**Welcome to E-XFL.COM**

**What is "Embedded - Microcontrollers"?**

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

**Applications of "Embedded - Microcontrollers"**

## Details

| | |
|---|---|
| Product Status | Active |
| Core Processor | PIC |
| Core Size | 8-Bit |
| Speed | 4MHz |
| Connectivity | - |
| Peripherals | Brown-out Detect/Reset, POR, WDT |
| Number of I/O | 13 |
| Program Memory Size | 1.75KB (1K x 14) |
| Program Memory Type | OTP |
| EEPROM Size | - |
| RAM Size | 96 x 8 |
| Voltage - Supply (Vcc/Vdd) | 2.5V ~ 5.5V |
| Data Converters | - |
| Oscillator Type | External |
| Operating Temperature | 0°C ~ 70°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 18-SOIC (0.295", 7.50mm Width) |
| Supplier Device Package | 18-SOIC |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16lc621a-04-so |

## 3.0   ARCHITECTURAL OVERVIEW

The high performance of the PIC16C62X family can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC16C62X uses a Harvard architecture, in which, program and data are accessed from separate memories using separate busses. This improves bandwidth over traditional von Neumann architecture, where program and data are fetched from the same memory. Separating program and data memory further allows instructions to be sized differently than 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. Consequently, all instructions (35) execute in a single cycle (200 ns @ 20 MHz) except for program branches.

The PIC16C620(A) and PIC16CR620A address 512 x 14 on-chip program memory. The PIC16C621(A) addresses 1K x 14 program memory. The PIC16C622(A) addresses 2K x 14 program memory. All program memory is internal.

The PIC16C62X 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 PIC16C62X 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 PIC16C62X simple yet efficient. In addition, the learning curve is reduced significantly.

The PIC16C62X devices contain an 8-bit ALU and working register. The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between 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 $\overline{\text{Borrow}}$ and $\overline{\text{Digit Borrow}}$ out bit, respectively, bit in subtraction. See the SUBLW and SUBWF instructions for examples.

A simplified block diagram is shown in Figure 3-1, with a description of the device pins in Table 3-1.

# PIC16C62X

## 3.1 Clocking Scheme/Instruction Cycle

The clock input (OSC1/CLKIN pin) 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.
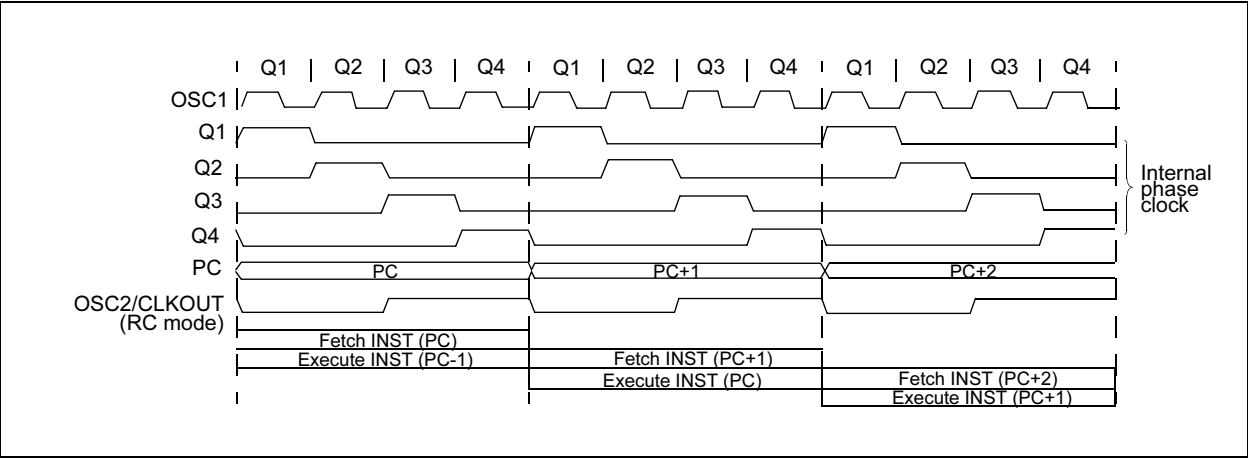
## 3.2 Instruction Flow/Pipelining

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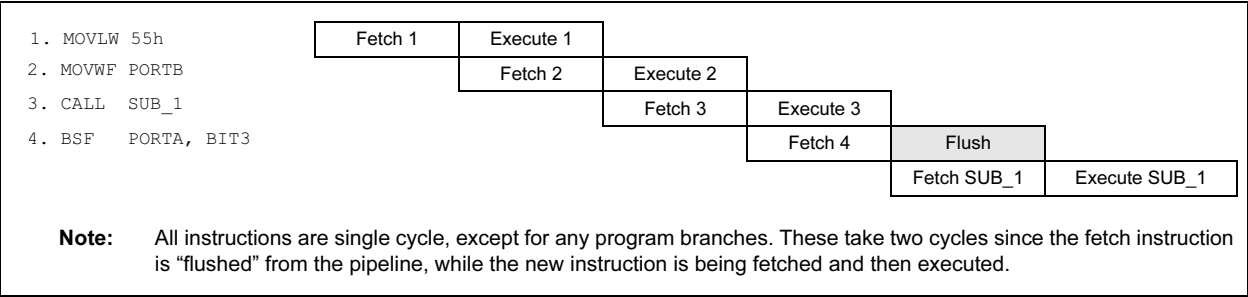
A fetch cycle begins with the program counter (PC) incrementing in Q1.

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**



| | |
|---|---|
| 1. MOVLW 55h | Fetch 1 / Execute 1 |
| 2. MOVWF PORTB | Fetch 2 / Execute 2 |
| 3. CALL  SUB_1 | Fetch 3 / Execute 3 |
| 4. BSF  PORTA, BIT3 | Fetch 4 / Flush / Fetch SUB_1 / Execute SUB_1 |

**Note:**  All instructions are single cycle, except for any program branches. These take two cycles since the fetch instruction is "flushed" from the pipeline, while the new instruction is being fetched and then executed.

# PIC16C62X

## 4.2 Data Memory Organization

The data memory (Figure 4-4, Figure 4-5, Figure 4-6 and Figure 4-7) is partitioned into two banks, which contain the General Purpose Registers and the Special Function Registers. Bank 0 is selected when the RP0 bit is cleared. Bank 1 is selected when the RP0 bit (STATUS <5>) is set. The Special Function Registers are located in the first 32 locations of each bank. Register locations 20-7Fh (Bank0) on the PIC16C620A/CR620A/621A and 20-7Fh (Bank0) and A0-BFh (Bank1) on the PIC16C622 and PIC16C622A are General Purpose Registers implemented as static RAM. Some Special Purpose Registers are mapped in Bank 1.

Addresses F0h-FFh of bank1 are implemented as common ram and mapped back to addresses 70h-7Fh in bank0 on the PIC16C620A/621A/622A/CR620A.

### 4.2.1 GENERAL PURPOSE REGISTER FILE

The register file is organized as 80 x 8 in the PIC16C620/621, 96 x 8 in the PIC16C620A/621A/CR620A and 128 x 8 in the PIC16C622(A). Each is accessed either directly or indirectly through the File Select Register FSR (Section 4.4).
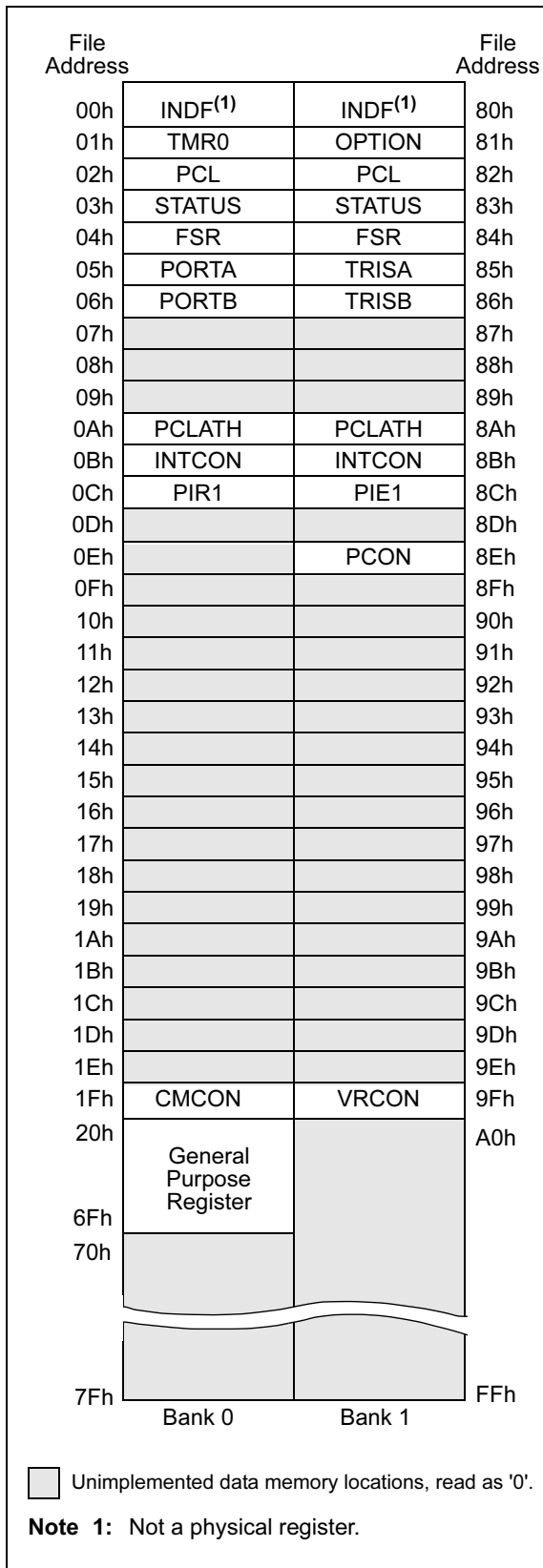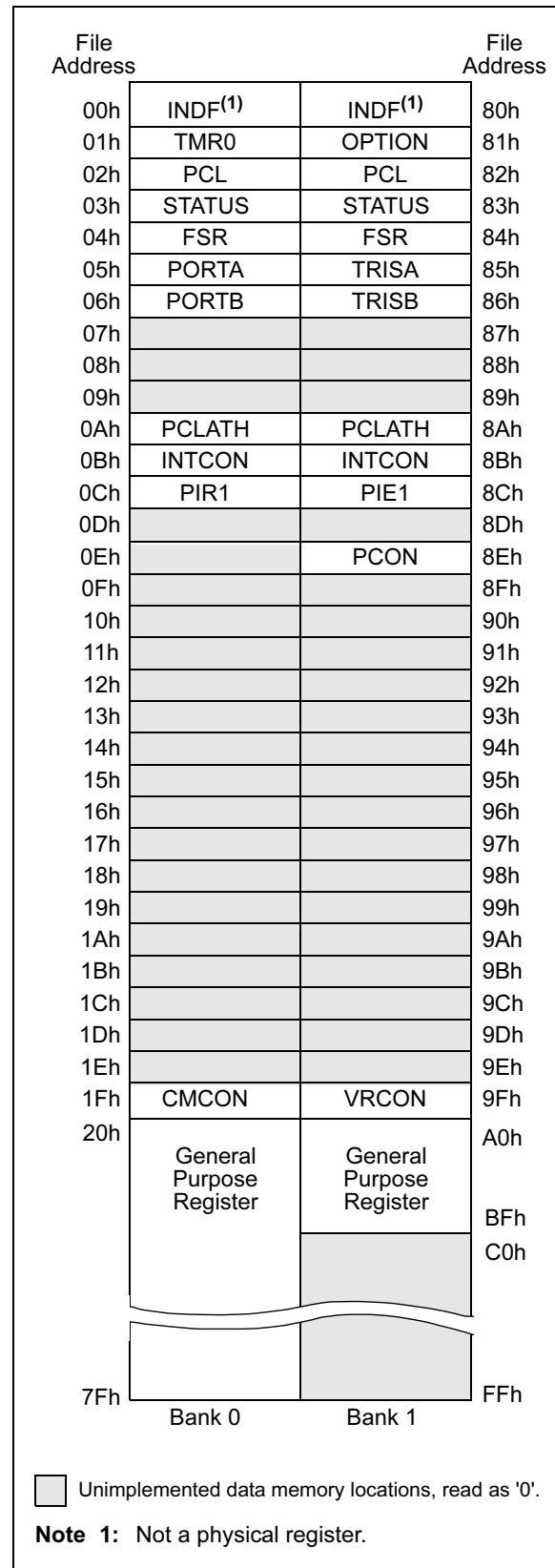
**FIGURE 4-4:** DATA MEMORY MAP FOR THE PIC16C620/621

| File Address | Bank 0 | Bank 1 | File Address |
|---|---|---|---|
| 00h | INDF[1] | INDF[1] | 80h |
| 01h | TMR0 | OPTION | 81h |
| 02h | PCL | PCL | 82h |
| 03h | STATUS | STATUS | 83h |
| 04h | FSR | FSR | 84h |
| 05h | PORTA | TRISA | 85h |
| 06h | PORTB | TRISB | 86h |
| 07h | | | 87h |
| 08h | | | 88h |
| 09h | | | 89h |
| 0Ah | PCLATH | PCLATH | 8Ah |
| 0Bh | INTCON | INTCON | 8Bh |
| 0Ch | PIR1 | PIE1 | 8Ch |
| 0Dh | | | 8Dh |
| 0Eh | | PCON | 8Eh |
| 0Fh | | | 8Fh |
| 10h | | | 90h |
| 11h | | | 91h |
| 12h | | | 92h |
| 13h | | | 93h |
| 14h | | | 94h |
| 15h | | | 95h |
| 16h | | | 96h |
| 17h | | | 97h |
| 18h | | | 98h |
| 19h | | | 99h |
| 1Ah | | | 9Ah |
| 1Bh | | | 9Bh |
| 1Ch | | | 9Ch |
| 1Dh | | | 9Dh |
| 1Eh | | | 9Eh |
| 1Fh | CMCON | VRCON | 9Fh |
| 20h | General Purpose Register | | A0h |
| 6Fh | | | |
| 70h | | | |
| 7Fh | | | FFh |

Unimplemented data memory locations, read as '0'.

**Note 1:** Not a physical register.

**FIGURE 4-5:** DATA MEMORY MAP FOR THE PIC16C622

| File Address | Bank 0 | Bank 1 | File Address |
|---|---|---|---|
| 00h | INDF[1] | INDF[1] | 80h |
| 01h | TMR0 | OPTION | 81h |
| 02h | PCL | PCL | 82h |
| 03h | STATUS | STATUS | 83h |
| 04h | FSR | FSR | 84h |
| 05h | PORTA | TRISA | 85h |
| 06h | PORTB | TRISB | 86h |
| 07h | | | 87h |
| 08h | | | 88h |
| 09h | | | 89h |
| 0Ah | PCLATH | PCLATH | 8Ah |
| 0Bh | INTCON | INTCON | 8Bh |
| 0Ch | PIR1 | PIE1 | 8Ch |
| 0Dh | | | 8Dh |
| 0Eh | | PCON | 8Eh |
| 0Fh | | | 8Fh |
| 10h | | | 90h |
| 11h | | | 91h |
| 12h | | | 92h |
| 13h | | | 93h |
| 14h | | | 94h |
| 15h | | | 95h |
| 16h | | | 96h |
| 17h | | | 97h |
| 18h | | | 98h |
| 19h | | | 99h |
| 1Ah | | | 9Ah |
| 1Bh | | | 9Bh |
| 1Ch | | | 9Ch |
| 1Dh | | | 9Dh |
| 1Eh | | | 9Eh |
| 1Fh | CMCON | VRCON | 9Fh |
| 20h | General Purpose Register | General Purpose Register | A0h |
| | | | BFh |
| | | | C0h |
| 7Fh | | | FFh |

Unimplemented data memory locations, read as '0'.

**Note 1:** Not a physical register.

## 6.2 Using Timer0 with External Clock

When an external clock input is used for Timer0, it must meet certain requirements. The external clock requirement is due to internal phase clock ($T_{OSC}$) synchronization. Also, there is a delay in the actual incrementing of Timer0 after synchronization.
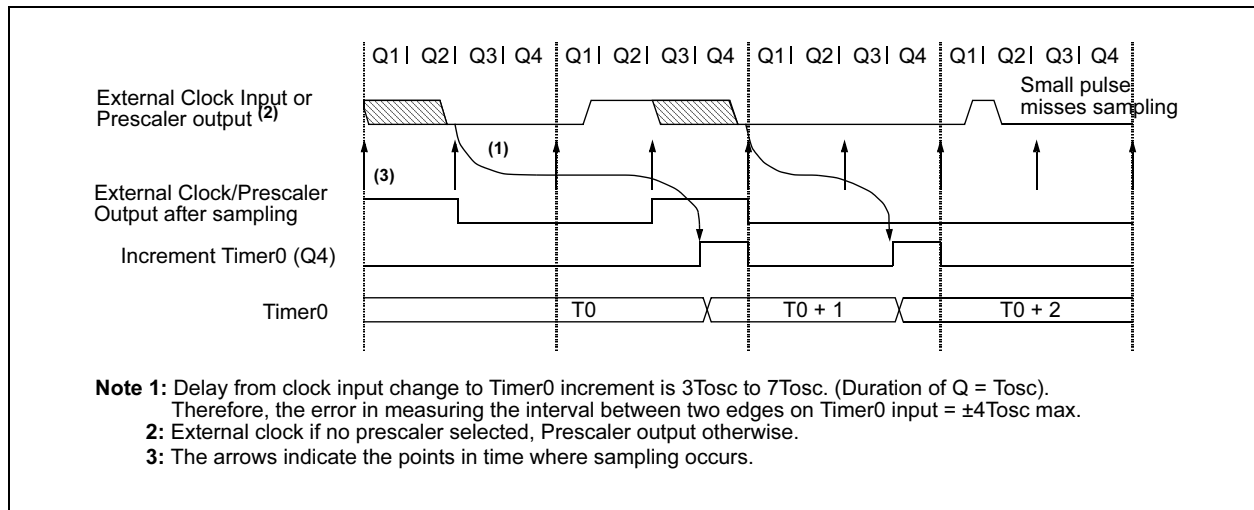
### 6.2.1 EXTERNAL CLOCK SYNCHRONIZATION

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 $2T_{OSC}$ (and a small RC delay of 20 ns) and low for at least $2T_{OSC}$ (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 prescaler, 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 $4T_{OSC}$ (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 TIMER0 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 TMR0 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**



Note 1: Delay from clock input change to Timer0 increment is 3Tosc to 7Tosc. (Duration of Q = Tosc).
Therefore, the error in measuring the interval between two edges on Timer0 input = ±4Tosc max.
2: External clock if no prescaler selected, Prescaler output otherwise.
3: The arrows indicate the points in time where sampling occurs.
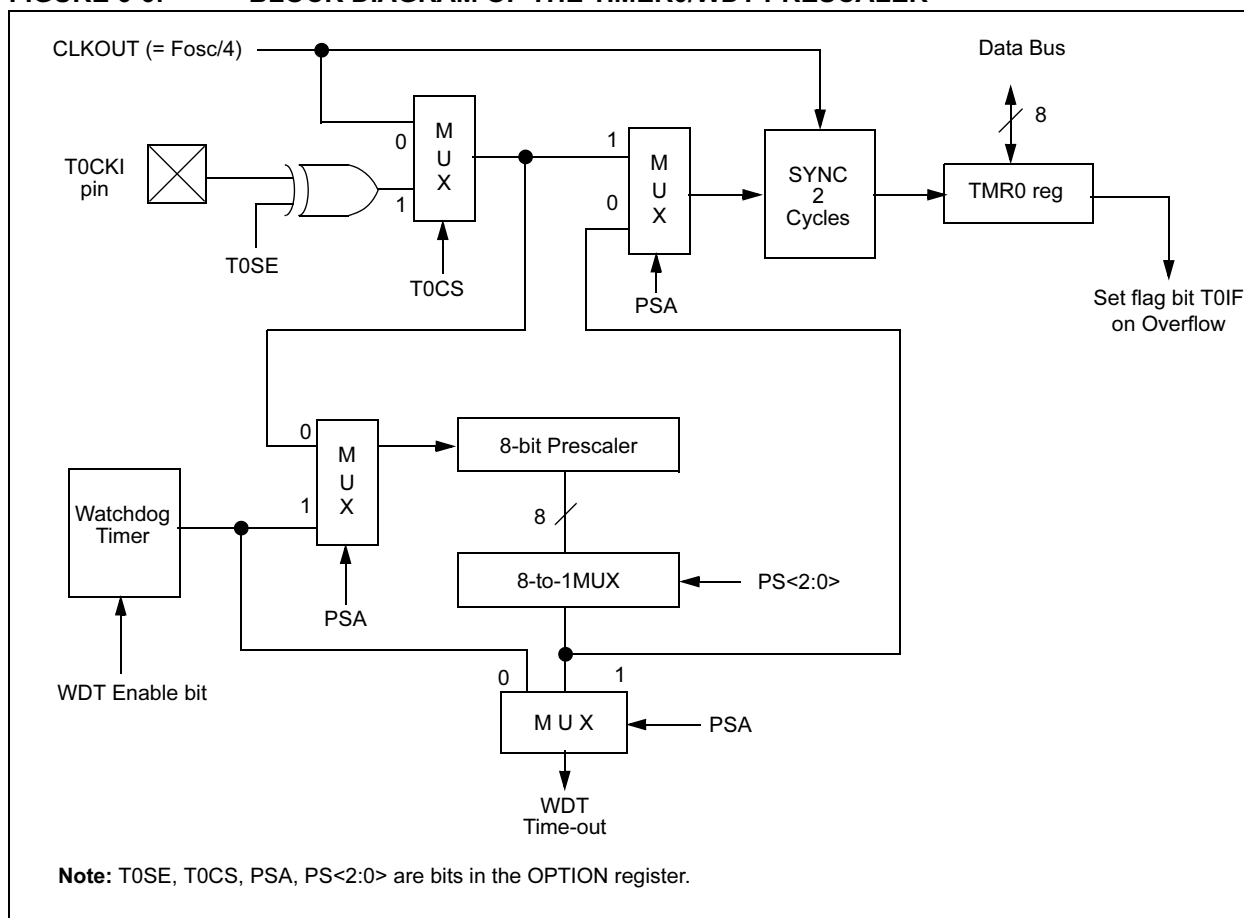
# PIC16C62X

## 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 exclusive 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 PS<2:0> 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 `CLRWDT` instruction will clear the prescaler along with the Watchdog Timer. The prescaler is not readable or writable.

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



Note: T0SE, T0CS, PSA, PS<2:0> are bits in the OPTION register.

## 9.2 Oscillator Configurations

### 9.2.1 OSCILLATOR TYPES

The PIC16C62X devices can be operated in four different oscillator options. 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

### 9.2.2 CRYSTAL OSCILLATOR / CERAMIC RESONATORS

In XT, LP or HS modes, a crystal or ceramic resonator is connected to the OSC1 and OSC2 pins to establish oscillation (Figure 9-1). The PIC16C62X 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 pin (Figure 9-2).

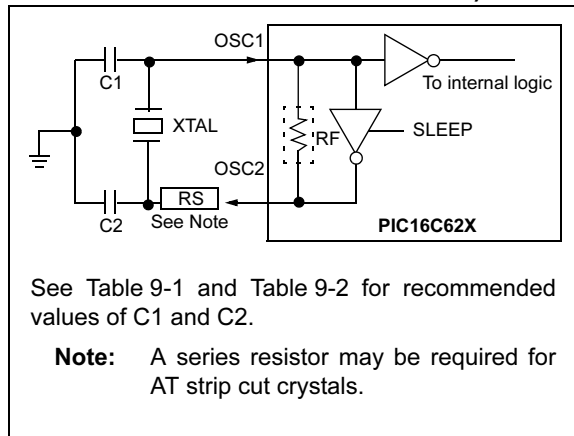**FIGURE 9-1: CRYSTAL OPERATION (OR CERAMIC RESONATOR) (HS, XT OR LP OSC CONFIGURATION)**



See Table 9-1 and Table 9-2 for recommended values of C1 and C2.

**Note:** A series resistor may be required for AT strip cut crystals.

**FIGURE 9-2: EXTERNAL CLOCK INPUT OPERATION (HS, XT OR LP OSC CONFIGURATION)**
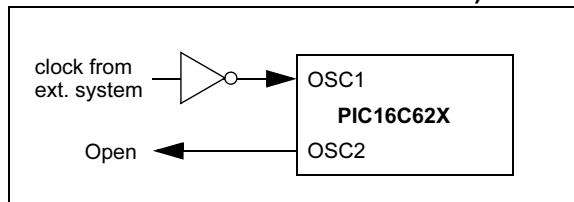


**TABLE 9-1: CAPACITOR SELECTION FOR CERAMIC RESONATORS**

| Ranges Characterized: | | | |
|---|---|---|---|
| Mode | Freq | OSC1(C1) | OSC2(C2) |
| XT | 455 kHz | 22 - 100 pF | 22 - 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 |

Higher capacitance increases the stability of the oscillator but also increases the start-up time. These values are for design guidance only. Since each resonator has its own characteristics, the user should consult the resonator manufacturer for appropriate values of external components.

**TABLE 9-2: CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR**

| Mode | Freq | OSC1(C1) | OSC2(C2) |
|---|---|---|---|
| LP | 32 kHz | 68 - 100 pF | 68 - 100 pF |
|  | 200 kHz | 15 - 30 pF | 15 - 30 pF |
| XT | 100 kHz | 68 - 150 pF | 150 - 200 pF |
|  | 2 MHz | 15 - 30 pF | 15 - 30 pF |
|  | 4 MHz | 15 - 30 pF | 15 - 30 pF |
| HS | 8 MHz | 15 - 30 pF | 15 - 30 pF |
|  | 10 MHz | 15 - 30 pF | 15 - 30 pF |
|  | 20 MHz | 15 - 30 pF | 15 - 30 pF |

Higher capacitance increases the stability of the oscillator but also increases the start-up time. These values are for design guidance only. Rs may be required in HS mode as well as XT mode to avoid overdriving crystals with low drive level specification. Since each crystal has its own characteristics, the user should consult the crystal manufacturer for appropriate values of external components.

**TABLE 9-6:    SUMMARY OF INTERRUPT REGISTERS**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on POR Reset | Value on all other RESETS[1] |
|---------|------|-------|-------|-------|-------|-------|-------|-------|-------|--------------------|------------------------------|
| 0Bh | INTCON | GIE | PEIE | T0IE | INTE | RBIE | T0IF | INTF | RBIF | 0000 000x | 0000 000u |
| 0Ch | PIR1 | — | CMIF | — | — | — | — | — | — | -0-- ---- | -0-- ---- |
| 8Ch | PIE1 | — | CMIE | — | — | — | — | — | — | -0-- ---- | -0-- ---- |

**Note   1:** Other (non Power-up) Resets include MCLR Reset, Brown-out Reset and Watchdog Timer Reset during normal operation.

## 9.6    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 (e.g., W register and STATUS register). This will have to be implemented in software.

Example 9-3 stores and restores the STATUS and W registers. The user register, W_TEMP, must be defined in both banks and must be defined at the same offset from the bank base address (i.e., W_TEMP is defined at 0x20 in Bank 0 and it must also be defined at 0xA0 in Bank 1). The user register, STATUS_TEMP, must be defined in Bank 0. The Example 9-3:

• Stores the W register
• Stores the STATUS register in Bank 0
• Executes the ISR code
• Restores the STATUS (and bank select bit register)
• Restores the W register

**EXAMPLE 9-3:      SAVING THE STATUS AND W REGISTERS IN RAM**

```
MOVWF   W_TEMP        ;copy W to temp register,
                      ;could be in either bank
SWAPF   STATUS,W      ;swap status to be saved
                      into W
BCF     STATUS,RP0    ;change to bank 0 regardless
                      ;of current bank
MOVWF   STATUS_TEMP   ;save status to bank 0
                      ;register
:
:       (ISR)
:
SWAPF   STATUS_TEMP,  ;swap STATUS_TEMP register
        W             ;into W, sets bank to origi-
                      nal
                      ;state
MOVWF   STATUS        ;move W into STATUS register
SWAPF   W_TEMP,F      ;swap W_TEMP
SWAPF   W_TEMP,W      ;swap W_TEMP into W
```

# PIC16C62X

| **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

Example 1: `SUBLW   0x02`

Before Instruction

|   | W | = | 1 |
|---|---|---|---|
|   | C | = | ? |

After Instruction

|   | W | = | 1 |
|---|---|---|---|
|   | C | = | 1; result is positive |

Example 2: Before Instruction

|   | W | = | 2 |
|---|---|---|---|
|   | C | = | ? |

After Instruction

|   | W | = | 0 |
|---|---|---|---|
|   | C | = | 1;  result is zero |

Example 3: Before Instruction

|   | W | = | 3 |
|---|---|---|---|
|   | C | = | ? |

After Instruction

|   | W | = | 0xFF |
|---|---|---|---|
|   | C | = | 0; result is negative |

| **SUBWF** | **Subtract W from f** |
|---|---|
| Syntax: | [ *label* ]   SUBWF  f,d |
| Operands: | $0 \leq f \leq 127$<br>$d \in [0,1]$ |
| Operation: | $(f) - (W) \rightarrow (dest)$ |
| 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

Example 1: `SUBWF   REG1,1`

Before Instruction

| REG1= | 3 |
|---|---|
| W   = | 2 |
| C   = | ? |

After Instruction

| REG1= | 1 |
|---|---|
| W   = | 2 |
| C   = | 1; result is positive |

Example 2: Before Instruction

| REG1= | 2 |
|---|---|
| W   = | 2 |
| C   = | ? |

After Instruction

| REG1= | 0 |
|---|---|
| W   = | 2 |
| C   = | 1; result is zero |

Example 3: Before Instruction

| REG1= | 1 |
|---|---|
| W   = | 2 |
| C   = | ? |

After Instruction

| REG1= | 0xFF |
|---|---|
| W   = | 2 |
| C   = | 0; result is negative |

| SWAPF | Swap Nibbles in f |
|---|---|
| Syntax: | [ *label* ]   SWAPF  f,d |
| Operands: | 0 ≤ f ≤ 127<br>d ∈ [0,1] |
| Operation: | (f<3:0>) → (dest<7:4>),<br>(f<7:4>) → (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 |
| Example | SWAPF  REG,   0 |

Before Instruction

    REG1    =    0xA5

After Instruction

    REG1    =    0xA5
    W       =    0x5A

| TRIS | Load TRIS Register |
|---|---|
| Syntax: | [ *label* ]   TRIS    f |
| Operands: | 5 ≤ f ≤ 7 |
| Operation: | (W) → 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 PICmicro® products, do not use this instruction.**

| XORLW | Exclusive OR Literal with W |
|---|---|
| Syntax: | [ *label*    XORLW   k ] |
| Operands: | 0 ≤ k ≤ 255 |
| Operation: | (W) .XOR. k → (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 |
| Example: | XORLW   0xAF |

Before Instruction

    W    =    0xB5

After Instruction

    W    =    0x1A

| XORWF | Exclusive OR W with f |
|---|---|
| Syntax: | [ *label* ]   XORWF   f,d |
| Operands: | 0 ≤ f ≤ 127<br>d ∈ [0,1] |
| Operation: | (W) .XOR. (f) → (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 |
| Example | XORWF   REG   1 |

Before Instruction

    REG     =    0xAF
    W       =    0xB5

After Instruction

    REG     =    0x1A
    W       =    0xB5

**FIGURE 12-7:** **PIC16CR62XA VOLTAGE-FREQUENCY GRAPH, 0°C $\leq$ TA $\leq$ +70°C**



**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.

   **2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.
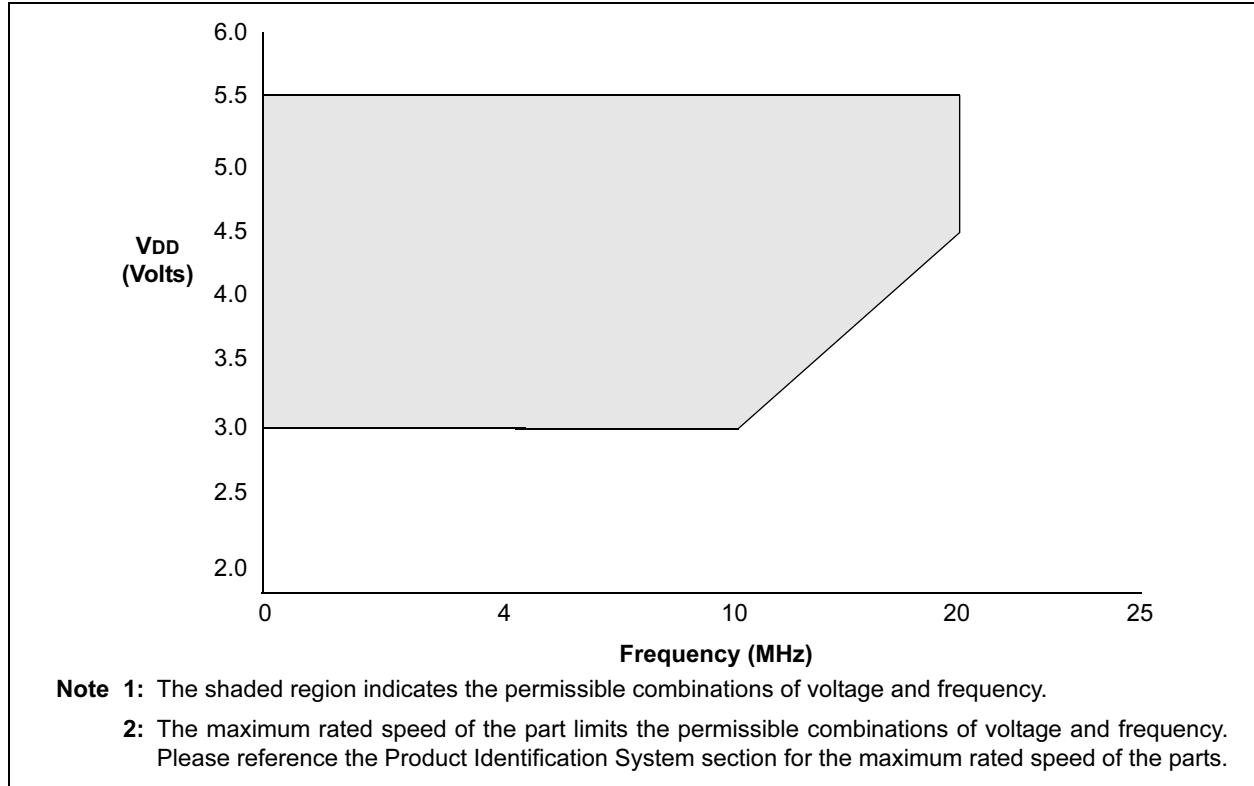
**FIGURE 12-8:** **PIC16CR62XA VOLTAGE-FREQUENCY GRAPH, -40°C $\leq$ TA $\leq$ 0°C, +70°C $\leq$ TA $\leq$ +125°C**



**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.

   **2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.

# PIC16C62X

**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.

**2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.

**FIGURE 12-10:** **PIC16C620A/C621A/C622A/CR620A - 40 VOLTAGE-FREQUENCY GRAPH,**
**0°C ≤ TA ≤ +70°C**



**Note 1:** The shaded region indicates the permissible combinations of voltage and frequency.

**2:** The maximum rated speed of the part limits the permissible combinations of voltage and frequency. Please reference the Product Identification System section for the maximum rated speed of the parts.

**3:** Operation between 20 to 40 MHz requires the following:

- VDD between 4.5V. and 5.5V
- OSC1 externally driven
- OSC2 not connected
- HS mode
- Commercial temperatures

Devices qualified for 40 MHz operation have -40 designation (ex: PIC16C620A-40/P).

## 12.1   DC Characteristics:   PIC16C62X-04 (Commercial, Industrial, Extended)
PIC16C62X-20 (Commercial, Industrial, Extended)
PIC16LC62X-04 (Commercial, Industrial, Extended) (CONT.)

| PIC16C62X | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature  -40°C  ≤ TA ≤ +85°C for industrial and<br>0°C  ≤ TA ≤ +70°C for commercial and<br>-40°C  ≤ TA ≤ +125°C for extended | | | | | |
|---|---|---|---|---|---|---|
| PIC16LC62X | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature  -40°C  ≤ TA ≤ +85°C for industrial and<br>0°C  ≤ TA ≤ +70°C for commercial and<br>-40°C  ≤ TA ≤ +125°C for extended<br>Operating voltage VDD range is the PIC16C62X range**.** | | | | | |
| **Param. No.** | **Sym** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| D022 | ΔIWDT | WDT Current[5] | — | 6.0 | 20 | μA | VDD=4.0V |
|  |  |  |  |  | 25 | μA | (125°C) |
| D022A | ΔIBOR | Brown-out Reset Current[5] | — | 350 | 425 | μA | BOD enabled, VDD = 5.0V |
| D023 | ΔICOMP | Comparator Current for each Comparator[5] | — | — | 100 | μA | VDD = 4.0V |
| D023A | ΔIVREF | VREF Current[5] | — | — | 300 | μA | VDD = 4.0V |
| D022 | ΔIWDT | WDT Current[5] | — | 6.0 | 15 | μA | VDD=3.0V |
| D022A | ΔIBOR | Brown-out Reset Current[5] | — | 350 | 425 | μA | BOD enabled, VDD = 5.0V |
| D023 | ΔICOMP | Comparator Current for each Comparator[5] | — | — | 100 | μA | VDD = 3.0V |
| D023A | ΔIVREF | VREF Current[5] | — | — | 300 | μA | VDD = 3.0V |
| 1A | FOSC | LP Oscillator Operating Frequency | 0 | — | 200 | kHz | All temperatures |
|  |  | RC Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
|  |  | XT Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
|  |  | HS Oscillator Operating Frequency | 0 | — | 20 | MHz | All temperatures |
| 1A | FOSC | LP Oscillator Operating Frequency | 0 | — | 200 | kHz | All temperatures |
|  |  | RC Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
|  |  | XT Oscillator Operating Frequency | 0 | — | 4 | MHz | All temperatures |
|  |  | HS Oscillator Operating Frequency | 0 | — | 20 | MHz | All temperatures |

\*      These parameters are characterized but not tested.

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

**Note 1:**   This is the limit to which 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 tri-stated, pulled to VDD,

MCLR = VDD; WDT enabled/disabled as specified.

**3:**   The power-down current in SLEEP mode does not depend on the oscillator type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD or VSS.

**4:**   For RC osc configuration, current through REXT is not included. The current through the resistor can be estimated by the formula:  Ir = VDD/2REXT (mA) with REXT in kΩ.

**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.

## 12.3 DC CHARACTERISTICS: PIC16CR62XA-04 (Commercial, Industrial, Extended)
### PIC16CR62XA-20 (Commercial, Industrial, Extended)
### PIC16LCR62XA-04 (Commercial, Industrial, Extended)

| PIC16CR62XA-04<br>PIC16CR62XA-20 | | | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature  -40°C  ≤ TA ≤ +85°C for industrial and<br>0°C  ≤ TA ≤ +70°C for commercial and<br>-40°C  ≤ TA ≤ +125°C for extended | | | | |
|---|---|---|---|---|---|---|---|
| **PIC16LCR62XA-04** | | | **Standard Operating Conditions (unless otherwise stated)**<br>Operating temperature  -40°C  ≤ TA ≤ +85°C for industrial and<br>0°C  ≤ TA ≤ +70°C for commercial and<br>-40°C  ≤ TA ≤ +125°C for extended | | | | |
| **Param. No.** | **Sym** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| D001 | $V_{DD}$ | Supply Voltage | 3.0 | — | 5.5 | V | See Figures 12-7, 12-8, 12-9 |
| D001 | $V_{DD}$ | Supply Voltage | 2.5 | — | 5.5 | V | See Figures 12-7, 12-8, 12-9 |
| D002 | $V_{DR}$ | RAM Data Retention Voltage[1] | — | 1.5* | — | V | Device in SLEEP mode |
| D002 | $V_{DR}$ | RAM Data Retention Voltage[1] | — | 1.5* | — | V | Device in SLEEP mode |
| D003 | $V_{POR}$ | $V_{DD}$ start voltage to ensure Power-on Reset | — | $V_{SS}$ | — | V | See section on Power-on Reset for details |
| D003 | $V_{POR}$ | $V_{DD}$ start voltage to ensure Power-on Reset | — | $V_{SS}$ | — | V | See section on Power-on Reset for details |
| D004 | $S_{VDD}$ | $V_{DD}$ rise rate to ensure Power-on Reset | 0.05* | — | — | V/ms | See section on Power-on Reset for details |
| D004 | $S_{VDD}$ | $V_{DD}$ rise rate to ensure Power-on Reset | 0.05* | — | — | V/ms | See section on Power-on Reset for details |
| D005 | $V_{BOR}$ | Brown-out Detect Voltage | 3.7 | 4.0 | 4.35 | V | BOREN configuration bit is cleared |
| D005 | $V_{BOR}$ | Brown-out Detect Voltage | 3.7 | 4.0 | 4.35 | V | BOREN configuration bit is cleared |
| D010 | $I_{DD}$ | Supply Current[2] | — | 1.2 | 1.7 | mA | $F_{OSC}$ = 4 MHz, $V_{DD}$ = 5.5V, WDT disabled, XT mode, (**Note 4**)* |
| | | | — | 500 | 900 | μA | $F_{OSC}$ = 4 MHz, $V_{DD}$ = 3.0V, WDT disabled, XT mode, (**Note 4**) |
| | | | — | 1.0 | 2.0 | mA | $F_{OSC}$ = 10 MHz, $V_{DD}$ = 3.0V, WDT disabled, HS mode, (**Note 6**) |
| | | | — | 4.0 | 7.0 | mA | $F_{OSC}$ = 20 MHz, $V_{DD}$ = 5.5V, WDT disabled*, HS mode |
| | | | — | 3.0 | 6.0 | mA | $F_{OSC}$ = 20 MHz, $V_{DD}$ = 4.5V, WDT disabled, HS mode |
| | | | — | 35 | 70 | μA | $F_{OSC}$ = 32 kHz, $V_{DD}$ = 3.0V, WDT disabled, LP mode |
| D010 | $I_{DD}$ | Supply Current[2] | — | 1.2 | 1.7 | mA | $F_{OSC}$ = 4.0 MHz, $V_{DD}$ = 5.5V, WDT disabled, XT mode, (**Note 4**)* |
| | | | — | 400 | 800 | μA | $F_{OSC}$ = 4.0 MHz, $V_{DD}$ = 2.5V, WDT disabled, XT mode (**Note 4**) |
| | | | — | 35 | 70 | μA | $F_{OSC}$ = 32 kHz, $V_{DD}$ = 2.5V, WDT disabled, LP mode |

**FIGURE 12-14:** RESET, WATCHDOG TIMER, OSCILLATOR START-UP TIMER AND POWER-UP TIMER TIMING
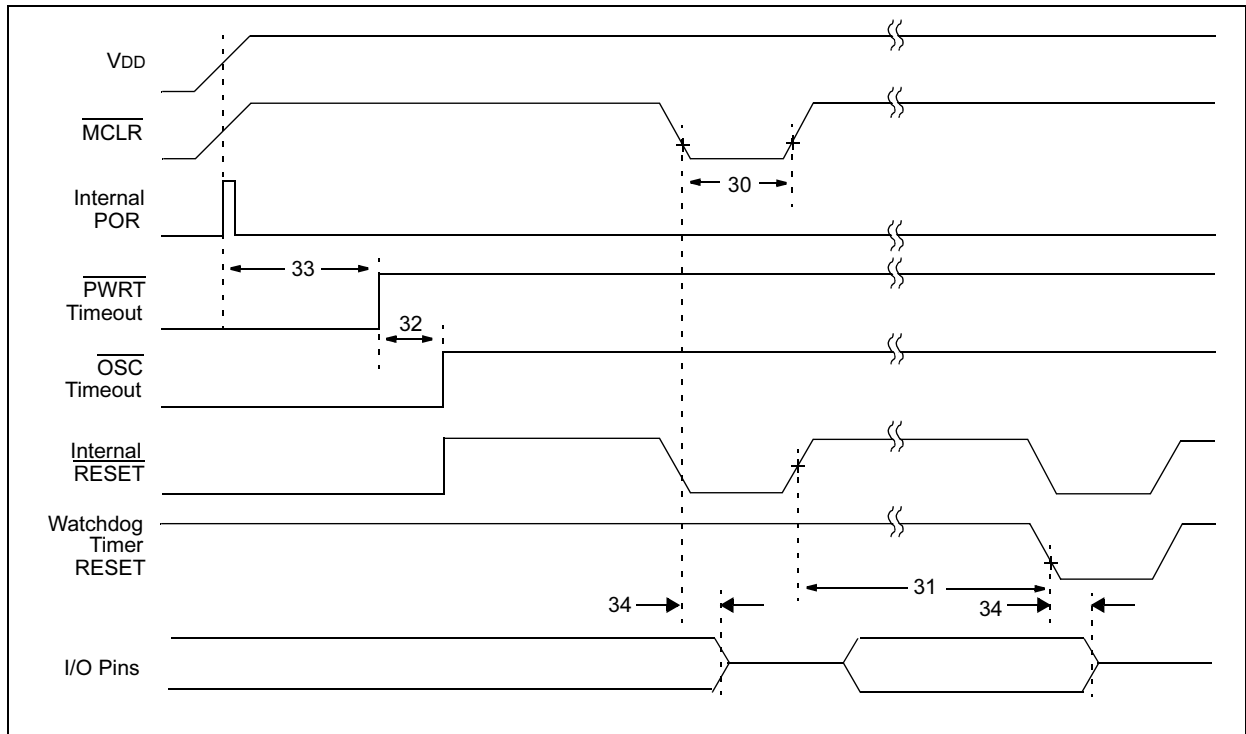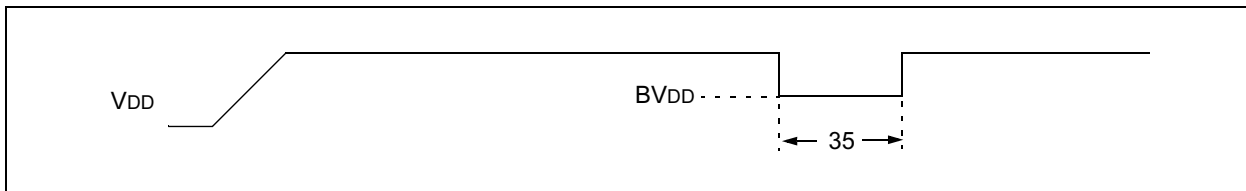


**FIGURE 12-15:** BROWN-OUT RESET TIMING



**TABLE 12-5:** 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) | 2000 | — | — | ns | -40° to +85°C |
| 31 | Twdt | Watchdog Timer Time-out Period (No Prescaler) | 7* | 18 | 33* | ms | VDD = 5.0V, -40° 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 = 5.0V, -40° to +85°C |
| 34 | TIOZ | I/O hi-impedance from MCLR low | | — | 2.0 | μs | |
| 35 | TBOR | Brown-out Reset Pulse Width | 100* | — | — | μs | 3.7V ≤ VDD ≤ 4.3V |

\*   These parameters are characterized but not tested.

†   Data in "Typ" column is at 5.0V, 25°C, unless otherwise stated. These parameters are for design guidance only and are not tested.

**FIGURE 12-16: TIMER0 CLOCK TIMING**
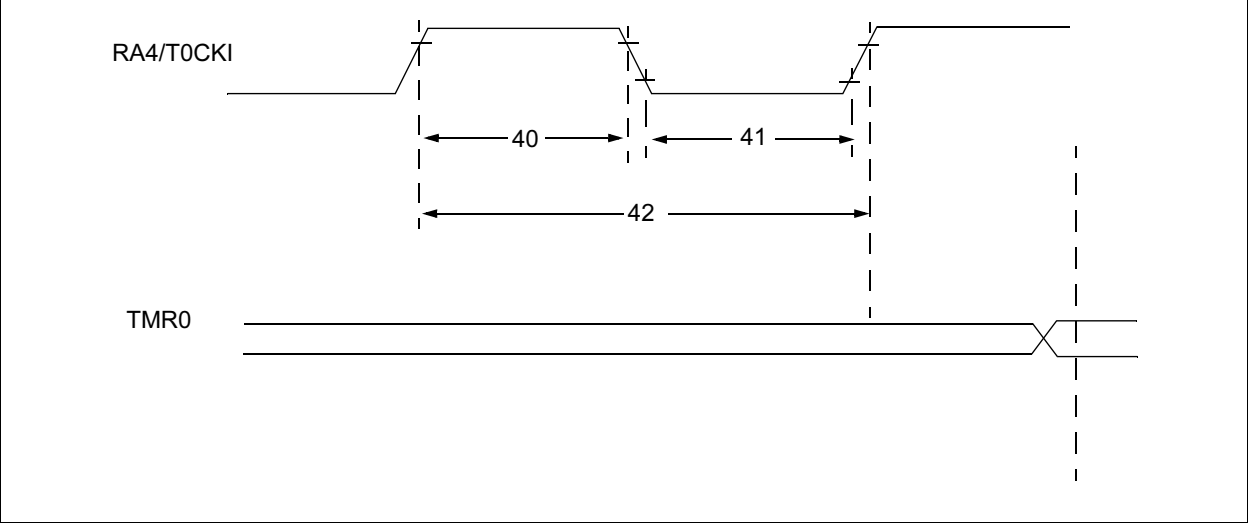


**TABLE 12-6: TIMER0 CLOCK REQUIREMENTS**

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

\* These parameters are characterized but not tested.

† Data in "Typ" column is at 5.0V, 25°C, unless otherwise stated. These parameters are for design guidance only and are not tested.

## 13.0   DEVICE CHARACTERIZATION INFORMATION

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

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

**FIGURE 13-1:**   IDD VS. FREQUENCY (XT MODE, VDD = 5.5V)
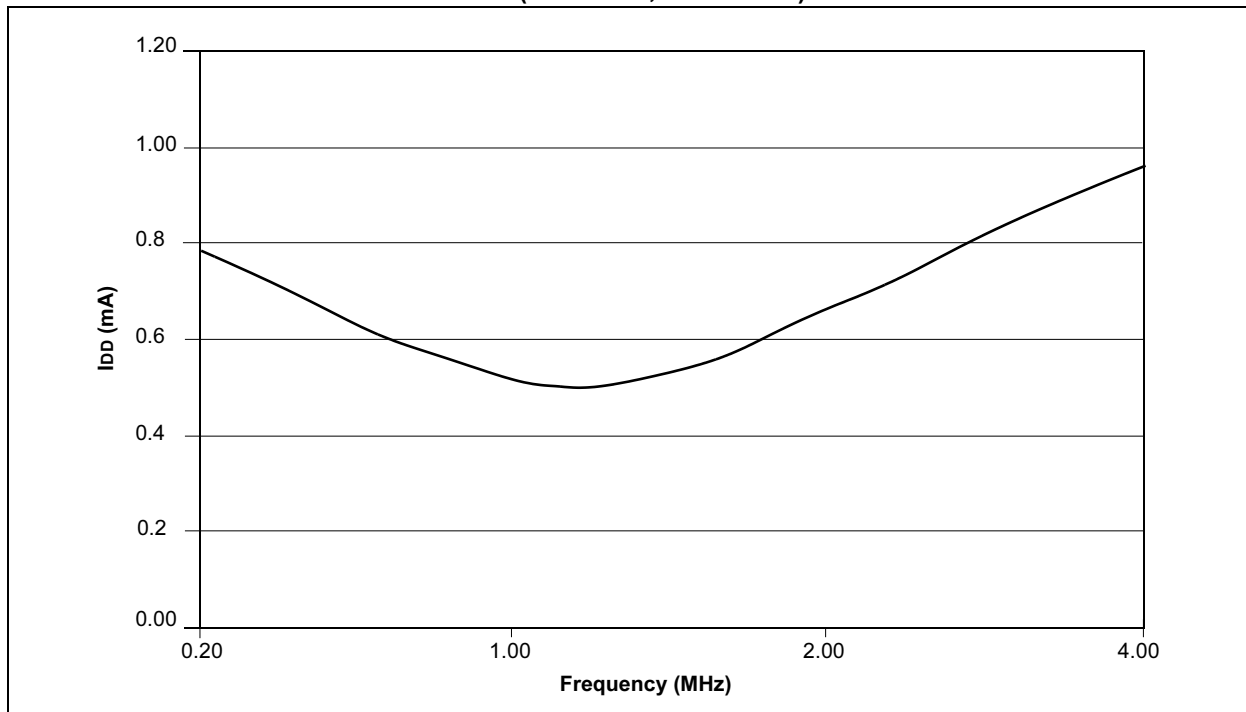


**FIGURE 13-2:**   PIC16C622A IPD VS. VDD (WDT DISABLE)
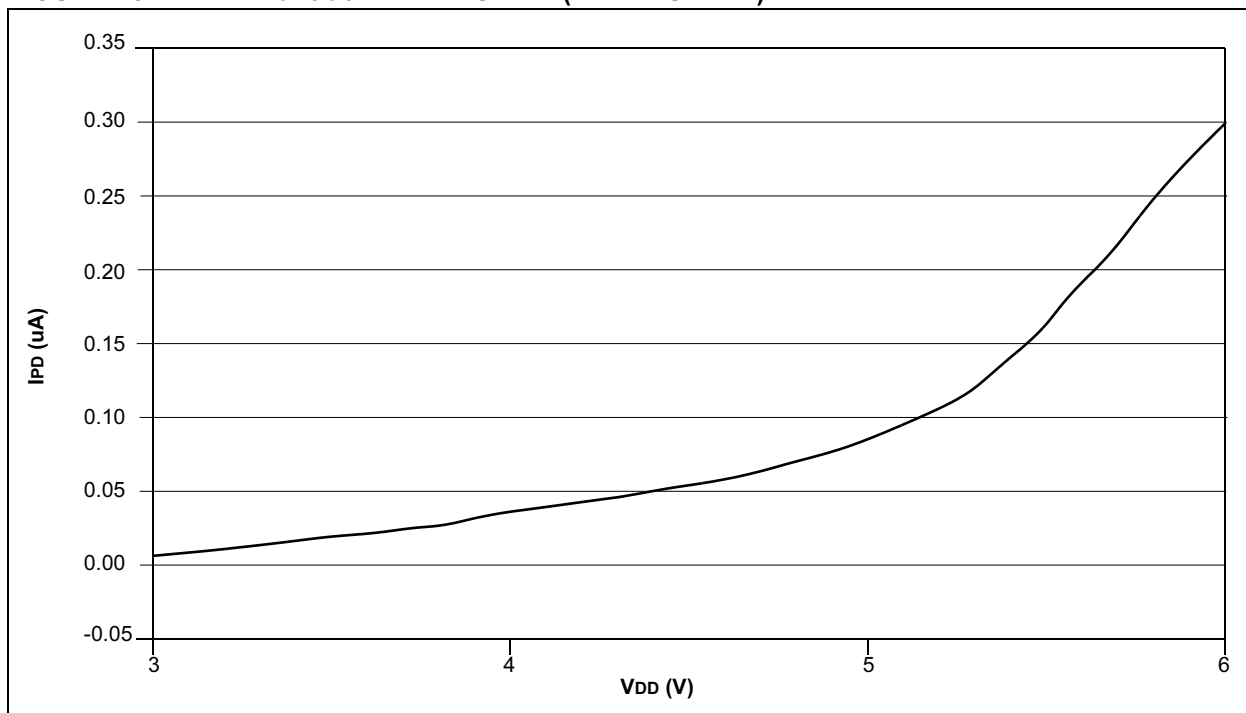
## APPENDIX A: ENHANCEMENTS

The following are the list of enhancements over the PIC16C5X microcontroller family:

1. Instruction word length is increased to 14 bits. This allows larger page sizes both in program memory (4K now as opposed to 512 before) and register file (up to 128 bytes now versus 32 bytes before).

2. A PC high latch register (PCLATH) is added to handle program memory paging. PA2, PA1, PA0 bits are removed from STATUS register.

3. Data memory paging is slightly redefined. 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. Timer0 clock input, T0CKI pin is also a port pin (RA4/T0CKI) and has a TRIS bit.

14. FSR is made a full 8-bit register.

15. "In-circuit programming" is made possible. The user can program PIC16CXX devices using only five pins: $V_{DD}$, $V_{SS}$, $V_{PP}$, RB6 (clock) and RB7 (data in/out).

16. PCON STATUS register is added with a Power-on-Reset ($\overline{POR}$) STATUS bit and a Brown-out Reset STATUS bit ($\overline{BOD}$).

17. Code protection scheme is enhanced such that portions of the program memory can be protected, while the remainder is unprotected.

18. PORTA inputs are now Schmitt Trigger inputs.

19. Brown-out Reset reset has been added.

20. Common RAM registers F0h-FFh implemented in bank1.

## 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.

# WORLDWIDE SALES AND SERVICE

## AMERICAS

**Corporate Office**
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7200 Fax: 480-792-7277
Technical Support: 480-792-7627
Web Address: http://www.microchip.com

**Atlanta**
3780 Mansell Road, Suite 130
Alpharetta, GA 30022
Tel: 770-640-0034 Fax: 770-640-0307

**Boston**
2 Lan Drive, Suite 120
Westford, MA 01886
Tel: 978-692-3848 Fax: 978-692-3821

**Chicago**
333 Pierce Road, Suite 180
Itasca, IL 60143
Tel: 630-285-0071 Fax: 630-285-0075

**Dallas**
4570 Westgrove Drive, Suite 160
Addison, TX 75001
Tel: 972-818-7423 Fax: 972-818-2924

**Detroit**
Tri-Atria Office Building
32255 Northwestern Highway, Suite 190
Farmington Hills, MI 48334
Tel: 248-538-2250 Fax: 248-538-2260

**Kokomo**
2767 S. Albright Road
Kokomo, Indiana 46902
Tel: 765-864-8360 Fax: 765-864-8387

**Los Angeles**
18201 Von Karman, Suite 1090
Irvine, CA 92612
Tel: 949-263-1888 Fax: 949-263-1338

**Phoenix**
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 480-792-7966 Fax: 480-792-4338

**San Jose**
Microchip Technology Inc.
2107 North First Street, Suite 590
San Jose, CA 95131
Tel: 408-436-7950 Fax: 408-436-7955

**Toronto**
6285 Northam Drive, Suite 108
Mississauga, Ontario L4V 1X5, Canada
Tel: 905-673-0699 Fax: 905-673-6509

## ASIA/PACIFIC

**Australia**
Microchip Technology Australia Pty Ltd
Marketing Support Division
Suite 22, 41 Rawson Street
Epping 2121, NSW
Australia
Tel: 61-2-9868-6733 Fax: 61-2-9868-6755

**China - Beijing**
Microchip Technology Consulting (Shanghai)
Co., Ltd., Beijing Liaison Office
Unit 915
Bei Hai Wan Tai Bldg.
No. 6 Chaoyangmen Beidajie
Beijing, 100027, No. China
Tel: 86-10-85282100 Fax: 86-10-85282104

**China - Chengdu**
Microchip Technology Consulting (Shanghai)
Co., Ltd., Chengdu Liaison Office
Rm. 2401-2402, 24th Floor,
Ming Xing Financial Tower
No. 88 TIDU Street
Chengdu 610016, China
Tel: 86-28-86766200 Fax: 86-28-86766599

**China - Fuzhou**
Microchip Technology Consulting (Shanghai)
Co., Ltd., Fuzhou Liaison Office
Unit 28F, World Trade Plaza
No. 71 Wusi Road
Fuzhou 350001, China
Tel: 86-591-7503506 Fax: 86-591-7503521

**China - Hong Kong SAR**
Microchip Technology Hongkong Ltd.
Unit 901-6, Tower 2, Metroplaza
223 Hing Fong Road
Kwai Fong, N.T., Hong Kong
Tel: 852-2401-1200 Fax: 852-2401-3431

**China - Shanghai**
Microchip Technology Consulting (Shanghai)
Co., Ltd.
Room 701, Bldg. B
Far East International Plaza
No. 317 Xian Xia Road
Shanghai, 200051
Tel: 86-21-6275-5700 Fax: 86-21-6275-5060

**China - Shenzhen**
Microchip Technology Consulting (Shanghai)
Co., Ltd., Shenzhen Liaison Office
Rm. 1812, 18/F, Building A, United Plaza
No. 5022 Binhe Road, Futian District
Shenzhen 518033, China
Tel: 86-755-82901380 Fax: 86-755-82966626

**China - Qingdao**
Rm. B505A, Fullhope Plaza,
No. 12 Hong Kong Central Rd.
Qingdao 266071, China
Tel: 86-532-5027355 Fax: 86-532-5027205

**India**
Microchip Technology Inc.
India Liaison Office
Marketing Support Division
Divyasree Chambers
1 Floor, Wing A (A3/A4)
No. 11, O'Shaugnessey Road
Bangalore, 560 025, India
Tel: 91-80-2290061 Fax: 91-80-2290062

**Japan**
Microchip Technology Japan K.K.
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 135-882
Tel: 82-2-554-7200 Fax: 82-2-558-5934

**Singapore**
Microchip Technology Singapore Pte Ltd.
200 Middle Road
#07-02 Prime Centre
Singapore, 188980
Tel: 65-6334-8870 Fax: 65-6334-8850

**Taiwan**
Microchip Technology (Barbados) Inc.,
Taiwan Branch
11F-3, No. 207
Tung Hua North Road
Taipei, 105, Taiwan
Tel: 886-2-2717-7175 Fax: 886-2-2545-0139

## EUROPE

**Austria**
Microchip Technology Austria GmbH
Durisolstrasse 2
A-4600 Wels
Austria
Tel: 43-7242-2244-399
Fax: 43-7242-2244-393

**Denmark**
Microchip Technology Nordic ApS
Regus Business Centre
Lautrup hoj 1-3
Ballerup DK-2750 Denmark
Tel: 45 4420 9895 Fax: 45 4420 9910

**France**
Microchip Technology SARL
Parc d'Activite du Moulin de Massy
43 Rue du Saule Trapu
Batiment A - ler Etage
91300 Massy, France
Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79

**Germany**
Microchip Technology GmbH
Steinheilstrasse 10
D-85737 Ismaning, Germany
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

**Italy**
Microchip Technology SRL
Via Quasimodo, 12
20025 Legnano (MI)
Milan, Italy
Tel: 39-0331-742611 Fax: 39-0331-466781

**United Kingdom**
Microchip Ltd.
505 Eskdale Road
Winnersh Triangle
Wokingham
Berkshire, England RG41 5TU
Tel: 44 118 921 5869 Fax: 44-118 921-5820

03/25/03