**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 | 10MHz |
| Connectivity | - |
| Peripherals | POR, WDT |
| Number of I/O | 5 |
| Program Memory Size | 1.75KB (1K x 14) |
| Program Memory Type | OTP |
| EEPROM Size | - |
| RAM Size | 128 x 8 |
| Voltage - Supply (Vcc/Vdd) | 3V ~ 5.5V |
| Data Converters | A/D 4x8b |
| Oscillator Type | Internal |
| Operating Temperature | -40°C ~ 85°C (TA) |
| Mounting Type | Surface Mount |
| Package / Case | 8-SOIC (0.209", 5.30mm Width) |
| Supplier Device Package | 8-SOIJ |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic12c671t-10i-sm |

## 2.0 PIC12C67X 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 PIC12C67X 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 example, the PIC12C67X device "type" is indicated in the device number:

1. **C**, as in PIC12**C**671. These devices have EPROM type memory and operate over the standard voltage range.
2. **LC**, as in PIC12**LC**671. These devices have EPROM type memory and operate over an extended voltage range.
3. **CE**, as in PIC12**CE**674. These devices have EPROM type memory, EEPROM data memory and operate over the standard voltage range.
4. **LCE**, as in PIC12**LCE**674. These devices have EPROM type memory, EEPROM data memory and operate over an extended voltage range.

### 2.1 UV Erasable Devices

The UV erasable version, offered in windowed package, is optimal for prototype development and pilot programs.

The UV erasable version can be erased and reprogrammed to any of the configuration modes. Microchip's PICSTART® Plus and PRO MATE® programmers both support the PIC12C67X. Third party programmers also are available; refer to the Microchip Third Party Guide for a list of sources.

| | |
|---|---|
| **Note:** | Please note that erasing the device will also erase the pre-programmed internal calibration value for the internal oscillator. The calibration value must be saved prior to erasing the part. |

### 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-Turn-Programming (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-Turn Programming (SQTP℠) 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.

### 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, and 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.

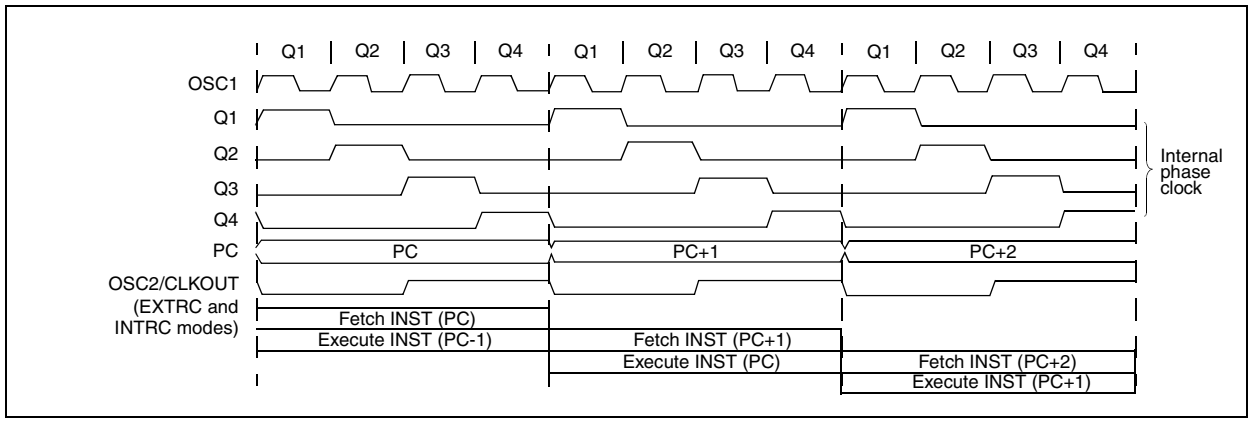
### 3.2 Instruction Flow/Pipelining

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 (i.e., `GOTO`), then two cycles are required to complete the instruction (Example 3-1).

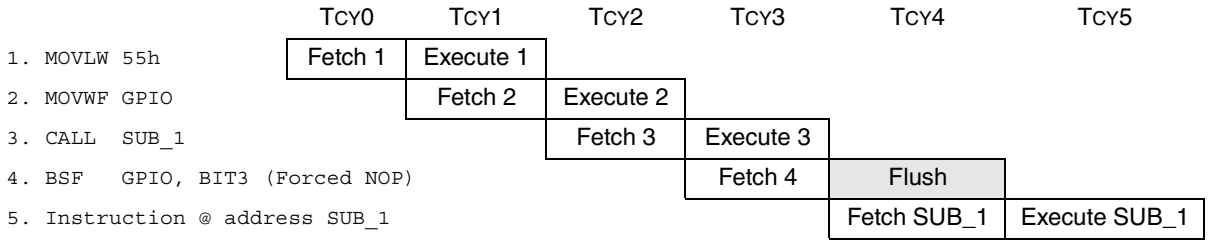
A fetch cycle begins with the program counter (PC) incrementing in Q1.

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**

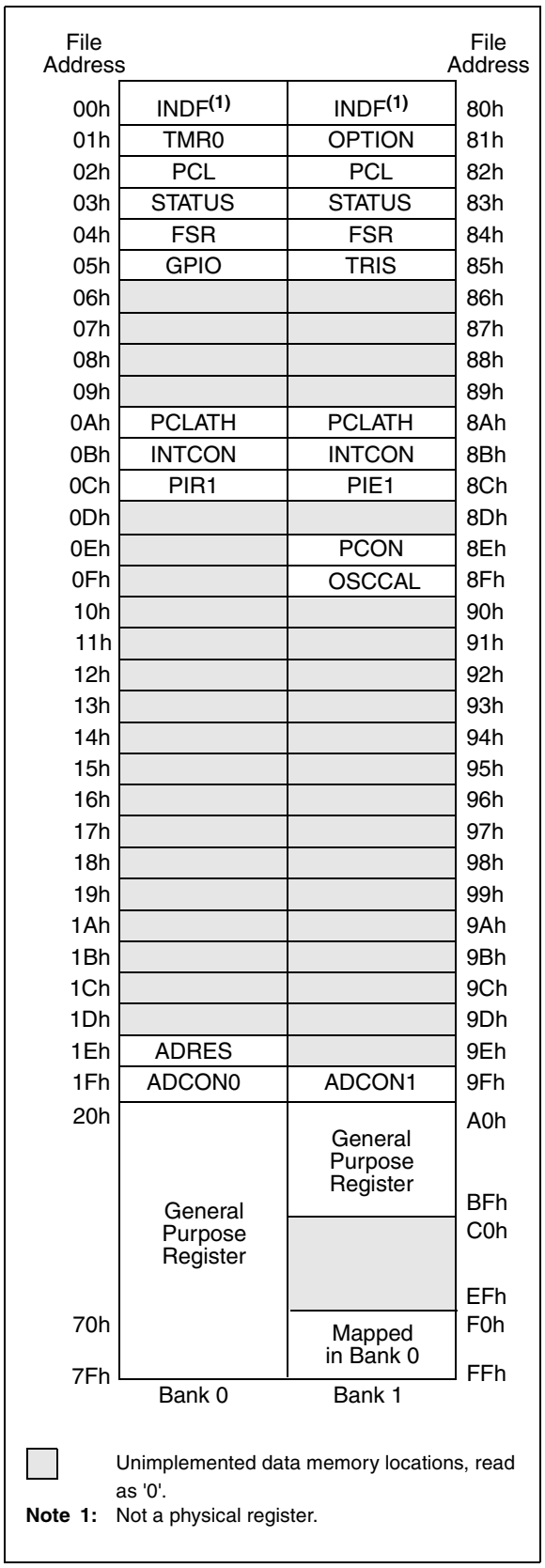



**EXAMPLE 3-1: INSTRUCTION PIPELINE FLOW**




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

**FIGURE 4-2: PIC12C67X REGISTER FILE MAP**




### 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: PIC12C67X SPECIAL FUNCTION REGISTER SUMMARY**

| Address | Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value on Power-on Reset | 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 | $\overline{TO}$ | $\overline{PD}$ | Z | DC | C | 0001 1xxx | 000q quuu |
| 04h(1) | FSR | Indirect data memory address pointer | | | | | | | | xxxx xxxx | uuuu uuuu |
| 05h | GPIO | SCL(5) | SDA(5) | GP5 | GP4 | GP3 | GP2 | GP1 | GP0 | 11xx xxxx | 11uu uuuu |
| 06h | — | Unimplemented | | | | | | | | — | — |
| 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 | GPIE | T0IF | INTF | GPIF | 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 | reserved | CHS1 | CHS0 | GO/$\overline{DONE}$ | reserved | ADON | 0000 0000 | 0000 0000 |

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

**Note 1:** 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 $\overline{MCLR}$ and Watchdog Timer Reset.

**4:** The IRP and RP1 bits are reserved on the PIC12C67X; always maintain these bits clear.

**5:** The SCL (GP7) and SDA (GP6) bits are unimplemented on the PIC12C671/672 and read as '0'.

4.2.2.1 STATUS REGISTER

The STATUS Register, shown in Register 4-1, contains the arithmetic status of the ALU, the RESET status and the bank select bits for data memory.

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

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

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

**Note 1:** Bits IRP and RP1 (STATUS<7:6>) are not used by the PIC12C67X and should be maintained clear. Use of these bits as general purpose R/W bits is NOT recommended, since this may affect upward compatibility with future products.

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

**REGISTER 4-1: STATUS REGISTER (ADDRESS 03h, 83h)**

| Reserved | Reserved | 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 |

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

bit 7: **IRP:** Register Bank Select bit (used for indirect addressing)
`1` = Bank 2, 3 (100h - 1FFh)
`0` = Bank 0, 1 (00h - FFh)
The IRP bit is reserved; always maintain this bit clear.

bit 6-5: **RP<1:0>:** Register Bank Select bits (used for direct addressing)
`11` = Bank 3 (180h - 1FFh)
`10` = Bank 2 (100h - 17Fh)
`01` = Bank 1 (80h - FFh)
`00` = Bank 0 (00h - 7Fh)
Each bank is 128 bytes. The RP1 bit is reserved; always maintain this bit clear.

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

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

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

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

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

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

4.2.2.5 PIR1 REGISTER

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

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

**REGISTER 4-5: PIR1 REGISTER (ADDRESS 0Ch)**

| U-0 | R/W-0 | U-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|---|---|---|---|---|---|---|---|
| — | ADIF | — | — | — | — | — | — |
| bit7 | | | | | | | bit0 |

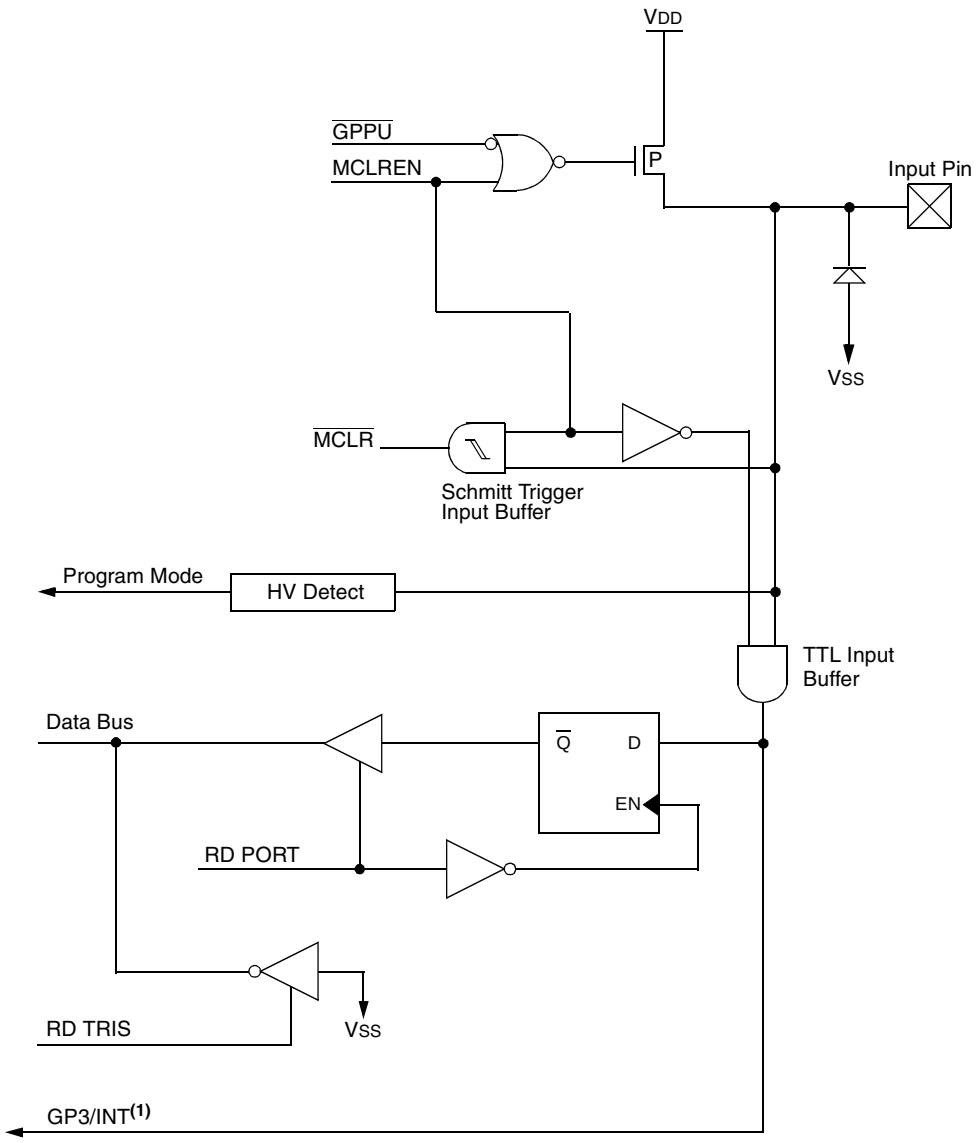
R = Readable bit
W = Writable bit
U = Unimplemented bit, read as '0'
- n = Value at POR reset

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

bit 6: **ADIF:** A/D Converter Interrupt Flag bit
1 = An A/D conversion completed (must be cleared in software)
0 = The A/D conversion is not complete

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

**FIGURE 5-3: BLOCK DIAGRAM OF GP3/$\overline{\text{MCLR}}$/VPP PIN**




**Note 1:** Wake-up on pin change interrupt for GP3.

## 6.5 Read Operations

Read operations are initiated in the same way as write operations with the exception that the R/$\overline{W}$ bit of the EEPROM address is set to one. There are three basic types of read operations; current address read, random read and sequential read.

### 6.5.1 CURRENT ADDRESS READ

The EEPROM contains an address counter that maintains the address of the last word accessed, internally incremented by one. Therefore, if the previous read access was to address n, the next current address read operation would access data from address n + 1. Upon receipt of the EEPROM address with the R/$\overline{W}$ bit set to one, the EEPROM issues an acknowledge and transmits the 8-bit data word. The processor will not acknowledge the transfer, but does generate a stop condition and the EEPROM discontinues transmission (Figure 6-8).

### 6.5.2 RANDOM READ

Random read operations allow the processor to access any memory location in a random manner. To perform this type of read operation, first the word address must be set. This is done by sending the word address to the EEPROM as part of a write operation. After the word address is sent, the processor generates a start condition following the acknowledge. This terminates the write operation, but not before the internal address pointer is set. Then the processor issues the control byte again, but with the R/$\overline{W}$ bit set to a one. The EEPROM will then issue an acknowledge and transmits the 8-bit data word. The processor will not acknowledge the transfer, but does generate a stop condition and the EEPROM discontinues transmission (Figure 6-9). After this command, the internal address counter will point to the address location following the one that was just read.

### 6.5.3 SEQUENTIAL READ

Sequential reads are initiated in the same way as a random read, except that after the device transmits the first data byte, the processor issues an acknowledge as opposed to a stop condition in a random read. This directs the EEPROM to transmit the next sequentially addressed 8-bit word (Figure 6-10).

To provide sequential reads, the EEPROM contains an internal address pointer, which is incremented by one at the completion of each read operation. This address pointer allows the entire memory contents to be serially read during one operation.

**FIGURE 6-8: CURRENT ADDRESS READ**

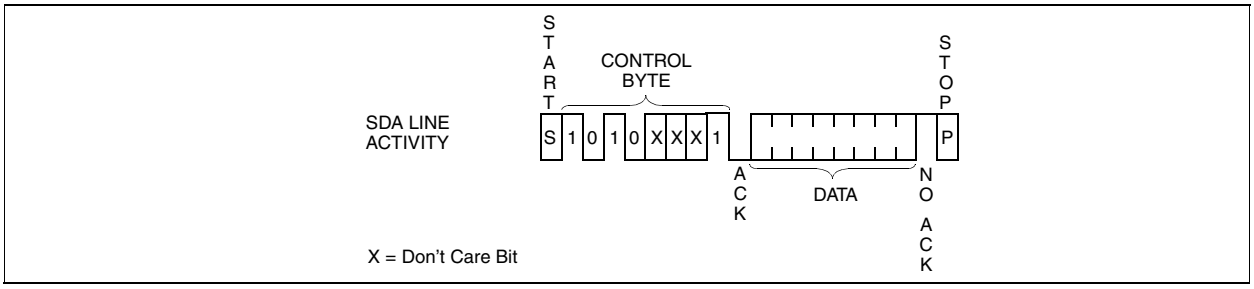



**FIGURE 6-9: RANDOM READ**

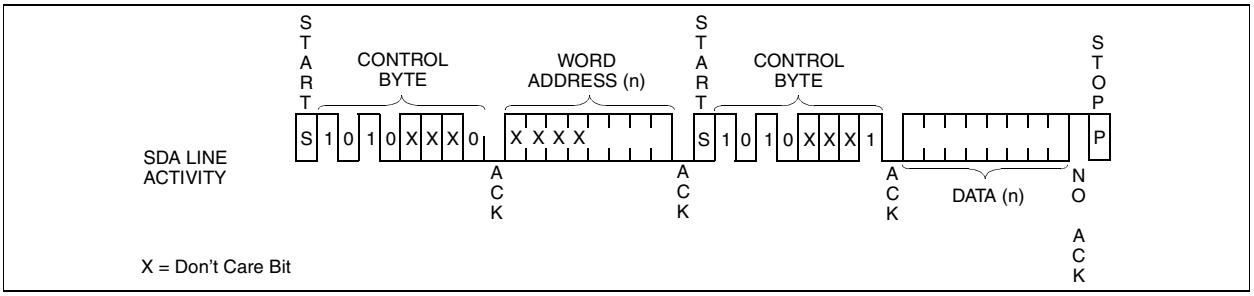
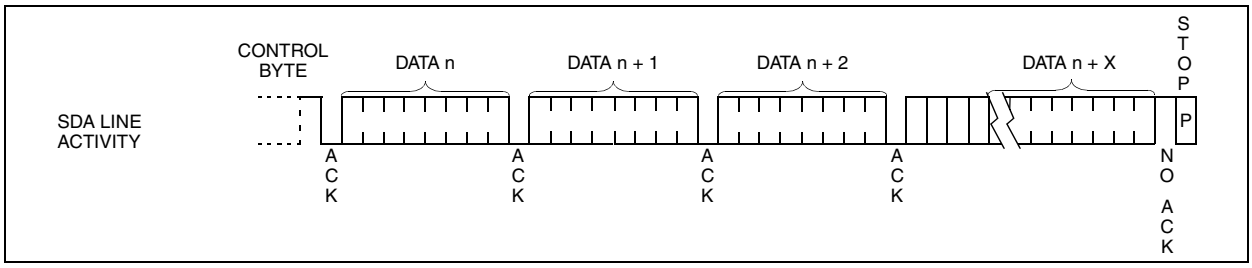



**FIGURE 6-10: SEQUENTIAL READ**



## 8.4 A/D Conversions

Example 8-2 shows how to perform an A/D conversion. The GPIO 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 GP0 channel.

| **Note:** | The GO/$\overline{\text{DONE}}$ bit should **NOT** be set in the same instruction that turns on the A/D. |
|---|---|

Clearing the GO/$\overline{\text{DONE}}$ 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 8-2: DOING AN A/D CONVERSION**

```
     BSF     STATUS, RP0         ; Select Page 1
     CLRF    ADCON1              ; Configure A/D inputs
     BSF     PIE1,   ADIE        ; Enable A/D interrupts
     BCF     STATUS, RP0         ; Select Page 0
     MOVLW   0xC1                ; RC Clock, A/D is on, Channel 0 is selected
     MOVWF   ADCON0              ;
     BCF     PIR1,   ADIF        ; Clear A/D interrupt flag bit
     BSF     INTCON, PEIE        ; Enable peripheral interrupts
     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.
```

## 9.0 SPECIAL FEATURES OF THE CPU

What sets a microcontroller apart from other processors are special circuits to deal with the needs of real-time applications. The PIC12C67X 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)
- Interrupts
- Watchdog Timer (WDT)
- SLEEP
- Code protection
- ID locations
- In-circuit serial programming

The PIC12C67X 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 INTRC/EXTRC oscillator option saves system cost, while the LP crystal option saves power. A set of configuration bits are used to select various options.

### 9.1 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.

**REGISTER 9-1: CONFIGURATION WORD**

| CP1 | CP0 | CP1 | CP0 | CP1 | CP0 | MCLRE | CP1 | CP0 | $\overline{\text{PWRTE}}$ | WDTE | FOSC2 | FOSC1 | FOSC0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| bit13 | | | | | | | | | | | | | bit0 |

| Register: | CONFIG |
|---|---|
| Address | 2007h |

bit 13-8, 6-5: **CP<1:0>:** Code Protection bit pairs(1)
- 11 = Code protection off
- 10 = Locations 400h through 7FEh code protected (do not use for PIC12C671 and PIC12CE673)
- 01 = Locations 200h through 7FEh code protected
- 00 = All memory is code protected

bit 7: **MCLRE:** Master Clear Reset Enable bit
- 1 = Master Clear Enabled
- 0 = Master Clear Disabled

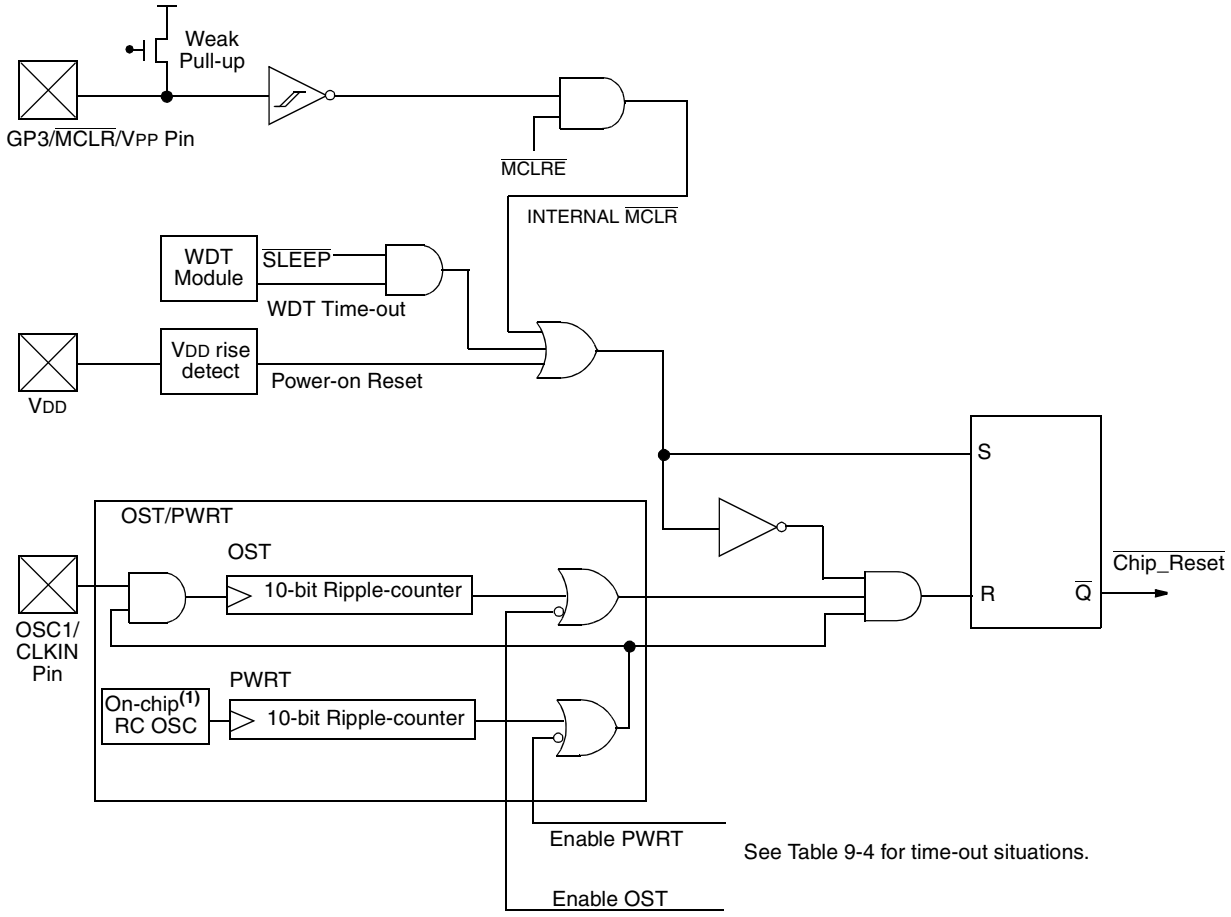
bit 4: **$\overline{\text{PWRTE}}$:** Power-up Timer Enable bit
- 1 = PWRT disabled
- 0 = PWRT enabled

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

bit 2-0: **FOSC<2:0>:** Oscillator Selection bits
- 111 = EXTRC, Clockout on OSC2
- 110 = EXTRC, OSC2 is I/O
- 101 = INTRC, Clockout on OSC2
- 100 = INTRC, OSC2 is I/O
- 011 = Invalid Selection
- 010 = HS Oscillator
- 001 = XT Oscillator
- 000 = LP Oscillator

**Note 1:** All of the CP<1:0> pairs have to be given the same value to enable the code protection scheme listed.

**FIGURE 9-6: SIMPLIFIED BLOCK DIAGRAM OF ON-CHIP RESET CIRCUIT**




**Note 1:** This is a separate oscillator from the RC oscillator of the CLKIN pin.

## 9.5 Interrupts

There are four sources of interrupt:

| Interrupt Sources |
|---|
| TMR0 Overflow Interrupt |
| External Interrupt GP2/INT pin |
| GPIO Port Change Interrupts (pins GP0, GP1, GP3) |
| 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 flag bits are set, regardless of the status of their corresponding mask bit or 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.

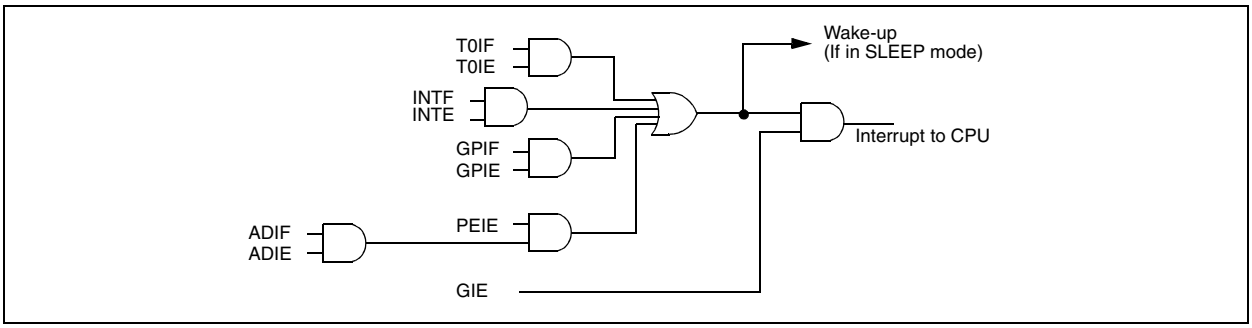
The GP2/INT, GPIO port change interrupt and the TMR0 overflow interrupt flags are contained in the INTCON register.

The peripheral interrupt flag ADIF, is contained in the Special Function Register PIR1. The corresponding interrupt enable bit is contained in Special Function Register PIE1, 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 repeated interrupts.

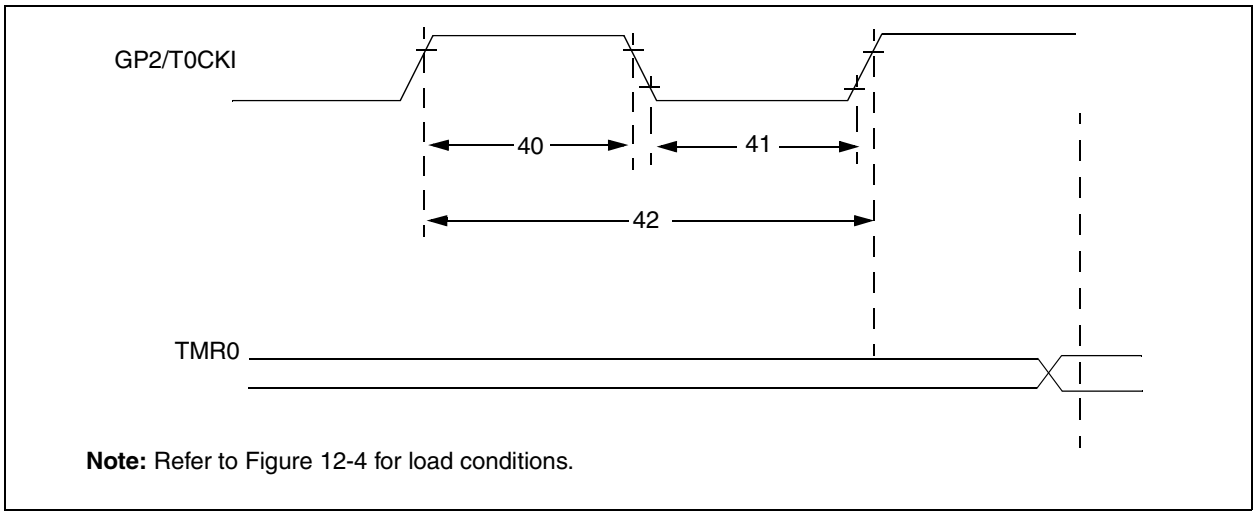
For external interrupt events, such as GPIO change interrupt, the interrupt latency will be three or four instruction cycles. The exact latency depends on when the interrupt event occurs (Figure 9-14). The latency is the same for one or two cycle instructions. Individual interrupt flag bits are set, regardless of the status of their corresponding mask bit or the GIE bit.

**FIGURE 9-13: INTERRUPT LOGIC**



**NOTES:**

**FIGURE 12-8: TIMER0 CLOCK TIMINGS**




**Note:** Refer to Figure 12-4 for load conditions.

**TABLE 12-5: TIMER0 AND TIMER1 EXTERNAL CLOCK REQUIREMENTS**

| Param No. | Sym | Characteristic | | Min | Typ† | Max | Units | Conditions |
|---|---|---|---|---|---|---|---|---|
| 40* | Tt0H | T0CKI High Pulse Width | No Prescaler | 0.5TCY + 20 | — | — | ns | Must also meet parameter 42 |
| | | | With Prescaler | 10 | — | — | ns | |
| 41* | Tt0L | T0CKI Low Pulse Width | No Prescaler | 0.5TCY + 20 | — | — | ns | Must also meet parameter 42 |
| | | | With Prescaler | 10 | — | — | ns | |
| 42* | Tt0P | T0CKI Period | No Prescaler | TCY + 40 | — | — | ns | |
| | | | With Prescaler | Greater of: 20 or TCY + 40 N | — | — | ns | N = prescale value (2, 4,..., 256) |
| 48 | TCKE2tmr1 | Delay from external clock edge to timer increment | | 2TOSC | — | 7Tosc | — | |

* These parameters are characterized but not tested.

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

**TABLE 12-6: GPIO PULL-UP RESISTOR RANGES**

| VDD (Volts) | Temperature (°C) | Min | Typ | Max | Units |
|---|---|---|---|---|---|
| GP0/GP1 | | | | | |
| 2.5 | –40 | 38K | 42K | 63K | Ω |
| | 25 | 42K | 48K | 63K | Ω |
| | 85 | 42K | 49K | 63K | Ω |
| | 125 | 50K | 55K | 63K | Ω |
| 5.5 | –40 | 15K | 17K | 20K | Ω |
| | 25 | 18K | 20K | 23K | Ω |
| | 85 | 19K | 22K | 25K | Ω |
| | 125 | 22K | 24K | 28K | Ω |
| GP3 | | | | | |
| 2.5 | –40 | 285K | 346K | 417K | Ω |
| | 25 | 343K | 414K | 532K | Ω |
| | 85 | 368K | 457K | 532K | Ω |
| | 125 | 431K | 504K | 593K | Ω |
| 5.5 | –40 | 247K | 292K | 360K | Ω |
| | 25 | 288K | 341K | 437K | Ω |
| | 85 | 306K | 371K | 448K | Ω |
| | 125 | 351K | 407K | 500K | Ω |

* These parameters are characterized but not tested.

**FIGURE 13-9: IOL vs. VOL, VDD = 5.5 V**

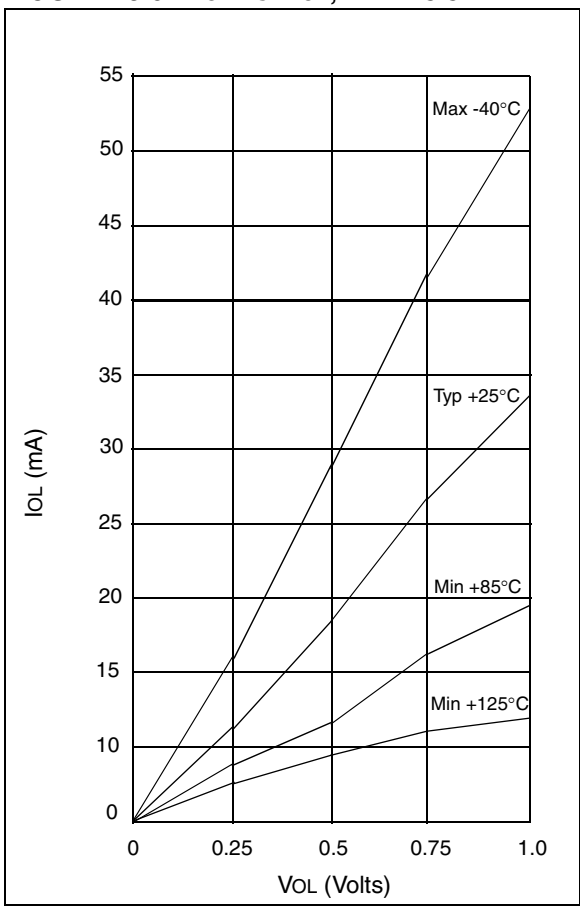
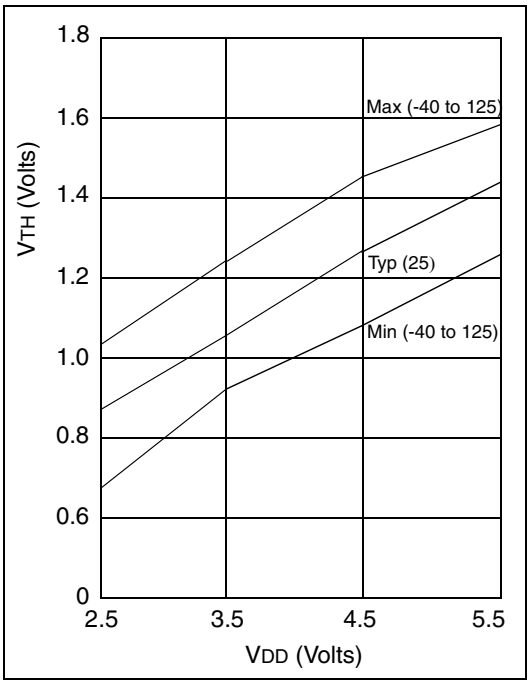
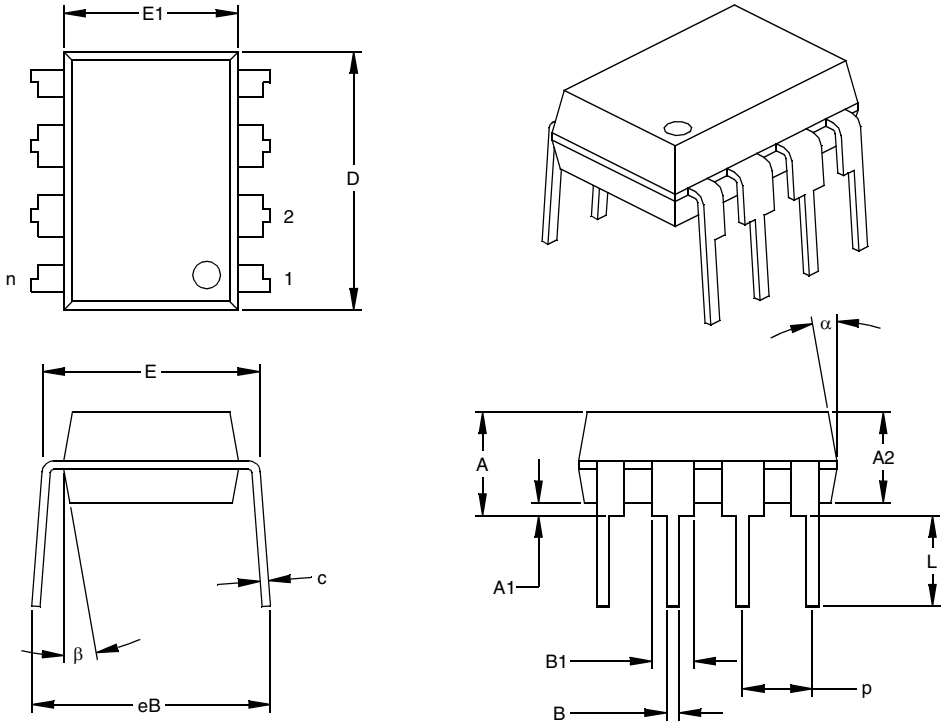



**FIGURE 13-10: VTH (INPUT THRESHOLD VOLTAGE) OF GPIO PINS vs. VDD**



### 8-Lead Plastic Dual In-line (P) – 300 mil (PDIP)

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




| Units | | INCHES* | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 8 | | | 8 | |
| Pitch | p | | .100 | | | 2.54 | |
| Top to Seating Plane | A | .140 | .155 | .170 | 3.56 | 3.94 | 4.32 |
| Molded Package Thickness | A2 | .115 | .130 | .145 | 2.92 | 3.30 | 3.68 |
| Base to Seating Plane | A1 | .015 | | | 0.38 | | |
| Shoulder to Shoulder Width | E | .300 | .313 | .325 | 7.62 | 7.94 | 8.26 |
| Molded Package Width | E1 | .240 | .250 | .260 | 6.10 | 6.35 | 6.60 |
| Overall Length | D | .360 | .373 | .385 | 9.14 | 9.46 | 9.78 |
| Tip to Seating Plane | L | .125 | .130 | .135 | 3.18 | 3.30 | 3.43 |
| Lead Thickness | c | .008 | .012 | .015 | 0.20 | 0.29 | 0.38 |
| Upper Lead Width | B1 | .045 | .058 | .070 | 1.14 | 1.46 | 1.78 |
| Lower Lead Width | B | .014 | .018 | .022 | 0.36 | 0.46 | 0.56 |
| Overall Row Spacing | eB | .310 | .370 | .430 | 7.87 | 9.40 | 10.92 |
| Mold Draft Angle Top | α | 5 | 10 | 15 | 5 | 10 | 15 |
| Mold Draft Angle Bottom | β | 5 | 10 | 15 | 5 | 10 | 15 |

*Controlling Parameter

Notes:

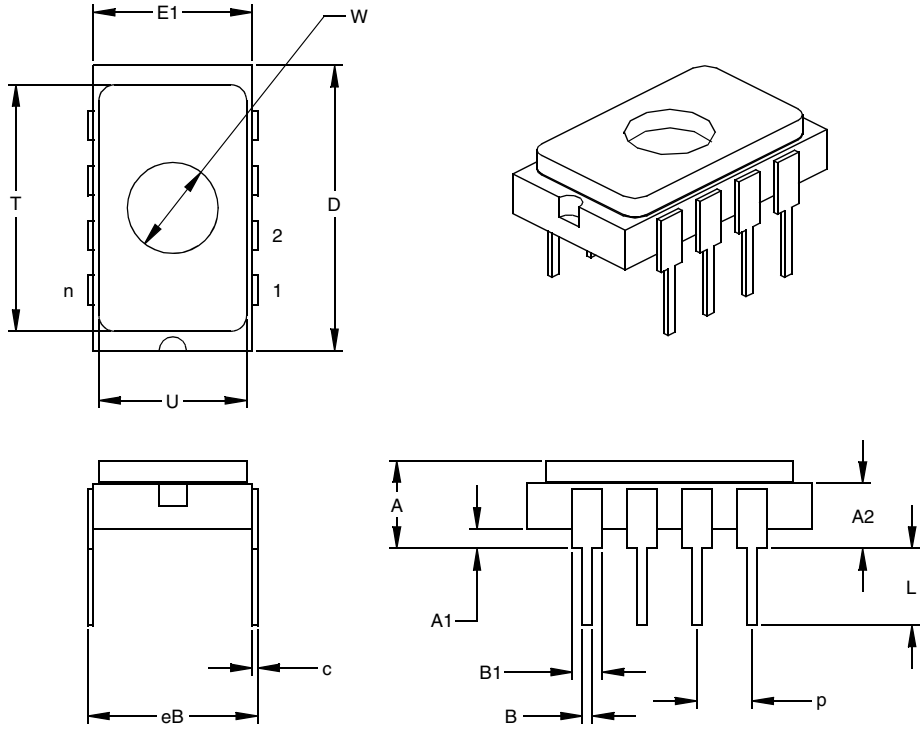
Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010” (0.254mm) per side.

JEDEC Equivalent: MS-001

Drawing No. C04-018

### 8-Lead Ceramic Side Brazed Dual In-line with Window (JW) – 300 mil

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




| Units | | INCHES* | | | MILLIMETERS | | |
|---|---|---|---|---|---|---|---|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 8 | | | 8 | |
| Pitch | p | | .100 | | | 2.54 | |
| Top to Seating Plane | A | .145 | .165 | .185 | 3.68 | 4.19 | 4.70 |
| Top of Body to Seating Plane | A2 | .103 | .123 | .143 | 2.62 | 3.12 | 3.63 |
| Standoff | A1 | .025 | .035 | .045 | 0.64 | 0.89 | 1.14 |
| Package Width | E1 | .280 | .290 | .300 | 7.11 | 7.37 | 7.62 |
| Overall Length | D | .510 | .520 | .530 | 12.95 | 13.21 | 13.46 |
| Tip to Seating Plane | L | .130 | .140 | .150 | 3.30 | 3.56 | 3.81 |
| Lead Thickness | c | .008 | .010 | .012 | 0.20 | 0.25 | 0.30 |
| Upper Lead Width | B1 | .050 | .055 | .060 | 1.27 | 1.40 | 1.52 |
| Lower Lead Width | B | .016 | .018 | .020 | 0.41 | 0.46 | 0.51 |
| Overall Row Spacing | eB | .296 | .310 | .324 | 7.52 | 7.87 | 8.23 |
| Window Diameter | W | .161 | .166 | .171 | 4.09 | 4.22 | 4.34 |
| Lid Length | T | .440 | .450 | .460 | 11.18 | 11.43 | 11.68 |
| Lid Width | U | .260 | .270 | .280 | 6.60 | 6.86 | 7.11 |

*Controlling Parameter
JEDC Equivalent: MS-015
Drawing No. C04-083

## THE MICROCHIP WEB SITE

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

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

## CUSTOMER CHANGE NOTIFICATION SERVICE

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

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

## CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

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

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

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

## READER RESPONSE

It is our intention to provide you with the best documentation possible to ensure successful use of your Microchip product. If you wish to provide your comments on organization, clarity, subject matter, and ways in which our documentation can better serve you, please FAX your comments to the Technical Publications Manager at (480) 792-4150.

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

TO: Technical Publications Manager  Total Pages Sent _____

RE: Reader Response

From: Name _____

Company _____

Address _____

City / State / ZIP / Country _____

Telephone: (_____) _____ - _____  FAX: (_____) _____ - _____

Application (optional):

Would you like a reply? ___ Y ___ N

Device: PIC16xxxxxx family  Literature Number: DS30561C

Questions:

1. What are the best features of this document?

2. How does this document meet your hardware and software development needs?

3. Do you find the organization of this document easy to follow? If not, why?

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

5. What deletions from the document could be made without affecting the overall usefulness?

6. Is there any incorrect or misleading information (what and where)?

7. How would you improve this document?