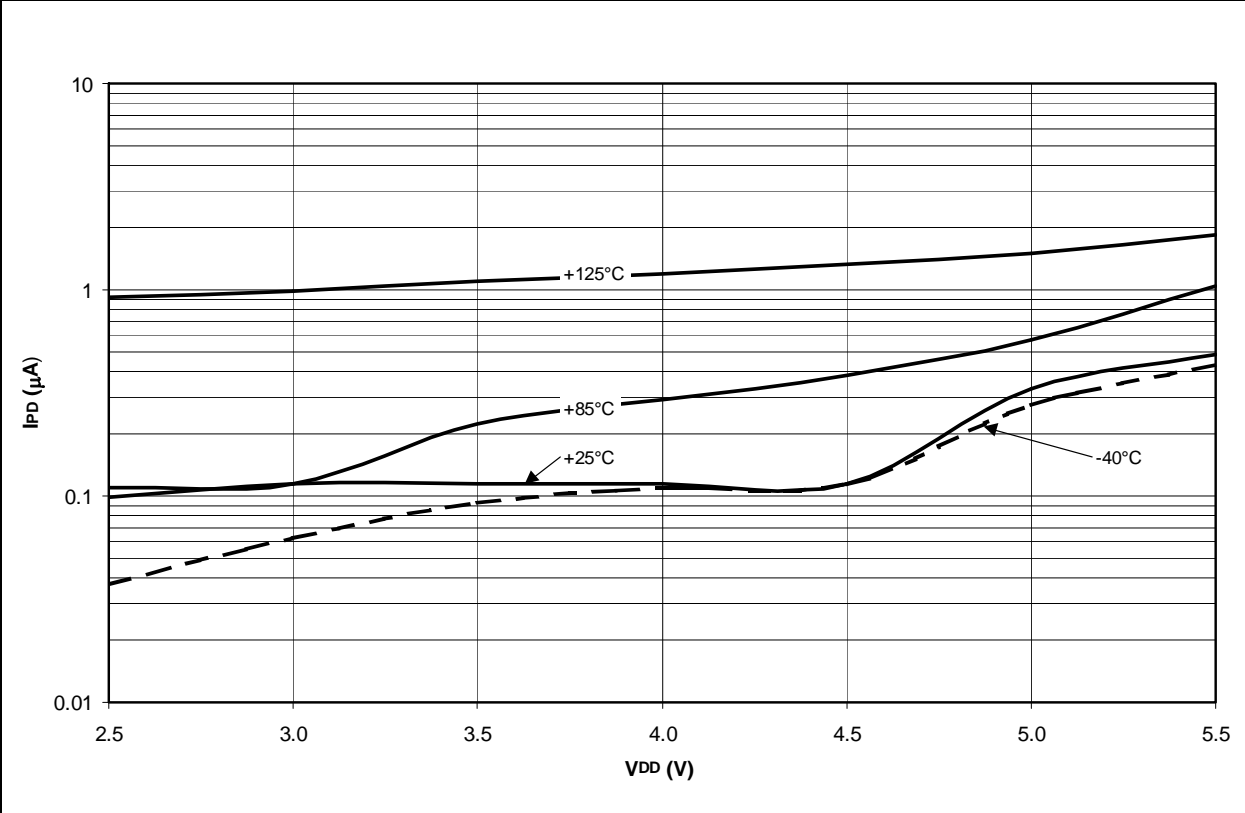
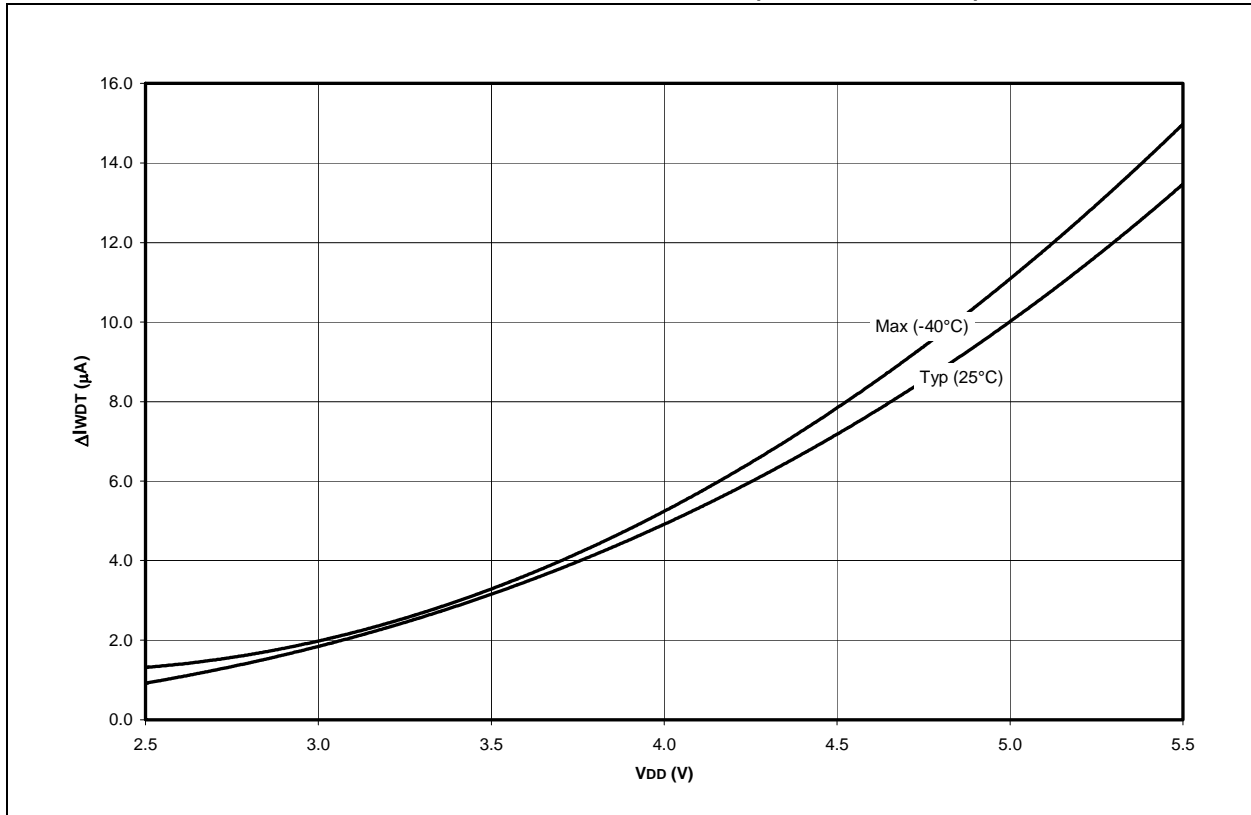


FIGURE 16-19: TYPICAL AND MAXIMUM ΔI_{WDT} VS. V_{DD} (-40°C TO +125°C)



PIC16C717/770/771

FIGURE 16-20: TYPICAL AND MAXIMUM ΔI_{TMR1} VS. V_{DD} (32 KHZ, -40°C TO $+125^{\circ}\text{C}$)

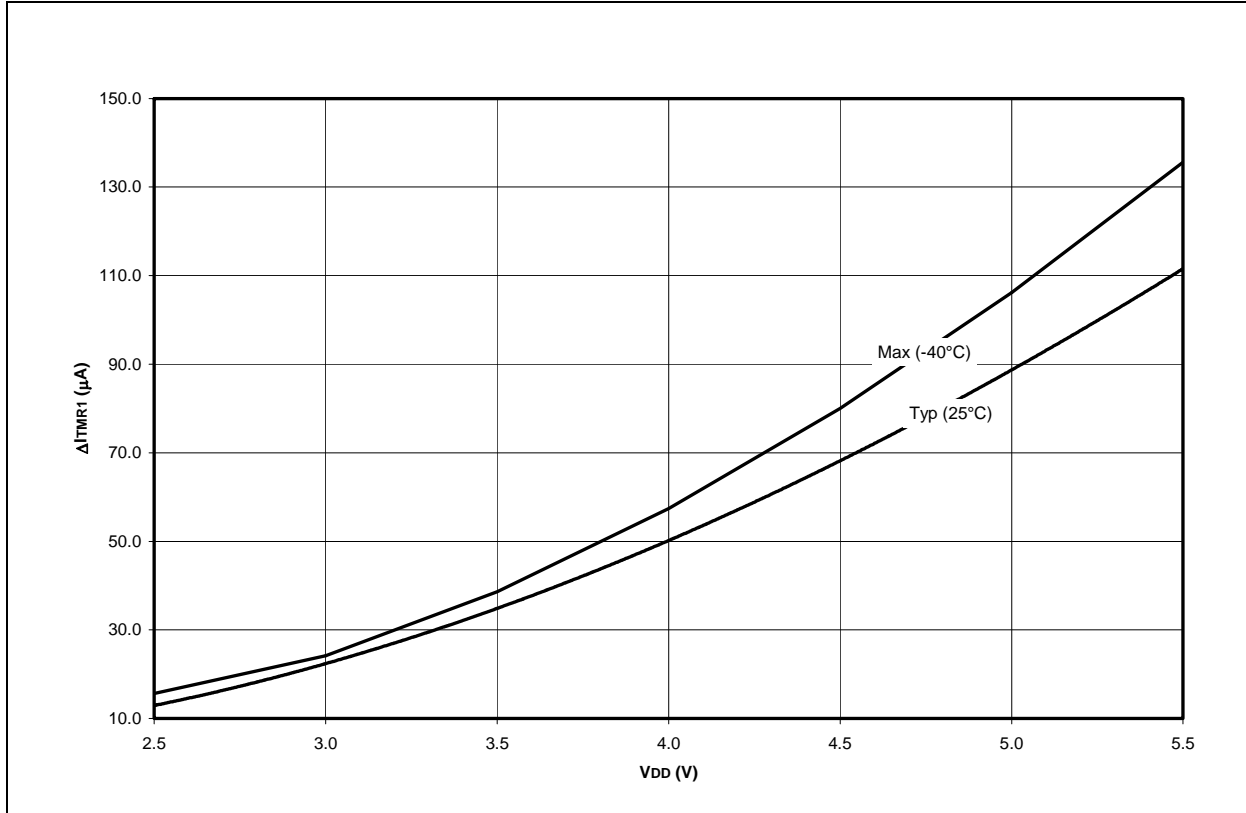


FIGURE 16-21: TYPICAL AND MAXIMUM ΔI_{VRL} VS. V_{DD} (-40°C TO $+125^{\circ}\text{C}$)

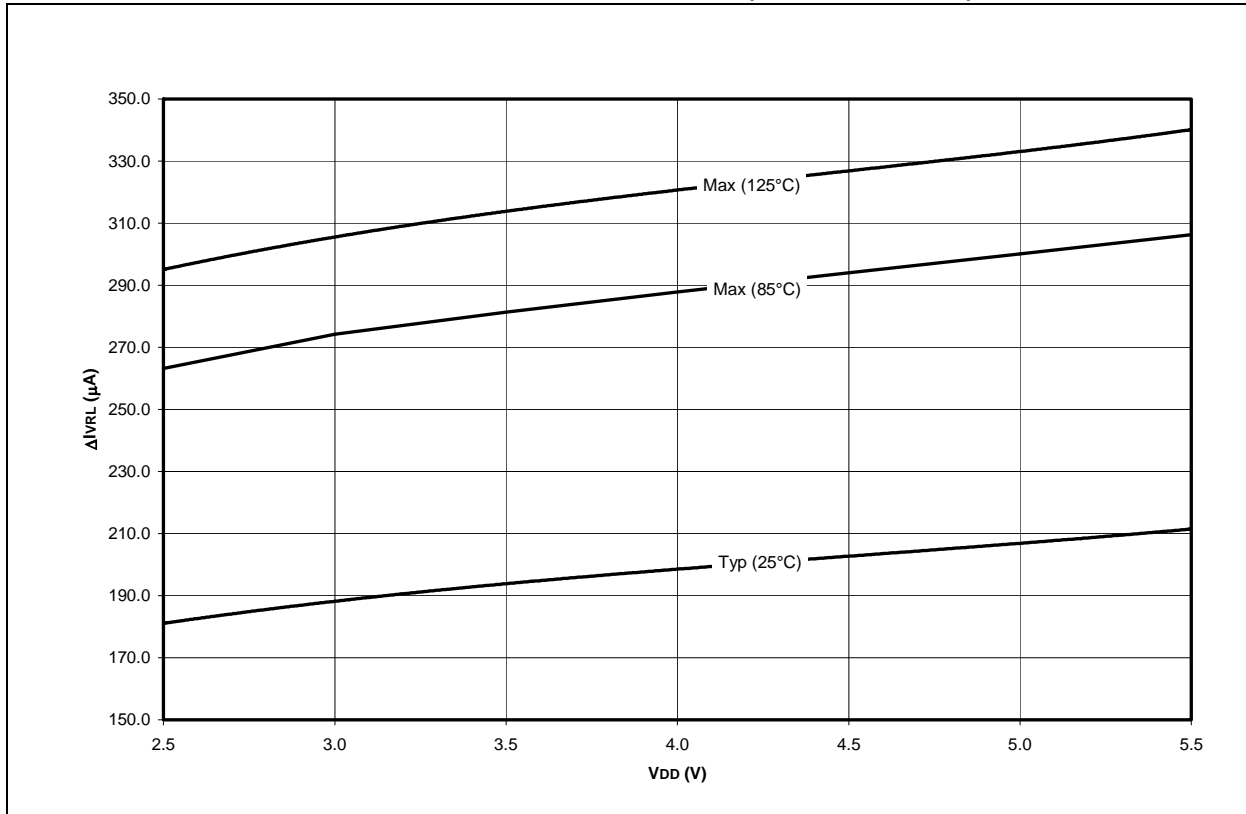


FIGURE 16-22: TYPICAL AND MAXIMUM ΔI_{VRH} VS. V_{DD} (-40°C TO +125°C)

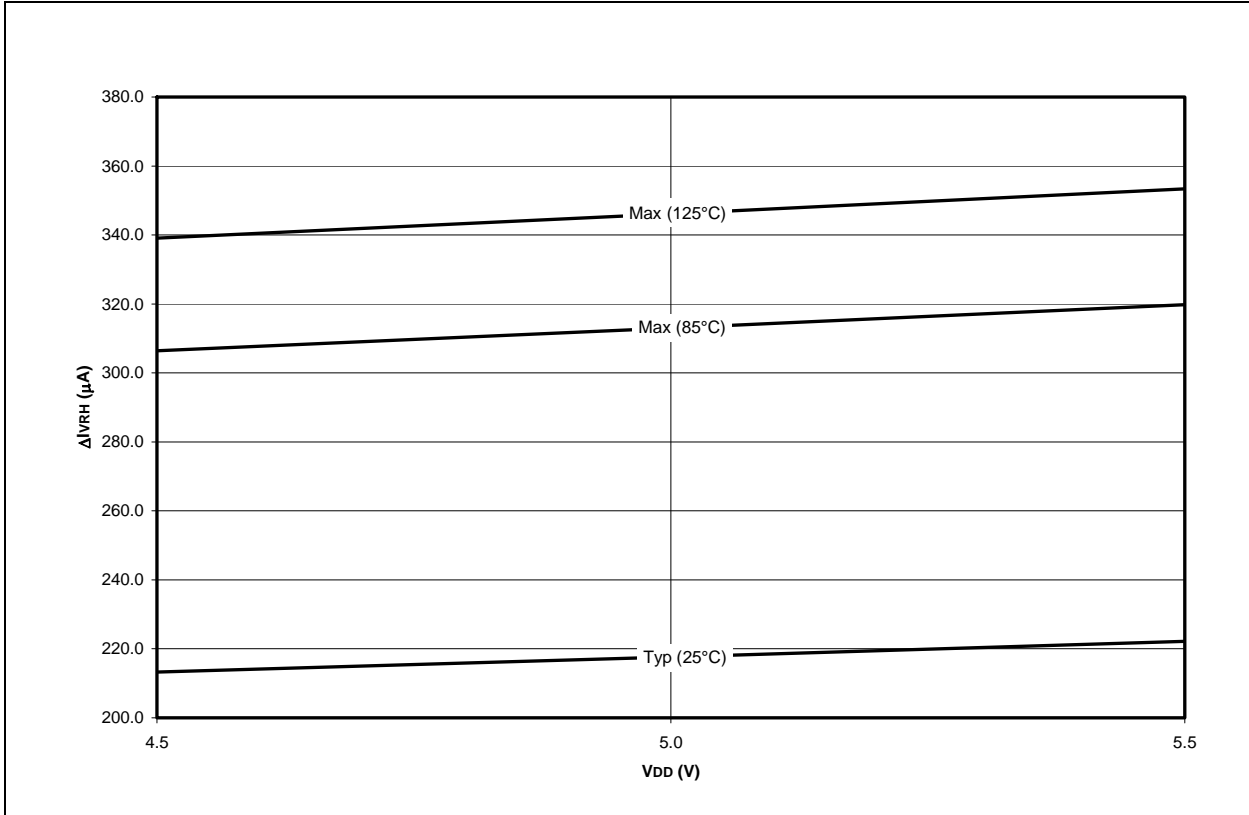
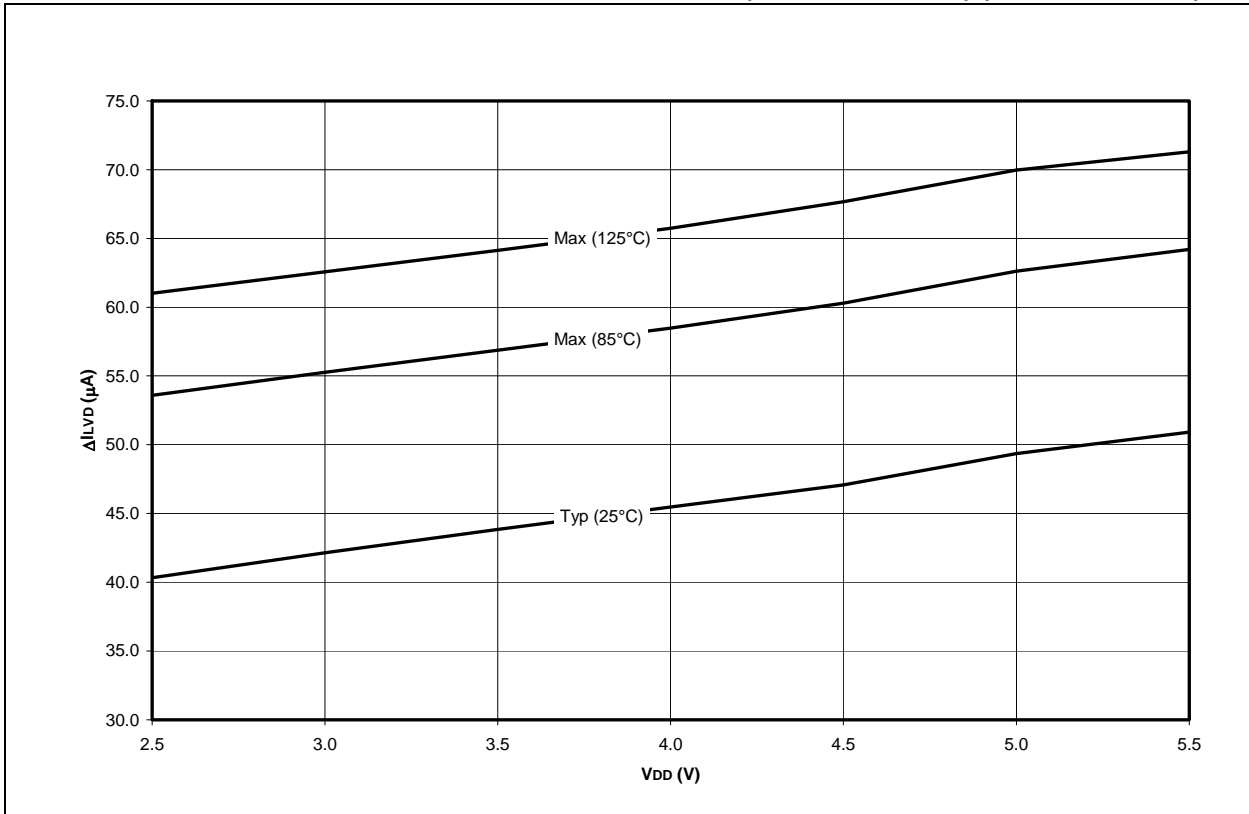


FIGURE 16-23: TYPICAL AND MAXIMUM ΔI_{LVD} VS. V_{DD} (-40°C TO +125°C) (LVD TRIP = 3.0V)



PIC16C717/770/771

FIGURE 16-24: TYPICAL AND MAXIMUM ΔI_{LVD} VS. V_{DD} (-40°C TO +125°C) (LVD TRIP = 4.5V)

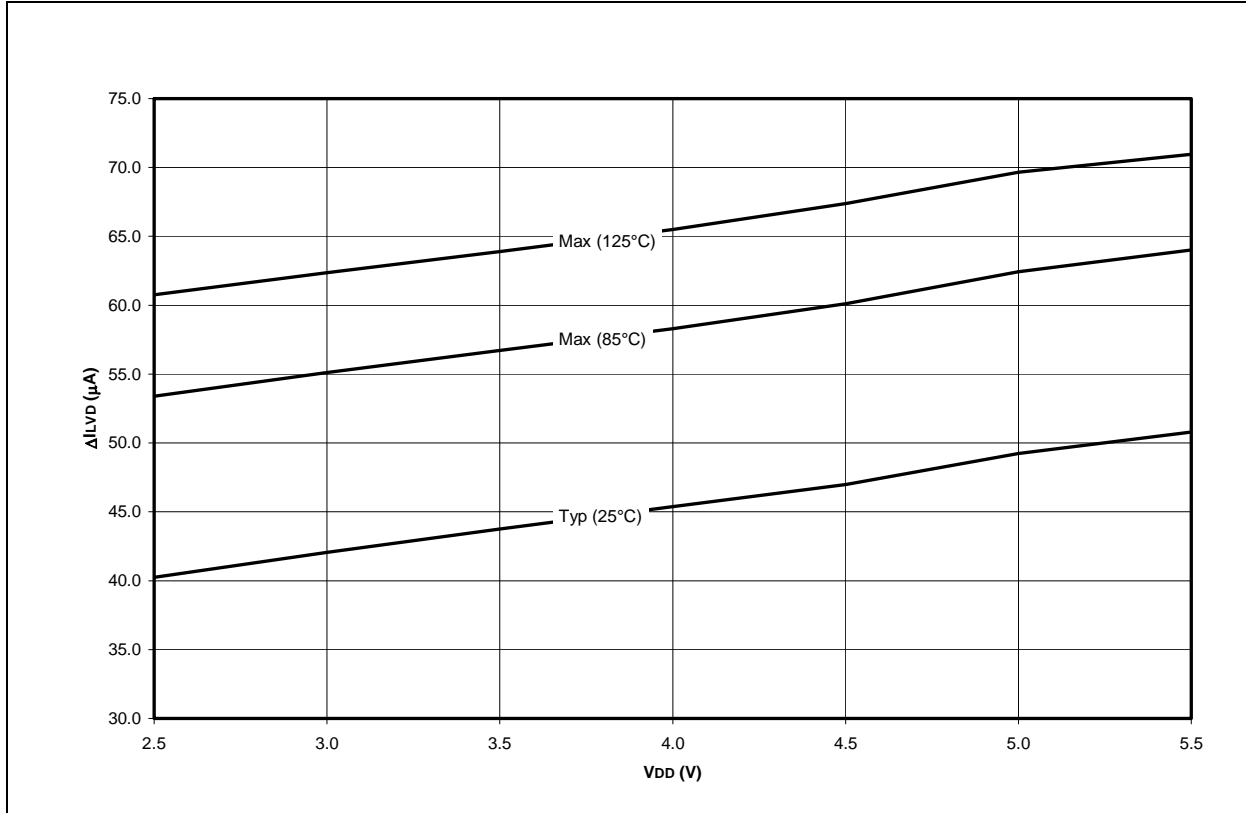


FIGURE 16-25: TYPICAL AND MAXIMUM ΔI_{BOR} VS. V_{DD} (-40°C TO +125°C) ($V_{BOR} = 2.5V$)

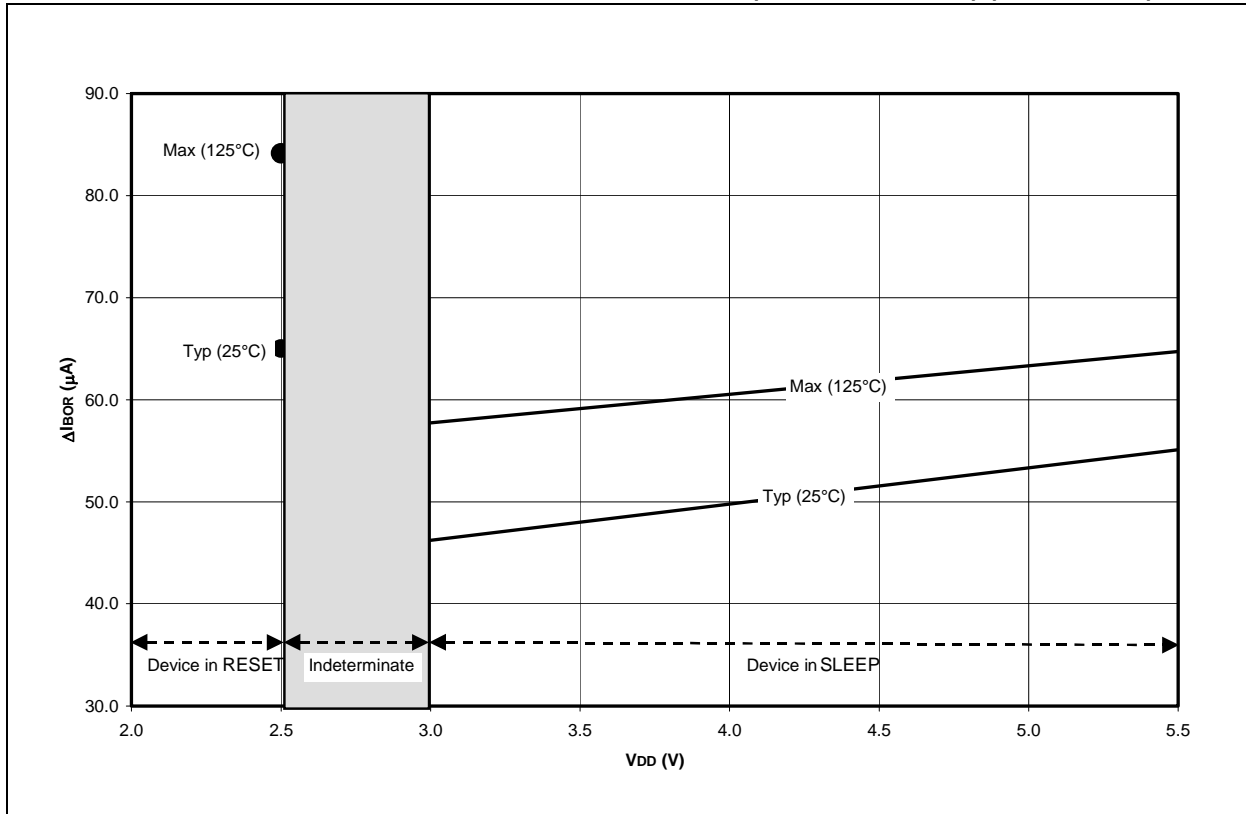


FIGURE 16-26: TYPICAL AND MAXIMUM ΔI_{BOR} VS. V_{DD} (-40°C TO +125°C) ($V_{BOR} = 4.5V$)

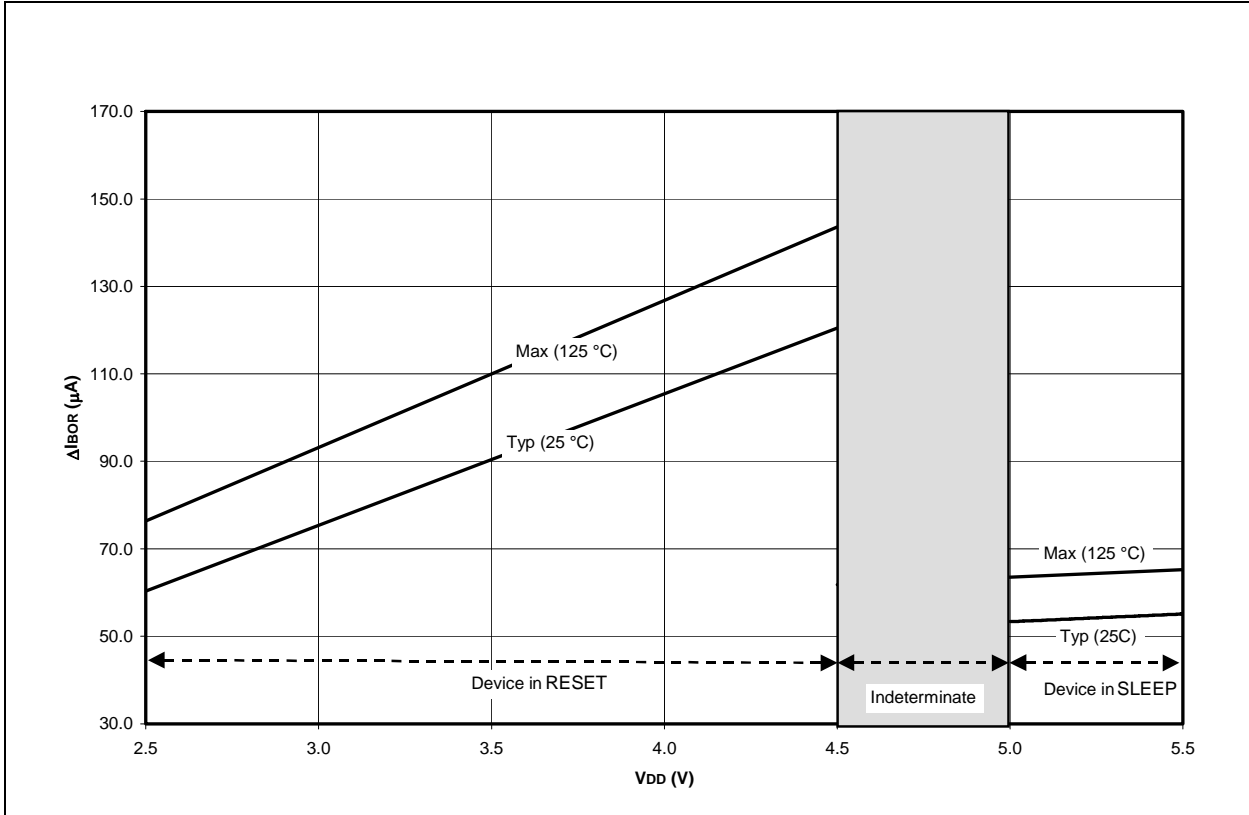
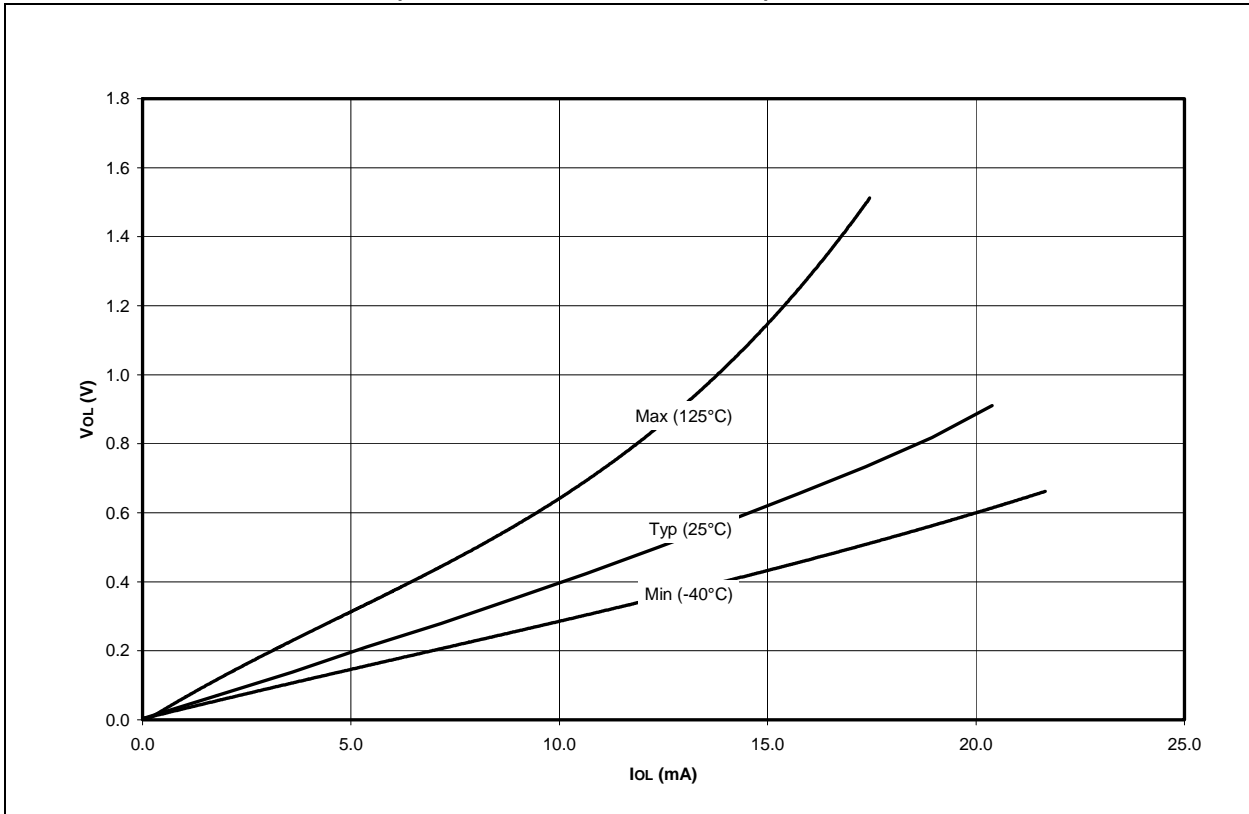


FIGURE 16-27: V_{OL} VS. I_{OL} (-40°C TO +125°C, $V_{DD} = 3.0V$)



PIC16C717/770/771

FIGURE 16-28: V_{OL} VS. I_{OL} (-40°C TO $+125^{\circ}\text{C}$, $V_{DD} = 5.0\text{V}$)

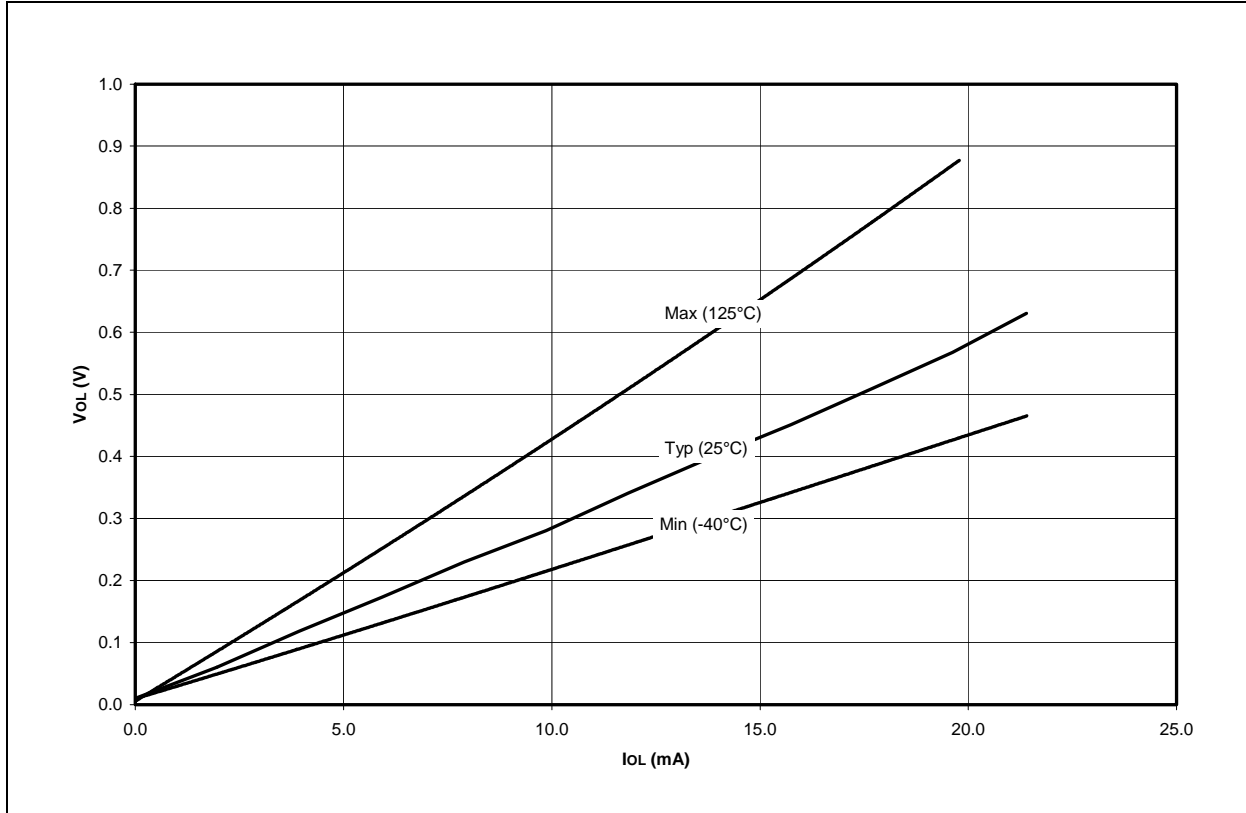


FIGURE 16-29: V_{OH} VS. I_{OH} (-40°C TO $+125^{\circ}\text{C}$, $V_{DD} = 3.0\text{V}$)

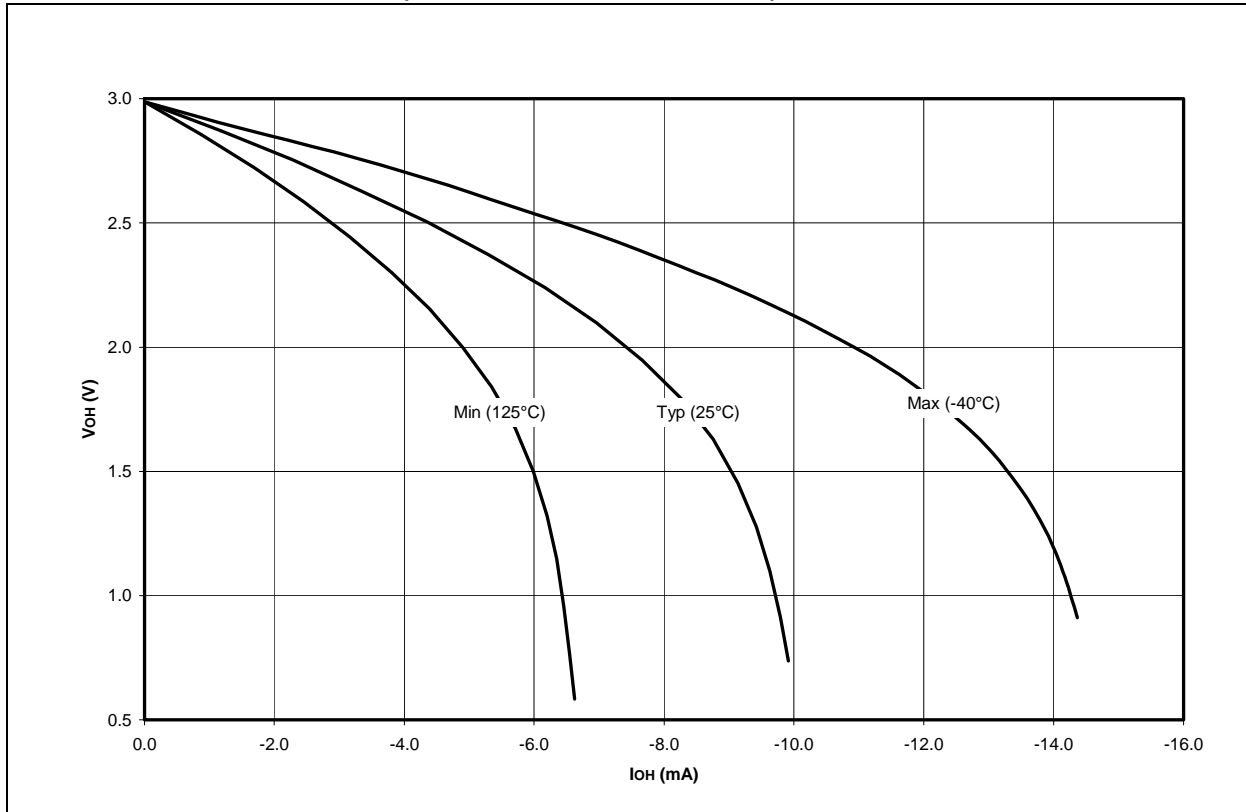


FIGURE 16-30: V_{OH} VS. I_{OH} (-40°C TO +125°C, $V_{DD} = 5.0V$)

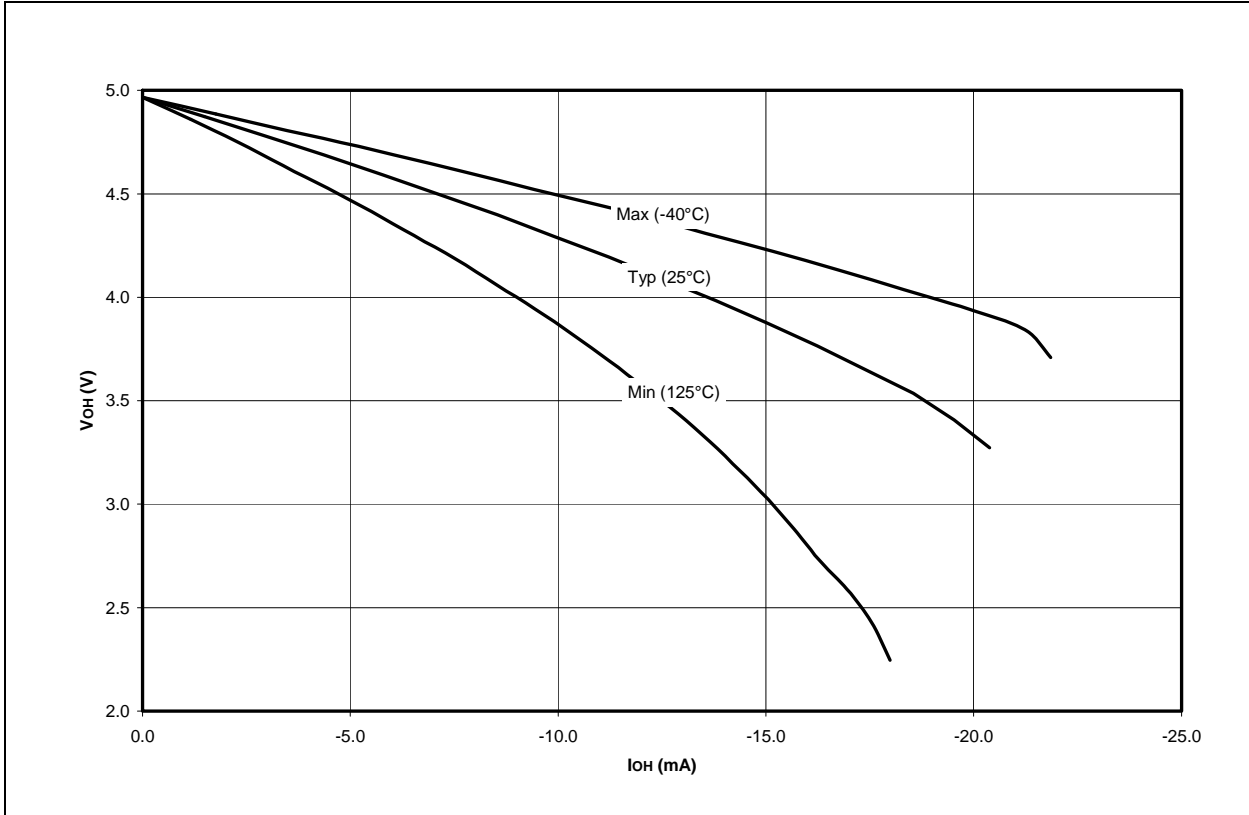
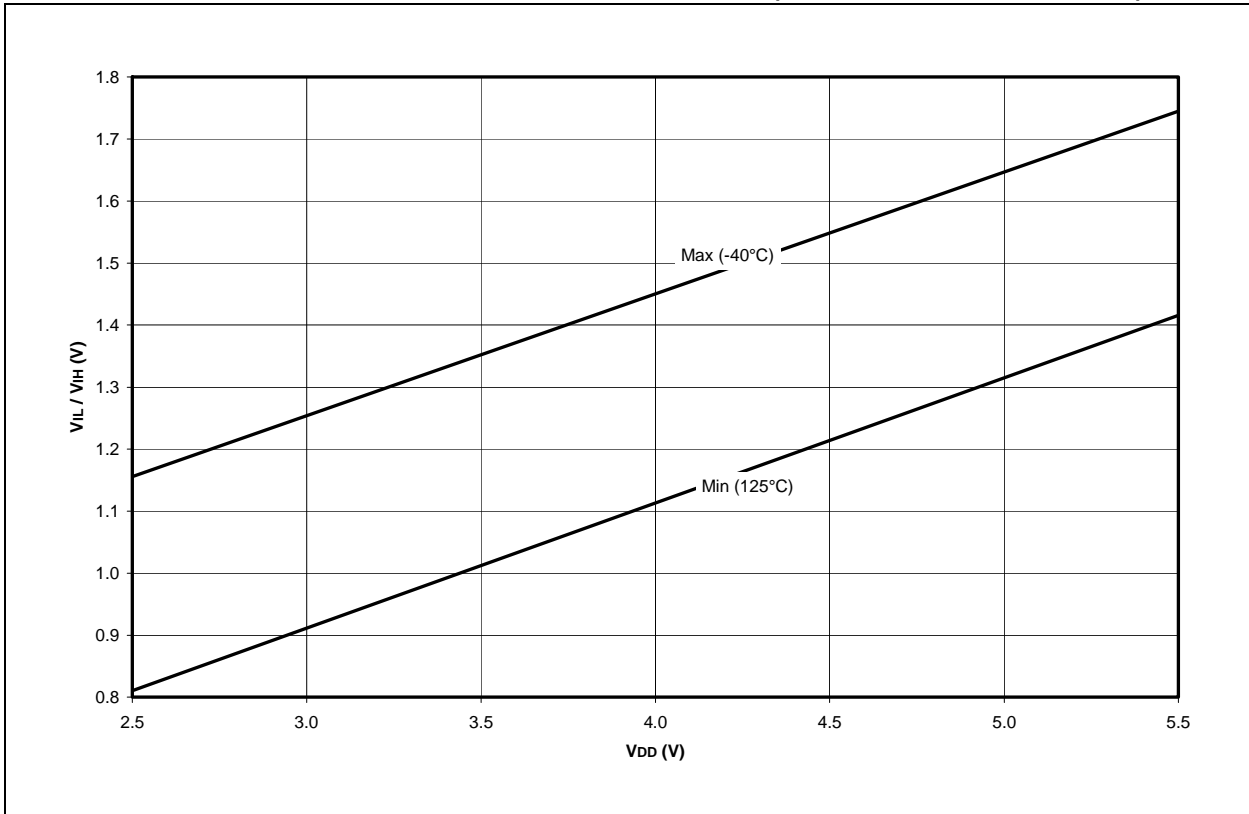


FIGURE 16-31: MINIMUM AND MAXIMUM V_{IH}/V_{IL} VS. V_{DD} (TTL INPUT, -40°C TO +125°C)



PIC16C717/770/771

FIGURE 16-32: MINIMUM AND MAXIMUM V_{IH}/V_{IL} VS. V_{DD} (ST INPUT, -40°C TO $+125^{\circ}\text{C}$)

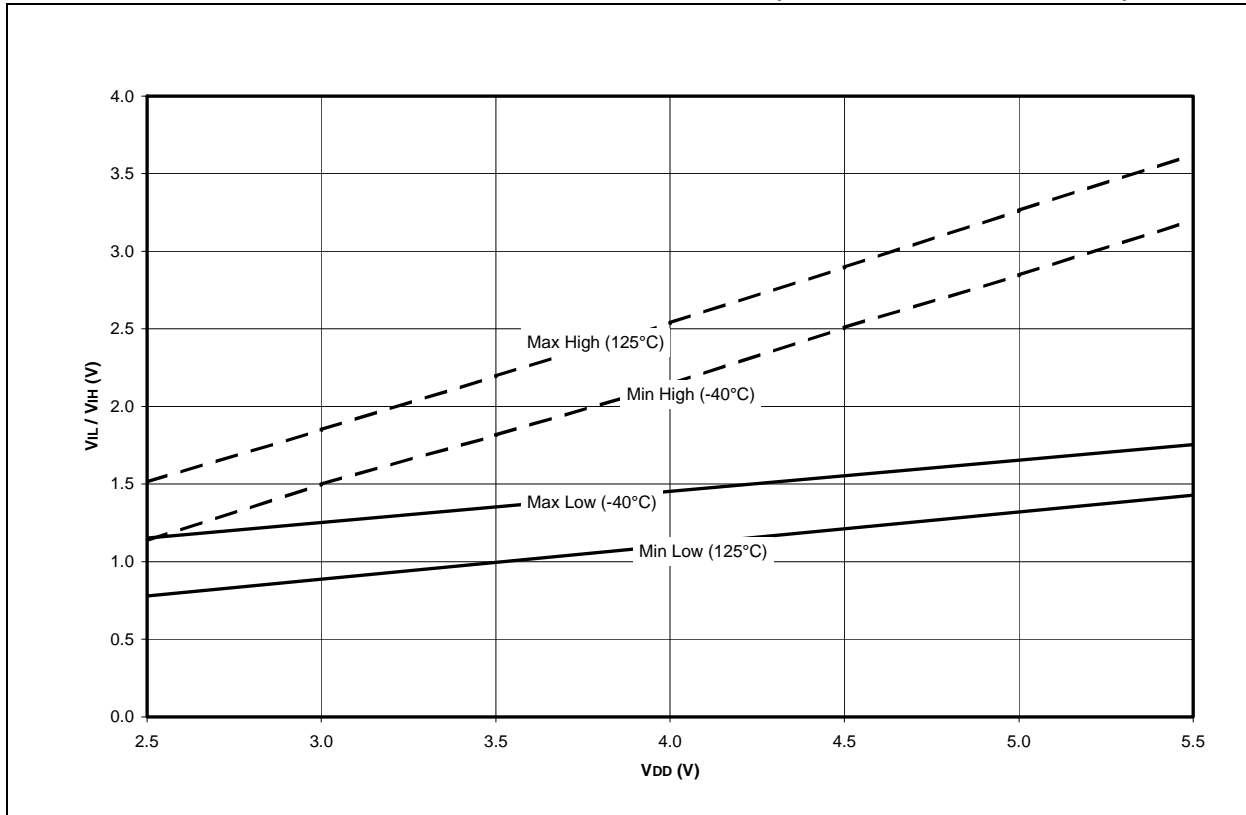
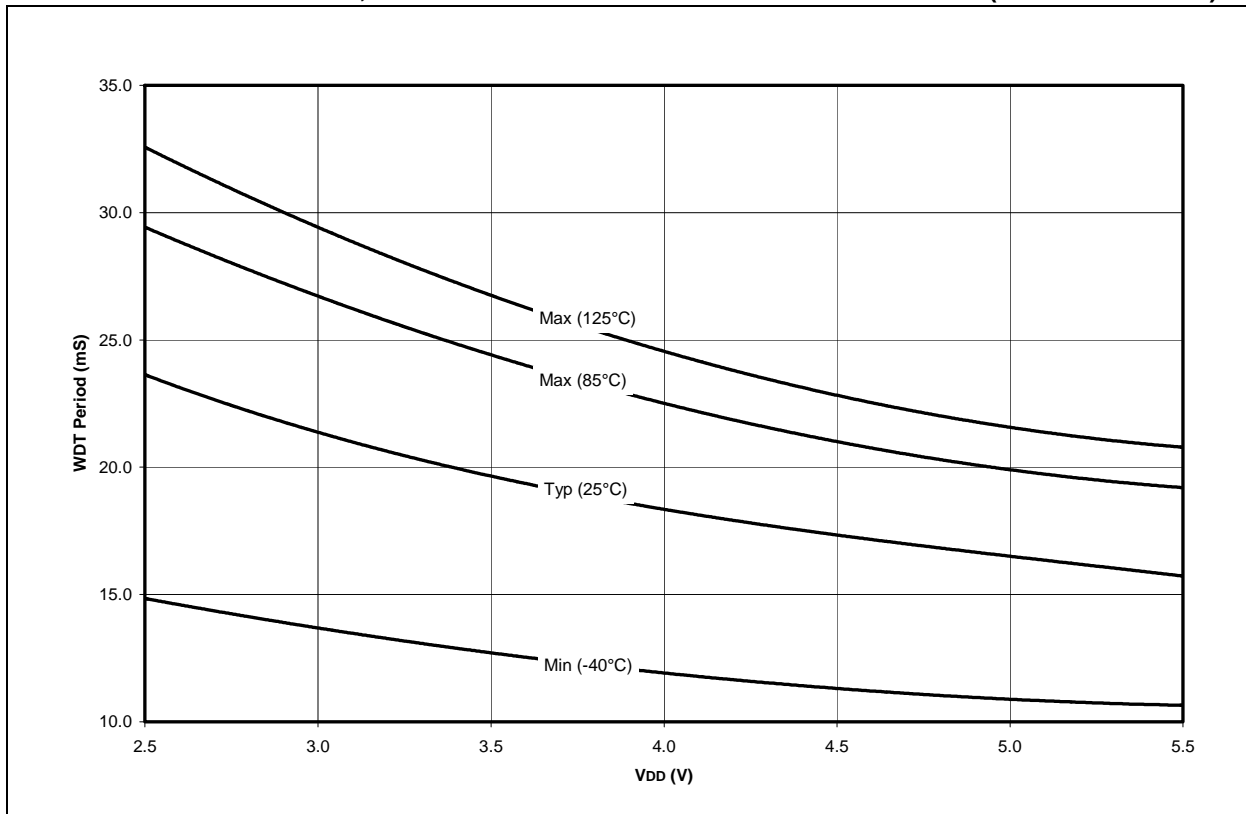


FIGURE 16-33: TYPICAL, MINIMUM AND MAXIMUM WDT PERIOD VS. V_{DD} (-40°C TO $+125^{\circ}\text{C}$)



17.0 PACKAGING INFORMATION

17.1 Package Marking Information

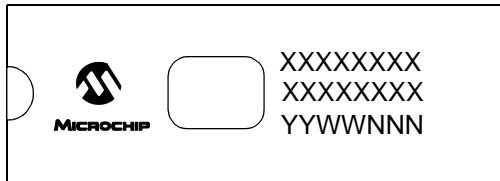
18-Lead PDIP



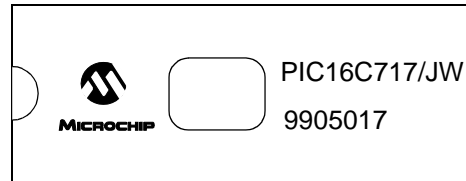
Example



18-Lead CERDIP Windowed



Example



18-Lead SOIC



Example



20-Lead PDIP



Example



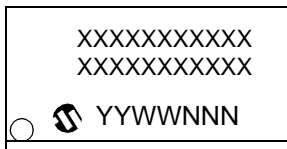
| | | |
|----------------|--------|--|
| Legend: | XX...X | Customer-specific information |
| | Y | Year code (last digit of calendar year) |
| | YY | Year code (last 2 digits of calendar year) |
| | WW | Week code (week of January 1 is week '01') |
| | NNN | Alphanumeric traceability code |
| | (e3) | Pb-free JEDEC designator for Matte Tin (Sn) |
| | * | This package is Pb-free. The Pb-free JEDEC designator (e3) can be found on the outer packaging for this package. |

Note: In the event the full Microchip part number cannot be marked on one line, it will be carried over to the next line, thus limiting the number of available characters for customer-specific information.

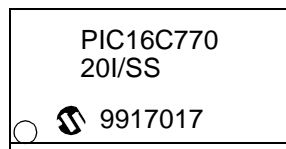
PIC16C717/770/771

17.1 Package Marking Information (Cont'd)

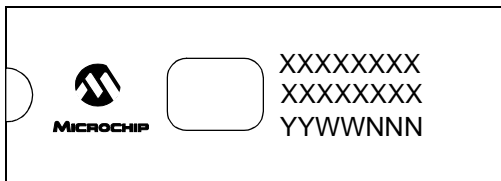
20-Lead SSOP



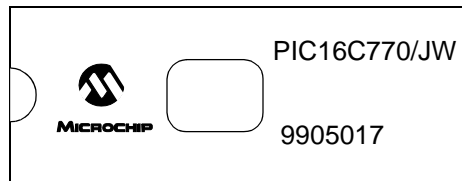
Example



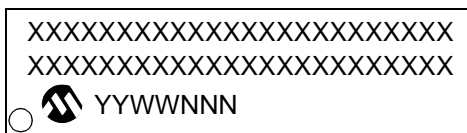
20-Lead CERDIP Windowed



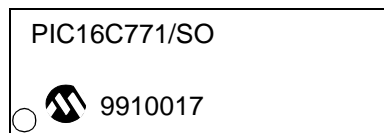
Example



20-Lead SOIC



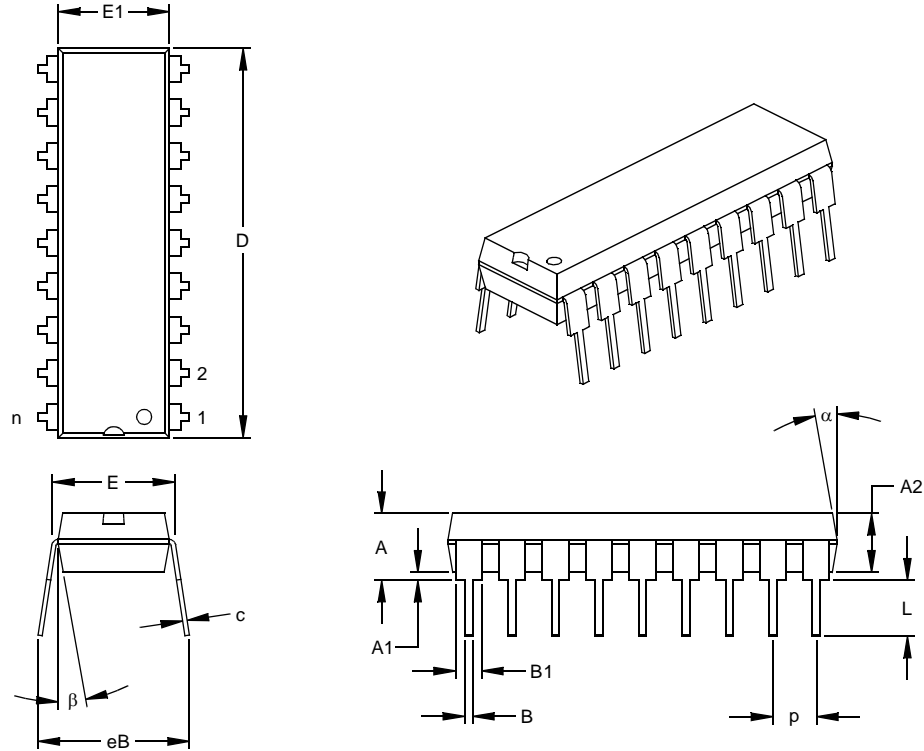
Example



PIC16C717/770/771

17.2 18-Lead Plastic Dual In-line (P) – 300 mil (PDIP)

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



| Dimension | Units | INCHES* | | | MILLIMETERS | | |
|----------------------------|-------|---------|------|------|-------------|-------|-------|
| | | 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 |

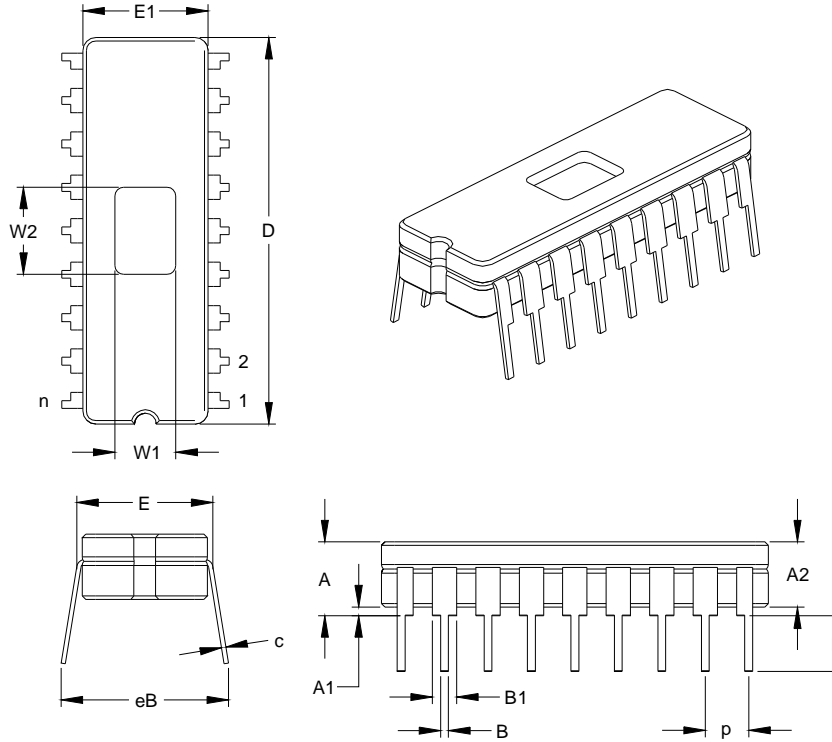
* Controlling Parameter
 § Significant Characteristic

Notes:
 Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.
 JEDEC Equivalent: MS-001
 Drawing No. C04-007

PIC16C717/770/771

17.3 18-Lead Ceramic Dual In-line with Window (JW) – 300 mil (CERDIP)

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



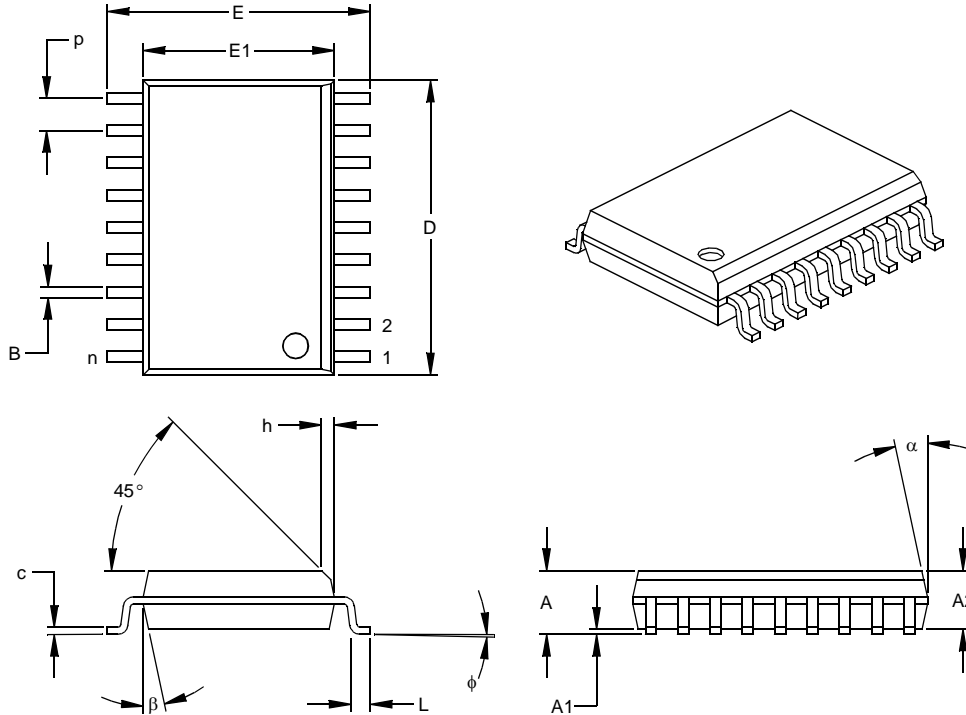
| Dimension Limits | Units | INCHES* | | | MILLIMETERS | | |
|----------------------------|-------|---------|------|------|-------------|-------|-------|
| | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 18 | | | 18 | |
| Pitch | P | | .100 | | | 2.54 | |
| Top to Seating Plane | A | .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

PIC16C717/770/771

17.4 18-Lead Plastic Small Outline (SO) – Wide, 300 mil (SOIC)

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



| Units | | INCHES* | | | MILLIMETERS | | |
|--------------------------|----|---------|------|------|-------------|-------|-------|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 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 |

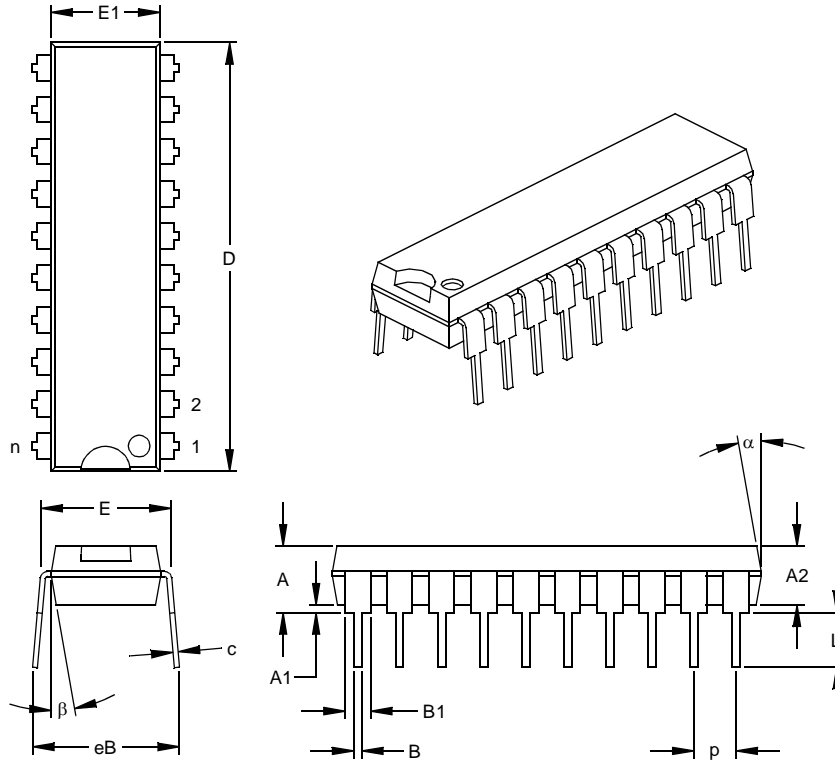
* Controlling Parameter
 § Significant Characteristic

Notes:
 Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.
 JEDEC Equivalent: MS-013
 Drawing No. C04-051

PIC16C717/770/771

17.5 20-Lead Plastic Dual In-line (P) – 300 mil (PDIP)

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



| Dimension Limits | Units | INCHES* | | | MILLIMETERS | | | |
|----------------------------|-------|---------|-------|-------|-------------|-------|-------|-------|
| | | 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
 § Significant Characteristic

Notes:

Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.

JEDEC Equivalent: MS-001

Drawing No. C04-019

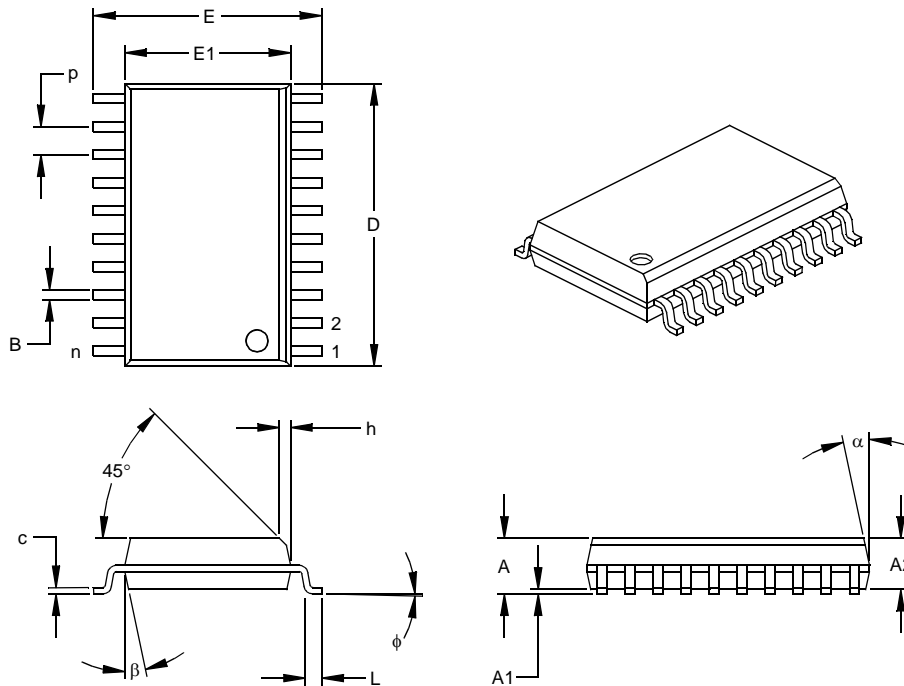
17.6 20-Lead Ceramic Dual In-line with Window (JW) – 300 mil (CERDIP)

DRAWING NOT AVAILABLE

PIC16C717/770/771

17.7 20-Lead Plastic Small Outline (SO) – Wide, 300 mi (SOIC)

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



| Units | | INCHES* | | | MILLIMETERS | | |
|--------------------------|----|---------|------|------|-------------|-------|-------|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 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

§ Significant Characteristic

Notes:

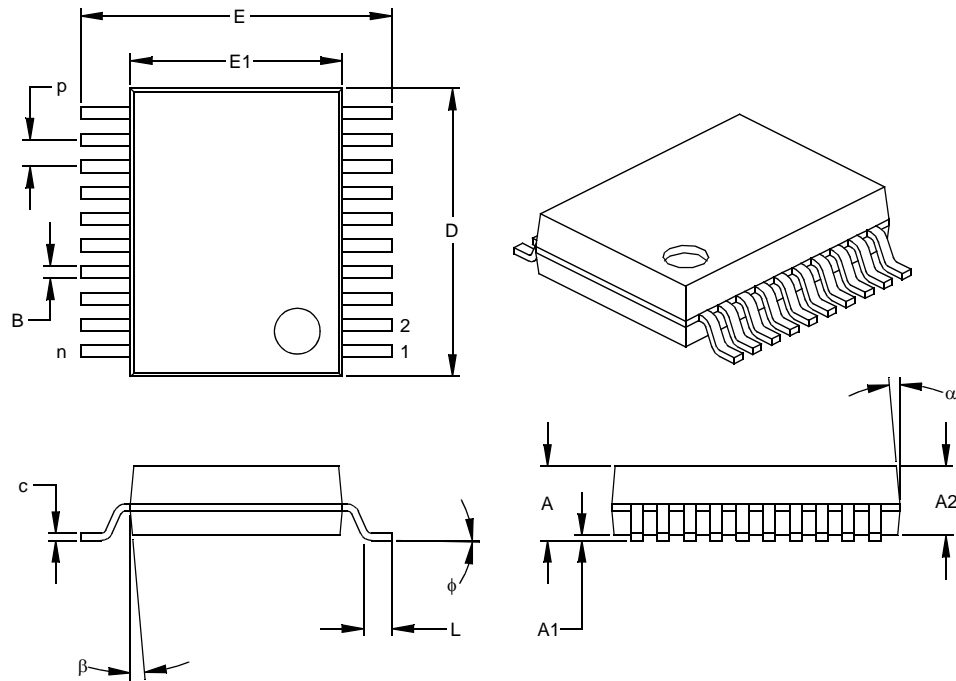
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.

JEDEC Equivalent: MS-013

Drawing No. C04-094

17.8 20-Lead Plastic Shrink Small Outline (SS) – 209 mil, 5.30 mm (SSOP)

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



| Dimension Limits | Units | INCHES* | | | MILLIMETERS | | |
|--------------------------|-------|---------|------|------|-------------|--------|--------|
| | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 20 | | | 20 | |
| Pitch | p | | .026 | | | 0.65 | |
| 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

§ Significant Characteristic

Notes:

Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" (0.254mm) per side.

JEDEC Equivalent: MO-150

Drawing No. C04-072

PIC16C717/770/771

NOTES:

APPENDIX A: REVISION HISTORY

| Version | Date | Revision Description |
|---------|----------|---|
| A | 09/14/99 | This is a new data sheet. However, the devices described in this data sheet are the upgrades to the devices found in the <i>PIC16C7X Data Sheet</i> , DS30390E. |
| B | 1/22/02 | Electrical Characteristics tables completed and characteristics graphs added. MSSP I ² C (Section 9.2) rewritten. General minor changes and corrections. |
| C | 1/28/13 | Added a note to each package outline drawing. |
| | | |

PIC16C717/770/771

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 AVDD and AVSS | 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 |

INDEX

A

| | |
|---|--------------------|
| A/D | 105 |
| A/D Converter Enable (ADIE Bit) | 17 |
| ADCON0 Register | 105 |
| ADCON1 Register | 105, 107 |
| ADRES Register | 105 |
| Block Diagram | 109 |
| Configuring Analog Port | 108 |
| Conversion time | 115 |
| Conversions | 111 |
| converter characteristics | 164, 165, 166, 170 |
| Faster Conversion - Lower Resolution Tradeoff | 115 |
| Internal Sampling Switch (Rss) Impedence | 113 |
| Operation During Sleep | 116 |
| Sampling Requirements | 113 |
| Sampling Time | 113 |
| Source Impedance | 113 |
| Special Event Trigger (ECCP) | 55 |
| A/D Conversion Clock | 110 |
| ACK | 77 |
| Acknowledge Data bit, AKD | 69 |
| Acknowledge Sequence Enable bit, AKE | 69 |
| Acknowledge Status bit, AKS | 69 |
| ACKSTAT | 87 |
| ADCON0 Register | 105 |
| ADCON1 Register | 105, 107 |
| ADRES | 105 |
| ADRES Register | 11, 12, 105, 116 |
| AKD | 69 |
| AKE | 69 |
| AKS | 69 |
| Analog-to-Digital Converter. <i>See</i> A/D | |
| Application Note AN578, "Use of the SSP Module in the I2C Multi-Master Environment." | 84 |
| Architecture | |
| PIC16C717/PIC16C717 Block Diagram | 5 |
| PIC16C770/771/PIC16C770/771 Block Diagram | 6 |
| Assembler | |
| MPASM Assembler | 141 |

B

| | |
|--|----------------|
| Banking, Data Memory | 9, 14 |
| Baud Rate Generator | 84 |
| BF | 66, 77, 87, 89 |
| Block Diagrams | |
| Baud Rate Generator | 84 |
| I ² C Master Mode | 83 |
| I ² C Module | 76 |
| RA3:RA0 and RA5 Port Pins | 26, 28, 29, 35 |
| SSP (I ² C Mode) | 76 |
| SSP (SPI Mode) | 70 |
| BOR. <i>See</i> Brown-out Reset | |
| BRG | 84 |
| Brown-out Reset (BOR) | 117, 123, 124 |
| Buffer Full bit, BF | 77 |
| Buffer Full Status bit, BF | 66 |
| Bus Arbitration | 94 |
| Bus Collision During a RESTART Condition | 97 |
| Bus Collision During a Start Condition | 95 |
| Bus Collision During a Stop Condition | 98 |
| Bus Collision Section | 94 |

C

| | |
|---|----------|
| Capture (ECCP Module) | 54 |
| Block Diagram | 54 |
| CCPR1H:CCPR1L Registers | 54 |
| Changing Between Capture Prescalers | 54 |
| ECCP Pin Configuration | 54 |
| Software Interrupt | 54 |
| Timer1 Mode Selection | 54 |
| Capture/Compare/PWM (ECCP) | |
| Capture Mode. <i>See</i> Capture | |
| Compare Mode. <i>See</i> Compare | |
| PWM Mode. <i>See</i> PWM | |
| CCP1CON | 13 |
| CCP2CON | 13 |
| CCPR1H Register | 11, 13 |
| CCPR1L Register | 13 |
| CCPR2H Register | 13 |
| CCPR2L Register | 13 |
| CKE | 66 |
| CKP | 67 |
| Clock Polarity Select bit, CKP | 67 |
| Code Examples | |
| Loading the SSPBUF register | 71 |
| Code Protection | 117, 131 |
| Compare (ECCP Module) | 54 |
| Block Diagram | 55 |
| CCPR1H:CCPR1L Registers | 54 |
| ECCP Pin Configuration | 54 |
| Software Interrupt | 55 |
| Special Event Trigger | 49, 55 |
| Timer1 Mode Selection | 54 |
| Configuration Bits | 117 |

D

| | |
|----------------------------------|---------------|
| D/A | 66 |
| Data Memory | 9 |
| Bank Select (RP Bits) | 9, 14 |
| General Purpose Registers | 9 |
| Register File Map | 10 |
| Special Function Registers | 11 |
| Data/Address bit, D/A | 66 |
| DC Characteristics | |
| PIC16C717/770/771 | 150, 151, 153 |
| Development Support | 141 |
| Device Differences | 208 |
| Direct Addressing | 23 |

E

| | |
|---------------------------------------|-----|
| Enhanced Capture/Compare/PWM (ECCP) | |
| CCP1 | |
| CCPR1H Register | 53 |
| CCPR1L Register | 53 |
| Enable (CCP1IE Bit) | 17 |
| Timer Resources | 54 |
| Errata | 3 |
| External Power-on Reset Circuit | 122 |

F

| | |
|-----------------------------|------------|
| Firmware Instructions | 133 |
| FSR Register | 11, 12, 13 |

G

| | |
|-------------------------------------|----|
| GCE | 69 |
| General Call Address Sequence | 82 |
| General Call Address Support | 82 |
| General Call Enable bit, GCE | 69 |

PIC16C717/770/771

| | |
|---|----------|
| I | |
| I/O Ports | 25 |
| I ² C | 76 |
| I ² C Master Mode Reception | 89 |
| I ² C Master Mode Restart Condition | 86 |
| I ² C Mode Selection | 76 |
| I ² C Module | |
| Acknowledge Sequence timing | 91 |
| Addressing | 77 |
| Baud Rate Generator | 84 |
| Block Diagram | 83 |
| BRG Block Diagram | 84 |
| BRG Reset due to SDA Collision | 96 |
| BRG Timing | 85 |
| Bus Arbitration | 94 |
| Bus Collision | 94 |
| Acknowledge | 94 |
| Restart Condition | 97 |
| Restart Condition Timing (Case1) | 97 |
| Restart Condition Timing (Case2) | 97 |
| Start Condition | 95 |
| Start Condition Timing | 95, 96 |
| Stop Condition | 98 |
| Stop Condition Timing (Case1) | 98 |
| Stop Condition Timing (Case2) | 98 |
| Transmit Timing | 94 |
| Bus Collision timing | 94 |
| Clock Arbitration | 93 |
| Clock Arbitration Timing (Master Transmit) | 93 |
| Conditions to not give ACK Pulse | 77 |
| General Call Address Support | 82 |
| Master Mode | 83 |
| Master Mode 7-bit Reception timing | 90 |
| Master Mode Operation | 84 |
| Master Mode Start Condition | 85 |
| Master Mode Transmission | 87 |
| Master Mode Transmit Sequence | 84 |
| Multi-Master Communication | 94 |
| Multi-master Mode | 84 |
| Operation | 76 |
| Repeat Start Condition timing | 86 |
| Slave Mode | 76 |
| Slave Reception | 78 |
| Slave Transmission | 80 |
| SSPBUF | 76 |
| Stop Condition Receive or Transmit timing | 92 |
| Stop Condition timing | 92 |
| Waveforms for 7-bit Reception | 78 |
| Waveforms for 7-bit Transmission | 80 |
| I ² C Slave Mode | 76 |
| ICEPIC In-Circuit Emulator | 142 |
| ID Locations | 117, 131 |
| In-Circuit Serial Programming (ICSP) | 117, 131 |
| INDF | 13 |
| INDF Register | 11, 12 |
| Indirect Addressing | 23 |
| FSR Register | 9 |
| Instruction Format | 133 |
| Instruction Set | 133 |
| ADDLW | 135 |
| ADDWF | 135 |
| ANDLW | 135 |
| ANDWF | 135 |
| BCF | 135 |
| BSF | 135 |
| BTFSC | 136 |
| BTFSS | 136 |
| CALL | 136 |
| CLRf | 136 |
| CLRWDT | 136 |
| COMF | 137 |
| DECF | 137 |
| DECFSZ | 137 |
| GOTO | 137 |
| INCF | 137 |
| INCFSZ | 137 |
| IORLW | 138 |
| IORWF | 138 |
| MOVF | 138 |
| MOVLW | 138 |
| MOVWF | 138 |
| NOP | 138 |
| RETFIE | 139 |
| RETLW | 139 |
| RETURN | 139 |
| RLF | 139 |
| RRF | 139 |
| SLEEP | 139 |
| SUBLW | 140 |
| SUBWF | 140 |
| SWAPF | 140 |
| XORLW | 140 |
| XORWF | 140 |
| Summary Table | 134 |
| INT Interrupt (RB0/INT). See Interrupt Sources | |
| INTCON | 13 |
| INTCON Register | 16 |
| GIE Bit | 16 |
| INTE Bit | 16 |
| INTF Bit | 16 |
| PEIE Bit | 16 |
| RBIE Bit | 16 |
| RBIF Bit | 16, 33 |
| TOIE Bit | 16 |
| TOIF Bit | 16 |
| Inter-Integrated Circuit (I ² C) | 65 |
| internal sampling switch (R _{ss}) impedance | 113 |
| Interrupt Sources | 117, 127 |
| Block Diagram | 127 |
| Capture Complete (ECCP) | 54 |
| Compare Complete (ECCP) | 55 |
| RB0/INT Pin, External | 128 |
| TMR0 Overflow | 46, 128 |
| TMR1 Overflow | 47, 49 |
| TMR2 to PR2 Match | 52 |
| TMR2 to PR2 Match (PWM) | 51, 56 |
| Interrupts | |
| Synchronous Serial Port Interrupt | 18 |
| Interrupts, Context Saving During | 128 |
| Interrupts, Enable Bits | |
| A/D Converter Enable (ADIE Bit) | 17 |
| CCP1 Enable (CCP1IE Bit) | 17, 54 |
| Global Interrupt Enable (GIE Bit) | 16, 127 |
| Interrupt-on-Change (RB7:RB4) Enable (RBIE Bit) | 16, 128 |
| Peripheral Interrupt Enable (PEIE Bit) | 16 |
| PSP Read/Write Enable (PSPIE Bit) | 17 |
| RB0/INT Enable (INTE Bit) | 16 |
| SSP Enable (SSPIE Bit) | 17 |
| TMR0 Overflow Enable (TOIE Bit) | 16 |
| TMR1 Overflow Enable (TMR1IE Bit) | 17 |

| | | | |
|--|---------------|---|-------------------------|
| TMR2 to PR2 Match Enable (TMR2IE Bit) | 17 | PICDEM 3 Low Cost PIC16CXXX | |
| USART Receive Enable (RCIE Bit) | 17, 18 | Demonstration Board..... | 144 |
| Interrupts, Flag Bits | | PICSTART Plus Entry Level | |
| CCP1 Flag (CCP1IF Bit)..... | 54 | Development Programmer..... | 143 |
| Interrupt on Change (RB7:RB4) Flag | | PIE1 Register | 17 |
| (RBIF Bit) | 16, 33, 128 | ADIE Bit..... | 17 |
| RB0/INT Flag (INTF Bit)..... | 16 | CCP1IE Bit | 17 |
| TMR0 Overflow Flag (T0IF Bit)..... | 16, 128 | PSPIE Bit..... | 17 |
| INTRC Mode | 120 | RCIE Bit..... | 17, 18 |
| K | | SSPIE Bit..... | 17 |
| KEELOQ Evaluation and Programming Tools | 144 | TMR1IE Bit | 17 |
| L | | TMR2IE Bit | 17 |
| LVDCON | 101 | PIE2 Register | 19 |
| M | | Pinout Descriptions | |
| Master Clear ($\overline{\text{MCLR}}$) | | PIC16C770 | 7 |
| $\overline{\text{MCLR}}$ Reset, Normal Operation | 121, 123, 124 | PIC16C770/771 | 7 |
| $\overline{\text{MCLR}}$ Reset, SLEEP | 121, 123, 124 | PIC16C771 | 7 |
| Memory Organization | | PIR1 Register | 18 |
| Data Memory | 9 | PIR2 Register | 20 |
| Program Memory | 9 | Pointer, FSR | 23 |
| MPLAB C17 and MPLAB C18 C Compilers..... | 141 | POR. See Power-on Reset | |
| MPLAB ICD In-Circuit Debugger | 143 | PORTA | 13 |
| MPLAB ICE High Performance Universal In-Circuit | | Initialization..... | 26 |
| Emulator with MPLAB IDE | 142 | PORTA Register..... | 25 |
| MPLAB Integrated Development Environment Software .. | 141 | TRISA Register..... | 25 |
| MPLINK Object Linker/MPLIB Object Librarian | 142 | PORTA Register..... | 11, 116 |
| Multi-Master Communication | 94 | PORTB | 13 |
| Multi-Master Mode | 84 | Initialization..... | 33 |
| O | | PORTB Register..... | 33 |
| OPCODE Field Descriptions | 133 | Pull-up Enable ($\overline{\text{RBPU}}$ Bit)..... | 15 |
| OPTION_REG Register | 15 | RB0/INT Edge Select (INTEDG Bit) | 15 |
| INTEDG Bit | 15 | RB0/INT Pin, External | 128 |
| PS Bits | 15, 45 | RB7:RB4 Interrupt on Change..... | 128 |
| PSA Bit..... | 15, 45 | RB7:RB4 Interrupt on Change Enable | |
| $\overline{\text{RBPU}}$ Bit..... | 15 | (RBIE Bit)..... | 16, 128 |
| T0CS Bit..... | 15, 45 | RB7:RB4 Interrupt on Change Flag | |
| T0SE Bit..... | 15, 45 | (RBIF Bit)..... | 16, 33, 128 |
| Oscillator Configuration..... | 119 | TRISB Register..... | 33 |
| CLKOUT | 120 | PORTB Register | 11, 116 |
| Dual Speed Operation for ER and | | Postscaler, Timer2 | |
| INTRC Modes | 120 | Select (TOUTPS Bits)..... | 51 |
| EC | 119, 123 | Postscaler, WDT | 45 |
| ER | 119, 123 | Assignment (PSA Bit) | 15, 45 |
| ER Mode | 120 | Block Diagram | 46 |
| HS | 119, 123 | Rate Select (PS Bits)..... | 15, 45 |
| INTRC | 119, 123 | Switching Between Timer0 and WDT | 46 |
| LP..... | 119, 123 | Power-down Mode. See SLEEP | |
| XT | 119, 123 | Power-on Reset (POR)..... | 117, 121, 122, 123, 124 |
| Oscillator, Timer1 | 47, 49 | Oscillator Start-up Timer (OST)..... | 117, 122 |
| Oscillator, WDT | 129 | Power Control (PCON) Register..... | 123 |
| P | | Power-down ($\overline{\text{PD}}$ Bit) | 14 |
| P | 66 | Power-on Reset Circuit, External | 122 |
| Packaging | 197 | Power-up Timer (PWRT) | 117, 122 |
| Paging, Program Memory | 9, 22 | Time-out ($\overline{\text{TO}}$ Bit) | 14 |
| Parallel Slave Port (PSP) | | Time-out Sequence | 123 |
| Read/Write Enable (PSPIE Bit)..... | 17 | Time-out Sequence on Power-up..... | 125, 126 |
| PCL Register..... | 11, 12 | PR2 Register | 12 |
| PCLATH Register | 11, 12, 13 | Prescaler, Capture | 54 |
| PCON Register | 21, 123 | Prescaler, Timer0 | 45 |
| PICDEM 1 Low Cost PIC MCU | | Assignment (PSA Bit) | 15, 45 |
| Demonstration Board..... | 143 | Block Diagram | 46 |
| PICDEM 17 Demonstration Board | 144 | Rate Select (PS Bits)..... | 15, 45 |
| PICDEM 2 Low Cost PIC16CXX | | Switching Between Timer0 and WDT | 46 |
| Demonstration Board..... | 143 | Prescaler, Timer1 | 48 |
| | | Select (T1CKPS Bits) | 47 |

PIC16C717/770/771

| | | | |
|--|----------|---|---------------|
| Prescaler, Timer2..... | 57 | S | |
| Select (T2CKPS Bits)..... | 51 | S..... | 66 |
| PRO MATE II Universal Device Programmer | 143 | SAE..... | 69 |
| Program Counter | | SCK..... | 70 |
| PCL Register..... | 22 | SCL..... | 76 |
| PCLATH Register | 22, 128 | SDA..... | 76 |
| Reset Conditions..... | 123 | SDI..... | 70 |
| Program Memory | 9 | SDO..... | 70 |
| Interrupt Vector | 9 | Serial Data In, SDI..... | 70 |
| Paging..... | 9, 22 | Serial Data Out, SDO | 70 |
| Program Memory Map | 9 | Slave Select Synchronization | 73 |
| READ (PMR)..... | 43 | Slave Select, SS..... | 70 |
| Reset Vector | 9 | SLEEP | 117, 121, 130 |
| Program Verification..... | 131 | SMP..... | 66 |
| Programmable Brown-out Reset (PBOR) | 121, 122 | Software Simulator (MPLAB SIM) | 142 |
| Programming, Device Instructions | 133 | SPE..... | 69 |
| PWM (CCP Module) | | Special Event Trigger. See Compare | |
| TMR2 to PR2 Match | 51 | Special Features of the CPU | 117 |
| TMR2 to PR2 Match Enable (TMR2IE Bit) | 17 | Special Function Registers | 11 |
| PWM (ECCP Module) | 56 | PIC16C717 | 11 |
| Block Diagram..... | 56 | PIC16C717/770/771 | 11 |
| CCPR1H:CCPR1L Registers..... | 56 | PIC16C770 | 11 |
| Duty Cycle..... | 57 | PIC16C771 | 11 |
| Output Diagram..... | 57 | Speed, Operating..... | 1 |
| Period..... | 56 | SPI | |
| TMR2 to PR2 Match | 56 | Master Mode..... | 72 |
| Q | | Serial Clock..... | 70 |
| Q Clock | 57 | Serial Data In..... | 70 |
| R | | Serial Data Out | 70 |
| R/W..... | 66 | Serial Peripheral Interface (SPI) | 65 |
| R/W bit | 80 | Slave Select..... | 70 |
| R/W bit | 78 | SPI clock..... | 72 |
| R/W bit | 77 | SPI Mode..... | 70 |
| RAM. See Data Memory | | SPI Clock Edge Select, CKE | 66 |
| RCE,Receive Enable bit, RCE | 69 | SPI Data Input Sample Phase Select, SMP | 66 |
| RCREG | 13 | SPI Master/Slave Connection | 71 |
| RCSTA Register..... | 13 | SPI Module | |
| Read/Write bit, R/W | 66 | Master/Slave Connection..... | 71 |
| Receive Overflow Indicator bit, SSPOV | 67 | Slave Mode..... | 73 |
| REFCON | 102 | Slave Select Synchronization | 73 |
| Register File | 9 | Slave Synch Timnig..... | 73 |
| Register File Map | 10 | SS | 70 |
| Registers | | SSP | 65 |
| FSR Summary | 13 | Block Diagram (SPI Mode) | 70 |
| INDF Summary | 13 | Enable (SSPIE Bit) | 17 |
| INTCON Summary..... | 13 | SPI Mode..... | 70 |
| PCL Summary..... | 13 | SSPADD..... | 77 |
| PCLATH Summary | 13 | SSPBUF | 72, 76 |
| PORTB Summary | 13 | SSPCON | 67 |
| SSPSTAT..... | 66, 101 | SSPCON2 | 69, 70 |
| STATUS Summary | 13 | SSPSR | 72, 77 |
| TMR0 Summary | 13 | SSPSTAT | 66, 76, 101 |
| TRISB Summary | 13 | TMR2 Output for Clock Shift..... | 51, 52 |
| Reset..... | 117, 121 | SSP I²C | |
| Block Diagram..... | 121 | SSP I ² C Operation | 76 |
| Brown-out Reset (BOR). See Brown-out Reset (BOR) | | SSP Module | |
| MCLR Reset. See MCLR | | SPI Master Mode..... | 72 |
| Power-on Reset (POR). See Power-on Reset (POR) | | SPI Master./Slave Connection..... | 71 |
| Reset Conditions for All Registers | 124 | SPI Slave Mode..... | 73 |
| Reset Conditions for PCON Register..... | 123 | SSPCON1 Register | 76 |
| Reset Conditions for Program Counter..... | 123 | SSP Overflow Detect bit, SSPOV..... | 77 |
| Reset Conditions for STATUS Register..... | 123 | SSPADD Register..... | 12 |
| WDT Reset. See Watchdog Timer (WDT) | | SSPBUF | 13, 76, 77 |
| Restart Condition Enabled bit, RSE | 69 | SSPBUF Register | 11 |
| Revision History | 207 | SSPCON..... | 67 |
| RSE..... | 69 | SSPCON Register | 11 |

| | | | |
|--|-------------|---|------------|
| SSPCON1 | 76 | Timer2 | |
| SSPCON2 | 69, 70 | Block Diagram | 52 |
| SSPEN | 67 | Postscaler. See Postscaler, Timer2 | |
| SSPIF | 18, 78 | PR2 Register | 51, 56 |
| SSPM | 68 | Prescaler. See Prescaler, Timer2 | |
| SSPOV | 67, 77, 89 | SSP Clock Shift | 51, 52 |
| SSPSTAT | 66, 76, 101 | T2CON Register | 51 |
| SSPSTAT Register | 12 | TMR2 Register | 51 |
| Stack | 22 | TMR2 to PR2 Match Enable (TMR2IE Bit) | 17 |
| Start bit (S) | 66 | TMR2 to PR2 Match Interrupt | 51, 52, 56 |
| Start Condition Enabled bit, SAE | 69 | Timing Diagrams | |
| STATUS Register | 14, 15, 128 | Acknowledge Sequence Timing | 91 |
| C Bit | 14 | Baud Rate Generator with Clock Arbitration | 85 |
| DC Bit | 14, 15 | BRG Reset Due to SDA Collision | 96 |
| IRP Bit | 14 | Brown-out Reset | 159 |
| \overline{PD} Bit | 14 | Bus Collision | |
| RP Bits | 14 | Start Condition Timing | 95 |
| \overline{TO} Bit | 14 | Bus Collision During a Restart Condition | |
| Z Bit | 14 | (Case 1) | 97 |
| Status Register | 14 | Bus Collision During a Restart Condition | |
| Stop bit (P) | 66 | (Case2) | 97 |
| Stop Condition Enable bit | 69 | Bus Collision During a Start Condition | |
| Synchronous Serial Port | 65 | (SCL = 0) | 96 |
| Synchronous Serial Port Enable bit, SSPEN | 67 | Bus Collision During a Stop Condition | 98 |
| Synchronous Serial Port Interrupt | 18 | Bus Collision for Transmit and Acknowledge | 94 |
| Synchronous Serial Port Mode Select bits, SSPM | 68 | Capture/Compare/PWM | 161 |
| T | | CLKOUT and I/O | 157 |
| T1CON | 13 | External Clock Timing | 157 |
| T1CON Register | 13, 47 | I ² C Bus Data | 177 |
| T1CKPS Bits | 47 | I ² C Master Mode First Start bit timing | 85 |
| T1OSCEN Bit | 47 | I ² C Master Mode Reception timing | 90 |
| $\overline{T1SYNC}$ Bit | 47 | I ² C Master Mode Transmission timing | 88 |
| TMR1CS Bit | 47 | Master Mode Transmit Clock Arbitration | 93 |
| TMR1ON Bit | 47 | Power-up Timer | 159 |
| T2CON Register | 13, 51 | Repeat Start Condition | 86 |
| T2CKPS Bits | 51 | Reset | 159 |
| TMR2ON Bit | 51 | Slave Synchronization | 73 |
| TOUTPS Bits | 51 | Start-up Timer | 159 |
| Timer0 | | Stop Condition Receive or Transmit | 92 |
| Block Diagram | 45 | Time-out Sequence on Power-up | 125, 126 |
| Clock Source Edge Select (T0SE Bit) | 15, 45 | Timer0 | 160 |
| Clock Source Select (T0CS Bit) | 15, 45 | Timer1 | 160 |
| Overflow Enable (T0IE Bit) | 16 | Wake-up from SLEEP via Interrupt | 131 |
| Overflow Flag (T0IF Bit) | 16, 128 | Watchdog Timer | 159 |
| Overflow Interrupt | 46, 128 | TMR0 | 13 |
| Prescaler. See Prescaler, Timer0 | | TMR0 Register | 11 |
| Timer1 | 47 | TMR1H | 13 |
| Block Diagram | 48 | TMR1H Register | 11 |
| Capacitor Selection | 49 | TMR1L | 13 |
| Clock Source Select (TMR1CS Bit) | 47 | TMR1L Register | 11 |
| External Clock Input Sync ($\overline{T1SYNC}$ Bit) | 47 | TMR2 | 13 |
| Module On/Off (TMR1ON Bit) | 47 | TMR2 Register | 11 |
| Oscillator | 47, 49 | TRISA Register | 12, 116 |
| Oscillator Enable (T1OSCEN Bit) | 47 | TRISB Register | 12, 116 |
| Overflow Enable (TMR1IE Bit) | 17 | TXREG | 13 |
| Overflow Interrupt | 47, 49 | U | |
| Prescaler. See Prescaler, Timer1 | | Update Address, UA | 66 |
| Special Event Trigger (ECCP) | 49, 55 | USART | |
| T1CON Register | 47 | Receive Enable (RCIE Bit) | 17, 18 |
| TMR1H Register | 47 | | |
| TMR1L Register | 47 | | |

W

| | |
|--|------------------------|
| W Register | 128 |
| Wake-up from SLEEP | 117, 130 |
| Interrupts | 123, 124 |
| MCLR Reset | 124 |
| Timing Diagram | 131 |
| WDT Reset | 124 |
| Watchdog Timer (WDT) | 117, 129 |
| Block Diagram | 129 |
| Enable (WDTE Bit) | 129 |
| Postscaler. See Postscaler, WDT | |
| Programming Considerations | 129 |
| RC Oscillator | 129 |
| Time-out Period | 129 |
| WDT Reset, Normal Operation | 121, 123, 124 |
| WDT Reset, SLEEP | 123, 124 |
| Waveform for General Call Address Sequence | 82 |
| WCOL | 67, 85, 87, 89, 91, 92 |
| WCOL Status Flag | 85 |
| Write Collision Detect bit, WCOL | 67 |
| WWW, On-Line Support | 3 |

THE MICROCHIP WEB SITE

Microchip provides online support via our WWW site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: <http://microchip.com/support>

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) 792-4150.

Please list the following information, and use this outline to provide us with your comments about this document.

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:

Literature Number: DS41120C

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 document easy to follow? If not, why?

4. What additions to the document do you think would enhance the structure and subject?

5. What deletions from the document 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?

PIC16C717/770/771

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.

| <u>PART NO.</u> | <u>X</u> | <u>/XX</u> | <u>XXX</u> |
|--------------------|--|------------|------------|
| Device | Temperature Range | Package | Pattern |
| 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) | | |
| Temperature Range: | - = 0°C to +70°C I = -40°C to +85°C E = -40°C to +125°C | | |
| Package | JW = Windowed CERDIP SO = SOIC P = PDIP SS = SSOP | | |
| Pattern | QTP, SQTP, Code or Special Requirements. Blank for OTP and Windowed devices. | | |

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.

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 Worldwide Site (www.microchip.com)

PIC16C717/770/771

NOTES:

NOTES:

PIC16C717/770/771

NOTES:

NOTES:

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

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

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

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights.

Trademarks

The Microchip name and logo, the Microchip logo, dsPIC, FlashFlex, KEELOQ, KEELOQ logo, MPLAB, PIC, PICmicro, PICSTART, PIC³² logo, rPIC, SST, SST Logo, SuperFlash and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MTP, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Silicon Storage Technology is a registered trademark of Microchip Technology Inc. in other countries.

Analog-for-the-Digital Age, Application Maestro, BodyCom, chipKIT, chipKIT logo, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, HI-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPASM, MPF, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Omniclient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICtail, REAL ICE, rLAB, Select Mode, SQI, Serial Quad I/O, Total Endurance, TSHARC, UniWinDriver, WiperLock, ZENA and Z-Scale are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

GestIC and ULPP are registered trademarks of Microchip Technology Germany II GmbH & Co. & KG, a subsidiary of Microchip Technology Inc., in other countries.

All other trademarks mentioned herein are property of their respective companies.

© 1999-2013, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 Printed on recycled paper.

ISBN: 9781620769713

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

**QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
= ISO/TS 16949 =**



MICROCHIP

Worldwide Sales and Service

AMERICAS

Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200
Fax: 480-792-7277
Technical Support:
<http://www.microchip.com/support>
Web Address:
www.microchip.com

Atlanta
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

Boston
Westborough, MA
Tel: 774-760-0087
Fax: 774-760-0088

Chicago
Itasca, IL
Tel: 630-285-0071
Fax: 630-285-0075

Cleveland
Independence, OH
Tel: 216-447-0464
Fax: 216-447-0643

Dallas
Addison, TX
Tel: 972-818-7423
Fax: 972-818-2924

Detroit
Farmington Hills, MI
Tel: 248-538-2250
Fax: 248-538-2260

Indianapolis
Noblesville, IN
Tel: 317-773-8323
Fax: 317-773-5453

Los Angeles
Mission Viejo, CA
Tel: 949-462-9523
Fax: 949-462-9608

Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

Toronto
Mississauga, Ontario,
Canada
Tel: 905-673-0699
Fax: 905-673-6509

ASIA/PACIFIC

Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

China - Beijing
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

China - Chengdu
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

China - Chongqing
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

China - Hangzhou
Tel: 86-571-2819-3187
Fax: 86-571-2819-3189

China - Hong Kong SAR
Tel: 852-2943-5100
Fax: 852-2401-3431

China - Nanjing
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

China - Qingdao
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

China - Shanghai
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

China - Shenyang
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

China - Shenzhen
Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

China - Wuhan
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

China - Xian
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

China - Xiamen
Tel: 86-592-2388138
Fax: 86-592-2388130

China - Zhuhai
Tel: 86-756-3210040
Fax: 86-756-3210049

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

India - Pune
Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

Japan - Osaka
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

Japan - Tokyo
Tel: 81-3-6880-3770
Fax: 81-3-6880-3771

Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 or
82-2-558-5934

Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

Taiwan - Hsin Chu
Tel: 886-3-5778-366
Fax: 886-3-5770-955

Taiwan - Kaohsiung
Tel: 886-7-213-7828
Fax: 886-7-330-9305

Taiwan - Taipei
Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

EUROPE

Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

UK - Wokingham
Tel: 44-118-921-5869
Fax: 44-118-921-5820

11/29/12

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

Microchip:

[PIC16C717-E/SS](#) [PIC16C717-E/SO](#) [PIC16LC770T/SO](#) [PIC16LC770T/SS](#) [PIC16C717-E/P](#) [PIC16LC770/P](#)
[PIC16C770T-E/SO](#) [PIC16C771T-E/SS](#) [PIC16C717T-E/SS](#) [PIC16C771T-E/SO](#) [PIC16C770T-E/SS](#) [PIC16C717T-](#)
[E/SO](#) [PIC16LC771-E/P](#) [PIC16C717T-I/SO](#) [PIC16LC771T/SS](#) [PIC16LC771T/SO](#) [PIC16LC771/P](#) [PIC16C717-I/SO](#)
[PIC16LC717-E/P](#) [PIC16LC717-I/P](#) [PIC16LC770T-E/SS](#) [PIC16LC771T-E/SO](#) [PIC16C771T/SO](#) [PIC16LC717T-E/SO](#)
[PIC16LC771T-E/SS](#) [PIC16C771T/SS](#) [PIC16LC717T-E/SS](#) [PIC16LC770T-E/SO](#) [PIC16LC717-E/SS](#) [PIC16LC717-](#)
[E/SO](#) [PIC16LC771-E/SO](#) [PIC16LC771-E/SS](#) [PIC16C770-I/SS](#) [PIC16C770-E/P](#) [PIC16C770-I/SO](#) [PIC16LC717T/SO](#)
[PIC16LC717T/SS](#) [PIC16LC770-I/SO](#) [PIC16LC770-I/SS](#) [PIC16C770/P](#) [PIC16LC770-E/SS](#) [PIC16LC770-E/SO](#)
[PIC16C771-I/SS](#) [PIC16C771-I/SO](#) [PIC16LC770-E/P](#) [PIC16C770/JW](#) [PIC16C771/SS](#) [PIC16C771/SO](#)
[PIC16C771/JW](#) [PIC16C771-E/P](#) [PIC16C771-I/P](#) [PIC16C717T/SO](#) [PIC16LC771-I/SS](#) [PIC16C717T/SS](#) [PIC16LC771-](#)
[I/SO](#) [PIC16LC771T-I/SS](#) [PIC16LC770T-I/SO](#) [PIC16LC770T-I/SS](#) [PIC16LC771T-I/SO](#) [PIC16LC717T-I/SS](#)
[PIC16LC717T-I/SO](#) [PIC16LC771/SO](#) [PIC16LC771/SS](#) [PIC16C771-E/SO](#) [PIC16C771-E/SS](#) [PIC16C770T/SO](#)
[PIC16C770-I/P](#) [PIC16C770T/SS](#) [PIC16C717T-I/SS](#) [PIC16LC771-I/P](#) [PIC16LC717-I/SO](#) [PIC16LC717-I/SS](#)
[PIC16C771T-I/SS](#) [PIC16C771T-I/SO](#) [PIC16C770T-I/SO](#) [PIC16C770T-I/SS](#) [PIC16LC717/SS](#) [PIC16LC717/SO](#)
[PIC16LC770-I/P](#) [PIC16C770/SO](#) [PIC16C770/SS](#) [PIC16LC717/P](#) [PIC16LC770/SS](#) [PIC16LC770/SO](#) [PIC16C717-I/P](#)
[PIC16C770-E/SS](#) [PIC16C770-E/SO](#) [PIC16C717-I/SS](#) [PIC16C717/JW](#) [PIC16C717/SO](#) [PIC16C717/SS](#) [PIC16C771/P](#)
[PIC16C717/P](#)