



PIC16C71X

8-Bit CMOS Microcontrollers with A/D Converter

Devices included in this data sheet:

- PIC16C710
- PIC16C71
- PIC16C711
- PIC16C715

PIC16C71X Microcontroller Core Features:

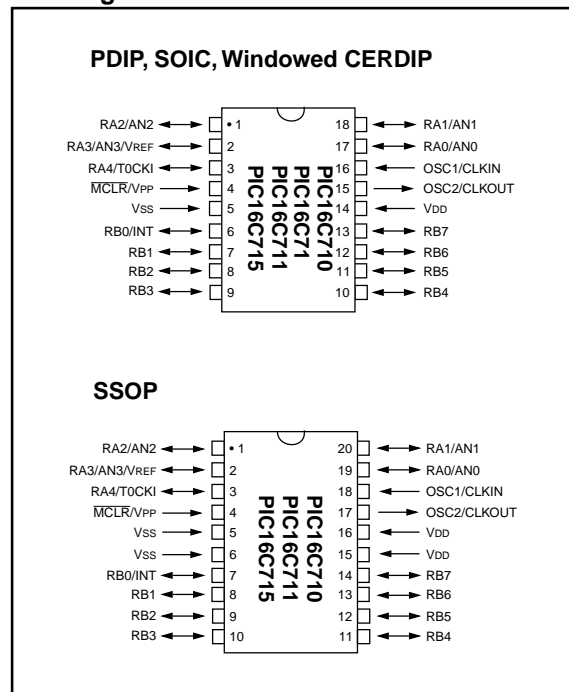
- High-performance RISC CPU
- Only 35 single word instructions to learn
- All single cycle instructions except for program branches which are two cycle
- Operating speed: DC - 20 MHz clock input
DC - 200 ns instruction cycle
- Up to 2K x 14 words of Program Memory, up to 128 x 8 bytes of Data Memory (RAM)
- Interrupt capability
- 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
- Programmable code-protection
- Power saving SLEEP mode
- Selectable oscillator options
- Low-power, high-speed CMOS EPROM technology
- Fully static design
- Wide operating voltage range: 2.5V to 6.0V
- High Sink/Source Current 25/25 mA
- Commercial, Industrial and Extended temperature ranges
- Program Memory Parity Error Checking Circuitry with Parity Error Reset (PER) (PIC16C715)
- Low-power consumption:
 - < 2 mA @ 5V, 4 MHz
 - 15 μ A typical @ 3V, 32 kHz
 - < 1 μ A typical standby current

PIC16C71X Peripheral Features:

- Timer0: 8-bit timer/counter with 8-bit prescaler
- 8-bit multichannel analog-to-digital converter
- Brown-out detection circuitry for Brown-out Reset (BOR)
- 13 I/O Pins with Individual Direction Control

| PIC16C7X Features | 710 | 71 | 711 | 715 |
|-------------------------------|-----|-----|-----|-----|
| Program Memory (EPROM) x 14 | 512 | 1K | 1K | 2K |
| Data Memory (Bytes) x 8 | 36 | 36 | 68 | 128 |
| I/O Pins | 13 | 13 | 13 | 13 |
| Timer Modules | 1 | 1 | 1 | 1 |
| A/D Channels | 4 | 4 | 4 | 4 |
| In-Circuit Serial Programming | Yes | Yes | Yes | Yes |
| Brown-out Reset | Yes | — | Yes | Yes |
| Interrupt Sources | 4 | 4 | 4 | 4 |

Pin Diagrams



PIC16C71X

Table of Contents

| | | |
|---|---|-----|
| 1.0 | General Description | 3 |
| 2.0 | PIC16C71X Device Varieties | 5 |
| 3.0 | Architectural Overview | 7 |
| 4.0 | Memory Organization | 11 |
| 5.0 | I/O Ports | 25 |
| 6.0 | Timer0 Module | 31 |
| 7.0 | Analog-to-Digital Converter (A/D) Module | 37 |
| 8.0 | Special Features of the CPU | 47 |
| 9.0 | Instruction Set Summary | 69 |
| 10.0 | Development Support | 85 |
| 11.0 | Electrical Characteristics for PIC16C710 and PIC16C711 | 89 |
| 12.0 | DC and AC Characteristics Graphs and Tables for PIC16C710 and PIC16C711 | 101 |
| 13.0 | Electrical Characteristics for PIC16C715 | 111 |
| 14.0 | DC and AC Characteristics Graphs and Tables for PIC16C715 | 125 |
| 15.0 | Electrical Characteristics for PIC16C71 | 135 |
| 16.0 | DC and AC Characteristics Graphs and Tables for PIC16C71 | 147 |
| 17.0 | Packaging Information | 155 |
| Appendix A: | | 161 |
| Appendix B: | Compatibility | 161 |
| Appendix C: | What's New | 162 |
| Appendix D: | What's Changed | 162 |
| Index | | 163 |
| PIC16C71X Product Identification System | | 173 |

To Our Valued Customers

We constantly strive to improve the quality of all our products and documentation. We have spent an exceptional amount of time to ensure that these documents are correct. However, we realize that we may have missed a few things. If you find any information that is missing or appears in error, please use the reader response form in the back of this data sheet to inform us. We appreciate your assistance in making this a better document.

1.0 GENERAL DESCRIPTION

The PIC16C71X is a family of low-cost, high-performance, CMOS, fully-static, 8-bit microcontrollers with integrated analog-to-digital (A/D) converters, in the PIC16CXX mid-range family.

All PIC16/17 microcontrollers employ an advanced RISC architecture. The PIC16CXX microcontroller family has enhanced core features, eight-level deep stack, and multiple internal and external interrupt sources. The separate instruction and data buses of the Harvard architecture allow a 14-bit wide instruction word with the separate 8-bit wide data. The two stage instruction pipeline allows all instructions to execute in a single cycle, except for program branches which require two cycles. A total of 35 instructions (reduced instruction set) are available. Additionally, a large register set gives some of the architectural innovations used to achieve a very high performance.

PIC16CXX microcontrollers typically achieve a 2:1 code compression and a 4:1 speed improvement over other 8-bit microcontrollers in their class.

The **PIC16C710/71** devices have 36 bytes of RAM, the **PIC16C711** has 68 bytes of RAM and the **PIC16C715** has 128 bytes of RAM. Each device has 13 I/O pins. In addition a timer/counter is available. Also a 4-channel high-speed 8-bit A/D is provided. The 8-bit resolution is ideally suited for applications requiring low-cost analog interface, e.g. thermostat control, pressure sensing, etc.

The PIC16C71X family has special features to reduce external components, thus reducing cost, enhancing system reliability and reducing power consumption. There are four oscillator options, of which the single pin RC oscillator provides a low-cost solution, the LP oscillator minimizes power consumption, XT is a standard crystal, and the HS is for High Speed crystals. The SLEEP (power-down) feature provides a power saving mode. The user can wake up the chip from SLEEP through several external and internal interrupts and resets.

A highly reliable Watchdog Timer with its own on-chip RC oscillator provides protection against software lock-up.

A UV erasable Cerdip packaged version is ideal for code development while the cost-effective One-Time-Programmable (OTP) version is suitable for production in any volume.

The PIC16C71X family fits perfectly in applications ranging from security and remote sensors to appliance control and automotive. The EPROM technology makes customization of application programs (transmitter codes, motor speeds, receiver frequencies, etc.) extremely fast and convenient. The small footprint packages make this microcontroller series perfect for all applications with space limitations. Low cost, low power, high performance, ease of use and I/O flexibility make the PIC16C71X very versatile even in areas where no microcontroller use has been considered before (e.g. timer functions, serial communication, capture and compare, PWM functions and coprocessor applications).

1.1 Family and Upward Compatibility

Users familiar with the PIC16C5X microcontroller family will realize that this is an enhanced version of the PIC16C5X architecture. Please refer to Appendix A for a detailed list of enhancements. Code written for the PIC16C5X can be easily ported to the PIC16CXX family of devices (Appendix B).

1.2 Development Support

PIC16C71X devices are supported by the complete line of Microchip Development tools.

Please refer to Section 10.0 for more details about Microchip's development tools.

PIC16C71X

TABLE 1-1: PIC16C71X FAMILY OF DEVICES

| | | PIC16C710 | PIC16C71 | PIC16C711 | PIC16C715 | PIC16C72 | PIC16CR72 ⁽¹⁾ |
|--------------------|--|-------------------------------|------------------|-------------------------------|-------------------------------|-------------------------|--------------------------|
| Clock | Maximum Frequency of Operation (MHz) | 20 | 20 | 20 | 20 | 20 | 20 |
| | EPROM Program Memory (x14 words) | 512 | 1K | 1K | 2K | 2K | — |
| Memory | ROM Program Memory (14K words) | — | — | — | — | — | 2K |
| | Data Memory (bytes) | 36 | 36 | 68 | 128 | 128 | 128 |
| Peripherals | Timer Module(s) | TMR0 | TMR0 | TMR0 | TMR0 | TMR0, TMR1, TMR2 | TMR0, TMR1, TMR2 |
| | Capture/Compare/PWM Module(s) | — | — | — | — | 1 | 1 |
| | Serial Port(s) (SPI/I ² C, USART) | — | — | — | — | SPI/I ² C | SPI/I ² C |
| | Parallel Slave Port | — | — | — | — | — | — |
| | A/D Converter (8-bit) Channels | 4 | 4 | 4 | 4 | 5 | 5 |
| Features | Interrupt Sources | 4 | 4 | 4 | 4 | 8 | 8 |
| | I/O Pins | 13 | 13 | 13 | 13 | 22 | 22 |
| | Voltage Range (Volts) | 2.5-6.0 | 3.0-6.0 | 2.5-6.0 | 2.5-5.5 | 2.5-6.0 | 3.0-5.5 |
| | In-Circuit Serial Programming | Yes | Yes | Yes | Yes | Yes | Yes |
| | Brown-out Reset | Yes | — | Yes | Yes | Yes | Yes |
| | Packages | 18-pin DIP, SOIC, 20-pin SSOP | 18-pin DIP, SOIC | 18-pin DIP, SOIC, 20-pin SSOP | 18-pin DIP, SOIC, 20-pin SSOP | 28-pin SDIP, SOIC, SSOP | 28-pin SDIP, SOIC, SSOP |

| | | PIC16C73A | PIC16C74A | PIC16C76 | PIC16C77 |
|--------------------|--|-----------------------------|-------------------------------------|-----------------------------|-------------------------------------|
| Clock | Maximum Frequency of Operation (MHz) | 20 | 20 | 20 | 20 |
| | EPROM Program Memory (x14 words) | 4K | 4K | 8K | 8K |
| Memory | Data Memory (bytes) | 192 | 192 | 376 | 376 |
| | Timer Module(s) | TMR0, TMR1, TMR2 | TMR0, TMR1, TMR2 | TMR0, TMR1, TMR2 | TMR0, TMR1, TMR2 |
| Peripherals | Capture/Compare/PWM Module(s) | 2 | 2 | 2 | 2 |
| | Serial Port(s) (SPI/I ² C, USART) | SPI/I ² C, USART | SPI/I ² C, USART | SPI/I ² C, USART | SPI/I ² C, USART |
| | Parallel Slave Port | — | Yes | — | Yes |
| | A/D Converter (8-bit) Channels | 5 | 8 | 5 | 8 |
| | Interrupt Sources | 11 | 12 | 11 | 12 |
| Features | I/O Pins | 22 | 33 | 22 | 33 |
| | Voltage Range (Volts) | 2.5-6.0 | 2.5-6.0 | 2.5-6.0 | 2.5-6.0 |
| | In-Circuit Serial Programming | Yes | Yes | Yes | Yes |
| | Brown-out Reset | Yes | Yes | Yes | Yes |
| | Packages | 28-pin SDIP, SOIC | 40-pin DIP; 44-pin PLCC, MQFP, TQFP | 28-pin SDIP, SOIC | 40-pin DIP; 44-pin PLCC, MQFP, TQFP |
| | | | | | |

All PIC16/17 Family devices have Power-on Reset, selectable Watchdog Timer, selectable code protect and high I/O current capability. All PIC16C7XX Family devices use serial programming with clock pin RB6 and data pin RB7.

Note 1: Please contact your local Microchip sales office for availability of these devices.

2.0 PIC16C71X DEVICE VARIETIES

A variety of frequency ranges and packaging options are available. Depending on application and production requirements, the proper device option can be selected using the information in the PIC16C71X Product Identification System section at the end of this data sheet. When placing orders, please use that page of the data sheet to specify the correct part number.

For the PIC16C71X family, there are two device "types" as indicated in the device number:

1. **C**, as in PIC16**C**71. These devices have EPROM type memory and operate over the standard voltage range.
2. **LC**, as in PIC16**LC**71. These devices have EPROM type memory and operate over an extended voltage range.

2.1 UV Erasable Devices

The UV erasable version, offered in Cerdip package is optimal for prototype development and pilot programs. This version can be erased and reprogrammed to any of the oscillator modes.

Microchip's PICSTART[®] Plus and PRO MATE[®] II programmers both support programming of the PIC16C71X.

2.2 One-Time-Programmable (OTP) Devices

The availability of OTP devices is especially useful for customers who need the flexibility for frequent code updates and small volume applications.

The OTP devices, packaged in plastic packages, permit the user to program them once. In addition to the program memory, the configuration bits must also be programmed.

2.3 Quick-Turnaround-Production (QTP) Devices

Microchip offers a QTP Programming Service for factory production orders. This service is made available for users who choose not to program a medium to high quantity of units and whose code patterns have stabilized. The devices are identical to the OTP devices but with all EPROM locations and configuration options already programmed by the factory. Certain code and prototype verification procedures apply before production shipments are available. Please contact your local Microchip Technology sales office for more details.

2.4 Serialized Quick-Turnaround Production (SQTPSM) Devices

Microchip offers a unique programming service where a few user-defined locations in each device are programmed with different serial numbers. The serial numbers may be random, pseudo-random, or sequential.

Serial programming allows each device to have a unique number which can serve as an entry-code, password, or ID number.

PIC16C71X

NOTES:

3.0 ARCHITECTURAL OVERVIEW

The high performance of the PIC16CXX family can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC16CXX uses a Harvard architecture, in which, program and data are accessed from separate memories using separate buses. This improves bandwidth over traditional von Neumann architecture in which program and data are fetched from the same memory using the same bus. Separating program and data buses further allows instructions to be sized differently than the 8-bit wide data word. Instruction opcodes are 14-bits wide making it possible to have all single word instructions. A 14-bit wide program memory access bus fetches a 14-bit instruction in a single cycle. A two-stage pipeline overlaps fetch and execution of instructions (Example 3-1). Consequently, all instructions (35) execute in a single cycle (200 ns @ 20 MHz) except for program branches.

The table below lists program memory (EPROM) and data memory (RAM) for each PIC16C71X device.

| Device | Program Memory | Data Memory |
|-----------|----------------|-------------|
| PIC16C710 | 512 x 14 | 36 x 8 |
| PIC16C71 | 1K x 14 | 36 x 8 |
| PIC16C711 | 1K x 14 | 68 x 8 |
| PIC16C715 | 2K x 14 | 128 x 8 |

The PIC16CXX can directly or indirectly address its register files or data memory. All special function registers, including the program counter, are mapped in the data memory. The PIC16CXX has an orthogonal (symmetrical) instruction set that makes it possible to carry out any operation on any register using any addressing mode. This symmetrical nature and lack of 'special optimal situations' make programming with the PIC16CXX simple yet efficient. In addition, the learning curve is reduced significantly.

PIC16CXX devices contain an 8-bit ALU and working register. The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between the data in the working register and any register file.

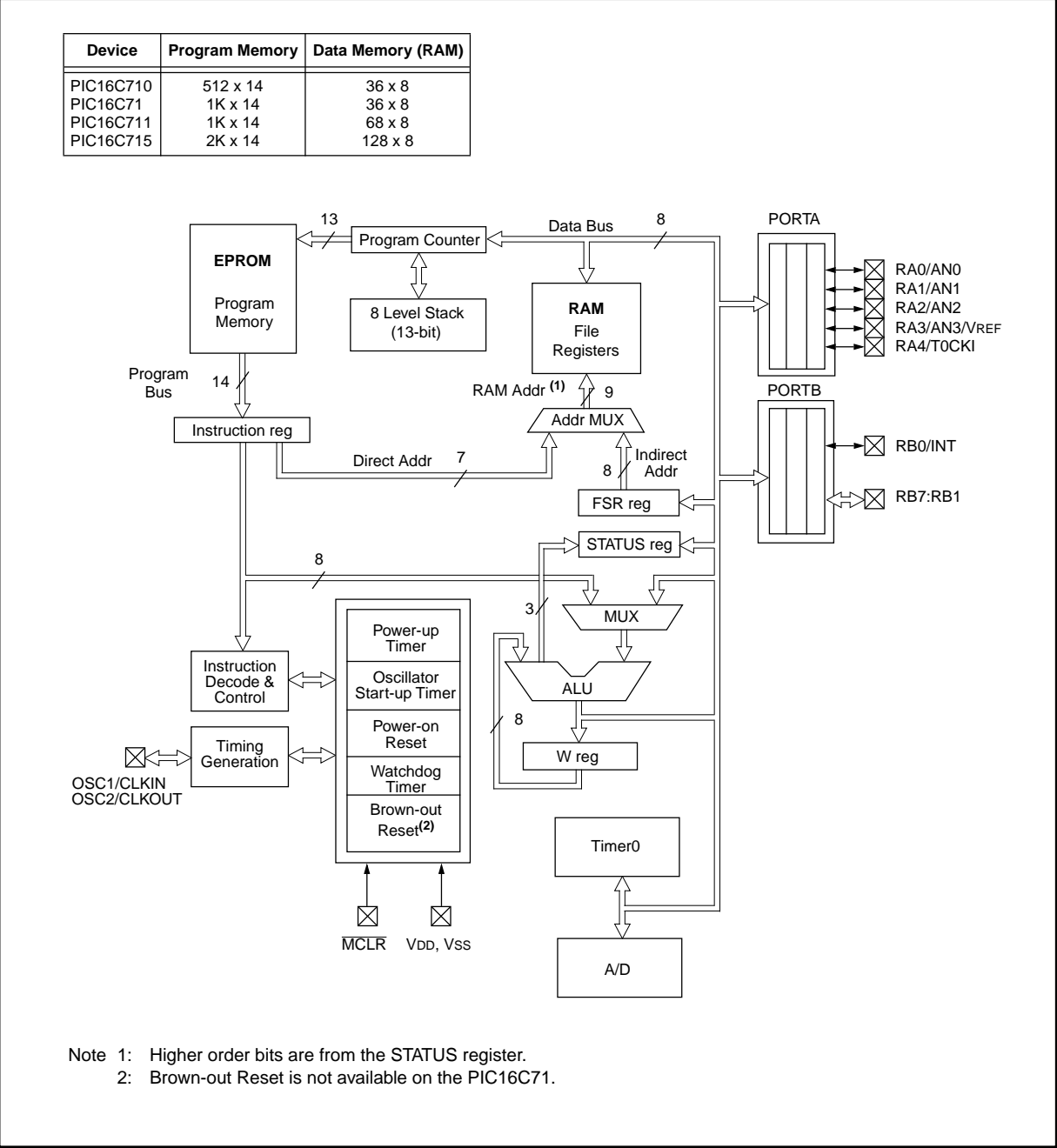
The ALU is 8-bits wide and capable of addition, subtraction, shift and logical operations. Unless otherwise mentioned, arithmetic operations are two's complement in nature. In two-operand instructions, typically one operand is the working register (W register). The other operand is a file register or an immediate constant. In single operand instructions, the operand is either the W register or a file register.

The W register is an 8-bit working register used for ALU operations. It is not an addressable register.

Depending on the instruction executed, the ALU may affect the values of the Carry (C), Digit Carry (DC), and Zero (Z) bits in the STATUS register. The C and DC bits operate as a borrow bit and a digit borrow out bit, respectively, in subtraction. See the `SUBLW` and `SUBWF` instructions for examples.

PIC16C71X

FIGURE 3-1: PIC16C71X BLOCK DIAGRAM



PIC16C71X

TABLE 3-1: PIC16C710/711/715 PINOUT DESCRIPTION

| Pin Name | DIP Pin# | SSOP Pin# ⁽⁴⁾ | SOIC Pin# | I/O/P Type | Buffer Type | Description |
|--------------|----------|--------------------------|-----------|------------|------------------------|---|
| OSC1/CLKIN | 16 | 18 | 16 | I | ST/CMOS ⁽³⁾ | Oscillator crystal input/external clock source input. |
| OSC2/CLKOUT | 15 | 17 | 15 | O | — | Oscillator crystal output. Connects to crystal or resonator in crystal oscillator mode. In RC mode, OSC2 pin outputs CLKOUT which has 1/4 the frequency of OSC1, and denotes the instruction cycle rate. |
| MCLR/VPP | 4 | 4 | 4 | I/P | ST | Master clear (reset) input or programming voltage input. This pin is an active low reset to the device. |
| RA0/AN0 | 17 | 19 | 17 | I/O | TTL | PORTA is a bi-directional I/O port. RA0 can also be analog input0 RA1 can also be analog input1 RA2 can also be analog input2 RA3 can also be analog input3 or analog reference voltage RA4 can also be the clock input to the Timer0 module. Output is open drain type. |
| RA1/AN1 | 18 | 20 | 18 | I/O | TTL | |
| RA2/AN2 | 1 | 1 | 1 | I/O | TTL | |
| RA3/AN3/VREF | 2 | 2 | 2 | I/O | TTL | |
| RA4/T0CKI | 3 | 3 | 3 | I/O | ST | |
| RB0/INT | 6 | 7 | 6 | I/O | TTL/ST ⁽¹⁾ | PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs. RB0 can also be the external interrupt pin. Interrupt on change pin. Interrupt on change pin. Interrupt on change pin. Serial programming clock. Interrupt on change pin. Serial programming data. |
| RB1 | 7 | 8 | 7 | I/O | TTL | |
| RB2 | 8 | 9 | 8 | I/O | TTL | |
| RB3 | 9 | 10 | 9 | I/O | TTL | |
| RB4 | 10 | 11 | 10 | I/O | TTL | |
| RB5 | 11 | 12 | 11 | I/O | TTL | |
| RB6 | 12 | 13 | 12 | I/O | TTL/ST ⁽²⁾ | |
| RB7 | 13 | 14 | 13 | I/O | TTL/ST ⁽²⁾ | |
| VSS | 5 | 4, 6 | 5 | P | — | Ground reference for logic and I/O pins. |
| VDD | 14 | 15, 16 | 14 | P | — | Positive supply for logic and I/O pins. |

Legend: I = input O = output I/O = input/output P = power
 — = Not used TTL = TTL input ST = Schmitt Trigger input

- Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
 Note 2: This buffer is a Schmitt Trigger input when used in serial programming mode.
 Note 3: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.
 Note 4: The PIC16C71 is not available in SSOP package.

PIC16C71X

3.1 Clocking Scheme/Instruction Cycle

The clock input (from OSC1) is internally divided by four to generate four non-overlapping quadrature clocks namely Q1, Q2, Q3 and Q4. Internally, the program counter (PC) is incremented every Q1, the instruction is fetched from the program memory and latched into the instruction register in Q4. The instruction is decoded and executed during the following Q1 through Q4. The clocks and instruction execution flow is shown in Figure 3-2.

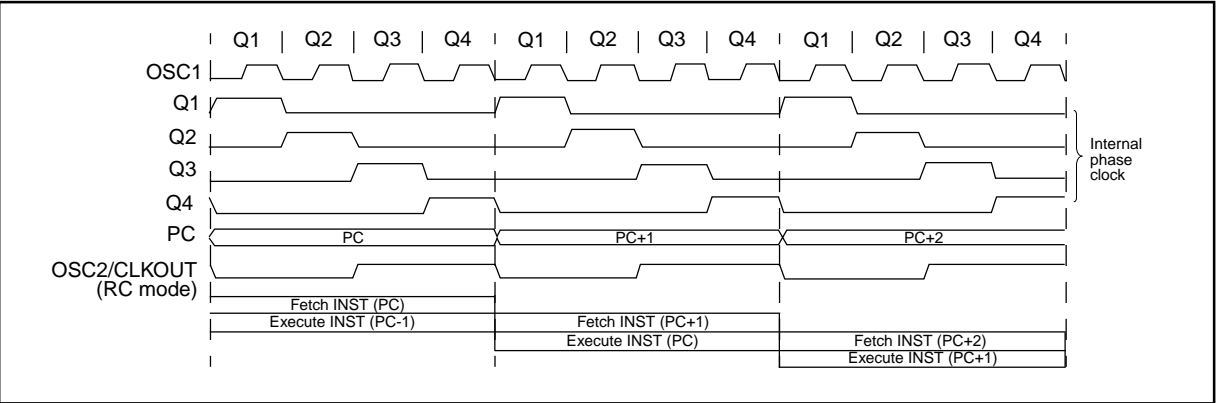
3.2 Instruction Flow/Pipelining

An “Instruction Cycle” consists of four Q cycles (Q1, Q2, Q3 and Q4). The instruction fetch and execute are pipelined such that fetch takes one instruction cycle while decode and execute takes another instruction cycle. However, due to the pipelining, each instruction effectively executes in one cycle. If an instruction causes the program counter to change (e.g. GOTO) then two cycles are required to complete the instruction (Example 3-1).

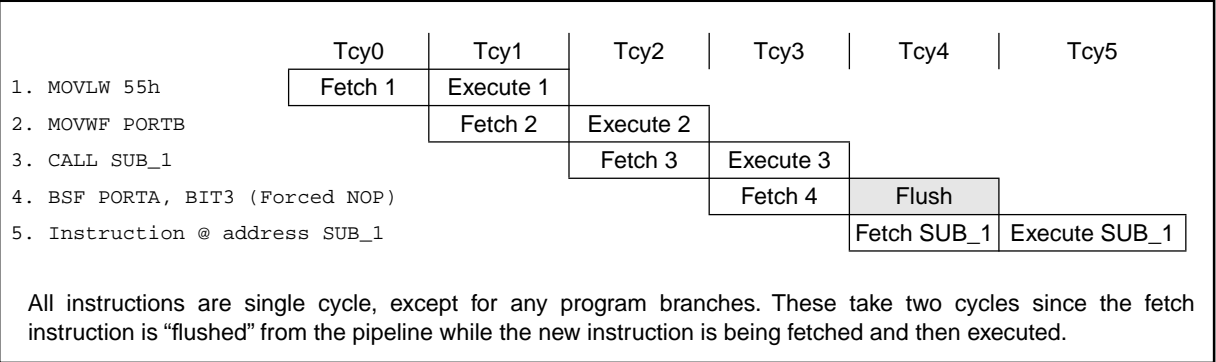
A fetch cycle begins with the program counter (PC) incrementing in Q1.

In the execution cycle, the fetched instruction is latched into the “Instruction Register” (IR) in cycle Q1. This instruction is then decoded and executed during the Q2, Q3, and Q4 cycles. Data memory is read during Q2 (operand read) and written during Q4 (destination write).

FIGURE 3-2: CLOCK/INSTRUCTION CYCLE



EXAMPLE 3-1: INSTRUCTION PIPELINE FLOW



4.0 MEMORY ORGANIZATION

4.1 Program Memory Organization

The PIC16C71X family has a 13-bit program counter capable of addressing an 8K x 14 program memory space. The amount of program memory available to each device is listed below:

| Device | Program Memory | Address Range |
|-----------|----------------|---------------|
| PIC16C710 | 512 x 14 | 0000h-01FFh |
| PIC16C71 | 1K x 14 | 0000h-03FFh |
| PIC16C711 | 1K x 14 | 0000h-03FFh |
| PIC16C715 | 2K x 14 | 0000h-07FFh |

For those devices with less than 8K program memory, accessing a location above the physically implemented address will cause a wraparound.

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

FIGURE 4-1: PIC16C710 PROGRAM MEMORY MAP AND STACK

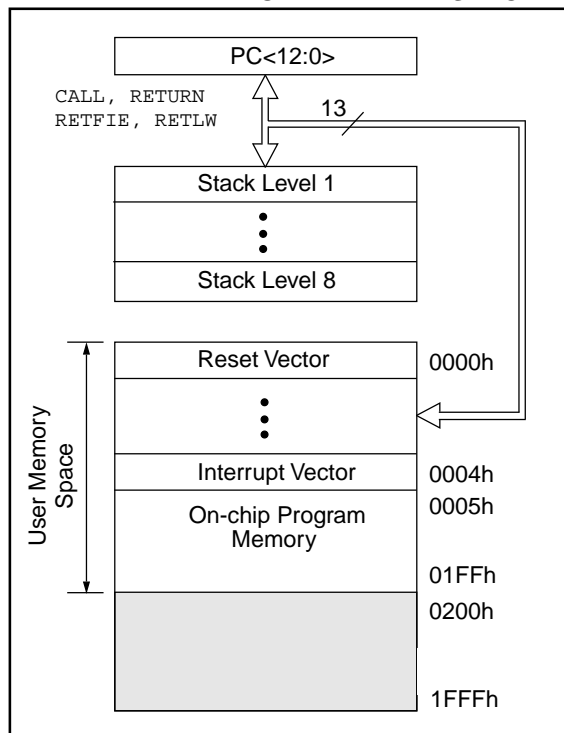


FIGURE 4-2: PIC16C71/711 PROGRAM MEMORY MAP AND STACK

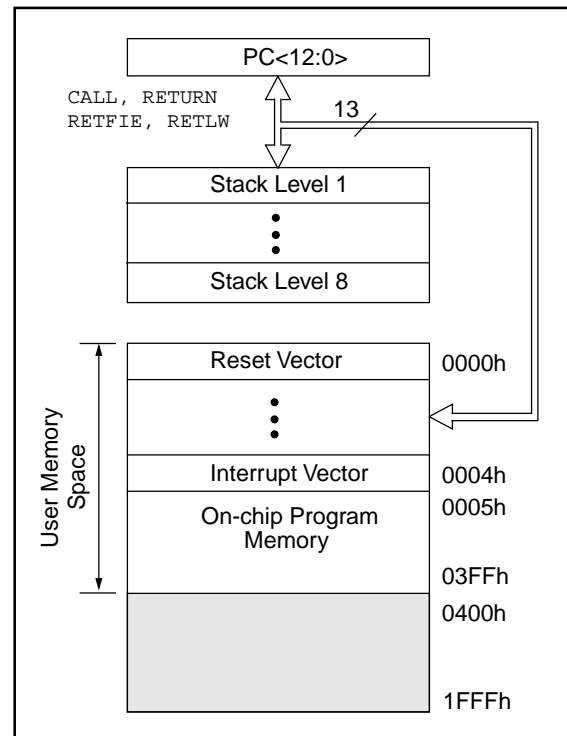
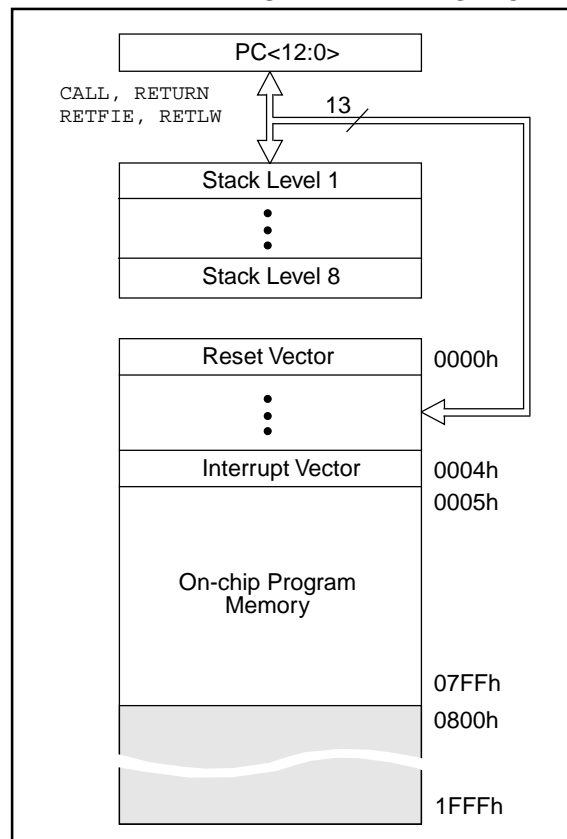


FIGURE 4-3: PIC16C715 PROGRAM MEMORY MAP AND STACK



PIC16C71X

4.2 Data Memory Organization

The data memory is partitioned into two Banks which contain the General Purpose Registers and the Special Function Registers. Bit RP0 is the bank select bit.

RP0 (STATUS<5>) = 1 → Bank 1

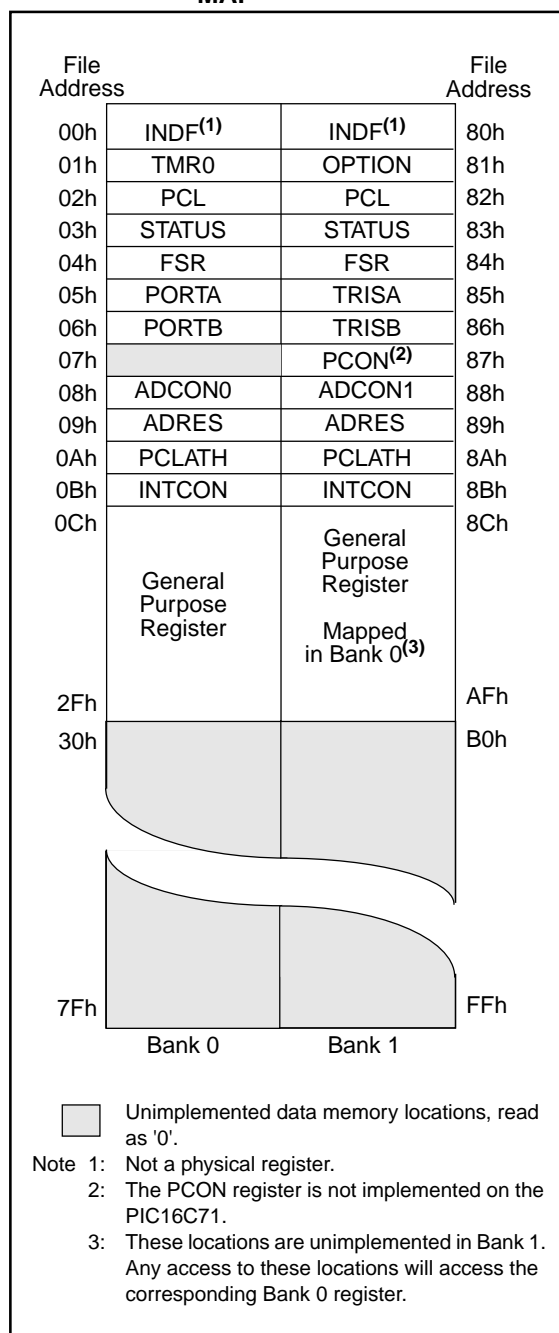
RP0 (STATUS<5>) = 0 → Bank 0

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. Both Bank 0 and Bank 1 contain special function registers. Some "high use" special function registers from Bank 0 are mirrored in Bank 1 for code reduction and quicker access.

4.2.1 GENERAL PURPOSE REGISTER FILE

The register file can be accessed either directly, or indirectly through the File Select Register FSR (Section 4.5).

FIGURE 4-4: PIC16C710/71 REGISTER FILE MAP



PIC16C71X

FIGURE 4-5: PIC16C711 REGISTER FILE MAP

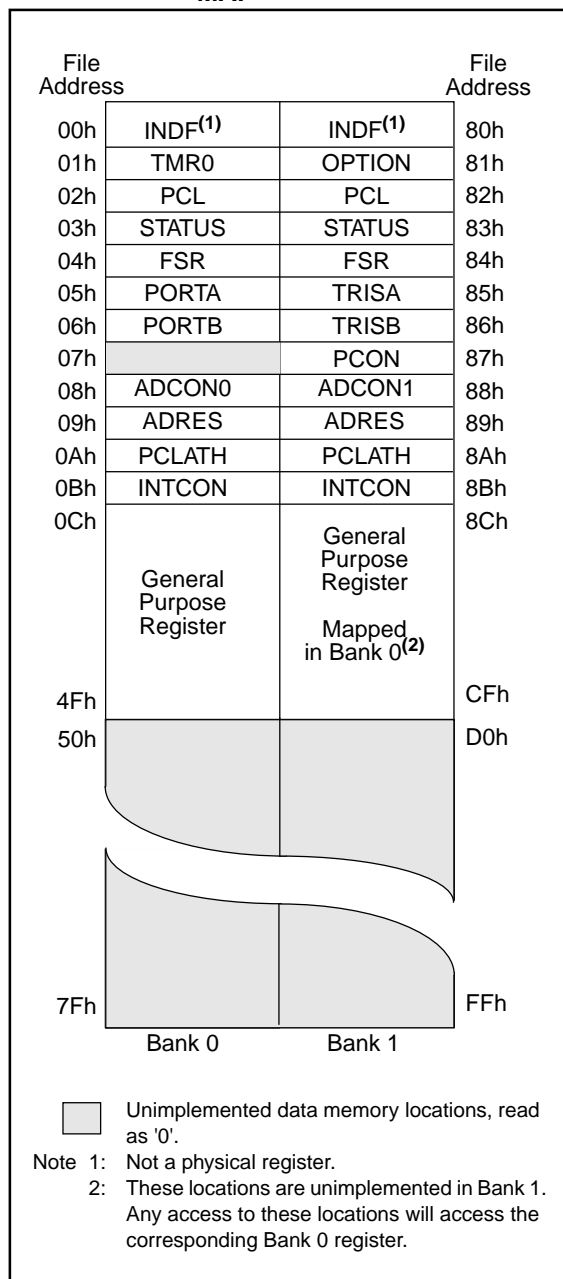
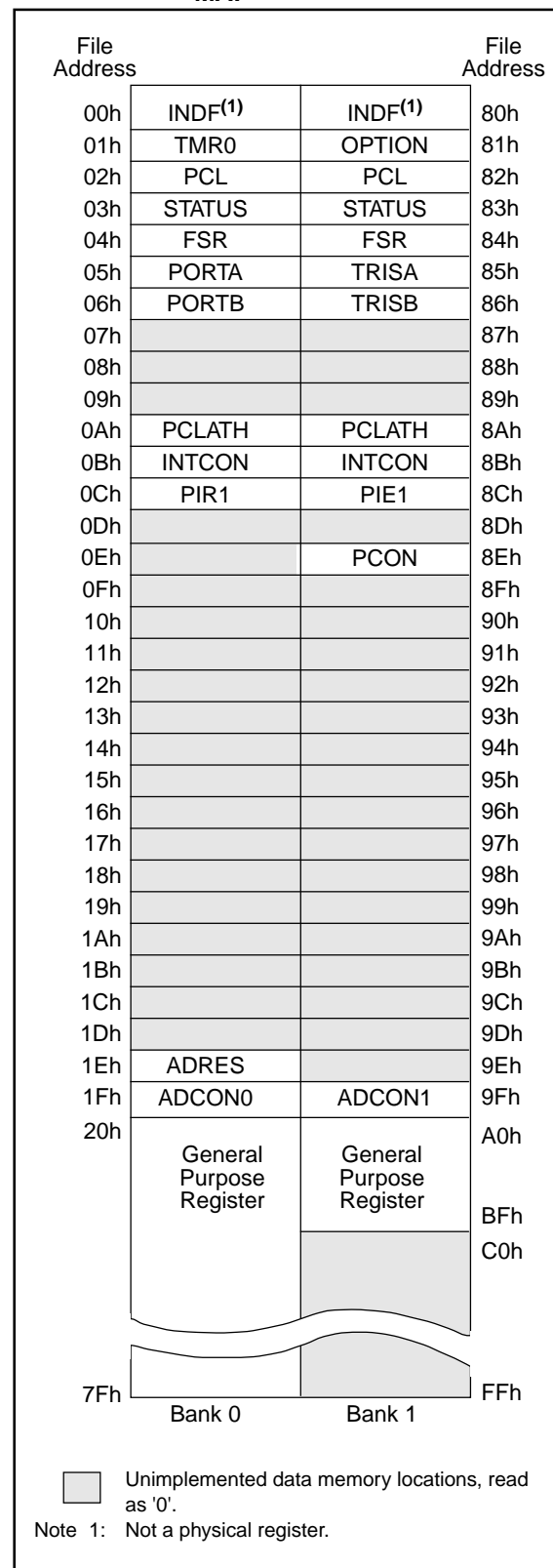


FIGURE 4-6: PIC16C715 REGISTER FILE MAP



PIC16C71X

4.2.2 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.

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

TABLE 4-1: PIC16C710/71/711 SPECIAL FUNCTION REGISTER SUMMARY

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on: POR, BOR | Value on all other resets (1) |
|----------------------|--------|--|--------------------|-------|--|-------|---------|-------|-----------|--------------------------|-------------------------------------|
| Bank 0 | | | | | | | | | | | |
| 00h ⁽³⁾ | INDF | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | 0000 0000 | 0000 0000 |
| 01h | TMR0 | Timer0 module's register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 02h ⁽³⁾ | PCL | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 0000 0000 |
| 03h ⁽³⁾ | STATUS | IRP ⁽⁵⁾ | RP1 ⁽⁵⁾ | RP0 | T0 | PD | Z | DC | C | 0001 1xxx | 000q quuu |
| 04h ⁽³⁾ | FSR | Indirect data memory address pointer | | | | | | | | xxxx xxxx | uuuu uuuu |
| 05h | PORTA | — | — | — | PORTA Data Latch when written: PORTA pins when read | | | | ---x 0000 | ---u 0000 | |
| 06h | PORTB | PORTB Data Latch when written: PORTB pins when read | | | | | | | | xxxx xxxx | uuuu uuuu |
| 07h | — | Unimplemented | | | | | | | | — | — |
| 08h | ADCON0 | ADCS1 | ADCS0 | (6) | CHS1 | CHS0 | GO/DONE | ADIF | ADON | 00-0 0000 | 00-0 0000 |
| 09h ⁽³⁾ | ADRES | A/D Result Register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 0Ah ^(2,3) | PCLATH | — | — | — | Write Buffer for the upper 5 bits of the Program Counter | | | | ---0 0000 | ---0 0000 | |
| 0Bh ⁽³⁾ | INTCON | GIE | ADIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| Bank 1 | | | | | | | | | | | |
| 80h ⁽³⁾ | INDF | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | 0000 0000 | 0000 0000 |
| 81h | OPTION | RBP0 | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |
| 82h ⁽³⁾ | PCL | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 0000 0000 |
| 83h ⁽³⁾ | STATUS | IRP ⁽⁵⁾ | RP1 ⁽⁵⁾ | RP0 | T0 | PD | Z | DC | C | 0001 1xxx | 000q quuu |
| 84h ⁽³⁾ | FSR | Indirect data memory address pointer | | | | | | | | xxxx xxxx | uuuu uuuu |
| 85h | TRISA | — | — | — | PORTA Data Direction Register | | | | ---1 1111 | ---1 1111 | |
| 86h | TRISB | PORTB Data Direction Control Register | | | | | | | | 1111 1111 | 1111 1111 |
| 87h ⁽⁴⁾ | PCON | — | — | — | — | — | — | POR | BOR | ---- --qq | ---- --uu |
| 88h | ADCON1 | — | — | — | — | — | — | PCFG1 | PCFG0 | ---- --00 | ---- --00 |
| 89h ⁽³⁾ | ADRES | A/D Result Register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 8Ah ^(2,3) | PCLATH | — | — | — | Write Buffer for the upper 5 bits of the Program Counter | | | | ---0 0000 | ---0 0000 | |
| 8Bh ⁽³⁾ | INTCON | GIE | ADIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |

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

- Note 1: Other (non power-up) resets include external reset through $\overline{\text{MCLR}}$ and Watchdog Timer Reset.
2: 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.
3: These registers can be addressed from either bank.
4: The PCON register is not physically implemented in the PIC16C71, read as '0'.
5: The IRP and RP1 bits are reserved on the PIC16C710/71/711, always maintain these bits clear.
6: Bit5 of ADCON0 is a General Purpose R/W bit for the PIC16C710/711 only. For the PIC16C71, this bit is unimplemented, read as '0'.

PIC16C71X

TABLE 4-2: PIC16C715 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, PER | Value on all other resets (3) |
|----------------------|--------|--|--------------------|-------|--|-------|---------|-------|-------|-------------------------|-------------------------------|
| Bank 0 | | | | | | | | | | | |
| 00h ⁽¹⁾ | INDF | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | 0000 0000 | 0000 0000 |
| 01h | TMR0 | Timer0 module's register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 02h ⁽¹⁾ | PCL | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 0000 0000 |
| 03h ⁽¹⁾ | STATUS | IRP ⁽⁴⁾ | RP1 ⁽⁴⁾ | RP0 | T0 | PD | Z | DC | C | 0001 1xxx | 000q quuu |
| 04h ⁽¹⁾ | FSR | Indirect data memory address pointer | | | | | | | | xxxx xxxx | uuuu uuuu |
| 05h | PORTA | — | — | — | PORTA Data Latch when written: PORTA pins when read | | | | | ---x 0000 | ---u 0000 |
| 06h | PORTB | PORTB Data Latch when written: PORTB pins when read | | | | | | | | xxxx xxxx | uuuu uuuu |
| 07h | — | Unimplemented | | | | | | | | — | — |
| 08h | — | Unimplemented | | | | | | | | — | — |
| 09h | — | Unimplemented | | | | | | | | — | — |
| 0Ah ^(1,2) | PCLATH | — | — | — | Write Buffer for the upper 5 bits of the Program Counter | | | | | ---0 0000 | ---0 0000 |
| 0Bh ⁽¹⁾ | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | — | — | — | — | -0-- ---- | -0-- ---- |
| 0Dh | — | Unimplemented | | | | | | | | — | — |
| 0Eh | — | Unimplemented | | | | | | | | — | — |
| 0Fh | — | Unimplemented | | | | | | | | — | — |
| 10h | — | Unimplemented | | | | | | | | — | — |
| 11h | — | Unimplemented | | | | | | | | — | — |
| 12h | — | Unimplemented | | | | | | | | — | — |
| 13h | — | Unimplemented | | | | | | | | — | — |
| 14h | — | Unimplemented | | | | | | | | — | — |
| 15h | — | Unimplemented | | | | | | | | — | — |
| 16h | — | Unimplemented | | | | | | | | — | — |
| 17h | — | Unimplemented | | | | | | | | — | — |
| 18h | — | Unimplemented | | | | | | | | — | — |
| 19h | — | Unimplemented | | | | | | | | — | — |
| 1Ah | — | Unimplemented | | | | | | | | — | — |
| 1Bh | — | Unimplemented | | | | | | | | — | — |
| 1Ch | — | Unimplemented | | | | | | | | — | — |
| 1Dh | — | Unimplemented | | | | | | | | — | — |
| 1Eh | ADRES | A/D Result Register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 1Fh | ADCON0 | ADCS1 | ADCS0 | CHS2 | CHS1 | CHS0 | GO/DONE | — | ADON | 0000 00-0 | 0000 00-0 |

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

Shaded locations are unimplemented, read as '0'.

Note 1: These registers can be addressed from either bank.

2: 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.

3: Other (non power-up) resets include external reset through MCLR and Watchdog Timer Reset.

4: The IRP and RP1 bits are reserved on the PIC16C715, always maintain these bits clear.

PIC16C71X

TABLE 4-2: PIC16C715 SPECIAL FUNCTION REGISTER SUMMARY (Cont.'d)

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on: POR, BOR, PER | Value on all other resets (3) |
|----------------------|--------|--|--------------------|-------------------------------|---|------------------------|-------|-------|-------|-------------------------------|-------------------------------------|
| Bank 1 | | | | | | | | | | | |
| 80h ⁽¹⁾ | INDF | Addressing this location uses contents of FSR to address data memory (not a physical register) | | | | | | | | 0000 0000 | 0000 0000 |
| 81h | OPTION | RBP $\overline{\text{U}}$ | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |
| 82h ⁽¹⁾ | PCL | Program Counter's (PC) Least Significant Byte | | | | | | | | 0000 0000 | 0000 0000 |
| 83h ⁽¹⁾ | STATUS | IRP ⁽⁴⁾ | RP1 ⁽⁴⁾ | RP0 | $\overline{\text{T0}}$ | $\overline{\text{PD}}$ | Z | DC | C | 0001 1xxx | 000q quuu |
| 84h ⁽¹⁾ | FSR | Indirect data memory address pointer | | | | | | | | xxxx xxxx | uuuu uuuu |
| 85h | TRISA | — | — | PORTA Data Direction Register | | | | | | --11 1111 | --11 1111 |
| 86h | TRISB | PORTB Data Direction Register | | | | | | | | 1111 1111 | 1111 1111 |
| 87h | — | Unimplemented | | | | | | | | — | — |
| 88h | — | Unimplemented | | | | | | | | — | — |
| 89h | — | Unimplemented | | | | | | | | — | — |
| 8Ah ^(1,2) | PCLATH | — | — | — | Write Buffer for the upper 5 bits of the PC | | | | | ---0 0000 | ---0 0000 |
| 8Bh ⁽¹⁾ | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 8Ch | PIE1 | — | ADIE | — | — | — | — | — | — | -0-- ---- | -0-- ---- |
| 8Dh | — | Unimplemented | | | | | | | | — | — |
| 8Eh | PCON | MPEEN | — | — | — | — | PER | POR | BOR | u--- -1qq | u--- -1uu |
| 8Fh | — | Unimplemented | | | | | | | | — | — |
| 90h | — | Unimplemented | | | | | | | | — | — |
| 91h | — | Unimplemented | | | | | | | | — | — |
| 92h | — | Unimplemented | | | | | | | | — | — |
| 93h | — | Unimplemented | | | | | | | | — | — |
| 94h | — | Unimplemented | | | | | | | | — | — |
| 95h | — | Unimplemented | | | | | | | | — | — |
| 96h | — | Unimplemented | | | | | | | | — | — |
| 97h | — | Unimplemented | | | | | | | | — | — |
| 98h | — | Unimplemented | | | | | | | | — | — |
| 99h | — | Unimplemented | | | | | | | | — | — |
| 9Ah | — | Unimplemented | | | | | | | | — | — |
| 9Bh | — | Unimplemented | | | | | | | | — | — |
| 9Ch | — | Unimplemented | | | | | | | | — | — |
| 9Dh | — | Unimplemented | | | | | | | | — | — |
| 9Eh | — | Unimplemented | | | | | | | | — | — |
| 9Fh | ADCON1 | — | — | — | — | — | — | PCFG1 | PCFG0 | ---- --00 | ---- --00 |

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

Note 1: These registers can be addressed from either bank.

2: 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.

3: Other (non power-up) resets include external reset through MCLR and Watchdog Timer Reset.

4: The IRP and RP1 bits are reserved on the PIC16C715, always maintain these bits clear.

4.2.2.1 STATUS REGISTER

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

The STATUS register, shown in Figure 4-7, 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 1: For those devices that do not use bits IRP and RP1 (STATUS<7:6>), maintain these bits clear to ensure upward compatibility with future products.

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

FIGURE 4-7: STATUS REGISTER (ADDRESS 03h, 83h)

| 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 |
| bit7 | | | | | | | bit0 |
| <p>bit 7: IRP: Register Bank Select bit (used for indirect addressing) 1 = Bank 2, 3 (100h - 1FFh) 0 = Bank 0, 1 (00h - FFh)</p> <p>bit 6-5: RP1:RP0: 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</p> <p>bit 4: \overline{TO}: Time-out bit 1 = After power-up, <code>CLRWDT</code> instruction, or <code>SLEEP</code> instruction 0 = A WDT time-out occurred</p> <p>bit 3: \overline{PD}: Power-down bit 1 = After power-up or by the <code>CLRWDT</code> instruction 0 = By execution of the <code>SLEEP</code> instruction</p> <p>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</p> <p>bit 1: DC: Digit carry/<u>borrow</u> bit (<code>ADDWF</code>, <code>ADDLW</code>, <code>SUBLW</code>, <code>SUBWF</code> instructions)(for <u>borrow</u> 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</p> <p>bit 0: C: Carry/<u>borrow</u> bit (<code>ADDWF</code>, <code>ADDLW</code>, <code>SUBLW</code>, <code>SUBWF</code> 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 <u>borrow</u> the polarity is reversed. A subtraction is executed by adding the two's complement of the second operand. For rotate (<code>RRF</code>, <code>RLF</code>) instructions, this bit is loaded with either the high or low order bit of the source register.</p> | | | | | | | |
| <p>R = Readable bit W = Writable bit U = Unimplemented bit, read as '0' - n = Value at POR reset</p> | | | | | | | |

PIC16C71X

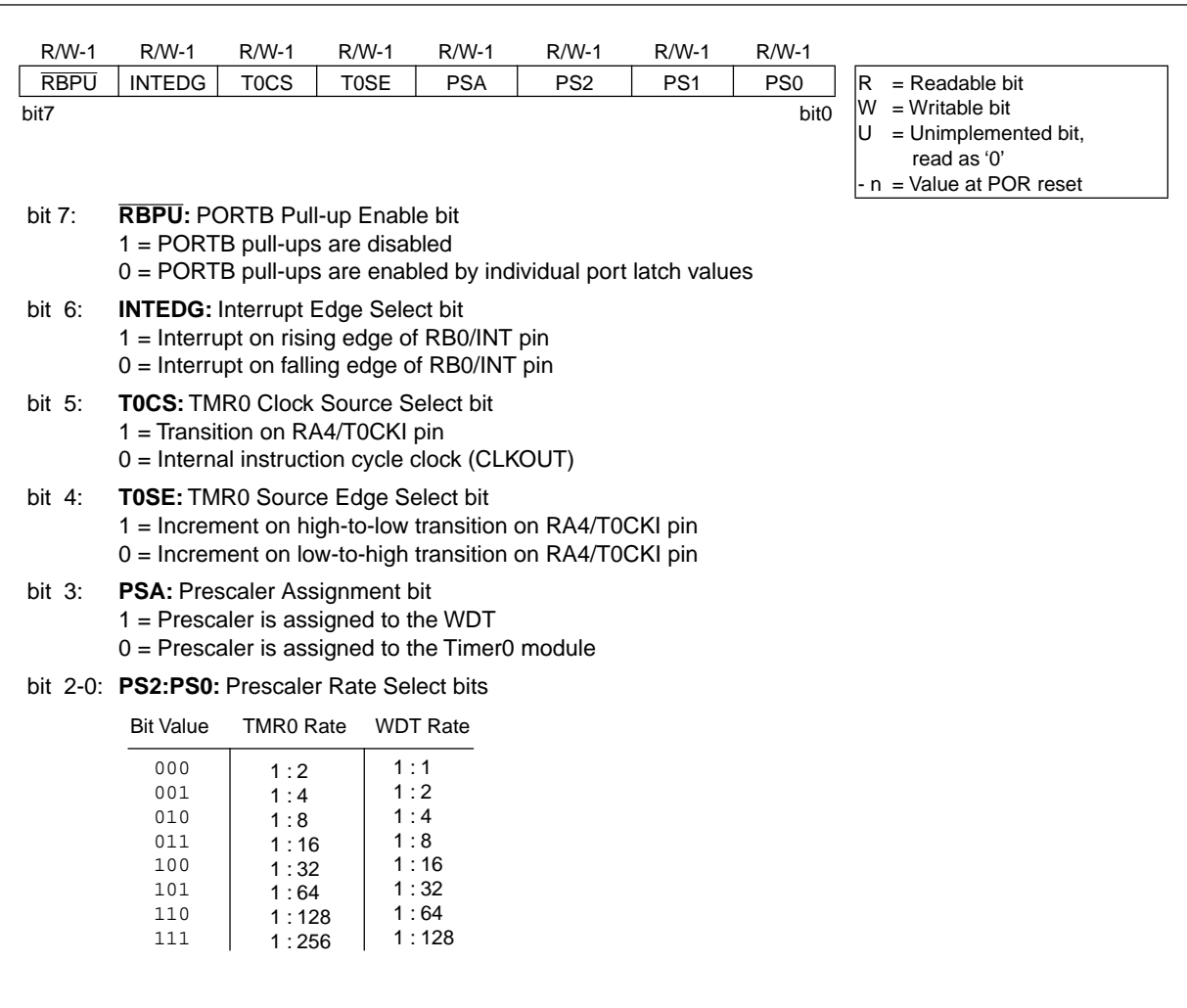
4.2.2.2 OPTION REGISTER

| | | | | |
|--------------------|-----|----|-----|-----|
| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|

The OPTION register is a readable and writable register which contains various control bits to configure the TMR0/WDT 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 by setting bit PSA (OPTION<3>).

FIGURE 4-8: OPTION REGISTER (ADDRESS 81h, 181h)



4.2.2.3 INTCON REGISTER

| | | | | |
|---------------------------|-----|----|-----|-----|
| Applicable Devices | 710 | 71 | 711 | 715 |
|---------------------------|-----|----|-----|-----|

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>).

FIGURE 4-9: INTCON REGISTER (ADDRESS 0Bh, 8Bh)

| 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 | ADIE | TOIE | INTE | RBIE | TOIF | INTF | RBIF |
| bit7 | | | | | | | bit0 |
| <p>bit 7: GIE:⁽¹⁾ Global Interrupt Enable bit 1 = Enables all un-masked interrupts 0 = Disables all interrupts</p> <p>bit 6: ADIE: A/D Converter Interrupt Enable bit 1 = Enables A/D interrupt 0 = Disables A/D interrupt</p> <p>bit 5: TOIE: TMR0 Overflow Interrupt Enable bit 1 = Enables the TMR0 interrupt 0 = Disables the TMR0 interrupt</p> <p>bit 4: INTE: RB0/INT External Interrupt Enable bit 1 = Enables the RB0/INT external interrupt 0 = Disables the RB0/INT external interrupt</p> <p>bit 3: RBIE: RB Port Change Interrupt Enable bit 1 = Enables the RB port change interrupt 0 = Disables the RB port change interrupt</p> <p>bit 2: TOIF: TMR0 Overflow Interrupt Flag bit 1 = TMR0 register has overflowed (must be cleared in software) 0 = TMR0 register did not overflow</p> <p>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</p> <p>bit 0: RBIF: RB Port Change Interrupt Flag bit 1 = At least one of the RB7:RB4 pins changed state (must be cleared in software) 0 = None of the RB7:RB4 pins have changed state</p> <p>Note 1: For the PIC16C71, if an interrupt occurs while the GIE bit is being cleared, the GIE bit may be unintentionally re-enabled by the RETFIE instruction in the user's Interrupt Service Routine. Refer to Section 8.5 for a detailed description.</p> | | | | | | | |
| <p>R = Readable bit W = Writable bit U = Unimplemented bit, read as '0' - n = Value at POR reset</p> | | | | | | | |
| <p>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.</p> | | | | | | | |

PIC16C71X

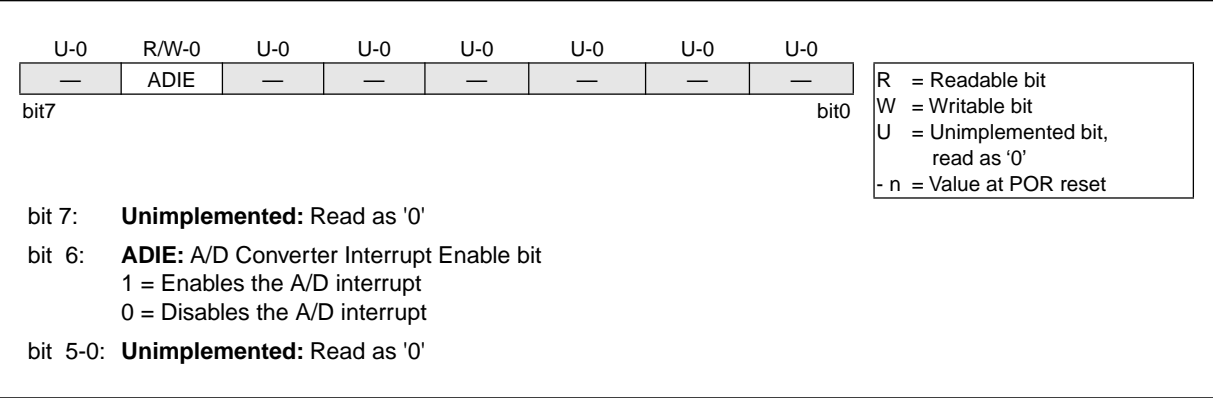
4.2.2.4 PIE1 REGISTER

| | | | | |
|---------------------------|-----|----|-----|-----|
| Applicable Devices | 710 | 71 | 711 | 715 |
|---------------------------|-----|----|-----|-----|

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

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

FIGURE 4-10: PIE1 REGISTER (ADDRESS 8Ch)



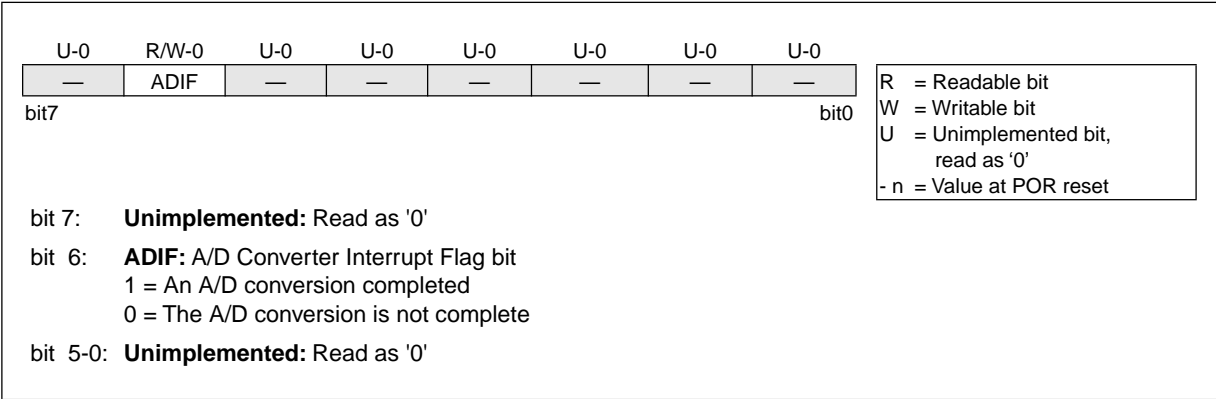
4.2.2.5 PIR1 REGISTER

| | | | | |
|--------------------|-----|----|-----|-----|
| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|

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.

FIGURE 4-11: PIR1 REGISTER (ADDRESS 0Ch)



PIC16C71X

4.2.2.6 PCON REGISTER

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

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 (BOR) condition from a Power-on Reset condition. For the PIC16C715 the PCON register also contains status bits MPEEN and PER. MPEEN reflects the value of the MPEEN bit in the configuration word. PER indicates a parity error reset has occurred.

Note: 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 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).

FIGURE 4-12: PCON REGISTER (ADDRESS 8Eh), PIC16C710/711

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

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

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

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

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

FIGURE 4-13: PCON REGISTER (ADDRESS 8Eh), PIC16C715

| R-U | U-0 | U-0 | U-0 | U-0 | R/W-1 | R/W-0 | R/W-q |
|-------|-----|-----|-----|-----|-------|-------|-------------------------------|
| MPEEN | — | — | — | — | PER | POR | $\overline{\text{BOR}}^{(1)}$ |
| bit7 | | | | | | | bit0 |

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

bit 7: **MPEEN:** Memory Parity Error Circuitry Status bit
Reflects the value of configuration word bit, MPEEN

bit 6-3: **Unimplemented:** Read as '0'

bit 2: **$\overline{\text{PER}}$:** Memory Parity Error Reset Status bit
1 = No Error occurred
0 = Program Memory Fetch Parity Error occurred (must be set in software after a Parity Error Reset)

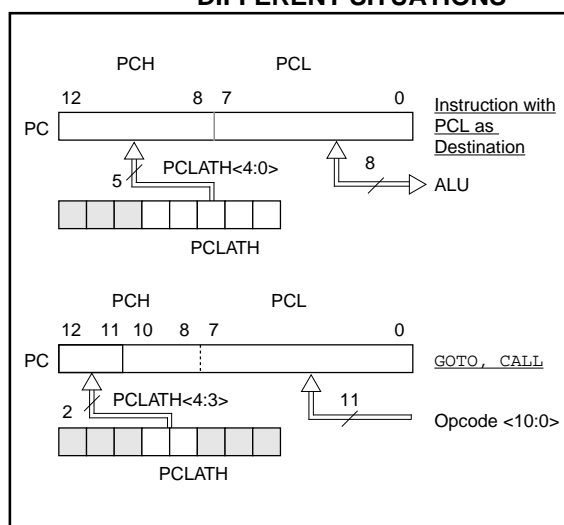
bit 1: **$\overline{\text{POR}}$:** Power-on Reset Status bit
1 = No Power-on Reset occurred
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)

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

4.3 PCL and PCLATH

The program counter (PC) is 13-bits wide. The low byte comes from the PCL register, which is a readable and writable register. The upper bits (PC<12:8>) are not readable, but are indirectly writable through the PCLATH register. On any reset, the upper bits of the PC will be cleared. Figure 4-14 shows the two situations for the loading of the PC. The upper example in the figure shows how the PC is loaded on a write to PCL (PCLATH<4:0> → PCH). The lower example in the figure shows how the PC is loaded during a CALL or GOTO instruction (PCLATH<4:3> → PCH).

FIGURE 4-14: LOADING OF PC IN DIFFERENT SITUATIONS



4.3.1 COMPUTED GOTO

A computed GOTO is accomplished by adding an offset to the program counter (`ADDWF PCL`). When doing a table read using a computed GOTO method, care should be exercised if the table location crosses a PCL memory boundary (each 256 byte block). Refer to the application note "Implementing a Table Read" (AN556).

4.3.2 STACK

The PIC16CXX family has 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 affected by a PUSH or POP operation.

The stack operates as a circular buffer. This means that 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).

Note 1: There are no status bits to indicate stack overflow or stack underflow conditions.

Note 2: There are no instructions/mnemonics called PUSH or POP. These are actions that occur from the execution of the CALL, RETURN, RETLW, and RETFIE instructions, or the vectoring to an interrupt address.

4.4 Program Memory Paging

The PIC16C71X devices ignore both paging bits (PCLATH<4:3>, which are used to access program memory when more than one page is available. The use of PCLATH<4:3> as general purpose read/write bits for the PIC16C71X is not recommended since this may affect upward compatibility with future products.

PIC16C71X

Example 4-1 shows the calling of a subroutine in page 1 of the program memory. This example assumes that PCLATH is saved and restored by the interrupt service routine (if interrupts are used).

EXAMPLE 4-1: CALL OF A SUBROUTINE IN PAGE 1 FROM PAGE 0

```

ORG 0x500
BSF    PCLATH,3    ;Select page 1 (800h-FFFh)
BCF    PCLATH,4    ;Only on >4K devices
CALL   SUB1_P1     ;Call subroutine in
:               ;page 1 (800h-FFFh)
:
:
ORG 0x900
SUB1_P1:           ;called subroutine
:               ;page 1 (800h-FFFh)
:
RETURN      ;return to Call subroutine
:               ;in page 0 (000h-7FFh)

```

4.5 Indirect Addressing, INDF and FSR Registers

The INDF register is not a physical register. Addressing the INDF register will cause indirect addressing.

Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses the register pointed to by the File Select Register, FSR. Reading the INDF register itself indirectly (FSR = '0') will read 00h. Writing to the INDF register indirectly results in a no-operation (although status bits may be affected). An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit (STATUS<7>), as shown in Figure 4-15. However, IRP is not used in the PIC16C71X devices.

A simple program to clear RAM locations 20h-2Fh using indirect addressing is shown in Example 4-2.

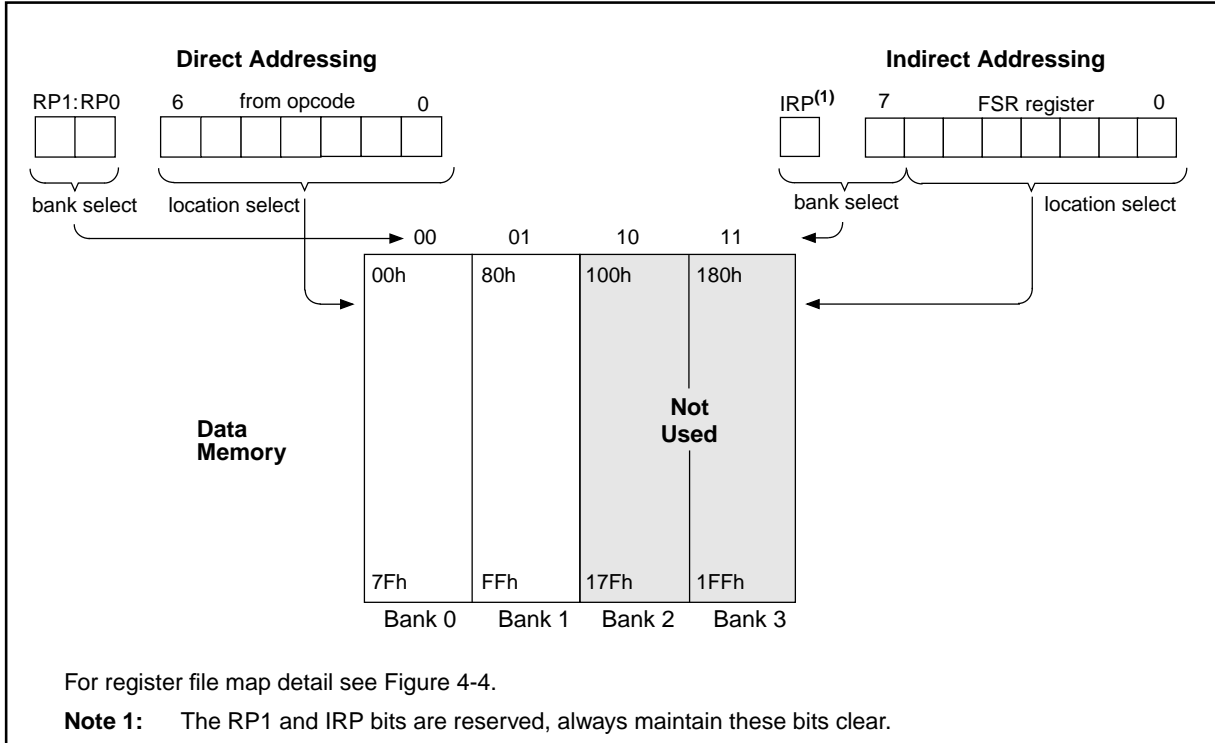
EXAMPLE 4-2: INDIRECT ADDRESSING

```

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

```

FIGURE 4-15: DIRECT/INDIRECT ADDRESSING



5.0 I/O PORTS

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

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.

5.1 PORTA and TRISA Registers

PORTA is a 5-bit latch.

The RA4/T0CKI pin is a Schmitt Trigger input and an open drain output. All other RA port pins have TTL input levels and full CMOS output drivers. All pins have data direction bits (TRIS registers) which can configure these pins as output or input.

Setting a TRISA register bit puts the corresponding output driver in a hi-impedance mode. Clearing a bit in the TRISA register puts the contents of the output latch on the selected pin(s).

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.

Pin RA4 is multiplexed with the Timer0 module clock input to become the RA4/T0CKI pin.

Other PORTA pins are multiplexed with analog inputs and analog VREF input. The operation of each pin is selected by clearing/setting the control bits in the ADCON1 register (A/D Control Register1).

Note: On a Power-on Reset, these pins are configured as analog inputs and read as '0'.

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

EXAMPLE 5-1: INITIALIZING PORTA

```
BCF    STATUS, RP0 ;
CLRF   PORTA       ; Initialize PORTA by
                   ; clearing output
                   ; data latches
BSF    STATUS, RP0 ; Select Bank 1
MOVLW  0xCF        ; Value used to
                   ; initialize data
                   ; direction
MOVWF  TRISA       ; Set RA<3:0> as inputs
                   ; RA<4> as outputs
                   ; TRISA<7:5> are always
                   ; read as '0'.
```

FIGURE 5-1: BLOCK DIAGRAM OF RA3:RA0 PINS

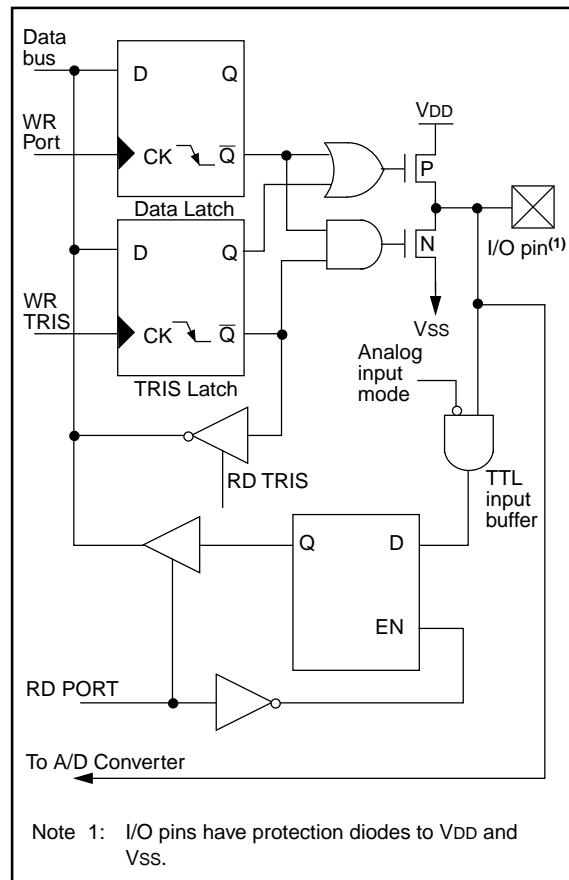
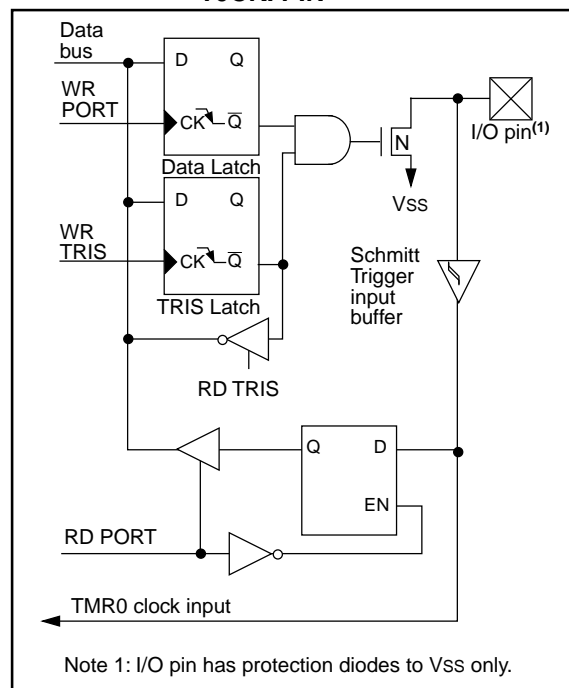


FIGURE 5-2: BLOCK DIAGRAM OF RA4/T0CKI PIN



PIC16C71X

TABLE 5-1: PORTA FUNCTIONS

| Name | Bit# | Buffer | Function |
|--------------|------|--------|--|
| RA0/AN0 | bit0 | TTL | Input/output or analog input |
| RA1/AN1 | bit1 | TTL | Input/output or analog input |
| RA2/AN2 | bit2 | TTL | Input/output or analog input |
| RA3/AN3/VREF | bit3 | TTL | Input/output or analog input/VREF |
| RA4/T0CKI | bit4 | ST | Input/output or external clock input for Timer0 Output is open drain type |

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

TABLE 5-2: 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 | — | — | — | RA4 | RA3 | RA2 | RA1 | RA0 | ---x 0000 | ---u 0000 |
| 85h | TRISA | — | — | — | PORTA Data Direction Register | | | | | ---1 1111 | ---1 1111 |
| 9Fh | ADCON1 | — | — | — | — | — | — | PCFG1 | PCFG0 | ---- --00 | ---- --00 |

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

5.2 PORTB and TRISB Registers

PORTB is an 8-bit wide bi-directional port. The corresponding data direction register is TRISB. Setting a bit in the TRISB register puts the corresponding output driver in a hi-impedance input mode. Clearing a bit in the TRISB register puts the contents of the output latch on the selected pin(s).

EXAMPLE 5-2: INITIALIZING PORTB

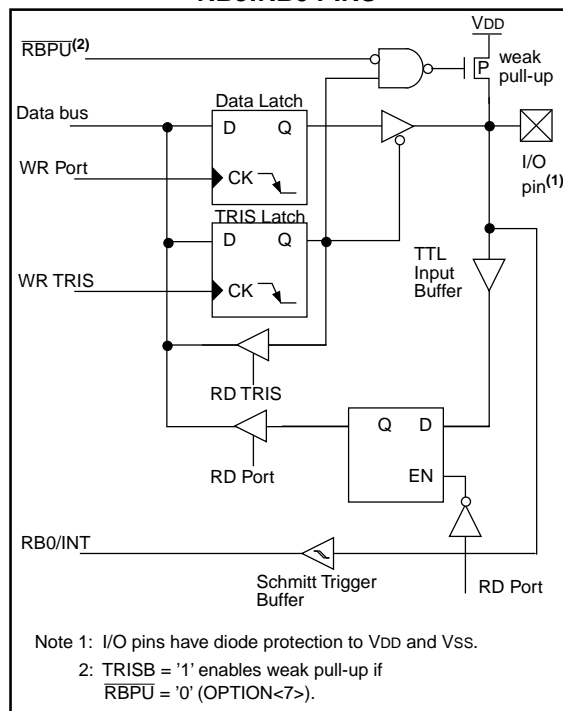
```
BCF    STATUS, RP0 ;
CLRF   PORTB       ; Initialize PORTB by
                   ; clearing output
                   ; data latches

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
```

Each of the PORTB pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is performed by clearing bit $\overline{\text{RBP}}\text{U}$ (OPTION<7>). 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.

FIGURE 5-3: BLOCK DIAGRAM OF RB7:RB0 PINS



Four of PORTB's pins, RB7:RB4, have an interrupt on change feature. Only pins configured as inputs can cause this interrupt to occur (i.e. any RB7:RB4 pin configured as an output is excluded from the interrupt on change comparison). The input pins (of RB7:RB4) are compared with the old value latched on the last read of PORTB. The "mismatch" outputs of RB7:RB4 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:

- Any read or write of PORTB. This will end the mismatch condition.
- 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.

This interrupt on mismatch feature, together with software configurable pull-ups on these four pins allow easy interface to a keypad and make it possible for wake-up on key-depression. Refer to the Embedded Control Handbook, "Implementing Wake-Up on Key Stroke" (AN552).

Note: For the PIC16C71 if a change on the I/O pin should occur when the read operation is being executed (start of the Q2 cycle), then interrupt flag bit RBIF may not get set.

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.

TABLE 5-4: 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 xxxx | uuuu uuuu |
| 86h, 186h | TRISB | PORTB Data Direction Register | | | | | | | | 1111 1111 | 1111 1111 |
| 81h, 181h | OPTION | $\overline{\text{RBPU}}$ | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |

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

PIC16C71X

5.3 I/O Programming Considerations

5.3.1 BI-DIRECTIONAL I/O PORTS

Any instruction which writes, operates internally as a read followed by a write operation. The BCF and BSF instructions, for example, read the register into the CPU, execute the bit operation and write the result back to the register. Caution must be used when these instructions are applied to a port with both inputs and outputs defined. For example, a BSF operation on bit5 of PORTB will cause all eight bits of PORTB to be read into the CPU. Then the BSF operation takes place on bit5 and PORTB is written to the output latches. If another bit of PORTB is used as a bi-directional I/O pin (e.g., bit0) and it is defined as an input at this time, the input signal present on the pin itself would be read into the CPU and rewritten to the data latch of this particular pin, overwriting the previous content. As long as the pin stays in the input mode, no problem occurs. However, if bit0 is switched to an output, the content of the data latch may now be unknown.

Reading the port register, reads the values of the port pins. Writing to the port register writes the value to the port latch. When using read-modify-write instructions (ex. BCF, BSF, etc.) on a port, the value of the port pins is read, the desired operation is done to this value, and this value is then written to the port latch.

Example 5-3 shows the effect of two sequential read-modify-write instructions on an I/O port.

EXAMPLE 5-3: READ-MODIFY-WRITE INSTRUCTIONS ON AN I/O PORT

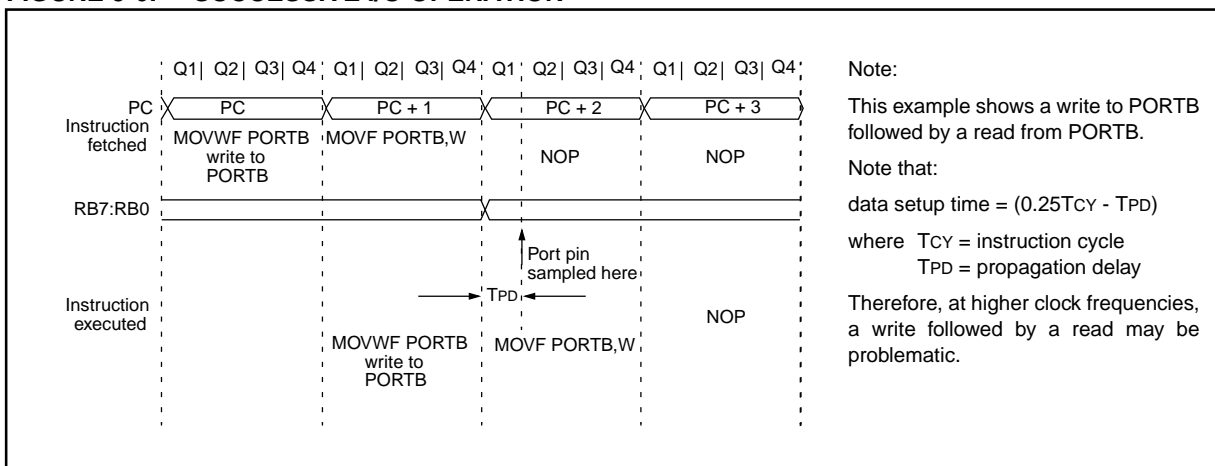
```
;Initial PORT settings: PORTB<7:4> Inputs
;                        PORTB<3:0> Outputs
;PORTB<7:6> have external pull-ups and are
;not connected to other circuitry
;
;                        PORT latch  PORT pins
;                        -----
BCF PORTB, 7      ; 01pp pppp    11pp pppp
BCF PORTB, 6      ; 10pp pppp    11pp pppp
BSF STATUS, RP0   ;
BCF TRISB, 7      ; 10pp pppp    11pp pppp
BCF TRISB, 6      ; 10pp pppp    10pp pppp
;
;Note that the user may have expected the
;pin values to be 00pp ppp. The 2nd BCF
;caused RB7 to be latched as the pin value
;(high).
```

A pin actively outputting a Low or High should not be driven from external devices at the same time in order to change the level on this pin ("wired-or", "wired-and"). The resulting high output currents may damage the chip.

5.3.2 SUCCESSIVE OPERATIONS ON I/O PORTS

The actual write to an I/O port happens at the end of an instruction cycle, whereas for reading, the data must be valid at the beginning of the instruction cycle (Figure 5-6). Therefore, care must be exercised if a write followed by a read operation is carried out on the same I/O port. The sequence of instructions should be such to allow the pin voltage to stabilize (load dependent) before the next instruction which causes that file to be read into the CPU is executed. Otherwise, the previous state of that pin may be read into the CPU rather than the new state. When in doubt, it is better to separate these instructions with a NOP or another instruction not accessing this I/O port.

FIGURE 5-6: SUCCESSIVE I/O OPERATION



6.0 TIMER0 MODULE

Applicable Devices 710 71 711 715

The Timer0 module timer/counter has the following features:

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

Figure 6-1 is a simplified block diagram of the Timer0 module.

Timer mode is selected by clearing bit T0CS (OPTION<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 (Figure 6-2 and Figure 6-3). The user can work around this by writing an adjusted value to the TMR0 register.

Counter mode is selected by setting bit T0CS (OPTION<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<4>). Clearing

bit T0SE selects the rising edge. Restrictions on the external clock input are discussed in detail in Section 6.2.

The prescaler is mutually exclusively shared between the Timer0 module and the Watchdog Timer. The prescaler assignment is controlled in software by control bit PSA (OPTION<3>). Clearing bit PSA will assign the prescaler to the Timer0 module. The prescaler is not readable or writable. When the prescaler is assigned to the Timer0 module, prescale values of 1:2, 1:4, ..., 1:256 are selectable. Section 6.3 details the operation of the prescaler.

6.1 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. See Figure 6-4 for Timer0 interrupt timing.

FIGURE 6-1: TIMER0 BLOCK DIAGRAM

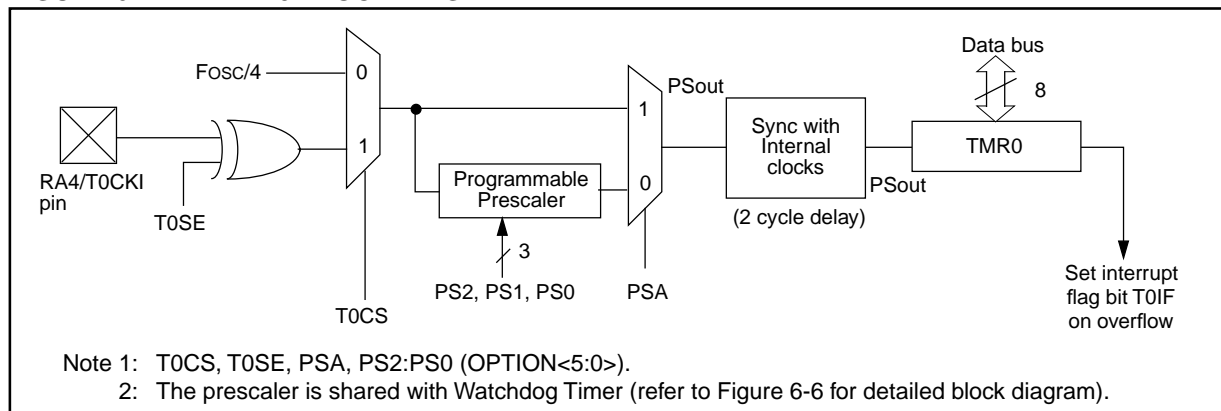
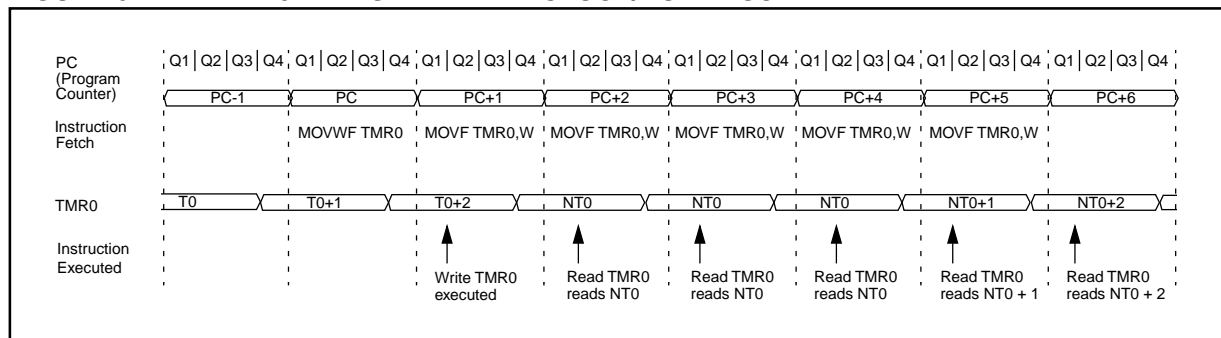


FIGURE 6-2: TIMER0 TIMING: INTERNAL CLOCK/NO PRESCALE



PIC16C71X

FIGURE 6-3: TIMER0 TIMING: INTERNAL CLOCK/PRESCALE 1:2

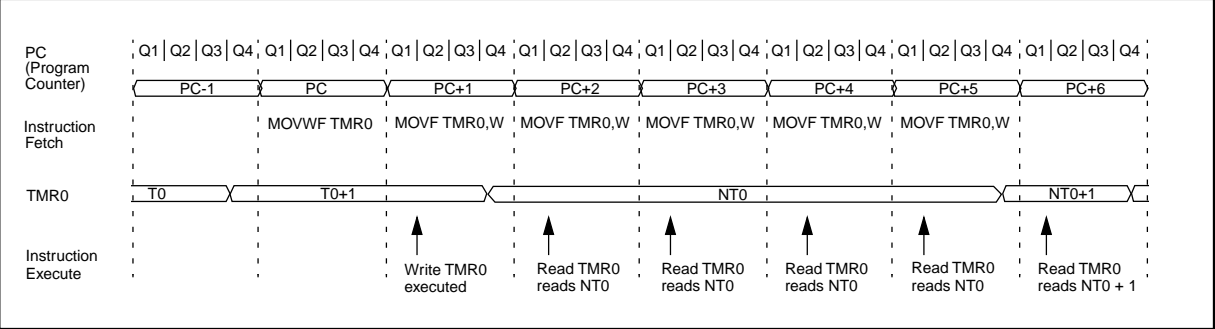
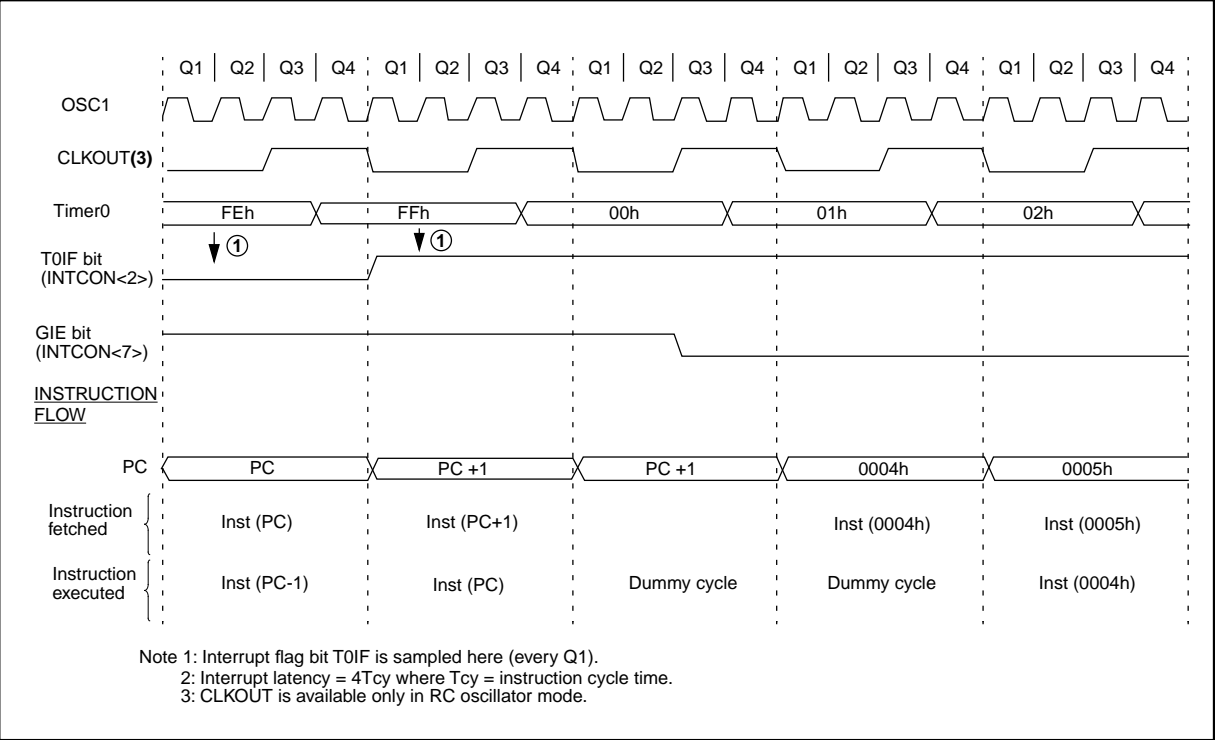


FIGURE 6-4: TIMER0 INTERRUPT TIMING



6.2 Using Timer0 with an External Clock

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.

6.2.1 EXTERNAL CLOCK SYNCHRONIZATION

When no prescaler is used, the external clock input is the same as the prescaler output. The synchronization of T0CKI with the internal phase clocks is accomplished by sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks (Figure 6-5). Therefore, it is necessary for T0CKI to be high for at least 2Tosc (and a small RC delay of 20 ns) and low for at least 2Tosc (and a small RC delay of 20 ns). Refer to the electrical specification of the desired device.

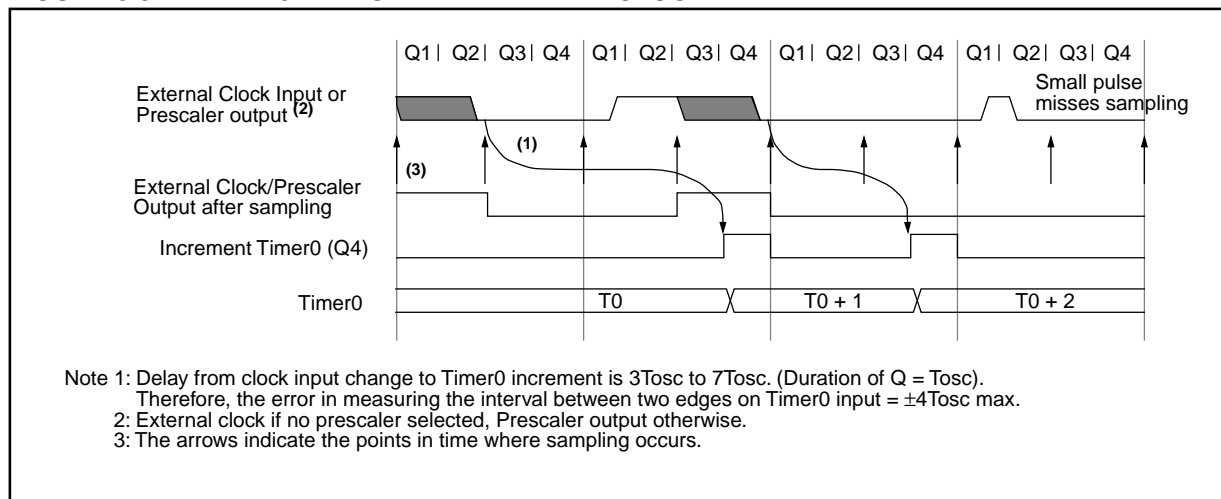
When a prescaler is used, the external clock input is divided by the asynchronous ripple-counter type pres-

caler so that the prescaler output is symmetrical. For the external clock to meet the sampling requirement, the ripple-counter must be taken into account. Therefore, it is necessary for T0CKI to have a period of at least 4Tosc (and a small RC delay of 40 ns) divided by the prescaler value. The only requirement on T0CKI high and low time is that they do not violate the minimum pulse width requirement of 10 ns. Refer to parameters 40, 41 and 42 in the electrical specification of the desired device.

6.2.2 TMR0 INCREMENT DELAY

Since the prescaler output is synchronized with the internal clocks, there is a small delay from the time the external clock edge occurs to the time the Timer0 module is actually incremented. Figure 6-5 shows the delay from the external clock edge to the timer incrementing.

FIGURE 6-5: TIMER0 TIMING WITH EXTERNAL CLOCK



PIC16C71X

6.3 Prescaler

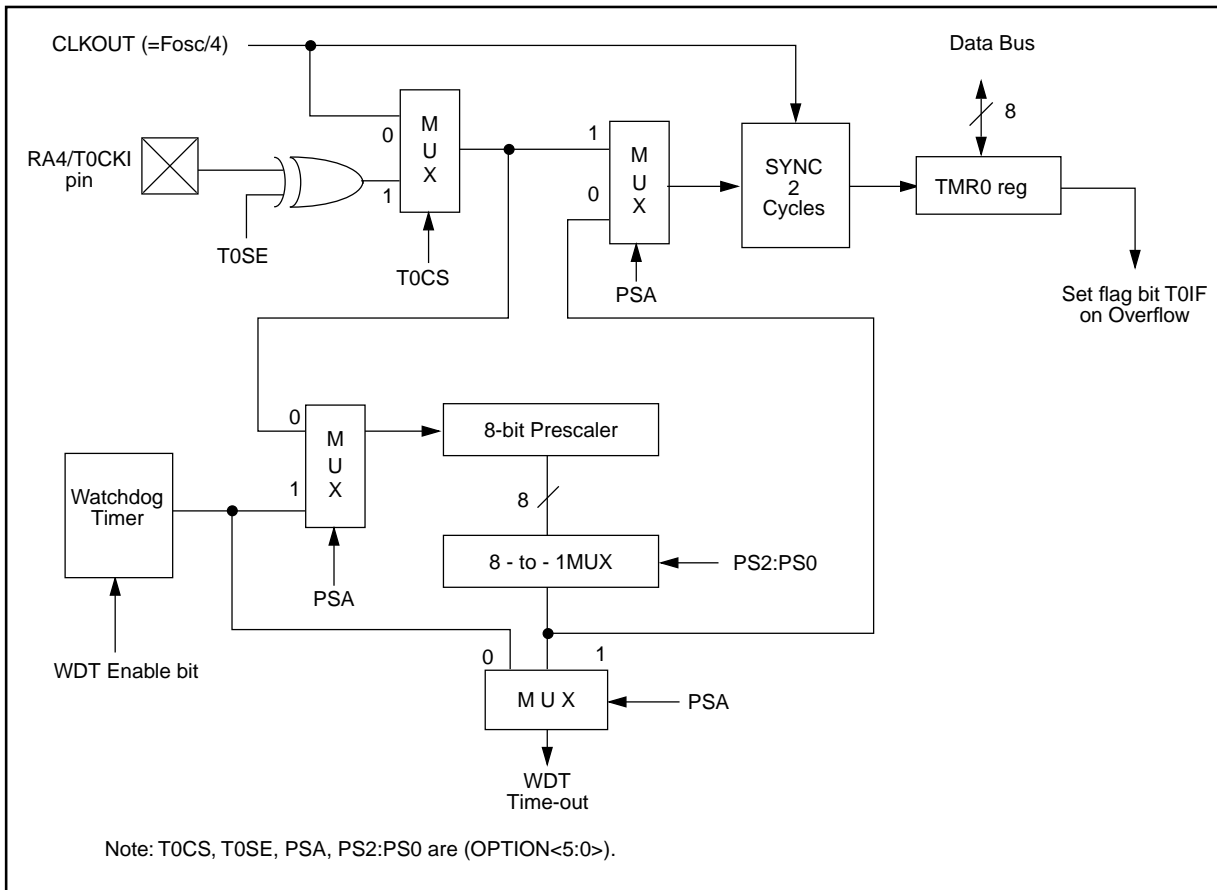
An 8-bit counter is available as a prescaler for the Timer0 module, or as a postscaler for the Watchdog Timer, respectively (Figure 6-6). For simplicity, this counter is being referred to as “prescaler” throughout this data sheet. Note that there is only one prescaler available which is mutually exclusively 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 PSA and PS2:PS0 bits (OPTION<3:0>) determine the prescaler assignment and prescale ratio.

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 CLRWDI instruction will clear the prescaler along with the Watchdog Timer. The prescaler is not readable or writable.

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

FIGURE 6-6: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER



6.3.1 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, the following instruction sequence (shown in Example 6-1) must be executed when changing the prescaler assignment from Timer0 to the WDT. This sequence must be followed even if the WDT is disabled.

EXAMPLE 6-1: CHANGING PRESCALER (TIMER0→WDT)

```
BCF    STATUS, RP0    ;Bank 0
CLRF   TMR0           ;Clear TMR0 & Prescaler
BSF    STATUS, RP0    ;Bank 1
CLRWDT                ;Clears WDT
MOVLW  b'xxxxlxxx'    ;Selects new prescale value
MOVWF  OPTION_REG     ;and assigns the prescaler to the WDT
BCF    STATUS, RP0    ;Bank 0
```

To change prescaler from the WDT to the Timer0 module use the sequence shown in Example 6-2.

EXAMPLE 6-2: CHANGING PRESCALER (WDT→TIMER0)

```
CLRWDT                ;Clear WDT and prescaler
BSF    STATUS, RP0    ;Bank 1
MOVLW  b'xxxx0xxx'    ;Select TMR0, new prescale value and
MOVWF  OPTION_REG     ;clock source
BCF    STATUS, RP0    ;Bank 0
```

TABLE 6-1: 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 | TMR0 | Timer0 module's register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 0Bh,8Bh, | INTCON | GIE | ADIE | TOIE | INTE | RBIE | TOIF | INTF | RBIF | 0000 000x | 0000 000u |
| 81h | OPTION | RBPU | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | 1111 1111 | 1111 1111 |
| 85h | TRISA | — | — | — | PORTA Data Direction Register | | | | | ---1 1111 | ---1 1111 |

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

PIC16C71X

NOTES:

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

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

The analog-to-digital (A/D) converter module has four analog inputs.

The A/D allows conversion of an analog input signal to a corresponding 8-bit digital number (refer to Application Note AN546 for use of A/D Converter). The output of the sample and hold is the input into the converter, which generates the result via successive approximation. The analog reference voltage is software selectable to either the device's positive supply voltage (VDD) or the voltage level on the RA3/AN3/VREF pin.

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 three registers. These registers are:

- A/D Result Register (ADRES)
- A/D Control Register 0 (ADCON0)
- A/D Control Register 1 (ADCON1)

The ADCON0 register, shown in Figure 7-1 and Figure 7-2, controls the operation of the A/D module. The ADCON1 register, shown in Figure 7-3 configures the functions of the port pins. The port pins can be configured as analog inputs (RA3 can also be a voltage reference) or as digital I/O.

FIGURE 7-1: ADCON0 REGISTER (ADDRESS 08h), PIC16C710/71/711

| R/W-0 | R/W-0 | U-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|-------|-------|-------|-------|-------|---------|-------|-------|
| ADCS1 | ADCS0 | — (1) | CHS1 | CHS0 | GO/DONE | ADIF | ADON |
| bit7 | | | | | | | bit0 |

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

bit 7-6: **ADCS1:ADCS0:** A/D Conversion Clock Select bits
00 = FOSC/2
01 = FOSC/8
10 = FOSC/32
11 = FRC (clock derived from an RC oscillation)

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

bit 4-3: **CHS1:CHS0:** Analog Channel Select bits
00 = channel 0, (RA0/AN0)
01 = channel 1, (RA1/AN1)
10 = channel 2, (RA2/AN2)
11 = channel 3, (RA3/AN3)

bit 2: **GO/DONE:** A/D Conversion Status bit
If ADON = 1:
1 = A/D conversion in progress (setting this bit starts the A/D conversion)
0 = A/D conversion not in progress (This bit is automatically cleared by hardware when the A/D conversion is complete)

bit 1: **ADIF:** A/D Conversion Complete Interrupt Flag bit
1 = conversion is complete (must be cleared in software)
0 = conversion is not complete

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

Note 1: Bit5 of ADCON0 is a General Purpose R/W bit for the PIC16C710/711 only. For the PIC16C71, this bit is unimplemented, read as '0'.

PIC16C71X

FIGURE 7-2: ADCON0 REGISTER (ADDRESS 1Fh), PIC16C715

| R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | U-0 | R/W-0 |
|-------|-------|-------|-------|-------|---------|-----|-------|
| ADCS1 | ADCS0 | — | CHS1 | CHS0 | GO/DONE | — | ADON |
| bit7 | | | | | | | bit0 |

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

bit 7-6: **ADCS1:ADCS0**: A/D Conversion Clock Select bits
 00 = Fosc/2
 01 = Fosc/8
 10 = Fosc/32
 11 = FRC (clock derived from an RC oscillation)

bit 5: **Unused**

bit 6-3: **CHS1:CHS0**: Analog Channel Select bits
 000 = channel 0, (RA0/AN0)
 001 = channel 1, (RA1/AN1)
 010 = channel 2, (RA2/AN2)
 011 = channel 3, (RA3/AN3)
 100 = channel 0, (RA0/AN0)
 101 = channel 1, (RA1/AN1)
 110 = channel 2, (RA2/AN2)
 111 = channel 3, (RA3/AN3)

bit 2: **GO/DONE**: A/D Conversion Status bit
 If ADON = 1
 1 = A/D conversion in progress (setting this bit starts the A/D conversion)
 0 = A/D conversion not in progress (This bit is automatically cleared by hardware when the A/D conversion is complete)

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

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

FIGURE 7-3: ADCON1 REGISTER, PIC16C710/71/711 (ADDRESS 88h), PIC16C715 (ADDRESS 9Fh)

| U-0 | U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 |
|------|-----|-----|-----|-----|-----|-------|-------|
| — | — | — | — | — | — | PCFG1 | PCFG0 |
| bit7 | | | | | | | bit0 |

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

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

bit 1-0: **PCFG1:PCFG0**: A/D Port Configuration Control bits

| PCFG1:PCFG0 | RA1 & RA0 | RA2 | RA3 | VREF |
|-------------|-----------|-----|------|------|
| 00 | A | A | A | VDD |
| 01 | A | A | VREF | RA3 |
| 10 | A | D | D | VDD |
| 11 | D | D | D | VDD |

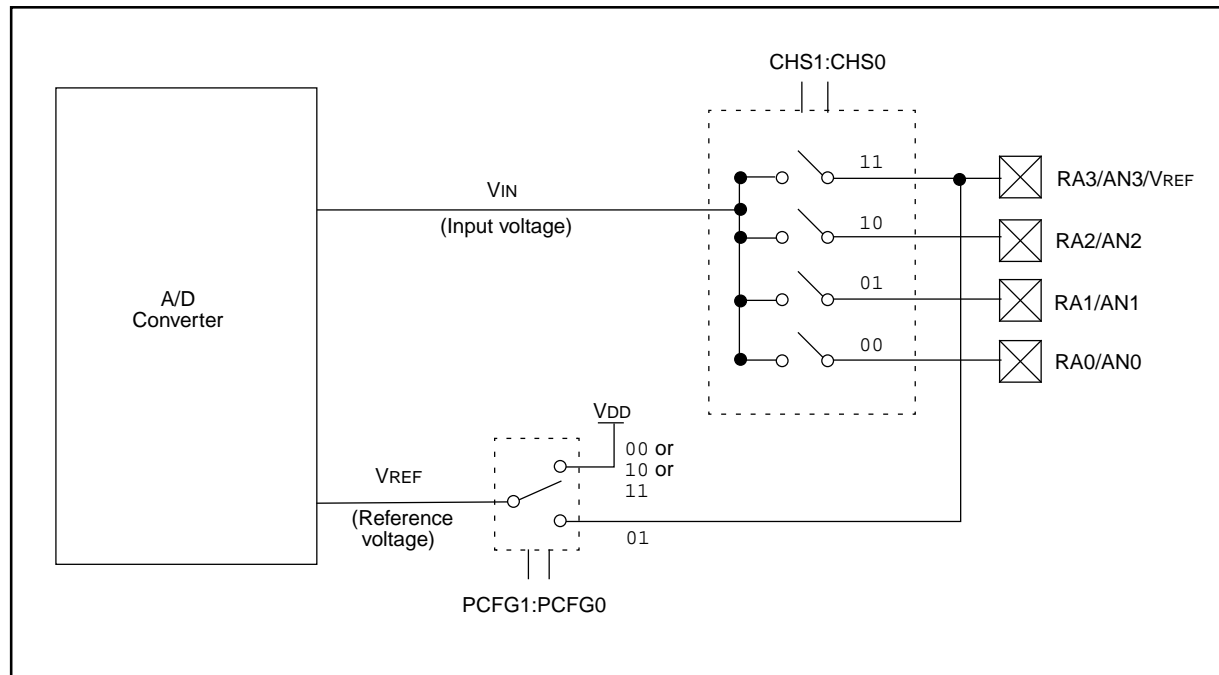
A = Analog input
D = Digital I/O

The ADRES register contains the result of the A/D conversion. When the A/D conversion is complete, the result is loaded into the ADRES register, the GO/DONE bit (ADCON0<2>) is cleared, and A/D interrupt flag bit ADIF is set. The block diagram of the A/D module is shown in Figure 7-4.

After the A/D module has been configured as desired, the selected channel must be acquired before the conversion is started. The analog input channels must have their corresponding TRIS bits selected as an input. To determine acquisition time, see Section 7.1. 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:

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

FIGURE 7-4: A/D BLOCK DIAGRAM



PIC16C71X

7.1 A/D Acquisition Requirements

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 7-5. 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), Figure 7-5. The source impedance affects the offset voltage at the analog input (due to pin leakage current). **The maximum recommended impedance for analog sources is 10 kΩ.** After the analog input channel is selected (changed) this acquisition must be done before the conversion can be started.

To calculate the minimum acquisition time, Equation 7-1 may be used. This equation calculates the acquisition time to within 1/2 LSB error is used (512 steps for the A/D). The 1/2 LSB error is the maximum error allowed for the A/D to meet its specified accuracy.

EQUATION 7-1: A/D MINIMUM CHARGING TIME

$$V_{\text{HOLD}} = (V_{\text{REF}} - (V_{\text{REF}}/512)) \cdot (1 - e^{(-T_{\text{CAP}}/\text{CHOLD}(\text{RIC} + \text{RSS} + \text{RS})))}$$

Given: $V_{\text{HOLD}} = (V_{\text{REF}}/512)$, for 1/2 LSB resolution

The above equation reduces to:

$$T_{\text{CAP}} = -(51.2 \text{ pF})(1 \text{ k}\Omega + \text{RSS} + \text{RS}) \ln(1/511)$$

Example 7-1 shows the calculation of the minimum required acquisition time T_{ACQ} . This calculation is based on the following system assumptions.

CHOLD = 51.2 pF

Rs = 10 kΩ

1/2 LSB error

VDD = 5V → Rss = 7 kΩ

Temp (application system max.) = 50°C

VHOLD = 0 @ t = 0

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

Note 2: The charge holding capacitor (CHOLD) is not discharged after each conversion.

Note 3: The maximum recommended impedance for analog sources is 10 kΩ. This is required to meet the pin leakage specification.

Note 4: After a conversion has completed, a 2.0TAD delay must complete before acquisition can begin again. During this time the holding capacitor is not connected to the selected A/D input channel.

EXAMPLE 7-1: CALCULATING THE MINIMUM REQUIRED ACQUISITION TIME

$T_{\text{ACQ}} = \text{Amplifier Settling Time} +$
 $\text{Holding Capacitor Charging Time} +$
 $\text{Temperature Coefficient}$

$$T_{\text{ACQ}} = 5 \mu\text{s} + T_{\text{CAP}} + [(\text{Temp} - 25^\circ\text{C})(0.05 \mu\text{s}/^\circ\text{C})]$$

$$T_{\text{CAP}} = -\text{CHOLD} (\text{RIC} + \text{RSS} + \text{RS}) \ln(1/511)$$

$$-51.2 \text{ pF} (1 \text{ k}\Omega + 7 \text{ k}\Omega + 10 \text{ k}\Omega) \ln(0.0020)$$

$$-51.2 \text{ pF} (18 \text{ k}\Omega) \ln(0.0020)$$

$$-0.921 \mu\text{s} (-6.2364)$$

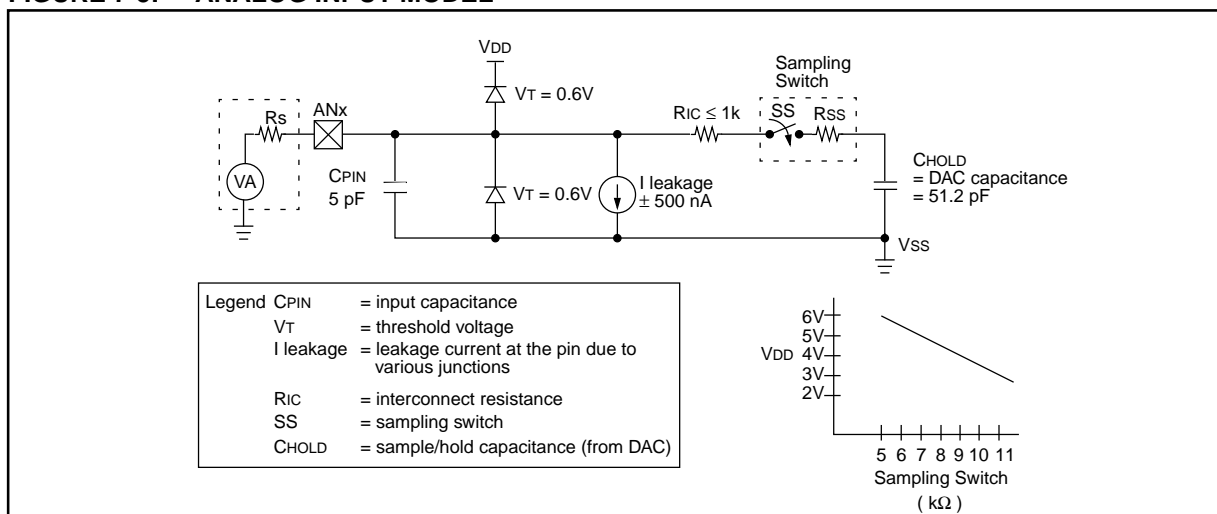
$$5.747 \mu\text{s}$$

$$T_{\text{ACQ}} = 5 \mu\text{s} + 5.747 \mu\text{s} + [(50^\circ\text{C} - 25^\circ\text{C})(0.05 \mu\text{s}/^\circ\text{C})]$$

$$10.747 \mu\text{s} + 1.25 \mu\text{s}$$

$$11.997 \mu\text{s}$$

FIGURE 7-5: ANALOG INPUT MODEL



7.2 Selecting the A/D Conversion Clock

The A/D conversion time per bit is defined as TAD. The A/D conversion requires 9.5TAD per 8-bit conversion. The source of the A/D conversion clock is software selectable. The four possible options for TAD are:

- 2Tosc
- 8Tosc
- 32Tosc
- Internal RC oscillator

For correct A/D conversions, the A/D conversion clock (TAD) must be selected to ensure a minimum TAD time of:

2.0 µs for the PIC16C71

1.6 µs for all other PIC16C71X devices

Table 7-1 and Table 7-2 show the resultant TAD times derived from the device operating frequencies and the A/D clock source selected.

7.3 Configuring Analog Port Pins

The ADCON1 and TRISA registers control the operation of the A/D port pins. The port pins that are desired as analog inputs must have their corresponding TRIS bits set (input). If the TRIS bit is cleared (output), the digital output level (VOH or VOL) will be converted.

The A/D operation is independent of the state of the CHS2:CHS0 bits and the TRIS bits.

Note 1: When reading the port register, all pins configured as analog input channels will read as cleared (a low level). Pins configured as digital inputs, will convert an analog input. Analog levels on a digitally configured input will not affect the conversion accuracy.

Note 2: Analog levels on any pin that is defined as a digital input (including the AN7:AN0 pins), may cause the input buffer to consume current that is out of the devices specification.

TABLE 7-1: TAD vs. DEVICE OPERATING FREQUENCIES, PIC16C71

| AD Clock Source (TAD) | | Device Frequency | | | | |
|-----------------------|-------------|---------------------------|---------------------------|---------------------------|-------------------------|-------------------------|
| Operation | ADCS1:ADCS0 | 20 MHz | 16 MHz | 4 MHz | 1 MHz | 333.33 kHz |
| 2Tosc | 00 | 100 ns ⁽²⁾ | 125 ns ⁽²⁾ | 500 ns ⁽²⁾ | 2.0 µs | 6 µs |
| 8Tosc | 01 | 400 ns ⁽²⁾ | 500 ns ⁽²⁾ | 2.0 µs | 8.0 µs | 24 µs ⁽³⁾ |
| 32Tosc | 10 | 1.6 µs ⁽²⁾ | 2.0 µs | 8.0 µs | 32.0 µs ⁽³⁾ | 96 µs ⁽³⁾ |
| RC ⁽⁵⁾ | 11 | 2 - 6 µs ^(1,4) | 2 - 6 µs ^(1,4) | 2 - 6 µs ^(1,4) | 2 - 6 µs ⁽¹⁾ | 2 - 6 µs ⁽¹⁾ |

Legend: Shaded cells are outside of recommended range.

Note 1: The RC source has a typical TAD time of 4 µs.

2: These values violate the minimum required TAD time.

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

4: When device frequency is greater than 1 MHz, the RC A/D conversion clock source is recommended for sleep operation only.

5: For extended voltage devices (LC), please refer to Electrical Specifications section.

TABLE 7-2: TAD vs. DEVICE OPERATING FREQUENCIES, PIC16C710/711, PIC16C715

| AD Clock Source (TAD) | | Device Frequency | | | |
|-----------------------|-------------|---------------------------|---------------------------|---------------------------|-------------------------|
| Operation | ADCS1:ADCS0 | 20 MHz | 5 MHz | 1.25 MHz | 333.33 kHz |
| 2Tosc | 00 | 100 ns ⁽²⁾ | 400 ns ⁽²⁾ | 1.6 µs | 6 µs |
| 8Tosc | 01 | 400 ns ⁽²⁾ | 1.6 µs | 6.4 µs | 24 µs ⁽³⁾ |
| 32Tosc | 10 | 1.6 µs | 6.4 µs | 25.6 µs ⁽³⁾ | 96 µs ⁽³⁾ |
| RC ⁽⁵⁾ | 11 | 2 - 6 µs ^(1,4) | 2 - 6 µs ^(1,4) | 2 - 6 µs ^(1,4) | 2 - 6 µs ⁽¹⁾ |

Legend: Shaded cells are outside of recommended range.

Note 1: The RC source has a typical TAD time of 4 µs.

2: These values violate the minimum required TAD time.

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

4: When device frequency is greater than 1 MHz, the RC A/D conversion clock source is recommended for sleep operation only.

5: For extended voltage devices (LC), please refer to Electrical Specifications section.

PIC16C71X

7.4 A/D Conversions

Example 7-2 shows how to perform an A/D conversion. The RA pins are configured as analog inputs. The analog reference (VREF) is the device VDD. The A/D interrupt is enabled, and the A/D conversion clock is FRC. The conversion is performed on the RA0 pin (channel 0).

Note: The GO/DONE bit should **NOT** be set in the same instruction that turns on the A/D.

Clearing the GO/DONE bit during a conversion will abort the current conversion. The ADRES register will NOT be updated with the partially completed A/D conversion sample. That is, the ADRES register will continue to contain the value of the last completed conversion (or the last value written to the ADRES register). After the A/D conversion is aborted, a 2TAD wait is required before the next acquisition is started. After this 2TAD wait, an acquisition is automatically started on the selected channel.

EXAMPLE 7-2: A/D CONVERSION

```
BSF    STATUS, RP0        ; Select Bank 1
CLRFB  ADCON1             ; Configure A/D inputs
BCF    STATUS, RP0        ; Select Bank 0
MOVLW  0xC1               ; RC Clock, A/D is on, Channel 0 is selected
MOVWF  ADCON0             ;
BSF    INTCON, ADIE       ; Enable A/D Interrupt
BSF    INTCON, GIE        ; Enable all interrupts
;
; Ensure that the required sampling time for the selected input channel has elapsed.
; Then the conversion may be started.
;
BSF    ADCON0, GO         ; Start A/D Conversion
:      ; The ADIF bit will be set and the GO/DONE bit
:      ; is cleared upon completion of the A/D Conversion.
```

7.4.1 FASTER CONVERSION - LOWER RESOLUTION TRADE-OFF

Not all applications require a result with 8-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 clock source of the A/D module may be switched so that the TAD time violates the minimum specified time (see the applicable electrical specification). Once the TAD time violates the minimum specified time, all the following A/D result bits are not valid (see A/D Conversion Timing in the Electrical Specifications section.) The clock sources may only be switched between the three oscillator versions (cannot be switched from/to RC). The equation to determine the time before the oscillator can be switched is as follows:

$$\text{Conversion time} = 2TAD + N \cdot TAD + (8 - N)(2TOSC)$$

Where: N = number of bits of resolution required.

Since the TAD is based from the device oscillator, the user must use some method (a timer, software loop, etc.) to determine when the A/D oscillator may be changed. Example 7-3 shows a comparison of time required for a conversion with 4-bits of resolution, versus the 8-bit resolution conversion. The example is for devices operating at 20 MHz and 16 MHz (The A/D clock is programmed for 32TOSC), and assumes that immediately after 6TAD, the A/D clock is programmed for 2TOSC.

The 2TOSC violates the minimum TAD time since the last 4-bits will not be converted to correct values.

EXAMPLE 7-3: 4-BIT vs. 8-BIT CONVERSION TIMES

| | Freq. (MHz) ⁽¹⁾ | Resolution | |
|---------------------------------|----------------------------|--------------|-------------|
| | | 4-bit | 8-bit |
| TAD | 20 | 1.6 μ s | 1.6 μ s |
| | 16 | 2.0 μ s | 2.0 μ s |
| TOSC | 20 | 50 ns | 50 ns |
| | 16 | 62.5 ns | 62.5 ns |
| 2TAD + N • TAD + (8 - N)(2TOSC) | 20 | 10 μ s | 16 μ s |
| | 16 | 12.5 μ s | 20 μ s |

Note 1: The PIC16C71 has a minimum TAD time of 2.0 μ s.
All other PIC16C71X devices have a minimum TAD time of 1.6 μ s.

PIC16C71X

7.5 A/D Operation During Sleep

The A/D module can operate during SLEEP mode. This requires that the A/D clock source be set to RC (ADCS1:ADCS0 = 11). When the RC clock source is selected, the A/D module waits one instruction cycle before starting the conversion. This allows the SLEEP instruction to be executed, which eliminates all digital switching noise from the conversion. When the conversion is completed the GO/DONE bit will be cleared, and the result loaded into the ADRES register. 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 will cause the present conversion to be aborted and the A/D module to be 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 set to RC (ADCS1:ADCS0 = 11). To perform an A/D conversion in SLEEP, ensure the SLEEP instruction immediately follows the instruction that sets the GO/DONE bit.

7.6 A/D Accuracy/Error

The absolute accuracy specified for the A/D converter includes the sum of all contributions for quantization error, integral error, differential error, full scale error, offset error, and monotonicity. It is defined as the maximum deviation from an actual transition versus an ideal transition for any code. The absolute error of the A/D converter is specified at ± 1 LSB for $V_{DD} = V_{REF}$ (over the device's specified operating range). However, the accuracy of the A/D converter will degrade as V_{DD} diverges from V_{REF} .

For a given range of analog inputs, the output digital code will be the same. This is due to the quantization of the analog input to a digital code. Quantization error is typically $\pm 1/2$ LSB and is inherent in the analog to digital conversion process. The only way to reduce quantization error is to increase the resolution of the A/D converter.

Offset error measures the first actual transition of a code versus the first ideal transition of a code. Offset error shifts the entire transfer function. Offset error can be calibrated out of a system or introduced into a system through the interaction of the total leakage current and source impedance at the analog input.

Gain error measures the maximum deviation of the last actual transition and the last ideal transition adjusted for offset error. This error appears as a change in slope of the transfer function. The difference in gain error to

full scale error is that full scale does not take offset error into account. Gain error can be calibrated out in software.

Linearity error refers to the uniformity of the code changes. Linearity errors cannot be calibrated out of the system. Integral non-linearity error measures the actual code transition versus the ideal code transition adjusted by the gain error for each code.

Differential non-linearity measures the maximum actual code width versus the ideal code width. This measure is unadjusted.

In systems where the device frequency is low, use of the A/D RC clock is preferred. At moderate to high frequencies, TAD should be derived from the device oscillator. TAD must not violate the minimum and should be $\leq 8 \mu s$ for preferred operation. This is because TAD, when derived from TOSC, is kept away from on-chip phase clock transitions. This reduces, to a large extent, the effects of digital switching noise. This is not possible with the RC derived clock. The loss of accuracy due to digital switching noise can be significant if many I/O pins are active.

In systems where the device will enter SLEEP mode after the start of the A/D conversion, the RC clock source selection is required. In this mode, the digital noise from the modules in SLEEP are stopped. This method gives high accuracy.

7.7 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 ADRES register is not modified for a Power-on Reset. The ADRES register will contain unknown data after a Power-on Reset.

7.8 Connection Considerations

If the input voltage exceeds the rail values (V_{SS} or V_{DD}) by greater than 0.2V, then the accuracy of the conversion is out of specification.

Note: Care must be taken when using the RA0 pin in A/D conversions due to its proximity to the OSC1 pin.

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 10 k Ω recommended specification. Any external components connected (via hi-impedance) to an analog input pin (capacitor, zener diode, etc.) should have very little leakage current at the pin.

7.9 Transfer Function

The ideal transfer function of the A/D converter is as follows: the first transition occurs when the analog input voltage (V_{AIN}) is Analog $V_{REF}/256$ (Figure 7-6).

7.10 References

A very good reference for understanding A/D converters is the "Analog-Digital Conversion Handbook" third edition, published by Prentice Hall (ISBN 0-13-03-2848-0).

FIGURE 7-6: A/D TRANSFER FUNCTION

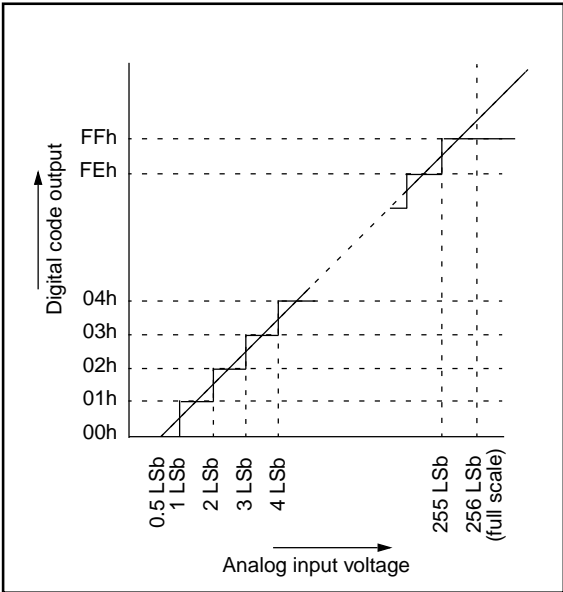
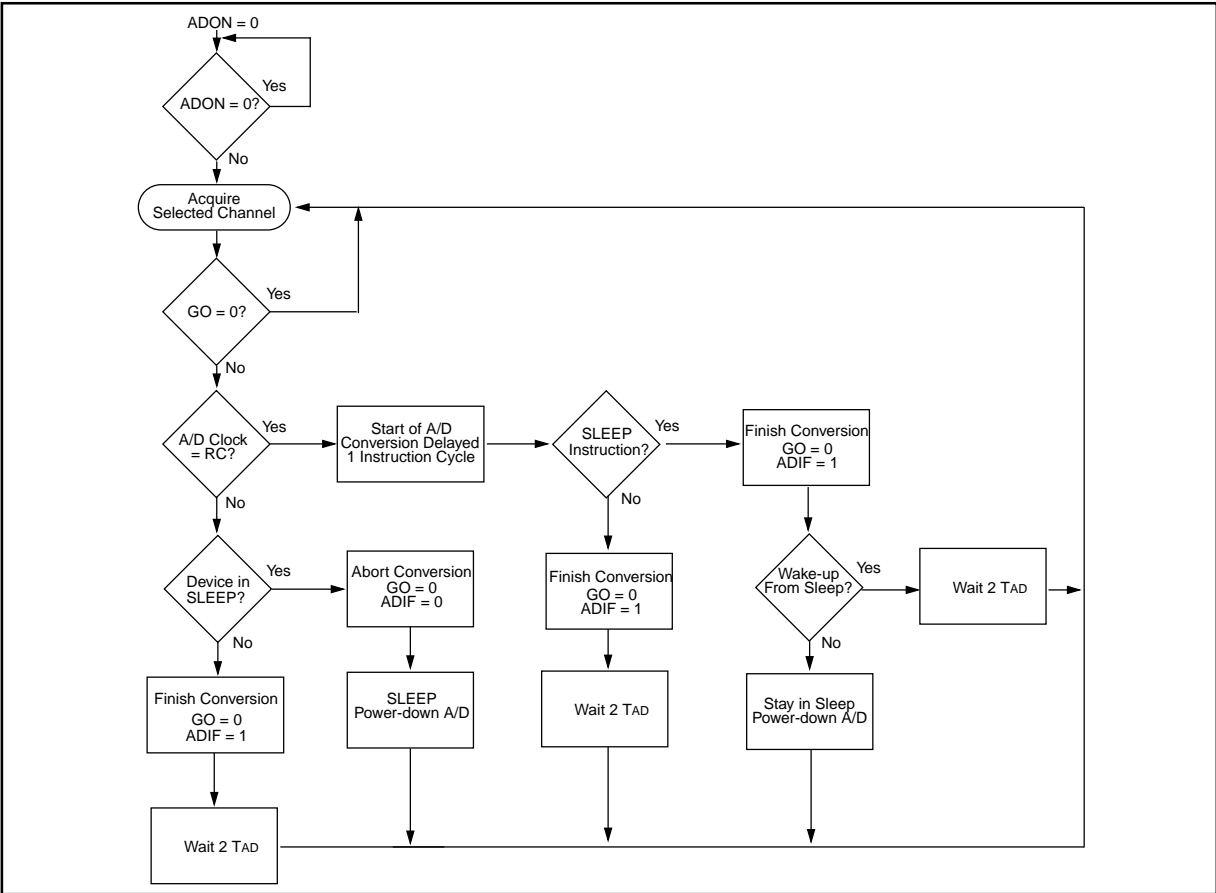


FIGURE 7-7: FLOWCHART OF A/D OPERATION



PIC16C71X

TABLE 7-3: REGISTERS/BITS ASSOCIATED WITH A/D, PIC16C710/71/711

| 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 | INTCON | GIE | ADIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 89h | ADRES | A/D Result Register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 08h | ADCON0 | ADCS1 | ADCS0 | — | CHS1 | CHS0 | GO/DONE | ADIF | ADON | 00-0 0000 | 00-0 0000 |
| 88h | ADCON1 | — | — | — | — | — | — | PCFG1 | PCFG0 | ---- --00 | ---- --00 |
| 05h | PORTA | — | — | — | RA4 | RA3 | RA2 | RA1 | RA0 | ---x 0000 | ---u 0000 |
| 85h | TRISA | — | — | — | PORTA Data Direction Register | | | | | ---1 1111 | ---1 1111 |

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

TABLE 7-4: REGISTERS/BITS ASSOCIATED WITH A/D, PIC16C715

| 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 | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | ADIF | — | — | — | — | — | — | -0-- ---- | -0-- ---- |
| 8Ch | PIE1 | — | ADIE | — | — | — | — | — | — | -0-- ---- | -0-- ---- |
| 1Eh | ADRES | A/D Result Register | | | | | | | | xxxx xxxx | uuuu uuuu |
| 1Fh | ADCON0 | ADCS1 | ADCS0 | CHS2 | CHS1 | CHS0 | GO/DONE | — | ADON | 0000 00-0 | 0000 00-0 |
| 9Fh | ADCON1 | — | — | — | — | — | — | PCFG1 | PCFG0 | ---- --00 | ---- --00 |
| 05h | PORTA | — | — | — | RA4 | RA3 | RA2 | RA1 | RA0 | ---x 0000 | ---u 0000 |
| 85h | TRISA | — | — | — | TRISA4 | TRISA3 | TRISA2 | TRISA1 | TRISA0 | ---1 1111 | ---1 1111 |

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

8.0 SPECIAL FEATURES OF THE CPU

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

What sets a microcontroller apart from other processors are special circuits to deal with the needs of real-time applications. The PIC16CXX family has a host of such 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) (PIC16C710/711/715)
 - Parity Error Reset (PER) (PIC16C715)
- Interrupts
- Watchdog Timer (WDT)
- SLEEP
- Code protection
- ID locations
- In-circuit serial programming

The PIC16CXX has 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 only, 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 RC oscillator option saves system cost while the LP crystal option saves power. A set of configuration bits are used to select various options.

8.1 Configuration Bits

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

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

FIGURE 8-1: CONFIGURATION WORD FOR PIC16C71

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|-----|-------|------|-------|-------|------------------|
| — | — | — | — | — | — | — | — | — | CP0 | PWRTE | WDTE | FOSC1 | FOSC0 | Register: CONFIG |
| bit13 | | | | | | | | | | | | | bit0 | Address 2007h |
| bit 13-5: Unimplemented: Read as '1' | | | | | | | | | | | | | | |
| bit 4: CP0: Code protection bit | | | | | | | | | | | | | | |
| 1 = Code protection off | | | | | | | | | | | | | | |
| 0 = All memory is code protected, but 00h - 3Fh is writable | | | | | | | | | | | | | | |
| bit 3: PWRTE: Power-up Timer Enable bit | | | | | | | | | | | | | | |
| 1 = Power-up Timer enabled | | | | | | | | | | | | | | |
| 0 = Power-up Timer disabled | | | | | | | | | | | | | | |
| bit 2: WDTE: Watchdog Timer Enable bit | | | | | | | | | | | | | | |
| 1 = WDT enabled | | | | | | | | | | | | | | |
| 0 = WDT disabled | | | | | | | | | | | | | | |
| bit 1-0: FOSC1:FOSC0: Oscillator Selection bits | | | | | | | | | | | | | | |
| 11 = RC oscillator | | | | | | | | | | | | | | |
| 10 = HS oscillator | | | | | | | | | | | | | | |
| 01 = XT oscillator | | | | | | | | | | | | | | |
| 00 = LP oscillator | | | | | | | | | | | | | | |

PIC16C71X

FIGURE 8-2: CONFIGURATION WORD, PIC16C710/711

| CP0 | CP0 | CP0 | CP0 | CP0 | CP0 | CP0 | BODEN | CP0 | CP0 | PWRT ¹ | WDTE | FOSC1 | FOSC0 | Register: CONFIG Address 2007h |
|---|-----|-----|-----|-----|-----|-----|-------|-----|-----|-------------------|------|-------|-------|-----------------------------------|
| bit13 | | | | | | | | | | | | | bit0 | |
| bit 13-7 CP0: Code protection bits ⁽²⁾ 5-4: 1 = Code protection off 0 = All memory is code protected, but 00h - 3Fh is writable bit 6: BODEN: Brown-out Reset Enable bit ⁽¹⁾ 1 = BOR enabled 0 = BOR disabled bit 3: PWRT¹: Power-up Timer Enable bit ⁽¹⁾ 1 = PWRT disabled 0 = PWRT enabled bit 2: WDTE: Watchdog Timer Enable bit 1 = WDT enabled 0 = WDT disabled bit 1-0: FOSC1:FOSC0: Oscillator Selection bits 11 = RC oscillator 10 = HS oscillator 01 = XT oscillator 00 = LP oscillator | | | | | | | | | | | | | | |
| Note 1: Enabling Brown-out Reset automatically enables Power-up Timer (PWRT) regardless of the value of bit PWRT ¹ . Ensure the Power-up Timer is enabled anytime Brown-out Reset is enabled. Note 2: All of the CP0 bits have to be given the same value to enable the code protection scheme listed. | | | | | | | | | | | | | | |

FIGURE 8-3: CONFIGURATION WORD, PIC16C715

| CP1 | CP0 | CP1 | CP0 | CP1 | CP0 | MPEEN | BODEN | CP1 | CP0 | PWRT ¹ | WDTE | FOSC1 | FOSC0 | Register: CONFIG Address 2007h |
|---|-----|-----|-----|-----|-----|-------|-------|-----|-----|-------------------|------|-------|-------|-----------------------------------|
| bit13 | | | | | | | | | | | | | bit0 | |
| bit 13-8 CP1:CP0: Code Protection bits ⁽²⁾ 5-4: 11 = Code protection off 10 = Upper half of program memory code protected 01 = Upper 3/4th of program memory code protected 00 = All memory is code protected bit 7: MPEEN: Memory Parity Error Enable 1 = Memory Parity Checking is enabled 0 = Memory Parity Checking is disabled bit 6: BODEN: Brown-out Reset Enable bit ⁽¹⁾ 1 = BOR enabled 0 = BOR disabled bit 3: PWRT¹: Power-up Timer Enable bit ⁽¹⁾ 1 = PWRT disabled 0 = PWRT enabled bit 2: WDTE: Watchdog Timer Enable bit 1 = WDT enabled 0 = WDT disabled bit 1-0: FOSC1:FOSC0: Oscillator Selection bits 11 = RC oscillator 10 = HS oscillator 01 = XT oscillator 00 = LP oscillator | | | | | | | | | | | | | | |
| Note 1: Enabling Brown-out Reset automatically enables Power-up Timer (PWRT) regardless of the value of bit PWRT ¹ . Ensure the Power-up Timer is enabled anytime Brown-out Reset is enabled. Note 2: All of the CP1:CP0 pairs have to be given the same value to enable the code protection scheme listed. | | | | | | | | | | | | | | |

8.2 Oscillator Configurations

8.2.1 OSCILLATOR TYPES

The PIC16CXX can be operated in four different oscillator modes. The user can program two configuration bits (FOSC1 and FOSC0) to select one of these four modes:

- LP Low Power Crystal
- XT Crystal/Resonator
- HS High Speed Crystal/Resonator
- RC Resistor/Capacitor

8.2.2 CRYSTAL OSCILLATOR/CERAMIC RESONATORS

In XT, LP or HS modes a crystal or ceramic resonator is connected to the OSC1/CLKIN and OSC2/CLKOUT pins to establish oscillation (Figure 8-4). The PIC16CXX 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. When in XT, LP or HS modes, the device can have an external clock source to drive the OSC1/CLKIN pin (Figure 8-5).

FIGURE 8-4: CRYSTAL/CERAMIC RESONATOR OPERATION (HS, XT OR LP OSC CONFIGURATION)

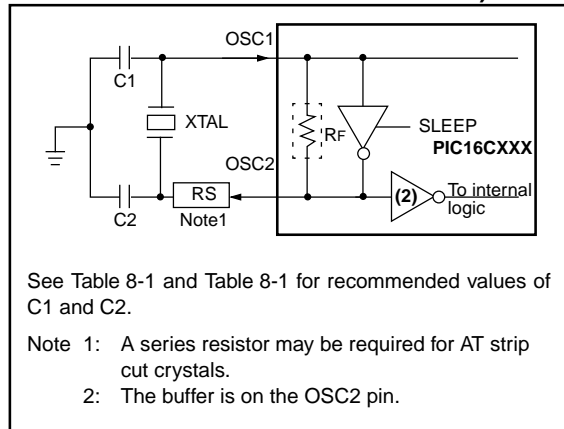


FIGURE 8-5: EXTERNAL CLOCK INPUT OPERATION (HS, XT OR LP OSC CONFIGURATION)

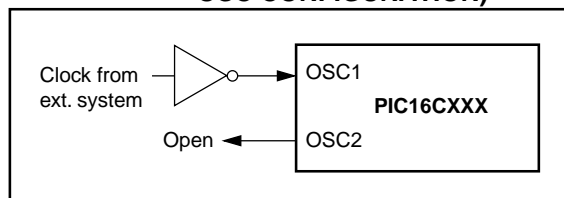


TABLE 8-1: CERAMIC RESONATORS, PIC16C71

| Ranges Tested: | | | |
|---|------------------------|-------------|-------------|
| Mode | Freq | OSC1 | OSC2 |
| XT | 455 kHz | 47 - 100 pF | 47 - 100 pF |
| | 2.0 MHz | 15 - 68 pF | 15 - 68 pF |
| | 4.0 MHz | 15 - 68 pF | 15 - 68 pF |
| HS | 8.0 MHz | 15 - 68 pF | 15 - 68 pF |
| | 16.0 MHz | 10 - 47 pF | 10 - 47 pF |
| These values are for design guidance only. See notes at bottom of page. | | | |
| Resonators Used: | | | |
| 455 kHz | Panasonic EFO-A455K04B | ± 0.3% | |
| 2.0 MHz | Murata Erie CSA2.00MG | ± 0.5% | |
| 4.0 MHz | Murata Erie CSA4.00MG | ± 0.5% | |
| 8.0 MHz | Murata Erie CSA8.00MT | ± 0.5% | |
| 16.0 MHz | Murata Erie CSA16.00MX | ± 0.5% | |
| All resonators used did not have built-in capacitors. | | | |

TABLE 8-2: CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR, PIC16C71

| Mode | Freq | OSC1 | OSC2 |
|---|---------|-------------|-------------|
| LP | 32 kHz | 33 - 68 pF | 33 - 68 pF |
| | 200 kHz | 15 - 47 pF | 15 - 47 pF |
| XT | 100 kHz | 47 - 100 pF | 47 - 100 pF |
| | 500 kHz | 20 - 68 pF | 20 - 68 pF |
| | 1 MHz | 15 - 68 pF | 15 - 68 pF |
| | 2 MHz | 15 - 47 pF | 15 - 47 pF |
| | 4 MHz | 15 - 33 pF | 15 - 33 pF |
| HS | 8 MHz | 15 - 47 pF | 15 - 47 pF |
| | 20 MHz | 15 - 47 pF | 15 - 47 pF |
| These values are for design guidance only. See notes at bottom of page. | | | |

PIC16C71X

**TABLE 8-3: CERAMIC RESONATORS,
PIC16C710/711/715**

| 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. | | | |
| Resonators Used: | | | |
| 455 kHz | Panasonic EFO-A455K04B | ± 0.3% | |
| 2.0 MHz | Murata Erie CSA2.00MG | ± 0.5% | |
| 4.0 MHz | Murata Erie CSA4.00MG | ± 0.5% | |
| 8.0 MHz | Murata Erie CSA8.00MT | ± 0.5% | |
| 16.0 MHz | Murata Erie CSA16.00MX | ± 0.5% | |
| All resonators used did not have built-in capacitors. | | | |

**TABLE 8-4: CAPACITOR SELECTION
FOR CRYSTAL OSCILLATOR,
PIC16C710/711/715**

| 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. | | | |
| Crystals Used | | | |
| 32 kHz | Epson C-001R32.768K-A | ± 20 PPM | |
| 200 kHz | STD XTL 200.000KHz | ± 20 PPM | |
| 1 MHz | ECS ECS-10-13-1 | ± 50 PPM | |
| 4 MHz | ECS ECS-40-20-1 | ± 50 PPM | |
| 8 MHz | EPSON CA-301 8.000M-C | ± 30 PPM | |
| 20 MHz | EPSON CA-301 20.000M-C | ± 30 PPM | |

- Note 1: Recommended values of C1 and C2 are identical to the ranges tested table.
- 2: Higher capacitance increases the stability of oscillator but also increases the start-up time.
- 3: Since each resonator/crystal has its own characteristics, the user should consult the resonator/crystal manufacturer for appropriate values of external components.
- 4: Rs may be required in HS mode as well as XT mode to avoid overdriving crystals with low drive level specification.

8.2.3 EXTERNAL CRYSTAL OSCILLATOR CIRCUIT

Either a prepackaged oscillator can be used or a simple oscillator circuit with TTL gates can be built. Prepackaged oscillators provide a wide operating range and better stability. A well-designed crystal oscillator will provide good performance with TTL gates. Two types of crystal oscillator circuits can be used; one with series resonance, or one with parallel resonance.

Figure 8-6 shows implementation of a parallel resonant oscillator circuit. The circuit is designed to use the fundamental frequency of the crystal. The 74AS04 inverter performs the 180-degree phase shift that a parallel oscillator requires. The 4.7 k Ω resistor provides the negative feedback for stability. The 10 k Ω potentiometer biases the 74AS04 in the linear region. This could be used for external oscillator designs.

FIGURE 8-6: EXTERNAL PARALLEL RESONANT CRYSTAL OSCILLATOR CIRCUIT

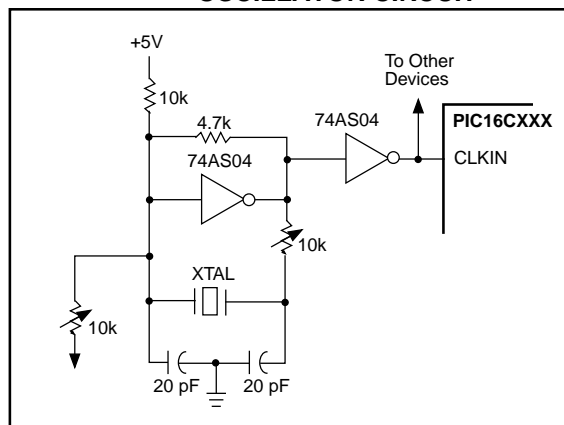
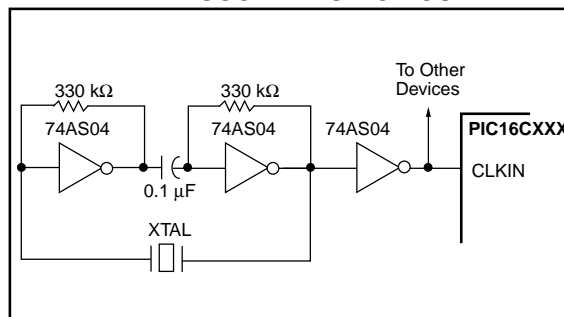


Figure 8-7 shows a series resonant oscillator circuit. This circuit is also designed to use the fundamental frequency of the crystal. The inverter performs a 180-degree phase shift in a series resonant oscillator circuit. The 330 k Ω resistors provide the negative feedback to bias the inverters in their linear region.

FIGURE 8-7: EXTERNAL SERIES RESONANT CRYSTAL OSCILLATOR CIRCUIT



8.2.4 RC OSCILLATOR

For timing insensitive applications the "RC" device option offers additional cost savings. The RC oscillator frequency is a function of the supply voltage, the resistor (R_{ext}) and capacitor (C_{ext}) values, and the operating temperature. In addition to this, the oscillator frequency will vary from unit to unit due to normal process parameter variation. Furthermore, the difference in lead frame capacitance between package types will also affect the oscillation frequency, especially for low C_{ext} values. The user also needs to take into account variation due to tolerance of external R and C components used. Figure 8-8 shows how the R/C combination is connected to the PIC16CXX. For R_{ext} values below 2.2 k Ω , the oscillator operation may become unstable, or stop completely. For very high R_{ext} values (e.g. 1 M Ω), the oscillator becomes sensitive to noise, humidity and leakage. Thus, we recommend to keep R_{ext} between 3 k Ω and 100 k Ω .

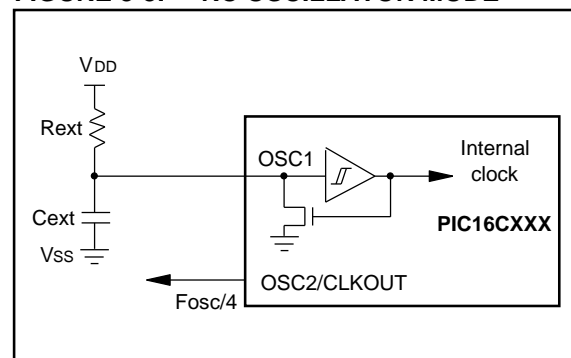
Although the oscillator will operate with no external capacitor ($C_{ext} = 0$ pF), we recommend using values above 20 pF for noise and stability reasons. With no or small external capacitance, the oscillation frequency can vary dramatically due to changes in external capacitances, such as PCB trace capacitance or package lead frame capacitance.

See characterization data for desired device for RC frequency variation from part to part due to normal process variation. The variation is larger for larger R (since leakage current variation will affect RC frequency more for large R) and for smaller C (since variation of input capacitance will affect RC frequency more).

See characterization data for desired device for variation of oscillator frequency due to V_{DD} for given R_{ext}/C_{ext} values as well as frequency variation due to operating temperature for given R, C, and V_{DD} values.

The oscillator frequency, divided by 4, is available on the OSC2/CLKOUT pin, and can be used for test purposes or to synchronize other logic (see Figure 3-2 for waveform).

FIGURE 8-8: RC OSCILLATOR MODE



PIC16C71X

8.3 Reset

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

The PIC16CXX differentiates between various kinds of reset:

- Power-on Reset (POR)
- $\overline{\text{MCLR}}$ reset during normal operation
- $\overline{\text{MCLR}}$ reset during SLEEP
- WDT Reset (normal operation)
- Brown-out Reset (BOR) (PIC16C710/711/715)
- Parity Error Reset (PIC16C715)

Some registers are not affected in any reset condition; their status is unknown on POR and unchanged in any other reset. Most other registers are reset to a "reset state" on Power-on Reset (POR), on the $\overline{\text{MCLR}}$ and

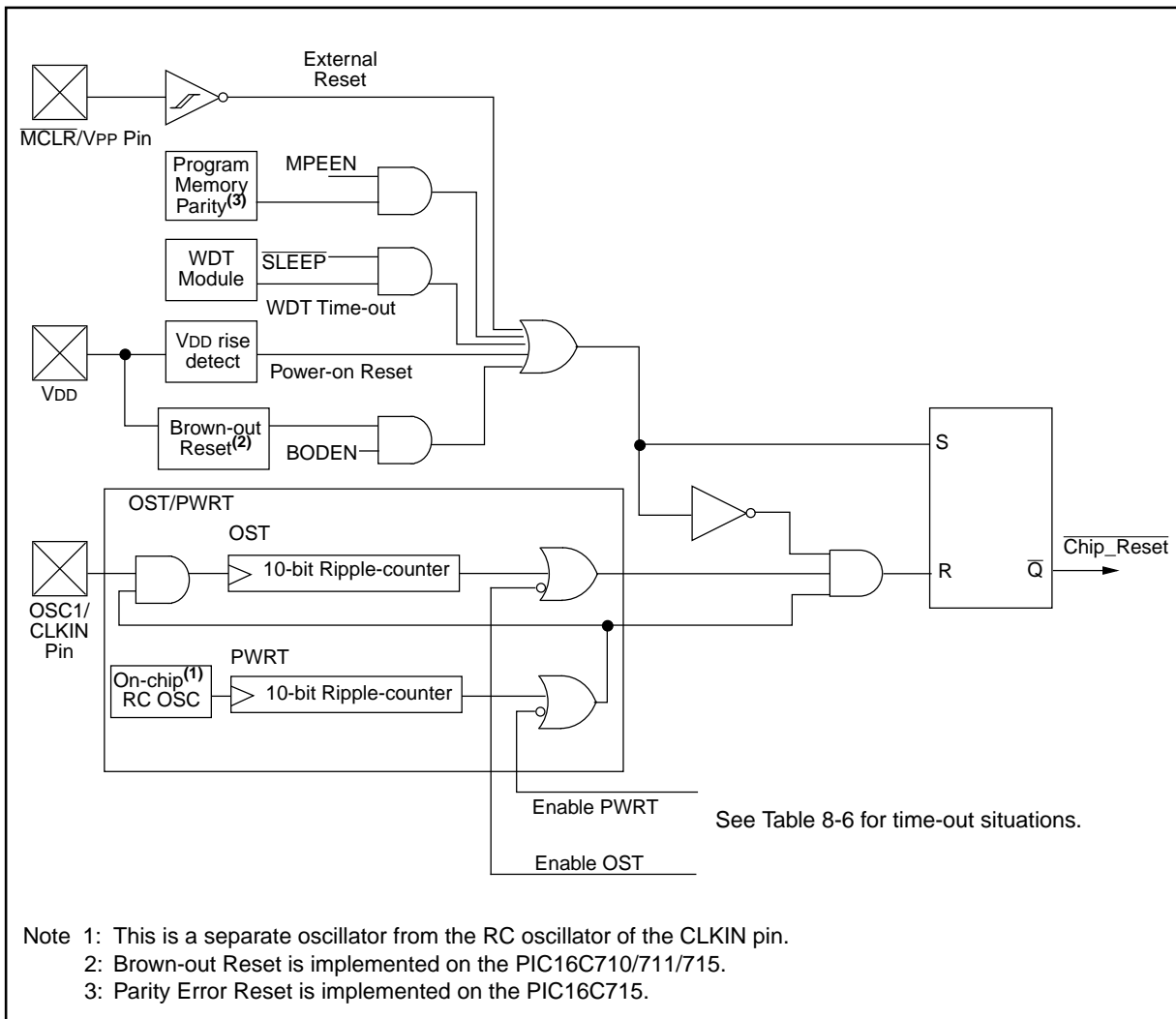
WDT Reset, on $\overline{\text{MCLR}}$ reset during SLEEP, and Brown-out Reset (BOR). They are not affected by a WDT Wake-up, which is viewed as the resumption of normal operation. The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits are set or cleared differently in different reset situations as indicated in Table 8-7, Table 8-8 and Table 8-9. These bits are used in software to determine the nature of the reset. See Table 8-10 and Table 8-11 for a full description of reset states of all registers.

A simplified block diagram of the on-chip reset circuit is shown in Figure 8-9.

The PIC16C710/711/715 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 8-9: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT



8.4 Power-on Reset (POR), Power-up Timer (PWRT) and Oscillator Start-up Timer (OST), and Brown-out Reset (BOR)

8.4.1 POWER-ON RESET (POR)

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

A Power-on Reset pulse is generated on-chip when VDD rise is detected (in the range of 1.5V - 2.1V). To take advantage of the POR, just tie the MCLR pin directly (or through a resistor) to VDD. This will eliminate external RC components usually needed to create a Power-on Reset. A maximum rise time for VDD is specified. See Electrical Specifications for details.

When the device starts normal operation (exits the reset condition), device operating parameters (voltage, frequency, temperature, ...) must be met to ensure operation. If these conditions are not met, the device must be held in reset until the operating conditions are met. Brown-out Reset may be used to meet the startup conditions.

For additional information, refer to Application Note AN607, "Power-up Trouble Shooting."

8.4.2 POWER-UP TIMER (PWRT)

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

The Power-up Timer provides a fixed 72 ms nominal time-out on power-up only, from the POR. 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 allows VDD to rise to an acceptable level. A configuration bit is provided to enable/disable the PWRT.

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

8.4.3 OSCILLATOR START-UP TIMER (OST)

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

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.

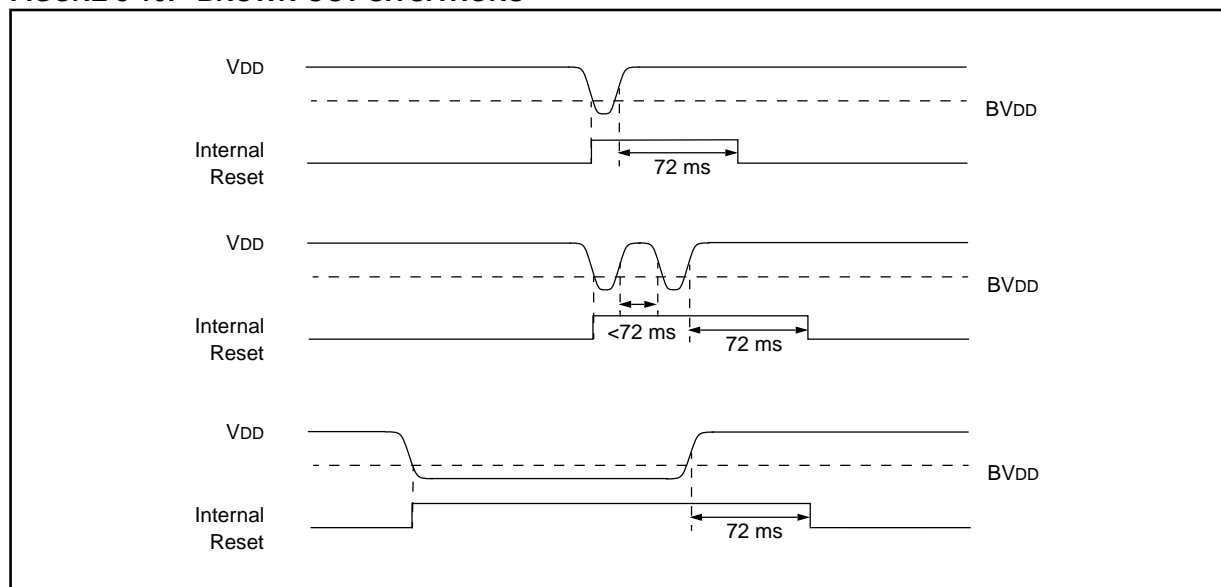
The OST time-out is invoked only for XT, LP and HS modes and only on Power-on Reset or wake-up from SLEEP.

8.4.4 BROWN-OUT RESET (BOR)

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

A configuration bit, BODEN, can disable (if clear/programmed) or enable (if set) the Brown-out Reset circuitry. If VDD falls below 4.0V (3.8V - 4.2V range) for greater than parameter #35, the brown-out situation will reset the chip. A reset may not occur if VDD falls below 4.0V for less than parameter #35. The chip will remain in Brown-out Reset until VDD rises above BVDD. The Power-up Timer will now be invoked and will keep the chip in RESET an additional 72 ms. If VDD drops below BVDD while the Power-up Timer is running, the chip will go back into a Brown-out Reset and the Power-up Timer will be initialized. Once VDD rises above BVDD, the Power-up Timer will execute a 72 ms time delay. The Power-up Timer should always be enabled when Brown-out Reset is enabled. Figure 8-10 shows typical brown-out situations.

FIGURE 8-10: BROWN-OUT SITUATIONS



PIC16C71X

8.4.5 TIME-OUT SEQUENCE

| | | | | |
|---------------------------|-----|----|-----|-----|
| Applicable Devices | 710 | 71 | 711 | 715 |
|---------------------------|-----|----|-----|-----|

On power-up the time-out sequence is as follows: First PWRT time-out is invoked after the POR time delay has expired. Then OST 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 8-11, Figure 8-12, and Figure 8-13 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 8-12). This is useful for testing purposes or to synchronize more than one PIC16CXX device operating in parallel.

Table 8-10 and Table 8-11 show the reset conditions for some special function registers, while Table 8-12 and Table 8-13 show the reset conditions for all the registers.

8.4.6 POWER CONTROL/STATUS REGISTER (PCON)

| | | | | |
|---------------------------|-----|----|-----|-----|
| Applicable Devices | 710 | 71 | 711 | 715 |
|---------------------------|-----|----|-----|-----|

The Power Control/Status Register, PCON has up to two bits, depending upon the device.

Bit0 is Brown-out Reset Status bit, $\overline{\text{BOR}}$. Bit $\overline{\text{BOR}}$ is unknown on a Power-on Reset. It must then be set by the user and checked on subsequent resets to see if bit $\overline{\text{BOR}}$ cleared, indicating a BOR occurred. The $\overline{\text{BOR}}$ bit is a "Don't Care" bit and is not necessarily predictable if the Brown-out Reset circuitry is disabled (by clearing bit BODEN in the Configuration Word).

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.

For the PIC16C715, bit2 is $\overline{\text{PER}}$ (Parity Error Reset). It is cleared on a Parity Error Reset and must be set by user software. It will also be set on a Power-on Reset.

For the PIC16C715, bit7 is MPEEN (Memory Parity Error Enable). This bit reflects the status of the MPEEN bit in configuration word. It is unaffected by any reset of interrupt.

8.4.7 PARITY ERROR RESET (PER)

| | | | | |
|---------------------------|-----|----|-----|-----|
| Applicable Devices | 710 | 71 | 711 | 715 |
|---------------------------|-----|----|-----|-----|

The PIC16C715 has on-chip parity bits that can be used to verify the contents of program memory. Parity bits may be useful in applications in order to increase overall reliability of a system.

There are two parity bits for each word of Program Memory. The parity bits are computed on alternating bits of the program word. One computation is performed using even parity, the other using odd parity. As a program executes, the parity is verified. The even parity bit is XOR'd with the even bits in the program memory word. The odd parity bit is negated and XOR'd with the odd bits in the program memory word. When an error is detected, a reset is generated and the $\overline{\text{PER}}$ flag bit 2 in the PCON register is cleared (logic '0'). This indication can allow software to act on a failure. However, there is no indication of the program memory location of the failure in Program Memory. This flag can only be set (logic '1') by software.

The parity array is user selectable during programming. Bit 7 of the configuration word located at address 2007h can be programmed (read as '0') to disable parity. If left unprogrammed (read as '1'), parity is enabled.

TABLE 8-5: TIME-OUT IN VARIOUS SITUATIONS, PIC16C71

| Oscillator Configuration | Power-up | | Wake-up from SLEEP |
|--------------------------|------------------|-----------|--------------------|
| | PWRTE = 1 | PWRTE = 0 | |
| XT, HS, LP | 72 ms + 1024Tosc | 1024Tosc | 1024Tosc |
| RC | 72 ms | — | — |

TABLE 8-6: TIME-OUT IN VARIOUS SITUATIONS, PIC16C710/711/715

| Oscillator Configuration | Power-up | | Brown-out | Wake-up from SLEEP |
|--------------------------|------------------|-----------|------------------|--------------------|
| | PWRTE = 0 | PWRTE = 1 | | |
| XT, HS, LP | 72 ms + 1024Tosc | 1024Tosc | 72 ms + 1024Tosc | 1024Tosc |
| RC | 72 ms | — | 72 ms | — |

TABLE 8-7: STATUS BITS AND THEIR SIGNIFICANCE, PIC16C71

| \overline{TO} | \overline{PD} | |
|-----------------|-----------------|--|
| 1 | 1 | Power-on Reset |
| 0 | x | Illegal, \overline{TO} is set on \overline{POR} |
| x | 0 | Illegal, \overline{PD} is set on \overline{POR} |
| 0 | 1 | WDT Reset |
| 0 | 0 | WDT Wake-up |
| u | u | \overline{MCLR} Reset during normal operation |
| 1 | 0 | \overline{MCLR} Reset during SLEEP or interrupt wake-up from SLEEP |

TABLE 8-8: STATUS BITS AND THEIR SIGNIFICANCE, PIC16C710/711

| \overline{POR} | \overline{BOR} | \overline{TO} | \overline{PD} | |
|------------------|------------------|-----------------|-----------------|--|
| 0 | x | 1 | 1 | Power-on Reset |
| 0 | x | 0 | x | Illegal, \overline{TO} is set on \overline{POR} |
| 0 | x | x | 0 | Illegal, \overline{PD} is set on \overline{POR} |
| 1 | 0 | x | x | Brown-out Reset |
| 1 | 1 | 0 | 1 | WDT Reset |
| 1 | 1 | 0 | 0 | WDT Wake-up |
| 1 | 1 | u | u | \overline{MCLR} Reset during normal operation |
| 1 | 1 | 1 | 0 | \overline{MCLR} Reset during SLEEP or interrupt wake-up from SLEEP |

TABLE 8-9: STATUS BITS AND THEIR SIGNIFICANCE, PIC16C715

| \overline{PER} | \overline{POR} | \overline{BOR} | \overline{TO} | \overline{PD} | |
|------------------|------------------|------------------|-----------------|-----------------|--|
| 1 | 0 | x | 1 | 1 | Power-on Reset |
| x | 0 | x | 0 | x | Illegal, \overline{TO} is set on \overline{POR} |
| x | 0 | x | x | 0 | Illegal, \overline{PD} is set on \overline{POR} |
| 1 | 1 | 0 | x | x | Brown-out Reset |
| 1 | 1 | 1 | 0 | 1 | WDT Reset |
| 1 | 1 | 1 | 0 | 0 | WDT Wake-up |
| 1 | 1 | 1 | u | u | \overline{MCLR} Reset during normal operation |
| 1 | 1 | 1 | 1 | 0 | \overline{MCLR} Reset during SLEEP or interrupt wake-up from SLEEP |
| 0 | 1 | 1 | 1 | 1 | Parity Error Reset |
| 0 | 0 | x | x | x | Illegal, \overline{PER} is set on \overline{POR} |
| 0 | x | 0 | x | x | Illegal, \overline{PER} is set on \overline{BOR} |

PIC16C71X

TABLE 8-10: RESET CONDITION FOR SPECIAL REGISTERS, PIC16C710/711/711

| Condition | Program Counter | STATUS Register | PCON Register PIC16C710/711 |
|------------------------------------|-----------------------|-----------------|--------------------------------|
| Power-on Reset | 000h | 0001 1xxx | ---- --0x |
| MCLR Reset during normal operation | 000h | 000u uuuu | ---- --uu |
| MCLR Reset during SLEEP | 000h | 0001 0uuu | ---- --uu |
| WDT Reset | 000h | 0000 1uuu | ---- --uu |
| WDT Wake-up | PC + 1 | uuu0 0uuu | ---- --uu |
| Brown-out Reset (PIC16C710/711) | 000h | 0001 1uuu | ---- --u0 |
| Interrupt wake-up from SLEEP | PC + 1 ⁽¹⁾ | uuu1 0uuu | ---- --uu |

Legend: u = unchanged, x = unknown, - = unimplemented bit read as '0'.

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).

TABLE 8-11: RESET CONDITION FOR SPECIAL REGISTERS, PIC16C715

| Condition | Program Counter | STATUS Register | PCON Register |
|------------------------------------|-----------------------|-----------------|---------------|
| Power-on Reset | 000h | 0001 1xxx | u--- -10x |
| MCLR Reset during normal operation | 000h | 000u uuuu | u--- -uuu |
| MCLR Reset during SLEEP | 000h | 0001 0uuu | u--- -uuu |
| WDT Reset | 000h | 0000 1uuu | u--- -uuu |
| WDT Wake-up | PC + 1 | uuu0 0uuu | u--- -uuu |
| Brown-out Reset | 000h | 0001 1uuu | u--- -uu0 |
| Parity Error Reset | 000h | uuu1 0uuu | u--- -0uu |
| Interrupt wake-up from SLEEP | PC + 1 ⁽¹⁾ | uuu1 0uuu | u--- -uuu |

Legend: u = unchanged, x = unknown, - = unimplemented bit read as '0'.

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).

TABLE 8-12: INITIALIZATION CONDITIONS FOR ALL REGISTERS, PIC16C710/711/711

| Register | Power-on Reset, Brown-out Reset ⁽⁵⁾ | MCLR Resets WDT Reset | Wake-up via WDT or Interrupt |
|---------------------|---|--------------------------|------------------------------------|
| W | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| INDF | N/A | N/A | N/A |
| TMR0 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PCL | 0000h | 0000h | PC + 1 ⁽²⁾ |
| STATUS | 0001 1xxx | 000q quuu ⁽³⁾ | uuuq quuu ⁽³⁾ |
| FSR | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PORTA | ---x 0000 | ---u 0000 | ---u uuuu |
| PORTB | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PCLATH | ---0 0000 | ---0 0000 | ---u uuuu |
| INTCON | 0000 000x | 0000 000u | uuuu uuuu ⁽¹⁾ |
| ADRES | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| ADCON0 | 00-0 0000 | 00-0 0000 | uu-u uuuu |
| OPTION | 1111 1111 | 1111 1111 | uuuu uuuu |
| TRISA | ---1 1111 | ---1 1111 | ---u uuuu |
| TRISB | 1111 1111 | 1111 1111 | uuuu uuuu |
| PCON ⁽⁴⁾ | ---- --0u | ---- --uu | ---- --uu |
| ADCON1 | ---- --00 | ---- --00 | ---- --uu |

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition

Note 1: One or more bits in INTCON will be affected (to cause wake-up).

2: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

3: See Table 8-10 for reset value for specific condition.

4: The PCON register is not implemented on the PIC16C71.

5: Brown-out reset is not implemented on the PIC16C71.

PIC16C71X

TABLE 8-13: INITIALIZATION CONDITIONS FOR ALL REGISTERS, PIC16C715

| Register | Power-on Reset, Brown-out Reset Parity Error Reset | MCLR Resets WDT Reset | Wake-up via WDT or Interrupt |
|----------|--|--------------------------|------------------------------------|
| W | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| INDF | N/A | N/A | N/A |
| TMR0 | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PCL | 0000 0000 | 0000 0000 | PC + 1 ⁽²⁾ |
| STATUS | 0001 1xxx | 000q quuu ⁽³⁾ | uuuq quuu ⁽³⁾ |
| FSR | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PORTA | ---x 0000 | ---u 0000 | ---u uuuu |
| PORTB | xxxx xxxx | uuuu uuuu | uuuu uuuu |
| PCLATH | ---0 0000 | ---0 0000 | ---u uuuu |
| INTCON | 0000 000x | 0000 000u | uuuu uuuu ⁽¹⁾ |
| PIR1 | -0-- ---- | -0-- ---- | -u-- ---- ⁽¹⁾ |
| ADCON0 | 0000 00-0 | 0000 00-0 | uuuu uu-u |
| OPTION | 1111 1111 | 1111 1111 | uuuu uuuu |
| TRISA | ---1 1111 | ---1 1111 | ---u uuuu |
| TRISB | 1111 1111 | 1111 1111 | uuuu uuuu |
| PIE1 | -0-- ---- | -0-- ---- | -u-- ---- |
| PCON | ---- -qqq | ---- -1uu | ---- -1uu |
| ADCON1 | ---- --00 | ---- --00 | ---- --uu |

Legend: u = unchanged, x = unknown, - = unimplemented bit, read as '0', q = value depends on condition

Note 1: One or more bits in INTCON and PIR1 will be affected (to cause wake-up).

2: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

3: See Table 8-11 for reset value for specific condition.

FIGURE 8-11: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ NOT TIED TO V_{DD}): CASE 1

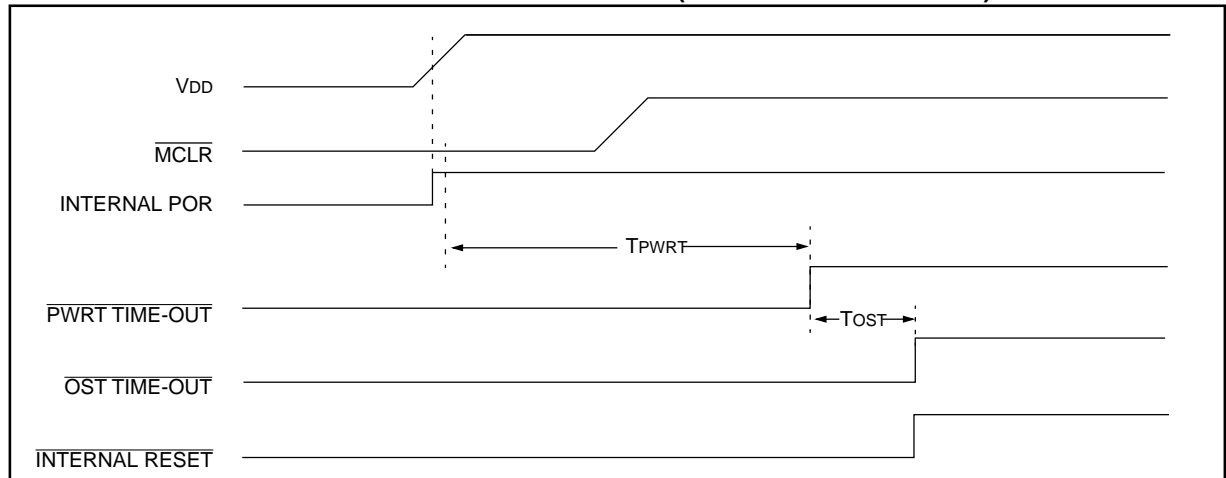


FIGURE 8-12: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ NOT TIED TO V_{DD}): CASE 2

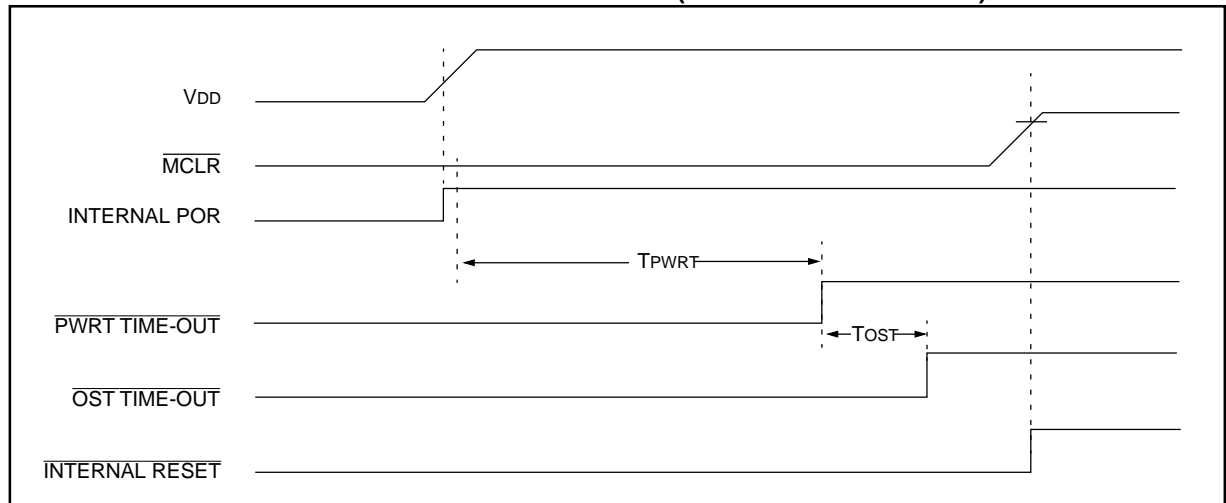
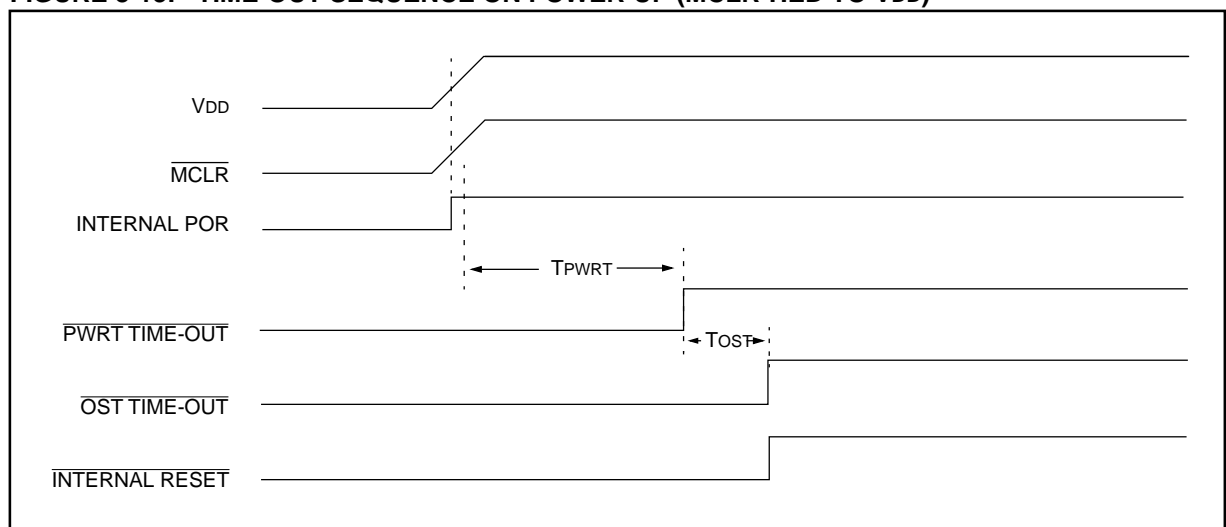


FIGURE 8-13: TIME-OUT SEQUENCE ON POWER-UP ($\overline{\text{MCLR}}$ TIED TO V_{DD})



PIC16C71X

FIGURE 8-14: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW VDD POWER-UP)

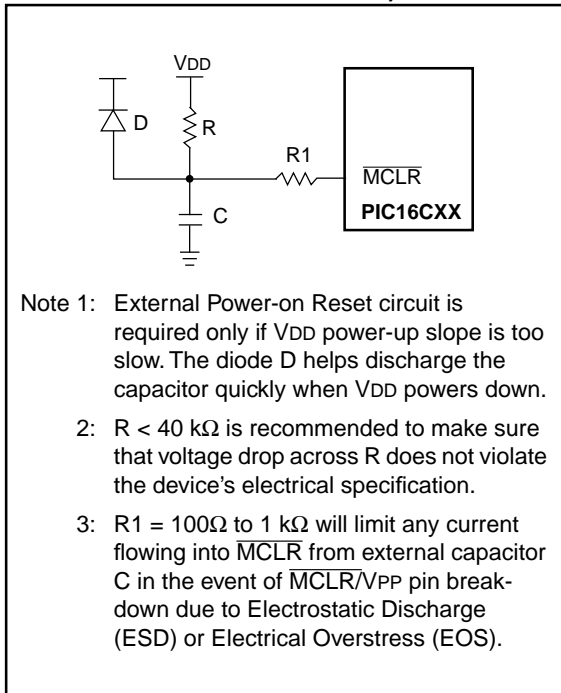


FIGURE 8-15: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 1

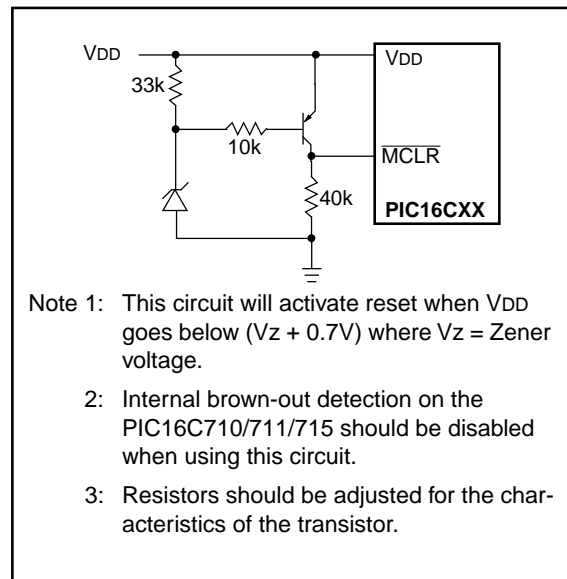
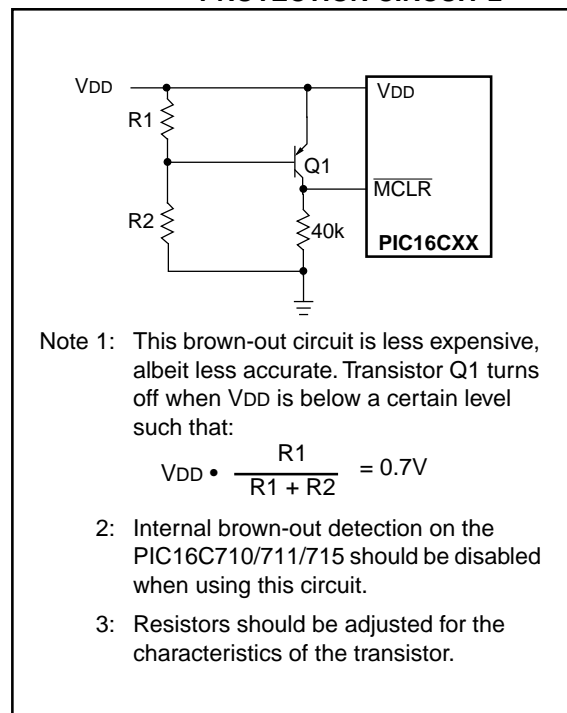


FIGURE 8-16: EXTERNAL BROWN-OUT PROTECTION CIRCUIT 2



8.5 Interrupts

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

The PIC16C71X family has 4 sources of interrupt.

| Interrupt Sources |
|--|
| External interrupt RB0/INT |
| TMR0 overflow interrupt |
| PORTB change interrupts (pins RB7:RB4) |
| A/D 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 (Figure 8-19). 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.

Note: For the PIC16C71

If an interrupt occurs while the Global Interrupt Enable (GIE) bit is being cleared, the GIE bit may unintentionally be re-enabled by the user's Interrupt Service Routine (the RETFIE instruction). The events that would cause this to occur are:

1. An instruction clears the GIE bit while an interrupt is acknowledged.
2. The program branches to the Interrupt vector and executes the Interrupt Service Routine.
3. The Interrupt Service Routine completes with the execution of the RETFIE instruction. This causes the GIE bit to be set (enables interrupts), and the program returns to the instruction after the one which was meant to disable interrupts.

Perform the following to ensure that interrupts are globally disabled:

```

LOOP BCF    INTCON, GIE    ; Disable global
                           ; interrupt bit
      BTFSC  INTCON, GIE    ; Global interrupt
                           ; disabled?
      GOTO   LOOP          ; NO, try again
      :                               ; Yes, continue
                           ; with program
                           ; flow

```

PIC16C71X

FIGURE 8-17: INTERRUPT LOGIC, PIC16C710, 71, 711

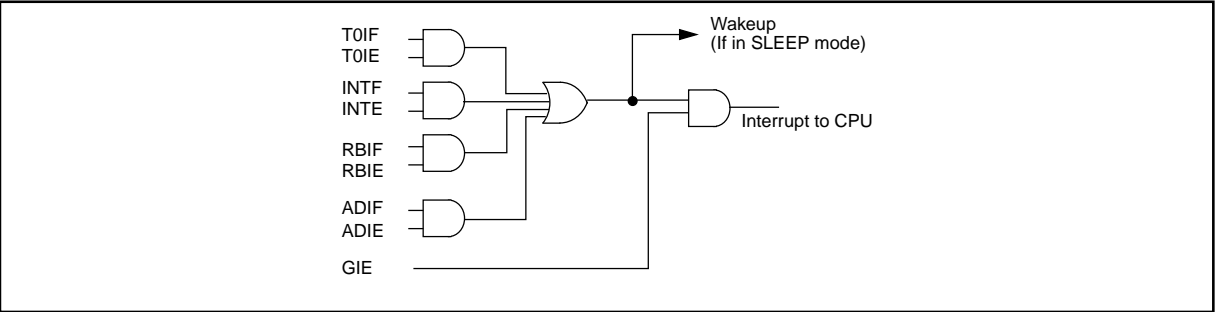
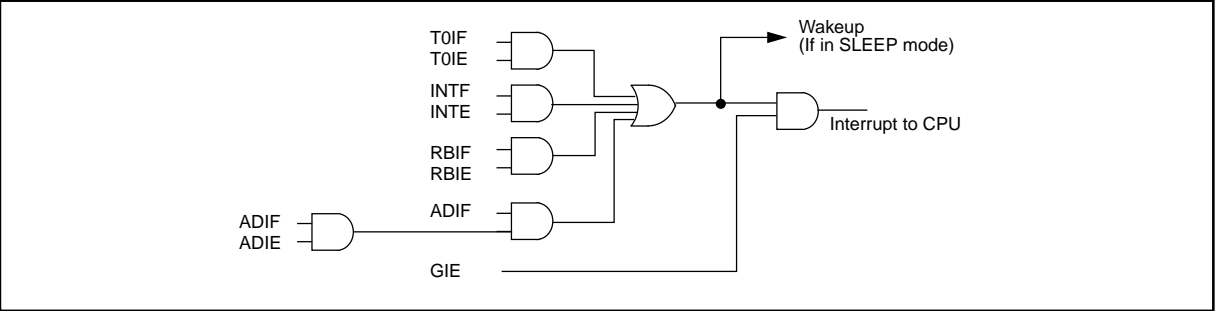


FIGURE 8-18: INTERRUPT LOGIC, PIC16C715



8.5.1 INT INTERRUPT

External interrupt on RB0/INT pin is edge triggered: either rising if bit INTEDG (OPTION<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 8.8 for details on SLEEP mode.

8.5.2 TMR0 INTERRUPT

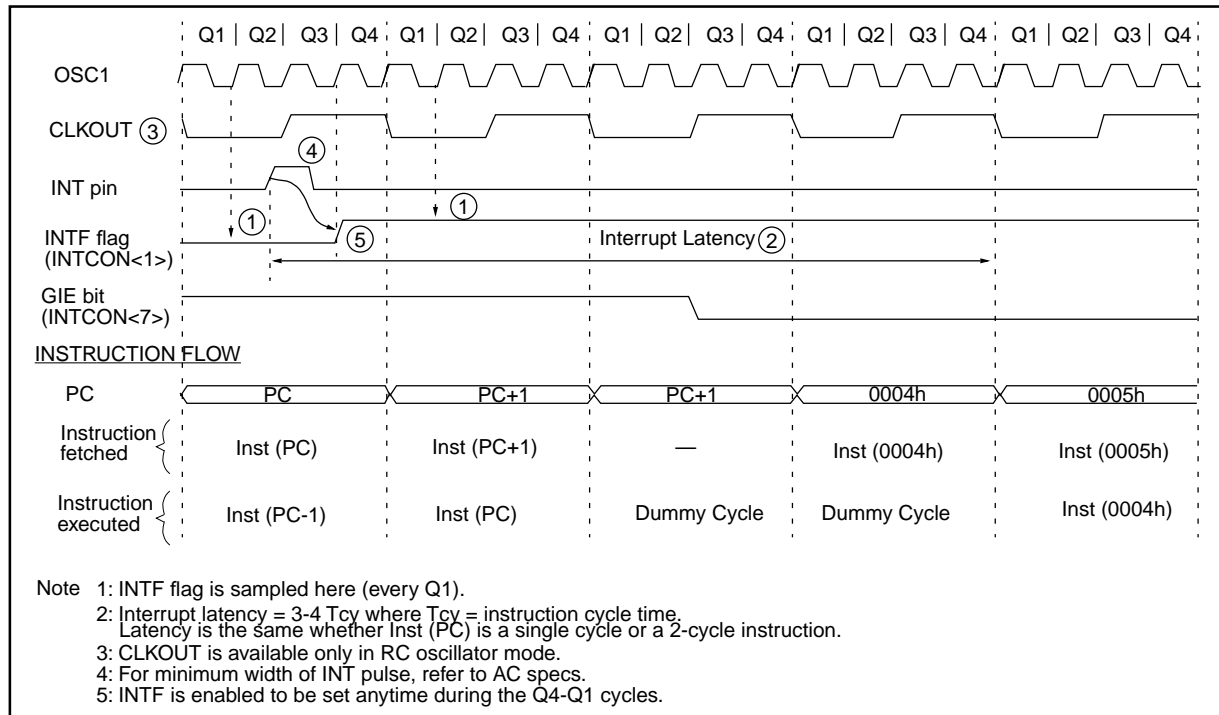
An overflow (FFh → 00h) in the TMR0 register will set flag bit T0IF (INTCON<2>). The interrupt can be enabled/disabled by setting/clearing enable bit T0IE (INTCON<5>). (Section 6.0)

8.5.3 PORTB INTCON CHANGE

An input change on PORTB<7:4> sets flag bit RBIF (INTCON<0>). The interrupt can be enabled/disabled by setting/clearing enable bit RBIE (INTCON<4>). (Section 5.2)

Note: For the PIC16C71 if a change on the I/O pin should occur when the read operation is being executed (start of the Q2 cycle), then the RBIF interrupt flag may not get set.

FIGURE 8-19: INT PIN INTERRUPT TIMING



PIC16C71X

8.6 Context Saving During Interrupts

During an interrupt, only the return PC value is saved on the stack. Typically, users may wish to save key registers during an interrupt i.e., W register and STATUS register. This will have to be implemented in software.

Example 8-1 stores and restores the STATUS and W registers. The user register, STATUS_TEMP, must be defined in bank 0.

The example:

- a) Stores the W register.
- b) Stores the STATUS register in bank 0.
- c) Executes the ISR code.
- d) Restores the STATUS register (and bank select bit).
- e) Restores the W register.

EXAMPLE 8-1: SAVING STATUS AND W REGISTERS IN RAM

```
MOVWF    W_TEMP          ;Copy W to TEMP register, could be bank one or zero
SWAPF    STATUS,W        ;Swap status to be saved into W
MOVWF    STATUS_TEMP     ;Save status to bank zero STATUS_TEMP register
:
: (ISR)
:
SWAPF    STATUS_TEMP,W   ;Swap STATUS_TEMP register into W
                        ;(sets bank to original state)
MOVWF    STATUS          ;Move W into STATUS register
SWAPF    W_TEMP,F        ;Swap W_TEMP
SWAPF    W_TEMP,W        ;Swap W_TEMP into W
```


8.7 Watchdog Timer (WDT)

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

The Watchdog Timer is as a free running on-chip RC oscillator which does not require any external components. This RC oscillator is separate from the RC oscillator of the OSC1/CLKIN pin. That means that the WDT will run, even if the clock on the OSC1/CLKIN and OSC2/CLKOUT pins 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 WDT can be permanently disabled by clearing configuration bit `WDTE` (Section 8.1).

8.7.1 WDT PERIOD

The WDT has a nominal time-out period of 18 ms, (with no prescaler). The time-out periods vary with temperature, V_{DD} and process variations from part to part (see DC specs). If longer time-out periods are desired, a prescaler with a division ratio of up to 1:128 can be

assigned to the WDT under software control by writing to the `OPTION` register. Thus, time-out periods up to 2.3 seconds can be realized.

The `CLRWDT` and `SLEEP` instructions clear the WDT and the postscaler, if assigned to the WDT, and prevent it from timing out and generating a device RESET condition.

The \overline{TO} bit in the `STATUS` register will be cleared upon a Watchdog Timer time-out.

8.7.2 WDT PROGRAMMING CONSIDERATIONS

It should also be taken into account that under worst case conditions (V_{DD} = Min., Temperature = Max., and max. WDT prescaler) it may take several seconds before a WDT time-out occurs.

Note: When a `CLRWDT` instruction is executed and the prescaler is assigned to the WDT, the prescaler count will be cleared, but the prescaler assignment is not changed.

FIGURE 8-20: WATCHDOG TIMER BLOCK DIAGRAM

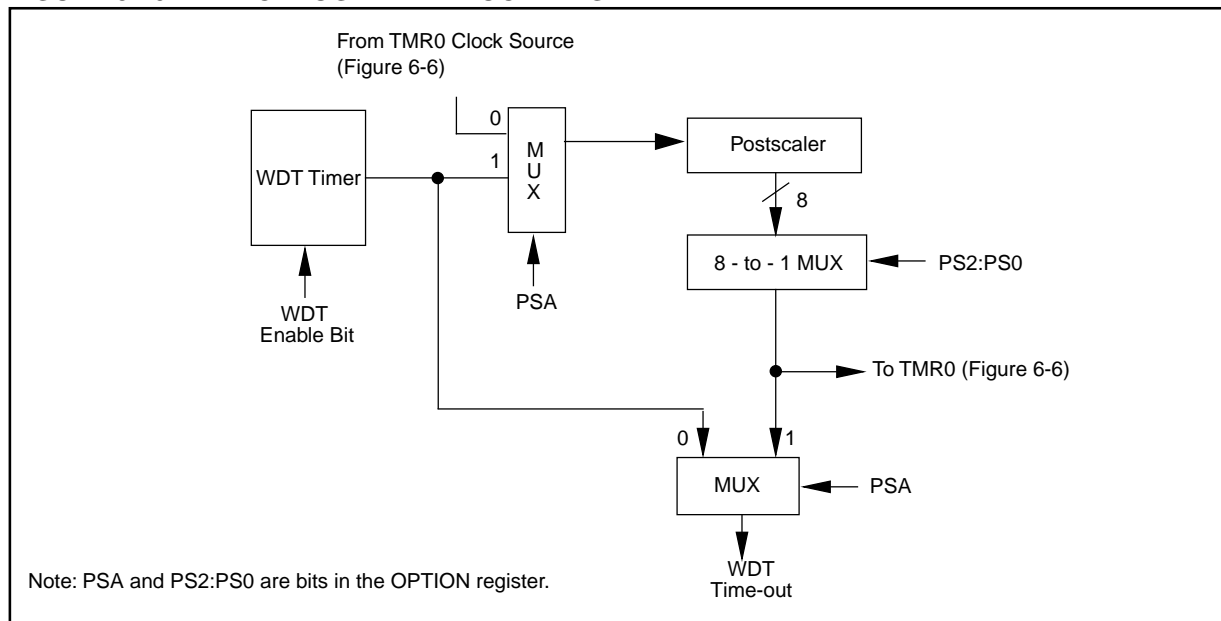


FIGURE 8-21: SUMMARY OF WATCHDOG TIMER REGISTERS

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|----------|--------------|-------|----------------------|-------|-------|----------------------|-------|-------|-------|
| 2007h | Config. bits | (1) | BODEN ⁽¹⁾ | CP1 | CP0 | PWRTE ⁽¹⁾ | WDTE | FOSC1 | FOSC0 |
| 81h,181h | OPTION | RBPUR | INTEDG | T0CS | T0SE | PSA | PS2 | PS1 | PS0 |

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

Note 1: See Figure 8-1, Figure 8-2 and Figure 8-3 for operation of these bits.

PIC16C71X

8.8 Power-down Mode (SLEEP)

Power-down mode is entered by executing a `SLEEP` instruction.

If enabled, the Watchdog Timer will be cleared but keeps running, the \overline{PD} bit (`STATUS<3>`) is cleared, the \overline{TO} (`STATUS<4>`) bit is set, and the oscillator driver is turned off. The I/O ports maintain the status they had, before the `SLEEP` instruction was executed (driving high, low, or hi-impedance).

For lowest current consumption in this mode, place all I/O pins at either 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.

The \overline{MCLR} pin must be at a logic high level (V_{IHMC}).

8.8.1 WAKE-UP FROM SLEEP

The device can wake up from `SLEEP` through one of the following events:

1. External reset input on \overline{MCLR} pin.
2. Watchdog Timer Wake-up (if `WDT` was enabled).
3. Interrupt from `INT` pin, `RB` port change, or some Peripheral Interrupts.

External \overline{MCLR} Reset will cause a device reset. All other events are considered a continuation of program execution and cause a "wake-up". The \overline{TO} and \overline{PD} bits in the `STATUS` register can be used to determine the cause of device reset. The \overline{PD} bit, which is set on power-up, is cleared when `SLEEP` is invoked. The \overline{TO} bit is cleared if a `WDT` time-out occurred (and caused wake-up).

The following peripheral interrupts can wake the device from `SLEEP`:

1. `TMR1` interrupt. `Timer1` must be operating as an asynchronous counter.
2. A/D conversion (when A/D clock source is `RC`).

Other peripherals cannot generate interrupts since during `SLEEP`, no on-chip `Q` 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.

8.8.2 WAKE-UP USING INTERRUPTS

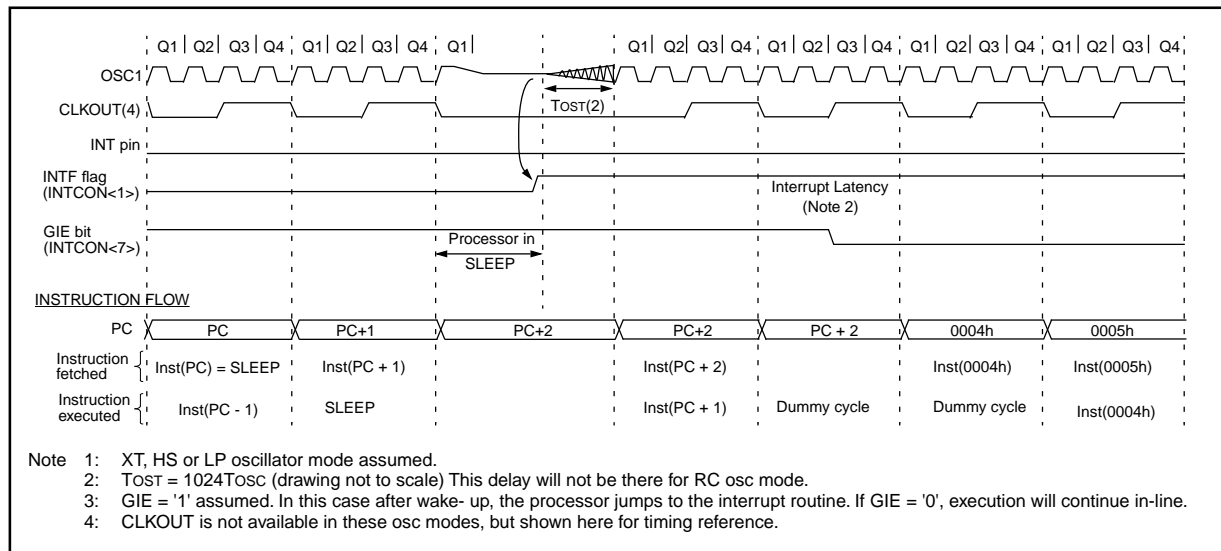
When global interrupts are disabled (`GIE` cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the the execution of a `SLEEP` instruction, the `SLEEP` instruction will complete as a `NOP`. Therefore, the `WDT` and `WDT` postscaler will not be cleared, the \overline{TO} bit will not be set and \overline{PD} bits will not be cleared.
- If the interrupt occurs **during or after** the execution of a `SLEEP` instruction, the device will immediately wake up from sleep. The `SLEEP` instruction will be completely executed before the wake-up. Therefore, the `WDT` and `WDT` postscaler will be cleared, the \overline{TO} bit will be set and the \overline{PD} bit will be cleared.

Even if the flag bits were checked before executing a `SLEEP` instruction, it may be possible for flag bits to become set before the `SLEEP` instruction completes. To determine whether a `SLEEP` instruction executed, test the \overline{PD} bit. If the \overline{PD} bit is set, the `SLEEP` instruction was executed as a `NOP`.

To ensure that the `WDT` is cleared, a `CLRWDT` instruction should be executed before a `SLEEP` instruction.

FIGURE 8-22: WAKE-UP FROM SLEEP THROUGH INTERRUPT



8.9 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.

8.10 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.

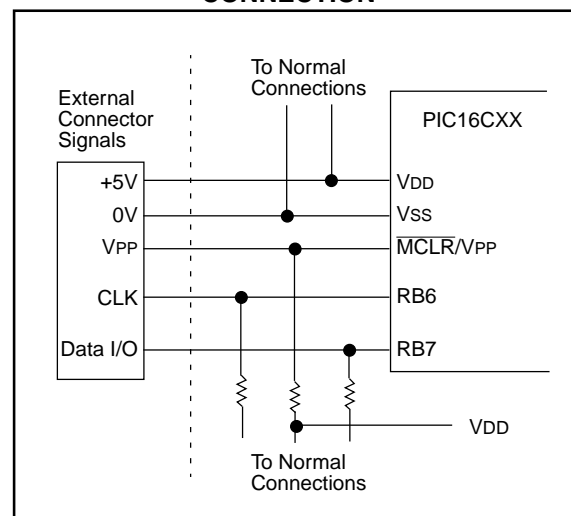
8.11 In-Circuit Serial Programming

PIC16CXX 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.

The device is placed into a program/verify mode by holding the RB6 and RB7 pins low while raising the MCLR (VPP) pin from VIL to VIH (see programming specification). RB6 becomes the programming clock and RB7 becomes the programming data. Both RB6 and RB7 are Schmitt Trigger inputs in this mode.

After reset, to place the device into programming/verify mode, the program counter (PC) is at location 00h. A 6-bit command is then supplied to the device. Depending on the command, 14-bits of program data are then supplied to or from the device, depending if the command was a load or a read. For complete details of serial programming, please refer to the PIC16C6X/7X Programming Specifications (Literature #DS30228).

FIGURE 8-23: TYPICAL IN-CIRCUIT SERIAL PROGRAMMING CONNECTION



PIC16C71X

NOTES:

9.0 INSTRUCTION SET SUMMARY

Each PIC16CXX 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 9-2 lists **byte-oriented**, **bit-oriented**, and **literal and control** operations. Table 9-1 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 9-1: 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 |
| label | Label name |
| TOS | Top of Stack |
| PC | Program Counter |
| PCLATH | Program Counter High Latch |
| GIE | Global Interrupt Enable bit |
| WDT | Watchdog Timer/Counter |
| TO | Time-out bit |
| PD | Power-down bit |
| dest | Destination either the W register or the specified register file location |
| [] | Options |
| () | Contents |
| → | Assigned to |
| < > | Register bit field |
| ∈ | In the set of |
| <i>italics</i> | User defined term (font is courier) |

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 9-2 lists the instructions recognized by the MPASM assembler.

Figure 9-1 shows the general formats that the instructions can have.

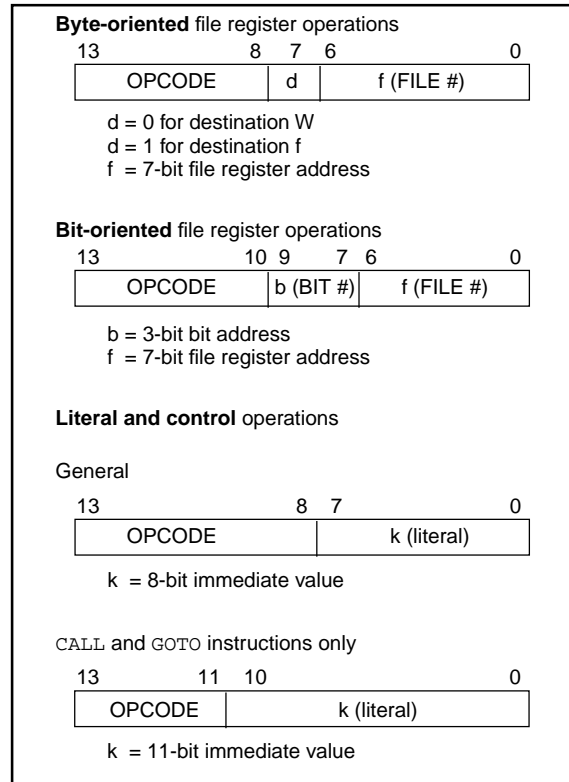
Note: To maintain upward compatibility with future PIC16CXX products, do not use the `OPTION` and `TRIS` instructions.

All examples use the following format to represent a hexadecimal number:

0xhh

where h signifies a hexadecimal digit.

FIGURE 9-1: GENERAL FORMAT FOR INSTRUCTIONS



PIC16C71X

TABLE 9-2: PIC16CXX 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 |
| CLRW | - | Clear W | 1 | 00 | 0001 | 0xxx | xxxx | 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 nibbles in f | 1 | 00 | 1110 | dfff | ffff | | 1,2 |
| XORWF | f, d | Exclusive OR W with f | 1 | 00 | 0110 | dfff | ffff | Z | 1,2 |
| BIT-ORIENTED FILE REGISTER OPERATIONS | | | | | | | | | |
| BCF | f, b | Bit Clear f | 1 | 01 | 00bb | bfff | ffff | | 1,2 |
| BSF | f, b | Bit Set f | 1 | 01 | 01bb | bfff | ffff | | 1,2 |
| BTFSC | f, b | Bit Test f, Skip if Clear | 1 (2) | 01 | 10bb | bfff | ffff | | 3 |
| BTFSS | f, b | Bit Test f, Skip if Set | 1 (2) | 01 | 11bb | bfff | ffff | | 3 |
| LITERAL AND CONTROL OPERATIONS | | | | | | | | | |
| ADDLW | k | Add literal and W | 1 | 11 | 111x | kkkk | kkkk | C,DC,Z | |
| ANDLW | k | AND literal with W | 1 | 11 | 1001 | kkkk | kkkk | Z | |
| CALL | k | Call subroutine | 2 | 10 | 0kkk | kkkk | kkkk | | |
| CLRWDT | - | Clear Watchdog Timer | 1 | 00 | 0000 | 0110 | 0100 | $\overline{\text{TO}}$, $\overline{\text{PD}}$ | |
| 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 | $\overline{\text{TO}}$, $\overline{\text{PD}}$ | |
| 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.

9.1 Instruction Descriptions

ADDLW Add Literal and W

| | | | | |
|-------------------|---|------------------|--------------|------------|
| Syntax: | [<i>label</i>] ADDLW k | | | |
| Operands: | $0 \leq k \leq 255$ | | | |
| Operation: | $(W) + k \rightarrow (W)$ | | | |
| Status Affected: | C, DC, Z | | | |
| Encoding: | 11 | 111x | kkkk | kkkk |
| Description: | The contents of the W register are added to the eight bit literal 'k' and the result is placed in the W register. | | | |
| Words: | 1 | | | |
| Cycles: | 1 | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 |
| | Decode | Read literal 'k' | Process data | Write to W |

Example:

```
ADDLW    0x15
Before Instruction
W = 0x10
After Instruction
W = 0x25
```

ADDWF Add W and f

| | | | | |
|-------------------|--|-------------------|--------------|---------------|
| Syntax: | [<i>label</i>] ADDWF f,d | | | |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ | | | |
| Operation: | $(W) + (f) \rightarrow (\text{dest})$ | | | |
| Status Affected: | C, DC, Z | | | |
| Encoding: | 00 | 0111 | dfff | ffff |
| 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'. | | | |
| Words: | 1 | | | |
| Cycles: | 1 | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 |
| | Decode | Read register 'f' | Process data | Write to Dest |

Example

```
ADDWF    FSR, 0
Before Instruction
W = 0x17
FSR = 0xC2
After Instruction
W = 0xD9
FSR = 0xC2
```

ANDLW AND Literal with W

| | | | | |
|-------------------|---|------------------|--------------|------------|
| Syntax: | [<i>label</i>] ANDLW k | | | |
| Operands: | $0 \leq k \leq 255$ | | | |
| Operation: | $(W) .\text{AND.} (k) \rightarrow (W)$ | | | |
| Status Affected: | Z | | | |
| Encoding: | 11 | 1001 | kkkk | kkkk |
| Description: | The contents of W register are AND'ed with the eight bit literal 'k'. The result is placed in the W register. | | | |
| Words: | 1 | | | |
| Cycles: | 1 | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 |
| | Decode | Read literal "k" | Process data | Write to W |

Example

```
ANDLW    0x5F
Before Instruction
W = 0xA3
After Instruction
W = 0x03
```

ANDWF AND W with f

| | | | | |
|-------------------|--|-------------------|--------------|---------------|
| Syntax: | [<i>label</i>] ANDWF f,d | | | |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ | | | |
| Operation: | $(W) .\text{AND.} (f) \rightarrow (\text{dest})$ | | | |
| Status Affected: | Z | | | |
| Encoding: | 00 | 0101 | dfff | ffff |
| 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'. | | | |
| Words: | 1 | | | |
| Cycles: | 1 | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 |
| | Decode | Read register 'f' | Process data | Write to Dest |

Example

```
ANDWF    FSR, 1
Before Instruction
W = 0x17
FSR = 0xC2
After Instruction
W = 0x17
FSR = 0x02
```

PIC16C71X

| BCF | | Bit Clear f | | | |
|-------------------|-------------------------------------|-------------------|--------------|--------------------|--|
| Syntax: | [label] BCF f,b | | | | |
| Operands: | 0 ≤ f ≤ 127 0 ≤ b ≤ 7 | | | | |
| Operation: | 0 → (f) | | | | |
| Status Affected: | None | | | | |
| Encoding: | 01 | 00bb | bfff | ffff | |
| Description: | Bit 'b' in register 'f' is cleared. | | | | |
| Words: | 1 | | | | |
| Cycles: | 1 | | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 | |
| | Decode | Read register 'f' | Process data | Write register 'f' | |

Example `BCF FLAG_REG, 7`

Before Instruction
 `FLAG_REG = 0xC7`
 After Instruction
 `FLAG_REG = 0x47`

| 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 | | | | | | | | |
| Encoding: | <table><tr><td>01</td><td>01bb</td><td>bfff</td><td>ffff</td></tr></table> | 01 | 01bb | bfff | ffff | | | | |
| 01 | 01bb | bfff | ffff | | | | | | |
| Description: | Bit 'b' in register 'f' is set. | | | | | | | | |
| Words: | 1 | | | | | | | | |
| Cycles: | 1 | | | | | | | | |
| Q Cycle Activity: | <table><tr><th>Q1</th><th>Q2</th><th>Q3</th><th>Q4</th></tr><tr><td>Decode</td><td>Read register 'f'</td><td>Process data</td><td>Write register 'f'</td></tr></table> | Q1 | Q2 | Q3 | Q4 | Decode | Read register 'f' | Process data | Write register 'f' |
| Q1 | Q2 | Q3 | Q4 | | | | | | |
| Decode | Read register 'f' | Process data | Write register 'f' | | | | | | |

Example `BSF FLAG_REG, 7`

Before Instruction
 `FLAG_REG = 0x0A`
 After Instruction
 `FLAG_REG = 0x8A`

| BTFSC | | Bit Test, Skip if Clear | | | |
|------------------|--|-------------------------|------|------|--|
| Syntax: | [label] BTFSC f,b | | | | |
| Operands: | $0 \leq f \leq 127$ $0 \leq b \leq 7$ | | | | |
| Operation: | skip if (f) = 0 | | | | |
| Status Affected: | None | | | | |
| Encoding: | 01 | 10bb | bfff | ffff | |
| Description: | If bit 'b' in register 'f' is '1' then the next instruction is executed. If bit 'b', in register 'f', is '0' then the next instruction is discarded, and a NOP is executed instead, making this a 2TCY instruction. | | | | |
| Words: | 1 | | | | |
| Cycles: | 1(2) | | | | |

If Skip: (2nd Cycle)

| Q1 | Q2 | Q3 | Q4 |
|-----|-----|-----|-----|
| NOP | NOP | NOP | NOP |

Example

```

HERE    BTFSC   FLAG,1
FALSE   GOTO    PROCESS_CODE
TRUE    :
         :
         :

```

Before Instruction
 `PC = address HERE`
 After Instruction
 if `FLAG<1> = 0`,
 `PC = address TRUE`
 if `FLAG<1> >= 1`,
 `PC = address FALSE`

| BTFSS | | | | | Bit Test f, Skip if Set | | | | |
|-------------------|--|--|-------------------|--|-------------------------|--|------|--|--|
| Syntax: | [label] BTFSS f,b | | | | | | | | |
| Operands: | $0 \leq f \leq 127$ | | | | | | | | |
| | $0 \leq b < 7$ | | | | | | | | |
| Operation: | skip if (f<b) = 1 | | | | | | | | |
| Status Affected: | None | | | | | | | | |
| Encoding: | 01 | | 11bb | | bfff | | ffff | | |
| Description: | If bit 'b' in register 'f' is '0' then the next instruction is executed. If bit 'b' is '1', then the next instruction is discarded and a NOP is executed instead, making this a 2TCY instruction. | | | | | | | | |
| Words: | 1 | | | | | | | | |
| Cycles: | 1(2) | | | | | | | | |
| Q Cycle Activity: | Q1 | | Q2 | | Q3 | | Q4 | | |
| | Decode | | Read register 'f' | | Process data | | NOP | | |
| If Skip: | (2nd Cycle) | | | | | | | | |
| | Q1 | | Q2 | | Q3 | | Q4 | | |
| | NOP | | NOP | | NOP | | NOP | | |

Example

```

HERE    BTFSC  FLAG, 1
FALSE   GOTO   PROCESS_CODE
TRUE    •
        •
        •

```

Before Instruction
PC = address HERE

After Instruction
if $FLAG<1> = 0$,
PC = address FALSE
if $FLAG<1> = 1$,
PC = address TRUE

| CALL | | Call Subroutine | | | | | | |
|-------------------|---|------------------------------------|--------------|-------------|----|------|------|------|
| Syntax: | [label] CALL k | | | | | | | |
| Operands: | 0 ≤ k ≤ 2047 | | | | | | | |
| Operation: | (PC)+ 1 → TOS, k → PC<10:0>, (PCLATH<4:3>) → PC<12:11> | | | | | | | |
| Status Affected: | None | | | | | | | |
| Encoding: | <table><tr><td>10</td><td>0kkk</td><td>kkkk</td><td>kkkk</td></tr></table> | | | | 10 | 0kkk | kkkk | kkkk |
| 10 | 0kkk | kkkk | kkkk | | | | | |
| Description: | Call Subroutine. First, return address (PC+1) is pushed onto the stack. The eleven bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a two cycle instruction. | | | | | | | |
| Words: | 1 | | | | | | | |
| Cycles: | 2 | | | | | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 | | | | |
| 1st Cycle | Decode | Read literal 'k', Push PC to Stack | Process data | Write to PC | | | | |
| 2nd Cycle | NOP | NOP | NOP | NOP | | | | |

Example

```

HERE    CALL   THERE

```

Before Instruction
PC = Address HERE

After Instruction
PC = Address THERE
TOS = Address HERE+1

PIC16C71X

| CLRF | Clear f | | | | | | | | |
|-------------------|--|--------------|--------------------|------|------|--------|-------------------|--------------|--------------------|
| Syntax: | [label] CLRF f | | | | | | | | |
| Operands: | $0 \leq f \leq 127$ | | | | | | | | |
| Operation: | 00h \rightarrow (f) 1 \rightarrow Z | | | | | | | | |
| Status Affected: | Z | | | | | | | | |
| Encoding: | <table><tr><td>00</td><td>0001</td><td>1fff</td><td>ffff</td></tr></table> | 00 | 0001 | 1fff | ffff | | | | |
| 00 | 0001 | 1fff | ffff | | | | | | |
| Description: | The contents of register 'f' are cleared and the Z bit is set. | | | | | | | | |
| Words: | 1 | | | | | | | | |
| Cycles: | 1 | | | | | | | | |
| Q Cycle Activity: | <table><tr><th>Q1</th><th>Q2</th><th>Q3</th><th>Q4</th></tr><tr><td>Decode</td><td>Read register 'f'</td><td>Process data</td><td>Write register 'f'</td></tr></table> | Q1 | Q2 | Q3 | Q4 | Decode | Read register 'f' | Process data | Write register 'f' |
| Q1 | Q2 | Q3 | Q4 | | | | | | |
| Decode | Read register 'f' | Process data | Write register 'f' | | | | | | |

Example

```

CLRF    FLAG_REG
Before Instruction
    FLAG_REG = 0x5A
After Instruction
    FLAG_REG = 0x00
    Z        = 1
  
```

| CLRW | Clear W | | | | | | | | |
|-------------------|--|--------------|------------|------|------|--------|-----|--------------|------------|
| Syntax: | [<i>label</i>] CLRW | | | | | | | | |
| Operands: | None | | | | | | | | |
| Operation: | 00h → (W) 1 → Z | | | | | | | | |
| Status Affected: | Z | | | | | | | | |
| Encoding: | <table><tr><td>00</td><td>0001</td><td>0xxx</td><td>xxxx</td></tr></table> | 00 | 0001 | 0xxx | xxxx | | | | |
| 00 | 0001 | 0xxx | xxxx | | | | | | |
| Description: | W register is cleared. Zero bit (Z) is set. | | | | | | | | |
| Words: | 1 | | | | | | | | |
| Cycles: | 1 | | | | | | | | |
| Q Cycle Activity: | <table><tr><td>Q1</td><td>Q2</td><td>Q3</td><td>Q4</td></tr><tr><td>Decode</td><td>NOP</td><td>Process data</td><td>Write to W</td></tr></table> | Q1 | Q2 | Q3 | Q4 | Decode | NOP | Process data | Write to W |
| Q1 | Q2 | Q3 | Q4 | | | | | | |
| Decode | NOP | Process data | Write to W | | | | | | |

Example

```

CLRWF
Before Instruction
    W = 0x5A
After Instruction
    W = 0x00
    Z = 1
  
```

| CLRWDTC | | Clear Watchdog Timer | | | | | | |
|-------------------|--|----------------------|--------------|-------------------|----|------|------|------|
| Syntax: | [<i>label</i>] CLRWDTC | | | | | | | |
| Operands: | None | | | | | | | |
| Operation: | 00h → WDT 0 → WDT prescaler, 1 → \overline{TO} 1 → \overline{PD} | | | | | | | |
| Status Affected: | \overline{TO} , \overline{PD} | | | | | | | |
| Encoding: | <table><tr><td>00</td><td>0000</td><td>0110</td><td>0100</td></tr></table> | | | | 00 | 0000 | 0110 | 0100 |
| 00 | 0000 | 0110 | 0100 | | | | | |
| Description: | CLRWDTC instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits \overline{TO} and \overline{PD} are set. | | | | | | | |
| Words: | 1 | | | | | | | |
| Cycles: | 1 | | | | | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 | | | | |
| | Decode | NOP | Process data | Clear WDT Counter | | | | |

Example

```

CLRWDTC
Before Instruction
    WDT counter = ?
After Instruction
    WDT counter = 0x00
    WDT prescaler = 0
     $\overline{TO}$  = 1
     $\overline{PD}$  = 1
  
```

| COMF | | Complement f | | | |
|-------------------|--|-------------------|--------------|---------------|--|
| Syntax: | [label] COMF f,d | | | | |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ | | | | |
| Operation: | $(\bar{f}) \rightarrow (\text{dest})$ | | | | |
| Status Affected: | Z | | | | |
| Encoding: | 00 | 1001 | dfff | ffff | |
| 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'. | | | | |
| Words: | 1 | | | | |
| Cycles: | 1 | | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 | |
| | Decode | Read register 'f' | Process data | Write to dest | |

Example

```
COMF    REG1, 0
```

Before Instruction
REG1 = 0x13

After Instruction
REG1 = 0x13
W = 0xEC

| DECf | Decrement f | | | | | | | | |
|-------------------|---|--------------|---------------|------|------|--------|-------------------|--------------|---------------|
| Syntax: | [label] DECf f,d | | | | | | | | |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ | | | | | | | | |
| Operation: | $(f) - 1 \rightarrow (\text{dest})$ | | | | | | | | |
| Status Affected: | Z | | | | | | | | |
| Encoding: | <table><tr><td>00</td><td>0011</td><td>dfff</td><td>ffff</td></tr></table> | 00 | 0011 | dfff | ffff | | | | |
| 00 | 0011 | dfff | ffff | | | | | | |
| 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'. | | | | | | | | |
| Words: | 1 | | | | | | | | |
| Cycles: | 1 | | | | | | | | |
| Q Cycle Activity: | <table><tr><td>Q1</td><td>Q2</td><td>Q3</td><td>Q4</td></tr><tr><td>Decode</td><td>Read register 'f'</td><td>Process data</td><td>Write to dest</td></tr></table> | Q1 | Q2 | Q3 | Q4 | Decode | Read register 'f' | Process data | Write to dest |
| Q1 | Q2 | Q3 | Q4 | | | | | | |
| Decode | Read register 'f' | Process data | Write to dest | | | | | | |

Example

```
DECf    CNT, 1
```

Before Instruction
CNT = 0x01
Z = 0

After Instruction
CNT = 0x00
Z = 1

| DECFSZ | | Decrement f, Skip if 0 | | | | | | | | | | |
|-------------------|--|------------------------|---------------|--|----|------|------|------|--------|-------------------|--------------|---------------|
| Syntax: | [label] DECFSZ f,d | | | | | | | | | | | |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ | | | | | | | | | | | |
| Operation: | $(f) - 1 \rightarrow (\text{dest}); \quad \text{skip if result} = 0$ | | | | | | | | | | | |
| Status Affected: | None | | | | | | | | | | | |
| Encoding: | <table border="1"><tr><td>00</td><td>1011</td><td>dfff</td><td>ffff</td></tr></table> | | | | 00 | 1011 | dfff | ffff | | | | |
| 00 | 1011 | dfff | ffff | | | | | | | | | |
| Description: | <p>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'.</p> <p>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.</p> | | | | | | | | | | | |
| Words: | 1 | | | | | | | | | | | |
| Cycles: | 1(2) | | | | | | | | | | | |
| Q Cycle Activity: | <table border="1"><tr><th>Q1</th><th>Q2</th><th>Q3</th><th>Q4</th></tr><tr><td>Decode</td><td>Read register 'f'</td><td>Process data</td><td>Write to dest</td></tr></table> | | | | Q1 | Q2 | Q3 | Q4 | Decode | Read register 'f' | Process data | Write to dest |
| Q1 | Q2 | Q3 | Q4 | | | | | | | | | |
| Decode | Read register 'f' | Process data | Write to dest | | | | | | | | | |
| If Skip: | (2nd Cycle) | | | | | | | | | | | |
| | <table border="1"><tr><th>Q1</th><th>Q2</th><th>Q3</th><th>Q4</th></tr><tr><td>NOP</td><td>NOP</td><td>NOP</td><td>NOP</td></tr></table> | | | | Q1 | Q2 | Q3 | Q4 | NOP | NOP | NOP | NOP |
| Q1 | Q2 | Q3 | Q4 | | | | | | | | | |
| NOP | NOP | NOP | NOP | | | | | | | | | |

Example

```
HERE    DECFSZ    CNT, 1
        GOTO      LOOP
CONTINUE •
        •
        •
```

Before Instruction
PC = address HERE

After Instruction
CNT = CNT - 1
if CNT = 0,
PC = address CONTINUE
if CNT \neq 0,
PC = address HERE+1

PIC16C71X

GOTO

Unconditional Branch

Syntax: [*label*] GOTO k

Operands: $0 \leq k \leq 2047$

Operation: $k \rightarrow PC<10:0>$
 $PCLATH<4:3> \rightarrow PC<12:11>$

Status Affected: None

Encoding:

| | | | |
|----|------|------|------|
| 10 | 1kkk | kkkk | kkkk |
|----|------|------|------|

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.

Words: 1

Cycles: 2

Q Cycle Activity:

| | Q1 | Q2 | Q3 | Q4 |
|-----------|--------|------------------|--------------|-------------|
| 1st Cycle | Decode | Read literal 'k' | Process data | Write to PC |
| 2nd Cycle | NOP | NOP | NOP | NOP |

Example

GOTO THERE

After Instruction

PC = Address THERE

INCF

Increment f

Syntax: [*label*] INCF f,d

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

Operation: $(f) + 1 \rightarrow (dest)$

Status Affected: Z

Encoding:

| | | | |
|----|------|------|------|
| 00 | 1010 | dfff | ffff |
|----|------|------|------|

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'.

Words: 1

Cycles: 1

Q Cycle Activity:

| | Q1 | Q2 | Q3 | Q4 |
|--|--------|-------------------|--------------|---------------|
| | Decode | Read register 'f' | Process data | Write to dest |

Example

INCF CNT, 1

Before Instruction

CNT = 0xFF

Z = 0

After Instruction

CNT = 0x00

Z = 1

INCFSZ Increment f, Skip if 0

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

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

Operation: $(f) + 1 \rightarrow (\text{dest})$, skip if result = 0

Status Affected: None

Encoding:

| | | | |
|----|------|------|------|
| 00 | 1111 | dfff | ffff |
|----|------|------|------|

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.

Words: 1

Cycles: 1(2)

Q Cycle Activity:

| Q1 | Q2 | Q3 | Q4 |
|--------|-------------------|--------------|---------------|
| Decode | Read register 'f' | Process data | Write to dest |

If Skip: (2nd Cycle)

| Q1 | Q2 | Q3 | Q4 |
|-----|-----|-----|-----|
| NOP | NOP | NOP | NOP |

Example

```

HERE      INCFSZ    CNT, 1
          GOTO      LOOP
CONTINUE  •
          •
          •
    
```

Before Instruction

PC = address HERE

After Instruction

```

CNT = CNT + 1
if CNT= 0,
PC = address CONTINUE
if CNT≠ 0,
PC = address HERE +1
    
```

IORLW Inclusive OR Literal with W

Syntax: [*label*] IORLW *k*

Operands: $0 \leq k \leq 255$

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

Status Affected: Z

Encoding:

| | | | |
|----|------|------|------|
| 11 | 1000 | kkkk | kkkk |
|----|------|------|------|

Description: The contents of the W register is OR'ed with the eight bit literal 'k'. The result is placed in the W register.

Words: 1

Cycles: 1

Q Cycle Activity:

| Q1 | Q2 | Q3 | Q4 |
|--------|------------------|--------------|------------|
| Decode | Read literal 'k' | Process data | Write to W |

Example

IORLW 0x35

Before Instruction

W = 0x9A

After Instruction

W = 0xBF

Z = 1

PIC16C71X

IORWF Inclusive OR W with f

| | | | | |
|-------------------|---|-------------------|--------------|---------------|
| Syntax: | [<i>label</i>] IORWF f,d | | | |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ | | | |
| Operation: | (W) .OR. (f) → (dest) | | | |
| Status Affected: | \bar{Z} | | | |
| Encoding: | 00 | 0100 | dfff | ffff |
| 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'. | | | |
| Words: | 1 | | | |
| Cycles: | 1 | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 |
| | Decode | Read register 'f' | Process data | Write to dest |

Example IORWF RESULT, 0

Before Instruction

RESULT = 0x13
W = 0x91

After Instruction

RESULT = 0x13
W = 0x93
Z = 1

MOVF Move f

| | | | | |
|-------------------|---|-------------------|--------------|---------------|
| Syntax: | [<i>label</i>] MOVF f,d | | | |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ | | | |
| Operation: | (f) → (dest) | | | |
| Status Affected: | Z | | | |
| Encoding: | 00 | 1000 | dfff | ffff |
| Description: | The contents of register f is 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. | | | |
| Words: | 1 | | | |
| Cycles: | 1 | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 |
| | Decode | Read register 'f' | Process data | Write to dest |

Example MOVF FSR, 0

After Instruction

W = value in FSR register
Z = 1

MOVLW Move Literal to W

| | | | | |
|-------------------|--|------------------|--------------|------------|
| Syntax: | [<i>label</i>] MOVLW k | | | |
| Operands: | $0 \leq k \leq 255$ | | | |
| Operation: | $k \rightarrow (W)$ | | | |
| Status Affected: | None | | | |
| Encoding: | 11 | 00xx | kkkk | kkkk |
| Description: | The eight bit literal 'k' is loaded into W register. The don't cares will assemble as 0's. | | | |
| Words: | 1 | | | |
| Cycles: | 1 | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 |
| | Decode | Read literal 'k' | Process data | Write to W |

Example

MOVLW 0x5A

After Instruction

W = 0x5A

MOVWF Move W to f

| | | | | |
|-------------------|--|-------------------|--------------|--------------------|
| Syntax: | [<i>label</i>] MOVWF f | | | |
| Operands: | $0 \leq f \leq 127$ | | | |
| Operation: | (W) → (f) | | | |
| Status Affected: | None | | | |
| Encoding: | 00 | 0000 | 1fff | ffff |
| Description: | Move data from W register to register 'f'. | | | |
| Words: | 1 | | | |
| Cycles: | 1 | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 |
| | Decode | Read register 'f' | Process data | Write register 'f' |

Example

MOVWF OPTION_REG

Before Instruction

OPTION = 0xFF
W = 0x4F

After Instruction

OPTION = 0x4F
W = 0x4F

| NOP | | No Operation | | | |
|-------------------|----------------------|--------------|------|------|--|
| Syntax: | [<i>label</i>] NOP | | | | |
| Operands: | None | | | | |
| Operation: | No operation | | | | |
| Status Affected: | None | | | | |
| Encoding: | 00 | 0000 | 0xx0 | 0000 | |
| Description: | No operation. | | | | |
| Words: | 1 | | | | |
| Cycles: | 1 | | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 | |
| | Decode | NOP | NOP | NOP | |
| Example | NOP | | | | |

| RETFIE | | Return from Interrupt | | | |
|-------------------|--|-----------------------|-----------------|--------------------|--|
| Syntax: | [<i>label</i>] RETFIE | | | | |
| Operands: | None | | | | |
| Operation: | TOS → PC, 1 → GIE | | | | |
| Status Affected: | None | | | | |
| Encoding: | 00 | 0000 | 0000 | 1001 | |
| Description: | Return from Interrupt. Stack is POPed and Top of Stack (TOS) is loaded in the PC. Interrupts are enabled by setting Global Interrupt Enable bit, GIE (INTCON<7>). This is a two cycle instruction. | | | | |
| Words: | 1 | | | | |
| Cycles: | 2 | | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 | |
| 1st Cycle | Decode | NOP | Set the GIE bit | Pop from the Stack | |
| 2nd Cycle | NOP | NOP | NOP | NOP | |

| OPTION | Load Option Register | | | | |
|------------------|---|------|------|------|------|
| Syntax: | [<i>label</i>] OPTION | | | | |
| Operands: | None | | | | |
| Operation: | (W) → OPTION | | | | |
| Status Affected: | None | | | | |
| Encoding: | <table><tr><td>00</td><td>0000</td><td>0110</td><td>0010</td></tr></table> | 00 | 0000 | 0110 | 0010 |
| 00 | 0000 | 0110 | 0010 | | |
| Description: | <p>The contents of the W register are loaded in the OPTION register. This instruction is supported for code compatibility with PIC16C5X products. Since OPTION is a readable/writable register, the user can directly address it.</p> | | | | |
| Words: | 1 | | | | |
| Cycles: | 1 | | | | |
| Example | <div>To maintain upward compatibility with future PIC16CXX products, do not use this instruction.</div> | | | | |

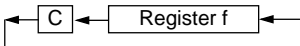
PIC16C71X

| RETLW | Return with Literal in W | | | |
|-------------------|---|------------------|------|--------------------------------|
| Syntax: | [<i>label</i>] RETLW k | | | |
| Operands: | 0 ≤ k ≤ 255 | | | |
| Operation: | k → (W); TOS → PC | | | |
| Status Affected: | None | | | |
| Encoding: | 11 | 01xx | kkkk | kkkk |
| 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. | | | |
| Words: | 1 | | | |
| Cycles: | 2 | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 |
| 1st Cycle | Decode | Read literal 'k' | NOP | Write to W, Pop from the Stack |
| 2nd Cycle | NOP | NOP | NOP | NOP |

| | | | | |
|--------------------|------------|------------------------|--------------|--|
| Example | CALL TABLE | ;W contains table | | |
| | | ;offset value | | |
| | • | ;W now has table value | | |
| | • | | | |
| | • | | | |
| | TABLE | ADDWF PC | ;W = offset | |
| | | RETLW k1 | ;Begin table | |
| | | RETLW k2 | ; | |
| | • | | | |
| | • | | | |
| | • | | | |
| | RETLW kn | ; End of table | | |
| Before Instruction | | | | |
| | W | = | 0x07 | |
| After Instruction | | | | |
| | W | = | value of k8 | |

| RETURN | Return from Subroutine | | | |
|-------------------|--|------|------|--------------------|
| Syntax: | [<i>label</i>] RETURN | | | |
| Operands: | None | | | |
| Operation: | TOS → PC | | | |
| Status Affected: | None | | | |
| Encoding: | 00 | 0000 | 0000 | 1000 |
| 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. | | | |
| Words: | 1 | | | |
| Cycles: | 2 | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 |
| 1st Cycle | Decode | NOP | NOP | Pop from the Stack |
| 2nd Cycle | NOP | NOP | NOP | NOP |

| | |
|---------|-----------------|
| Example | RETURN |
| | After Interrupt |
| | PC = TOS |

| RLF | | Rotate Left f through Carry | | | | | | | |
|-------------------|--|-----------------------------|--------------|---------------|--|----|------|------|------|
| Syntax: | [<i>label</i>] RLF f,d | | | | | | | | |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ | | | | | | | | |
| Operation: | See description below | | | | | | | | |
| Status Affected: | C | | | | | | | | |
| Encoding: | <table><tr><td>00</td><td>1101</td><td>dfff</td><td>ffff</td></tr></table> | | | | | 00 | 1101 | dfff | ffff |
| 00 | 1101 | dfff | ffff | | | | | | |
| Description: | <p>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'.</p>  | | | | | | | | |
| Words: | 1 | | | | | | | | |
| Cycles: | 1 | | | | | | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 | | | | | |
| | Decode | Read register 'f' | Process data | Write to dest | | | | | |

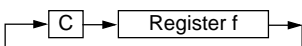
Example

```

RLF      REG1,0

Before Instruction
    REG1  = 1110 0110
    C     = 0

After Instruction
    REG1  = 1110 0110
    W     = 1100 1100
    C     = 1
  
```

| RRF | | Rotate Right f through Carry | | | | | | |
|-------------------|---|------------------------------|--------------|---------------|----|------|------|------|
| Syntax: | [<i>label</i>] RRF f,d | | | | | | | |
| Operands: | $0 \leq f \leq 127$ $d \in [0,1]$ | | | | | | | |
| Operation: | See description below | | | | | | | |
| Status Affected: | C | | | | | | | |
| Encoding: | <table><tr><td>00</td><td>1100</td><td>dfff</td><td>ffff</td></tr></table> | | | | 00 | 1100 | dfff | ffff |
| 00 | 1100 | dfff | ffff | | | | | |
| Description: | <p>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'.</p>  | | | | | | | |
| Words: | 1 | | | | | | | |
| Cycles: | 1 | | | | | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 | | | | |
| | Decode | Read register 'f' | Process data | Write to dest | | | | |

Example

```

RRF      REG1,0

Before Instruction
    REG1  = 1110 0110
    C     = 0

After Instruction
    REG1  = 1110 0110
    W     = 0111 0011
    C     = 0
  
```

PIC16C71X

SLEEP

Syntax: [*label*] SLEEP

Operands: None

Operation: 00h → WDT,
0 → WDT prescaler,
1 → \overline{TO} ,
0 → \overline{PD}

Status Affected: \overline{TO} , \overline{PD}

Encoding:

| | | | |
|----|------|------|------|
| 00 | 0000 | 0110 | 0011 |
|----|------|------|------|

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 8.8 for more details.

Words: 1

Cycles: 1

Q Cycle Activity:

| Q1 | Q2 | Q3 | Q4 |
|--------|-----|-----|-------------|
| Decode | NOP | NOP | Go to Sleep |

Example: SLEEP

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

Encoding:

| | | | |
|----|------|------|------|
| 11 | 110x | kkkk | kkkk |
|----|------|------|------|

Description: The W register is subtracted (2's complement method) from the eight bit literal 'k'. The result is placed in the W register.

Words: 1

Cycles: 1

Q Cycle Activity:

| Q1 | Q2 | Q3 | Q4 |
|--------|------------------|--------------|------------|
| Decode | Read literal 'k' | Process data | Write to W |

Example 1: SUBLW 0x02

Before Instruction

W = 1
C = ?
Z = ?

After Instruction

W = 1
C = 1; result is positive
Z = 0

Example 2: Before Instruction

W = 2
C = ?
Z = ?

After Instruction

W = 0
C = 1; result is zero
Z = 1

Example 3: Before Instruction

W = 3
C = ?
Z = ?

After Instruction

W = 0xFF
C = 0; result is negative
Z = 0

SUBWF Subtract W from f

Syntax: `[label] SUBWF f,d`

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

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

Status Affected: C, DC, Z

Encoding:

| | | | |
|----|------|------|------|
| 00 | 0010 | dfff | ffff |
|----|------|------|------|

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'.

Words: 1

Cycles: 1

Q Cycle Activity:

| Q1 | Q2 | Q3 | Q4 |
|--------|-------------------|--------------|---------------|
| Decode | Read register 'f' | Process data | Write to dest |

Example 1: `SUBWF REG1, 1`

Before Instruction

```
REG1 = 3
W    = 2
C    = ?
Z    = ?
```

After Instruction

```
REG1 = 1
W    = 2
C    = 1; result is positive
Z    = 0
```

Example 2: Before Instruction

```
REG1 = 2
W    = 2
C    = ?
Z    = ?
```

After Instruction

```
REG1 = 0
W    = 2
C    = 1; result is zero
Z    = 1
```

Example 3: Before Instruction

```
REG1 = 1
W    = 2
C    = ?
Z    = ?
```

After Instruction

```
REG1 = 0xFF
W    = 2
C    = 0; result is negative
Z    = 0
```

SWAPF Swap Nibbles in f

Syntax: `[label] SWAPF f,d`

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

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

Status Affected: None

Encoding:

| | | | |
|----|------|------|------|
| 00 | 1110 | dfff | ffff |
|----|------|------|------|

Description: The upper and lower nibbles of register 'f' are exchanged. If 'd' is 0 the result is placed in W register. If 'd' is 1 the result is placed in register 'f'.

Words: 1

Cycles: 1

Q Cycle Activity:

| Q1 | Q2 | Q3 | Q4 |
|--------|-------------------|--------------|---------------|
| Decode | Read register 'f' | Process data | Write to dest |

Example `SWAPF REG, 0`

Before Instruction

```
REG1 = 0xA5
```

After Instruction

```
REG1 = 0xA5
W    = 0x5A
```

TRIS Load TRIS Register

Syntax: `[label] TRIS f`

Operands: $5 \leq f \leq 7$

Operation: $(W) \rightarrow \text{TRIS register } f$;

Status Affected: None

Encoding:

| | | | |
|----|------|------|------|
| 00 | 0000 | 0110 | 0fff |
|----|------|------|------|

Description: The instruction is supported for code compatibility with the PIC16C5X products. Since TRIS registers are readable and writable, the user can directly address them.

Words: 1

Cycles: 1

Example

To maintain upward compatibility with future PIC16CXX products, do not use this instruction.

PIC16C71X

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 | | | |
| Encoding: | 11 | 1010 | kkkk | kkkk |
| Description: | The contents of the W register are XOR'ed with the eight bit literal 'k'. The result is placed in the W register. | | | |
| Words: | 1 | | | |
| Cycles: | 1 | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 |
| | Decode | Read literal 'k' | Process data | Write to W |

Example: XORLW 0xAF

Before Instruction

W = 0xB5

After Instruction

W = 0x1A

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 (dest) | | | |
| Status Affected: | Z | | | |
| Encoding: | 00 | 0110 | dfff | ffff |
| 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'. | | | |
| Words: | 1 | | | |
| Cycles: | 1 | | | |
| Q Cycle Activity: | Q1 | Q2 | Q3 | Q4 |

Example XORWF REG 1

Before Instruction

REG = 0xAF

W = 0xB5

After Instruction

REG = 0x1A

W = 0xB5

10.0 DEVELOPMENT SUPPORT

10.1 Development Tools

The PICmicro™ microcontrollers are supported with a full range of hardware and software development tools:

- PICMASTER/PICMASTER CE Real-Time In-Circuit Emulator
- ICEPIC Low-Cost PIC16C5X and PIC16CXXX In-Circuit Emulator
- PRO MATE® II Universal Programmer
- PICSTART® Plus Entry-Level Prototype Programmer
- PICDEM-1 Low-Cost Demonstration Board
- PICDEM-2 Low-Cost Demonstration Board
- PICDEM-3 Low-Cost Demonstration Board
- MPASM Assembler
- MPLAB™ SIM Software Simulator
- MPLAB-C (C Compiler)
- Fuzzy Logic Development System (fuzzyTECH®-MP)

10.2 PICMASTER: High Performance Universal In-Circuit Emulator with MPLAB IDE

The PICMASTER Universal In-Circuit Emulator is intended to provide the product development engineer with a complete microcontroller design tool set for all microcontrollers in the PIC12CXXX, PIC14C000, PIC16C5X, PIC16CXXX and PIC17CXX families. PICMASTER is supplied with the MPLAB™ Integrated Development Environment (IDE), which allows editing, "make" and download, and source debugging from a single environment.

Interchangeable target probes allow the system to be easily reconfigured for emulation of different processors. The universal architecture of the PICMASTER allows expansion to support all new Microchip microcontrollers.

The PICMASTER 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 compatible 386 (and higher) machine platform and Microsoft Windows® 3.x environment were chosen to best make these features available to you, the end user.

A CE compliant version of PICMASTER is available for European Union (EU) countries.

10.3 ICEPIC: Low-Cost PIC16CXXX In-Circuit Emulator

ICEPIC is a low-cost in-circuit emulator solution for the Microchip PIC16C5X and PIC16CXXX families of 8-bit OTP microcontrollers.

ICEPIC is designed to operate on PC-compatible machines ranging from 286-AT® through Pentium™ based machines under Windows 3.x environment. ICEPIC features real time, non-intrusive emulation.

10.4 PRO MATE II: Universal Programmer

The PRO MATE II Universal Programmer is a full-featured programmer capable of operating in stand-alone mode as well as PC-hosted mode.

The PRO MATE II has programmable VDD and VPP supplies which allows it to verify programmed memory at VDD min and VDD max for maximum reliability. It has an LCD display for displaying error messages, keys to enter commands and a modular detachable socket assembly to support various package types. In stand-alone mode the PRO MATE II can read, verify or program PIC12CXXX, PIC14C000, PIC16C5X, PIC16CXXX and PIC17CXX devices. It can also set configuration and code-protect bits in this mode.

10.5 PICSTART Plus Entry Level Development System

The PICSTART programmer is an easy-to-use, low-cost prototype programmer. It connects to the PC via one of the COM (RS-232) ports. MPLAB Integrated Development Environment software makes using the programmer simple and efficient. PICSTART Plus is not recommended for production programming.

PICSTART Plus supports all PIC12CXXX, PIC14C000, PIC16C5X, PIC16CXXX and PIC17CXX devices with up to 40 pins. Larger pin count devices such as the PIC16C923 and PIC16C924 may be supported with an adapter socket.

PIC16C71X

10.6 PICDEM-1 Low-Cost PIC16/17 Demonstration Board

The PICDEM-1 is a simple board which demonstrates the capabilities of several of Microchip's microcontrollers. The microcontrollers supported are: PIC16C5X (PIC16C54 to PIC16C58A), PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 and PIC17C44. All necessary hardware and software is included to run basic demo programs. The users can program the sample microcontrollers provided with the PICDEM-1 board, on a PRO MATE II or PICSTART-Plus programmer, and easily test firmware. The user can also connect the PICDEM-1 board to the PICMASTER emulator and download the firmware to the emulator for testing. Additional prototype area is available for the user to build some additional hardware and connect it to the microcontroller socket(s). Some of the features include an RS-232 interface, a potentiometer for simulated analog input, push-button switches and eight LEDs connected to PORTB.

10.7 PICDEM-2 Low-Cost PIC16CXX Demonstration Board

The PICDEM-2 is a simple demonstration board that supports the PIC16C62, PIC16C64, PIC16C65, PIC16C73 and PIC16C74 microcontrollers. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM-2 board, on a PRO MATE II programmer or PICSTART-Plus, and easily test firmware. The PICMASTER emulator may also be used with the PICDEM-2 board to test firmware. Additional prototype area has been provided to the user for adding additional hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push-button switches, a potentiometer for simulated analog input, a Serial EEPROM to demonstrate usage of the I²C bus and separate headers for connection to an LCD module and a keypad.

10.8 PICDEM-3 Low-Cost PIC16CXXX Demonstration Board

The PICDEM-3 is a simple demonstration board that supports the PIC16C923 and PIC16C924 in the PLCC package. It will also support future 44-pin PLCC microcontrollers with a LCD Module. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM-3 board, on a PRO MATE II programmer or PICSTART Plus with an adapter socket, and easily test firmware. The PICMASTER emulator may also be used with the PICDEM-3 board to test firmware. Additional prototype area has been provided to the user for adding hardware and connecting it to the microcontroller socket(s). Some of the features include

an RS-232 interface, push-button switches, a potentiometer for simulated analog input, a thermistor and separate headers for connection to an external LCD module and a keypad. Also provided on the PICDEM-3 board is an LCD panel, with 4 commons and 12 segments, that is capable of displaying time, temperature and day of the week. The PICDEM-3 provides an additional RS-232 interface and Windows 3.1 software for showing the demultiplexed LCD signals on a PC. A simple serial interface allows the user to construct a hardware demultiplexer for the LCD signals.

10.9 MPLAB Integrated Development Environment Software

The MPLAB IDE Software brings an ease of software development previously unseen in the 8-bit microcontroller market. MPLAB is a windows based application which contains:

- A full featured editor
- Three operating modes
 - editor
 - emulator
 - simulator
- A project manager
- Customizable tool bar and key mapping
- A status bar with project information
- Extensive on-line help

MPLAB allows you to:

- Edit your source files (either assembly or 'C')
- One touch assemble (or compile) and download to PIC16/17 tools (automatically updates all project information)
- Debug using:
 - source files
 - absolute listing file
- Transfer data dynamically via DDE (soon to be replaced by OLE)
- Run up to four emulators on the same PC

The ability to use MPLAB with Microchip's simulator allows a consistent platform and the ability to easily switch from the low cost simulator to the full featured emulator with minimal retraining due to development tools.

10.10 Assembler (MPASM)

The MPASM Universal Macro Assembler is a PC-hosted symbolic assembler. It supports all microcontroller series including the PIC12C5XX, PIC14000, PIC16C5X, PIC16CXXX, and PIC17CXX families.

MPASM offers full featured Macro capabilities, conditional assembly, and several source and listing formats. It generates various object code formats to support Microchip's development tools as well as third party programmers.

MPASM allows full symbolic debugging from PICMASTER, Microchip's Universal Emulator System.

MPASM has the following features to assist in developing software for specific use applications.

- Provides translation of Assembler source code to object code for all Microchip microcontrollers.
- Macro assembly capability.
- Produces all the files (Object, Listing, Symbol, and special) required for symbolic debug with Microchip's emulator systems.
- Supports Hex (default), Decimal and Octal source and listing formats.

MPASM provides a rich directive language to support programming of the PIC16/17. Directives are helpful in making the development of your assemble source code shorter and more maintainable.

10.11 Software Simulator (MPLAB-SIM)

The MPLAB-SIM Software Simulator allows code development in a PC host environment. It allows the user to simulate the PIC16/17 series microcontrollers on an instruction level. On any given instruction, the user may examine or modify any of the data areas or provide external stimulus to any of the pins. The input/output radix can be set by the user and the execution can be performed in; single step, execute until break, or in a trace mode.

MPLAB-SIM fully supports symbolic debugging using MPLAB-C and MPASM. The Software Simulator offers the low cost flexibility to develop and debug code outside of the laboratory environment making it an excellent multi-project software development tool.

10.12 C Compiler (MPLAB-C)

The MPLAB-C Code Development System is a complete 'C' compiler and integrated development environment for Microchip's PIC16/17 family of microcontrollers. The compiler provides powerful integration capabilities and ease of use not found with other compilers.

For easier source level debugging, the compiler provides symbol information that is compatible with the MPLAB IDE memory display.

10.13 Fuzzy Logic Development System (fuzzyTECH-MP)

fuzzyTECH-MP fuzzy logic development tool is available in two versions - a low cost introductory version, MP Explorer, for designers to gain a comprehensive working knowledge of fuzzy logic system design; and a full-featured version, *fuzzyTECH-MP*, edition for implementing more complex systems.

Both versions include Microchip's *fuzzyLAB™* demonstration board for hands-on experience with fuzzy logic systems implementation.

10.14 MP-DriveWay™ – Application Code Generator

MP-DriveWay is an easy-to-use Windows-based Application Code Generator. With MP-DriveWay you can visually configure all the peripherals in a PIC16/17 device and, with a click of the mouse, generate all the initialization and many functional code modules in C language. The output is fully compatible with Microchip's MPLAB-C C compiler. The code produced is highly modular and allows easy integration of your own code. MP-DriveWay is intelligent enough to maintain your code through subsequent code generation.

10.15 SEEVAL® Evaluation and Programming System

The SEEVAL SEEPROM Designer's Kit supports all Microchip 2-wire and 3-wire Serial EEPROMs. The kit includes everything necessary to read, write, erase or program special features of any Microchip SEEPROM product including Smart Serials™ and secure serials. The Total Endurance™ Disk is included to aid in trade-off analysis and reliability calculations. The total kit can significantly reduce time-to-market and result in an optimized system.

10.16 KEELoq® Evaluation and Programming Tools

KEELOQ evaluation and programming tools support Microchips HCS Secure Data Products. The HCS evaluation kit includes an LCD display to show changing codes, a decoder to decode transmissions, and a programming interface to program test transmitters.

PIC16C71X

TABLE 10-1: DEVELOPMENT TOOLS FROM MICROCHIP

| | PIC12C5XX | PIC14000 | PIC16C5X | PIC16CXXX | PIC16C6X | PIC16C7XX | PIC16C8X | PIC16C9XX | PIC17C4X | PIC17C75X | 24CXX 25CXX 93CXX | HCS200 HCS300 HCS301 |
|---|-----------|----------|----------|-----------|----------|-----------|----------|-----------|----------|----------------|-------------------------|----------------------------|
| Emulator Products | | | | | | | | | | | | |
| PICMASTER [®] /PICMASTER-CE In-Circuit Emulator | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | Available 3Q97 | | |
| ICEPIC Low-Cost In-Circuit Emulator | ✓ | | ✓ | ✓ | ✓ | ✓ | ✓ | | | | | |
| Software Tools | | | | | | | | | | | | |
| MPLAB [™] Integrated Development Environment | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | |
| MPLAB [™] C Compiler | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | |
| fuzzyTECH [®] -MP Explorer/Edition Fuzzy Logic Dev. Tool | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | |
| MP-DriveWay [™] Applications Code Generator | | | ✓ | ✓ | ✓ | ✓ | ✓ | | ✓ | | | |
| Total Endurance [™] Software Model | | | | | | | | | | | ✓ | |
| Programmers | | | | | | | | | | | | |
| PICSTART [®] Lite Ultra Low-Cost Dev. Kit | | | ✓ | | ✓ | ✓ | ✓ | | | | | |
| PICSTART [®] Plus Low-Cost Universal Dev. Kit | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | ✓ |
| PRO MATE [®] II Universal Programmer | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| KEELOQ [®] Programmer | | | | | | | | | | | | |
| SEEVAL [®] Designers Kit | | | | | | | | | | | ✓ | |
| PICDEM-1 | | | ✓ | ✓ | | | ✓ | | ✓ | | | |
| PICDEM-2 | | | | | ✓ | ✓ | | | | | | |
| PICDEM-3 | | | | | | | | ✓ | | | | |
| KEELOQ [®] Evaluation Kit | | | | | | | | | | | | ✓ |
| Demo Boards | | | | | | | | | | | | |

PIC16C71X

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

11.0 ELECTRICAL CHARACTERISTICS FOR PIC16C710 AND PIC16C711

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, MCLR, and RA4)..... | -0.3V to (VDD + 0.3V) |
| Voltage on VDD with respect to VSS | -0.3 to +7.5V |
| Voltage on MCLR with respect to VSS..... | 0 to +14V |
| Voltage on RA4 with respect to VSS | 0 to +14V |
| 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 | 200 mA |
| Maximum current sourced by PORTA | 200 mA |
| Maximum current sunk by PORTB..... | 200 mA |
| Maximum current sourced by PORTB..... | 200 mA |

Note 1: Power dissipation is calculated as follows: $P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OI} \times I_{OL})$

† 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.

TABLE 11-1: CROSS REFERENCE OF DEVICE SPECS FOR OSCILLATOR CONFIGURATIONS AND FREQUENCIES OF OPERATION (COMMERCIAL DEVICES)

| OSC | PIC16C710-04 PIC16C711-04 | PIC16C710-10 PIC16C711-10 | PIC16C710-20 PIC16C711-20 | PIC16LC710-04 PIC16LC711-04 | PIC16C710/JW PIC16C711/JW |
|-----|--|---|---|--|--|
| RC | VDD: 4.0V to 6.0V IDD: 5 mA max. at 5.5V IPD: 21 µA max. at 4V Freq: 4 MHz max. | VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max. | VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max. | VDD: 2.5V to 6.0V IDD: 3.8 mA typ. at 3.0V IPD: 5.0 µA typ. at 3V Freq: 4 MHz max. | VDD: 4.0V to 6.0V IDD: 5 mA max. at 5.5V IPD: 21 µA max. at 4V Freq: 4 MHz max. |
| XT | VDD: 4.0V to 6.0V IDD: 5 mA max. at 5.5V IPD: 21 µA max. at 4V Freq: 4 MHz max. | VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max. | VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max. | VDD: 2.5V to 6.0V IDD: 3.8 mA typ. at 3.0V IPD: 5.0 µA typ. at 3V Freq: 4 MHz max. | VDD: 4.0V to 6.0V IDD: 5 mA max. at 5.5V IPD: 21 µA max. at 4V Freq: 4 MHz max. |
| HS | VDD: 4.5V to 5.5V IDD: 13.5 mA typ. at 5.5V IPD: 1.5 µA typ. at 4.5V Freq: 4 MHz max. | VDD: 4.5V to 5.5V IDD: 30 mA max. at 5.5V IPD: 1.5 µA typ. at 4.5V Freq: 10 MHz max. | VDD: 4.5V to 5.5V IDD: 30 mA max. at 5.5V IPD: 1.5 µA typ. at 4.5V Freq: 20 MHz max. | Not recommended for use in HS mode | VDD: 4.5V to 5.5V IDD: 30 mA max. at 5.5V IPD: 1.5 µA typ. at 4.5V Freq: 10 MHz max. |
| LP | VDD: 4.0V to 6.0V IDD: 52.5 µA typ. at 32 kHz, 4.0V IPD: 0.9 µA typ. at 4.0V Freq: 200 kHz max. | Not recommended for use in LP mode | Not recommended for use in LP mode | VDD: 2.5V to 6.0V IDD: 48 µA max. at 32 kHz, 3.0V IPD: 5.0 µA max. at 3.0V Freq: 200 kHz max. | VDD: 2.5V to 6.0V IDD: 48 µA max. at 32 kHz, 3.0V IPD: 5.0 µA max. at 3.0V Freq: 200 kHz max. |

PIC16C71X

| | | | | |
|--------------------|-----|----|-----|-----|
| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|

11.1 DC Characteristics: PIC16C710-04 (Commercial, Industrial, Extended)
PIC16C711-04 (Commercial, Industrial, Extended)
PIC16C710-10 (Commercial, Industrial, Extended)
PIC16C711-10 (Commercial, Industrial, Extended)
PIC16C710-20 (Commercial, Industrial, Extended)
PIC16C711-20 (Commercial, Industrial, Extended)

| Standard Operating Conditions (unless otherwise stated) | | | | | | | |
|--|--|-------|------------------|---------------------------|----------------------|----------------------|---|
| DC CHARACTERISTICS | | | | | | | |
| Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended) | | | | | | | |
| Param. No. | Characteristic | Sym | Min | Typ† | Max | Units | Conditions |
| D001 D001A | Supply Voltage | VDD | 4.0 4.5 | - - | 6.0 5.5 | V V | XT, RC and LP osc configuration HS osc configuration |
| D002* | RAM Data Retention Voltage (Note 1) | VDR | - | 1.5 | - | V | |
| D003 | VDD start voltage to ensure internal Power-on Reset signal | VPOR | - | VSS | - | V | See section on Power-on Reset for details |
| D004* | VDD rise rate to ensure internal Power-on Reset signal | SVDD | 0.05 | - | - | V/ms | See section on Power-on Reset for details |
| D005 | Brown-out Reset Voltage | BVDD | 3.7 3.7 | 4.0 4.0 | 4.3 4.4 | V V | BODEN configuration bit is enabled Extended Range Only |
| D010 D013 | Supply Current (Note 2) | IDD | - - | 2.7 13.5 | 5 30 | mA mA | XT, RC osc configuration FOSC = 4 MHz, VDD = 5.5V (Note 4) HS osc configuration FOSC = 20 MHz, VDD = 5.5V |
| D015 | Brown-out Reset Current (Note 5) | ΔIBOR | - | 300* | 500 | μA | BOR enabled VDD = 5.0V |
| D020 D021 D021A D021B | Power-down Current (Note 3) | IPD | - - - - | 10.5 1.5 1.5 1.5 | 42 21 24 30 | μA μA μA μA | VDD = 4.0V, WDT enabled, -40°C to +85°C VDD = 4.0V, WDT disabled, -0°C to +70°C VDD = 4.0V, WDT disabled, -40°C to +85°C VDD = 4.0V, WDT disabled, -40°C to +125°C |
| D023 | Brown-out Reset Current (Note 5) | ΔIBOR | - | 300* | 500 | μA | BOR enabled VDD = 5.0V |

* 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 and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula $I_r = VDD/2R_{ext}$ (mA) with Rext in kOhm.

5: The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.

PIC16C71X

Applicable Devices 710 71 711 715

11.2 DC Characteristics: PIC16LC710-04 (Commercial, Industrial, Extended) PIC16LC711-04 (Commercial, Industrial, Extended)

| Standard Operating Conditions (unless otherwise stated) | | | | | | | |
|--|--|------------|------------|--------|------------|--------|--|
| DC CHARACTERISTICS | | | | | | | |
| Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended) | | | | | | | |
| Param No. | Characteristic | Sym | Min | Typ† | Max | Units | Conditions |
| D001 | Supply Voltage Commercial/Industrial Extended | VDD VDD | 2.5 3.0 | - - | 6.0 6.0 | V V | LP, XT, RC osc configuration (DC - 4 MHz) LP, XT, RC osc configuration (DC - 4 MHz) |
| D002* | RAM Data Retention Voltage (Note 1) | VDR | - | 1.5 | - | V | |
| D003 | VDD start voltage to ensure internal Power-on Reset signal | VPOR | - | VSS | - | V | See section on Power-on Reset for details |
| D004* | VDD rise rate to ensure internal Power-on Reset signal | SVDD | 0.05 | - | - | V/ms | See section on Power-on Reset for details |
| D005 | Brown-out Reset Voltage | BVDD | 3.7 | 4.0 | 4.3 | V | BODEN configuration bit is enabled |
| D010 | Supply Current (Note 2) | IDD | - | 2.0 | 3.8 | mA | XT, RC osc configuration FOSC = 4 MHz, VDD = 3.0V (Note 4) |
| D010A | | | - | 22.5 | 48 | μA | LP osc configuration FOSC = 32 kHz, VDD = 3.0V, WDT disabled |
| D015 | Brown-out Reset Current (Note 5) | ΔIBOR | - | 300* | 500 | μA | BOR enabled VDD = 5.0V |
| D020 | Power-down Current (Note 3) | IPD | - | 7.5 | 30 | μA | VDD = 3.0V, WDT enabled, -40°C to +85°C |
| D021 | | | - | 0.9 | 5 | μA | VDD = 3.0V, WDT disabled, 0°C to +70°C |
| D021A | | | - | 0.9 | 5 | μA | VDD = 3.0V, WDT disabled, -40°C to +85°C |
| D021B | | | - | 0.9 | 10 | μA | VDD = 3.0V, WDT disabled, -40°C to +125°C |
| D023 | Brown-out Reset Current (Note 5) | ΔIBOR | - | 300* | 500 | μA | BOR enabled VDD = 5.0V |

* 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 and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula $I_r = VDD/2R_{ext}$ (mA) with Rext in kOhm.

5: The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.

PIC16C71X

| | | | | |
|--------------------|-----|----|-----|-----|
| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|

11.3 DC Characteristics: PIC16C710-04 (Commercial, Industrial, Extended)
PIC16C711-04 (Commercial, Industrial, Extended)
PIC16C710-10 (Commercial, Industrial, Extended)
PIC16C711-10 (Commercial, Industrial, Extended)
PIC16C710-20 (Commercial, Industrial, Extended)
PIC16C711-20 (Commercial, Industrial, Extended)
PIC16LC710-04 (Commercial, Industrial, Extended)
PIC16LC711-04 (Commercial, Industrial, Extended)

| Standard Operating Conditions (unless otherwise stated) | | | | | | | |
|--|-----------------------------|-----------------|-----------------|-------|---------------------|-------|---|
| Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended) | | | | | | | |
| Operating voltage VDD range as described in DC spec Section 11.1 and Section 11.2. | | | | | | | |
| Param No. | Characteristic | Sym | Min | Typ † | Max | Units | Conditions |
| Input Low Voltage | | | | | | | |
| D030 | I/O ports | V _{IL} | V _{SS} | - | 0.15V _{DD} | V | For entire VDD range 4.5 ≤ VDD ≤ 5.5V |
| D030A | with TTL buffer | | V _{SS} | - | 0.8V | V | |
| D031 | with Schmitt Trigger buffer | | V _{SS} | - | 0.2V _{DD} | V | |
| D032 | MCLR, OSC1 (in RC mode) | | V _{SS} | - | 0.2V _{DD} | V | |
| D033 | OSC1 (in XT, HS and LP) | | V _{SS} | - | 0.3V _{DD} | V | Note1 |
| Input High Voltage | | | | | | | |
| D040 | I/O ports | V _{IH} | | - | | | 4.5 ≤ VDD ≤ 5.5V |
| D040A | with TTL buffer | | 2.0 | - | VDD | V | |
| | | | 0.25VDD + 0.8V | - | VDD | V | For entire VDD range |
| D041 | with Schmitt Trigger buffer | | 0.8VDD | - | VDD | V | For entire VDD range |
| D042 | MCLR, RB0/INT | | 0.8VDD | - | VDD | V | Note1 |
| D042A | OSC1 (XT, HS and LP) | | 0.7VDD | - | VDD | V | |
| D043 | OSC1 (in RC mode) | | 0.9VDD | - | VDD | V | |
| D070 | PORTB weak pull-up current | IPURB | 50 | 250 | 400 | μA | VDD = 5V, VPIN = VSS |
| Input Leakage Current (Notes 2, 3) | | | | | | | |
| D060 | I/O ports | I _{IL} | - | - | ±1 | μA | VSS ≤ VPIN ≤ VDD, Pin at hi-impedance |
| D061 | MCLR, RA4/T0CKI | | - | - | ±5 | μA | VSS ≤ VPIN ≤ VDD |
| D063 | OSC1 | | - | - | ±5 | μA | VSS ≤ VPIN ≤ VDD, XT, HS and LP osc configuration |

* 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: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C7X be driven with external clock in RC mode.

2: The leakage current on the 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.

3: Negative current is defined as current sourced by the pin.

PIC16C71X

Applicable Devices 710 71 711 715

| Standard Operating Conditions (unless otherwise stated) | | | | | | | |
|---|--|-------|-----------|-------|-----|-------|---|
| DC CHARACTERISTICS Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended) Operating voltage VDD range as described in DC spec Section 11.1 and Section 11.2. | | | | | | | |
| Param No. | Characteristic | Sym | Min | Typ † | Max | Units | Conditions |
| D080 | Output Low Voltage I/O ports | VOL | - | - | 0.6 | V | IoL = 8.5 mA, VDD = 4.5V, -40°C to +85°C |
| D080A | | | - | - | 0.6 | V | IoL = 7.0 mA, VDD = 4.5V, -40°C to +125°C |
| D083 | OSC2/CLKOUT (RC osc config) | | - | - | 0.6 | V | IoL = 1.6 mA, VDD = 4.5V, -40°C to +85°C |
| D083A | | | - | - | 0.6 | V | IoL = 1.2 mA, VDD = 4.5V, -40°C to +125°C |
| D090 | Output High Voltage I/O ports (Note 3) | VOH | VDD - 0.7 | - | - | V | IoH = -3.0 mA, VDD = 4.5V, -40°C to +85°C |
| D090A | | | VDD - 0.7 | - | - | V | IoH = -2.5 mA, VDD = 4.5V, -40°C to +125°C |
| D092 | OSC2/CLKOUT (RC osc config) | | VDD - 0.7 | - | - | V | IoH = -1.3 mA, VDD = 4.5V, -40°C to +85°C |
| D092A | | | VDD - 0.7 | - | - | V | IoH = -1.0 mA, VDD = 4.5V, -40°C to +125°C |
| D130* | Open-Drain High Voltage | VOD | - | - | 14 | V | RA4 pin |
| Capacitive Loading Specs on Output Pins | | | | | | | |
| D100 | OSC2 pin | COSC2 | - | - | 15 | pF | In XT, HS and LP modes when external clock is used to drive OSC1. |
| D101 | All I/O pins and OSC2 (in RC mode) | CIO | - | - | 50 | pF | |

* 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: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C7X be driven with external clock in RC mode.
- 2: The leakage current on the 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.
- 3: Negative current is defined as current sourced by the pin.

PIC16C71X

| | | | | |
|--------------------|-----|----|-----|-----|
| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|

11.4 Timing Parameter Symbology

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

- 1. TppS2ppS
- 2. TppS

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

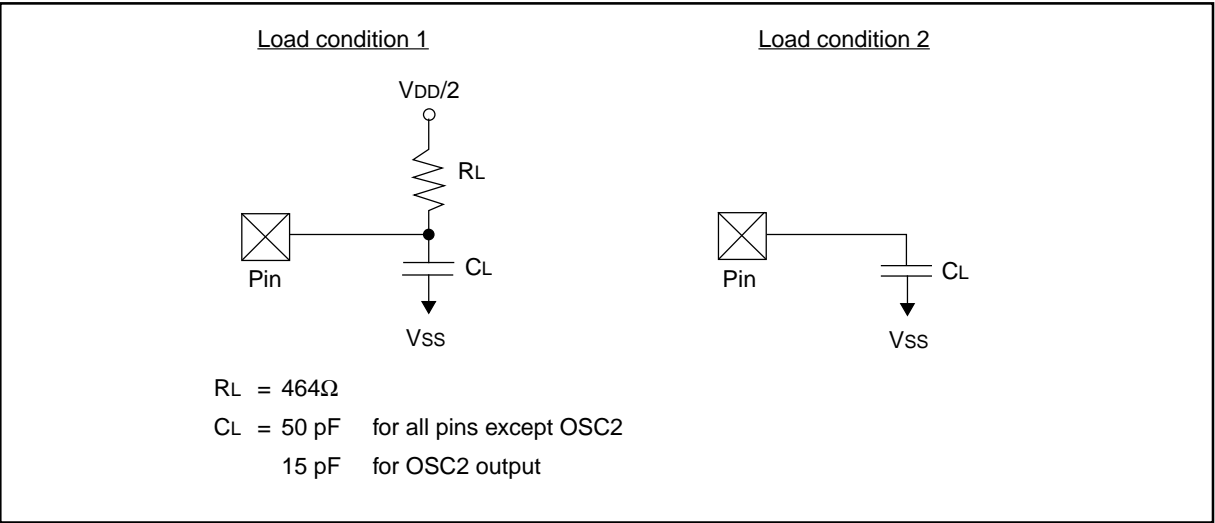
Lowercase letters (pp) and their meanings:

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

Uppercase letters and their meanings:

| | | |
|----------|------------------------|--------------|
| S | | |
| F | Fall | P |
| H | High | R |
| I | Invalid (Hi-impedance) | V |
| L | Low | Z |
| | | Period |
| | | Rise |
| | | Valid |
| | | Hi-impedance |

FIGURE 11-1: LOAD CONDITIONS



11.5 Timing Diagrams and Specifications

FIGURE 11-2: EXTERNAL CLOCK TIMING

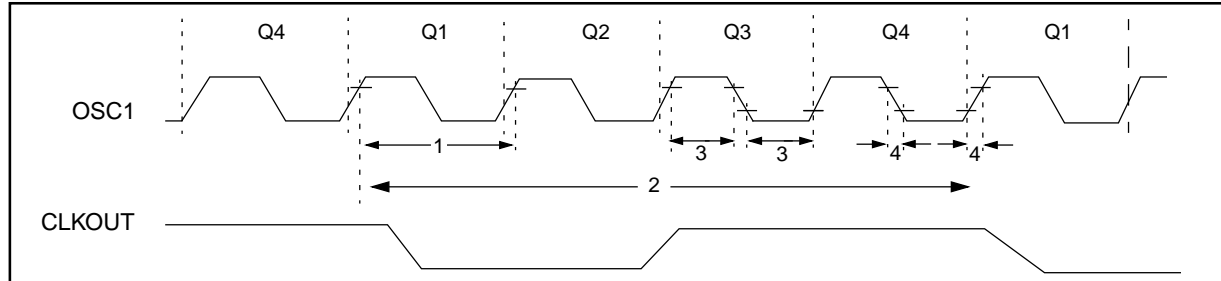


TABLE 11-2: EXTERNAL CLOCK TIMING REQUIREMENTS

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|------------|---|-----|------|--------|-------|-------------------|
| | Fosc | External CLKIN Frequency (Note 1) | DC | — | 4 | MHz | XT osc mode |
| | | | DC | — | 4 | MHz | HS osc mode (-04) |
| | | | DC | — | 10 | MHz | HS osc mode (-10) |
| | | | DC | — | 20 | MHz | HS osc mode (-20) |
| | | | DC | — | 200 | kHz | LP osc mode |
| | | Oscillator Frequency (Note 1) | DC | — | 4 | MHz | RC osc mode |
| | | | 0.1 | — | 4 | MHz | XT osc mode |
| | | | 4 | — | 20 | MHz | HS osc mode |
| | | | 5 | — | 200 | kHz | LP osc mode |
| | | | — | — | — | — | — |
| 1 | Tosc | External CLKIN Period (Note 1) | 250 | — | — | ns | XT osc mode |
| | | | 250 | — | — | ns | HS osc mode (-04) |
| | | | 100 | — | — | ns | HS osc mode (-10) |
| | | | 50 | — | — | ns | HS osc mode (-20) |
| | | | 5 | — | — | μs | LP osc mode |
| | | Oscillator Period (Note 1) | 250 | — | — | ns | RC osc mode |
| | | | 250 | — | 10,000 | ns | XT osc mode |
| | | | 250 | — | 250 | ns | HS osc mode (-04) |
| | | | 100 | — | 250 | ns | HS osc mode (-10) |
| | | | 50 | — | 250 | ns | HS osc mode (-20) |
| 2 | TCY | Instruction Cycle Time (Note 1) | 200 | — | DC | ns | TCY = 4/FOSC |
| | | | — | — | — | — | — |
| | | | — | — | — | — | — |
| | | | — | — | — | — | — |
| 3 | TosL, TosH | External Clock in (OSC1) High or Low Time | 50 | — | — | ns | XT oscillator |
| | | | 2.5 | — | — | μs | LP oscillator |
| | | | 10 | — | — | ns | HS oscillator |
| 4 | TosR, TosF | External Clock in (OSC1) Rise or Fall Time | — | — | 25 | ns | XT oscillator |
| | | | — | — | 50 | ns | LP oscillator |
| | | | — | — | 15 | ns | HS oscillator |

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

Note 1: Instruction cycle period (TCY) equals four times the input oscillator time-base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min." values with an external clock applied to the OSC1/CLKIN pin. When an external clock input is used, the "Max." cycle time limit is "DC" (no clock) for all devices. OSC2 is disconnected (has no loading) for the PIC16C710/711.

PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 11-3: CLKOUT AND I/O TIMING

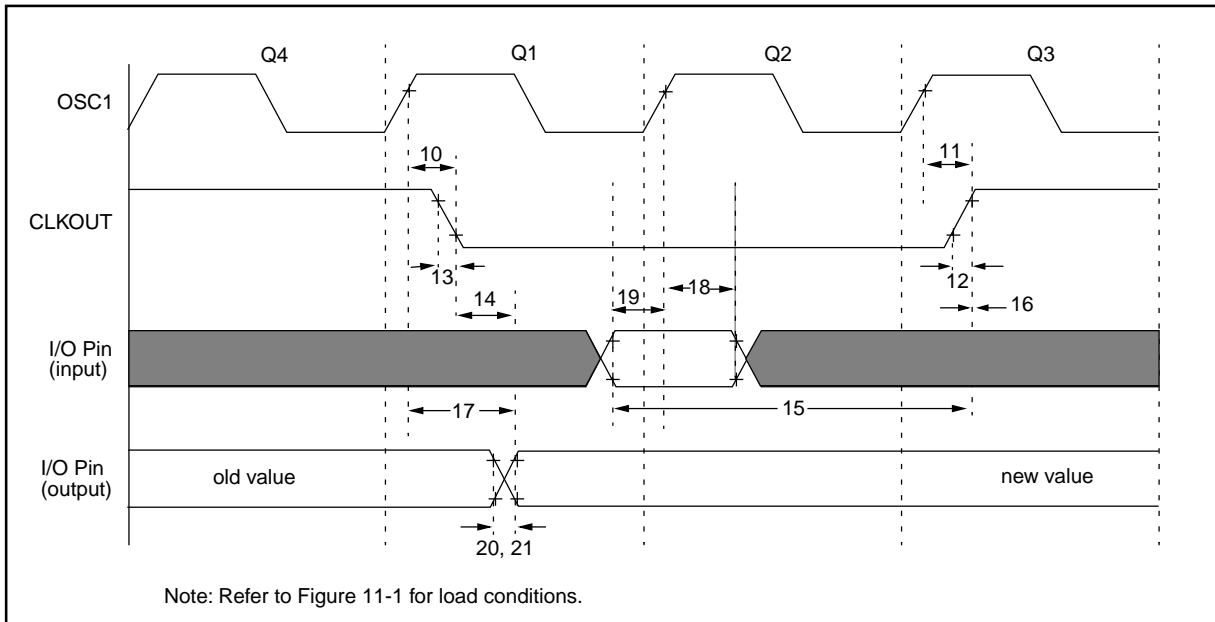


TABLE 11-3: CLKOUT AND I/O TIMING REQUIREMENTS

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|----------|---|--------------------------|------|-------------------------|-------|------------|
| 10* | TosH2ckL | OSC1↑ to CLKOUT↓ | — | 15 | 30 | ns | Note 1 |
| 11* | TosH2ckH | OSC1↑ to CLKOUT↑ | — | 15 | 30 | ns | Note 1 |
| 12* | TckR | CLKOUT rise time | — | 5 | 15 | ns | Note 1 |
| 13* | TckF | CLKOUT fall time | — | 5 | 15 | 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* | TckH2ioI | Port in hold after CLKOUT ↑ | 0 | — | — | ns | Note 1 |
| 17* | TosH2ioV | OSC1↑ (Q1 cycle) to Port out valid | — | — | 80 - 100 | ns | |
| 18* | TosH2ioI | OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time) | TBD | — | — | ns | |
| 19* | TioV2osH | Port input valid to OSC1↑ (I/O in setup time) | TBD | — | — | ns | |
| 20* | TioR | Port output rise time | PIC16C710/711 — | 10 | 25 | ns | |
| | | | PIC16LC710/711 — | — | 60 | ns | |
| 21* | TioF | Port output fall time | PIC16C710/711 — | 10 | 25 | ns | |
| | | | PIC16LC710/711 — | — | 60 | ns | |
| 22††* | Tinp | INT pin high or low time | 20 | — | — | ns | |
| 23††* | Trbp | RB7:RB4 change INT high or low time | 20 | — | — | ns | |

* These parameters are characterized but not tested.

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

†† These parameters are asynchronous events not related to any internal clock edges.

Note 1: Measurements are taken in RC Mode where CLKOUT output is 4 x T_{OSC}.

FIGURE 11-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING

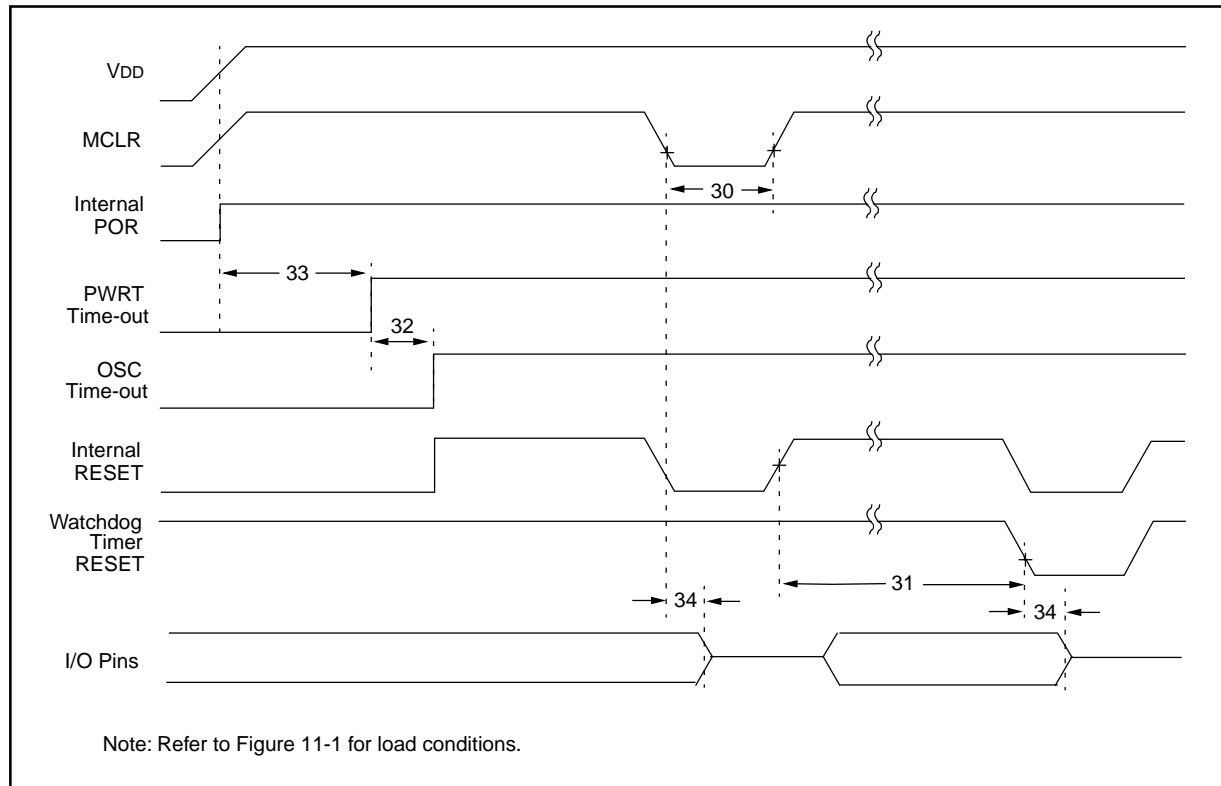


FIGURE 11-5: BROWN-OUT RESET TIMING

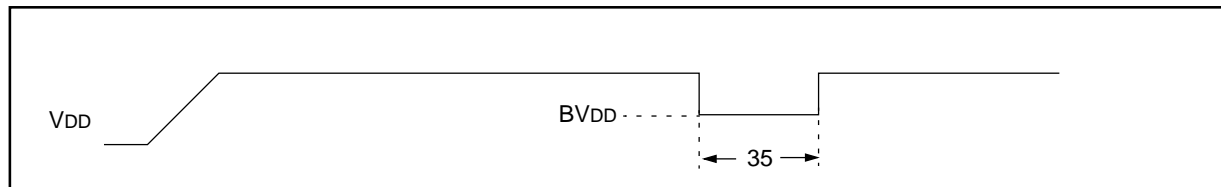


TABLE 11-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER, AND BROWN-OUT RESET REQUIREMENTS

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|-------|--|-----|----------|------|-------|---------------------------|
| 30 | Tmcl | MCLR Pulse Width (low) | 1 | — | — | μs | VDD = 5V, -40°C to +125°C |
| 31 | Twdt | Watchdog Timer Time-out Period (No Prescaler) | 7* | 18 | 33* | ms | VDD = 5V, -40°C to +125°C |
| 32 | Tost | Oscillation Start-up Timer Period | — | 1024Tosc | — | — | Tosc = OSC1 period |
| 33 | Tpwrt | Power up Timer Period | 28* | 72 | 132* | ms | VDD = 5V, -40°C to +125°C |
| 34 | Tioz | I/O Hi-impedance from MCLR Low or Watchdog Timer Reset | — | — | 1.1 | μs | |
| 35 | TBOR | Brown-out Reset pulse width | 100 | — | — | μs | 3.8V ≤ VDD ≤ 4.2V |

* 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.

PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 11-6: TIMER0 EXTERNAL CLOCK TIMINGS

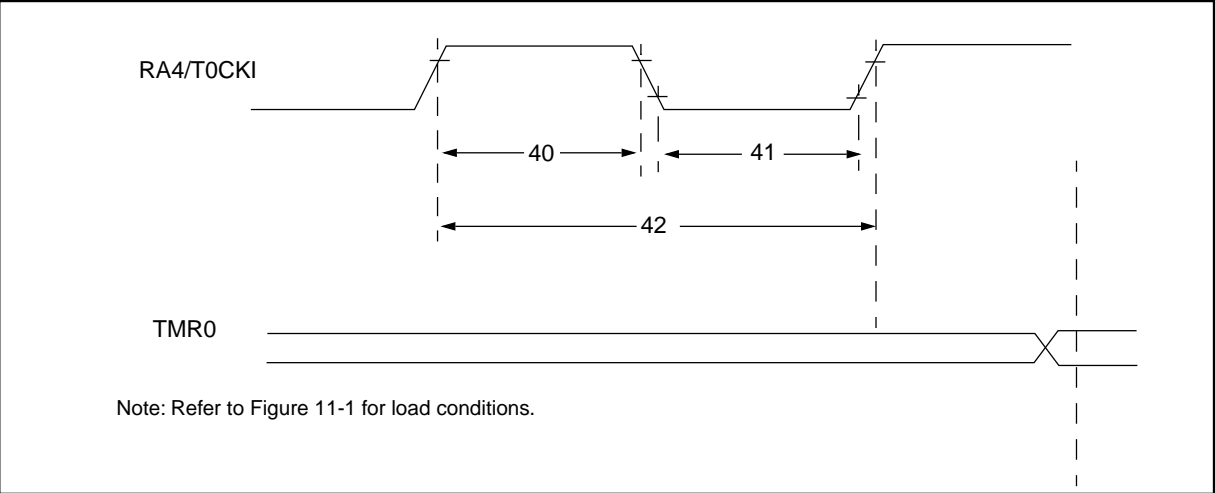


TABLE 11-5: TIMER0 EXTERNAL CLOCK REQUIREMENTS

| 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 | | Greater of: $20 \text{ ns or } \frac{T_{CY} + 40^*}{N}$ | — | — | ns | N = prescale value (2, 4,..., 256) |
| 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.

PIC16C71X

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

TABLE 11-6: A/D CONVERTER CHARACTERISTICS:
PIC16C710/711-04 (COMMERCIAL, INDUSTRIAL, EXTENDED)
PIC16C710/711-10 (COMMERCIAL, INDUSTRIAL, EXTENDED)
PIC16C710/711-20 (COMMERCIAL, INDUSTRIAL, EXTENDED)
PIC16LC710/711-04 (COMMERCIAL, INDUSTRIAL, EXTENDED)

| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|-----------|------|--|----------------|------------|-----------------|------------|---|
| A01 | NR | Resolution | — | — | 8-bits | bit | $V_{REF} = V_{DD}$, $V_{SS} \leq AIN \leq V_{REF}$ |
| A02 | EABS | Absolute error | — | — | $< \pm 1$ | LSb | $V_{REF} = V_{DD}$, $V_{SS} \leq AIN \leq V_{REF}$ |
| A03 | EIL | Integral linearity error | — | — | $< \pm 1$ | LSb | $V_{REF} = V_{DD}$, $V_{SS} \leq AIN \leq V_{REF}$ |
| A04 | EDL | Differential linearity error | — | — | $< \pm 1$ | LSb | $V_{REF} = V_{DD}$, $V_{SS} \leq AIN \leq V_{REF}$ |
| A05 | EFS | Full scale error | — | — | $< \pm 1$ | LSb | $V_{REF} = V_{DD}$, $V_{SS} \leq AIN \leq V_{REF}$ |
| A06 | EOFF | Offset error | — | — | $< \pm 1$ | LSb | $V_{REF} = V_{DD}$, $V_{SS} \leq AIN \leq V_{REF}$ |
| A10 | — | Monotonicity | — | guaranteed | — | — | $V_{SS} \leq V_{AIN} \leq V_{REF}$ |
| A20 | VREF | Reference voltage | 2.5V | — | $V_{DD} + 0.3$ | V | |
| A25 | VAIN | Analog input voltage | $V_{SS} - 0.3$ | — | $V_{REF} + 0.3$ | V | |
| A30 | ZAIN | Recommended impedance of analog voltage source | — | — | 10.0 | k Ω | |
| A40 | IAD | A/D conversion current (V_{DD}) | — | 180 | — | μA | Average current consumption when A/D is on. (Note 1) |
| A50 | IREF | VREF input current (Note 2) | 10 | — | 1000 | μA | During VAIN acquisition. Based on differential of V_{HOLD} to VAIN. To charge C_{HOLD} see Section 7.1. During A/D Conversion cycle |
| | | | — | — | 10 | μA | |

* 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 current is from RA3 pin or VDD pin, whichever is selected as reference input.

PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 11-7: A/D CONVERSION TIMING

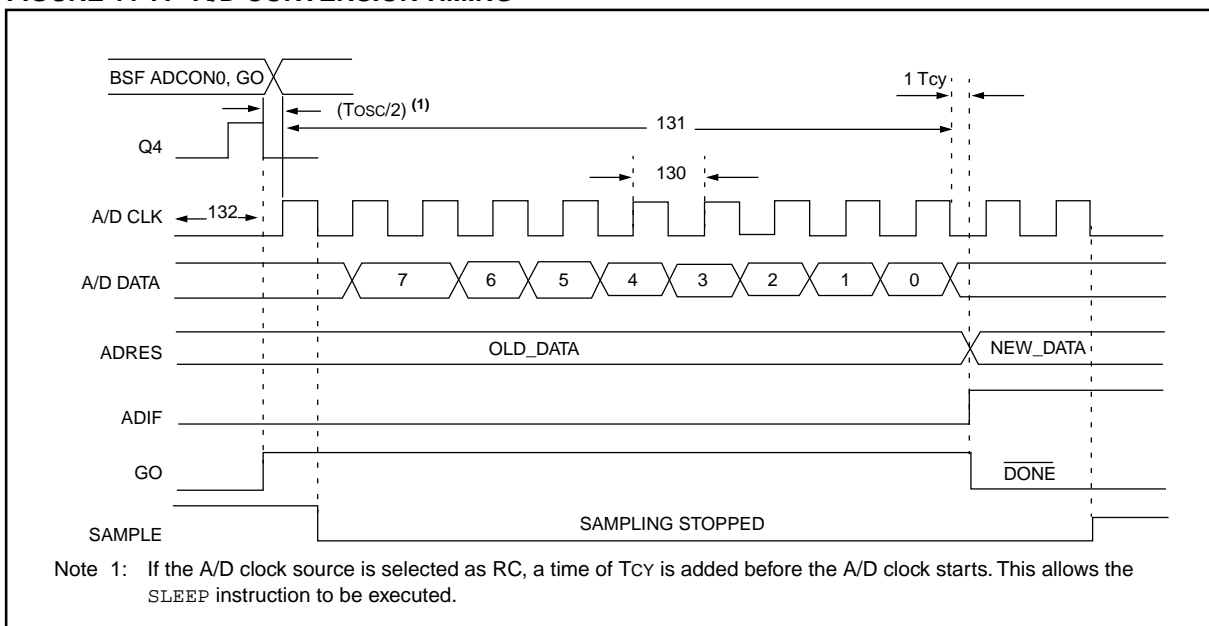


TABLE 11-7: A/D CONVERSION REQUIREMENTS

| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|-----------|------|--|----------------|--------------|-----|---------|---|
| 130 | TAD | A/D clock period | PIC16C710/711 | 1.6 | — | — | μs TOSC based, $V_{REF} \geq 3.0V$ |
| | | | PIC16LC710/711 | 2.0 | — | — | μs TOSC based, V_{REF} full range |
| | | | PIC16C710/711 | 2.0* | 4.0 | 6.0 | μs A/D RC mode |
| | | | PIC16LC710/711 | 3.0* | 6.0 | 9.0 | μs A/D RC mode |
| 131 | TCNV | Conversion time (not including S/H time). (Note 1) | — | 9.5 | — | TAD | |
| 132 | TACQ | Acquisition time | Note 2 | 20 | — | μs | The minimum time is the amplifier settling time. This may be used if the "new" input voltage has not changed by more than 1 LSb (i.e., 19.5 mV @ 5.12V) from the last sampled voltage (as stated on CHOLD). |
| | | | 5* | — | — | μs | |
| 134 | TGO | Q4 to AD clock start | — | $T_{osc}/2§$ | — | — | If the A/D clock source is selected as RC, a time of T_{cy} is added before the A/D clock starts. This allows the SLEEP instruction to be executed. |
| 135 | TSWC | Switching from convert → sample time | 1.5§ | — | — | TAD | |

* 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.

§ This specification ensured by design.

Note 1: ADRES register may be read on the following T_{cy} cycle.

2: See Section 7.1 for min conditions.

12.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES FOR PIC16C710 AND PIC16C711

The graphs and tables provided in this section are for design guidance and are not tested or guaranteed.

In some graphs or tables the data presented are outside specified operating range (i.e., outside specified VDD range). This is for information only and devices are guaranteed to operate properly only within the specified range.

Note: The data presented in this section is a statistical summary of data collected on units from different lots over a period of time and matrix samples. 'Typical' represents the mean of the distribution at, 25°C, while 'max' or 'min' represents (mean +3σ) and (mean -3σ) respectively where σ is standard deviation.

FIGURE 12-1: TYPICAL IPD vs. VDD (WDT DISABLED, RC MODE)

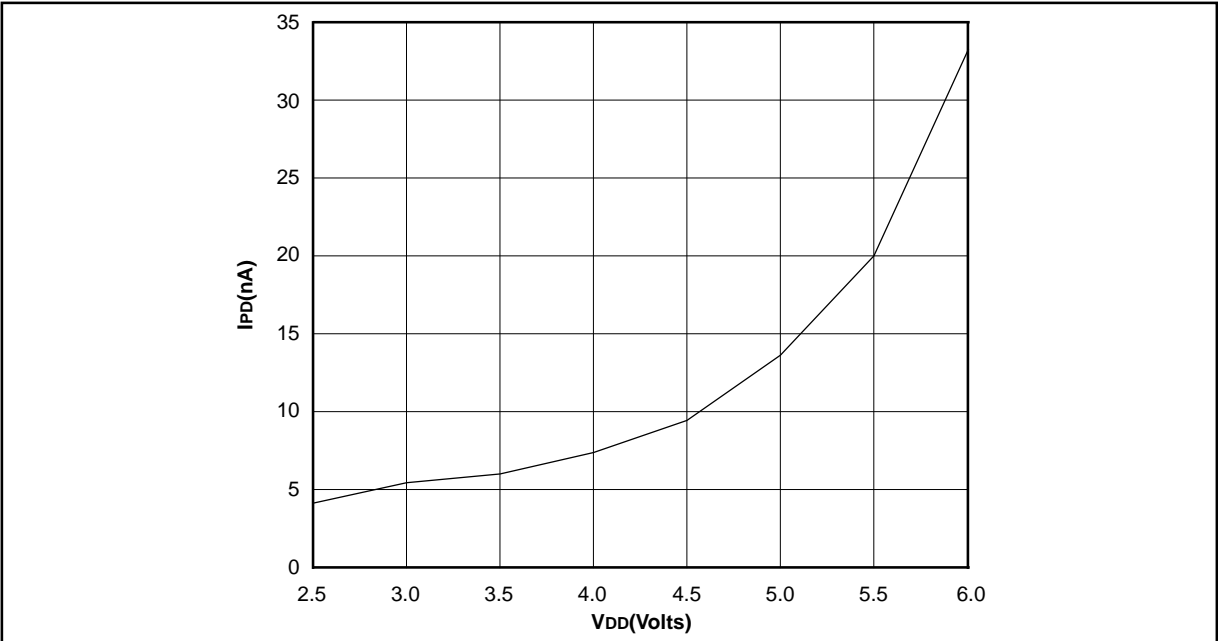
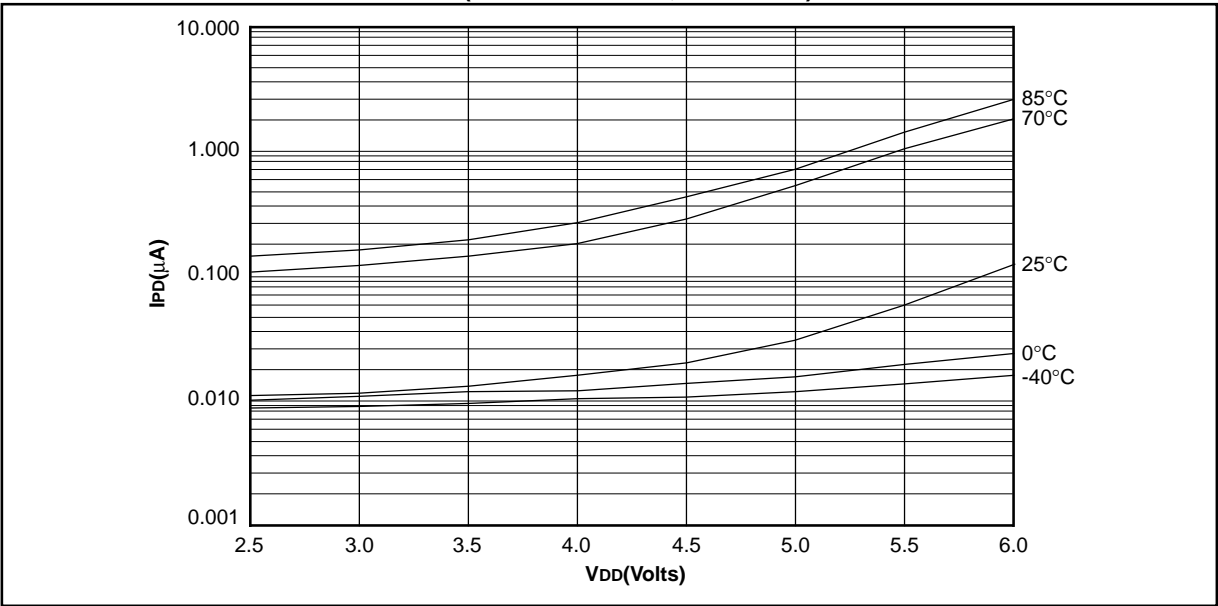


FIGURE 12-2: MAXIMUM IPD vs. VDD (WDT DISABLED, RC MODE)



PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 12-3: TYPICAL IPD vs. VDD @ 25°C (WDT ENABLED, RC MODE)

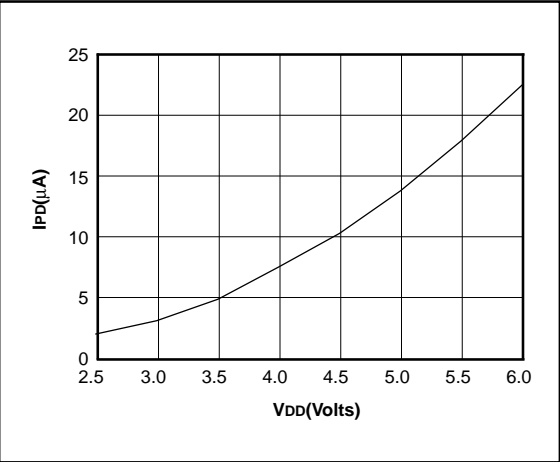


FIGURE 12-4: MAXIMUM IPD vs. VDD (WDT ENABLED, RC MODE)

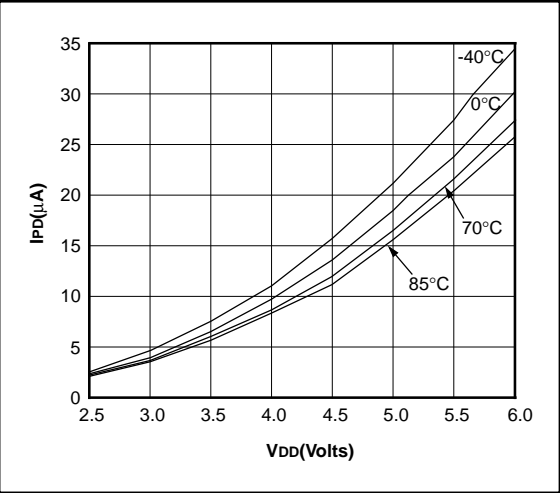


FIGURE 12-5: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

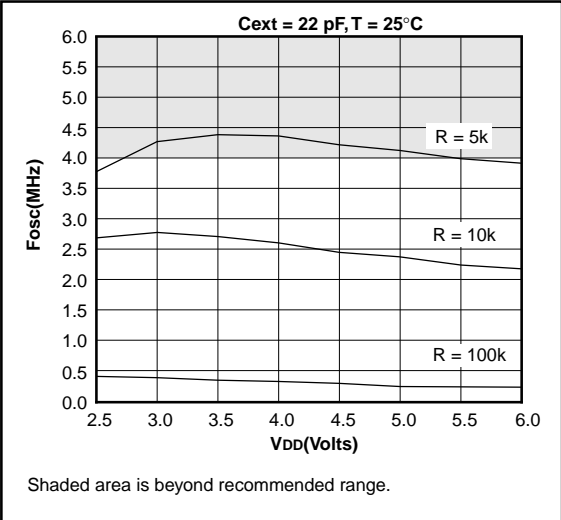


FIGURE 12-6: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

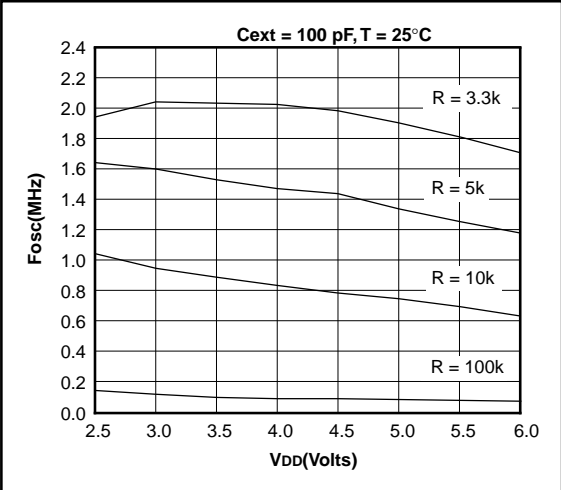


FIGURE 12-7: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

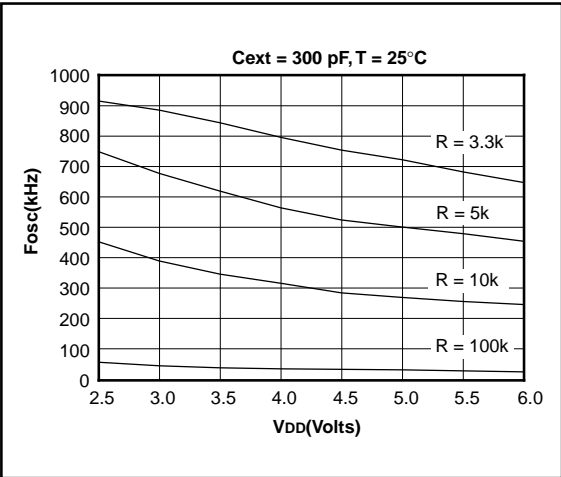


FIGURE 12-8: TYPICAL IPD vs. VDD BROWN-OUT DETECT ENABLED (RC MODE)

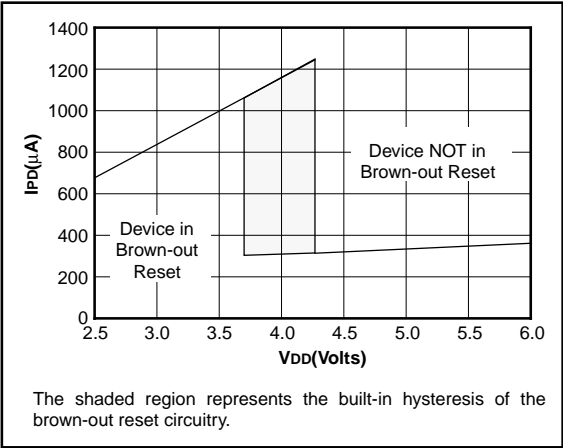


FIGURE 12-9: MAXIMUM IPD vs. VDD BROWN-OUT DETECT ENABLED (85°C TO -40°C, RC MODE)

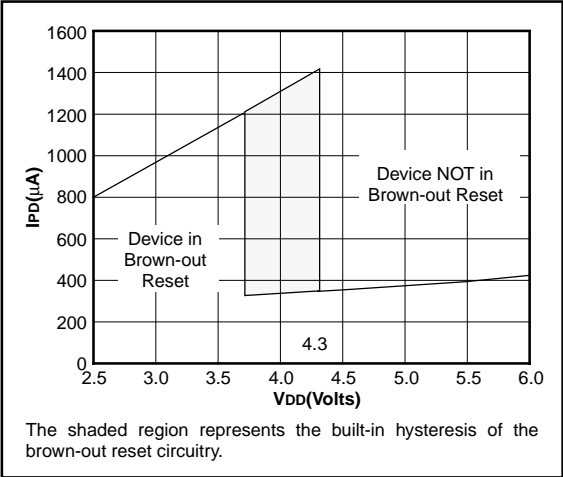


FIGURE 12-10: TYPICAL IPD vs. TIMER1 ENABLED (32 kHz, RC0/RC1 = 33 pF/33 pF, RC MODE)

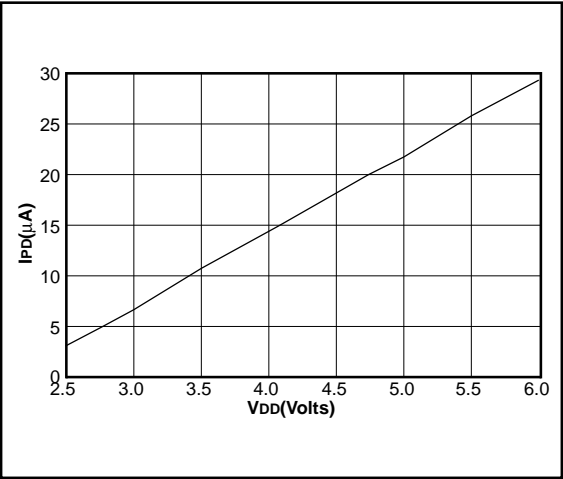
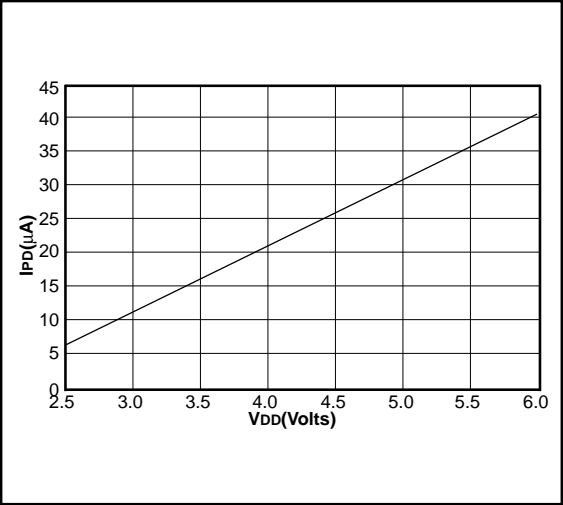


FIGURE 12-11: MAXIMUM IPD vs. TIMER1 ENABLED (32 kHz, RC0/RC1 = 33 pF/33 pF, 85°C TO -40°C, RC MODE)



PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 12-12: TYPICAL I_{DD} vs. FREQUENCY (RC MODE @ 22 pF, 25°C)

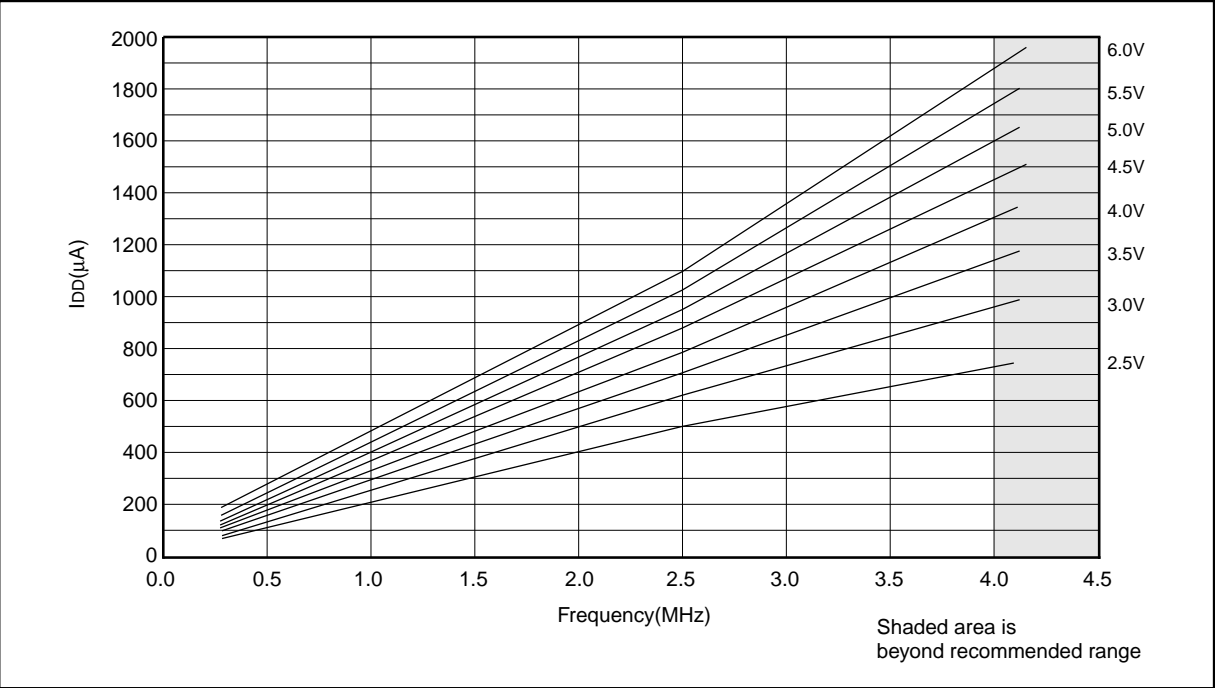
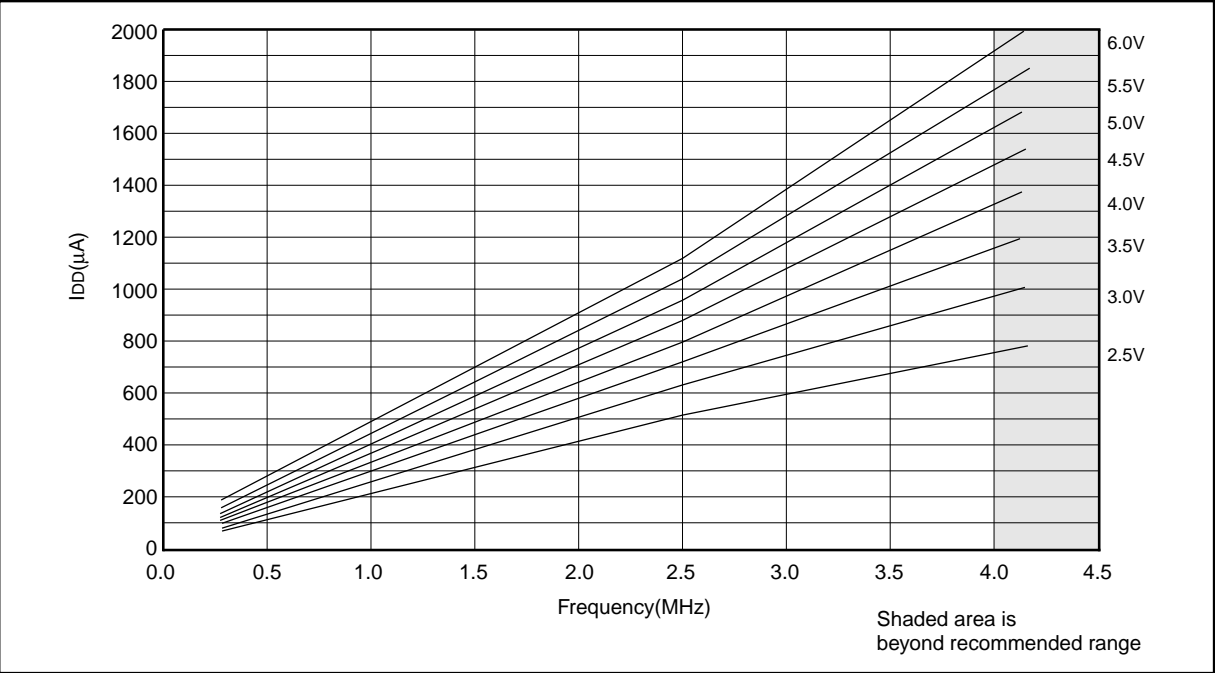


FIGURE 12-13: MAXIMUM I_{DD} vs. FREQUENCY (RC MODE @ 22 pF, -40°C TO 85°C)



PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 12-14: TYPICAL I_{DD} vs. FREQUENCY (RC MODE @ 100 pF, 25°C)

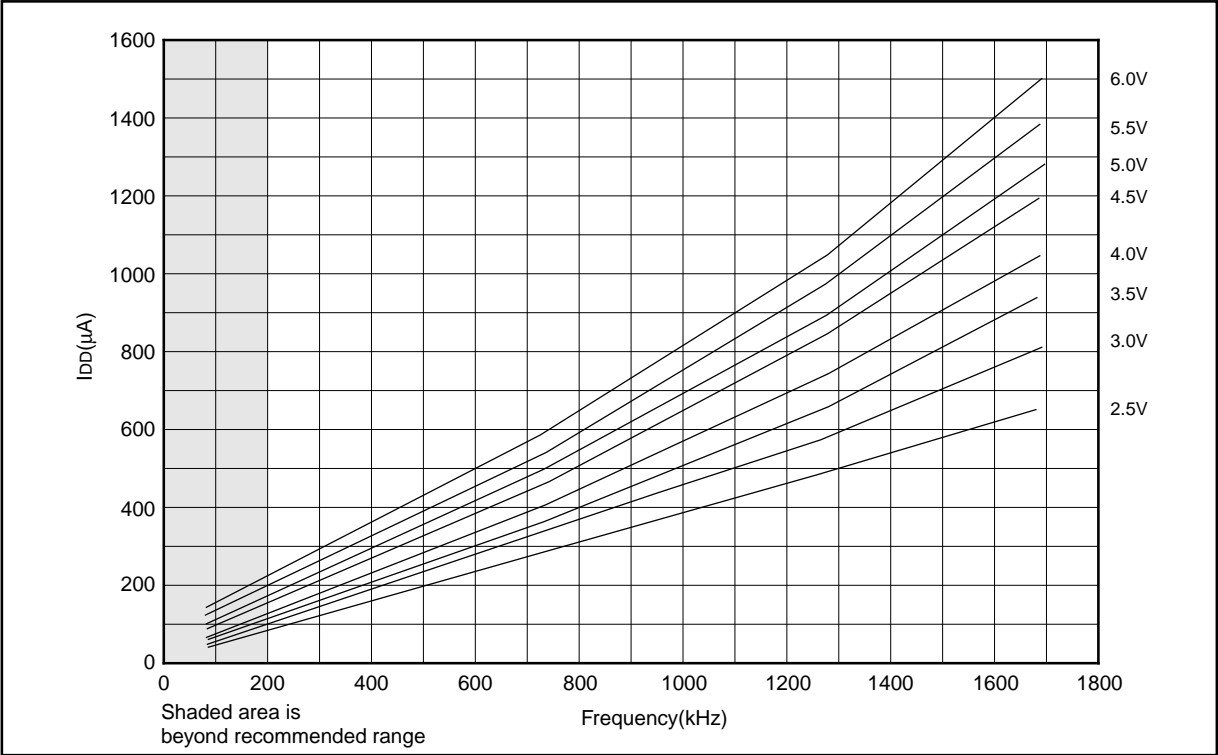
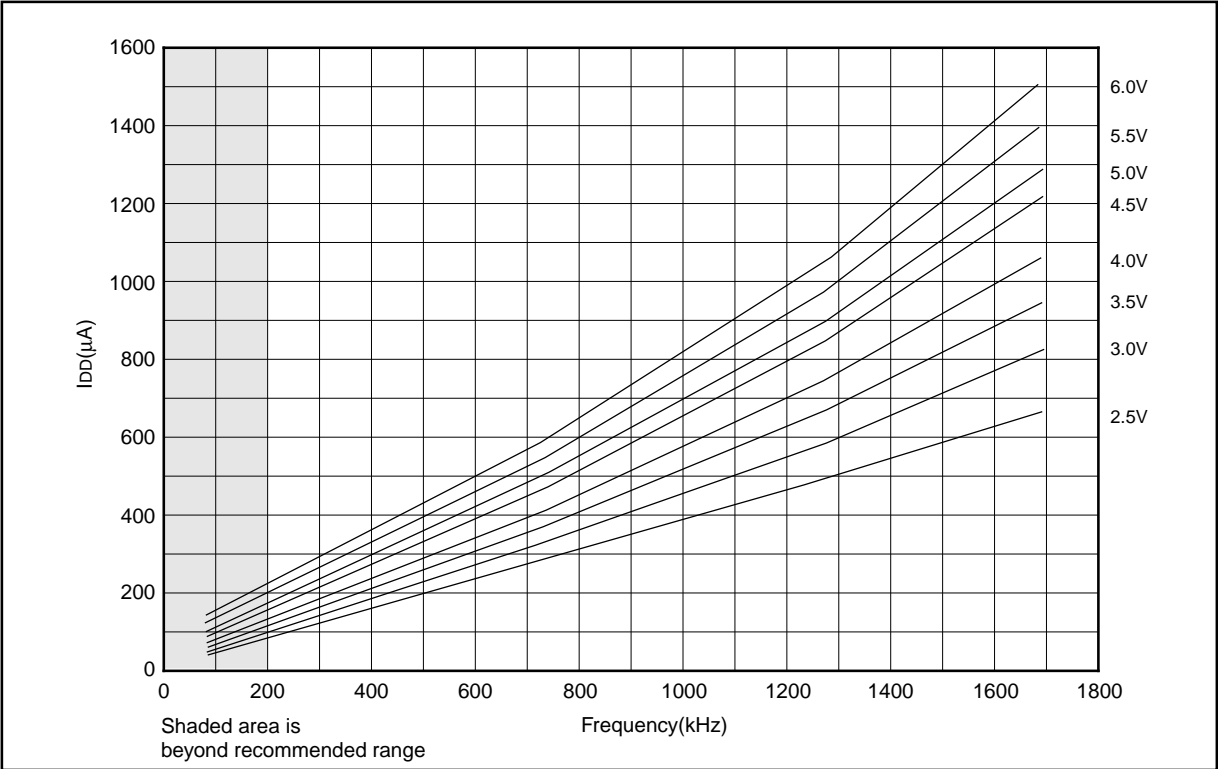


FIGURE 12-15: MAXIMUM I_{DD} vs. FREQUENCY (RC MODE @ 100 pF, -40°C TO 85°C)



PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 12-16: TYPICAL I_{DD} vs. FREQUENCY (RC MODE @ 300 pF, 25°C)

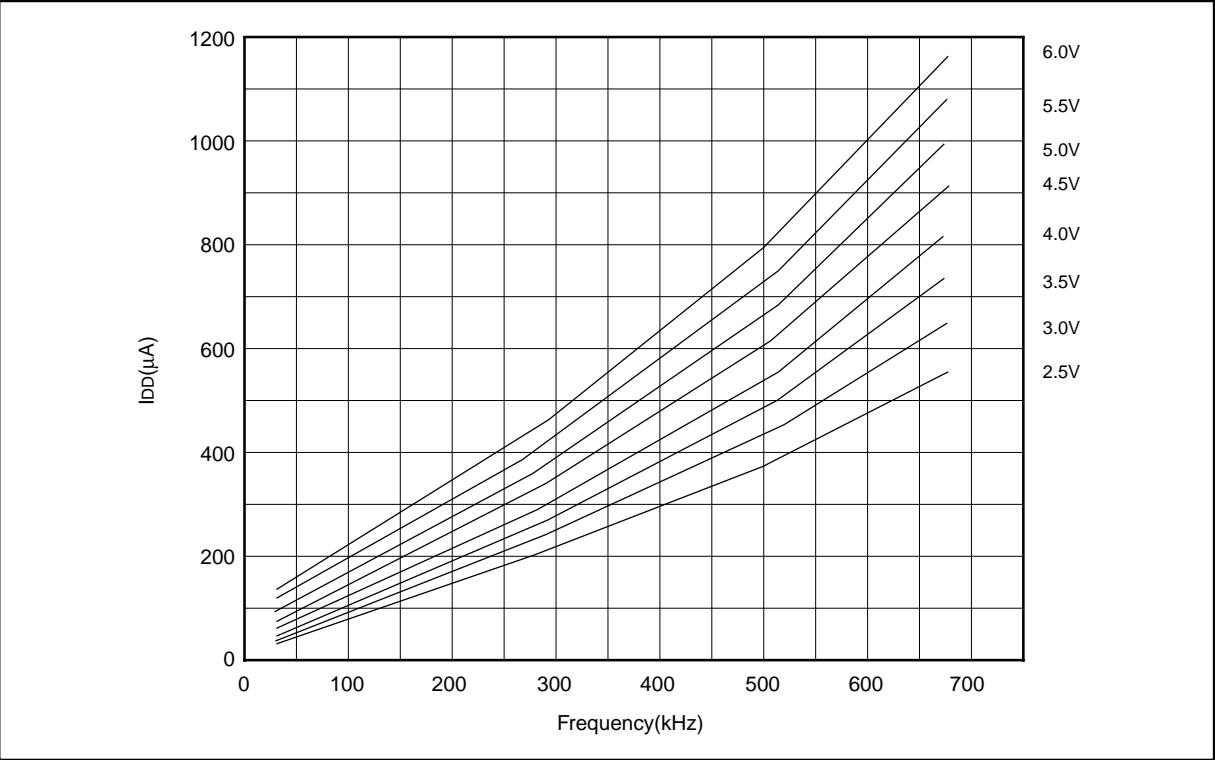


FIGURE 12-17: MAXIMUM I_{DD} vs. FREQUENCY (RC MODE @ 300 pF, -40°C TO 85°C)

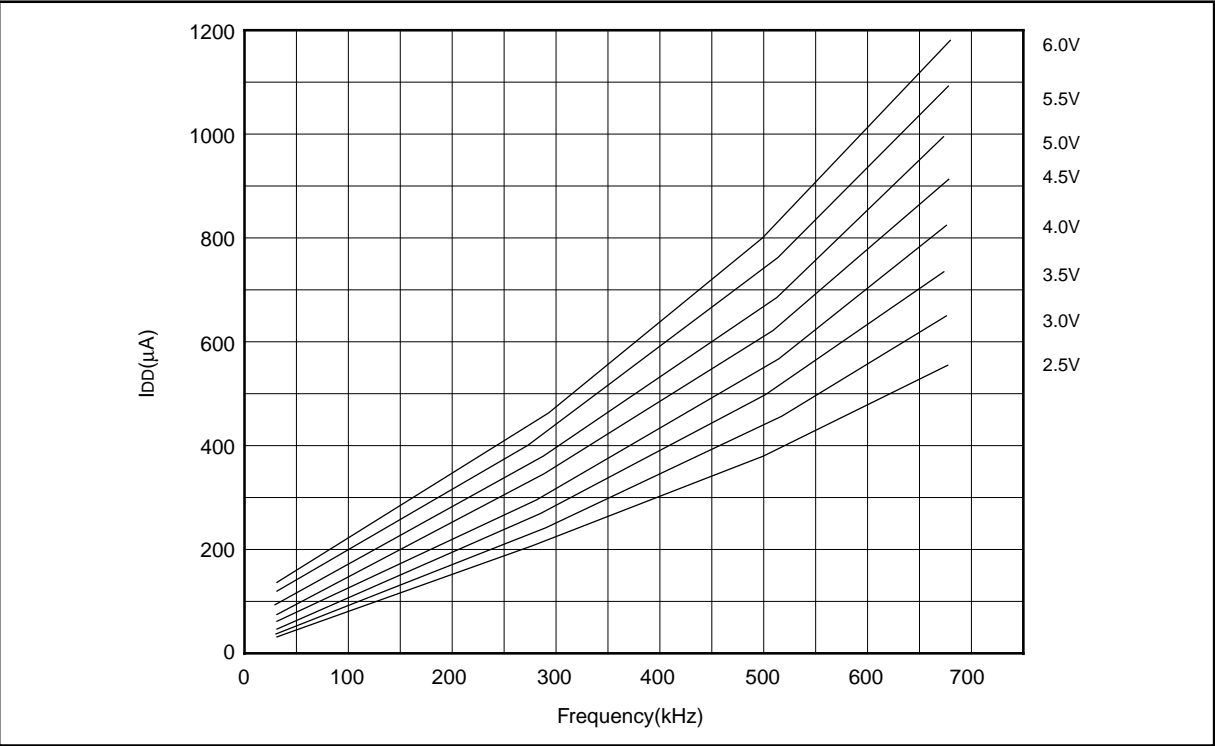


FIGURE 12-18: TYPICAL I_{DD} vs. CAPACITANCE @ 500 kHz (RC MODE)

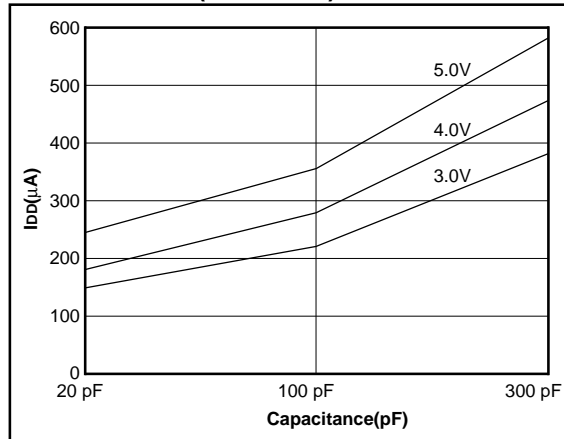


TABLE 12-1: RC OSCILLATOR FREQUENCIES

| Cext | Rext | Average | |
|--------|------|-----------------|--------|
| | | Fosc @ 5V, 25°C | |
| 22 pF | 5k | 4.12 MHz | ± 1.4% |
| | 10k | 2.35 MHz | ± 1.4% |
| | 100k | 268 kHz | ± 1.1% |
| 100 pF | 3.3k | 1.80 MHz | ± 1.0% |
| | 5k | 1.27 MHz | ± 1.0% |
| | 10k | 688 kHz | ± 1.2% |
| | 100k | 77.2 kHz | ± 1.0% |
| 300 pF | 3.3k | 707 kHz | ± 1.4% |
| | 5k | 501 kHz | ± 1.2% |
| | 10k | 269 kHz | ± 1.6% |
| | 100k | 28.3 kHz | ± 1.1% |

The percentage variation indicated here is part to part variation due to normal process distribution. The variation indicated is ± 3 standard deviation from average value for $V_{DD} = 5V$.

FIGURE 12-19: TRANSCONDUCTANCE(g_m) OF HS OSCILLATOR vs. V_{DD}

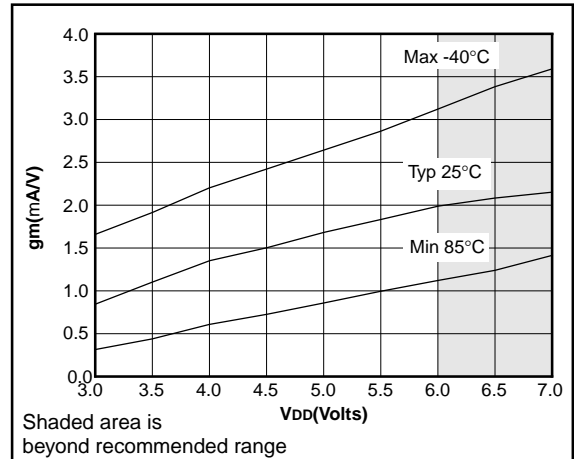


FIGURE 12-20: TRANSCONDUCTANCE(g_m) OF LP OSCILLATOR vs. V_{DD}

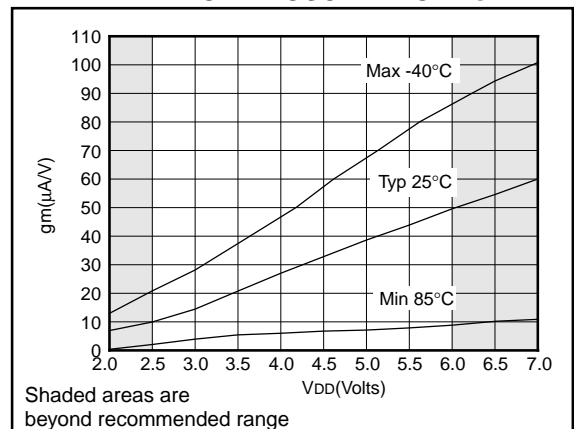
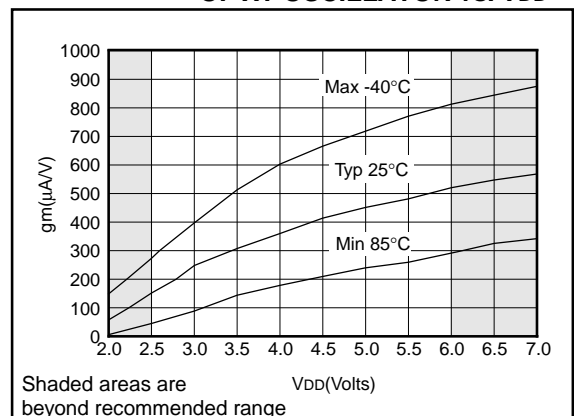


FIGURE 12-21: TRANSCONDUCTANCE(g_m) OF XT OSCILLATOR vs. V_{DD}



PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 12-22: TYPICAL XTAL STARTUP TIME vs. VDD (LP MODE, 25°C)

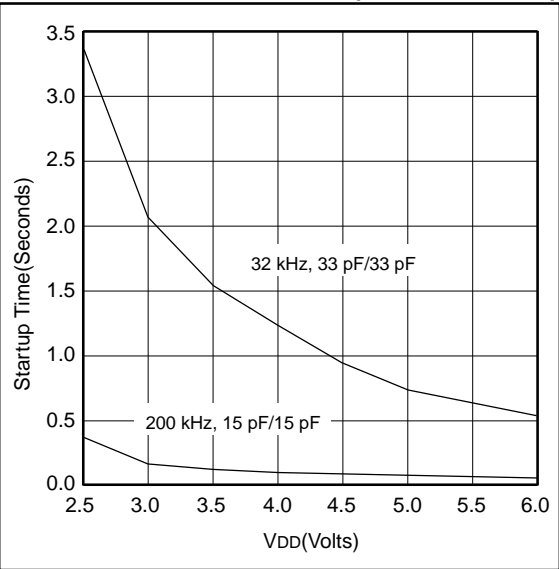


FIGURE 12-23: TYPICAL XTAL STARTUP TIME vs. VDD (HS MODE, 25°C)

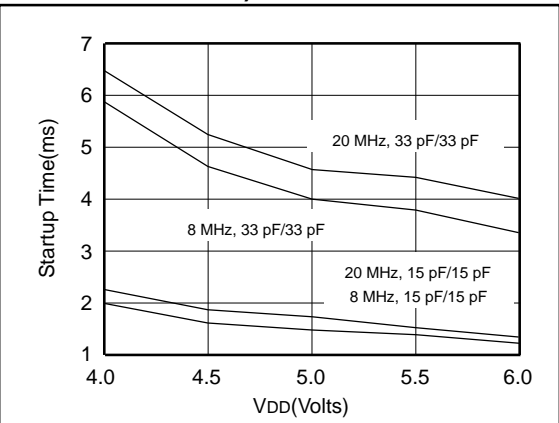


FIGURE 12-24: TYPICAL XTAL STARTUP TIME vs. VDD (XT MODE, 25°C)

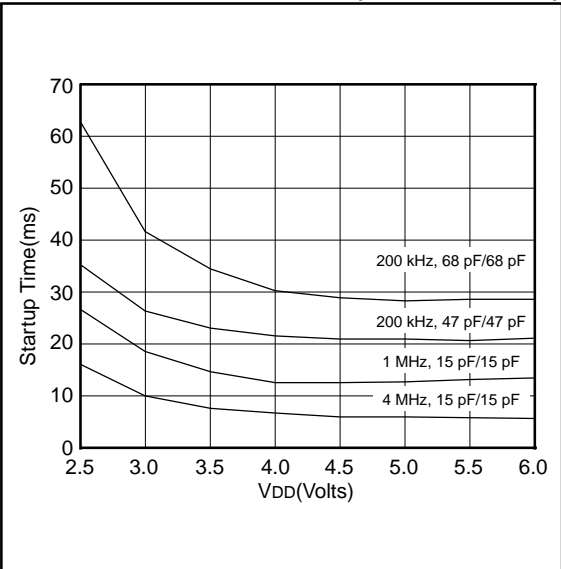


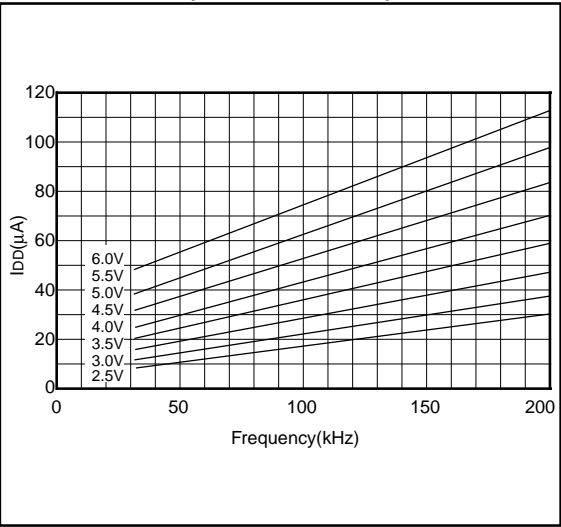
TABLE 12-2: CAPACITOR SELECTION FOR CRYSTAL OSCILLATORS

| 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 |
| Crystals Used | | | |
| 32 kHz | Epson C-001R32.768K-A | ± 20 PPM | |
| 200 kHz | STD XTL 200.000KHz | ± 20 PPM | |
| 1 MHz | ECS ECS-10-13-1 | ± 50 PPM | |
| 4 MHz | ECS ECS-40-20-1 | ± 50 PPM | |
| 8 MHz | EPSON CA-301 8.000M-C | ± 30 PPM | |
| 20 MHz | EPSON CA-301 20.000M-C | ± 30 PPM | |

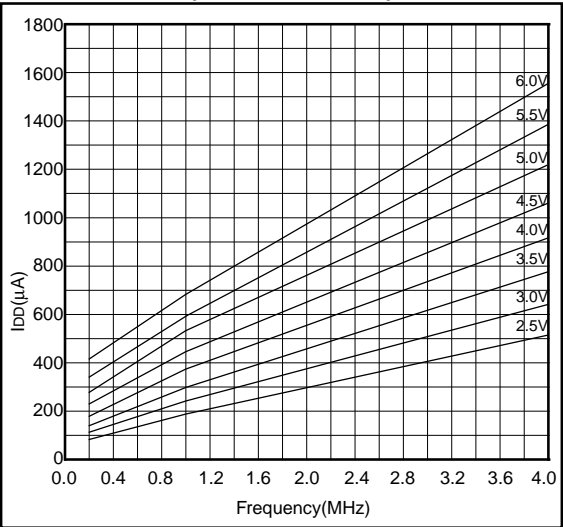
PIC16C71X

Applicable Devices 710 71 711 715

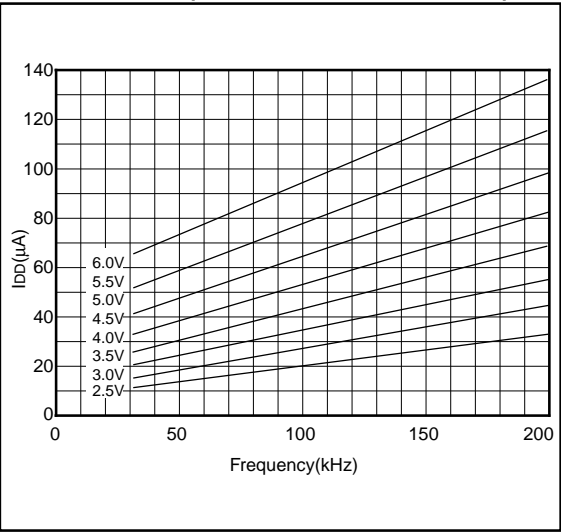
**FIGURE 12-25: TYPICAL I_{DD} vs. FREQUENCY
(LP MODE, 25°C)**



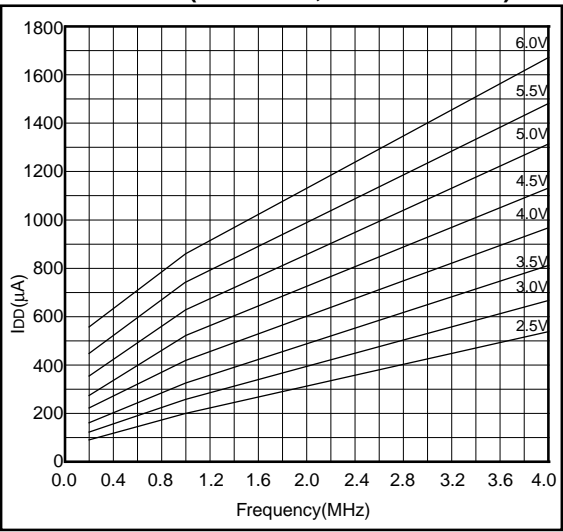
**FIGURE 12-27: TYPICAL I_{DD} vs. FREQUENCY
(XT MODE, 25°C)**



**FIGURE 12-26: MAXIMUM I_{DD} vs.
FREQUENCY
(LP MODE, 85°C TO -40°C)**



**FIGURE 12-28: MAXIMUM I_{DD} vs.
FREQUENCY
(XT MODE, -40°C TO 85°C)**



PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 12-29: TYPICAL I_{DD} vs. FREQUENCY
(HS MODE, 25°C)

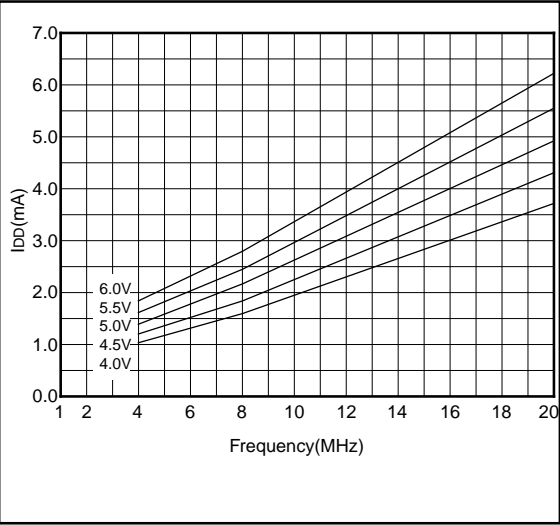
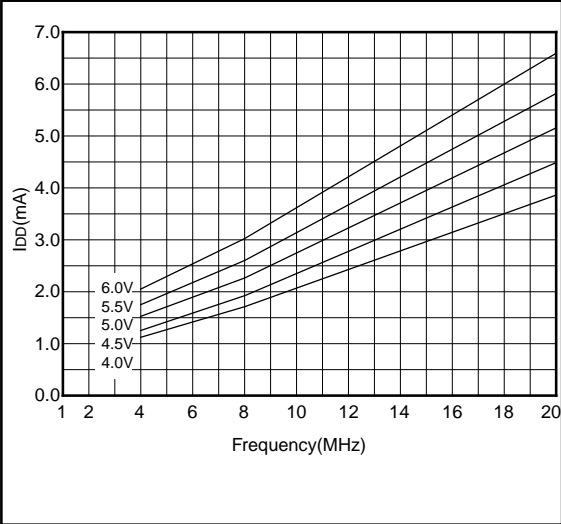


FIGURE 12-30: MAXIMUM I_{DD} vs. FREQUENCY
(HS MODE, -40°C TO 85°C)



13.0 ELECTRICAL CHARACTERISTICS FOR PIC16C715

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 and MCLR)..... | -0.3V to (VDD + 0.3V) |
| Voltage on VDD with respect to VSS | 0 to +7.5V |
| Voltage on MCLR with respect to VSS..... | 0 to +14V |
| Voltage on RA4 with respect to VSS | 0 to +14V |
| 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 | 200 mA |
| Maximum current sourced by PORTA | 200 mA |
| Maximum current sunk by PORTB..... | 200 mA |
| Maximum current sourced by PORTB..... | 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.

PIC16C71X

Applicable Devices 710 71 711 715

TABLE 13-1: CROSS REFERENCE OF DEVICE SPECS FOR OSCILLATOR CONFIGURATIONS AND FREQUENCIES OF OPERATION (COMMERCIAL DEVICES)

| OSC | PIC16C715-04 | PIC16C715-10 | PIC16C715-20 | PIC16LC715-04 | PIC16C715/JW |
|-----|--|---|---|--|--|
| RC | VDD: 4.0V to 5.5V IDD: 5 mA max. at 5.5V IPD: 21 µA max. at 4V Freq: 4 MHz max. | VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max. | VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max. | VDD: 2.5V to 5.5V IDD: 2.0 mA typ. at 3.0V IPD: 0.9 µA typ. at 3V Freq: 4 MHz max. | VDD: 4.0V to 5.5V IDD: 5 mA max. at 5.5V IPD: 21 µA max. at 4V Freq: 4 MHz max. |
| XT | VDD: 4.0V to 5.5V IDD: 5 mA max. at 5.5V IPD: 21 µA max. at 4V Freq: 4 MHz max. | VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max. | VDD: 4.5V to 5.5V IDD: 2.7 mA typ. at 5.5V IPD: 1.5 µA typ. at 4V Freq: 4 MHz max. | VDD: 2.5V to 5.5V IDD: 2.0 mA typ. at 3.0V IPD: 0.9 µA typ. at 3V Freq: 4 MHz max. | VDD: 4.0V to 5.5V IDD: 5 mA max. at 5.5V IPD: 21 µA max. at 4V Freq: 4 MHz max. |
| HS | VDD: 4.5V to 5.5V IDD: 13.5 mA typ. at 5.5V IPD: 1.5 µA typ. at 4.5V Freq: 4 MHz max. | VDD: 4.5V to 5.5V IDD: 30 mA max. at 5.5V IPD: 1.5 µA typ. at 4.5V Freq: 10 MHz max. | VDD: 4.5V to 5.5V IDD: 30 mA max. at 5.5V IPD: 1.5 µA typ. at 4.5V Freq: 20 MHz max. | Do not use in HS mode | VDD: 4.5V to 5.5V IDD: 30 mA max. at 5.5V IPD: 1.5 µA typ. at 4.5V Freq: 10 MHz max. |
| LP | VDD: 4.0V to 5.5V IDD: 52.5 µA typ. at 32 kHz, 4.0V IPD: 0.9 µA typ. at 4.0V Freq: 200 kHz max. | Do not use in LP mode | Do not use in LP mode | VDD: 2.5V to 5.5V IDD: 48 µA max. at 32 kHz, 3.0V IPD: 5.0 µA max. at 3.0V Freq: 200 kHz max. | VDD: 2.5V to 5.5V IDD: 48 µA max. at 32 kHz, 3.0V IPD: 5.0 µA max. at 3.0V Freq: 200 kHz max. |

The shaded sections indicate oscillator selections which are tested for functionality, but not for MIN/MAX specifications. It is recommended that the user select the device type that ensures the specifications required.

PIC16C71X

Applicable Devices 710 71 711 715

13.1 DC Characteristics: PIC16C715-04 (Commercial, Industrial, Extended) PIC16C715-10 (Commercial, Industrial, Extended) PIC16C715-20 (Commercial, Industrial, Extended))

| Standard Operating Conditions (unless otherwise stated) | | | | | | | |
|--|--|-------|------------|---------------------------|----------------------|----------------------|---|
| DC CHARACTERISTICS | | | | | | | |
| Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended) | | | | | | | |
| Param. No. | Characteristic | Sym | Min | Typ† | Max | Units | Conditions |
| D001 D001A | Supply Voltage | VDD | 4.0 4.5 | - - | 5.5 5.5 | V V | XT, RC and LP osc configuration HS osc configuration |
| D002* | RAM Data Retention Voltage (Note 1) | VDR | - | 1.5 | - | V | Device in SLEEP mode |
| D003 | VDD start voltage to ensure internal Power-on Reset signal | VPOR | - | VSS | - | V | See section on Power-on Reset for details |
| D004* | VDD rise rate to ensure internal Power-on Reset signal | SVDD | 0.05 | - | - | V/ms | See section on Power-on Reset for details |
| D005 | Brown-out Reset Voltage | BVDD | 3.7 | 4.0 | 4.3 | V | BODEN configuration bit is enabled |
| D010 | Supply Current (Note 2) | IDD | - | 2.7 | 5 | mA | XT, RC osc configuration (PIC16C715-04) FOSC = 4 MHz, VDD = 5.5V (Note 4) |
| D013 | | | | 13.5 | 30 | mA | HS osc configuration (PIC16C715-20) FOSC = 20 MHz, VDD = 5.5V |
| D015 | | | | 300* | 500 | μA | BOR enabled VDD = 5.0V |
| D020 D021 D021A D021B | Power-down Current (Note 3) | IPD | - | 10.5 1.5 1.5 1.5 | 42 21 24 30 | μA μA μA μA | VDD = 4.0V, WDT enabled, -40°C to +85°C VDD = 4.0V, WDT disabled, -0°C to +70°C VDD = 4.0V, WDT disabled, -40°C to +85°C VDD = 4.0V, WDT disabled, -40°C to +125°C |
| D023 | Brown-out Reset Current (Note 5) | ΔIBOR | - | 300* | 500 | μA | BOR enabled VDD = 5.0V |

* 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 in SLEEP mode 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 and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula $I_r = V_{DD}/2R_{ext}$ (mA) with Rext in kOhm.

5: The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.

PIC16C71X

Applicable Devices 710 71 711 715

13.2 DC Characteristics: PIC16LC715-04 (Commercial, Industrial)

| DC CHARACTERISTICS | | Standard Operating Conditions (unless otherwise stated) | | | | | |
|--------------------|--|---|------|------|-----|-------|---|
| | | Operating temperature 0°C ≤ TA ≤ +70°C (commercial) | | | | | |
| | | -40°C ≤ TA ≤ +85°C (industrial) | | | | | |
| Param No. | Characteristic | Sym | Min | Typ† | Max | Units | Conditions |
| D001 | Supply Voltage | VDD | 2.5 | - | 5.5 | V | LP, XT, RC osc configuration (DC - 4 MHz) |
| D002* | RAM Data Retention Voltage (Note 1) | VDR | - | 1.5 | - | V | Device in SLEEP mode |
| D003 | VDD start voltage to ensure internal Power-on Reset signal | VPOR | - | VSS | - | V | See section on Power-on Reset for details |
| D004* | VDD rise rate to ensure internal Power-on Reset signal | SVDD | 0.05 | - | - | V/ms | See section on Power-on Reset for details |
| D005 | Brown-out Reset Voltage | BVDD | 3.7 | 4.0 | 4.3 | V | BODEN configuration bit is enabled |
| D010 | Supply Current (Note 2) | IDD | - | 2.0 | 3.8 | mA | XT, RC osc configuration FOSC = 4 MHz, VDD = 3.0V (Note 4) |
| D010A | | | - | 22.5 | 48 | μA | LP osc configuration FOSC = 32 kHz, VDD = 3.0V, WDT disabled |
| D015 | Brown-out Reset Current (Note 5) | ΔIBOR | - | 300* | 500 | μA | BOR enabled VDD = 5.0V |
| D020 | Power-down Current (Note 3) | IPD | - | 7.5 | 30 | μA | VDD = 3.0V, WDT enabled, -40°C to +85°C |
| D021 | | | - | 0.9 | 5 | μA | VDD = 3.0V, WDT disabled, 0°C to +70°C |
| D021A | | | - | 0.9 | 5 | μA | VDD = 3.0V, WDT disabled, -40°C to +85°C |
| D023 | Brown-out Reset Current (Note 5) | ΔIBOR | - | 300* | 500 | μA | BOR enabled VDD = 5.0V |

* 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 in SLEEP mode 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 and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula $I_r = VDD/2R_{ext}$ (mA) with Rext in kOhm.

5: The Δ current is the additional current consumed when this peripheral is enabled. This current should be added to the base IDD or IPD measurement.

PIC16C71X

Applicable Devices 710 71 711 715

13.3 DC Characteristics: PIC16C715-04 (Commercial, Industrial, Extended) PIC16C715-10 (Commercial, Industrial, Extended) PIC16C715-20 (Commercial, Industrial, Extended) PIC16LC715-04 (Commercial, Industrial))

| DC CHARACTERISTICS <div> Standard Operating Conditions (unless otherwise stated) Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended) Operating voltage VDD range as described in DC spec Section 13.1 and Section 13.2. </div> | | | | | | | |
|---|--|-------|--------|-------|--------|-------|---|
| Param No. | Characteristic | Sym | Min | Typ † | Max | Units | Conditions |
| D030 D031 D032 D033 | Input Low Voltage I/O ports with TTL buffer with Schmitt Trigger buffer MCLR, RA4/T0CKI, OSC1 (in RC mode) OSC1 (in XT, HS and LP) | VIL | VSS | - | 0.5V | V | Note1 |
| | | | VSS | - | 0.2VDD | V | |
| | | | VSS | - | 0.2VDD | V | |
| | | | VSS | - | 0.3VDD | V | |
| D040 D040A D041 D042 D042A D043 | Input High Voltage I/O ports with TTL buffer with Schmitt Trigger buffer MCLR, RA4/T0CKI RB0/INT OSC1 (XT, HS and LP) OSC1 (in RC mode) | VIH | 2.0 | - | VDD | V | 4.5 ≤ VDD ≤ 5.5V For VDD > 5.5V or VDD < 4.5V For entire VDD range Note1 |
| | | | 0.8VDD | - | VDD | V | |
| | | | 0.8VDD | - | VDD | V | |
| | | | 0.8VDD | - | VDD | V | |
| | | | 0.7VDD | - | VDD | V | |
| | | | 0.9VDD | - | VDD | V | |
| D070 | PORTB weak pull-up current | IPURB | 50 | 250 | 400 | µA | VDD = 5V, VPIN = VSS |
| D060 D061 D063 | Input Leakage Current (Notes 2, 3) I/O ports MCLR, RA4/T0CKI OSC1 | IIL | - | - | ±1 | µA | VSS ≤ VPIN ≤ VDD, Pin at hi-impedance |
| | | | - | - | ±5 | µA | VSS ≤ VPIN ≤ VDD |
| | | | - | - | ±5 | µA | VSS ≤ VPIN ≤ VDD, XT, HS and LP osc configuration |
| D080 D080A D083 D083A | Output Low Voltage I/O ports OSC2/CLKOUT (RC osc config) | VOL | - | - | 0.6 | V | IOL = 8.5 mA, VDD = 4.5V, -40°C to +85°C |
| | | | - | - | 0.6 | V | IOL = 7.0 mA, VDD = 4.5V, -40°C to +125°C |
| | | | - | - | 0.6 | V | IOL = 1.6 mA, VDD = 4.5V, -40°C to +85°C |
| | | | - | - | 0.6 | V | IOL = 1.2 mA, VDD = 4.5V, -40°C to +125°C |

† 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: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C7X be driven with external clock in RC mode.

2: The leakage current on the 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.

3: Negative current is defined as coming out of the pin.

PIC16C71X

Applicable Devices 710 71 711 715

| Standard Operating Conditions (unless otherwise stated) Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended) Operating voltage VDD range as described in DC spec Section 13.1 and Section 13.2. | | | | | | | |
|--|--|-------|-----------|-------|-----|-------|---|
| DC CHARACTERISTICS | | | | | | | |
| Param No. | Characteristic | Sym | Min | Typ † | Max | Units | Conditions |
| D090 | Output High Voltage I/O ports (Note 3) | VOH | VDD - 0.7 | - | - | V | IOH = -3.0 mA, VDD = 4.5V, -40°C to +85°C |
| D090A | | | VDD - 0.7 | - | - | V | IOH = -2.5 mA, VDD = 4.5V, -40°C to +125°C |
| D092 | | | VDD - 0.7 | - | - | V | IOH = -1.3 mA, VDD = 4.5V, -40°C to +85°C |
| D092A | | | VDD - 0.7 | - | - | V | IOH = -1.0 mA, VDD = 4.5V, -40°C to +125°C |
| D100 | Capacitive Loading Specs on Output Pins OSC2 pin | COSC2 | - | - | 15 | pF | In XT, HS and LP modes when external clock is used to drive OSC1. |
| D101 | All I/O pins and OSC2 (in RC mode) | CIO | - | - | 50 | pF | |

† 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: In RC oscillator configuration, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C7X be driven with external clock in RC mode.
- 2: The leakage current on the 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.
- 3: Negative current is defined as coming out of the pin.

PIC16C71X

| | | | | |
|---------------------------|-----|----|-----|-----|
| Applicable Devices | 710 | 71 | 711 | 715 |
|---------------------------|-----|----|-----|-----|

13.4 Timing Parameter Symbology

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

1. TppS2ppS
2. TppS

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

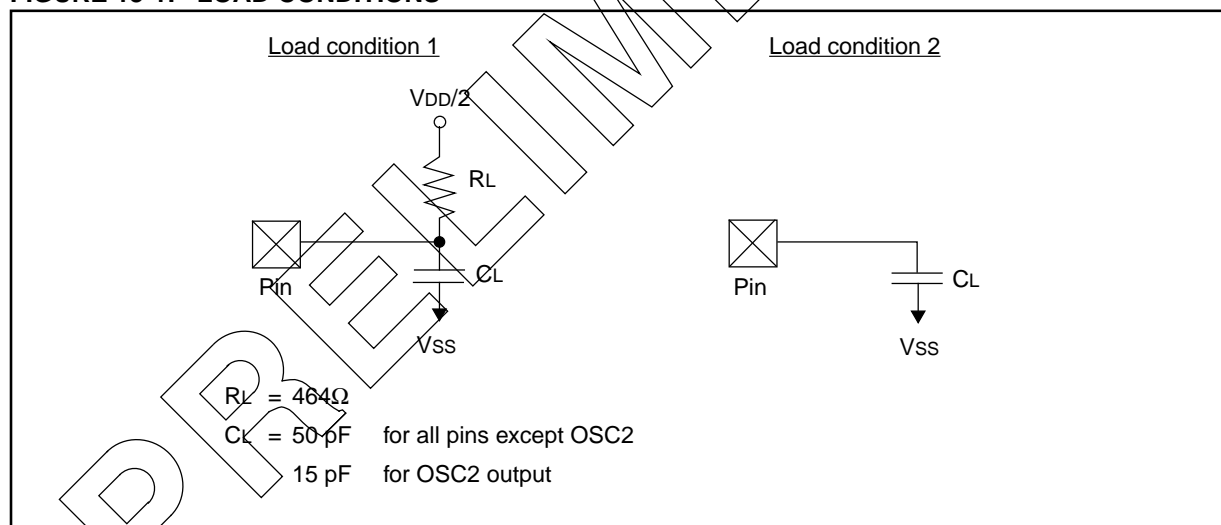
Lowercase letters (pp) and their meanings:

| | | | |
|----|-------------------|-----|------------------------------------|
| 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 | | P | | R | | V | | Z | |
|---|------------------------|---|--|---|--|---|--|---|--|
| F | Fall | | | | | | | | |
| H | High | | | | | | | | |
| I | Invalid (Hi-impedance) | | | | | | | | |
| L | Low | | | | | | | | |

FIGURE 13-1: LOAD CONDITIONS



PIC16C71X

Applicable Devices 710 71 711 715

13.5 Timing Diagrams and Specifications

FIGURE 13-2: EXTERNAL CLOCK TIMING

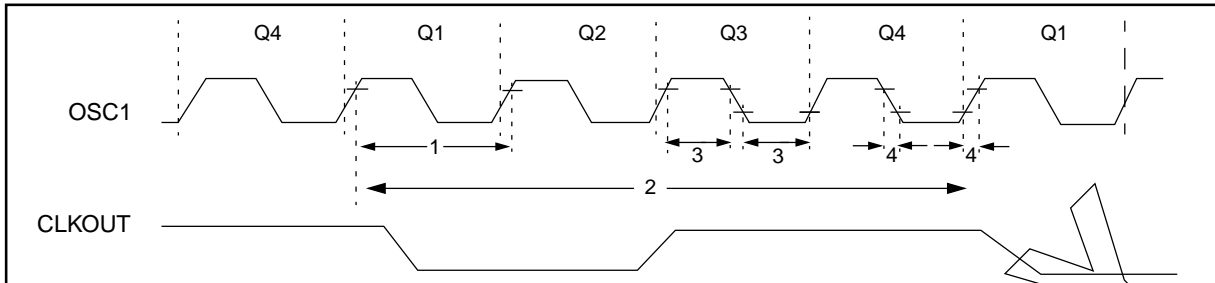


TABLE 13-2: CLOCK TIMING REQUIREMENTS

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|---------------|---|-----|------|--------|-------|----------------------------|
| | Fos | External CLKIN Frequency (Note 1) | DC | — | 4 | MHz | XT osc mode |
| | | | DC | — | 4 | MHz | HS osc mode (PIC16C715-04) |
| | | | DC | — | 20 | MHz | HS osc mode (PIC16C715-20) |
| | | | DC | — | 200 | kHz | LP osc mode |
| | | Oscillator Frequency (Note 1) | DC | — | 4 | MHz | RC osc mode |
| | | | 0.1 | — | 4 | MHz | XT osc mode |
| | | | 4 | — | 4 | MHz | HS osc mode (PIC16C715-04) |
| | | | 4 | — | 10 | MHz | HS osc mode (PIC16C715-10) |
| | | | 4 | — | 20 | MHz | HS osc mode (PIC16C715-20) |
| | | | 5 | — | 200 | kHz | LP osc mode |
| 1 | Tosc | External CLKIN Period (Note 1) | 250 | — | — | ns | XT osc mode |
| | | | 250 | — | — | ns | HS osc mode (PIC16C715-04) |
| | | | 100 | — | — | ns | HS osc mode (PIC16C715-10) |
| | | | 50 | — | — | ns | HS osc mode (PIC16C715-20) |
| | | | 5 | — | — | μs | LP osc mode |
| | | Oscillator Period (Note 1) | 250 | — | — | ns | RC osc mode |
| | | | 250 | — | 10,000 | ns | XT osc mode |
| | | | 250 | — | 250 | ns | HS osc mode (PIC16C715-04) |
| | | | 100 | — | 250 | ns | HS osc mode (PIC16C715-10) |
| | | | 50 | — | 250 | ns | HS osc mode (PIC16C715-20) |
| 2 | Tcy | Instruction Cycle Time (Note 1) | 200 | — | DC | ns | Tcy = 4/Fosc |
| 3 | TosL, TosH | External Clock in (OSC1) High or Low Time | 50 | — | — | ns | XT oscillator |
| | | | 2.5 | — | — | μs | LP oscillator |
| | | | 10 | — | — | ns | HS oscillator |
| 4 | TosR, TosF | External Clock in (OSC1) Rise or Fall Time | — | — | 25 | ns | XT oscillator |
| | | | — | — | 50 | ns | LP oscillator |
| | | | — | — | 15 | ns | HS oscillator |

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

Note 1: Instruction cycle period (Tcy) equals four times the input oscillator time-base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min." values with an external clock applied to the OSC1/CLKIN pin. When an external clock input is used, the "Max." cycle time limit is "DC" (no clock) for all devices. OSC2 is disconnected (has no loading) for the PIC16C715.

FIGURE 13-3: CLKOUT AND I/O TIMING

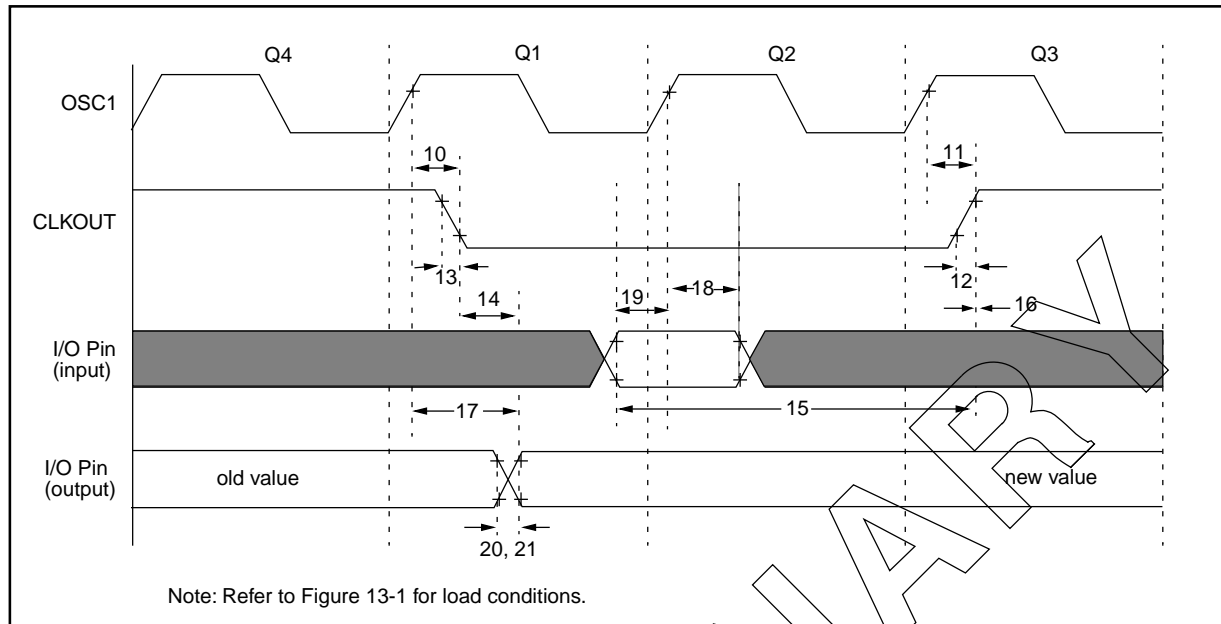


TABLE 13-3: CLKOUT AND I/O TIMING REQUIREMENTS

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|----------|---|--------------|------|-------------|-------|------------|
| 10* | TosH2ckL | OSC1↑ to CLKOUT↓ | — | 15 | 30 | ns | Note 1 |
| 11* | TosH2ckH | OSC1↑ to CLKOUT↑ | — | 15 | 30 | ns | Note 1 |
| 12* | TckR | CLKOUT rise time | — | 5 | 15 | ns | Note 1 |
| 13* | TckF | CLKOUT fall time | — | 5 | 15 | ns | Note 1 |
| 14* | TckL2ioV | CLKOUT ↓ to Port out valid | — | — | 0.5TCY + 20 | ns | Note 1 |
| 15* | TioV2ckH | Port in valid before CLKOUT ↑ | 0.25TCY + 25 | — | — | ns | Note 1 |
| 16* | TckH2ioL | Port in hold after CLKOUT ↑ | 0 | — | — | ns | Note 1 |
| 17* | TosH2ioV | OSC1↑ (Q1 cycle) to Port out valid | — | — | 80 - 100 | ns | |
| 18* | TosH2ioL | OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time) | TBD | — | — | ns | |
| 19* | TioV2osH | Port input valid to OSC1↑ (I/O in setup time) | TBD | — | — | ns | |
| 20* | TioR | Port output rise time | — | 10 | 25 | ns | |
| 21* | TioF | Port output fall time | — | 10 | 25 | ns | |
| | | PIC16C715 | — | — | 60 | ns | |
| | | PIC16LC715 | — | — | 60 | ns | |
| 22††* | Tinp | INT pin high or low time | 20 | — | — | ns | |
| 23††* | Trsp | RB7:RB4 change INT high or low time | 20 | — | — | ns | |

* These parameters are characterized but not tested.

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

†† These parameters are asynchronous events not related to any internal clock edges.

Note 1: Measurements are taken in RC Mode where CLKOUT output is 4 x TOSC.

PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 13-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, AND POWER-UP TIMER TIMING

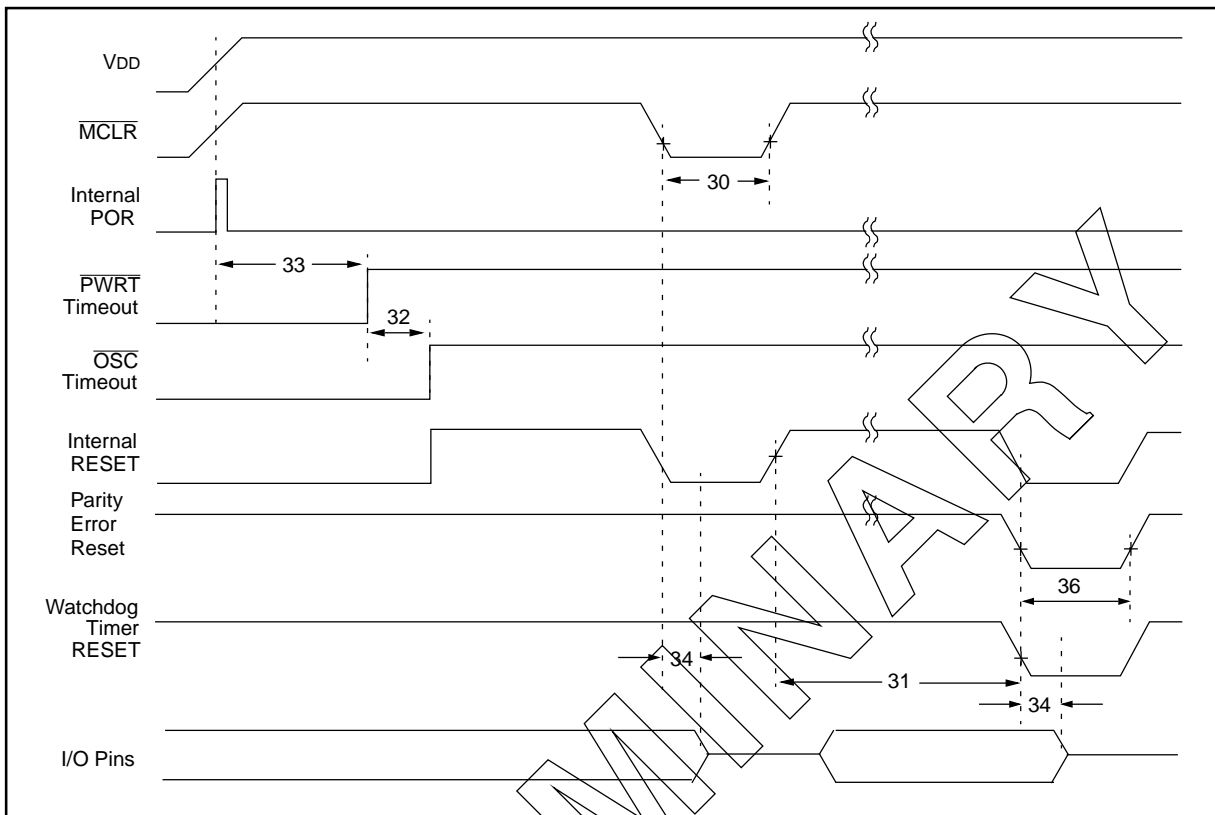


FIGURE 13-5: BROWN-OUT RESET TIMING

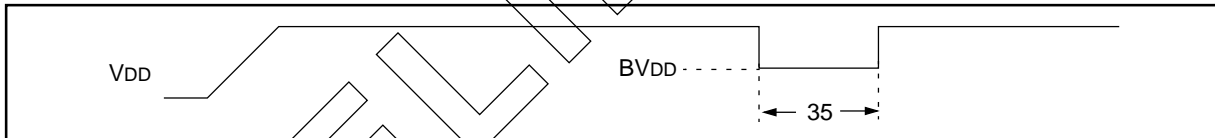


TABLE 13-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER, POWER-UP TIMER, AND BROWN-OUT RESET REQUIREMENTS

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|-------|--|-----|----------|-----|-------|---------------------------|
| 30 | Tmcl | MCLR Pulse Width (low) | 2 | — | — | μs | VDD = 5V, -40°C to +125°C |
| 31* | Twdt | Watchdog Timer Time-out Period (No Prescaler) | 7 | 18 | 33 | ms | VDD = 5V, -40°C to +125°C |
| 32 | Tost | Oscillation Start-up Timer Period | — | 1024Tosc | — | — | Tosc = OSC1 period |
| 33* | Tpwrt | Power up Timer Period | 28 | 72 | 132 | ms | VDD = 5V, -40°C to +125°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 ≤ BVDD (D005) |
| 36 | TPER | Parity Error Reset | — | TBD | — | μs | |

* 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 13-6: TIMER0 CLOCK TIMINGS

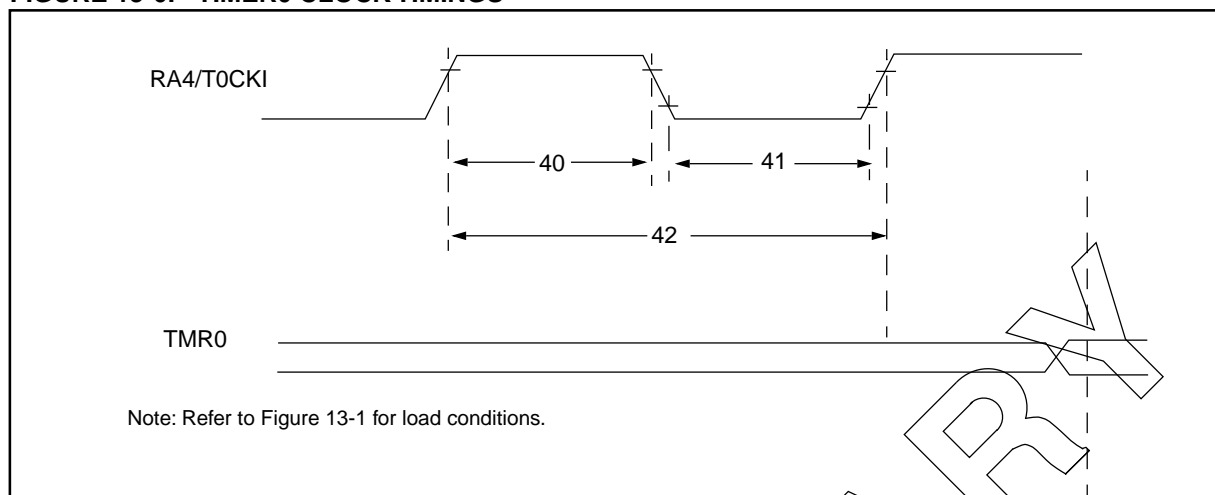


TABLE 13-5: TIMER0 CLOCK REQUIREMENTS

| Param No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|-----------|-----------|---|---|--------------------|------------|-------|---------------------------------------|
| 40 | Tt0H | T0CKI High Pulse Width | No Prescaler | $0.5T_{CY} + 20^*$ | — | ns | |
| | | | With Prescaler | 10* | — | ns | |
| 41 | Tt0L | T0CKI Low Pulse Width | No Prescaler | $0.5T_{CY} + 20^*$ | — | ns | |
| | | | With Prescaler | 10* | — | ns | |
| 42 | Tt0P | T0CKI Period | Greater of: $20\mu s$ or $\frac{T_{CY} + 40^*}{N}$ | | — | ns | N = prescale value (1, 2, 4,..., 256) |
| 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.

PIC16C71X

Applicable Devices 710 71 711 715

TABLE 13-6: A/D CONVERTER CHARACTERISTICS:
PIC16C715-04 (COMMERCIAL, INDUSTRIAL, EXTENDED)
PIC16C715-10 (COMMERCIAL, INDUSTRIAL, EXTENDED)
PIC16C715-20 (COMMERCIAL, INDUSTRIAL, EXTENDED)

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|------|--|----------------|------------|-----------------------|---------------|--|
| | NR | Resolution | — | — | 8-bits | — | $V_{REF} = V_{DD}, V_{SS} \leq AIN \leq V_{REF}$ |
| | NINT | Integral error | — | — | less than ± 1 LSb | — | $V_{REF} = V_{DD}, V_{SS} \leq AIN \leq V_{REF}$ |
| | NDIF | Differential error | — | — | less than ± 1 LSb | — | $V_{REF} = V_{DD}, V_{SS} \leq AIN \leq V_{REF}$ |
| | NFS | Full scale error | — | — | less than ± 1 LSb | — | $V_{REF} = V_{DD}, V_{SS} \leq AIN \leq V_{REF}$ |
| | NOFF | Offset error | — | — | less than ± 1 LSb | — | $V_{REF} = V_{DD}, V_{SS} \leq AIN \leq V_{REF}$ |
| | — | Monotonicity | — | guaranteed | — | — | $V_{SS} \leq AIN \leq V_{REF}$ |
| | VREF | Reference voltage | 2.5V | — | $V_{DD} + 0.3$ | V | |
| | VAIN | Analog input voltage | $V_{SS} - 0.3$ | — | $V_{REF} + 0.3$ | V | |
| | ZAIN | Recommended impedance of analog voltage source | — | — | 10.0 | k Ω | |
| | IAD | A/D conversion current (V_{DD}) | — | 180 | — | μ A | Average current consumption when A/D is on. (Note 1) |
| | IREF | VREF input current (Note 2) | — | — | 1 10 | mA μ A | During sampling All other times |

* 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 current is from RA3 pin or VDD pin, whichever is selected as reference input.

**TABLE 13-7: A/D CONVERTER CHARACTERISTICS:
PIC16LC715-04 (COMMERCIAL, INDUSTRIAL)**

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|------|--|----------------|------------|-----------------------|---------------|--|
| | NR | Resolution | — | — | 8-bits | — | $V_{REF} = V_{DD}$, $V_{SS} \leq AIN \leq V_{REF}$ |
| | NINT | Integral error | — | — | less than ± 1 LSb | — | $V_{REF} = V_{DD}$, $V_{SS} \leq AIN \leq V_{REF}$ |
| | NDIF | Differential error | — | — | less than ± 1 LSb | — | $V_{REF} = V_{DD}$, $V_{SS} \leq AIN \leq V_{REF}$ |
| | NFS | Full scale error | — | — | less than ± 1 LSb | — | $V_{REF} = V_{DD}$, $V_{SS} \leq AIN \leq V_{REF}$ |
| | NOFF | Offset error | — | — | less than ± 1 LSb | — | $V_{REF} = V_{DD}$, $V_{SS} \leq AIN \leq V_{REF}$ |
| | — | Monotonicity | — | guaranteed | — | — | $V_{SS} \leq AIN \leq V_{REF}$ |
| | VREF | Reference voltage | 2.5V | — | $V_{DD} + 0.3$ | V | |
| | VAIN | Analog input voltage | $V_{SS} - 0.3$ | — | $V_{REF} + 0.3$ | V | |
| | ZAIN | Recommended impedance of analog voltage source | — | — | 10.0 | k Ω | |
| | IAD | A/D conversion current (V_{DD}) | — | 90 | — | μ A | Average current consumption when A/D is on. (Note 1) |
| | IREF | VREF input current (Note 2) | — | — | 1 10 | mA μ A | During sampling All other times |

* 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 current is from RA3 pin or VDD pin, whichever is selected as reference input.

PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 13-7: A/D CONVERSION TIMING

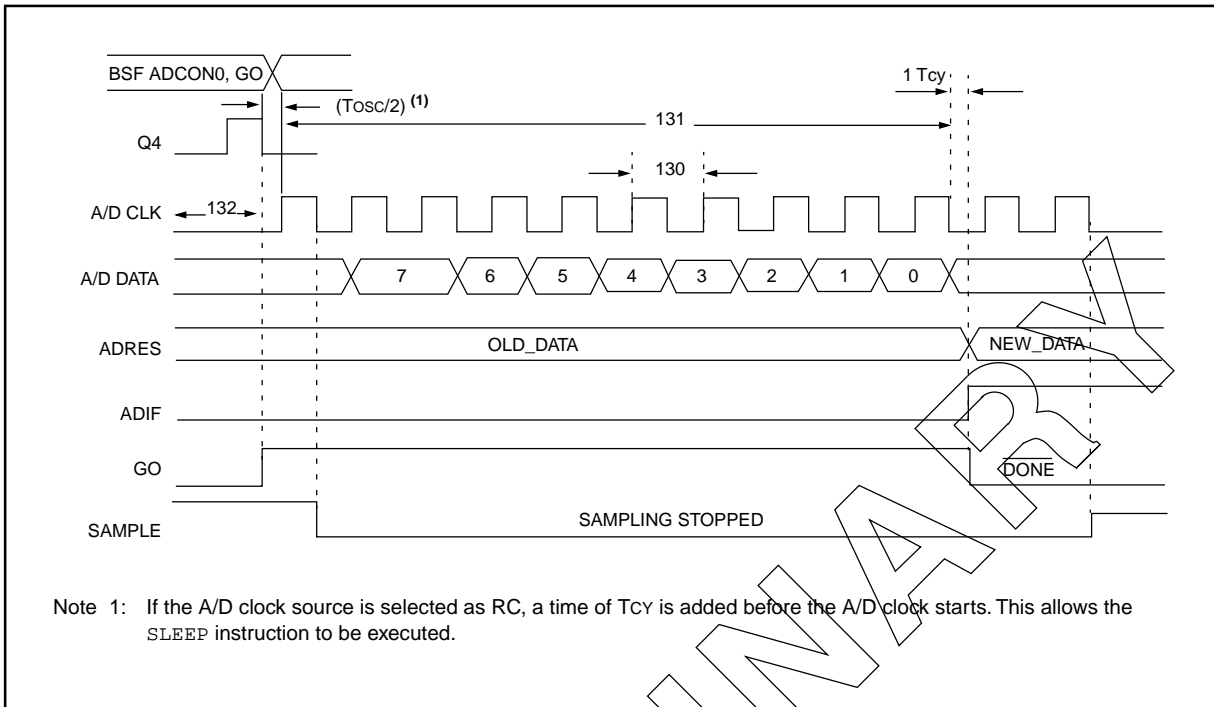


TABLE 13-8: A/D CONVERSION REQUIREMENTS

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|------|--|------------|------------|------------|----------|---|
| 130 | TAD | A/D clock period | 1.6 2.0 | — | — | μs μs | VREF ≥ 3.0V VREF full range |
| 130 | TAD | A/D Internal RC Oscillator source | 3.0 2.0 | 6.0 4.0 | 9.0 6.0 | μs μs | ADCS1:ADCS0 = 11 (RC oscillator source) PIC16LC715, VDD = 3.0V PIC16C715 |
| 131 | TCNV | Conversion time (not including S/H time). Note 1 | — | 9.5TAD | — | — | |
| 132 | TACQ | Acquisition time | Note 2 | 20 | — | μs | |

* 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.

14.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES FOR PIC16C715

The graphs and tables provided in this section are for design guidance and are not tested or guaranteed.

In some graphs or tables the data presented are outside specified operating range (i.e., outside specified V_{DD} range). This is for information only and devices are guaranteed to operate properly only within the specified range.

Note: The data presented in this section is a statistical summary of data collected on units from different lots over a period of time and matrix samples. 'Typical' represents the mean of the distribution at 25°C, while 'max' or 'min' represents (mean +3 σ) and (mean -3 σ) respectively where σ is standard deviation.

FIGURE 14-1: TYPICAL I_{PD} vs. V_{DD} (WDT DISABLED, RC MODE)

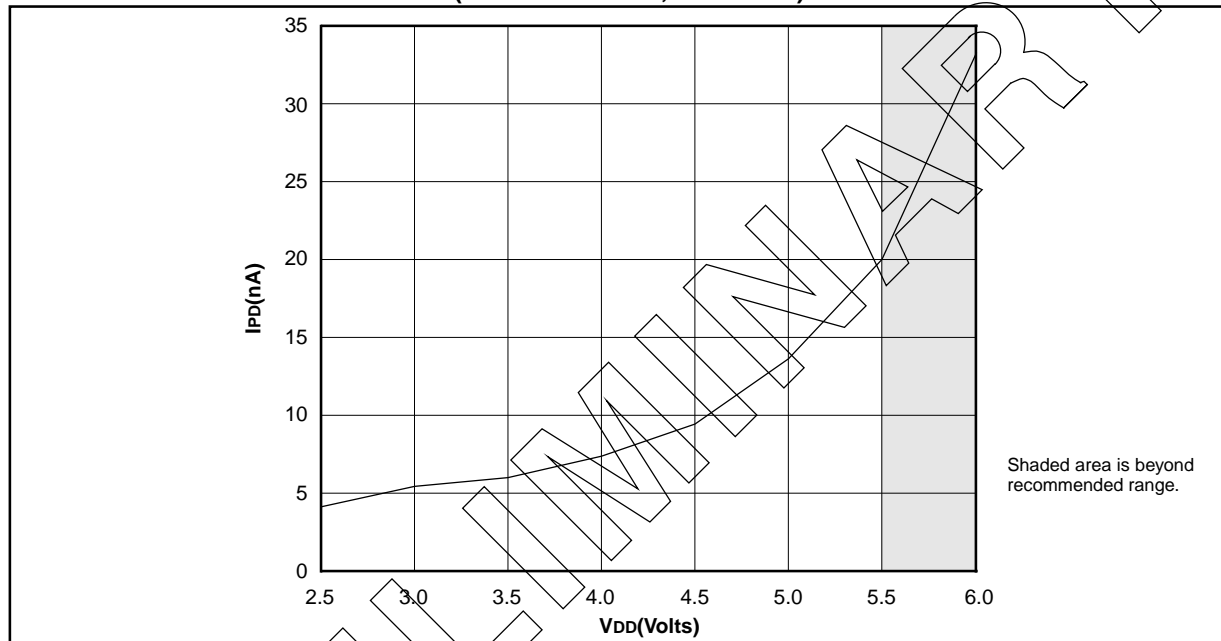
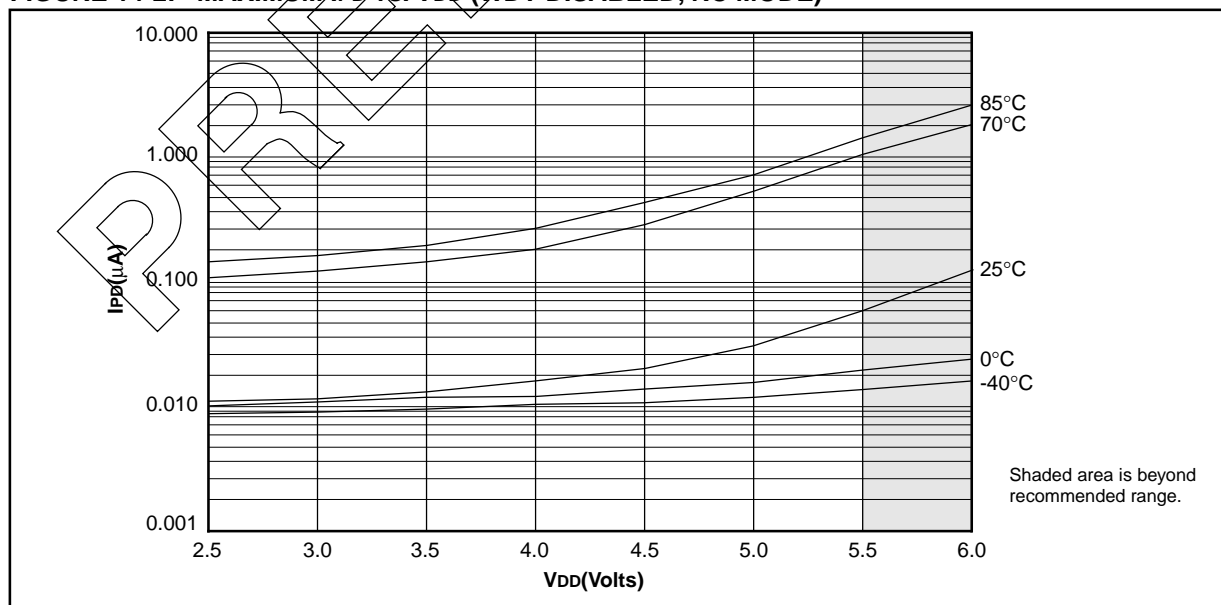


FIGURE 14-2: MAXIMUM I_{PD} vs. V_{DD} (WDT DISABLED, RC MODE)



PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 14-3: TYPICAL IPD vs. VDD @ 25°C (WDT ENABLED, RC MODE)

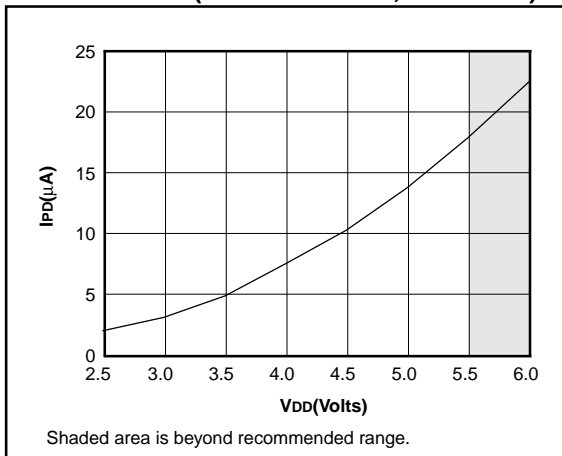


FIGURE 14-4: MAXIMUM IPD vs. VDD (WDT ENABLED, RC MODE)

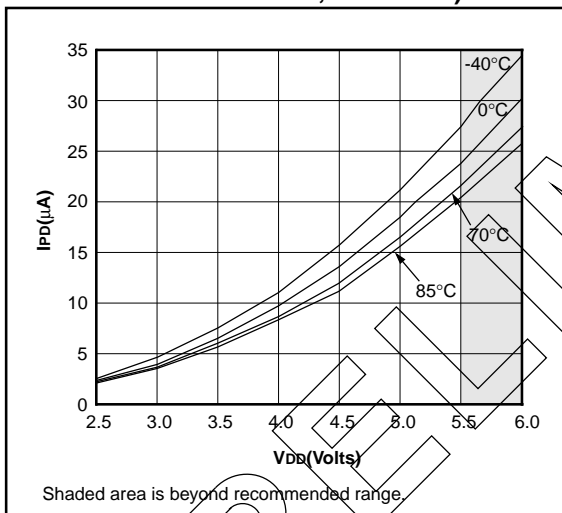


FIGURE 14-5: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

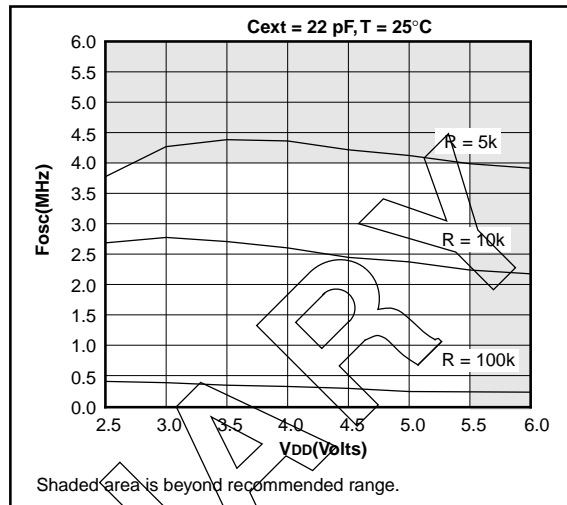


FIGURE 14-6: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

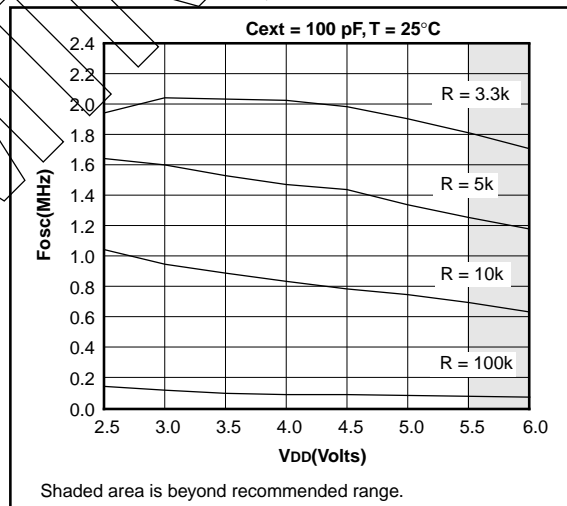


FIGURE 14-7: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

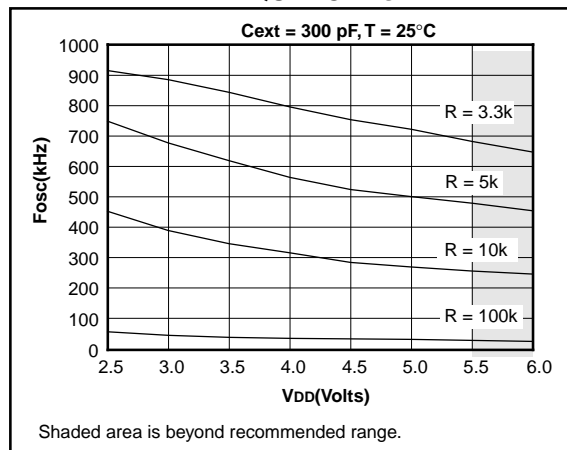


FIGURE 14-8: TYPICAL I_{PD} vs. V_{DD} BROWN-OUT DETECT ENABLED (RC MODE)

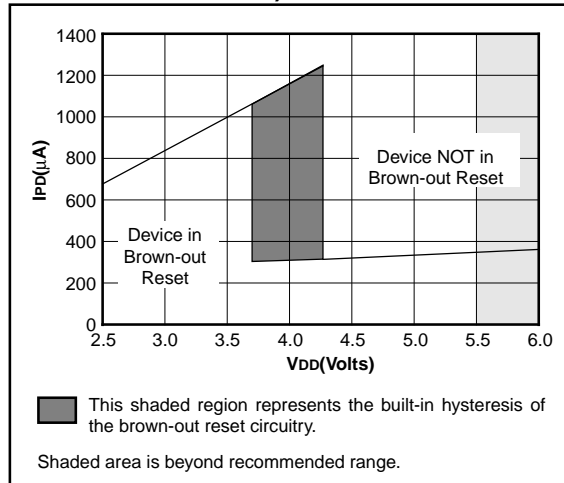


FIGURE 14-10: TYPICAL I_{PD} vs. TIMER1 ENABLED (32 kHz, RC0/RC1 = 33 pF/33 pF, RC MODE)

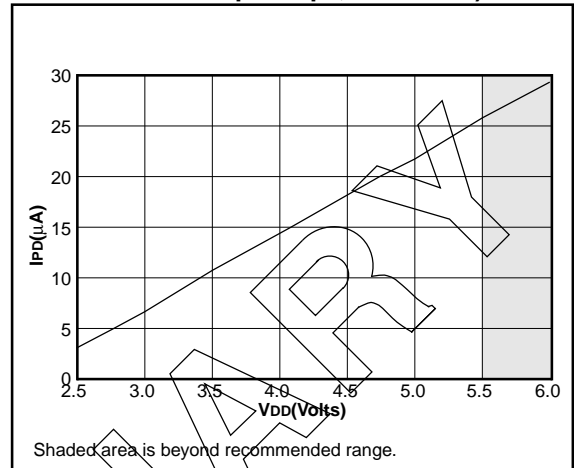


FIGURE 14-9: MAXIMUM I_{PD} vs. V_{DD} BROWN-OUT DETECT ENABLED (85°C TO -40°C, RC MODE)

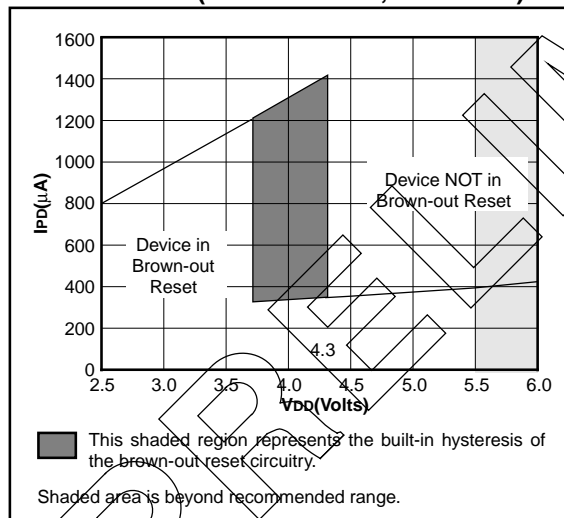
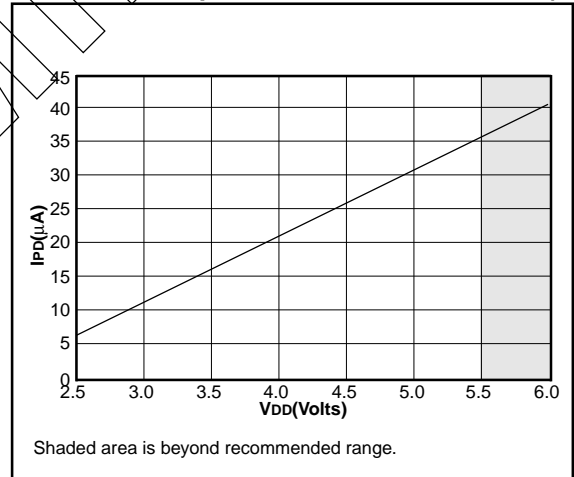


FIGURE 14-11: MAXIMUM I_{PD} vs. TIMER1 ENABLED (32 kHz, RC0/RC1 = 33 pF/33 pF, 85°C TO -40°C, RC MODE)



PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 14-12: TYPICAL I_{DD} vs. FREQUENCY (RC MODE @ 22 pF, 25°C)

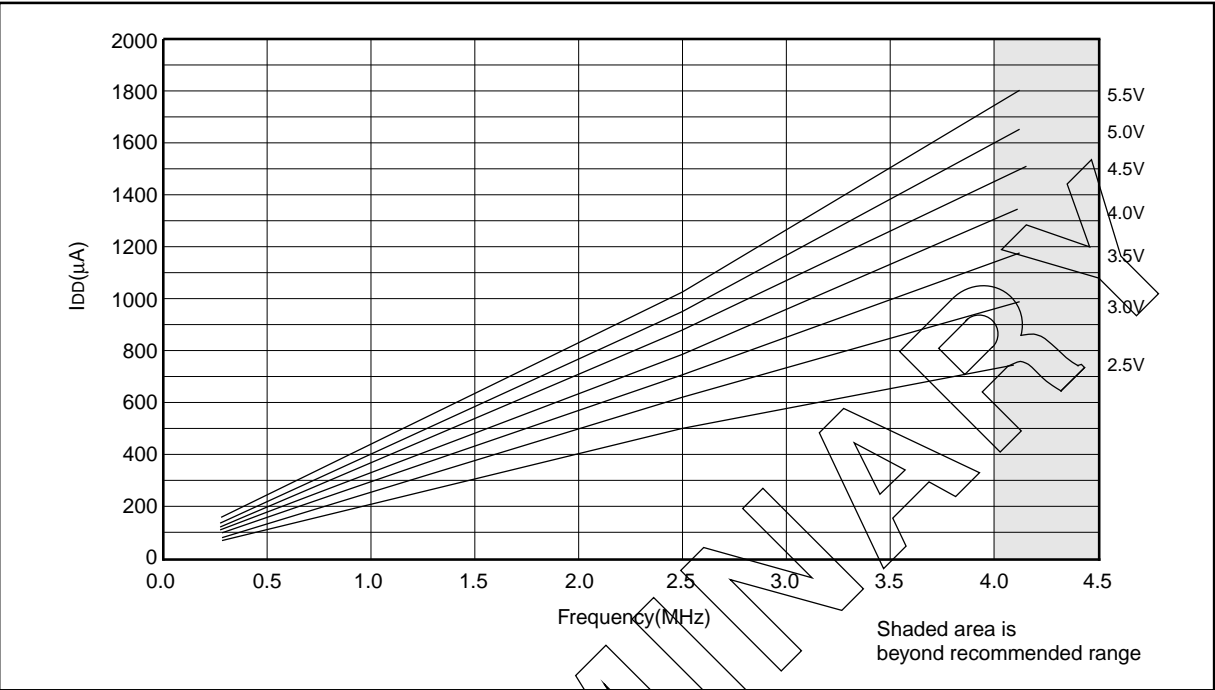
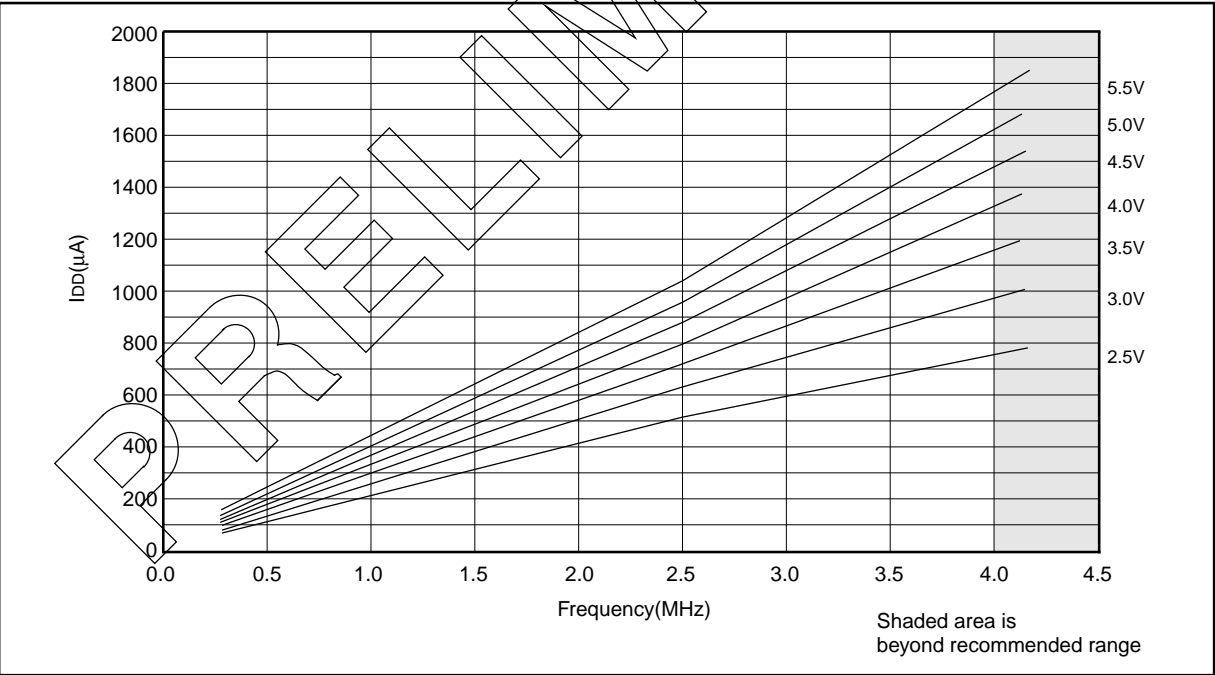


FIGURE 14-13: MAXIMUM I_{DD} vs. FREQUENCY (RC MODE @ 22 pF, -40°C TO 85°C)



PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 14-14: TYPICAL I_{DD} vs. FREQUENCY (RC MODE @ 100 pF, 25°C)

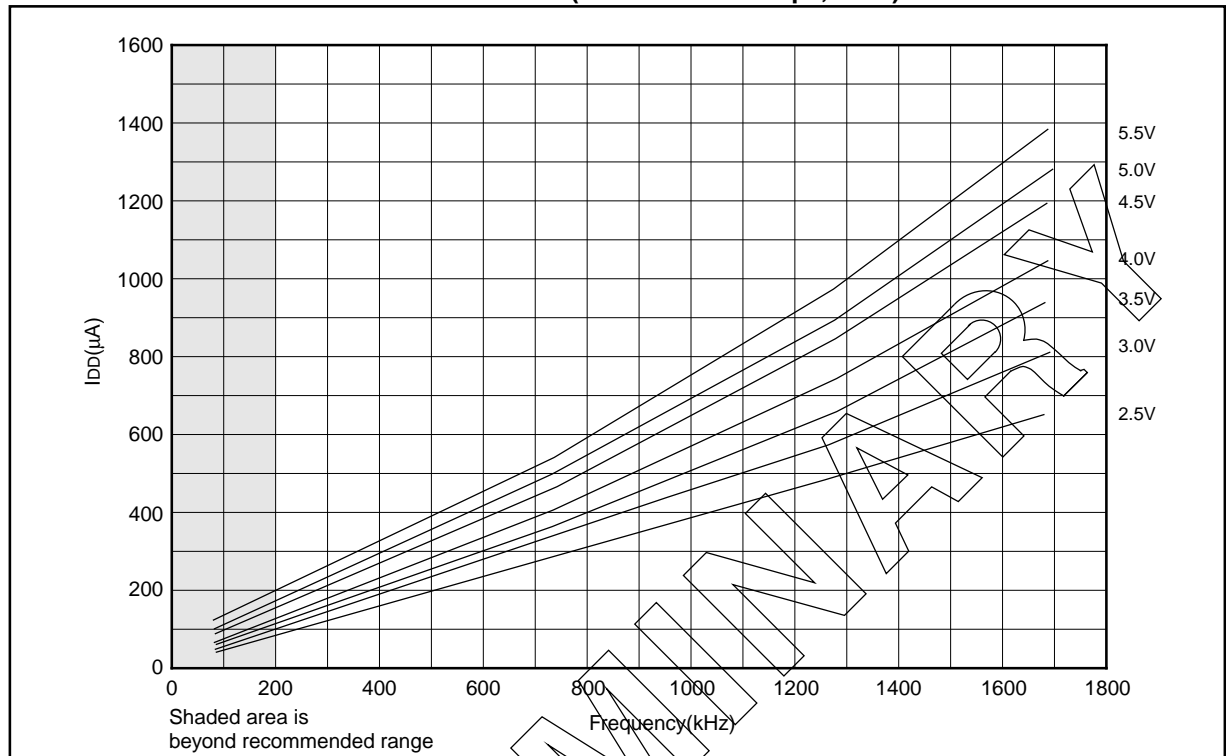
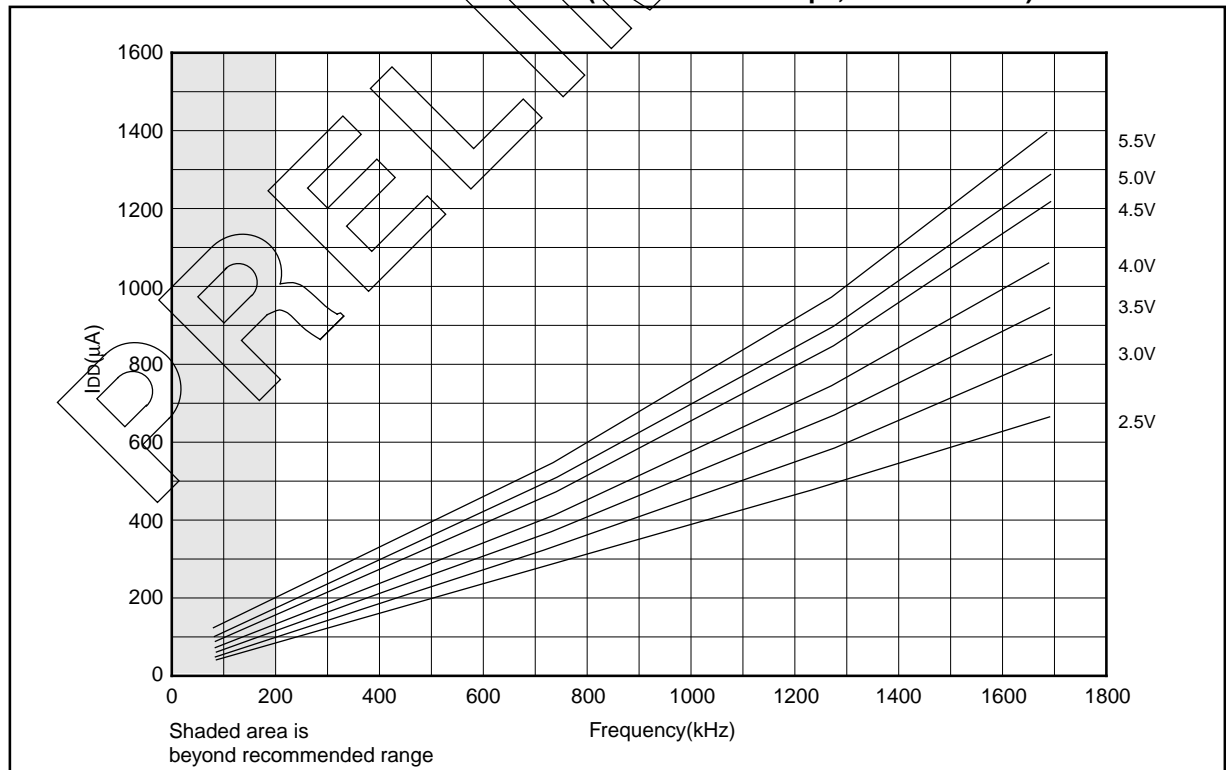


FIGURE 14-15: MAXIMUM I_{DD} vs. FREQUENCY (RC MODE @ 100 pF, -40°C TO 85°C)



PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 14-16: TYPICAL I_{DD} vs. FREQUENCY (RC MODE @ 300 pF, 25°C)

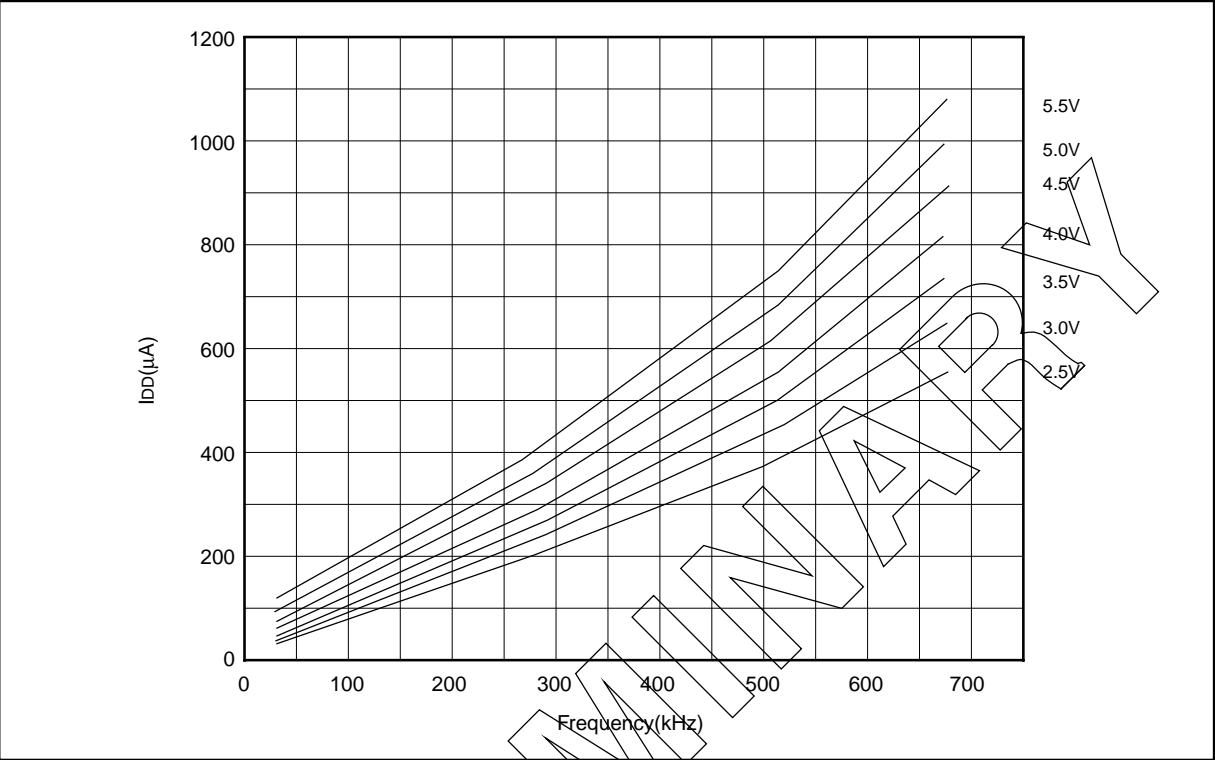


FIGURE 14-17: MAXIMUM I_{DD} vs. FREQUENCY (RC MODE @ 300 pF, -40°C TO 85°C)

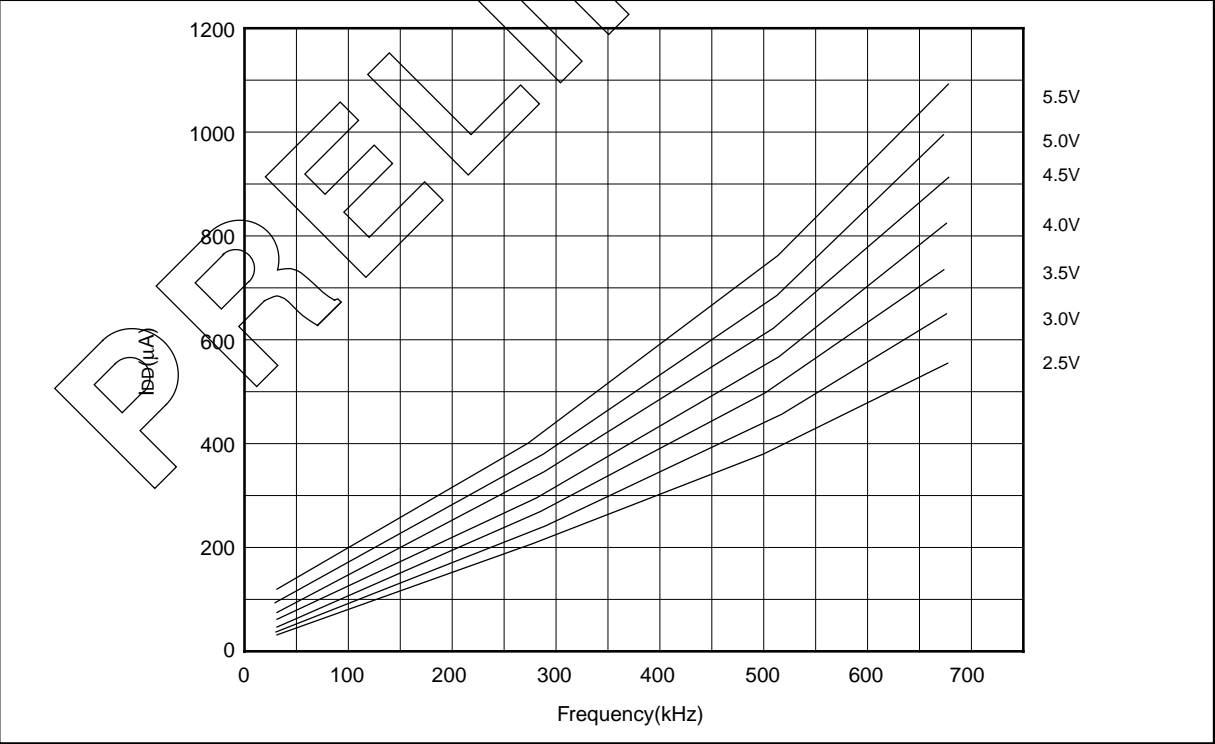


FIGURE 14-18: TYPICAL I_{DD} vs. CAPACITANCE @ 500 kHz (RC MODE)

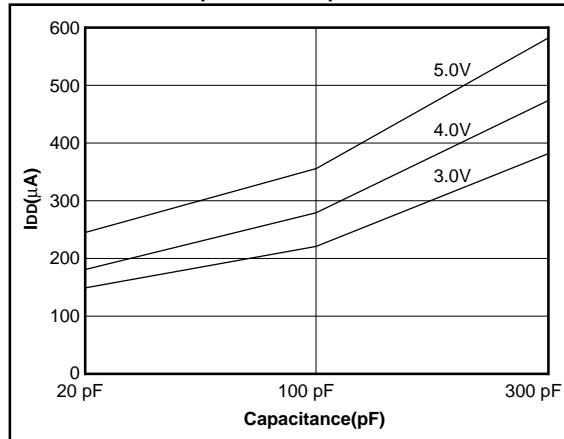


TABLE 14-1: RC OSCILLATOR FREQUENCIES

| Cext | Rext | Average | |
|--------|------|-----------------|--------|
| | | Fosc @ 5V, 25°C | |
| 22 pF | 5k | 4.12 MHz | ± 1.4% |
| | 10k | 2.35 MHz | ± 1.4% |
| | 100k | 268 kHz | ± 1.1% |
| 100 pF | 3.3k | 1.80 MHz | ± 1.0% |
| | 5k | 1.27 MHz | ± 1.0% |
| | 10k | 688 kHz | ± 1.2% |
| | 100k | 77.2 kHz | ± 1.0% |
| 300 pF | 3.3k | 707 kHz | ± 1.4% |
| | 5k | 501 kHz | ± 1.2% |
| | 10k | 269 kHz | ± 1.6% |
| | 100k | 28.3 kHz | ± 1.1% |

The percentage variation indicated here is part to part variation due to normal process distribution. The variation indicated is ± 3 standard deviation from average value for $V_{DD} = 5V$.

FIGURE 14-19: TRANSCONDUCTANCE(gm) OF HS OSCILLATOR vs. V_{DD}

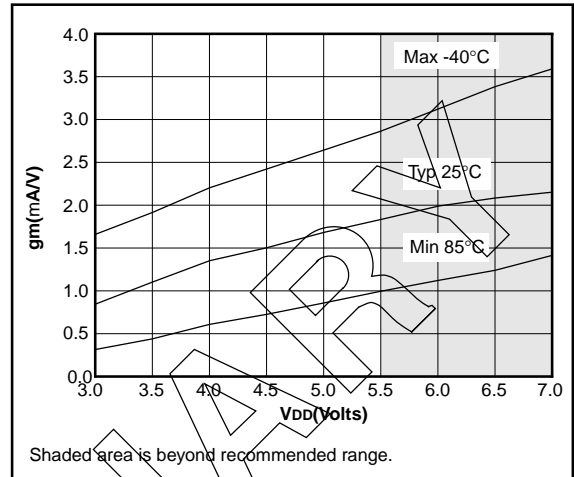


FIGURE 14-20: TRANSCONDUCTANCE(gm) OF LP OSCILLATOR vs. V_{DD}

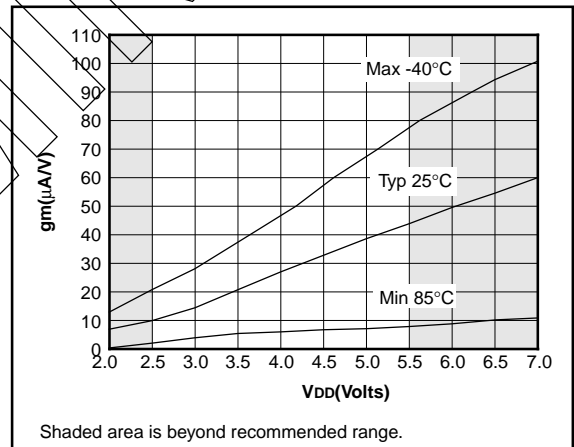
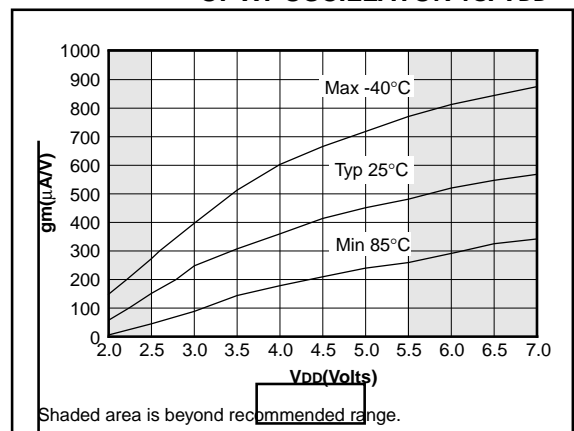


FIGURE 14-21: TRANSCONDUCTANCE(gm) OF XT OSCILLATOR vs. V_{DD}



PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 14-22: TYPICAL XTAL STARTUP TIME vs. VDD (LP MODE, 25°C)

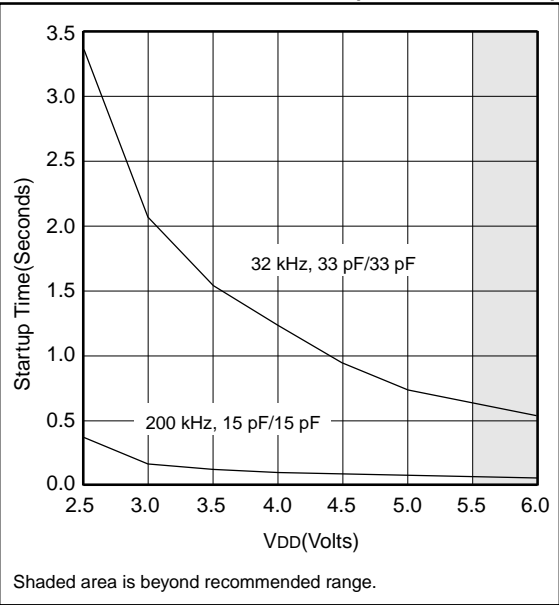


FIGURE 14-24: TYPICAL XTAL STARTUP TIME vs. VDD (XT MODE, 25°C)

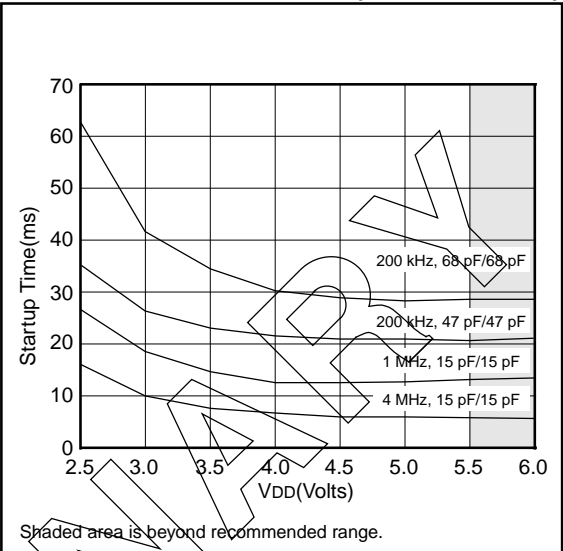


FIGURE 14-23: TYPICAL XTAL STARTUP TIME vs. VDD (HS MODE, 25°C)

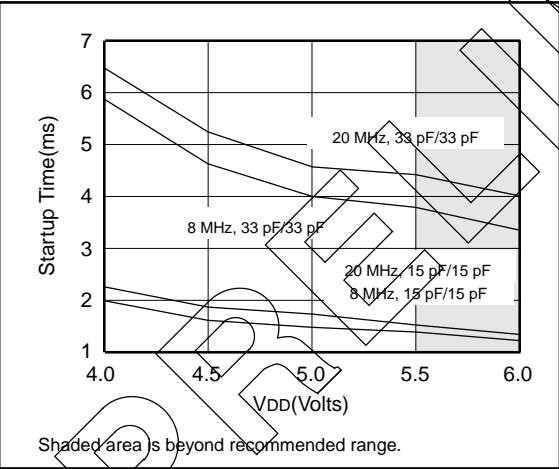


TABLE 14-2: CAPACITOR SELECTION FOR CRYSTAL OSCILLATORS

| 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 |
| Crystals Used | | | |
| 32 kHz | Epson C-001R32.768K-A | ± 20 PPM | |
| 200 kHz | STD XTL 200.000KHz | ± 20 PPM | |
| 1 MHz | ECS ECS-10-13-1 | ± 50 PPM | |
| 4 MHz | ECS ECS-40-20-1 | ± 50 PPM | |
| 8 MHz | EPSON CA-301 8.000M-C | ± 30 PPM | |
| 20 MHz | EPSON CA-301 20.000M-C | ± 30 PPM | |

FIGURE 14-25: TYPICAL I_{DD} vs. FREQUENCY
(LP MODE, 25°C)

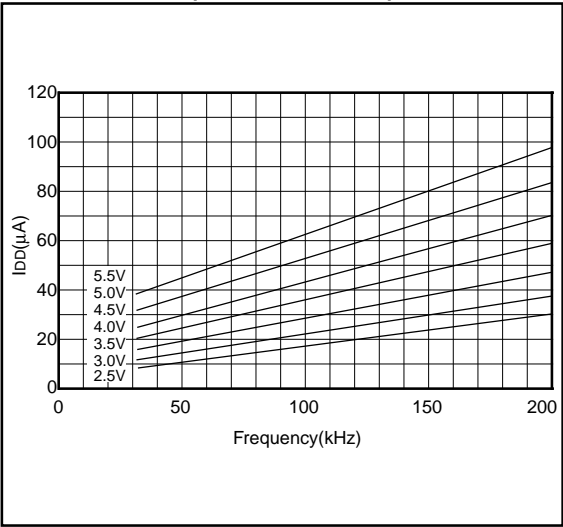


FIGURE 14-27: TYPICAL I_{DD} vs. FREQUENCY
(XT MODE, 25°C)

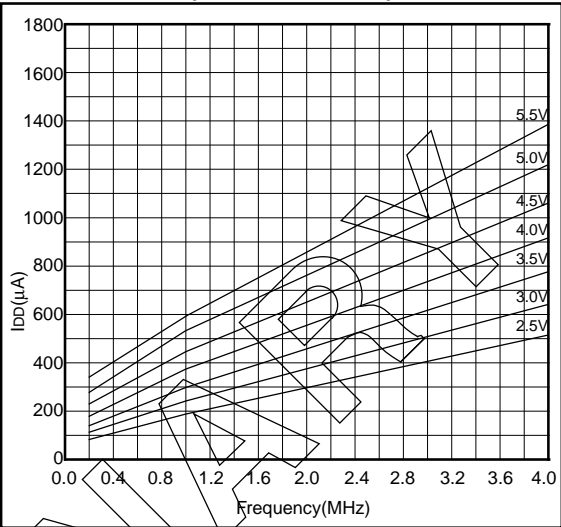


FIGURE 14-26: MAXIMUM I_{DD} vs.
FREQUENCY
(LP MODE, 85°C TO -40°C)

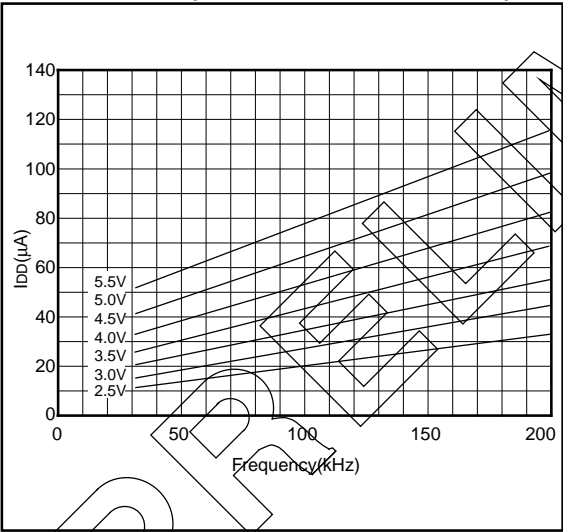
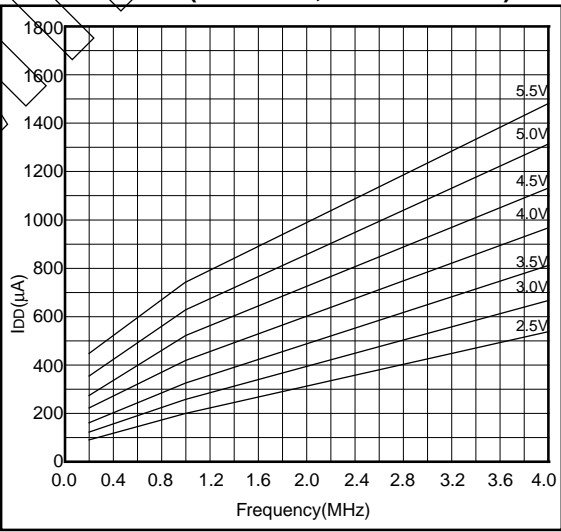


FIGURE 14-28: MAXIMUM I_{DD} vs.
FREQUENCY
(XT MODE, -40°C TO 85°C)



PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 14-29: TYPICAL I_{DD} vs. FREQUENCY
(HS MODE, 25°C)

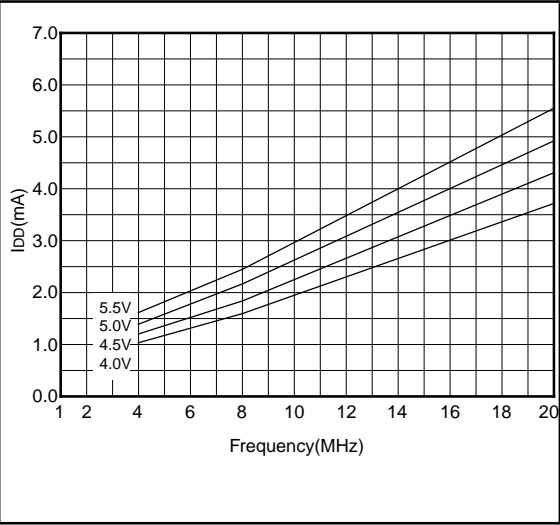
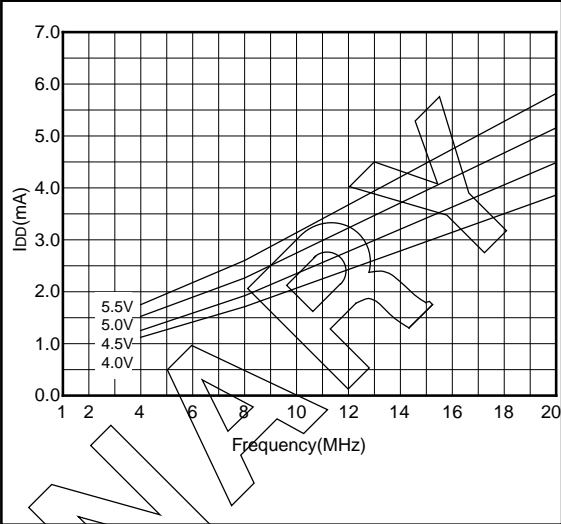


FIGURE 14-30: MAXIMUM I_{DD} vs. FREQUENCY
(HS MODE, -40°C TO 85°C)



PIC16C71X

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

15.0 ELECTRICAL CHARACTERISTICS FOR PIC16C71

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 |
| Voltage on $\overline{\text{MCLR}}$ with respect to Vss (Note 2)..... | 0 to +14V |
| Voltage on RA4 with respect to Vss | 0 to +14V |
| Total power dissipation (Note 1) | 800 mW |
| Maximum current out of Vss pin | 150 mA |
| Maximum current into VDD pin | 100 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 | 20 mA |
| Maximum current sunk by PORTA | 80 mA |
| Maximum current sourced by PORTA | 50 mA |
| Maximum current sunk by PORTB..... | 150 mA |
| Maximum current sourced by PORTB..... | 100 mA |

Note 1: Power dissipation is calculated as follows: $P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$

Note 2: Voltage spikes below Vss at the $\overline{\text{MCLR}}$ pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100Ω should be used when applying a “low” level to the $\overline{\text{MCLR}}$ pin rather than pulling this pin directly to Vss.

† 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.

TABLE 15-1: CROSS REFERENCE OF DEVICE SPECS FOR OSCILLATOR CONFIGURATIONS AND FREQUENCIES OF OPERATION (COMMERCIAL DEVICES)

| OSC | PIC16C71-04 | PIC16C71-20 | PIC16LC71-04 | JW Devices |
|-----|--|---|--|--|
| RC | VDD: 4.0V to 6.0V IDD: 3.3 mA max. at 5.5V IPD: 14 μA max. at 4V Freq: 4 MHz max. | VDD: 4.5V to 5.5V IDD: 1.8 mA typ. at 5.5V IPD: 1.0 μA typ. at 4V Freq: 4 MHz max. | VDD: 3.0V to 6.0V IDD: 1.4 mA typ. at 3.0V IPD: 0.6 μA typ. at 3V Freq: 4 MHz max. | VDD: 4.0V to 6.0V IDD: 3.3 mA max. at 5.5V IPD: 14 μA max. at 4V Freq: 4 MHz max. |
| XT | VDD: 4.0V to 6.0V IDD: 3.3 mA max. at 5.5V IPD: 14 μA max. at 4V Freq: 4 MHz max. | VDD: 4.5V to 5.5V IDD: 1.8 mA typ. at 5.5V IPD: 1.0 μA typ. at 4V Freq: 4 MHz max. | VDD: 3.0V to 6.0V IDD: 1.4 mA typ. at 3.0V IPD: 0.6 μA typ. at 3V Freq: 4 MHz max. | VDD: 4.0V to 6.0V IDD: 3.3 mA max. at 5.5V IPD: 14 μA max. at 4V Freq: 4 MHz max. |
| HS | VDD: 4.5V to 5.5V IDD: 13.5 mA typ. at 5.5V IPD: 1.0 μA typ. at 4.5V Freq: 4 MHz max. | VDD: 4.5V to 5.5V IDD: 30 mA max. at 5.5V IPD: 1.0 μA typ. at 4.5V Freq: 20 MHz max. | Not recommended for use in HS mode | VDD: 4.5V to 5.5V IDD: 30 mA max. at 5.5V IPD: 1.0 μA typ. at 4.5V Freq: 20 MHz max. |
| LP | VDD: 4.0V to 6.0V IDD: 15 μA typ. at 32 kHz, 4.0V IPD: 0.6 μA typ. at 4.0V Freq: 200 kHz max. | Not recommended for use in LP mode | VDD: 3.0V to 6.0V IDD: 32 μA max. at 32 kHz, 3.0V IPD: 9 μA max. at 3.0V Freq: 200 kHz max. | VDD: 3.0V to 6.0V IDD: 32 μA max. at 32 kHz, 3.0V IPD: 9 μA max. at 3.0V Freq: 200 kHz max. |

The shaded sections indicate oscillator selections which are tested for functionality, but not for MIN/MAX specifications. It is recommended that the user select the device type that ensures the specifications required.

PIC16C71X

| | | | | |
|--------------------|-----|----|-----|-----|
| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|

15.1 DC Characteristics: PIC16C71-04 (Commercial, Industrial) PIC16C71-20 (Commercial, Industrial)

| DC CHARACTERISTICS | | Standard Operating Conditions (unless otherwise stated) Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial) | | | | | |
|-----------------------|--|---|-------------|-----------------|----------------|----------------|--|
| Param No. | Characteristic | Sym | Min | Typ† | Max | Units | Conditions |
| D001 D001A | Supply Voltage | VDD | 4.0 4.5 | - - | 6.0 5.5 | V V | XT, RC and LP osc configuration HS osc configuration |
| D002* | RAM Data Retention Voltage (Note 1) | VDR | - | 1.5 | - | V | |
| D003 | VDD start voltage to ensure internal Power-on Reset signal | VPOR | - | VSS | - | V | See section on Power-on Reset for details |
| D004* | VDD rise rate to ensure internal Power-on Reset signal | SVDD | 0.05 | - | - | V/ms | See section on Power-on Reset for details |
| D010 D013 | Supply Current (Note 2) | IDD | - - | 1.8 13.5 | 3.3 30 | mA mA | XT, RC osc configuration FOSC = 4 MHz, VDD = 5.5V (Note 4) HS osc configuration FOSC = 20 MHz, VDD = 5.5V |
| D020 D021 D021A | Power-down Current (Note 3) | IPD | - - - | 7 1.0 1.0 | 28 14 16 | μA μA μA | VDD = 4.0V, WDT enabled, -40°C to +85°C VDD = 4.0V, WDT disabled, -0°C to +70°C VDD = 4.0V, WDT disabled, -40°C to +85°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 and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula $I_r = VDD/2R_{ext}$ (mA) with Rext in kOhm.

PIC16C71X

Applicable Devices 710 71 711 715

15.2 DC Characteristics: PIC16LC71-04 (Commercial, Industrial)

| DC CHARACTERISTICS | | Standard Operating Conditions (unless otherwise stated) | | | | | |
|--------------------|--|---|------|------|-----|-------|---|
| | | Operating temperature 0°C ≤ TA ≤ +70°C (commercial) | | | | | |
| | | -40°C ≤ TA ≤ +85°C (industrial) | | | | | |
| Param No. | Characteristic | Sym | Min | Typ† | Max | Units | Conditions |
| D001 | Supply Voltage | VDD | 3.0 | - | 6.0 | V | XT, RC, and LP osc configuration |
| D002* | RAM Data Retention Voltage (Note 1) | VDR | - | 1.5 | - | V | |
| D003 | VDD start voltage to ensure internal Power-on Reset signal | VPOR | - | VSS | - | V | See section on Power-on Reset for details |
| D004* | VDD rise rate to ensure internal Power-on Reset signal | SVDD | 0.05 | - | - | V/ms | See section on Power-on Reset for details |
| D010 | Supply Current (Note 2) | IDD | - | 1.4 | 2.5 | mA | XT, RC osc configuration FOSC = 4 MHz, VDD = 3.0V (Note 4) |
| D010A | | | | 15 | 32 | μA | LP osc configuration FOSC = 32 kHz, VDD = 3.0V, WDT disabled |
| D020 | Power-down Current (Note 3) | IPD | - | 5 | 20 | μA | VDD = 3.0V, WDT enabled, -40°C to +85°C |
| D021 | | | | 0.6 | 9 | μA | VDD = 3.0V, WDT disabled, 0°C to +70°C |
| D021A | | | | 0.6 | 12 | μA | VDD = 3.0V, WDT disabled, -40°C to +85°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 and VSS.

4: For RC osc configuration, current through Rext is not included. The current through the resistor can be estimated by the formula $I_r = VDD/2R_{ext}$ (mA) with Rext in kOhm.

PIC16C71X

| | | | | |
|--------------------|-----|----|-----|-----|
| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|

15.3 DC Characteristics: PIC16C71-04 (Commercial, Industrial) PIC16C71-20 (Commercial, Industrial) PIC16LC71-04 (Commercial, Industrial)

| Standard Operating Conditions (unless otherwise stated) Operating temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ (commercial) $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (industrial) Operating voltage V_{DD} range as described in DC spec Section 15.1 and Section 15.2. | | | | | | | |
|---|---|------------|----------------------|-------|--------------|---------------|---|
| Param No. | Characteristic | Sym | Min | Typ † | Max | Units | Conditions |
| D030 | Input Low Voltage I/O ports | V_{IL} | V_{SS} | - | 0.15V | V | For entire V_{DD} range $4.5 \leq V_{DD} \leq 5.5\text{V}$ |
| D031 | with TTL buffer | | V_{SS} | - | 0.8V | V | |
| D032 | with Schmitt Trigger buffer | | V_{SS} | - | 0.2 V_{DD} | V | |
| D033 | $\overline{\text{MCLR}}$, OSC1 (in RC mode) OSC1 (in XT, HS and LP) | | V_{SS} | - | 0.3 V_{DD} | V | |
| D040 | Input High Voltage I/O ports (Note 4) | V_{IH} | 2.0 | - | V_{DD} | V | For entire V_{DD} range $4.5 \leq V_{DD} \leq 5.5\text{V}$ |
| D040A | with TTL buffer | | 0.25 V_{DD} + 0.8V | - | V_{DD} | V | |
| D041 | with Schmitt Trigger buffer | | 0.85 V_{DD} | - | V_{DD} | V | |
| D042 | $\overline{\text{MCLR}}$, RB0/INT | | 0.85 V_{DD} | - | V_{DD} | V | |
| D042A | OSC1 (XT, HS and LP) | | 0.7 V_{DD} | - | V_{DD} | V | |
| D043 | OSC1 (in RC mode) | | 0.9 V_{DD} | - | V_{DD} | V | |
| D070 | PORTB weak pull-up current | I_{PURB} | 50 | 250 | †400 | μA | $V_{DD} = 5\text{V}$, $V_{PIN} = V_{SS}$ |
| D060 | Input Leakage Current (Notes 2, 3) I/O ports | I_{IL} | - | - | ±1 | μA | $V_{SS} \leq V_{PIN} \leq V_{DD}$, Pin at hi-impedance |
| D061 | $\overline{\text{MCLR}}$, RA4/T0CKI | | - | - | ±5 | μA | $V_{SS} \leq V_{PIN} \leq V_{DD}$ |
| D063 | OSC1 | | - | - | ±5 | μA | $V_{SS} \leq V_{PIN} \leq V_{DD}$, XT, HS and LP osc configuration |
| D080 | Output Low Voltage I/O ports | V_{OL} | - | - | 0.6 | V | $I_{OL} = 8.5\text{mA}$, $V_{DD} = 4.5\text{V}$, -40°C to +85°C |
| D083 | OSC2/CLKOUT (RC osc config) | | - | - | 0.6 | V | $I_{OL} = 1.6\text{mA}$, $V_{DD} = 4.5\text{V}$, -40°C to +85°C |
| D090 | Output High Voltage I/O ports (Note 3) | V_{OH} | $V_{DD} - 0.7$ | - | - | V | $I_{OH} = -3.0\text{mA}$, $V_{DD} = 4.5\text{V}$, -40°C to +85°C |
| D092 | OSC2/CLKOUT (RC osc config) | | $V_{DD} - 0.7$ | - | - | V | $I_{OH} = -1.3\text{mA}$, $V_{DD} = 4.5\text{V}$, -40°C to +85°C |
| D130* | Open-Drain High Voltage | V_{OD} | - | - | 14 | V | RA4 pin |

† 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: In RC oscillator configuration, the OSC1 pin is a Schmitt trigger input. It is not recommended that the PIC16C71 be driven with external clock in RC mode.
- 2: 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.
- 3: Negative current is defined as current sourced by the pin.
- 4: PIC16C71 Rev. "Ax" INT pin has a TTL input buffer. PIC16C71 Rev. "Bx" INT pin has a Schmitt Trigger input buffer.

PIC16C71X

| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|
|--------------------|-----|----|-----|-----|

| DC CHARACTERISTICS | | Standard Operating Conditions (unless otherwise stated) | | | | | |
|--------------------|--|---|-----|------|-----|-------|---|
| | | Operating temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ (commercial) $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (industrial) Operating voltage V_{DD} range as described in DC spec Section 15.1 and Section 15.2. | | | | | |
| Param No. | Characteristic | Sym | Min | Typ† | Max | Units | Conditions |
| D100 | Capacitive Loading Specs on Output Pins | | | | | | |
| | OSC2 pin | COSC2 | | | 15 | pF | In XT, HS and LP modes when external clock is used to drive OSC1. |
| D101 | All I/O pins and OSC2 (in RC mode) | CIO | | | 50 | pF | |

† 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: In RC oscillator configuration, the OSC1 pin is a Schmitt trigger input. It is not recommended that the PIC16C71 be driven with external clock in RC mode.
- 2: 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.
- 3: Negative current is defined as current sourced by the pin.
- 4: PIC16C71 Rev. "Ax" INT pin has a TTL input buffer. PIC16C71 Rev. "Bx" INT pin has a Schmitt Trigger input buffer.

PIC16C71X

| | | | | |
|--------------------|-----|----|-----|-----|
| Applicable Devices | 710 | 71 | 711 | 715 |
|--------------------|-----|----|-----|-----|

15.4 Timing Parameter Symbolology

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

- 1. TppS2ppS
- 2. TppS

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

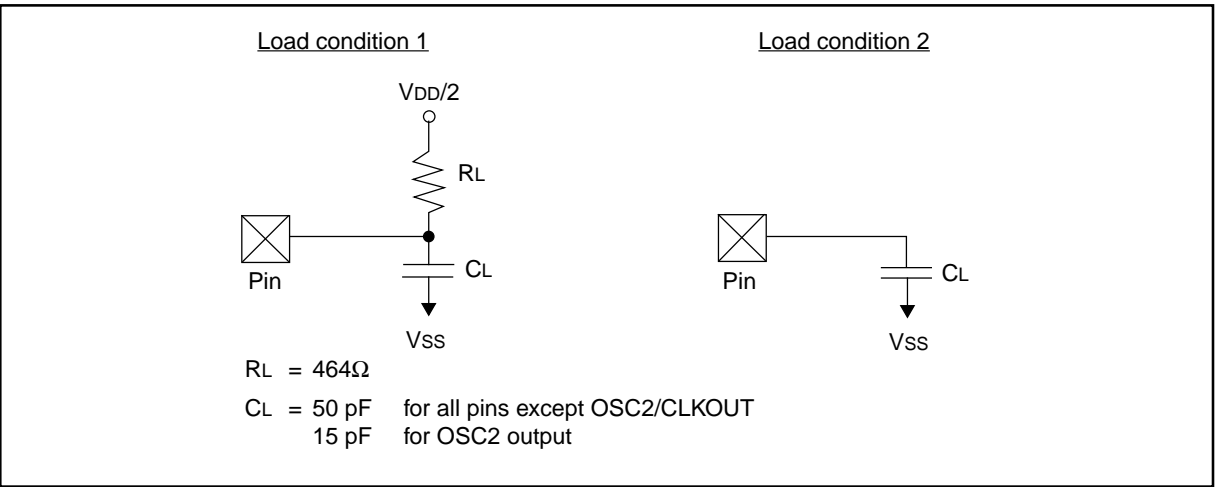
Lowercase letters (pp) and their meanings:

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

Uppercase letters and their meanings:

| | | |
|----------|------------------------|--------------|
| S | | |
| F | Fall | P |
| H | High | R |
| I | Invalid (Hi-impedance) | V |
| L | Low | Z |
| | | Period |
| | | Rise |
| | | Valid |
| | | Hi-impedance |

FIGURE 15-1: LOAD CONDITIONS



15.5 Timing Diagrams and Specifications

FIGURE 15-2: EXTERNAL CLOCK TIMING

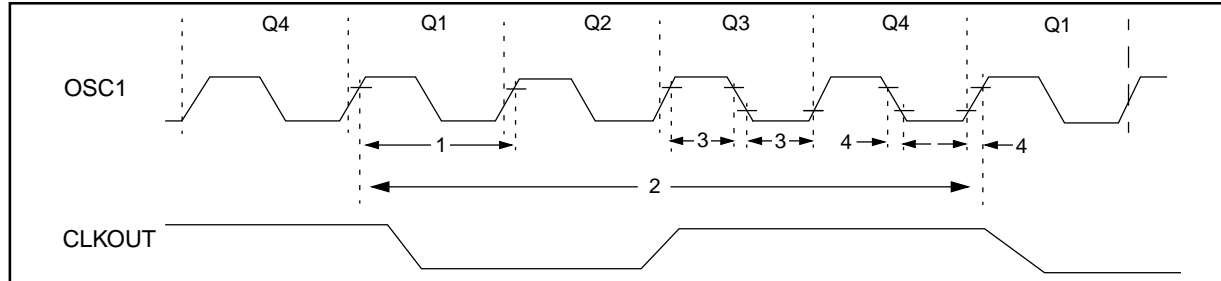


TABLE 15-2: EXTERNAL CLOCK TIMING REQUIREMENTS

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|------------|---|-----|------|--------|-------|-------------------|
| | Fosc | External CLKIN Frequency (Note 1) | DC | — | 4 | MHz | XT osc mode |
| | | | DC | — | 4 | MHz | HS osc mode (-04) |
| | | | DC | — | 20 | MHz | HS osc mode (-20) |
| | | | DC | — | 200 | kHz | LP osc mode |
| | | Oscillator Frequency (Note 1) | DC | — | 4 | MHz | RC osc mode |
| | | | 0.1 | — | 4 | MHz | XT osc mode |
| | | | 1 | — | 4 | MHz | HS osc mode |
| | | | 1 | — | 20 | MHz | HS osc mode |
| 1 | Tosc | External CLKIN Period (Note 1) | 250 | — | — | ns | XT osc mode |
| | | | 250 | — | — | ns | HS osc mode (-04) |
| | | | 50 | — | — | ns | HS osc mode (-20) |
| | | | 5 | — | — | μs | LP osc mode |
| | | Oscillator Period (Note 1) | 250 | — | — | ns | RC osc mode |
| | | | 250 | — | 10,000 | ns | XT osc mode |
| | | | 250 | — | 1,000 | ns | HS osc mode (-04) |
| | | | 50 | — | 1,000 | ns | HS osc mode (-20) |
| 2 | TCY | Instruction Cycle Time (Note 1) | 1.0 | TCY | DC | μs | TCY = 4/Fosc |
| | | | 50 | — | — | ns | XT oscillator |
| | | | 2.5 | — | — | μs | LP oscillator |
| 3 | TosL, TosH | External Clock in (OSC1) High or Low Time | 10 | — | — | ns | HS oscillator |
| | | | 25 | — | — | ns | XT oscillator |
| | | | 50 | — | — | ns | LP oscillator |
| 4 | TosR, TosF | External Clock in (OSC1) Rise or Fall Time | 15 | — | — | ns | HS oscillator |
| | | | 25 | — | — | ns | XT oscillator |
| | | | 50 | — | — | ns | LP oscillator |

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

Note 1: Instruction cycle period (TCY) equals four times the input oscillator time-base period. All specified values are based on characterization data for that particular oscillator type under standard operating conditions with the device executing code. Exceeding these specified limits may result in an unstable oscillator operation and/or higher than expected current consumption. All devices are tested to operate at "min." values with an external clock applied to the OSC1/CLKIN pin. When an external clock input is used, the "Max." cycle time limit is "DC" (no clock) for all devices. OSC2 is disconnected (has no loading) for the PIC16C71.

PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 15-3: CLKOUT AND I/O TIMING

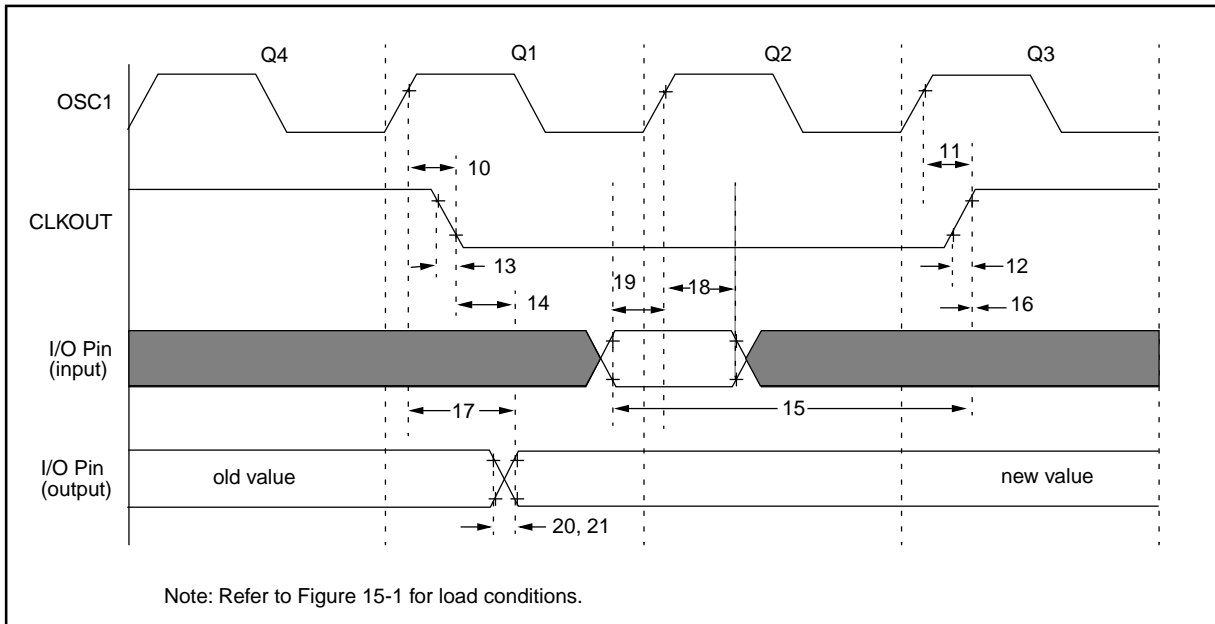


TABLE 15-3: CLKOUT AND I/O TIMING REQUIREMENTS

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|----------|---|--------------------------|------|-------------------------|-------|------------|
| 10* | TosH2ckL | OSC1↑ to CLKOUT↓ | — | 15 | 30 | ns | Note 1 |
| 11* | TosH2ckH | OSC1↑ to CLKOUT↑ | — | 15 | 30 | ns | Note 1 |
| 12* | TckR | CLKOUT rise time | — | 5 | 15 | ns | Note 1 |
| 13* | TckF | CLKOUT fall time | — | 5 | 15 | 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* | TckH2ioI | Port in hold after CLKOUT ↑ | 0 | — | — | ns | Note 1 |
| 17* | TosH2ioV | OSC1↑ (Q1 cycle) to Port out valid | — | — | 80 - 100 | ns | |
| 18* | TosH2ioI | OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time) | PIC16C71 | 100 | — | ns | |
| | | | PIC16LC71 | 200 | — | ns | |
| 19* | TioV2osH | Port input valid to OSC1↑ (I/O in setup time) | 0 | — | — | ns | |
| 20* | TioR | Port output rise time | PIC16C71 | — | 10 | ns | |
| | | | PIC16LC71 | — | — | 60 | ns |
| 21* | TioF | Port output fall time | PIC16C71 | — | 10 | ns | |
| | | | PIC16LC71 | — | — | 60 | ns |
| 22††* | Tinp | INT pin high or low time | 20 | — | — | ns | |
| 23††* | Trbp | RB7:RB4 change INT high or low time | 20 | — | — | ns | |

* These parameters are characterized but not tested.

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

†† These parameters are asynchronous events not related to any internal clock edges.

Note 1: Measurements are taken in RC Mode where CLKOUT output is 4 x TOSC.

FIGURE 15-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING

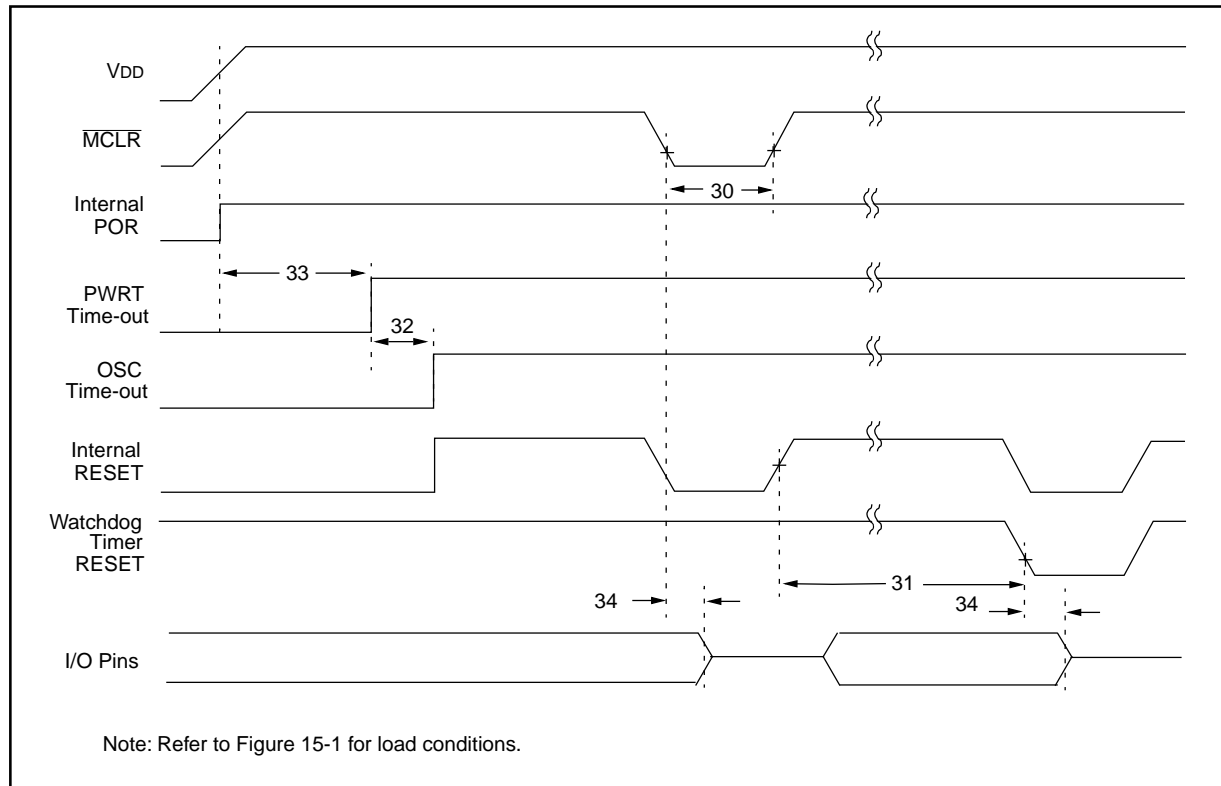


TABLE 15-4: RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER REQUIREMENTS

| Parameter No. | Sym | Characteristic | Min | Typ† | Max | Units | Conditions |
|---------------|-------|---|-----|-----------|------|-------|--------------------------|
| 30 | Tmcl | MCLR Pulse Width (low) | 200 | — | — | ns | 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 High Impedance from MCLR Low | — | — | 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.

PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 15-5: TIMER0 EXTERNAL CLOCK TIMINGS

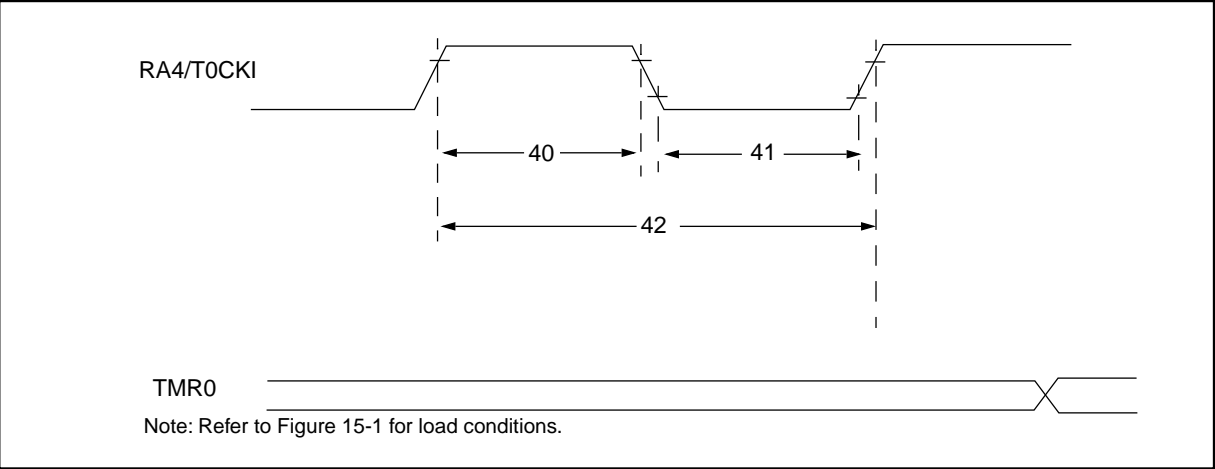


TABLE 15-5: TIMER0 EXTERNAL CLOCK REQUIREMENTS

| 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: $20 \text{ ns or } \frac{T_{CY} + 40}{N}$ | | | | |

* 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.

PIC16C71X

Applicable Devices 710 71 711 715

TABLE 15-6: A/D CONVERTER CHARACTERISTICS

| Param No. | Sym | Characteristic | | Min | Typ† | Max | Units | Conditions |
|-----------|------|--|-----------|----------------|------------|----------------|------------|---|
| A01 | NR | Resolution | | — | — | 8 bits | bits | $V_{REF} = V_{DD} = 5.12V$, $V_{SS} \leq V_{AIN} \leq V_{REF}$ |
| A02 | EABS | Absolute error | PIC16C71 | — | — | $< \pm 1$ | LSb | $V_{REF} = V_{DD} = 5.12V$, $V_{SS} \leq V_{AIN} \leq V_{REF}$ |
| | | | PIC16LC71 | — | — | $< \pm 2$ | LSb | $V_{REF} = V_{DD} = 3.0V$ (Note 3) |
| A03 | EIL | Integral linearity error | PIC16C71 | — | — | $< \pm 1$ | LSb | $V_{REF} = V_{DD} = 5.12V$, $V_{SS} \leq V_{AIN} \leq V_{REF}$ |
| | | | PIC16LC71 | — | — | $< \pm 2$ | LSb | $V_{REF} = V_{DD} = 3.0V$ (Note 3) |
| A04 | EDL | Differential linearity error | PIC16C71 | — | — | $< \pm 1$ | LSb | $V_{REF} = V_{DD} = 5.12V$, $V_{SS} \leq V_{AIN} \leq V_{REF}$ |
| | | | PIC16LC71 | — | — | $< \pm 2$ | LSb | $V_{REF} = V_{DD} = 3.0V$ (Note 3) |
| A05 | EFS | Full scale error | PIC16C71 | — | — | $< \pm 1$ | LSb | $V_{REF} = V_{DD} = 5.12V$, $V_{SS} \leq V_{AIN} \leq V_{REF}$ |
| | | | PIC16LC71 | — | — | $< \pm 2$ | LSb | $V_{REF} = V_{DD} = 3.0V$ (Note 3) |
| A06 | EOFF | Offset error | PIC16C71 | — | — | $< \pm 1$ | LSb | $V_{REF} = V_{DD} = 5.12V$, $V_{SS} \leq V_{AIN} \leq V_{REF}$ |
| | | | PIC16LC71 | — | — | $< \pm 2$ | LSb | $V_{REF} = V_{DD} = 3.0V$ (Note 3) |
| A10 | — | Monotonicity | | — | guaranteed | — | — | $V_{SS} \leq V_{AIN} \leq V_{REF}$ |
| A20 | VREF | Reference voltage | | 3.0V | — | $V_{DD} + 0.3$ | V | |
| A25 | VAIN | Analog input voltage | | $V_{SS} - 0.3$ | — | VREF | V | |
| A30 | ZAIN | Recommended impedance of analog voltage source | | — | — | 10.0 | k Ω | |
| A40 | IAD | A/D conversion current (V_{DD}) | | — | 180 | — | μA | Average current consumption when A/D is on. (Note 1) |
| A50 | IREF | VREF input current (Note 2) | PIC16C71 | 10 | — | 1000 | μA | During VAIN acquisition. Based on differential of VHOLD to VAIN. To charge CHOLD see Section 7.1. |
| | | | | — | — | 40 | μA | During A/D Conversion cycle |
| | | | PIC16LC71 | — | — | 1 | mA | During VAIN acquisition. Based on differential of VHOLD to VAIN. To charge CHOLD see Section 7.1. |
| | | | | — | — | 10 | μA | 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 current is from RA3 pin or VDD pin, whichever is selected as reference input.
- 3: These specifications apply if $V_{REF} = 3.0V$ and if $V_{DD} \geq 3.0V$. VAIN must be between V_{SS} and V_{REF} .

PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 15-6: A/D CONVERSION TIMING

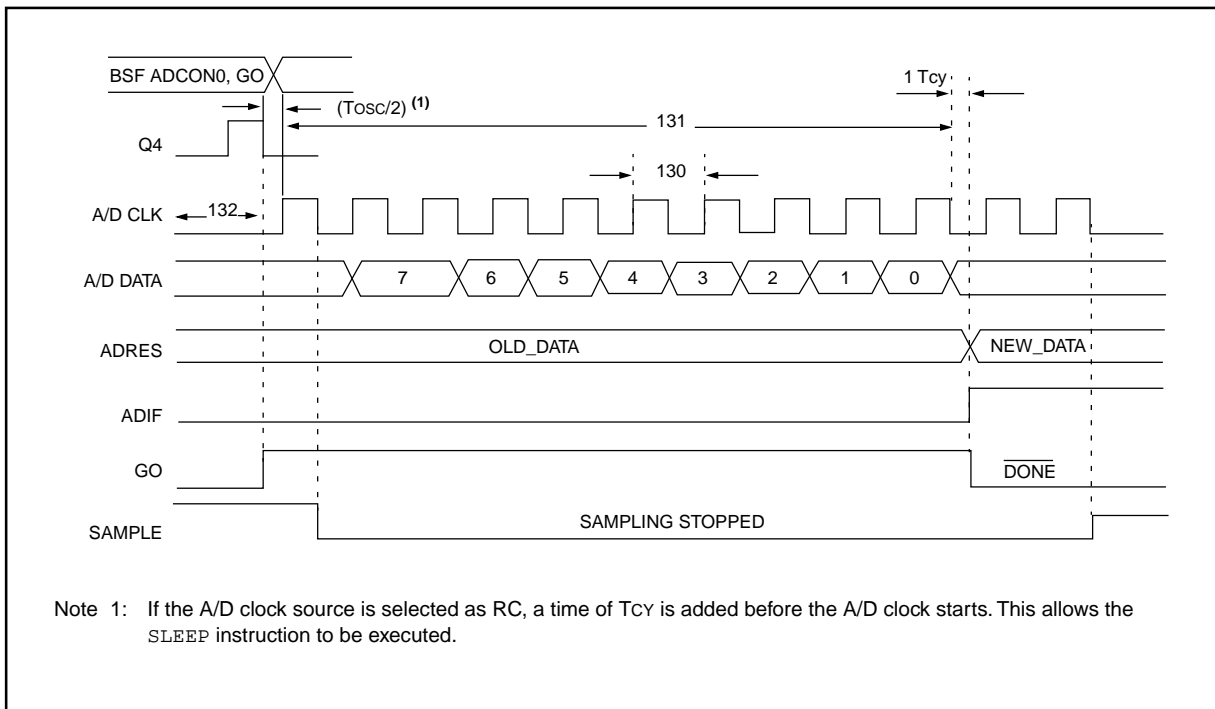


TABLE 15-7: A/D CONVERSION REQUIREMENTS

| Param No. | Sym | Characteristic | | Min | Typ† | Max | Units | Conditions |
|-----------|------|---|--------------------|--------|---------|-----|-------|---|
| 130 | TAD | A/D clock period | PIC16 C 71 | 2.0 | — | — | μs | TOSC based, VREF ≥ 3.0V |
| | | | PIC16 L C71 | 2.0 | — | — | μs | TOSC based, VREF full range |
| | | | PIC16 C 71 | 2.0 | 4.0 | 6.0 | μs | A/D RC Mode |
| | | | PIC16 L C71 | 3.0 | 6.0 | 9.0 | μs | A/D RC Mode |
| 131 | TCNV | Conversion time (not including S/H time) (Note 1) | | — | 9.5 | — | TAD | |
| 132 | TACQ | Acquisition time | | Note 2 | 20 | — | μs | The minimum time is the amplifier settling time. This may be used if the "new" input voltage has not changed by more than 1 LSB (i.e., 19.5 mV @ 5.12V) from the last sampled voltage (as stated on CHOLD). |
| | | | | 5* | — | — | μs | |
| 134 | TGO | Q4 to A/D clock start | | — | Tosc/2§ | — | — | If the A/D clock source is selected as RC, a time of Tcy is added before the A/D clock starts. This allows the SLEEP instruction to be executed. |
| 135 | TSWC | Switching from convert → sample time | | 1.5§ | — | — | TAD | |

* 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 specifications ensured by design.

Note 1: ADRES register may be read on the following T_{CY} cycle.

2: See Section 7.1 for min conditions.

16.0 DC AND AC CHARACTERISTICS GRAPHS AND TABLES FOR PIC16C71

The graphs and tables provided in this section are for design guidance and are not tested or guaranteed. In some graphs or tables the data presented are outside specified operating range (e.g. outside specified V_{DD} range). This is for information only and devices are guaranteed to operate properly only within the specified range.

Note: The data presented in this section is a statistical summary of data collected on units from different lots over a period of time and matrix samples. 'Typical' represents the mean of the distribution while 'max' or 'min' represents (mean + 3σ) and (mean - 3σ) respectively where σ is standard deviation.

FIGURE 16-1: TYPICAL RC OSCILLATOR FREQUENCY vs. TEMPERATURE

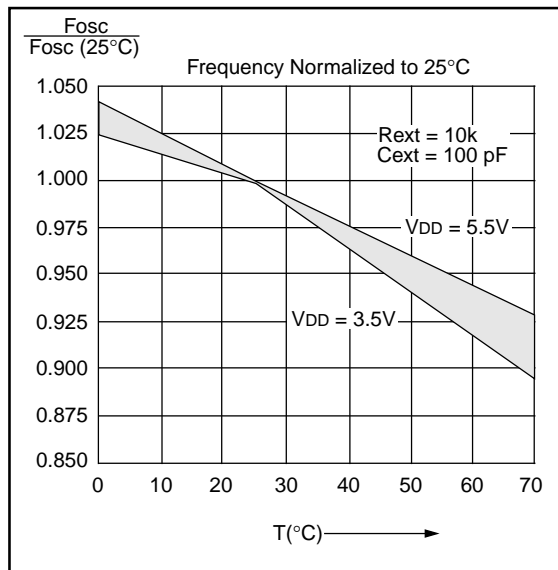


FIGURE 16-2: TYPICAL RC OSCILLATOR FREQUENCY vs. V_{DD}

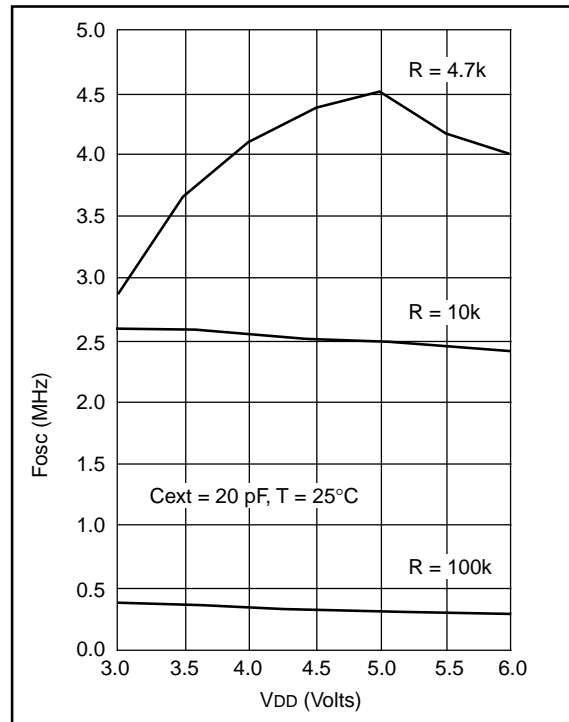
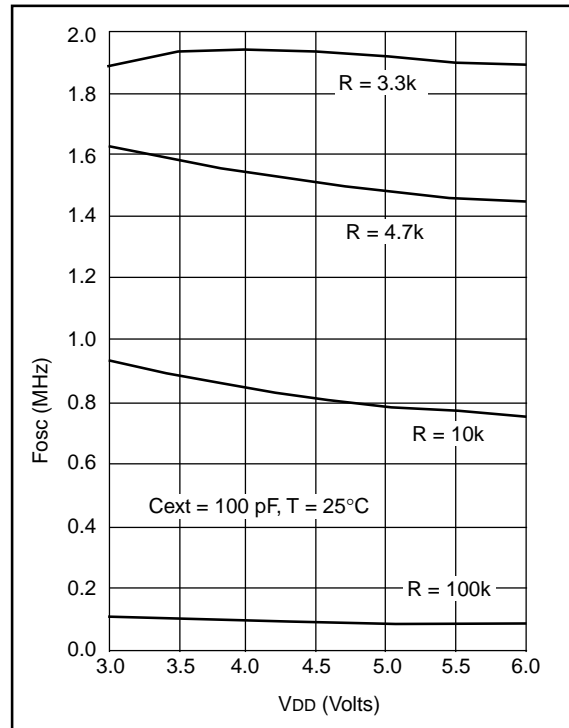


FIGURE 16-3: TYPICAL RC OSCILLATOR FREQUENCY vs. V_{DD}



PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 16-4: TYPICAL RC OSCILLATOR
FREQUENCY vs. VDD

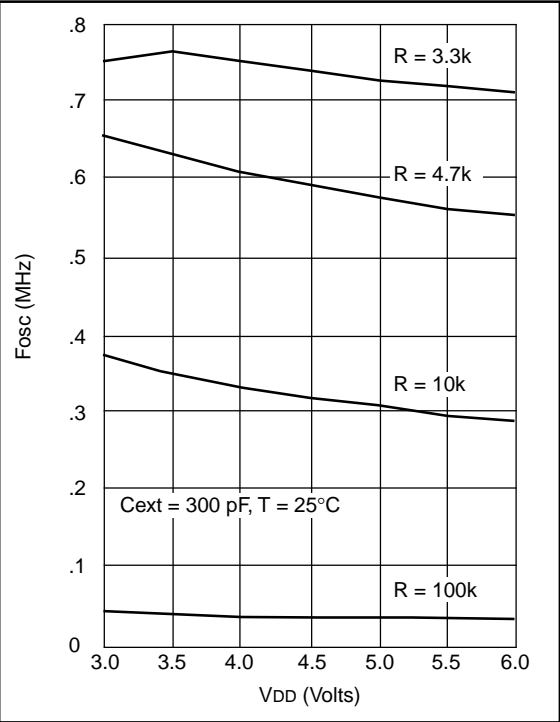


FIGURE 16-5: TYPICAL IPD vs. VDD
WATCHDOG TIMER
DISABLED 25°C

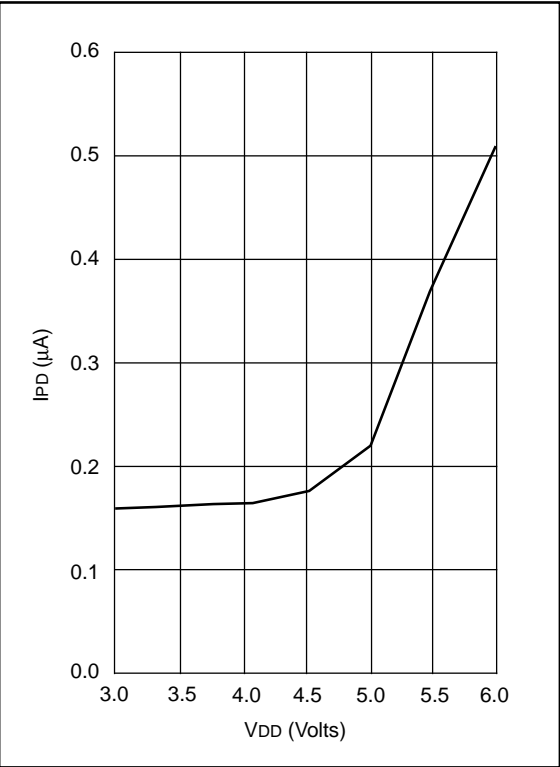
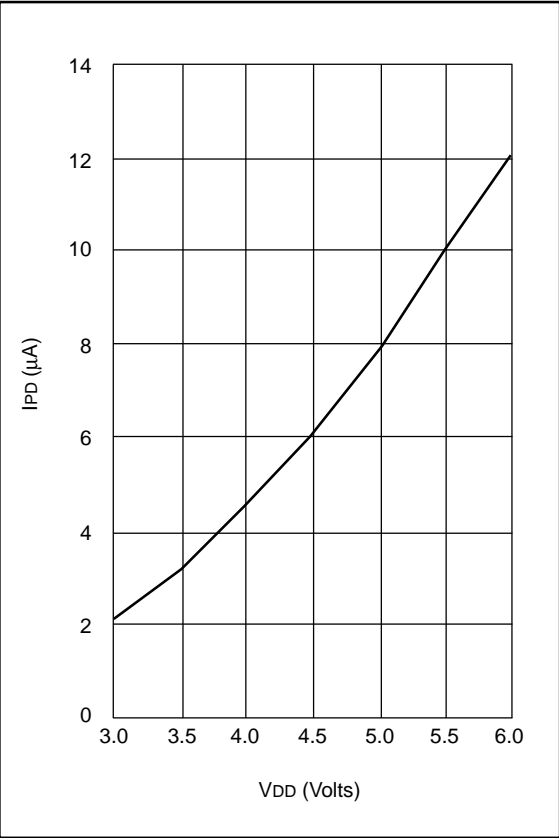


TABLE 16-1: RC OSCILLATOR
FREQUENCIES

| Cext | Rext | Average | |
|--------|------|-----------------|---------|
| | | Fosc @ 5V, 25°C | |
| 20 pF | 4.7k | 4.52 MHz | ±17.35% |
| | 10k | 2.47 MHz | ±10.10% |
| | 100k | 290.86 kHz | ±11.90% |
| 100 pF | 3.3k | 1.92 MHz | ±9.43% |
| | 4.7k | 1.49 MHz | ±9.83% |
| | 10k | 788.77 kHz | ±10.92% |
| | 100k | 88.11 kHz | ±16.03% |
| 300 pF | 3.3k | 726.89 kHz | ±10.97% |
| | 4.7k | 573.95 kHz | ±10.14% |
| | 10k | 307.31 kHz | ±10.43% |
| | 100k | 33.82 kHz | ±11.24% |

The percentage variation indicated here is part to part variation due to normal process distribution. The variation indicated is ± 3 standard deviation from average value for VDD = 5V.

FIGURE 16-6: TYPICAL IPD vs. VDD
WATCHDOG TIMER ENABLED
25°C



Data based on matrix samples. See first page of this section for details.

FIGURE 16-7: MAXIMUM IPD vs. VDD
WATCHDOG DISABLED

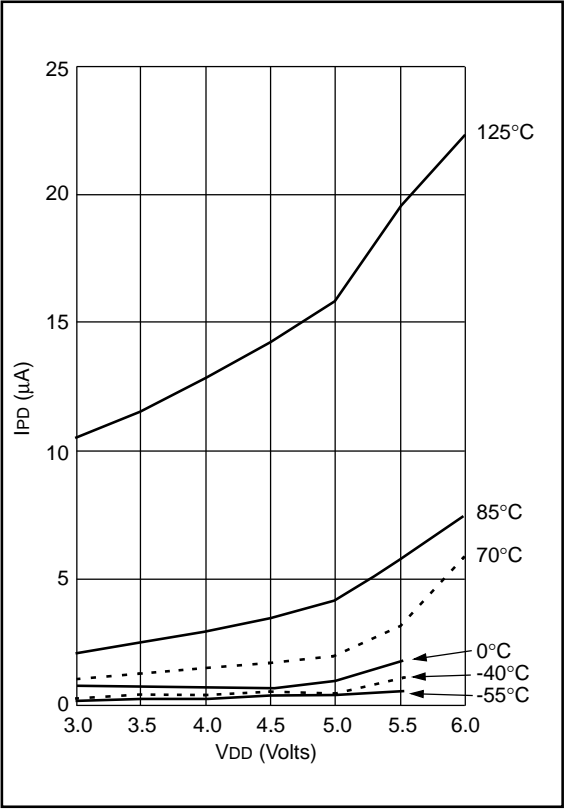


FIGURE 16-8: MAXIMUM IPD vs. VDD
WATCHDOG ENABLED

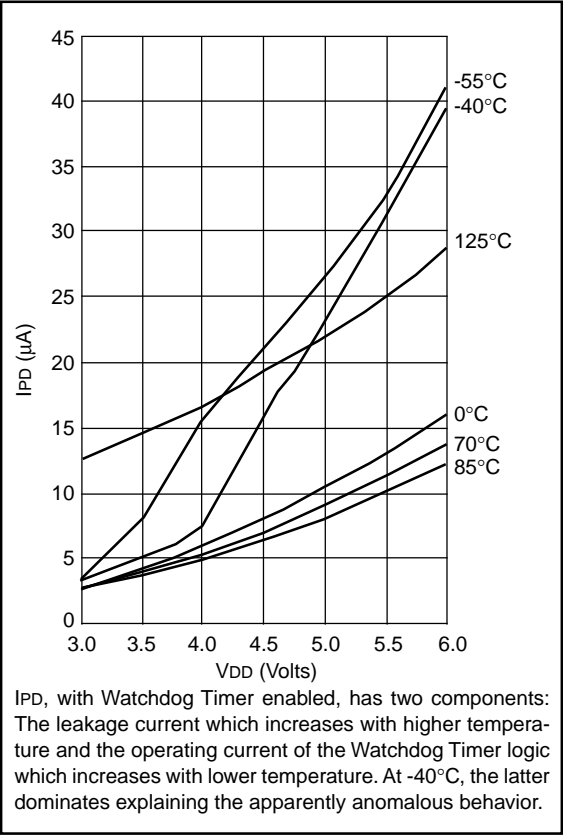
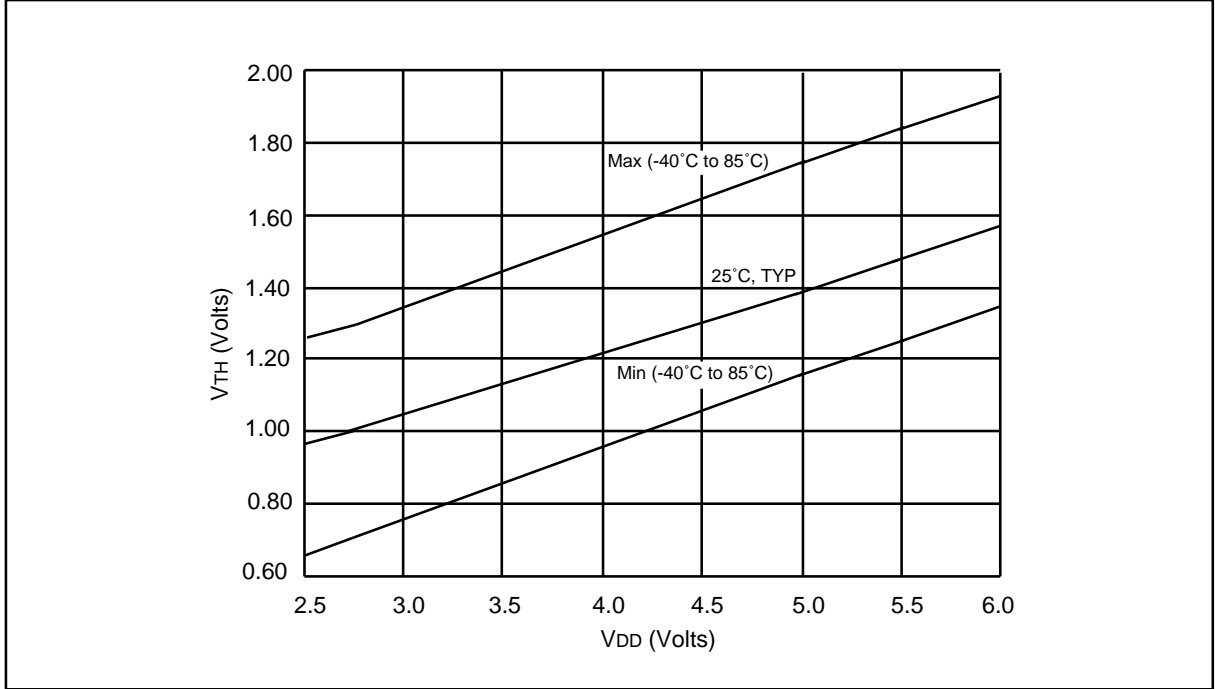


FIGURE 16-9: VTH (INPUT THRESHOLD VOLTAGE) OF I/O PINS vs. VDD



Data based on matrix samples. See first page of this section for details.

PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 16-10: V_{IH} , V_{IL} OF \overline{MCLR} , $T0CKI$ AND $OSC1$ (IN RC MODE) vs. V_{DD}

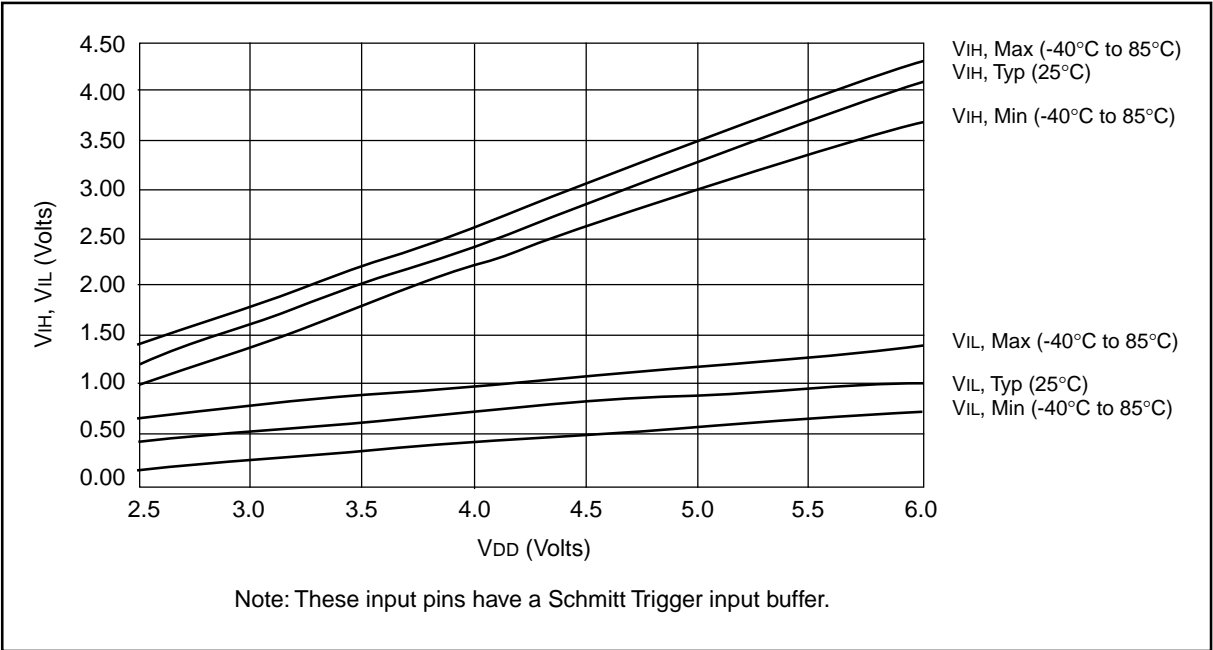
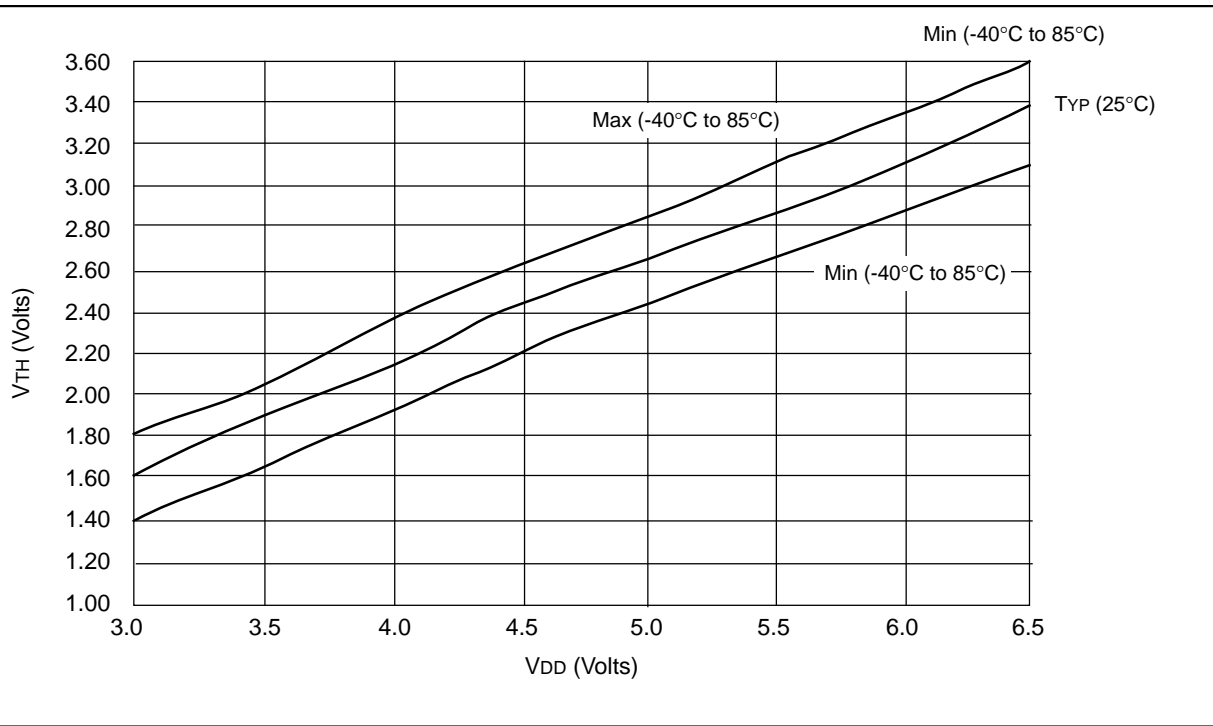


FIGURE 16-11: V_{TH} (INPUT THRESHOLD VOLTAGE) OF $OSC1$ INPUT (IN XT, HS, AND LP MODES) vs. V_{DD}



Data based on matrix samples. See first page of this section for details.

FIGURE 16-12: TYPICAL I_{DD} vs. FREQ (EXT CLOCK, 25°C)

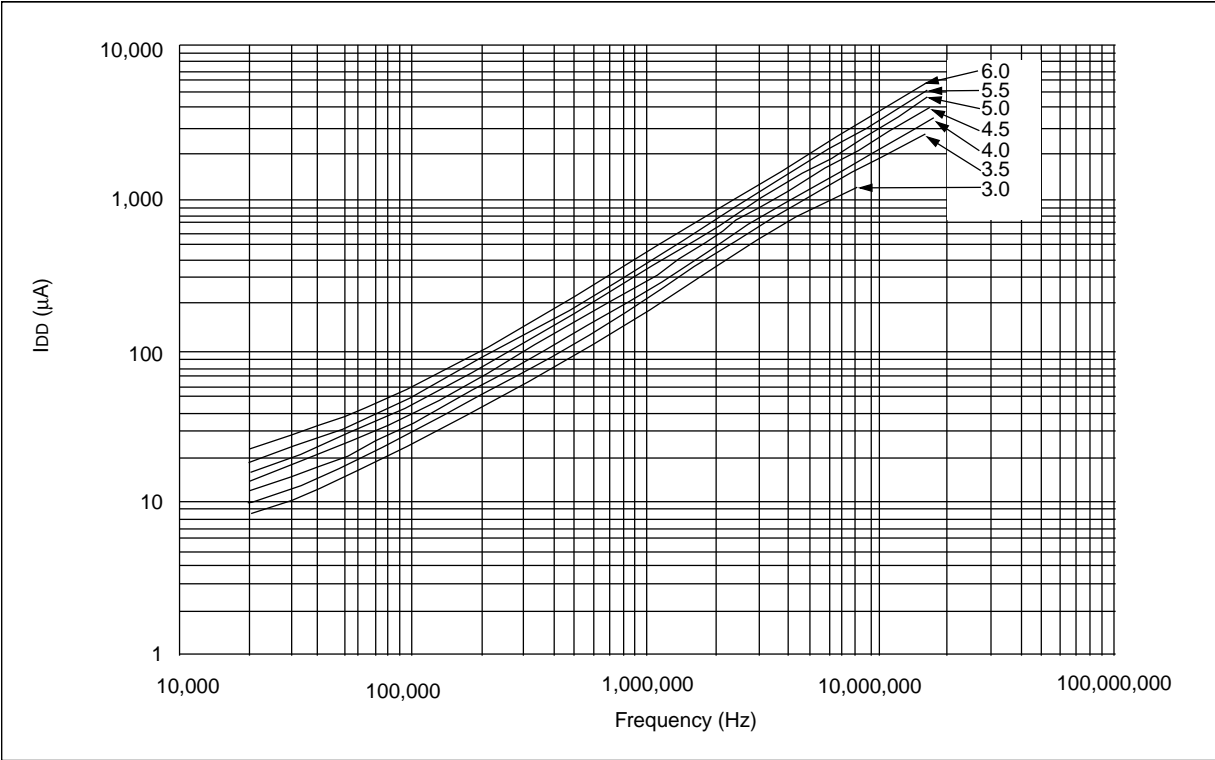
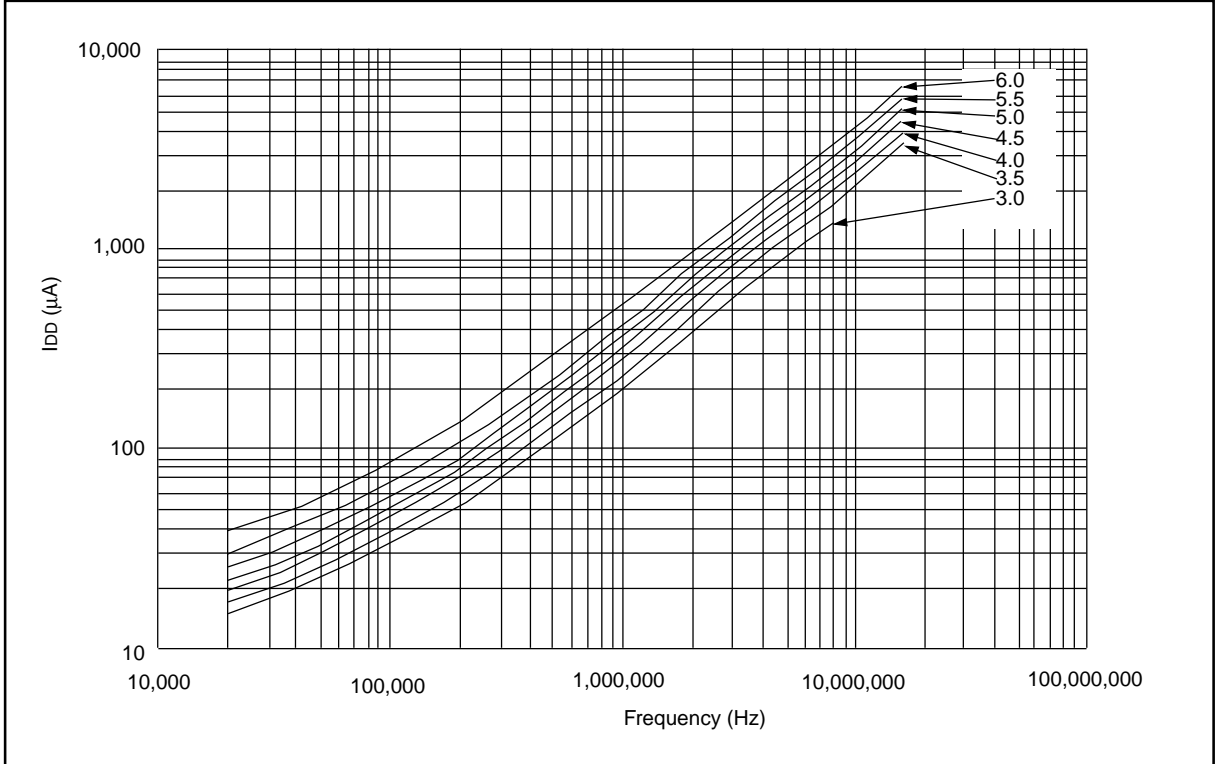


FIGURE 16-13: MAXIMUM I_{DD} vs. FREQ (EXT CLOCK, -40° TO +85°C)



Data based on matrix samples. See first page of this section for details.

PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 16-14: MAXIMUM I_{DD} vs. FREQ WITH A/D OFF (EXT CLOCK, -55° TO +125°C)

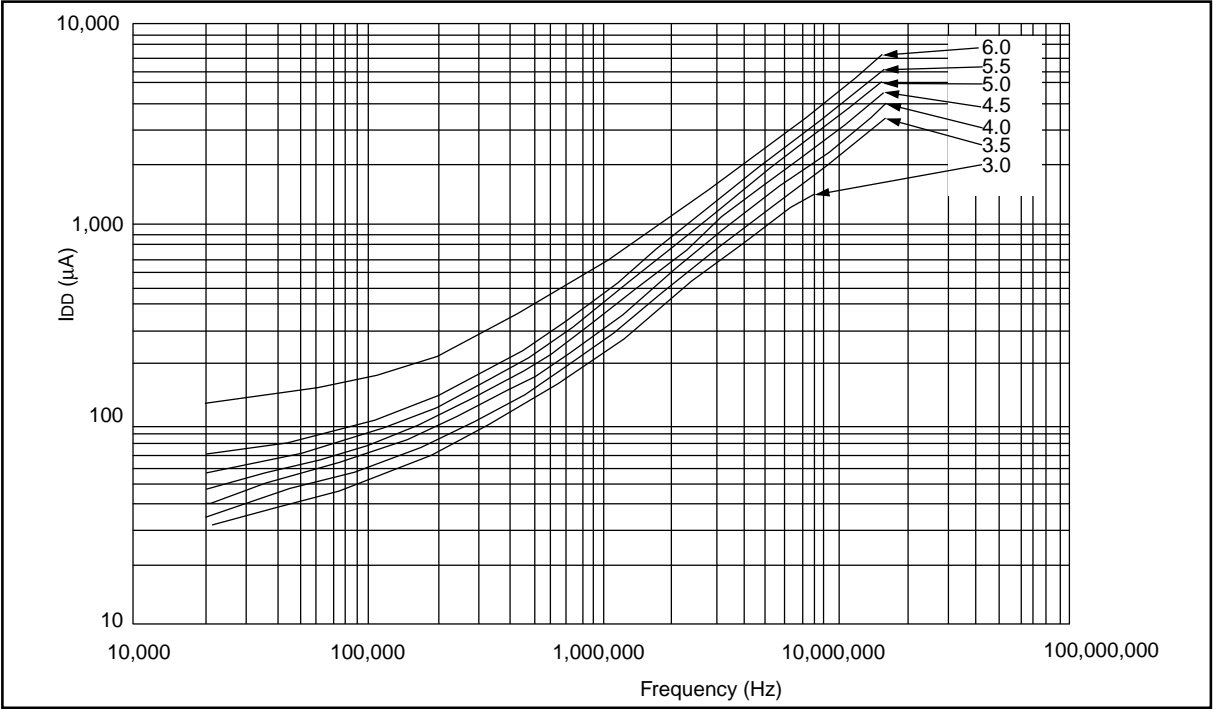


FIGURE 16-15: WDT TIMER TIME-OUT PERIOD vs. V_{DD}

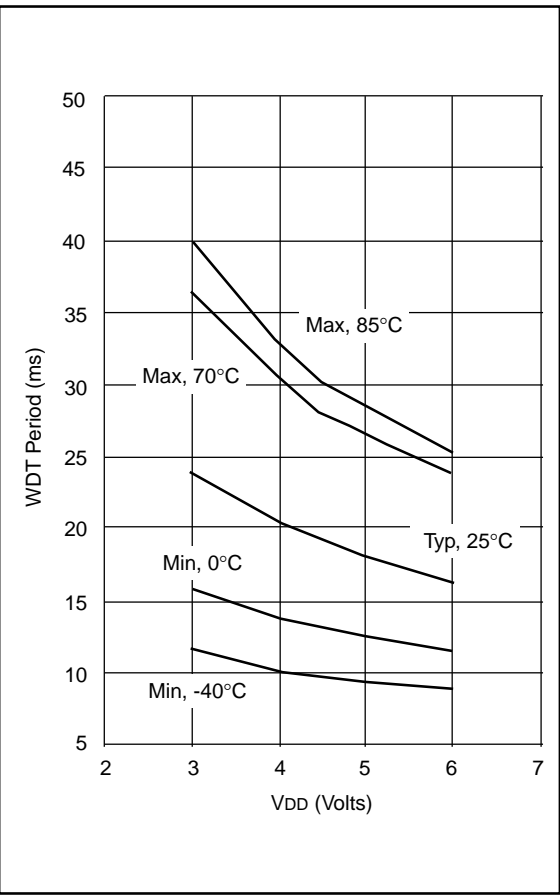
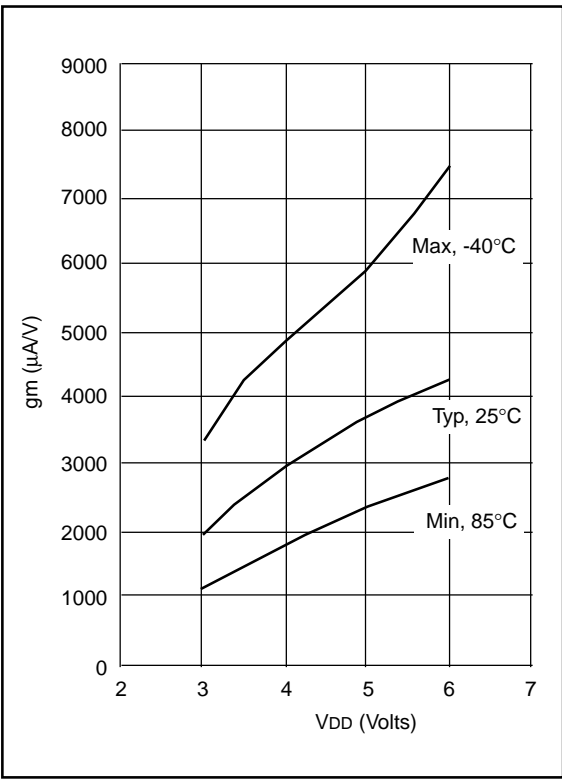


FIGURE 16-16: TRANSCONDUCTANCE (gm) OF HS OSCILLATOR vs. V_{DD}



Data based on matrix samples. See first page of this section for details.

FIGURE 16-17: TRANSCONDUCTANCE (gm) OF LP OSCILLATOR vs. VDD

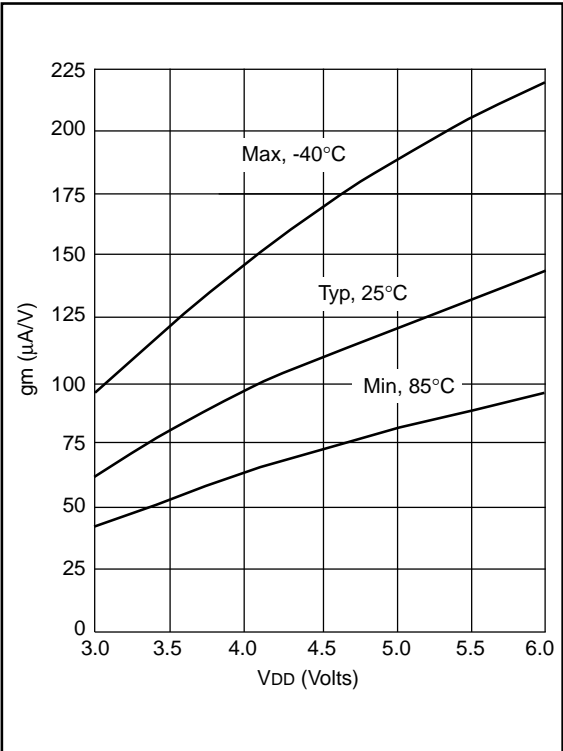


FIGURE 16-18: TRANSCONDUCTANCE (gm) OF XT OSCILLATOR vs. VDD

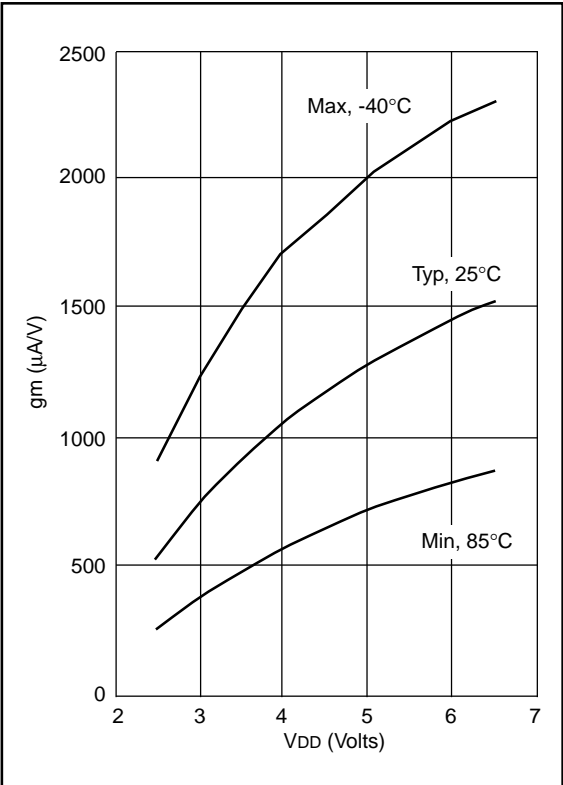


FIGURE 16-19: IOH vs. VOH, VDD = 3V

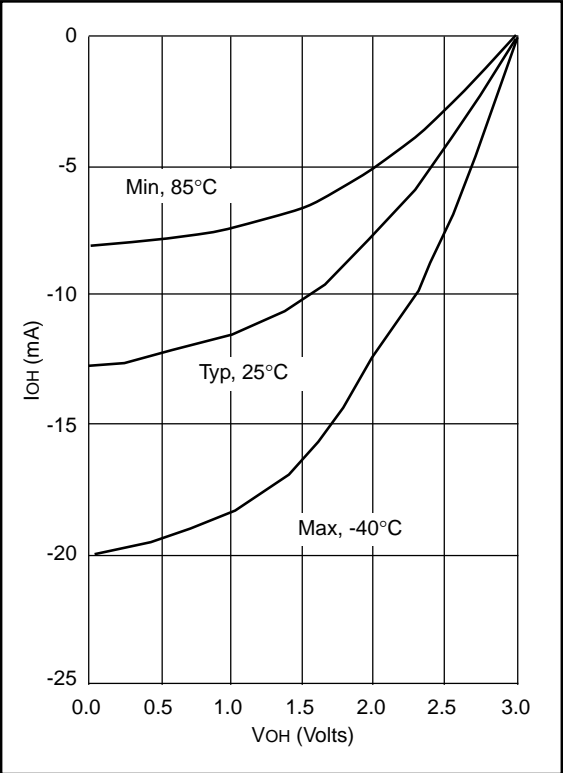
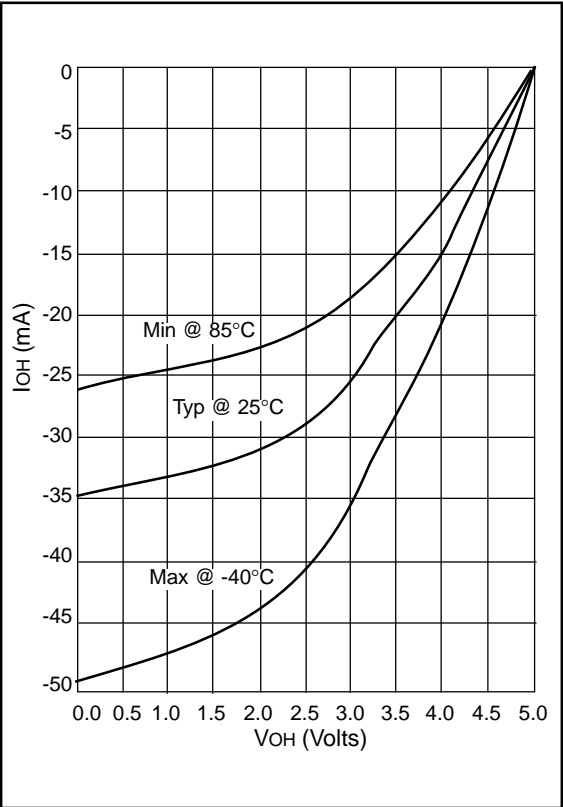


FIGURE 16-20: IOH vs. VOH, VDD = 5V



Data based on matrix samples. See first page of this section for details.

PIC16C71X

Applicable Devices 710 71 711 715

FIGURE 16-21: IOL vs. VOL, VDD = 3V

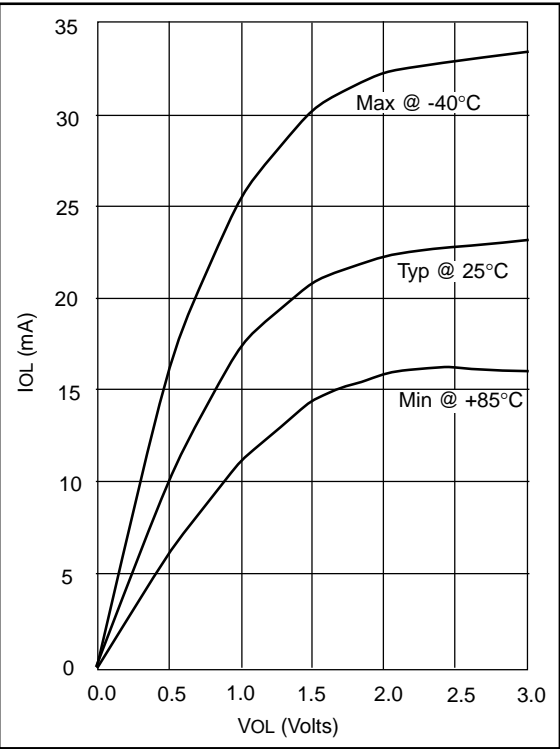
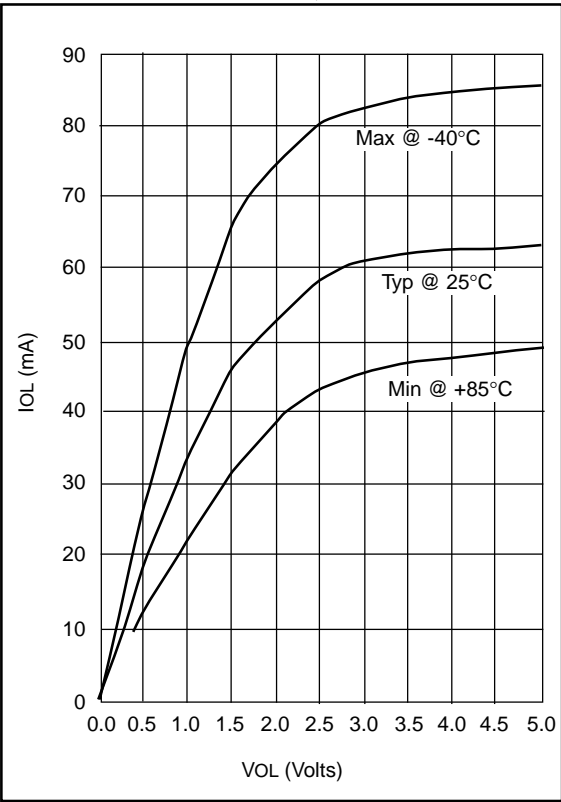


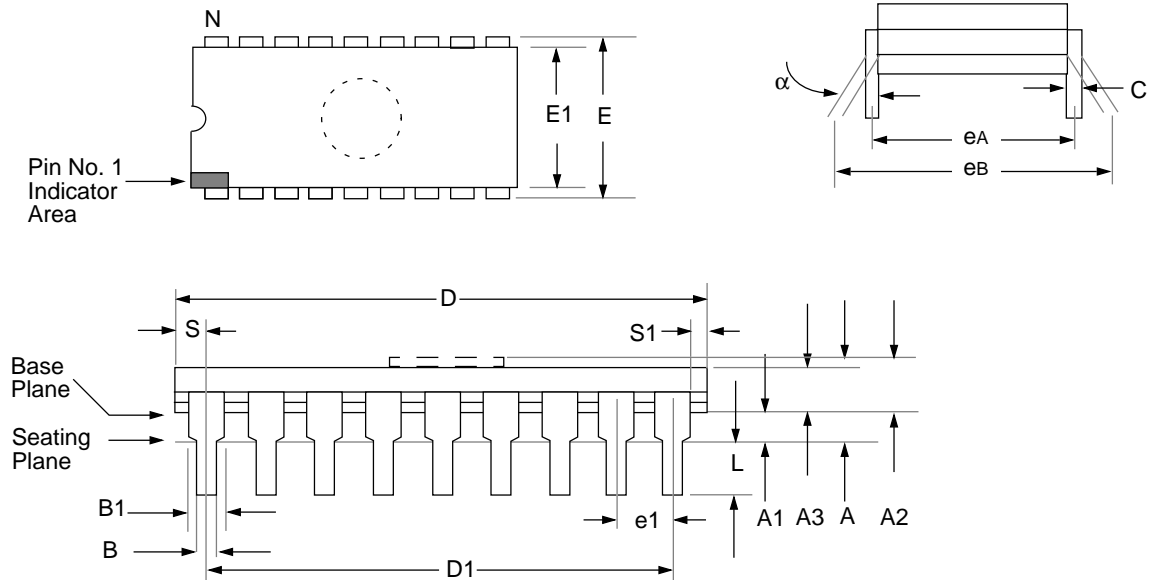
FIGURE 16-22: IOL vs. VOL, VDD = 5V



Data based on matrix samples. See first page of this section for details.

17.0 PACKAGING INFORMATION

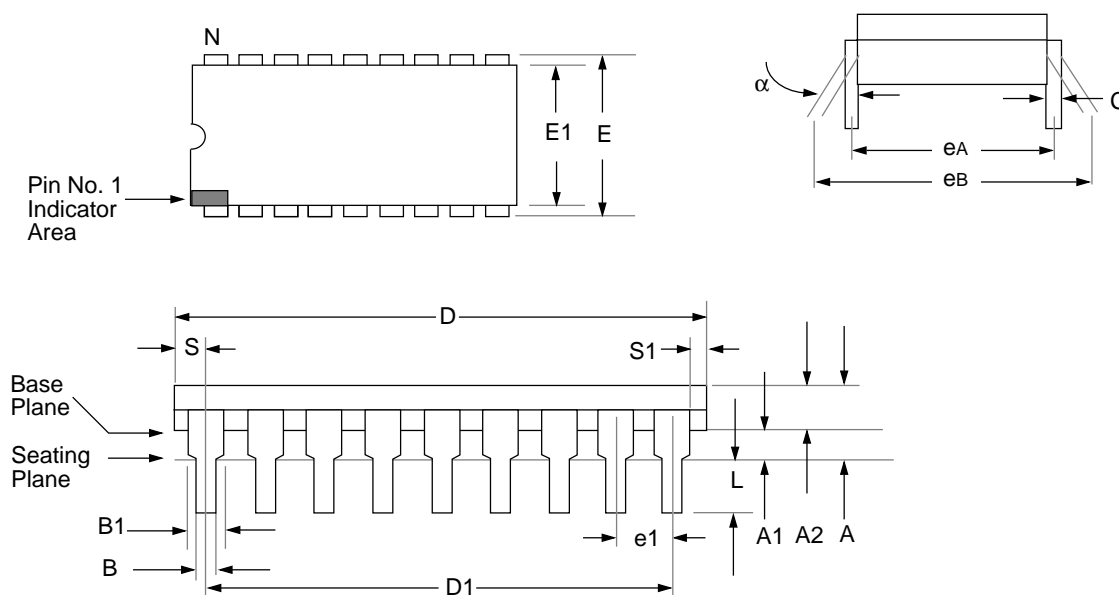
17.1 18-Lead Ceramic Cerdip Dual In-line with Window (300 mil) (JW)



| Package Group: Ceramic Cerdip Dual In-Line (CDP) | | | | | | |
|--|-------------|--------|-----------|--------|-------|-----------|
| Symbol | Millimeters | | | Inches | | |
| | Min | Max | Notes | Min | Max | Notes |
| α | 0° | 10° | | 0° | 10° | |
| A | — | 5.080 | | — | 0.200 | |
| A1 | 0.381 | 1.7780 | | 0.015 | 0.070 | |
| A2 | 3.810 | 4.699 | | 0.150 | 0.185 | |
| A3 | 3.810 | 4.445 | | 0.150 | 0.175 | |
| B | 0.355 | 0.585 | | 0.014 | 0.023 | |
| B1 | 1.270 | 1.651 | Typical | 0.050 | 0.065 | Typical |
| C | 0.203 | 0.381 | Typical | 0.008 | 0.015 | Typical |
| D | 22.352 | 23.622 | | 0.880 | 0.930 | |
| D1 | 20.320 | 20.320 | Reference | 0.800 | 0.800 | Reference |
| E | 7.620 | 8.382 | | 0.300 | 0.330 | |
| E1 | 5.588 | 7.874 | | 0.220 | 0.310 | |
| e1 | 2.540 | 2.540 | Reference | 0.100 | 0.100 | Reference |
| eA | 7.366 | 8.128 | Typical | 0.290 | 0.320 | Typical |
| eB | 7.620 | 10.160 | | 0.300 | 0.400 | |
| L | 3.175 | 3.810 | | 0.125 | 0.150 | |
| N | 18 | 18 | | 18 | 18 | |
| S | 0.508 | 1.397 | | 0.020 | 0.055 | |
| S1 | 0.381 | 1.270 | | 0.015 | 0.050 | |

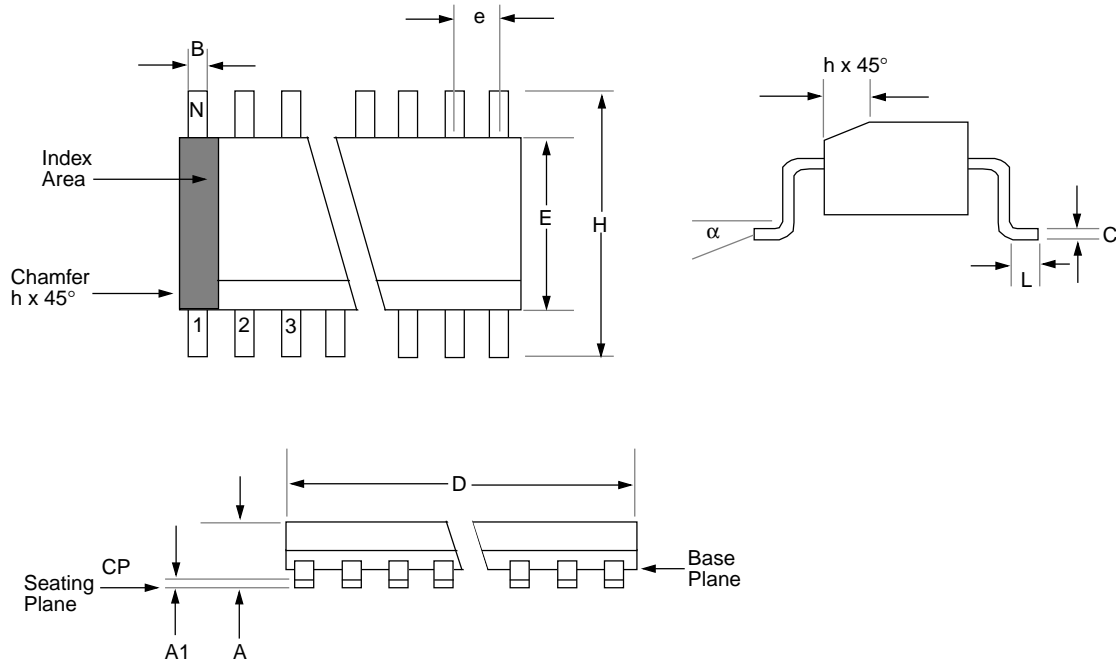
PIC16C71X

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



| Package Group: Plastic Dual In-Line (PLA) | | | | | | |
|---|-------------|--------|-----------|--------|-------|-----------|
| Symbol | Millimeters | | | Inches | | |
| | Min | Max | Notes | Min | Max | Notes |
| α | 0° | 10° | | 0° | 10° | |
| A | — | 4.064 | | — | 0.160 | |
| A1 | 0.381 | — | | 0.015 | — | |
| A2 | 3.048 | 3.810 | | 0.120 | 0.150 | |
| B | 0.355 | 0.559 | | 0.014 | 0.022 | |
| B1 | 1.524 | 1.524 | Reference | 0.060 | 0.060 | Reference |
| C | 0.203 | 0.381 | Typical | 0.008 | 0.015 | Typical |
| D | 22.479 | 23.495 | | 0.885 | 0.925 | |
| D1 | 20.320 | 20.320 | Reference | 0.800 | 0.800 | Reference |
| E | 7.620 | 8.255 | | 0.300 | 0.325 | |
| E1 | 6.096 | 7.112 | | 0.240 | 0.280 | |
| e1 | 2.489 | 2.591 | Typical | 0.098 | 0.102 | Typical |
| eA | 7.620 | 7.620 | Reference | 0.300 | 0.300 | Reference |
| eB | 7.874 | 9.906 | | 0.310 | 0.390 | |
| L | 3.048 | 3.556 | | 0.120 | 0.140 | |
| N | 18 | 18 | | 18 | 18 | |
| S | 0.889 | — | | 0.035 | — | |
| S1 | 0.127 | — | | 0.005 | — | |

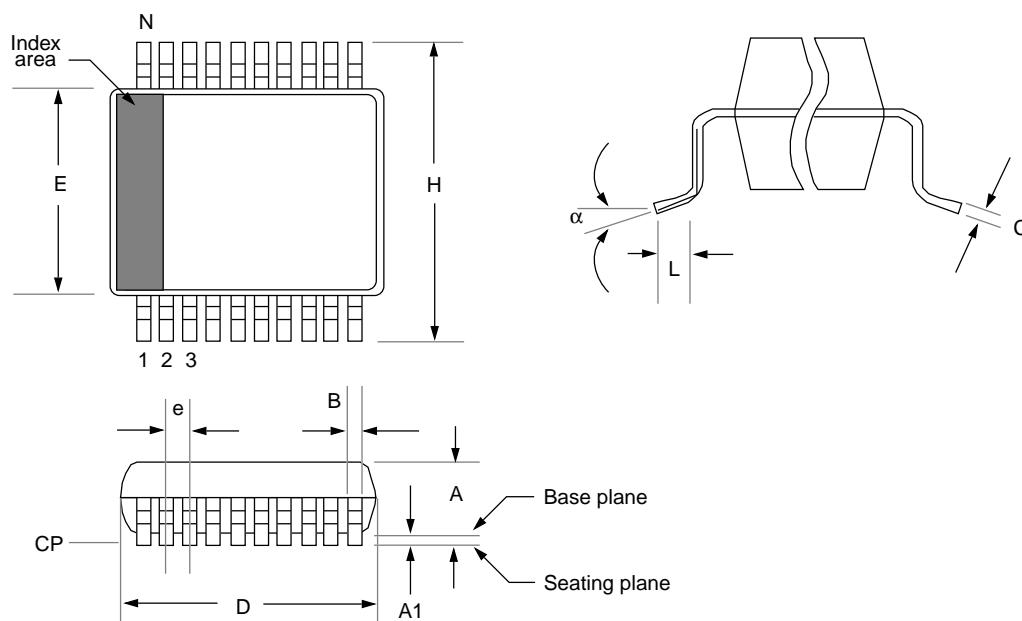
17.3 18-Lead Plastic Surface Mount (SOIC - Wide, 300 mil Body)(SO)



| Package Group: Plastic SOIC (SO) | | | | | | |
|----------------------------------|-------------|--------|-----------|--------|-------|-----------|
| Symbol | Millimeters | | | Inches | | |
| | Min | Max | Notes | Min | Max | Notes |
| α | 0° | 8° | | 0° | 8° | |
| A | 2.362 | 2.642 | | 0.093 | 0.104 | |
| A1 | 0.101 | 0.300 | | 0.004 | 0.012 | |
| B | 0.355 | 0.483 | | 0.014 | 0.019 | |
| C | 0.241 | 0.318 | | 0.009 | 0.013 | |
| D | 11.353 | 11.735 | | 0.447 | 0.462 | |
| E | 7.416 | 7.595 | | 0.292 | 0.299 | |
| e | 1.270 | 1.270 | Reference | 0.050 | 0.050 | Reference |
| H | 10.007 | 10.643 | | 0.394 | 0.419 | |
| h | 0.381 | 0.762 | | 0.015 | 0.030 | |
| L | 0.406 | 1.143 | | 0.016 | 0.045 | |
| N | 18 | 18 | | 18 | 18 | |
| CP | — | 0.102 | | — | 0.004 | |

PIC16C71X

17.4 20-Lead Plastic Surface Mount (SSOP - 209 mil Body 5.30 mm) (SS)



| Package Group: Plastic SSOP | | | | | | |
|-----------------------------|-------------|-------|-----------|--------|-------|-----------|
| Symbol | Millimeters | | | Inches | | |
| | Min | Max | Notes | Min | Max | Notes |
| α | 0° | 8° | | 0° | 8° | |
| A | 1.730 | 1.990 | | 0.068 | 0.078 | |
| A1 | 0.050 | 0.210 | | 0.002 | 0.008 | |
| B | 0.250 | 0.380 | | 0.010 | 0.015 | |
| C | 0.130 | 0.220 | | 0.005 | 0.009 | |
| D | 7.070 | 7.330 | | 0.278 | 0.289 | |
| E | 5.200 | 5.380 | | 0.205 | 0.212 | |
| e | 0.650 | 0.650 | Reference | 0.026 | 0.026 | Reference |
| H | 7.650 | 7.900 | | 0.301 | 0.311 | |
| L | 0.550 | 0.950 | | 0.022 | 0.037 | |
| N | 20 | 20 | | 20 | 20 | |
| CP | - | 0.102 | | - | 0.004 | |

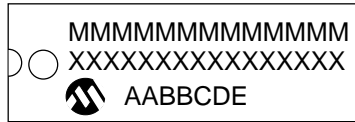
Note 1: Dimensions D1 and E1 do not include mold protrusion. Allowable mold protrusion is 0.25mm (0.010") per side. D1 and E1 dimensions including mold mismatch.

2: Dimension "b" does not include Dambar protrusion, allowable Dambar protrusion shall be 0.08mm (0.003") max.

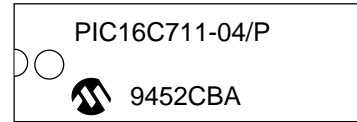
3: This outline conforms to JEDEC MS-026.

17.5 Package Marking Information

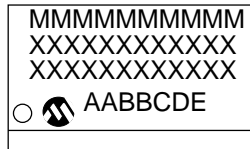
18-Lead PDIP



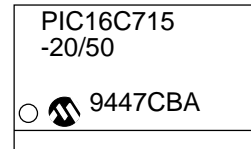
Example



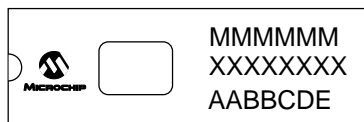
18-Lead SOIC



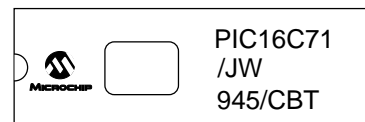
Example



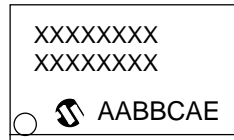
18-Lead CERDIP Windowed



Example



20-Lead SSOP



Example



| | | |
|---------|----------------|---|
| Legend: | MM...M | Microchip part number information |
| | XX...X | Customer specific information* |
| | AA | Year code (last 2 digits of calendar year) |
| | BB | Week code (week of January 1 is week '01') |
| | C | Facility code of the plant at which wafer is manufactured. C = Chandler, Arizona, U.S.A. S = Tempe, Arizona, U.S.A. |
| | D ₁ | Mask revision number for microcontroller |
| | E | Assembly code of the plant or country of origin in which part was assembled. |

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

* Standard OTP marking consists of Microchip part number, year code, week code, facility code, mask revision number, and assembly code. For OTP marking beyond this, certain price adders apply. Please check with your Microchip Sales Office. For QTP devices, any special marking adders are included in QTP price.

PIC16C71X

NOTES:

APPENDIX A:

The following are the list of modifications over the PIC16C5X microcontroller family:

1. Instruction word length is increased to 14-bits. This allows larger page sizes both in program memory (1K now as opposed to 512 before) and register file (68 bytes now versus 32 bytes before).
2. A PC high latch register (PCLATH) is added to handle program memory paging. Bits PA2, PA1, PA0 are removed from STATUS register.
3. Data memory paging is redefined slightly. STATUS register is modified.
4. Four new instructions have been added: RETURN, RETFIE, ADDLW, and SUBLW. Two instructions TRIS and OPTION are being phased out although they are kept for compatibility with PIC16C5X.
5. OPTION and TRIS registers are made addressable.
6. Interrupt capability is added. Interrupt vector is at 0004h.
7. Stack size is increased to 8 deep.
8. Reset vector is changed to 0000h.
9. Reset of all registers is revisited. Five different reset (and wake-up) types are recognized. Registers are reset differently.
10. Wake up from SLEEP through interrupt is added.
11. Two separate timers, Oscillator Start-up Timer (OST) and Power-up Timer (PWRT) are included for more reliable power-up. These timers are invoked selectively to avoid unnecessary delays on power-up and wake-up.
12. PORTB has weak pull-ups and interrupt on change feature.
13. T0CKI pin is also a port pin (RA4) now.
14. FSR is made a full eight bit register.
15. "In-circuit serial programming" is made possible. The user can program PIC16CXX devices using only five pins: VDD, VSS, MCLR/VPP, RB6 (clock) and RB7 (data in/out).
16. PCON status register is added with a Power-on Reset status bit ($\overline{\text{POR}}$).
17. Code protection scheme is enhanced such that portions of the program memory can be protected, while the remainder is unprotected.
18. Brown-out protection circuitry has been added. Controlled by configuration word bit BODEN. Brown-out reset ensures the device is placed in a reset condition if VDD dips below a fixed set-point.

APPENDIX B: COMPATIBILITY

To convert code written for PIC16C5X to PIC16CXX, the user should take the following steps:

1. Remove any program memory page select operations (PA2, PA1, PA0 bits) for CALL, GOTO.
2. Revisit any computed jump operations (write to PC or add to PC, etc.) to make sure page bits are set properly under the new scheme.
3. Eliminate any data memory page switching. Redefine data variables to reallocate them.
4. Verify all writes to STATUS, OPTION, and FSR registers since these have changed.
5. Change reset vector to 0000h.

PIC16C71X

APPENDIX C: WHAT'S NEW

1. Consolidated all pin compatible 18-pin A/D based devices into one data sheet.

APPENDIX D: WHAT'S CHANGED

1. Minor changes, spelling and grammatical changes.
2. Low voltage operation on the PIC16LC710/711/715 has been reduced from 3.0V to 2.5V.
3. Part numbers of the PIC16C70 and PIC16C71A have changed to PIC16C710 and PIC16C711, respectively.

INDEX

A

A/D

| | |
|--|--------------|
| Accuracy/Error | 44 |
| ADIF bit | 39 |
| Analog Input Model Block Diagram | 40 |
| Analog-to-Digital Converter | 37 |
| Configuring Analog Port Pins | 41 |
| Configuring the Interrupt | 39 |
| Configuring the Module | 39 |
| Connection Considerations | 44 |
| Conversion Clock | 41 |
| Conversion Time | 43 |
| Conversions | 42 |
| Converter Characteristics | 99, 122, 145 |
| Delays | 40 |
| Effects of a Reset | 44 |
| Equations | 40 |
| Faster Conversion - Lower Resolution Trade-off | 43 |
| Flowchart of A/D Operation | 45 |
| GO/DONE bit | 39 |
| Internal Sampling Switch (Rss) Impedence | 40 |
| Minimum Charging Time | 40 |
| Operation During Sleep | 44 |
| Sampling Requirements | 40 |
| Source Impedence | 40 |
| Time Delays | 40 |
| Transfer Function | 45 |
| Absolute Maximum Ratings | 89, 111, 135 |

AC Characteristics

| | |
|-----------------|-----|
| PIC16C710 | 101 |
| PIC16C711 | 101 |
| PIC16C715 | 125 |

| | |
|-----------------------|------------|
| ADCON0 Register | 37 |
| ADCON1 | 37 |
| ADCON1 Register | 14, 37 |
| ADCS0 bit | 37 |
| ADCS1 bit | 37 |
| ADIE bit | 19, 20 |
| ADIF bit | 21, 37 |
| ADON bit | 37 |
| ADRES Register | 15, 37, 39 |
| ALU | 7 |

Application Notes

| | |
|--|----|
| AN546 | 37 |
| AN552 | 27 |
| AN556 | 23 |
| AN607, Power-up Trouble Shooting | 53 |

Architecture

| | |
|-------------------|---|
| Harvard | 7 |
| Overview | 7 |
| von Neumann | 7 |

Assembler

| | |
|-----------------------|----|
| MPASM Assembler | 86 |
|-----------------------|----|

B

Block Diagrams

| | |
|-----------------------------|----|
| Analog Input Model | 40 |
| On-Chip Reset Circuit | 52 |
| PIC16C71X | 8 |
| RA3/RA0 Port Pins | 25 |
| RA4/T0CKI Pin | 25 |
| RB3:RB0 Port Pins | 27 |
| RB7:RB4 Pins | 28 |

| | |
|-----------------------------|--------|
| RB7:RB4 Port Pins | 28 |
| Timer0 | 31 |
| Timer0/WDT Prescaler | 34 |
| Watchdog Timer | 65 |
| BODEN bit | 48 |
| BOR bit | 22, 54 |
| Brown-out Reset (BOR) | 53 |

C

| | |
|--|--------|
| C bit | 17 |
| C16C71 | 47 |
| Carry bit | 7 |
| CHS0 bit | 37 |
| CHS1 bit | 37 |
| Clocking Scheme | 10 |
| Code Examples | |
| Call of a Subroutine in Page 1 from Page 0 | 24 |
| Changing Prescaler (Timer0 to WDT) | 35 |
| Changing Prescaler (WDT to Timer0) | 35 |
| Doing an A/D Conversion | 42 |
| I/O Programming | 30 |
| Indirect Addressing | 24 |
| Initializing PORTA | 25 |
| Initializing PORTB | 27 |
| Saving STATUS and W Registers in RAM | 64 |
| Code Protection | 47, 67 |
| Computed GOTO | 23 |
| Configuration Bits | 47 |
| CP0 bit | 47, 48 |
| CP1 bit | 48 |

D

| | |
|-------------------------------------|----------|
| DC bit | 17 |
| DC Characteristics | 147 |
| PIC16C71 | 136 |
| PIC16C710 | 90, 101 |
| PIC16C711 | 90, 101 |
| PIC16C715 | 113, 125 |
| Development Support | 3, 85 |
| Development Tools | 85 |
| Diagrams - See Block Diagrams | |
| Digit Carry bit | 7 |
| Direct Addressing | 24 |

E

Electrical Characteristics

| | |
|---|-----|
| PIC16C71 | 135 |
| PIC16C710 | 89 |
| PIC16C711 | 89 |
| PIC16C715 | 111 |
| External Brown-out Protection Circuit | 60 |
| External Power-on Reset Circuit | 60 |

F

Family of Devices

| | |
|---|------------|
| PIC16C71X | 4 |
| FOSC0 bit | 47, 48 |
| FOSC1 bit | 47, 48 |
| FSR Register | 15, 16, 24 |
| Fuzzy Logic Dev. System (fuzzyTECH®-MP) | 87 |

G

| | |
|---------------------------|--------|
| General Description | 3 |
| GIE bit | 19, 61 |
| GO/DONE bit | 37 |

PIC16C71X

I

| | |
|---|----------------|
| I/O Ports | |
| PORTA | 25 |
| PORTB | 27 |
| Section | 25 |
| I/O Programming Considerations | 30 |
| ICEPIC Low-Cost PIC16CXXX In-Circuit Emulator | 85 |
| In-Circuit Serial Programming | 47, 67 |
| INDF Register | 14, 16, 24 |
| Indirect Addressing | 24 |
| Instruction Cycle | 10 |
| Instruction Flow/Pipelining | 10 |
| Instruction Format | 69 |
| Instruction Set | |
| ADDLW | 71 |
| ADDWF | 71 |
| ANDLW | 71 |
| ANDWF | 71 |
| BCF | 72 |
| BSF | 72 |
| BTFS | 72 |
| BTFS | 73 |
| CALL | 73 |
| CLRF | 74 |
| CLRW | 74 |
| CLRWD | 74 |
| COMF | 75 |
| DECF | 75 |
| DECFSZ | 75 |
| GOTO | 76 |
| INCF | 76 |
| INCF | 77 |
| IORLW | 77 |
| IORWF | 78 |
| MOVF | 78 |
| MOVLW | 78 |
| MOVWF | 78 |
| NOP | 79 |
| OPTION | 79 |
| RETFIE | 79 |
| RETLW | 80 |
| RETURN | 80 |
| RLF | 81 |
| RRF | 81 |
| SLEEP | 82 |
| SUBLW | 82 |
| SUBWF | 83 |
| SWAPF | 83 |
| TRIS | 83 |
| XORLW | 84 |
| XORWF | 84 |
| Section | 69 |
| Summary Table | 70 |
| INT Interrupt | 63 |
| INTCON Register | 19 |
| INTE bit | 19 |
| INTEDG bit | 18, 63 |
| Internal Sampling Switch (Rss) Impedance | 40 |
| Interrupts | 47 |
| A/D | 61 |
| External | 61 |
| PORTB Change | 61 |
| PortB Change | 63 |
| RB7:RB4 Port Change | 27 |
| Section | 61 |
| TMR0 | 63 |
| TMR0 Overflow | 61 |
| INTF bit | 19 |
| IRP bit | 17 |
| K | |
| KeelQ [®] Evaluation and Programming Tools | 87 |
| L | |
| Loading of PC | 23 |
| LP | 54 |
| M | |
| MCLR | 52, 56 |
| Memory | |
| Data Memory | 12 |
| Program Memory | 11 |
| Register File Maps | |
| PIC16C71 | 12 |
| PIC16C710 | 12 |
| PIC16C711 | 13 |
| PIC16C715 | 13 |
| MP-DriveWay [™] - Application Code Generator | 87 |
| MPEEN bit | 22, 48 |
| MPLAB [™] C | 87 |
| MPLAB [™] Integrated Development Environment | |
| Software | 86 |
| O | |
| OPCODE | 69 |
| OPTION Register | 18 |
| Orthogonal | 7 |
| OSC selection | 47 |
| Oscillator | |
| HS | 49, 54 |
| LP | 49, 54 |
| RC | 49 |
| XT | 49, 54 |
| Oscillator Configurations | 49 |
| Oscillator Start-up Timer (OST) | 53 |
| P | |
| Packaging | |
| 18-Lead Cerdip w/Window | 155 |
| 18-Lead PDIP | 156 |
| 18-Lead SOIC | 157 |
| 20-Lead SSOP | 158 |
| Paging, Program Memory | 23 |
| PCL Register | 14, 15, 16, 23 |
| PCLATH | 57, 58 |
| PCLATH Register | 14, 15, 16, 23 |
| PCON Register | 22, 54 |
| PD bit | 17, 52, 55 |
| PER bit | 22 |
| PIC16C71 | 147 |
| AC Characteristics | 147 |
| PICDEM-1 Low-Cost PIC16/17 Demo Board | 86 |
| PICDEM-2 Low-Cost PIC16CXX Demo Board | 86 |
| PICDEM-3 Low-Cost PIC16CXXX Demo Board | 86 |
| PICMASTER [®] In-Circuit Emulator | 85 |
| PICSTART [®] Plus Entry Level Development System | 85 |
| PIE1 Register | 20 |
| Pin Functions | |
| MCLR/VPP | 9 |
| OSC1/CLKIN | 9 |
| OSC2/CLKOUT | 9 |
| RA0/AN0 | 9 |
| RA1/AN1 | 9 |

PIC16C71X

| | | | |
|---|----------------|--|------------------|
| RA2/AN2 | 9 | PIC16C711 | 13 |
| RA3/AN3/VREF | 9 | PIC16C715 | 13 |
| RA4/T0CKI | 9 | Reset Conditions | 56 |
| RB0/INT | 9 | Summary | 14-?? |
| RB1 | 9 | Reset | 47, 52 |
| RB2 | 9 | Reset Conditions for Special Registers | 56 |
| RB3 | 9 | RP0 bit | 12, 17 |
| RB4 | 9 | RP1 bit | 17 |
| RB5 | 9 | S | |
| RB6 | 9 | SEEVAL® Evaluation and Programming System | 87 |
| RB7 | 9 | Services | |
| VDD | 9 | One-Time-Programmable (OTP) Devices | 5 |
| VSS | 9 | Quick-Turnaround-Production (QTP) Devices | 5 |
| Pinout Descriptions | | Serialized Quick-Turnaround Production (SQTP) Devices | 5 |
| PIC16C71 | 9 | SLEEP | 47, 52 |
| PIC16C710 | 9 | Software Simulator (MPLAB™ SIM) | 87 |
| PIC16C711 | 9 | Special Features of the CPU | 47 |
| PIC16C715 | 9 | Special Function Registers | |
| PIR1 Register | 21 | PIC16C71 | 14 |
| POP | 23 | PIC16C710 | 14 |
| POR | 53, 54 | PIC16C711 | 14 |
| Oscillator Start-up Timer (OST) | 47, 53 | Special Function Registers, Section | 14 |
| Power Control Register (PCON) | 54 | Stack | 23 |
| Power-on Reset (POR) | 47, 53, 57, 58 | Overflows | 23 |
| Power-up Timer (PWRT) | 47, 53 | Underflow | 23 |
| Time-out Sequence | 54 | STATUS Register | 17 |
| Time-out Sequence on Power-up | 59 | T | |
| TO | 52, 55 | T0CS bit | 18 |
| T0 bit | 22, 54 | T0IE bit | 19 |
| Port RB Interrupt | 63 | T0IF bit | 19 |
| PORTA | 57, 58 | TAD | 41 |
| PORTA Register | 14, 15, 25 | Timer0 | |
| PORTB | 57, 58 | RTCC | 57, 58 |
| PORTB Register | 14, 15, 27 | Timers | |
| Power-down Mode (SLEEP) | 66 | Timer0 | |
| Prescaler, Switching Between Timer0 and WDT | 35 | Block Diagram | 31 |
| PRO MATE® II Universal Programmer | 85 | External Clock | 33 |
| Program Branches | 7 | External Clock Timing | 33 |
| Program Memory | | Increment Delay | 33 |
| Paging | 23 | Interrupt | 31 |
| Program Memory Maps | | Interrupt Timing | 32 |
| PIC16C71 | 11 | Prescaler | 34 |
| PIC16C710 | 11 | Prescaler Block Diagram | 34 |
| PIC16C711 | 11 | Section | 31 |
| PIC16C715 | 11 | Switching Prescaler Assignment | 35 |
| Program Verification | 67 | Synchronization | 33 |
| PS0 bit | 18 | T0CKI | 33 |
| PS1 bit | 18 | T0IF | 63 |
| PS2 bit | 18 | Timing | 31 |
| PSA bit | 18 | TMR0 Interrupt | 63 |
| PUSH | 23 | Timing Diagrams | |
| PWRT | | A/D Conversion | 100, 124, 146 |
| Power-up Timer (PWRT) | 53 | Brown-out Reset | 53, 97 |
| PWRTE bit | 47, 48 | CLKOUT and I/O | 96, 119, 142 |
| R | | External Clock Timing | 95, 118, 141 |
| RBIE bit | 19 | Power-up Timer | 97, 143 |
| RBIF bit | 19, 27, 63 | Reset | 97, 143 |
| RBPU bit | 18 | Start-up Timer | 97, 143 |
| RC | 54 | Time-out Sequence | 59 |
| RC Oscillator | 51, 54 | Timer0 | 31, 98, 121, 144 |
| Read-Modify-Write | 30 | Timer0 Interrupt Timing | 32 |
| Register File | 12 | Timer0 with External Clock | 33 |
| Registers | | Wake-up from SLEEP through Interrupt | 67 |
| Maps | | Watchdog Timer | 97, 143 |
| PIC16C71 | 12 | | |
| PIC16C710 | 12 | | |

PIC16C71X

| | |
|----------------------------|------------|
| T \overline{O} bit | 17 |
| TOSE bit | 18 |
| TRISA Register | 14, 16, 25 |
| TRISB Register | 14, 16, 27 |
| Two's Complement | 7 |

U

| | |
|----------------------------|---|
| Upward Compatibility | 3 |
| UV Erasable Devices | 5 |

W

| | |
|----------------------------------|----------------|
| W Register | |
| ALU | 7 |
| Wake-up from SLEEP | 66 |
| Watchdog Timer (WDT) | 47, 52, 56, 65 |
| WDT | 56 |
| Block Diagram | 65 |
| Programming Considerations | 65 |
| Timeout | 57, 58 |
| WDT Period | 65 |
| WDTE bit | 47, 48 |

Z

| | |
|----------------|----|
| Z bit | 17 |
| Zero bit | 7 |

LIST OF EXAMPLES

| | | |
|---------------|--|----|
| Example 3-1: | Instruction Pipeline Flow | 10 |
| Example 4-1: | Call of a Subroutine in Page 1 from Page 0 | 24 |
| Example 4-2: | Indirect Addressing | 24 |
| Example 5-1: | Initializing PORTA | 25 |
| Example 5-2: | Initializing PORTB | 27 |
| Example 5-3: | Read-Modify-Write Instructions on an I/O Port | 30 |
| Example 6-1: | Changing Prescaler (Timer0→WDT) | 35 |
| Example 6-2: | Changing Prescaler (WDT→Timer0) | 35 |
| Equation 7-1: | A/D Minimum Charging Time | 40 |
| Example 7-1: | Calculating the Minimum Required Acquisition Time | 40 |
| Example 7-2: | A/D Conversion | 42 |
| Example 7-3: | 4-bit vs. 8-bit Conversion Times | 43 |
| Example 8-1: | Saving STATUS and W Registers in RAM | 64 |

LIST OF FIGURES

| | | |
|--------------|--|----|
| Figure 3-1: | PIC16C71X Block Diagram | 8 |
| Figure 3-2: | Clock/Instruction Cycle | 10 |
| Figure 4-1: | PIC16C710 Program Memory Map and Stack | 11 |
| Figure 4-2: | PIC16C711 Program Memory Map and Stack | 11 |
| Figure 4-3: | PIC16C715 Program Memory Map and Stack | 11 |
| Figure 4-4: | PIC16C710/71 Register File Map | 12 |
| Figure 4-5: | PIC16C711 Register File Map | 13 |
| Figure 4-6: | PIC16C715 Register File Map | 13 |
| Figure 4-7: | Status Register (Address 03h, 83h) | 17 |
| Figure 4-8: | OPTION Register (Address 81h, 181h) | 18 |
| Figure 4-9: | INTCON Register (Address 0Bh, 8Bh) | 19 |
| Figure 4-10: | PIE1 Register (Address 8Ch) | 20 |
| Figure 4-11: | PIR1 Register (Address 0Ch) | 21 |
| Figure 4-12: | PCON Register (Address 8Eh), PIC16C710/711 | 22 |
| Figure 4-13: | PCON Register (Address 8Eh), PIC16C715 | 22 |
| Figure 4-14: | Loading of PC In Different Situations | 23 |
| Figure 4-15: | Direct/Indirect Addressing | 24 |
| Figure 5-1: | Block Diagram of RA3:RA0 Pins | 25 |
| Figure 5-2: | Block Diagram of RA4/T0CKI Pin | 25 |
| Figure 5-3: | Block Diagram of RB3:RB0 Pins | 27 |
| Figure 5-4: | Block Diagram of RB7:RB4 Pins (PIC16C71) | 28 |
| Figure 5-5: | Block Diagram of RB7:RB4 Pins (PIC16C710/711/715) | 28 |
| Figure 5-6: | Successive I/O Operation | 30 |
| Figure 6-1: | Timer0 Block Diagram | 31 |
| Figure 6-2: | Timer0 Timing: Internal Clock/ No Prescale | 31 |
| Figure 6-3: | Timer0 Timing: Internal Clock/ Prescale 1:2 | 32 |
| Figure 6-4: | Timer0 Interrupt Timing | 32 |
| Figure 6-5: | Timer0 Timing with External Clock | 33 |
| Figure 6-6: | Block Diagram of the Timer0/ WDT Prescaler | 34 |
| Figure 7-1: | ADCON0 Register (Address 08h), PIC16C710/71/711 | 37 |
| Figure 7-2: | ADCON0 Register (Address 1Fh), PIC16C715 | 38 |

| | | | | | |
|--------------|--|-----|---------------|--|-----|
| Figure 7-3: | ADCON1 Register, PIC16C710/71/711 (Address 88h), PIC16C715 (Address 9Fh)..... | 38 | Figure 12-9: | Maximum IPD vs. VDD Brown-out Detect Enabled (85°C to -40°C, RC Mode)..... | 103 |
| Figure 7-4: | A/D Block Diagram..... | 39 | Figure 12-10: | Typical IPD vs. Timer1 Enabled (32 kHz, RC0/RC1 = 33 pF/33 pF, RC Mode) | 103 |
| Figure 7-5: | Analog Input Model | 40 | Figure 12-11: | Maximum IPD vs. Timer1 Enabled (32 kHz, RC0/RC1 = 33 pF/33 pF, 85°C to -40°C, RC Mode) | 103 |
| Figure 7-6: | A/D Transfer Function..... | 45 | Figure 12-12: | Typical IDD vs. Frequency (RC Mode @ 22 pF, 25°C) | 104 |
| Figure 7-7: | Flowchart of A/D Operation..... | 45 | Figure 12-13: | Maximum IDD vs. Frequency (RC Mode @ 22 pF, -40°C to 85°C) | 104 |
| Figure 8-1: | Configuration Word for PIC16C71 | 47 | Figure 12-14: | Typical IDD vs. Frequency (RC Mode @ 100 pF, 25°C) | 105 |
| Figure 8-2: | Configuration Word, PIC16C710/711..... | 48 | Figure 12-15: | Maximum IDD vs. Frequency (RC Mode @ 100 pF, -40°C to 85°C) | 105 |
| Figure 8-3: | Configuration Word, PIC16C715..... | 48 | Figure 12-16: | Typical IDD vs. Frequency (RC Mode @ 300 pF, 25°C) | 106 |
| Figure 8-4: | Crystal/Ceramic Resonator Operation (HS, XT or LP OSC Configuration) | 49 | Figure 12-17: | Maximum IDD vs. Frequency (RC Mode @ 300 pF, -40°C to 85°C) | 106 |
| Figure 8-5: | External Clock Input Operation (HS, XT or LP OSC Configuration) | 49 | Figure 12-18: | Typical IDD vs. Capacitance @ 500 kHz (RC Mode) | 107 |
| Figure 8-6: | External Parallel Resonant Crystal Oscillator Circuit..... | 51 | Figure 12-19: | Transconductance(gm) of HS Oscillator vs. VDD..... | 107 |
| Figure 8-7: | External Series Resonant Crystal Oscillator Circuit..... | 51 | Figure 12-20: | Transconductance(gm) of LP Oscillator vs. VDD | 107 |
| Figure 8-8: | RC Oscillator Mode | 51 | Figure 12-21: | Transconductance(gm) of XT Oscillator vs. VDD | 107 |
| Figure 8-9: | Simplified Block Diagram of On-chip Reset Circuit..... | 52 | Figure 12-22: | Typical XTAL Startup Time vs. VDD (LP Mode, 25°C) | 108 |
| Figure 8-10: | Brown-out Situations | 53 | Figure 12-23: | Typical XTAL Startup Time vs. VDD (HS Mode, 25°C)..... | 108 |
| Figure 8-11: | Time-out Sequence on Power-up (MCLR not Tied to VDD): Case 1..... | 59 | Figure 12-24: | Typical XTAL Startup Time vs. VDD (XT Mode, 25°C) | 108 |
| Figure 8-12: | Time-out Sequence on Power-up (MCLR Not Tied To VDD): Case 2..... | 59 | Figure 12-25: | Typical IDD vs. Frequency (LP Mode, 25°C)..... | 109 |
| Figure 8-13: | Time-out Sequence on Power-up (MCLR Tied to VDD) | 59 | Figure 12-26: | Maximum IDD vs. Frequency (LP Mode, 85°C to -40°C) | 109 |
| Figure 8-14: | External Power-on Reset Circuit (for Slow VDD Power-up)..... | 60 | Figure 12-27: | Typical IDD vs. Frequency (XT Mode, 25°C)..... | 109 |
| Figure 8-15: | External Brown-out Protection Circuit 1 | 60 | Figure 12-28: | Maximum IDD vs. Frequency (XT Mode, -40°C to 85°C) | 109 |
| Figure 8-16: | External Brown-out Protection Circuit 2 | 60 | Figure 12-29: | Typical IDD vs. Frequency (HS Mode, 25°C) | 110 |
| Figure 8-17: | Interrupt Logic, PIC16C710, 71, 711..... | 62 | Figure 12-30: | Maximum IDD vs. Frequency (HS Mode, -40°C to 85°C) | 110 |
| Figure 8-18: | Interrupt Logic, PIC16C715..... | 62 | Figure 13-1: | Load Conditions..... | 117 |
| Figure 8-19: | INT Pin Interrupt Timing..... | 63 | Figure 13-2: | External Clock Timing..... | 118 |
| Figure 8-20: | Watchdog Timer Block Diagram | 65 | Figure 13-3: | CLKOUT and I/O Timing..... | 119 |
| Figure 8-21: | Summary of Watchdog Timer Registers ... | 65 | Figure 13-4: | Reset, Watchdog Timer, Oscillator Start-Up Timer, and Power-Up Timer Timing | 120 |
| Figure 8-22: | Wake-up from Sleep Through Interrupt.... | 67 | Figure 13-5: | Brown-out Reset Timing | 120 |
| Figure 8-23: | Typical In-Circuit Serial Programming Connection | 67 | Figure 13-6: | Timer0 Clock Timings..... | 121 |
| Figure 9-1: | General Format for Instructions | 69 | Figure 13-7: | A/D Conversion Timing..... | 124 |
| Figure 11-1: | Load Conditions | 94 | Figure 14-1: | Typical IPD vs. VDD (WDT Disabled, RC Mode) | 125 |
| Figure 11-2: | External Clock Timing | 95 | Figure 14-2: | Maximum IPD vs. VDD (WDT Disabled, RC Mode) | 125 |
| Figure 11-3: | CLKOUT and I/O Timing | 96 | Figure 14-3: | Typical IPD vs. VDD @ 25°C (WDT Enabled, RC Mode)..... | 126 |
| Figure 11-4: | Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing | 97 | Figure 14-4: | Maximum IPD vs. VDD (WDT Enabled, RC Mode)..... | 126 |
| Figure 11-5: | Brown-out Reset Timing..... | 97 | Figure 14-5: | Typical RC Oscillator Frequency vs. VDD | 126 |
| Figure 11-6: | Timer0 External Clock Timings | 98 | | | |
| Figure 11-7: | A/D Conversion Timing | 100 | | | |
| Figure 12-1: | Typical IPD vs. VDD (WDT Disabled, RC Mode) | 101 | | | |
| Figure 12-2: | Maximum IPD vs. VDD (WDT Disabled, RC Mode) | 101 | | | |
| Figure 12-3: | Typical IPD vs. VDD @ 25°C (WDT Enabled, RC Mode) | 102 | | | |
| Figure 12-4: | Maximum IPD vs. VDD (WDT Enabled, RC Mode) | 102 | | | |
| Figure 12-5: | Typical RC Oscillator Frequency vs. VDD | 102 | | | |
| Figure 12-6: | Typical RC Oscillator Frequency vs. VDD | 102 | | | |
| Figure 12-7: | Typical RC Oscillator Frequency vs. VDD | 102 | | | |
| Figure 12-8: | Typical IPD vs. VDD Brown-out Detect Enabled (RC Mode) | 103 | | | |

PIC16C71X

| | | | | | |
|---------------|---|-----|---------------|---|-----|
| Figure 14-6: | Typical RC Oscillator Frequency vs. VDD..... | 126 | Figure 16-4: | Typical RC Oscillator Frequency vs. VDD | 148 |
| Figure 14-7: | Typical RC Oscillator Frequency vs. VDD..... | 126 | Figure 16-5: | Typical I _{pd} vs. VDD Watchdog Timer Disabled 25°C..... | 148 |
| Figure 14-8: | Typical I _{PD} vs. VDD Brown-out Detect Enabled (RC Mode) | 127 | Figure 16-6: | Typical I _{pd} vs. VDD Watchdog Timer Enabled 25°C..... | 148 |
| Figure 14-9: | Maximum I _{PD} vs. VDD Brown-out Detect Enabled (85°C to -40°C, RC Mode) | 127 | Figure 16-7: | Maximum I _{pd} vs. VDD Watchdog Disabled..... | 149 |
| Figure 14-10: | Typical I _{PD} vs. Timer1 Enabled (32 kHz, RC0/RC1 = 33 pF/33 pF, RC Mode)..... | 127 | Figure 16-8: | Maximum I _{pd} vs. VDD Watchdog Enabled..... | 149 |
| Figure 14-11: | Maximum I _{PD} vs. Timer1 Enabled (32 kHz, RC0/RC1 = 33 pF/33 pF, 85°C to -40°C, RC Mode)..... | 127 | Figure 16-9: | V _{th} (Input Threshold Voltage) of I/O Pins vs. VDD..... | 149 |
| Figure 14-12: | Typical I _{DD} vs. Frequency (RC Mode @ 22 pF, 25°C)..... | 128 | Figure 16-10: | V _{IH} , V _{IL} of $\overline{\text{MCLR}}$, T0CKI and OSC1 (in RC Mode) vs. VDD | 150 |
| Figure 14-13: | Maximum I _{DD} vs. Frequency (RC Mode @ 22 pF, -40°C to 85°C)..... | 128 | Figure 16-11: | V _{TH} (Input Threshold Voltage) of OSC1 Input (in XT, HS, and LP Modes) vs. VDD | 150 |
| Figure 14-14: | Typical I _{DD} vs. Frequency (RC Mode @ 100 pF, 25°C)..... | 129 | Figure 16-12: | Typical I _{DD} vs. Freq (Ext Clock, 25°C).... | 151 |
| Figure 14-15: | Maximum I _{DD} vs. Frequency (RC Mode @ 100 pF, -40°C to 85°C)..... | 129 | Figure 16-13: | Maximum, I _{DD} vs. Freq (Ext Clock, -40° to +85°C)..... | 151 |
| Figure 14-16: | Typical I _{DD} vs. Frequency (RC Mode @ 300 pF, 25°C)..... | 130 | Figure 16-14: | Maximum I _{DD} vs. Freq with A/D Off (Ext Clock, -55° to +125°C) | 152 |
| Figure 14-17: | Maximum I _{DD} vs. Frequency (RC Mode @ 300 pF, -40°C to 85°C)..... | 130 | Figure 16-15: | WDT Timer Time-out Period vs. VDD..... | 152 |
| Figure 14-18: | Typical I _{DD} vs. Capacitance @ 500 kHz (RC Mode)..... | 131 | Figure 16-16: | Transconductance (gm) of HS Oscillator vs. VDD..... | 152 |
| Figure 14-19: | Transconductance(gm) of HS Oscillator vs. VDD | 131 | Figure 16-17: | Transconductance (gm) of LP Oscillator vs. VDD | 153 |
| Figure 14-20: | Transconductance(gm) of LP Oscillator vs. VDD..... | 131 | Figure 16-18: | Transconductance (gm) of XT Oscillator vs. VDD | 153 |
| Figure 14-21: | Transconductance(gm) of XT Oscillator vs. VDD | 131 | Figure 16-19: | I _{OH} vs. V _{OH} , VDD = 3V | 153 |
| Figure 14-22: | Typical XTAL Startup Time vs. VDD (LP Mode, 25°C)..... | 132 | Figure 16-20: | I _{OH} vs. V _{OH} , VDD = 5V | 153 |
| Figure 14-23: | Typical XTAL Startup Time vs. VDD (HS Mode, 25°C) | 132 | Figure 16-21: | I _{OL} vs. V _{OL} , VDD = 3V | 154 |
| Figure 14-24: | Typical XTAL Startup Time vs. VDD (XT Mode, 25°C)..... | 132 | Figure 16-22: | I _{OL} vs. V _{OL} , VDD = 5V | 154 |
| Figure 14-25: | Typical I _{DD} vs. Frequency (LP Mode, 25°C) | 133 | | | |
| Figure 14-26: | Maximum I _{DD} vs. Frequency (LP Mode, 85°C to -40°C) | 133 | | | |
| Figure 14-27: | Typical I _{DD} vs. Frequency (XT Mode, 25°C) | 133 | | | |
| Figure 14-28: | Maximum I _{DD} vs. Frequency (XT Mode, -40°C to 85°C) | 133 | | | |
| Figure 14-29: | Typical I _{DD} vs. Frequency (HS Mode, 25°C)..... | 134 | | | |
| Figure 14-30: | Maximum I _{DD} vs. Frequency (HS Mode, -40°C to 85°C)..... | 134 | | | |
| Figure 15-1: | Load Conditions | 140 | | | |
| Figure 15-2: | External Clock Timing | 141 | | | |
| Figure 15-3: | CLKOUT and I/O Timing | 142 | | | |
| Figure 15-4: | Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Timing | 143 | | | |
| Figure 15-5: | Timer0 External Clock Timings | 144 | | | |
| Figure 15-6: | A/D Conversion Timing | 146 | | | |
| Figure 16-1: | Typical RC Oscillator Frequency vs. Temperature..... | 147 | | | |
| Figure 16-2: | Typical RC Oscillator Frequency vs. VDD..... | 147 | | | |
| Figure 16-3: | Typical RC Oscillator Frequency vs. VDD..... | 147 | | | |

LIST OF TABLES

| | | | | | |
|-------------|---|----|-------------|--|-----|
| Table 1-1: | PIC16C71X Family of Devices..... | 4 | Table 11-6: | A/D Converter Characteristics: PIC16C710/711-04 (Commercial, Industrial, Extended) PIC16C710/711-10 (Commercial, Industrial, Extended) PIC16C710/711-20 (Commercial, Industrial, Extended) PIC16C710/711-04 (Commercial, Industrial, Extended) | 99 |
| Table 3-1: | PIC16C710/711/715 Pinout Description | 9 | Table 11-7: | A/D Conversion Requirements | 100 |
| Table 4-1: | PIC16C710/711 Special Function Register Summary | 14 | Table 12-1: | RC Oscillator Frequencies..... | 107 |
| Table 4-2: | PIC16C715 Special Function Register Summary | 15 | Table 12-2: | Capacitor Selection for Crystal Oscillators | 108 |
| Table 5-1: | PORTA Functions | 26 | Table 13-1: | Cross Reference of Device Specs for Oscillator Configurations and Frequencies of Operation (Commercial Devices) | 112 |
| Table 5-2: | Summary of Registers Associated with PORTA | 26 | Table 13-2: | Clock Timing Requirements..... | 118 |
| Table 5-3: | PORTB Functions | 28 | Table 13-3: | CLKOUT and I/O Timing Requirements . | 119 |
| Table 5-4: | Summary of Registers Associated with PORTB | 29 | Table 13-4: | Reset, Watchdog Timer, Oscillator Start-up Timer, Power-up Timer, and Brown-out Reset Requirements..... | 120 |
| Table 6-1: | Registers Associated with Timer0 | 35 | Table 13-5: | Timer0 Clock Requirements | 121 |
| Table 7-1: | TAD vs. Device Operating Frequencies, PIC16C71..... | 41 | Table 13-6: | A/D Converter Characteristics: PIC16C715-04 (Commercial, Industrial, Extended) PIC16C715-10 (Commercial, Industrial, Extended) PIC16C715-20 (Commercial, Industrial, Extended) | 122 |
| Table 7-2: | TAD vs. Device Operating Frequencies, PIC16C710/711, PIC16C715..... | 41 | Table 13-7: | A/D Converter Characteristics: PIC16C715-04 (Commercial, Industrial) | 123 |
| Table 7-3: | Registers/Bits Associated with A/D, PIC16C710/711..... | 46 | Table 13-8: | A/D Conversion Requirements | 124 |
| Table 7-4: | Registers/Bits Associated with A/D, PIC16C715..... | 46 | Table 14-1: | RC Oscillator Frequencies..... | 131 |
| Table 8-1: | Ceramic Resonators, PIC16C71 | 49 | Table 14-2: | Capacitor Selection for Crystal Oscillators | 132 |
| Table 8-2: | Capacitor Selection For Crystal Oscillator, PIC16C71..... | 49 | Table 15-1: | Cross Reference of Device Specs for Oscillator Configurations and Frequencies of Operation (Commercial Devices) | 135 |
| Table 8-3: | Ceramic Resonators, PIC16C710/711/715..... | 50 | Table 15-2: | External Clock Timing Requirements | 141 |
| Table 8-4: | Capacitor Selection for Crystal Oscillator, PIC16C710/711/715..... | 50 | Table 15-3: | CLKOUT and I/O Timing Requirements . | 142 |
| Table 8-5: | Time-out in Various Situations, PIC16C71..... | 54 | Table 15-4: | Reset, Watchdog Timer, Oscillator Start-up Timer and Power-up Timer Requirements | 143 |
| Table 8-6: | Time-out in Various Situations, PIC16C710/711/715..... | 54 | Table 15-5: | Timer0 External Clock Requirements | 144 |
| Table 8-7: | Status Bits and Their Significance, PIC16C71..... | 55 | Table 15-6: | A/D Converter Characteristics | 145 |
| Table 8-8: | Status Bits and Their Significance, PIC16C710/711..... | 55 | Table 15-7: | A/D Conversion Requirements | 146 |
| Table 8-9: | Status Bits and Their Significance, PIC16C715..... | 55 | Table 16-1: | RC Oscillator Frequencies..... | 148 |
| Table 8-10: | Reset Condition for Special Registers, PIC16C710/711..... | 56 | | | |
| Table 8-11: | Reset Condition for Special Registers, PIC16C715..... | 56 | | | |
| Table 8-12: | Initialization Conditions For All Registers, PIC16C710/711..... | 57 | | | |
| Table 8-13: | Initialization Conditions for All Registers, PIC16C715..... | 58 | | | |
| Table 9-1: | Opcode Field Descriptions | 69 | | | |
| Table 9-2: | PIC16CXX Instruction Set..... | 70 | | | |
| Table 10-1: | Development Tools From Microchip | 88 | | | |
| Table 11-1: | Cross Reference of Device Specs for Oscillator Configurations and Frequencies of Operation (Commercial Devices)..... | 89 | | | |
| Table 11-2: | External Clock Timing Requirements..... | 95 | | | |
| Table 11-3: | CLKOUT and I/O Timing Requirements.... | 96 | | | |
| Table 11-4: | Reset, Watchdog Timer, Oscillator Start-up Timer, Power-up Timer, and Brown-out Reset Requirements | 97 | | | |
| Table 11-5: | Timer0 External Clock Requirements | 98 | | | |

PIC16C71X

NOTES:

ON-LINE SUPPORT

Microchip provides two methods of on-line support. These are the Microchip BBS and the Microchip World Wide Web (WWW) site.

Use Microchip's Bulletin Board Service (BBS) to get current information and help about Microchip products. Microchip provides the BBS communication channel for you to use in extending your technical staff with microcontroller and memory experts.

To provide you with the most responsive service possible, the Microchip systems team monitors the BBS, posts the latest component data and software tool updates, provides technical help and embedded systems insights, and discusses how Microchip products provide project solutions.

The web site, like the BBS, is used by Microchip as a means to make files and information easily available to customers. To view the site, the user must have access to the Internet and a web browser, such as Netscape or Microsoft Explorer. Files are also available for FTP download from our FTP site.

Connecting to the Microchip Internet Web Site

The Microchip web site is available by using your favorite Internet browser to attach to:

www.microchip.com

The file transfer site is available by using an FTP service to connect to:

<ftp://ftp.futureone.com/pub/microchip>

The web site and file transfer site provide a variety of services. Users may download files for the latest Development Tools, Data Sheets, Application Notes, User's Guides, Articles and Sample Programs. A variety of Microchip specific business information is also available, including listings of Microchip sales offices, distributors and factory representatives. Other data available for consideration is:

- Latest Microchip Press Releases
- Technical Support Section with Frequently Asked Questions
- Design Tips
- Device Errata
- Job Postings
- Microchip Consultant Program Member Listing
- Links to other useful web sites related to Microchip Products

Connecting to the Microchip BBS

Connect worldwide to the Microchip BBS using either the Internet or the CompuServe® communications network.

Internet:

You can telnet or ftp to the Microchip BBS at the address: **mchibbbs.microchip.com**

CompuServe Communications Network:

When using the BBS via the Compuserve Network, in most cases, a local call is your only expense. The Microchip BBS connection does not use CompuServe membership services, therefore you do not need CompuServe membership to join Microchip's BBS. There is no charge for connecting to the Microchip BBS.

The procedure to connect will vary slightly from country to country. Please check with your local CompuServe agent for details if you have a problem. CompuServe service allow multiple users various baud rates depending on the local point of access.

The following connect procedure applies in most locations.

1. Set your modem to 8-bit, No parity, and One stop (8N1). This is not the normal CompuServe setting which is 7E1.
2. Dial your local CompuServe access number.
3. Depress the **<Enter>** key and a garbage string will appear because CompuServe is expecting a 7E1 setting.
4. Type +, depress the **<Enter>** key and "Host Name:" will appear.
5. Type MCHIPBBS, depress the **<Enter>** key and you will be connected to the Microchip BBS.

In the United States, to find the CompuServe phone number closest to you, set your modem to 7E1 and dial (800) 848-4480 for 300-2400 baud or (800) 331-7166 for 9600-14400 baud connection. After the system responds with "Host Name:", type NETWORK, depress the **<Enter>** key and follow CompuServe's directions.

For voice information (or calling from overseas), you may call (614) 723-1550 for your local CompuServe number.

Microchip regularly uses the Microchip BBS to distribute technical information, application notes, source code, errata sheets, bug reports, and interim patches for Microchip systems software products. For each SIG, a moderator monitors, scans, and approves or disapproves files submitted to the SIG. No executable files are accepted from the user community in general to limit the spread of computer viruses.

Systems Information and Upgrade Hot Line

The Systems Information and Upgrade Line provides system users a listing of the latest versions of all of Microchip's development systems software products. Plus, this line provides information on how customers can receive any currently available upgrade kits. The Hot Line Numbers are:

1-800-755-2345 for U.S. and most of Canada, and

1-602-786-7302 for the rest of the world.

970301

Trademarks: The Microchip name, logo, PIC, PICSTART, PICMASTER and PRO MATE are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries. *FlexROM*, *MPLAB* and *fuzzyLAB*, are trademarks and SQTP is a service mark of Microchip in the U.S.A.

fuzzyTECH is a registered trademark of Inform Software Corporation. IBM, IBM PC-AT are registered trademarks of International Business Machines Corp. Pentium is a trademark of Intel Corporation. Windows is a trademark and MS-DOS, Microsoft Windows are registered trademarks of Microsoft Corporation. CompuServe is a registered trademark of CompuServe Incorporated.

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

PIC16C71X

PIC16C71X PRODUCT IDENTIFICATION SYSTEM

To order or obtain information, e.g., on pricing or delivery refer to the factory or the listed sales office.

| PART NO. | -XX | X | /XX | XXX | | Examples |
|----------|-----|---|-----|-----|---------------------------|--|
| | | | | | Pattern: | QTP, SQTP, Code or Special Requirements |
| | | | | | Package: | JW = Windowed Cerdip SO = SOIC SP = Skinny plastic dip P = PDIP SS = SSOP |
| | | | | | Temperature Range: | - = 0°C to +70°C I = -40°C to +85°C E = -40°C to +125°C |
| | | | | | Frequency Range: | 04 = 200 kHz (PIC16C7X-04) 04 = 4 MHz 10 = 10 MHz 20 = 20 MHz |
| | | | | | Device | PIC16C7X :V _{DD} range 4.0V to 6.0V PIC16C7XT :V _{DD} range 4.0V to 6.0V (Tape/Reel) PIC16LC7X :V _{DD} range 2.5V to 6.0V PIC16LC7XT :V _{DD} range 2.5V to 6.0V (Tape/Reel) |

a) PIC16C71 - 04/P 301
Commercial Temp.,
PDIP Package, 4 MHz,
normal V_{DD} limits, QTP
pattern #301

* JW Devices are UV erasable and can be programmed to any device configuration. JW Devices meet the electrical requirement of each oscillator type (including LC devices).

Sales and Support

Products supported by a preliminary Data Sheet may possibly 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 (see below)
2. The Microchip Corporate Literature Center U.S. FAX: (602) 786-7277
3. The Microchip's Bulletin Board, via your local CompuServe number (CompuServe membership NOT required).

Please specify which device, revision of silicon and Data Sheet (include Literature #) you are using.

For latest version information and upgrade kits for Microchip Development Tools, please call 1-800-755-2345 or 1-602-786-7302.

PIC16C71X

NOTES:

NOTES:



WORLDWIDE SALES AND SERVICE

AMERICAS

Corporate Office

Microchip Technology Inc.
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-786-7200 Fax: 480-786-7277
Technical Support: 480-786-7627
Web Address: <http://www.microchip.com>

Atlanta

Microchip Technology Inc.
500 Sugar Mill Road, Suite 200B
Atlanta, GA 30350
Tel: 770-640-0034 Fax: 770-640-0307

Boston

Microchip Technology Inc.
5 Mount Royal Avenue
Marlborough, MA 01752
Tel: 508-480-9990 Fax: 508-480-8575

Chicago

Microchip Technology Inc.
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 630-285-0071 Fax: 630-285-0075

Dallas

Microchip Technology Inc.
4570 Westgrove Drive, Suite 160
Addison, TX 75248
Tel: 972-818-7423 Fax: 972-818-2924

Dayton

Microchip Technology Inc.
Two Prestige Place, Suite 150
Miamisburg, OH 45342
Tel: 937-291-1654 Fax: 937-291-9175

Detroit

Microchip Technology Inc.
Tri-Atria Office Building
32255 Northwestern Highway, Suite 190
Farmington Hills, MI 48334
Tel: 248-538-2250 Fax: 248-538-2260

Los Angeles

Microchip Technology Inc.
18201 Von Karman, Suite 1090
Irvine, CA 92612
Tel: 949-263-1888 Fax: 949-263-1338

New York

Microchip Technology Inc.
150 Motor Parkway, Suite 202
Hauppauge, NY 11788
Tel: 631-273-5305 Fax: 631-273-5335

San Jose

Microchip Technology Inc.
2107 North First Street, Suite 590
San Jose, CA 95131
Tel: 408-436-7950 Fax: 408-436-7955

AMERICAS (continued)

Toronto

Microchip Technology Inc.
5925 Airport Road, Suite 200
Mississauga, Ontario L4V 1W1, Canada
Tel: 905-405-6279 Fax: 905-405-6253

ASIA/PACIFIC

Hong Kong

Microchip Asia Pacific
Unit 2101, Tower 2
Metroplaza
223 Hing Fong Road
Kwai Fong, N.T., Hong Kong
Tel: 852-2-401-1200 Fax: 852-2-401-3431

Beijing

Microchip Technology, Beijing
Unit 915, 6 Chaoyangmen Bei Dajie
Dong Erhuan Road, Dongcheng District
New China Hong Kong Manhattan Building
Beijing 100027 PRC
Tel: 86-10-85282100 Fax: 86-10-85282104

India

Microchip Technology Inc.
India Liaison Office
No. 6, Legacy, Convent Road
Bangalore 560 025, India
Tel: 91-80-229-0061 Fax: 91-80-229-0062

Japan

Microchip Technology Intl. Inc.
Benex S-1 6F
3-18-20, Shinyokohama
Kohoku-Ku, Yokohama-shi
Kanagawa 222-0033 Japan
Tel: 81-45-471-6166 Fax: 81-45-471-6122

Korea

Microchip Technology Korea
168-1, Youngbo Bldg. 3 Floor
Samsung-Dong, Kangnam-Ku
Seoul, Korea
Tel: 82-2-554-7200 Fax: 82-2-558-5934

Shanghai

Microchip Technology
RM 406 Shanghai Golden Bridge Bldg.
2077 Yan'an Road West, Hong Qiao District
Shanghai, PRC 200335
Tel: 86-21-6275-5700 Fax: 86-21-6275-5060

ASIA/PACIFIC (continued)

Singapore

Microchip Technology Singapore Pte Ltd.
200 Middle Road
#07-02 Prime Centre
Singapore 188980
Tel: 65-334-8870 Fax: 65-334-8850

Taiwan, R.O.C

Microchip Technology Taiwan
10F-1C 207
Tung Hua North Road
Taipei, Taiwan, ROC
Tel: 886-2-2717-7175 Fax: 886-2-2545-0139

EUROPE

United Kingdom

Arizona Microchip Technology Ltd.
505 Eskdale Road
Winnersh Triangle
Wokingham
Berkshire, England RG41 5TU
Tel: 44 118 921 5858 Fax: 44-118 921-5835

Denmark

Microchip Technology Denmark ApS
Regus Business Centre
Lautrup høj 1-3
Ballerup DK-2750 Denmark
Tel: 45 4420 9895 Fax: 45 4420 9910

France

Arizona Microchip Technology SARL
Parc d'Activite du Moulin de Massy
43 Rue du Saule Trappu
Batiment A - 1er Etage
91300 Massy, France
Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79

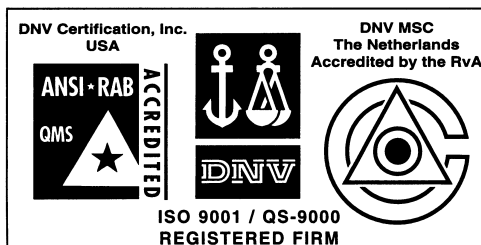
Germany

Arizona Microchip Technology GmbH
Gustav-Heinemann-Ring 125
D-81739 München, Germany
Tel: 49-89-627-144 0 Fax: 49-89-627-144-44

Italy

Arizona Microchip Technology SRL
Centro Direzionale Colleoni
Palazzo Taurus 1 V. Le Colleoni 1
20041 Agrate Brianza
Milan, Italy
Tel: 39-039-65791-1 Fax: 39-039-6899883

11/15/99



Microchip received QS-9000 quality system certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona in July 1999. The Company's quality system processes and procedures are QS-9000 compliant for its PICmicro® 8-bit MCUs, KEELOQ® code hopping devices, Serial EEPROMs and microperipheral products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001 certified.

All rights reserved. © 1999 Microchip Technology Incorporated. Printed in the USA. 11/99 Printed on recycled paper.

Information contained in this publication regarding device applications and the like is intended for suggestion only and may be superseded by updates. No representation or warranty is given and no liability is assumed by Microchip Technology Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Microchip's products as critical components in life support systems is not authorized except with express written approval by Microchip. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights. The Microchip logo and name are registered trademarks of Microchip Technology Inc. in the U.S.A. and other countries. All rights reserved. All other trademarks mentioned herein are the property of their respective companies.