MTA81010

PICSEE™ 28-Pin MCU with Serial EEPROM Multi-Chip Module

FEATURES

- Multi-chip module
- PIC16C54 Microcontroller and 24LC01B Serial EEPROM in a single package
- 512 x 12 EPROM program memory
- 128 x 8 Serial EEPROM data memory
- Separate V_{DD} inputs for Microcontroller and Serial EEPROM
- ESD protection > 4,000V

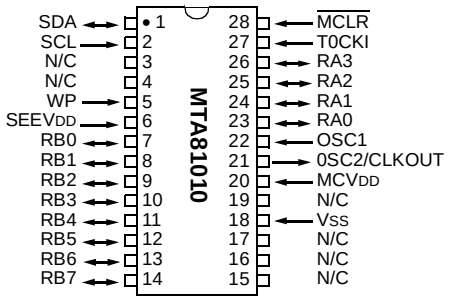
MICROCONTROLLER FEATURES

High Performance RISC-like CPU

- Only 33 single word instructions to learn
- All single-cycle instructions except for program branches which are two-cycle
- Operating speed: DC - 4 MHz clock input
- DC - 1 µs instruction cycle
- 12-bit wide instructions
- 8-bit wide data path
- 25 x 8 general purpose registers (SRAM)
- 7 special function hardware registers
- 2 level deep hardware stack
- Direct, indirect and relative addressing modes for data and instructions

PACKAGE TYPE

PDIP, SOIC



Peripheral Features

- 12 I/O pins with individual direction control
- 8-bit real time clock/counter (T0CKI) with 8-bit programmable prescaler
- Power on reset
- Oscillator start-up timer
- Watchdog timer (WDT) with its own on-chip RC oscillator for reliable operation
- Security EPROM fuse for code-protection
- Power saving SLEEP mode
- EPROM fuse selectable oscillator options:
  - Low cost RC oscillator: RC
  - Standard crystal/resonator: XT
  - Power saving low frequency crystal: LP

CMOS Technology

- Low-power, high-speed CMOS EPROM technology
- Factory programming (QTP) available for EPROM
- Fully static design
- Wide operating voltage range:
  - Commercial: 2.5V to 6.25V
  - Industrial: 2.5V to 6.25V
- Low power consumption
  - < 2 mA typical @ 5V, 4 MHz
  - 15 µA typical @ 3V, 32 kHz
  - < 3 µA typical standby current @ 3V, 0°C to 70°C

SERIAL EEPROM FEATURES

- Single supply with operation down to 2.5 volts
- Low power CMOS technology
  - 1 mA active current typical
  - 10 µA standby current typical at 5.5V
  - 5 µA standby current typical at 3.0V
- Organized as a single block of 128 bytes (128 x 8)
- Two-wire serial interface bus
- Schmitt trigger, filtered inputs for noise suppression
- Output slope control to eliminate ground bounce
- 100 kHz and 400 kHz compatibility
- Self-timed write cycle (including auto-erase)
- Page-write buffer for up to 8 bytes
- 2 ms typical write cycle time for page-write
- Hardware write protect for entire memory
- Can be operated as a serial ROM
- 1,000,000 erase/write cycles (typical)
- Data retention > 40 years

MTA81010 PIN DESCRIPTIONS

Name	Function	Description
RA3: RA0	I/O PORT A	4 input/output lines.
RB7: RB0	I/O PORT B	8 input/output lines.
T0CKI	Clock input to TMR0 register	Schmitt Trigger Input.
		Clock input to T0CKI register. Must be tied to V_{SS} or V_{DD} if not in use to avoid unintended entering of test modes and to reduce current consumption.
$\overline{MCLR}$	Master Clear	Schmitt Trigger Input.
		A “Low” voltage on this input generates a RESET for the microcontroller.
		A rising voltage triggers the on-chip oscillator start-up timer which keeps the chip in RESET mode for about 18 ms. This input must be tied directly, or via a pull-up resistor, to V_{DD}.
OSC1	Oscillator (input)	XT and LP devices: Input terminal for crystal, ceramic resonator, or external clock generator.
		RC devices: Driver terminal for external RC combination to establish oscillation.
OSC2/CLKOUT	Oscillator (output)	For XT and LP devices: Output terminal for crystal and ceramic resonator. Do not connect any other load to this output. Leave open if external clock generator is used.
		For RC devices: A CLKOUT signal with a frequency of 1/4 $F_{OSC}1$ is put out on this pin.
MCV_{DD}	Power supply	
V_{SS}	Ground	
SDA	Serial Address/Data I/O	This is a bidirectional pin used to transfer addresses and data into and data out of the device. It is an open drain terminal.
		For normal data transfer SDA is allowed to change only during SCL low. Changes during SCL high are reserved for indicating the START and STOP.
SCL	Serial Clock	This input is used to synchronize the data transfer from and to the device.
WP	Write Protect Input	This pin must be connected to either V_{SS} or $SEEV_{DD}$.
		If tied to V_{SS}, normal memory operation is enabled (read/write the entire memory 00-7F or 00-FF).
		If tied to $SEEV_{DD}$, WRITE operations are inhibited. The entire memory will be write-protected. Read operations are not affected.
		This feature allows the user to use the 24LC01B as a serial ROM when WP is enabled (tied to $SEEV_{DD}$).
$SEEV_{DD}$	+2.5V to 5.5V Power Supply	
N/C	No (internal) Connection	

Table of Contents **Page**

1.0 GENERAL DESCRIPTION

The MTA81010 from Microchip Technology Inc. offers the unique combination of a EPROM-based microcontroller and a Serial EEPROM in a single package. It is a multi-chip module that combines a PIC16C54, low cost, high performance, 8-bit, fully static, EPROM-based CMOS microcontroller with a 24LC01B, a 1K-bit Serial Electrically Erasable PROM. Combining these two popular chips into a single package reduces system cost, board area, and inventory.

The microcontroller and EEPROM are electrically independent, sharing only a common ground (V_{SS}). Independent power sources are valuable in power-conscious applications where it may be desirable to power down the EEPROM when it is not being accessed. One of the microcontroller's output pins can supply power directly to the internal EEPROM's power pin thus avoiding the use of external power switching components.

The microcontroller and serial EEPROM are exactly equivalent to their respective individual chips, the PIC16C54 and 24LC01B.

1.1 Applications

The MTA81010 is ideally suited to a wide variety of applications including but not limited to; keyless entry, remote control, smart cards, and automotive controllers. The EPROM program memory makes customization of application programs fast and convenient. The EEPROM data memory is ideal for storing configuration information, access codes, serial numbers, and adaptive lookup tables. The small footprint packages available for through hole or surface mounting make MTA81010 perfect for applications with physical space limitations. Low-cost, low-power, high-performance, ease of use, I/O flexibility, and nonvolatile EEPROM memory makes the MTA81010 the micro-controller of choice for a wide variety of systems.

1.2 MTA81010 Series Overview

Depending on application and production requirements the proper device option can be selected using the table in this section. When placing orders, please use the "MTA81010 Product Identification System" on the back page of this data sheet to specify the correct part number.

1.2.1 ONE-TIME-PROGRAMMABLE (OTP) DEVICES

The availability of OTP devices is especially useful for customers expecting frequent code changes and updates. OTP devices have the oscillator type pre-configured by the factory, and they are tested only for this special configuration (including voltage and frequency ranges, current consumption).

The program EPROM is erased, allowing the user to write the application code into it. In addition, the watchdog timer can be disabled, and/or the code protection logic can be activated by programming special EPROM fuses. The sixteen special EPROM bits for ID code storage are also user programmable.

1.2.2 QUICK-TURNAROUND-PRODUCTION (QTP) DEVICES

Microchip offers a QTP Programming Service for factory production orders. This service is made available for users who chose 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 fuse options already programmed by the factory. Certain code and prototype verification procedures do apply before production shipments are available. Please contact your Microchip Technology sales office for more details.

TABLE 1-1: OVERVIEW

Part #	EPROM	EEPROM	RAM*	I/O†	Package Options
MTA81010	512 x 12	128 x 8	32 x 8	13	28L PDIP (600 mill), 28L SOIC (300 mil)

* Including special function registers.
† Includes T0CKI pin.

2.0 MICROCONTROLLER SECTION

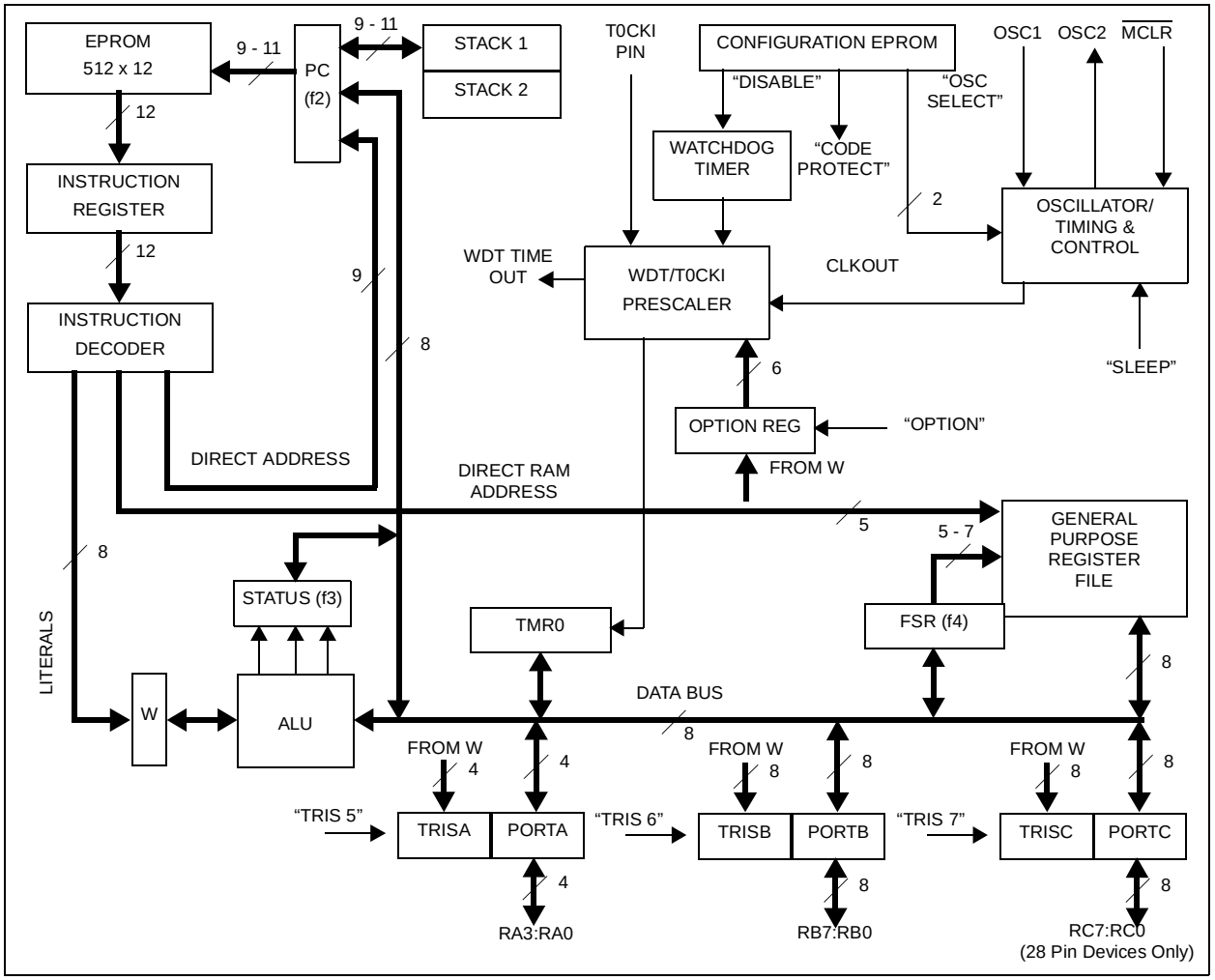
The MTA81010 from Microchip Technology contains low-cost, high-performance, 8-bit, fully static, EPROM-based CMOS microcontroller. It employs a RISC-like architecture with only 33 single word/single cycle instructions to learn. All instructions are single cycle except for program branches which take two cycles. The 12-bit wide instructions are highly symmetrical resulting in 2:1 code compression over other 8-bit microcontrollers in its class. The easy to use and easy to remember instruction set reduces development time significantly.

The MTA81010 product is equipped with special features that reduce system cost and power requirements. The power on reset and oscillator start up timer eliminate the need for external reset circuitry. There are three oscillator configurations to choose from, including power saving LP (Low Power) oscillator, standard XT oscillator and cost saving RC oscillator. Power saving SLEEP mode, watchdog timer and code protection features improves system cost, power and reliability.

TABLE 2-1: MICROCONTROLLER PIN FUNCTIONS

Name	Function
RA0 - RA3	I/O PORT A
RB0 - RB7	I/O PORT B
T0CKI	Real Time Clock/Counter
$\overline{MCLR}$	Master CLear
OSC1	Oscillator (input)
OSC2/CLKOUT	Oscillator (output)
MCV_{DD}	Power Supply
V_{SS}	Ground
N/C	No (internal) Connection

FIGURE 2-1: MTA81010 MICROCONTROLLER CHIP BLOCK DIAGRAM



2.1 MICROCONTROLLER ARCHITECTURAL DESCRIPTION

2.1.1 HARVARD ARCHITECTURE

The MTA81XXX family contains a, high-speed, fully static CMOS CPU with EPROM, RAM, and I/O on a single die.

The architecture is based on a register file concept with separate bus and memories for data and instructions (Harvard architecture). The data bus and memory (RAM) are 8-bits wide while the program bus and program memory (EPROM) have a width of 12-bits. This concept allows a simple yet powerful instruction set designed to emphasize bit, byte and register operations under high speed with overlapping instruction fetch and execution cycles. That means that, while one instruction is executed, the following instruction is already being read from the program memory. A block diagram of the microcontroller core is given in Figure 2-2.

2.1.2 CLOCKING SCHEME/INSTRUCTION CYCLE

The clock input (from pin OSC1) is internally divided by four to generate four non overlapping quadrature clocks namely Q1, Q2, Q3 and Q4. Internally, PC is incremented every Q1, instruction is fetched from program memory and latched into instruction register in Q4. It is decoded and executed during the following Q1 through Q4. The clocks and instruction execution flow is shown in Figure 2-1.

2.1.3 DATA REGISTER FILE

The 8-bit data bus connects two basic functional elements together: the Register File composed of 32 addressable 8-bit registers including the I/O Ports, and an 8-bit wide Arithmetic Logic Unit. The 32 bytes of RAM are directly addressable (Figure 3-2). Data can be addressed direct or indirect using the file select register (f4). Immediate data addressing is supported by special "literal" instructions which load data from program memory into the W register.

The register file is divided into two functional groups: operational registers and general purpose registers. The operational registers include the Real Time Clock Counter (T0CKI) register, the Program Counter (PC), the Status Register, the I/O registers (PORTs), and the File Select Register. The general purpose registers are used for data and control information under command of the instructions.

In addition, special purpose registers are used to control the I/O port configuration, and the prescaler options.

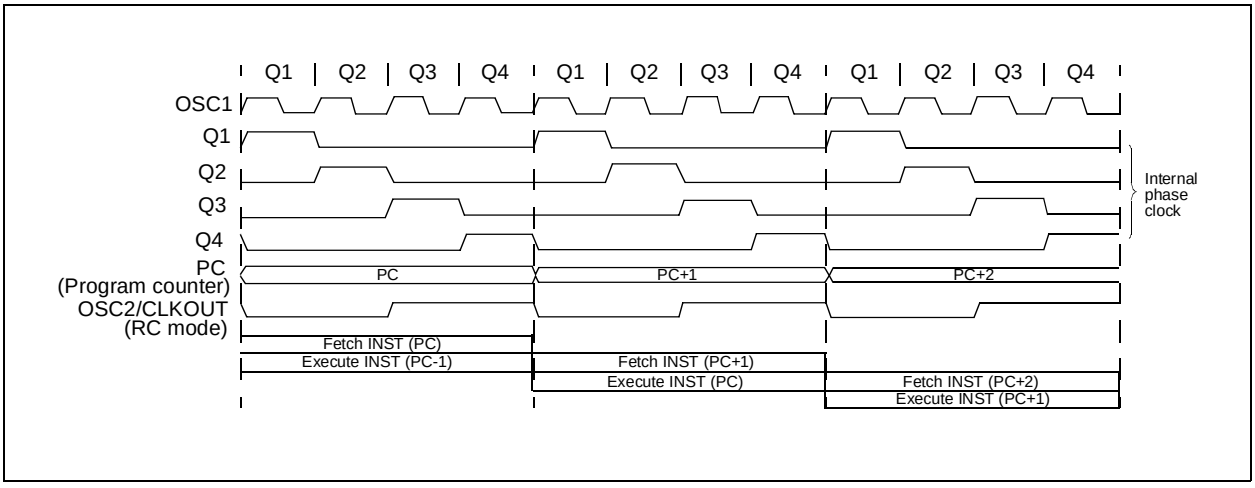
2.1.4 ARITHMETIC/LOGIC UNIT (ALU)

The 8-bit wide ALU contains one temporary working register (W Register). It performs arithmetic and Boolean functions between data held in the W Register and any file register. It also does single operand operations on either the W register or any file register.

2.1.5 PROGRAM MEMORY

512 words of 12-bit wide on-chip program memory (EPROM) can be directly addressed (Figure 3-6). Sequencing of microinstructions is controlled via the Program Counter (PC) which automatically increments to execute in-line programs. Program control operations, supporting direct, indirect, relative addressing modes, can be performed by Bit Test and Skip instructions, Call instructions, Jump instructions or by loading computed addresses into the PC. In addition, an on-chip two-level stack is employed to provide easy to use subroutine nesting.

FIGURE 2-2: CLOCKS/INSTRUCTION CYCLE



3.0 OPERATIONAL REGISTER FILES

3.1 Timer0 (TMR0) Module

The TMR0 module timer/counter has the following features:

- 8-bit timer/counter
- Readable and writable
- 8-bit software programmable prescaler
- Internal or external clock select
- Edge select for external clock

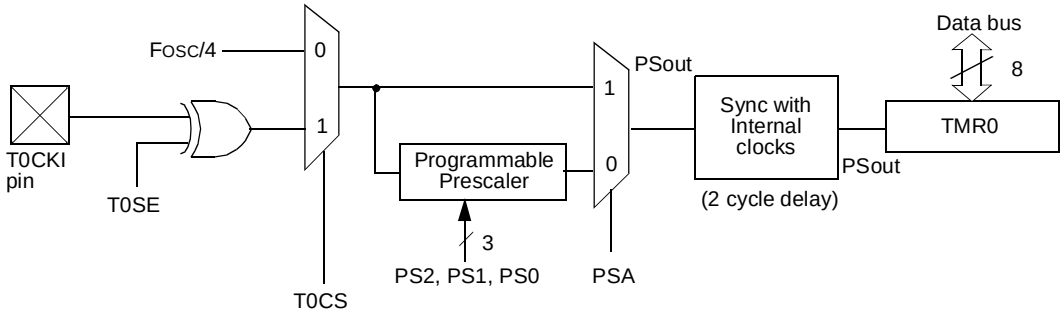
Figure 3-1 is a simplified block diagram of the TMR0 module, while Figure 3-2 shows the electrical structure of the TMR0 input.

Timer mode is selected by clearing the T0CS bit (OPTION<5>). In timer mode, the TMR0 module will increment every instruction cycle (without prescaler). If TMR0 is written, increment is inhibited for the following two cycles (Figure 3-3 and Figure 3-4). The user can work around this by writing an adjusted value to the TMR0 module.

Counter mode is selected by setting the T0CS bit (OPTION<5>). In this mode TMR0 will increment either on every rising or falling edge of pin T0CKI. The incrementing edge is determined by the T0 source edge (T0SE) select bit (OPTION<4>). Clearing the T0SE bit selects the rising edge. Restrictions on the external clock input are discussed in detail in Section 3.1.1.

The prescaler is shared between the TMR0 module and the watchdog timer. The prescaler assignment is controlled in software by the control bit PSA (OPTION<3>). Setting the PSA bit will assign the prescaler to the WDT and cause the prescale for TMR0 to be 1:1. Clearing the PSA bit will assign the prescaler to TMR0. The prescaler is neither readable nor writable. When the prescaler is assigned to the TMR0 module, prescale value of 1:2, 1:4, ..., 1:256 are selectable. Section 6.2 details the operation of the prescaler.

FIGURE 3-1: TMR0 BLOCK DIAGRAM



Note 1: Bits, T0CS, T0SE, PSA, PS2, PS1 and PS0 are located in the OPTION register.
2: The prescaler is shared with Watchdog Timer (Figure 6-6).

FIGURE 3-2: ELECTRICAL STRUCTURE OF T0CKI PIN

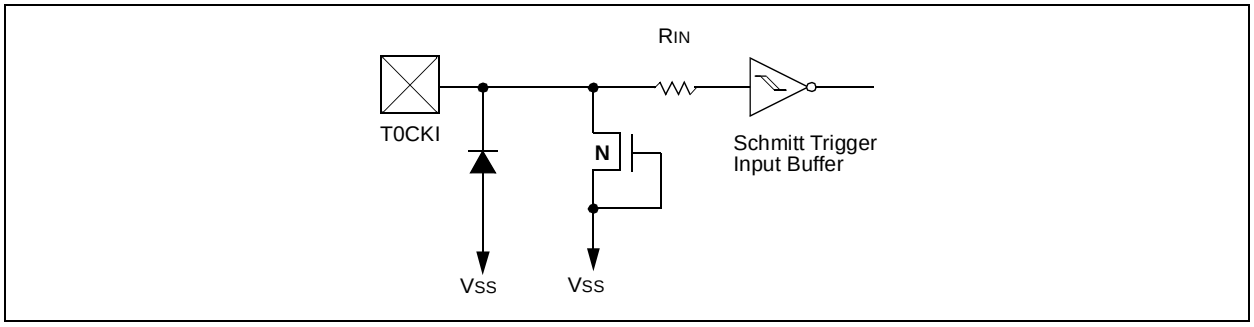


FIGURE 3-3: TMR0 TIMING: INTERNAL CLOCK/NO PRESCALE

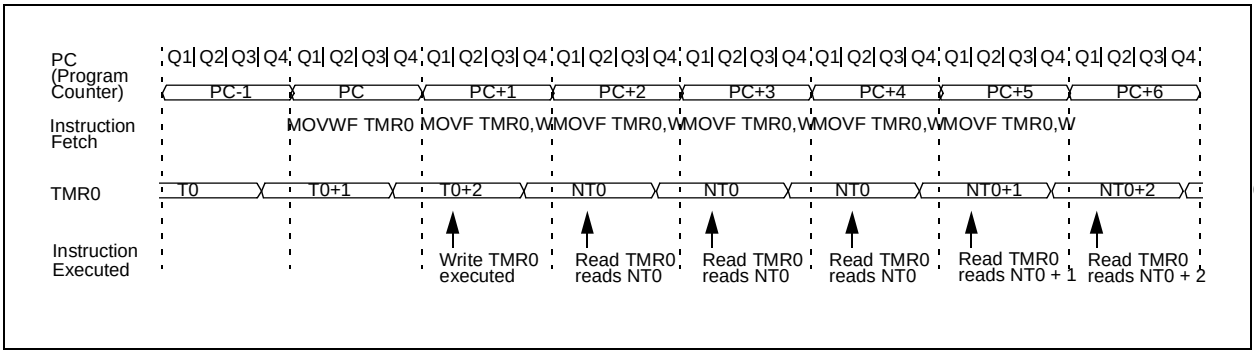
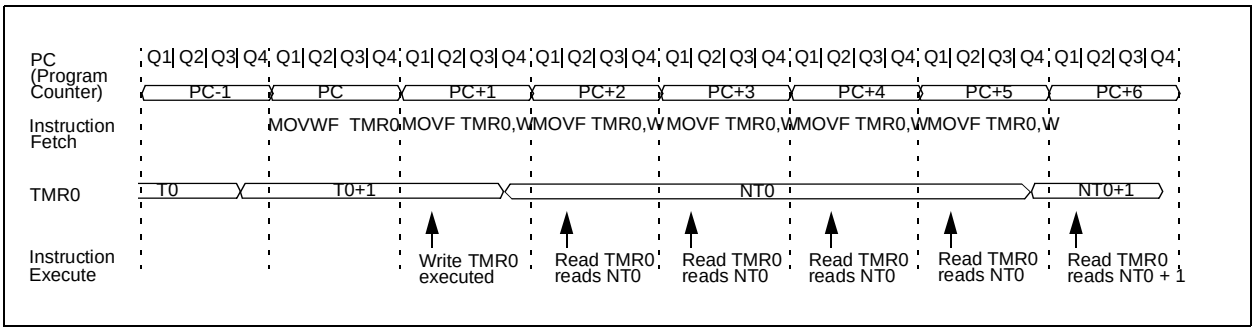


FIGURE 3-4: TMR0 TIMING: INTERNAL CLOCK/PRESCALE 1:2



3.1.1 USING TMR0 WITH EXTERNAL CLOCK

When an external clock input is used for TMR0, it must meet certain requirements to be able to synchronize with the internal phase clock (TOSC). Also, there is a delay in the actual incrementing of TMR0 after synchronization.

3.1.1.1 EXTERNAL CLOCK SYNCHRONIZATION

When no prescaler is used, the external clock input is the same as the prescaler output. Synchronizing T0CKI with the internal phase clocks requires sampling the prescaler output on the Q2 and Q4 cycles of the internal phase clocks (Figure 3-5). Therefore, it is necessary for T0CKI to be high for at least 2 TOSC (plus a small RC delay) and low for at least 2 TOSC (plus a small RC delay). Refer to the appropriate electrical specification table.

When a prescaler is used, the external clock input is divided by the asynchronous ripple-counter type prescaler to ensure 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 4 TOSC (plus a small RC delay) divided by the prescaler value. The only limitation on T0CKI high and low time is that they are greater than the minimum pulse width requirement of 10 ns. Refer to parameters 40, 41 and 42 in the appropriate electrical specification section.

3.1.1.2 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 TMR0 module is actually incremented. Figure 3-5 shows the delay from the external clock edge to the timer incrementing.

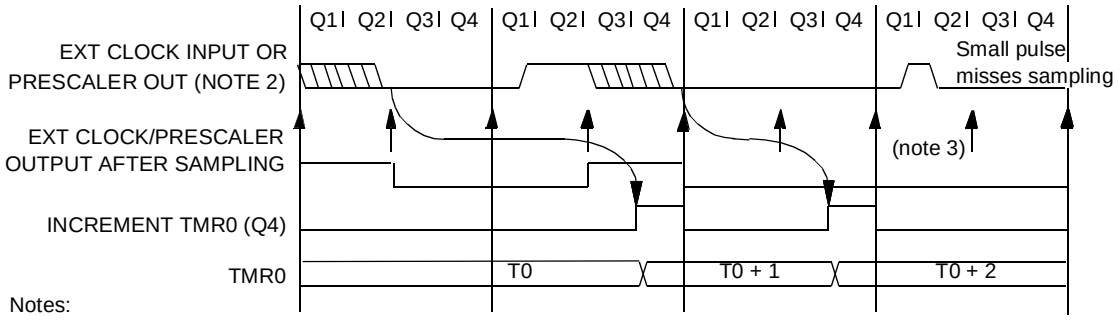
3.1.2 PRESCALER

An 8-bit counter is available as a prescaler for the TMR0 module, or as a post-scaler for the Watchdog Timer, (Section 3.1.1.2). For simplicity, this counter is being referred to as “prescaler” throughout this data sheet. Note that there is only one prescaler available which is mutually exclusive between the TMR0 module and the Watchdog Timer. Thus, a prescaler assignment for the TMR0 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 TMR0 module, all instructions writing to the TMR0 module (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 neither readable nor writable. On a RESET, the prescaler contains all '0's.

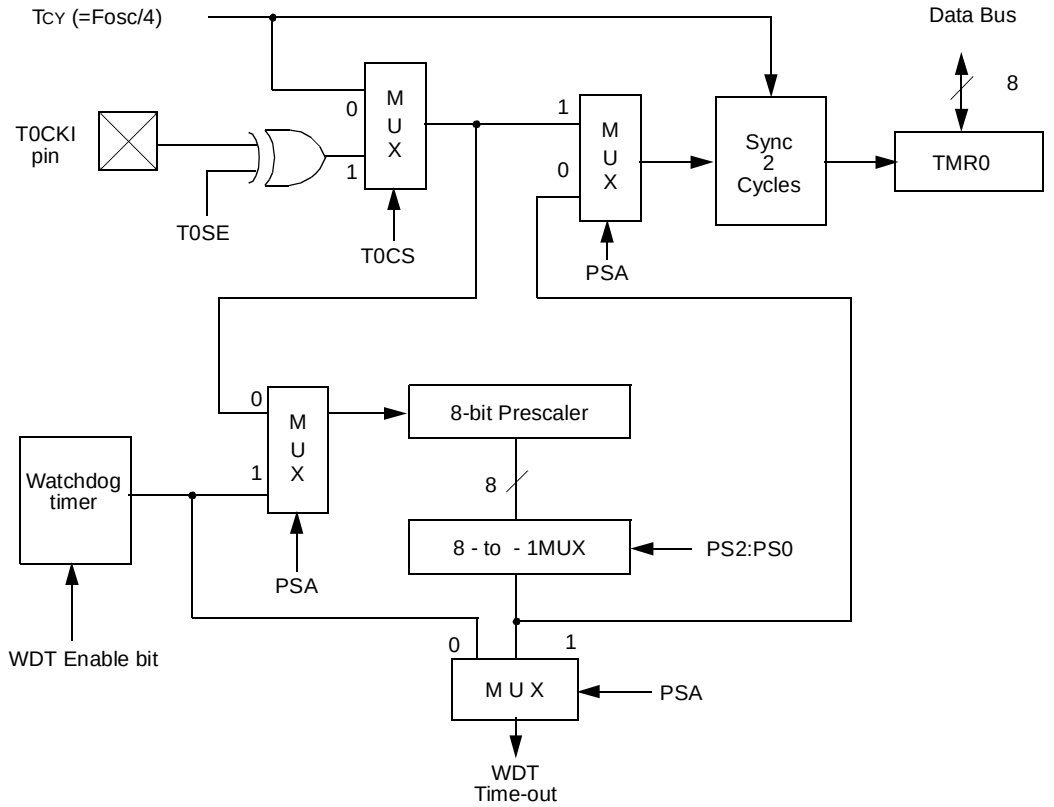
FIGURE 3-5: TIMER0 TIMING WITH EXTERNAL CLOCK



Notes:

1. Delay from clock input change to TMR0 increment is 3 Tosc to 7 Tosc. (Duration of Q = tosc). Therefore, the error in measuring the interval between two edges on TMR0 input = ± 4 Tosc max.
2. External clock if no prescaler selected, Prescaler output otherwise.
3. The arrows indicate the points in time where sampling occurs.

FIGURE 3-6: BLOCK DIAGRAM OF THE TMR0/WDT PRESCALER



Note: T0SE, T0CS, PSA, PS2:PS0 are bits in the OPTION register.

3.1.2.1 SWITCHING PRESCALER ASSIGNMENT

The prescaler assignment is fully under software control (i.e., it can be changed “on the fly” during program execution). To avoid an unintended device RESET, the following instruction sequence (Example 3-1) must be executed when changing the prescaler from TMR0 to WDT.

EXAMPLE 3-1: CHANGING PRESCALER (TMR0→WDT)

```
CLRF   TMR0          ;Clear TMR0
CLRWDT               ;Clears WDT and
                     ;prescaler
MOVLW  'xxxx1xxx'b   ;Select new prescale
OPTION               ;value
```

To change prescaler from the WDT to the TMR0 module, use the sequence shown in Example 3-2. This sequence must be used even if the WDT is disabled.

Note that a `CLRWDT` instruction should be executed before switching the prescaler.

EXAMPLE 3-2: CHANGING PRESCALER (WDT→TMR0)

```
CLRWDT                ;Clear WDT and
                      ;prescaler
MOVLW     'xxxx0xxx'b ;Select TMR0, new
                      ;prescale value and
                      ;clock source
OPTION
```

TABLE 3-1: SUMMARY OF TMR0 REGISTERS

Register Name	Function	Address	Power-on Reset Value
TMR0	Timer/counter register	01h	xxxx xxxx
OPTION	Configuration and prescaler assignment bits for TMR0. (Figure 4-4)	N/A	--11 1111

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

Note: For reset values of registers in other reset situations, refer to Table 4-1.

TABLE 3-2: REGISTERS ASSOCIATED WITH TMR0

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
01	TMR0	Timer0 - 8-bit real-time clock counter							
N/A	OPTION	—	—	T0CS	T0SE	PSA	PS2	PS1	PS0

Legend: — = Unimplemented, read as '0'.

Note: Shaded cells are not used by TMR0 module.

3.2 Memory Organization

3.2.1 PROGRAM MEMORY ORGANIZATION

Up to 512 words of 12-bit wide on-chip program memory (EPROM/ROM) can be directly addressed. Larger program memories can be addressed by selecting one of up to four available pages of 512 words each (Figure 3-7). Sequencing of instructions is controlled via the Program Counter (PC) which automatically increments to execute in-line programs. Program control operations supporting direct, indirect, and relative addressing modes, can be performed by bit test and skip instructions, call instructions, jump instructions or by loading computed addresses into the PC. In addition, an on-chip two-level stack is employed to provide easy to use subroutine nesting.

3.2.2 DATA MEMORY ORGANIZATION

The 8-bit data bus connects two basic functional elements together: the Register File composed of up to 80 addressable 8-bit registers including the I/O Ports, and an 8-bit wide Arithmetic Logic Unit. 32 bytes of RAM are directly addressable while a “banking” scheme, with banks of 16 bytes each, is employed to address larger data memories (Figure 3-8). Data can be addressed directly, or indirectly using the file select register. Immediate data addressing is supported by special “literal” instructions which load data from program memory into the W register.

The register file is divided into two functional groups: Special Function registers and General Purpose registers. The special function registers include the Timer0 (TMR0) register, the Program Counter (PC), the Status Register, the I/O registers (ports), and the File Select Register (FSR). The general purpose registers are used for data and control information under command of the instructions.

In addition, special purpose registers are used to control the I/O port configuration and prescaler options.

3.2.2.1 GENERAL PURPOSE REGISTER FILE

The register file is accessed either directly or indirectly through the file select register FSR (Section 3.2.5).

3.2.2.2 SPECIAL FUNCTION REGISTERS

The Special Function Registers are registers used by the CPU and peripheral functions to control the operation of the device (Table 3-3).

The special registers can be classified into two sets. The special registers associated with the “core” functions are described in this section. Those related to the operation of the peripheral features are described in the section for each peripheral feature.

FIGURE 3-7: PROGRAM MEMORY ORGANIZATION

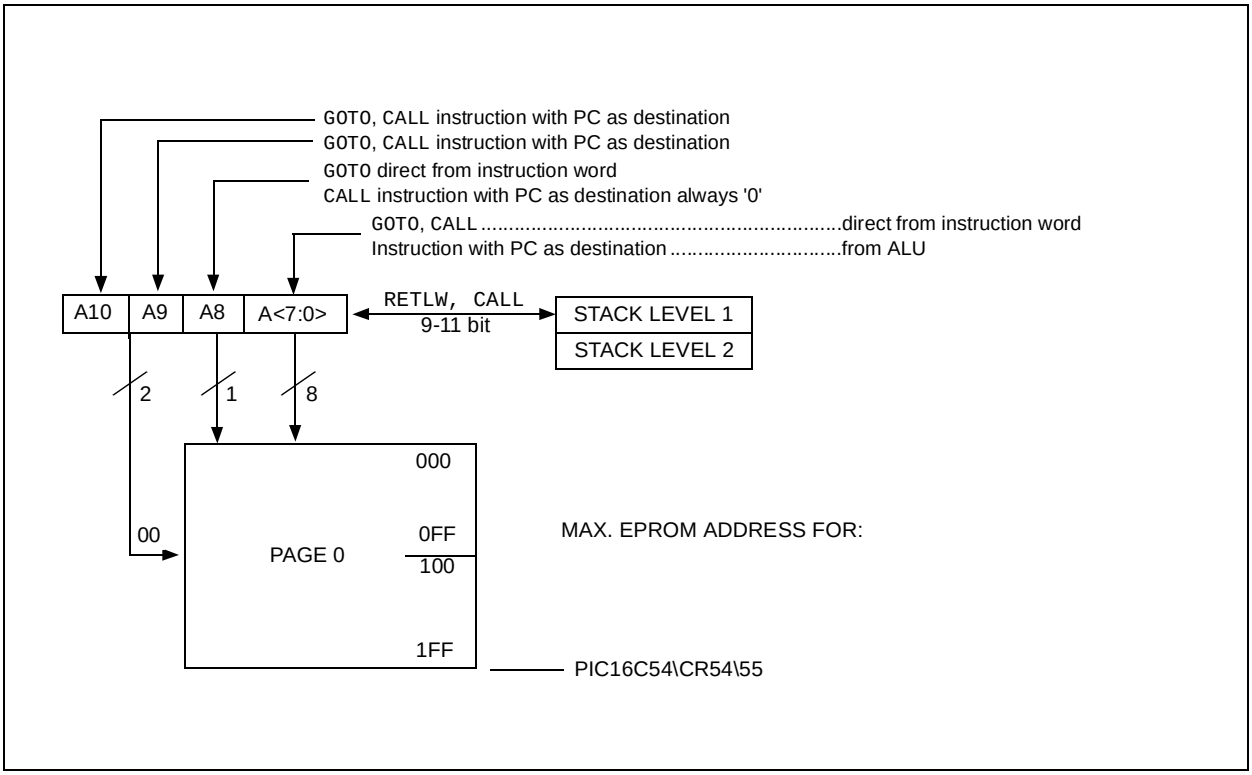
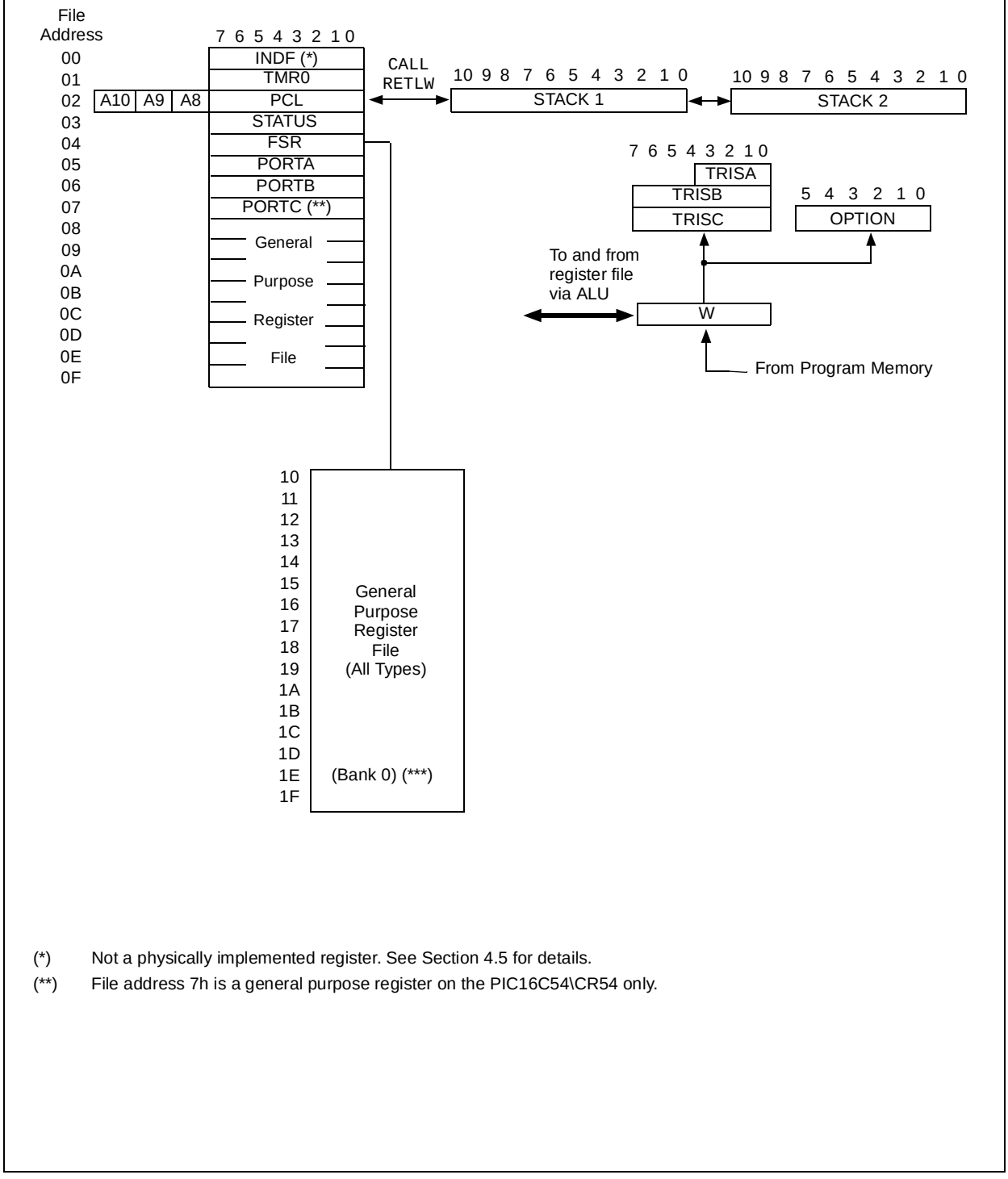


FIGURE 3-8: DATA MEMORY MAP



(*) Not a physically implemented register. See Section 4.5 for details.

(**) File address 7h is a general purpose register on the PIC16C54\CR54 only.

TABLE 3-3: MTA81010 REGISTER FILE 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 $\overline{MCLR}$ and WDT Reset
00h	INDF	Uses contents of FSR to address data memory (not a physical register)								---- ----	---- ----
01h	TMR0	8-bit real-time clock/counter								xxxx xxxx	uuuu uuuu
02h	PCL	Low order 8 bits of PC								1111 1111	1111 1111
03h	STATUS	PA2	PA1	PA0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	000? ?uuu
04h	FSR	Indirect data memory address pointer 0								xxxx xxxx	uuuu uuuu
05h	PORTA	—	—	—	—	RA3	RA2	RA1	RA0	---- xxxx	---- uuuu
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
07h	PORTC (Note 2)	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu

Legend: x = unknown, u = unchanged. - = Unimplemented, Read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. The upper bits can be set or cleared by writing to PA1: PA0 (STATUS<6:5>).

2: File address 7h is a general purpose register on the PIC16C54\CR54.

3: Shading indicates unimplemented bits.

3.2.3 STATUS REGISTER

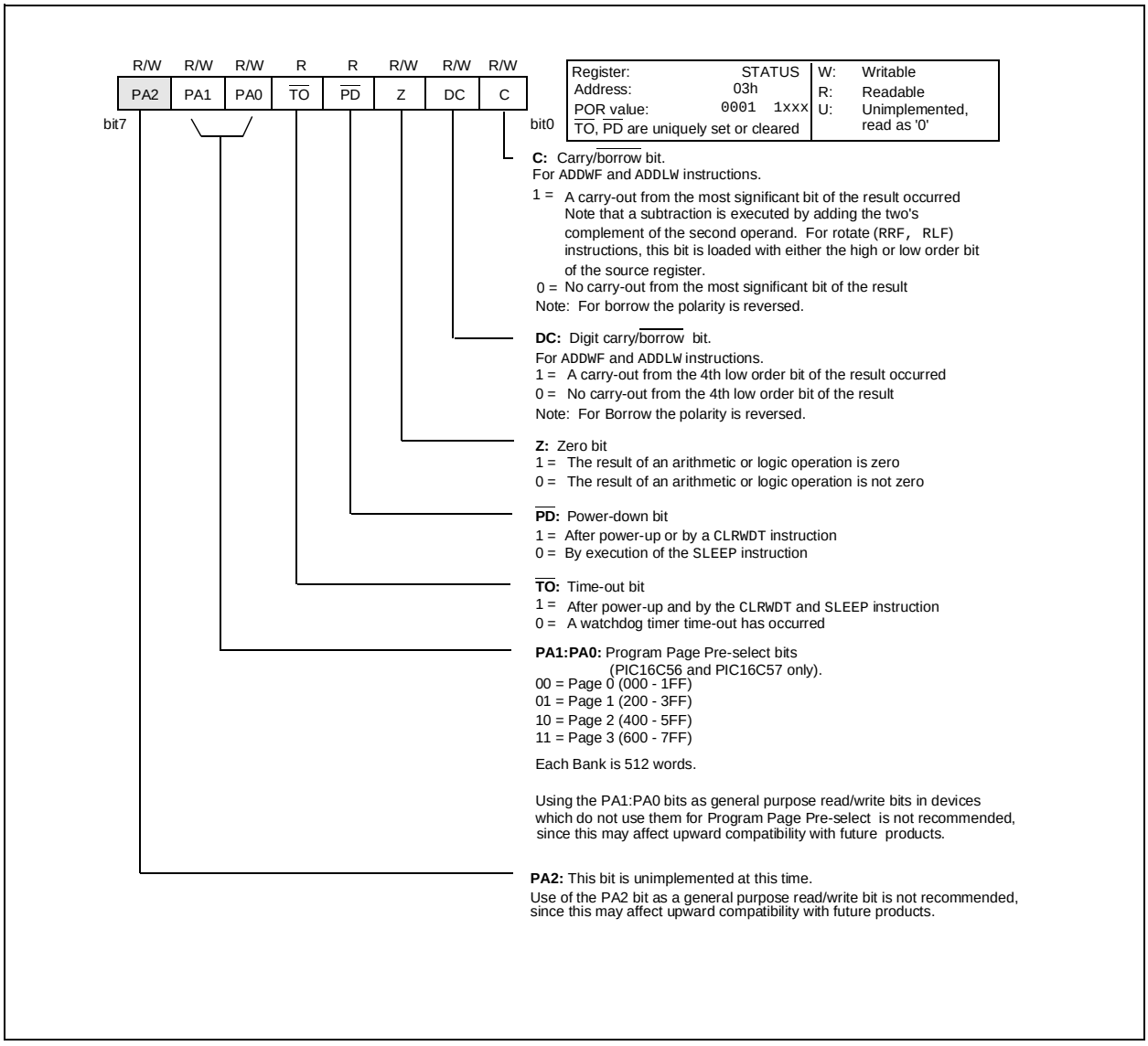
This register contains the arithmetic status of the ALU, the RESET status, and the page preselect bits for program memories larger than 512 words .

As with any other register, the STATUS register can be the destination for any instruction. However, the STATUS bits are set after the following write. Furthermore, the $\overline{TO}$ and $\overline{PD}$ bits are not writable. Therefore, the result of an instruction with STATUS register as destination may be different than intended. For example, `CLRF STATUS` will clear all bits except for $\overline{TO}$ and $\overline{PD}$ and then set the Z bit and leave status register as `000u u100` (where u = unchanged).

Therefore, only `BCF`, `BSF` and `MOVWF` instructions should be used to alter the STATUS register because these instructions do not affect any STATUS bit.

For other instructions, which do affect STATUS bits, see Table 9-1, Instruction Set Summary.

FIGURE 3-9: STATUS REGISTER

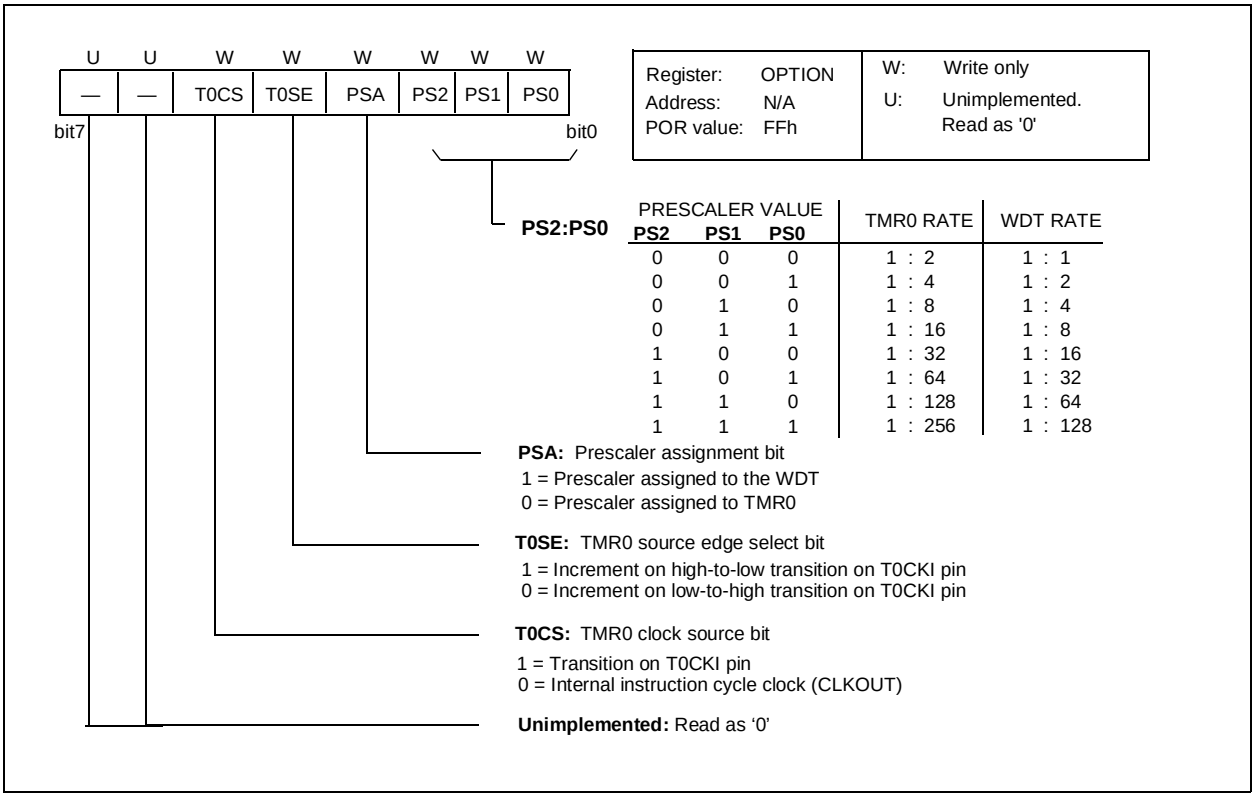


3.2.4 OPTION REGISTER

The OPTION register is a 6-bit wide, write-only register which contains various control bits to configure the TMR0/WDT prescaler, the external INT interrupt and TMR0.

By executing the `OPTION` instruction, the contents of the W register will be transferred to the OPTION register. A RESET sets the OPTION register to all '1's.

FIGURE 3-10: OPTION REGISTER



3.2.5 INDIRECT DATA ADDRESSING, INDF AND FSR REGISTERS

The INDF register is not a physical register and is used in conjunction with the FSR register to perform indirect addressing.

Indirect addressing is possible by using the INDF register. Any instruction using the INDF register actually accesses data pointed to by the file select register (FSR). Reading INDF itself (i.e., FSR = 0) indirectly will produce 00h. Writing to the INDF register indirectly results in a no-operation (although STATUS bits may be affected).

A simple program to clear RAM location 20h-2Fh using indirect addressing is shown in Example 3-3.

EXAMPLE 3-3: INDIRECT ADDRESSING

```
        movlw   0x10    ; Initialize pointer
        movwf   FSR     ; to RAM
Next    clrf    INDF    ; Clear loc
        incf    FSR     ; increment pntr
        btfsc   FSR,4   ; All done?
        goto    Next    ; No, clear next
                        ; Location
       .
       .
       .
```

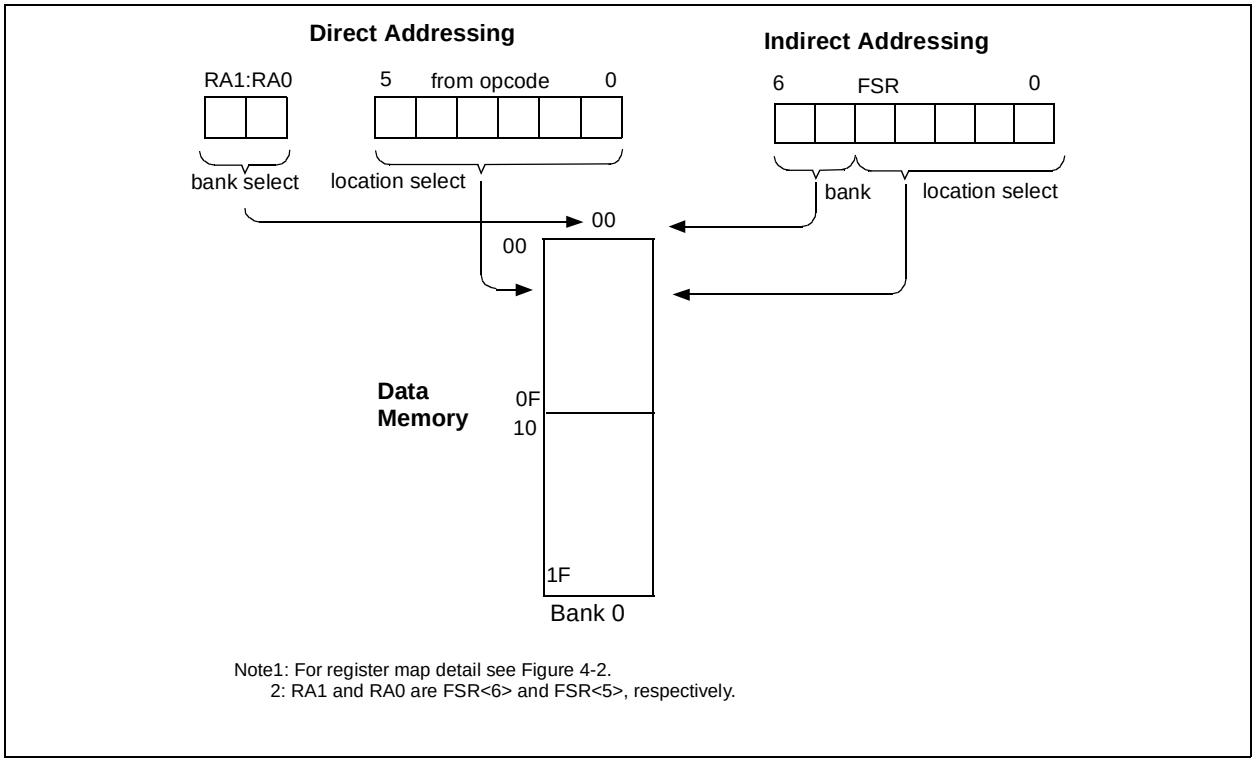
3.2.5.1 FILE SELECT REGISTER (FSR)

The FSR is either a 5-bit (PIC16C54\CR54) wide register. It is used in conjunction with the INDF register to indirectly address the data memory area. The FSR<4:0> bits are the pointer for data memory addresses 00h to 1Fh. FSR<4:0> toggles between the 16 lower (ooh-0Fh) and 16 upper (10h-1Fh) register files. When clear, FSR<4> points to the lower 16 register files and, when set, points to the upper 16 register files. FSR<3:0> provide the value to address the specific register file within each 16 file area.

When not performing indirect addressing, the FSR can be used as a 5-bit (FSR<4:0>) wide general purpose register. However, this is not recommended to help ensure future upward code compatibility.

PIC16C54\CR54: Does not use banking. FSR<7:5> are unimplemented and read as '1's.

FIGURE 3-11: DIRECT/INDIRECT ADDRESSING



Note1: For register map detail see Figure 4-2.
2: RA1 and RA0 are FSR<6> and FSR<5>, respectively.

3.2.6 PROGRAM COUNTER

The program counter generates the addresses for up to 2048 x 12 on-chip EPROM/ROM cells containing the program instruction words (Figure 3-7).

Depending on the device type, the program counter and its associated two-level hardware stack is 9-bits or 11-bits wide.

TABLE 3-4: PROGRAM COUNTER STACK WIDTH

Part #	PC width	Stack width
PIC16C54\CR54	9-bit	9-bit

The program counter is set to all '1's upon RESET. During program execution it is auto-incremented with each instruction unless the result of that instruction changes the PC itself.

a) The GOTO instruction allows the direct loading of the lower nine program counter bits (PC<8:0>). In the case where the program memory is greater than 512 bytes, the upper two bits of PC (PC<10:9>) are loaded with page select bits PA1:PA0 (STATUS<6:5>). Thus, GOTO allows jumps to any location on any page.

b) The CALL instruction loads the lower 8-bits of the PC directly, while the ninth bit is cleared to '0'. The PC value, incremented by one, will be pushed into the stack. In cases where the program memory is greater than 512 bytes, the upper 2-bits of PC (PC<10:9>) are loaded with Page Select bits PA1:PA0 (STATUS<6:5>).

c) The RETLW instruction loads the program counter with the Top Of Stack (TOS) contents.

d) If the PC is the destination in any instruction (e.g., MOVWF PC, ADDWF PC, or BSF PC,5) then the computed 8-bit result will be loaded into the lower 8-bits of program counter. The ninth bit of PC will be cleared. In case where the program memory is greater than 512 bytes, PC<10:9> will be loaded with Page Select bits PA1:PA0 (STATUS<6:5>).

It should be noted that because bit8 (ninth bit) of the PC is cleared in the CALL instruction or any instruction which writes to the PC (e.g., MOVWF PC), all subroutine calls or computed jumps are limited to the first 256 locations of any program memory page (512 words long).

Incrementing the program counter when it is pointing to the last address of a selected memory page is also possible and will cause the program to continue in the next higher page. However, the page preselect bits in the STATUS register will not be changed, and the next GOTO, CALL, ADDWF PC, or MOVWF PC instruction will return to the previous page, unless the page preselect bits have been updated under program control. For example, an NOP at location 1FFh (page 0) increments the PC to 200h (page 1). A GOTO xxx at 200h will return the program to address xxxh on page 0 (assuming that PA1:PA0 is clear).

Upon a RESET, page 0 is pre-selected while the program counter addresses the last location in the last page. Thus, a GOTO instruction at this location will automatically cause the program to continue in page 0.

3.2.7 W (WORKING) REGISTER

The W register holds the second operand in two operand instructions and/or supports the internal data transfer.

3.2.7.1 TIME OUT AND POWER DOWN STATUS BITS ($\overline{TO}$, $\overline{PD}$)

The $\overline{TO}$ and $\overline{PD}$ bits in the STATUS register can be tested to determine if a RESET condition has been caused by a Watchdog Timer time-out, a power-up condition, or a wake-up from SLEEP by the Watchdog Timer or $\overline{MCLR}$ pin.

These STATUS bits are only affected by events listed in Table 3-5.

TABLE 3-5: EVENTS AFFECTING $\overline{TO}$/$\overline{PD}$ STATUS BITS

Event	$\overline{TO}$	$\overline{PD}$	Remarks
Power-up	1	1	
WDT Timeout	0	x	No effect on $\overline{PD}$
SLEEP instruction	1	0	
CLRWDT instruction	1	1	

A WDT timeout will occur regardless of the status of the $\overline{TO}$ bit. A SLEEP instruction will be executed, regardless of the status of the $\overline{PD}$ bit. Table 3-6 reflects the status of $\overline{TO}$ and $\overline{PD}$ after the corresponding event.

TABLE 3-6: $\overline{TO}$/$\overline{PD}$ STATUS AFTER RESET

$\overline{TO}$	$\overline{PD}$	RESET was caused by
0	0	WDT wake-up from SLEEP
0	1	WDT time-out (not during SLEEP)
1	0	$\overline{MCLR}$ wake-up from SLEEP
1	1	Power-up
u	u	= Low pulse on $\overline{MCLR}$ input

The $\overline{TO}$ and $\overline{PD}$ bits maintain their status (u) until an event of Table 4-3 occurs. A low-pulse on the $\overline{MCLR}$ input does not change the $\overline{TO}$ and $\overline{PD}$ status bits.

4.0 I/O REGISTERS (PORTS)

The I/O registers can be written and read under program control like any other register of the register file. However, "read" instructions (e.g., MOVF 6,W) always read the I/O pins, regardless if a pin is defined as "input" or "output." Upon a RESET, all I/O ports are defined as "input" (= high impedance mode) as the I/O control registers (TRISA, TRISB) are all set to "ones."

The execution of a "TRIS f" instruction with corresponding "zeros" in the W-register is necessary to define any of the I/O pins as output.

4.1 f5 (Port A)

4-bit I/O register. Low order 4 bits only are used (RA3:RA0). Bits 4-7 are unimplemented and read as "zeros."

4.2 f6 (Port B)

8-bit I/O register.

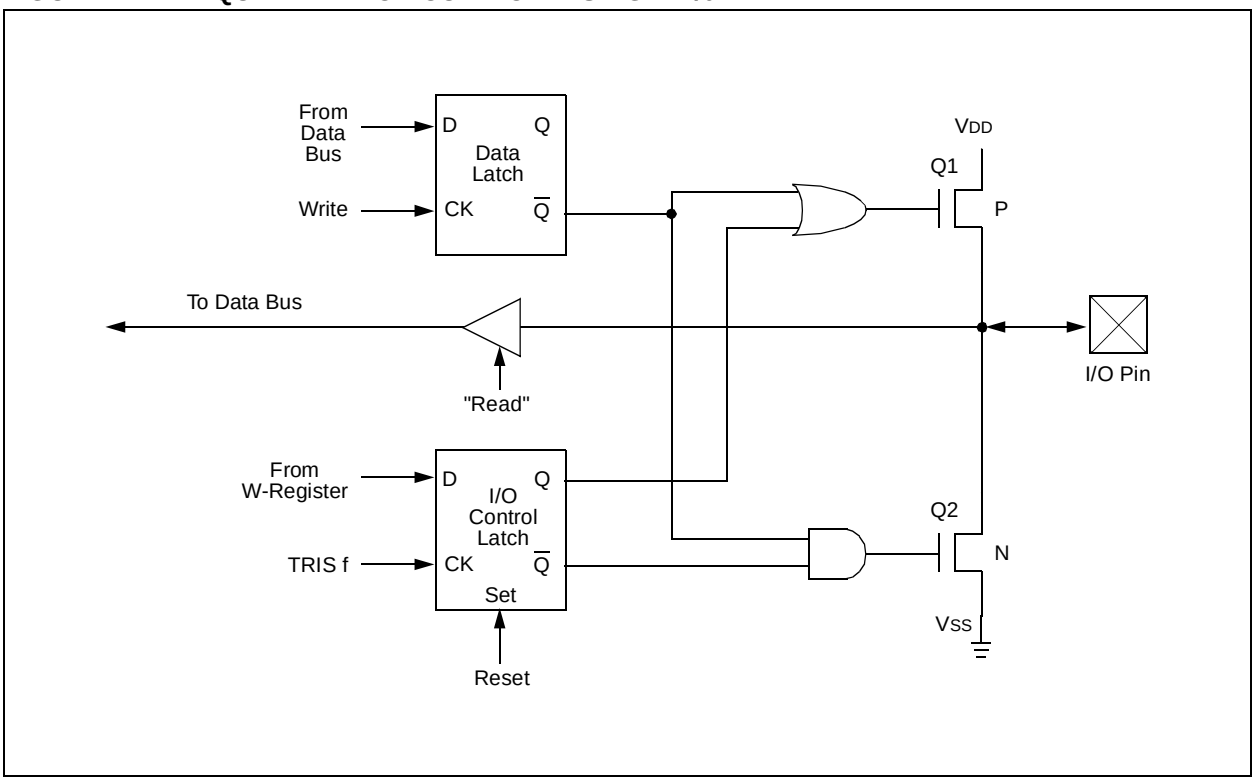
4.3 f7

General purpose register.

4.4 I/O Interfacing

The equivalent circuit for an I/O port bit is shown in Figure 4-1. All ports may be used for both input and output operations. For input operations these ports are non-latching. Any input must be present until read by an input instruction (e.g., MOVF 6, W). The outputs are latched and remain unchanged until the output latch is rewritten. To use a port pin as output, the corresponding direction control bit (in TRISA, TRISB) must be set to zero. For use as an input, the corresponding TRIS bit must be "one". Any I/O pin can be programmed individually as input or output.

FIGURE 4-1: EQUIVALENT CIRCUIT FOR A SINGLE I/0 PIN



4.5 I/O PROGRAMMING CONSIDERATIONS

4.5.1 BIDIRECTIONAL I/O PORTS

a) Some instructions operate internally as read followed by write operations. The BCF and BSF instructions, for example, read the entire port into the CPU, execute the bit operation, and then re-output the result. Caution must be used when these instructions are applied to a port where one or more pins are used as input/ outputs. For example, a BSF operation on bit5 of f6 (Port B) will cause all eight bits of f6 to be read into the CPU. Then the BSF operation takes place on bit5 and f6 is re-output to the output latches. If another bit of f6 is used as a bidirectional I/O pin (say 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 into output mode later on, the content of the data latch may now be unknown.

b) A pin actively outputting a "0" or "1" 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.
For "wired-or" outputs (assuming negative logic), it is recommended to use external pull-up resistors on the corresponding pins. The pin should be left in high-impedance mode, unless a "0" has to be output. Thus, external devices can drive this pin "0" as well. "Wired-and" outputs can be realized in the same way, but with external pull-down resistors and only actively driving the "1" level from the PIC16C54. The resistor values are user selectable, but should not force output currents above the specified limits (Section 15.0).

4.5.2 SUCCESSIVE OPERATIONS ON I/O PORTS

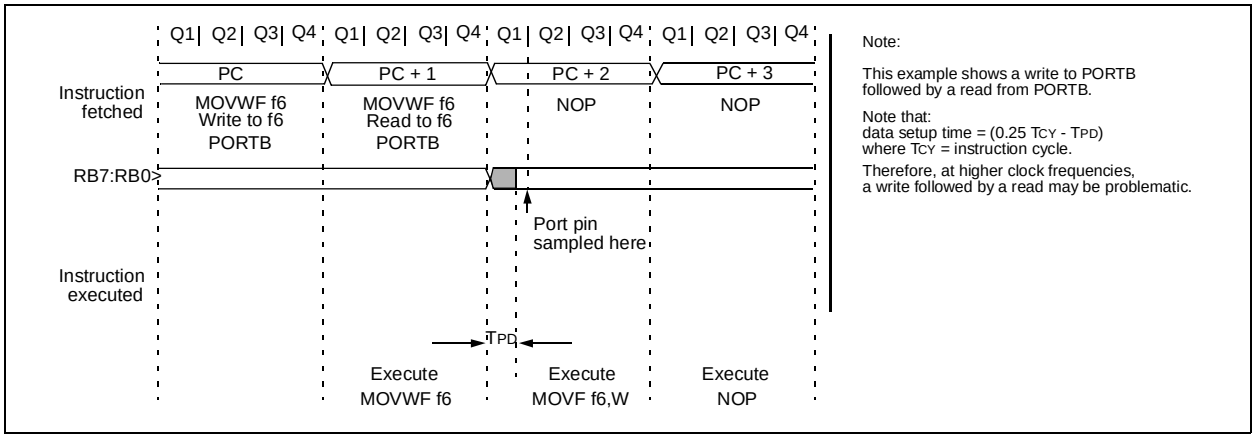
The actual write to an I/O port happens at the end of an instruction cycle, whereas for reading, the data must be valid at the beginning (Figure 4-2) of the instruction cycle. Therefore, care must be exercised if a write followed by a read operation is carried out on the same I/O port. The sequence of 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 an other instruction not accessing this I/O port.

4.5.3 OPERATION IN NOISY ENVIRONMENT

In noisy application environments, such as keyboards which are exposed to ESD (Electro Static Discharge), register contents can get corrupted due to noise spikes.

The on-chip watchdog timer will take care of all situations involving program sequence "lockups." However, if an I/O control register gets corrupted, the program sequence may still be executed properly although an input pin may have switched unintentionally to an output. In this case, the program would always read the same value on this pin. This may result, for example, in a keyboard "lockup" situation without leading to a watchdog timer time-out. Thus, it is recommended to redefine all I/O pins in regular time intervals (inputs as well as outputs). The optimal strategy is to update the I/O control register every time before reading input data or writing output data.

FIGURE 4-2: I/O PORT READ/WRITE TIMING



5.0 GENERAL PURPOSE REGISTERS

f08h - f1Fh: are general purpose register files.

6.0 SPECIAL PURPOSE REGISTERS

6.1 W Working Register

Holds second operand in two operand instructions and/ or supports the internal data transfer.

6.2 TRISA: I/O Control Register For Port A (f5)

Only bits 0-3 are available. The corresponding I/O port (f5) is only 4-bit wide.

6.3 TRISB: I/O Control Register For Port B (f6)

The I/O control registers will be loaded with the content of the W register by execution of the TRIS f instruction. A "1" in the I/O control register puts the corresponding I/O pin into a high impedance mode. A "0" puts the contents of file register f5 or f6, respectively, out on the selected I/O pins.

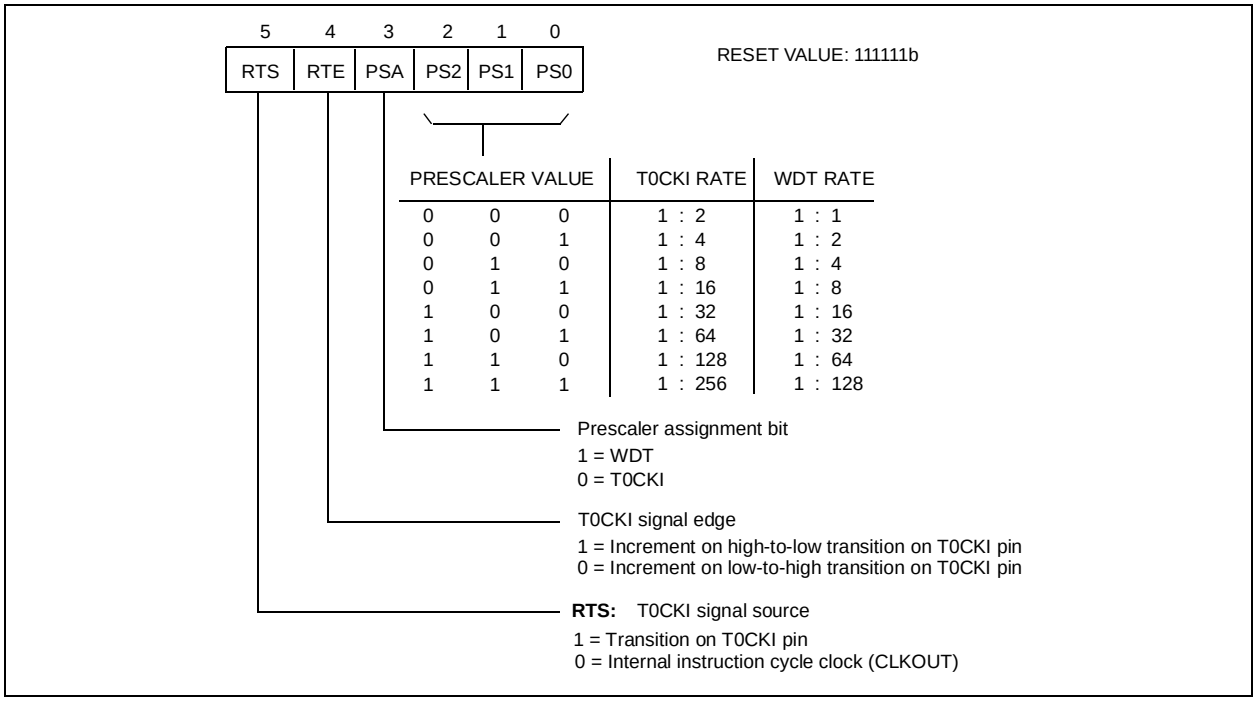
These registers are "write-only" and are set to all "ones" upon a RESET.

6.4 OPTION: Prescaler/T0CKI Option Register

Defines prescaler assignment (T0CKI or WDT), prescaler value, signal source and signal edge for the T0CKI. The OPTION register is "write-only" and is 6-bit wide.

By executing the `OPTION` instruction, the contents of the W register will be transferred to the option register. Upon a RESET, the option register is set to all "ones".

FIGURE 6-1: I/O PORT READ/WRITE TIMING



7.0 RESET CONDITION

A RESET can be caused by applying power to the chip (power-up), pulling the MCLR input "low", or by a Watchdog timer time-out. The device will stay in RESET as long as the oscillator start-up timer (OST) is active or the MCLR input is "low."

The oscillator start-up timer is activated as soon as MCLR input is sensed to be high. This implies that in case of power on reset with MCLR tied to V_{DD} the OST starts from power-up. In case of WDT time-out, it will start at the end of the time-out (since MCLR is high). In case of MCLR reset, the OST will start when MCLR goes high. The nominal OST time-out period is 18 ms. See Section 12.0 for detailed information on OST and power on reset.

During a RESET condition the state of the PIC16C54 is defined as:

- The oscillator is running, or will be started (power-up or wake-up from SLEEP).
- All I/O port pins (RA3:RA0, RB7:RB0) are put into the high-impedance state by setting the "TRIS" registers to all "ones" (= input mode).
- The Program Counter is set to all "ones".
- The OPTION register is set to all "ones".
- The Watchdog Timer and its prescaler are cleared.
- The upper three bits (page select bits) in the Status Register (f3) are cleared to "zero."
- "RC" devices only: The "CLKOUT" signal on the OSC2 pin is held at a "low" level.

8.0 PRESCALER

An 8-bit counter is available as a prescaler for the T0CKI, or as a post-scaler for the watchdog timer, respectively (Figure 8-1). 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 T0CKI and the watchdog timer. Thus, a prescaler assignment for the T0CKI means that there is no prescaler for the watchdog timer, and vice versa.

The PSA and PS2:PS0 bits in the OPTION register determine the prescaler assignment and pre-scale ratio.

When assigned to the T0CKI, all instructions writing to the T0CKI (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.

8.1 Switching Prescaler Assignment

Changing prescaler from T0CKI to WDT

The prescaler assignment is fully under software control (i.e., it can be changed "on the fly" during program execution). To avoid an unintended device RESET, the following instruction sequence must be executed when changing the prescaler assignment from T0CKI to WDT:

```
1. MOVLW B'xx0x0xxx'    ; Select internal clock and select new
2. OPTION               ; prescaler value. If new prescale value
                        ; is = '000' or '001', then select any other
                        ; prescale value temporarily.
3. CLRF 1               ; Clear T0CKI and prescaler.
4. MOVLW B'xxxx1xxx'    ; Select WDT, do not change prescale
                        ; value.
5. OPTION
6. CLRWDT               ; Clears WDT and prescaler.
7. MOVLW B'xxxx1xxx'    ; Select new prescale value.
8. OPTION
```

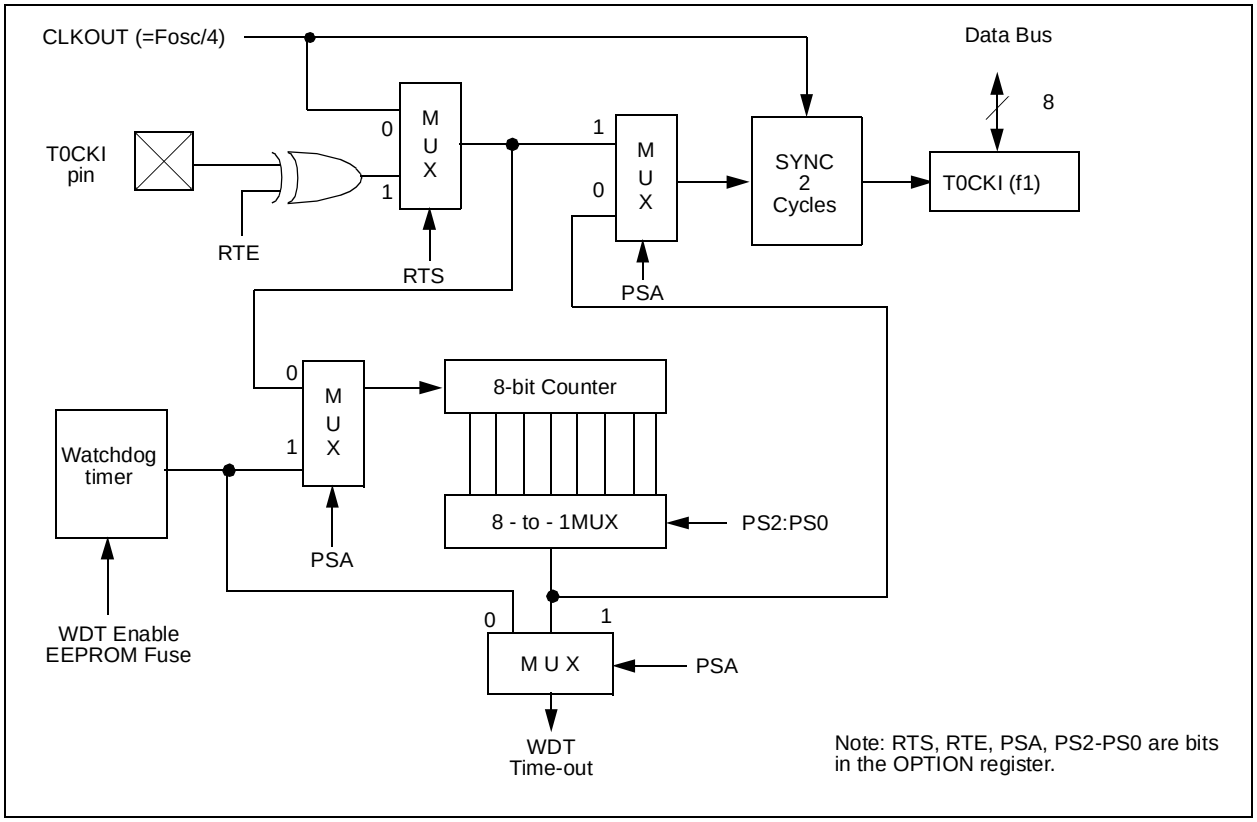
Steps 1 and 2 are only required if an external T0CKI source is used. Steps 7 and 8 are necessary only if the desired prescale value is '000' or '001'.

Changing prescaler from WDT to T0CKI

To change prescaler from WDT to T0CKI use the following sequence:

```
1. CLRWDT               ; Clear WDT and prescaler
2. MOVLW B'xxxx0xxx'    ; Select T0CKI, new prescale value
                        ; and clock source
3. OPTION
```

FIGURE 8-1: BLOCK DIAGRAM T0CKI/WDT PRESCALER



9.0 BASIC INSTRUCTION SET SUMMARY

Each PIC16C54 instruction is a 12-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 PIC16C54 instruction set summary in Table 9-1 lists byte-oriented, bit-oriented, and literal and control operations.

For byte-oriented instructions, "f" represents a file register designator and "d" represents a destination designator. The file register designator specifies which one of the thirty-two PIC16C54 file registers is to be utilized 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 nine bit constant or literal value.

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. 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.

Notes to Table 9-1

Note 1: The 9th bit of the program counter will be forced to a "zero" by any instruction that writes to the PC (f2) except for GOTO (e.g., CALL, MOVWF 2 etc.). See Section 3.3 for details.

Note 2: When an I/O register is modified as a function of itself (e.g., MOVF 6,1), 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 output and is driven low by an external device, the data will be written back with a '0'.

Note 3: The instruction "TRIS f", where f = 5 or 6 causes the contents of the W register to be written to the tristate latches of the specified file (port). A "one" forces the pin to a high impedance state and disables the output buffers.

Note 4: If this instruction is executed on file register f1 (and, where applicable, d=1), the prescaler will be cleared if assigned to the T0CKI.

TABLE 9-1: INSTRUCTION SET SUMMARY

BYTE-ORIENTED FILE REGISTER OPERATIONS

(11-6)	(5)	(4-0)
OPCODE	d	f(FILE #)

d = 0 for destination W
d = 1 for destination f

Instruction-Binary (Hex)				Name	Mnemonic, Operands		Operation	Status Affected	Notes
0001	11df	ffff	1Cf	Add W and f	**ADDWF**	**f, d**	W + f → d	C,DC,Z	1,2,4
0001	01df	ffff	14f	AND W and f	**ANDWF**	**f, d**	W ∧ f → d	Z	2,4
0000	011f	ffff	06f	Clear f	**CLRF**	**f**	0 → f	Z	4
0000	0100	0000	040	Clear W	**CLRW**	**-**	0 → W	Z	
0010	01df	ffff	24f	Complement f	**COMF**	**f, d**	$\bar{f}$ → d	Z	2,4
0000	11df	ffff	0Cf	Decrement f	**DECF**	**f, d**	f -1 → d	Z	2,4
0010	11df	ffff	2Cf	Decrement f, Skip if Zero	**DECFSZ**	**f, d**	f - 1 → d, skip if zero	None	2,4
0010	10df	ffff	28f	Increment f	**INCF**	**f, d**	f + 1 → d	Z	2,4
0011	11df	ffff	3Cf	Increment f,Skip if zero	**INCFSZ**	**f, d**	f + 1 → d, skip if zero	None	2,4
0001	00df	ffff	10f	Inclusive OR W and f	**IORWF**	**f, d**	W ∨ f → d	Z	2,4
0010	00df	ffff	20f	Move f	**MOVF**	**f, d**	f → d	Z	2,4
0000	001f	ffff	02f	Move W to f	**MOVWF**	**f**	W → f	None	1,4
0000	0000	0000	000	No Operation	**NOP**	**-**	-	None	
0011	01df	ffff	34f	Rotate left f	**RLF**	**f, d**	f(n) → d(n+1), C → d(0), f(7) → C	C	2,4
0011	00df	ffff	30f	Rotate right f	**RRF**	**f, d**	f(n) → d (n-1), C → d(7), f(0) → C	C	2,4
0000	10df	ffff	08f	Subtract W from f	**SUBWF**	**f, d**	f - W → d [f + $\overline{W}$ + 1 → d]	C,DC,Z	1,2,4
0011	10df	ffff	38f	Swap halves f	**SWAPF**	**f, d**	f(0-3) ↔ f(4-7) → d	None	2,4
0001	10df	ffff	18f	Exclusive OR W and f	**XORWF**	**f, d**	W ⊕ f → d	Z	2,4

BIT-ORIENTED FILE REGISTER OPERATIONS

(11-8)	(7-5)	(4-0)
OPCODE	b(bit #)	f(FILE #)

Instruction-Binary (Hex)				Name	Mnemonic, Operands		Operation	Status Affected	Notes
0100	bbbf	ffff	4bf	Bit Clear f	**BCF**	**f, b**	0 → f(b)	None	2,4
0101	bbbf	ffff	5bf	Bit Set f	**BSF**	**f, b**	1 → f(b)	None	
0110	bbbf	ffff	6bf	Bit Test f,Skip if Clear	**BTFSC**	**f, b**	Test bit (b) in file (f): Skip if clear	None	
0111	bbbf	ffff	7bf	Bit Test f, Skip if Set	**BTFSS**	**f, b**	Test bit (b) in file (f): Skip if set	None	

LITERAL AND CONTROL OPERATIONS

(11-8)	(7-0)
OPCODE	k(literal)

Instruction-Binary (Hex)				Name	Mnemonic, Operands		Operation	Status Affected	Notes
1110	kkkk	kkkk	Ekk	AND Literal and W	**ANDLW**	**k**	k ∧ W→ W	Z	
1001	kkkk	kkkk	9kk	Call subroutine	**CALL**	**k**	PC + 1 → Stack, k → PC	None	1
0000	0000	0100	004	Clear Watchdog timer	**CLRWDT**	**-**	0 → WDT (and prescaler, if assigned)	$\overline{TO}$, $\overline{PD}$	
101k	kkkk	kkkk	Akk	Go To address (k is 9 bit)	**GOTO**	**k**	k → PC (9 bits)	None	
1101	kkkk	kkkk	Dkk	Incl. OR Literal and W	**IORLW**	**k**	k ∨ W → W	Z	
1100	kkkk	kkkk	Ckk	Move Literal to W	**MOVLW**	**k**	k → W	None	
0000	0000	0010	002	Load OPTION register	**OPTION**	**-**	W → OPTION register	None	
1000	kkkk	kkkk	8kk	Return, place Literal in W	**RETLW**	**k**	k → W, Stack → PC	None	
0000	0000	0011	003	Go into standby mode	**SLEEP**	**-**	0 → WDT, stop oscillator	$\overline{TO}$, $\overline{PD}$	
0000	0000	0fff	00f	Tri-State™ port f	**TRIS**	**f**	W→ I/O control register f	None	3
1111	kkkk	kkkk	Fkk	Excl. OR Literal and W	**XORLW**	**k**	k ⊕ W → W	Z	

Note 1: See previous page.

9.1 Instruction Description

ADDWF **ADD W to f**

Syntax: ADDWF f,d

Encoding:

0001	11df	ffff

Words: 1

Cycles: 1

Operation: (W + f) → d

Status bits: C, DC, Z

Description: Add the contents of the W register to register “f”. If “d” is 0 the result is stored in the W register. If “d” is 1 the result is stored back in register “f”.

ANDLW **AND Literal and W**

Syntax: ANDLW k

Encoding:

1110	kkkk	kkkk

Words: 1

Cycles: 1

Operation: (W .AND. k) → W

Status bits: Z

Description: The contents of W register are AND’ed with the eight bit literal “k”. The result is placed in the W register.

ANDWF **AND W with f**

Syntax: ANDWF f,d

Encoding:

0001	01df	ffff

Words: 1

Cycles: 1

Operation: (W .AND. f) → d

Status bits: Z

Description: AND the W register with register “f”. If “d” is 0 the result is stored in the W register. If “d” is 1 the result is stored back in register “f”.

BCF **Bit Clear f**

Syntax: BCF f,b

Encoding:

0100	bbbf	ffff

Words: 1

Cycles: 1

Operation: 0 → f(b)

Status bits: None

Description: Bit “b” in register “f” is reset to 0.

BSF **Bit Set f**

Syntax: BSF f,b

Encoding:

0101	bbbf	ffff

Words: 1

Cycles: 1

Operation: 1 → f(b)

Status bits: None

Description: Bit “b” in register “f” is set to 1.

BTFSC **Bit Test, skip if Clear**

Syntax: BTFSC f,b

Encoding:

0110	bbbf	ffff

Words: 1

Cycles: 1(2)

Operation: skip if f(b) = 0

Status bits: None

Description: If bit “b” in register “f” is “0” then the next instruction is skipped.

If bit “b” is “0”, the next instruction, fetched during the current instruction execution, is discarded and a NOP is executed instead making this a 2-cycle instruction.

BTFSS **Bit Test, skip if Set**

Syntax: BTFSS f,b

Encoding:

0111	bbbf	ffff

Words: 1

Cycles: 1 (2)

Operation: skip if f(b) = 1

Status bits: None

Description: If bit “b” in register “f” is “1” then the next instruction is skipped.

If bit "b" is "1", the next instruction, fetched during the current instruction execution, is discarded and a NOP is executed instead making this a 2-cycle instruction.

CALL **Subroutine Call**

Syntax: CALL k

Encoding:

1001	kkkk	kkkk

Words: 1

Cycles: 2

Operation: PC + 1 → TOS; k → PC<7:0>, '0' → PC<8>, PA2, PA1, PA0 → PC<11:9>;

Status bits: None

Description: Subroutine call. First, return address (PC + 1) is pushed into the stack. The 8-bit value is loaded into PC bits <7:0>. PC bit9 is cleared. PC <2:0> bits are loaded into PC <11:9>. CALL is a 2-cycle instruction.

CLRF **Clear f and Clear d**

Syntax: CLRF f,d

Encoding:

0000	011f	ffff

Words: 1

Cycles: 1

Operation: 00h → f, 00h → d

Status bits: None

Description: The contents of register "f" are set to 0. If "d" is 0 the contents of both data memory location "f" and W register are set to 0. If "d" is 1 the only contents of register "f" are set to 0.

CLRW **Clear W Register**

Syntax: CLRW

Encoding:

0000	0100	0000

Words: 1

Cycles: 1

Operation: 00h → W

Status bits: Z

Description: W registered is cleared. Zero bit (Z) is set.

CLRWDT **Clear Watchdog Timer**

Syntax: CLRWDT

Encoding:

0000	0000	0100

Words: 1

Cycles: 1

Operation: 00h → WDT, 0 → WDT prescaler,

Status bits: 1 → TO, 1 → PD

Description: CLRWDT instruction resets the Watchdog timer. It also resets the prescaler of the WDT. Status bits TO and PD are set.

COMF **Complement f**

Syntax: COMF f,d

Encoding:

0010	01df	ffff

Words: 1

Cycles: 1

Operation: f → d

Status bits: Z

Description: The contents of register "f" are complemented. If "d" is 0 the result is stored in W. If "d" is 1 the result is stored back in register "f".

DECF **Decrement f**

Syntax: DECF f,d

Encoding:

0010	11df	ffff

Words: 1

Cycles: 1

Operation: (f-1) → d

Status bits: C, DC, Z

Description: Decrement register "f". If "d" is 0 the result is stored in the W register. If "d" is 1 the result is stored back in register "f".

DECFSZ **Decrement f, skip if 0**

Syntax: DECFSZ f,d

Encoding:

0010	11df	ffff

Words: 1

Cycles: 1 (2)

Operation: (f - 1) → d; skip if result = 0

Status bits: None

Description: The contents of register "f" are decremented. If "d" is 0 the result is placed in the W register. If "d" is 1 the result is placed back in register "f". If the result is 0 the next instruction is skipped.

If the result is 0, the next instruction, which is already fetched, is discarded. A NOP is executed instead making it a 2-cycle instruction.

GOTO **Unconditional Branch**

Syntax: GOTOk

Encoding:

101k	kkkk	kkkk

Words: 1

Cycles: 2

Operation: k → PC<8:0>, PA2, PA1, PA0 → PC<11:9>;

Status bits: None

Description: GOTO is an unconditional branch. The eleven bit immediate value is loaded into PC bits <10:0>. The upper bits of PC are loaded from PCLATH <4:3>. GOTO is a 2-cycle instruction.

INCF **Increment f**

Syntax: INCF f,d

Encoding:

0010	10df	ffff

Words: 1

Cycles: 1

Operation: (f + 1) → d

Status bits: C, DC, Z

Description: The contents of register “f” are incremented. If “d” is 0 the result is placed in the W register. If “d” is 1 the result is placed back in register “f”.

INCFSZ **Increment f, skip if 0**

Syntax: INCFSZ f,d

Encoding:

0011	11df	ffff

Words: 1

Cycles: 1 (2)

Operation: (f + 1) → d, skip if result = 0

Status bits: None

Description: The contents of register “f” are incremented. If “d” is 0 the result is placed in the W register. If “d” is 1 the result is placed back in register “f”. If the result is 0 the next instruction is skipped.

If the result is 0, the next instruction, which is already fetched, is discarded. A NOP is executed instead making it a 2-cycle instruction.

IORLW **Inclusive OR Literal with W**

Syntax: IORLW k

Encoding:

1101	kkkk	kkkk

Words: 1

Cycles: 1

Operation: (W .OR. k) → W

Status bits: Z

Description: The contents of the W register are OR’ed with the 8-bit literal “k”. The result is placed in the W register.

IORWF **Inclusive OR W with f**

Syntax: IORWF f,d

Encoding:

0001	00df	ffff

Words: 1

Cycles: 1

Operation: (W .OR. f) → d

Status bits: Z

Description: Inclusive OR 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”.

MOVF **Move f**

Syntax: MOVF f,d

Encoding:

0010	00df	ffff

Words: 1

Cycles: 1

Operation: (f) → d

Status bits: Z

Description: The contents of register "f" are moved. If "d" is 0 the result is placed in the W register. If "d" is 1 the result is placed back in register "f".

MOVLW **Move Literal to W**

Syntax: MOVLW k

Encoding:

1100	kkkk	kkkk

Words: 1

Cycles: 1

Operation: k → W

Status bits: None

Description: The 8-bit literal “k” is loaded into W register.

MOVWF **Move W to f**

Syntax: MOVWF f

Encoding:

0000	001f	ffff

Words: 1

Cycles: 1

Operation: W → f

Status bits: None

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

NOP **No Operation**

Syntax: NOP

Encoding:

0000	0000	0000

Words: 1

Cycles: 1

Operation: No operation

Status bits: None

Description: No operation

OPTION **Load Option Register**

Syntax: OPTION

Encoding:

0000	0000	0010

Words: 1

Cycles: 1

Operation: W → OPTION;

Status bits: None

Description: The contents of the W register is loaded in the OPTION register.

RETLW **Return Literal to W**

Syntax: RETLW k

Encoding:

1000	kkkk	kkkk

Words: 1

Cycles: 2

Operation: k → W; TOS → PC;

Status bits: None

Description: The W register is loaded with the 8-bit literal "k". The program counter is loaded from the top of the stack (the return address). This is a 2-cycle instruction.

RLF **Rotate Left f through Carry**

Syntax: RLF f,d

Encoding:

0011	01df	ffff

Words: 1

Cycles: 1

Operation: f<n> → d<n+1>, f<7> → C, C → d<0>;

Status bits: C

Description: The contents of register "f" are rotated one bit to the left through the Carry Flag. If "d" is 0 the result is placed in the W register. If "d" is 1 the result is stored back in register "f".

RRF **Rotate Right f through Carry**

Syntax: RRF f,d

Encoding:

0011	01df	ffff

Words: 1

Cycles: 1

Operation: f<n> → d<n-1>, f<0> → C, C → d<7>;

Status bits: C

Description: The contents of register "f" are rotated one bit to the right through the Carry Flag. If "d" is 0 the result is placed in the W register. If "d" is 1 the result is placed back in register "f".

SLEEP

Syntax: SLEEP

Encoding:

0000	0000	0011

Words: 1

Cycles: 1

Operation: 0 → PD, 1 → TO; 00h → WDT, 0 → WDT prescaler;

Status bits: TO, PD

Description: The power down status bit (PD) is cleared. Time-out status bit (TO) is set. Watchdog Timer and its prescaler are cleared.

The processor is put into SLEEP mode with the oscillator stopped. See section on SLEEP mode for more details.

SUBWF **Subtract W from f**

Syntax: SUBWF f,d

Encoding:

0000	10df	ffff

Words: 1

Cycles: 1

Operation: (f-W) → d

Status bits: C, DC, Z

```
;SUBWF Example #1
;
clrf    0x20    ;f(20h)=0
movlw   1       ;wreg=1
subwf   0x20    ;f(20h)=f(20h)-wreg=0-1=FFh
                ;Carry=0; Result is negative
;
;SUBWF Example #2
;
movlw   0xFF;
movwf   0x20    ;f(20h)=FFh
clrw            ;wreg=0
subwf   0x20    ;f(20h)=f(20h)-wreg=FFh-0=FFh
                ;Carry=1:Result is positive
;
```

Description: Subtract (2's complement method) the W register from register “f”. If “d” is 0 the result is stored in the W register. If “d” is 1 the result is stored back in register “f”.

SWAPF **Swap f**

Syntax: SWAPF f,d

Encoding:

0011	10df	ffff

Words: 1

Cycles: 1

Operation: f<0:3> → d<4:7>, f<4:7> → d<0:3>;

Status bits: None

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

TRIS **Load TRIS Register**

Syntax: TRIS f

Encoding:

0000	0000	0fff

Words: 1

Cycles: 1

Operation: W → TRIS register f;

Status bits: None

Description: TRIS register f (f = 5, 6 or 7) is loaded with the contents of the W register.

XORLW **Exclusive OR literal with W**

Syntax: XORLWk

Encoding:

1111	kkkk	kkkk

Words: 1

Cycles: 1

Operation: (W .XOR. k) → W

Status bits: Z

Description: The contents of the W register are XOR’ed with the eight bit literal “k”. The result is placed in the W register.

XORWF **Exclusive OR W with f**

Syntax: XORWF f,d

Encoding:

0001	10df	ffff

Words: 1

Cycles: 1

Operation: (W .XOR. f) → d

Status bits: Z

Description: Exclusive OR the contents of the W register with register “f”. If “d” is 0 the result is stored in the W register. If “d” is 1 the result is stored back in register “f”.

10.0 WATCHDOG TIMER (WDT)

The watchdog timer is realized as a free running on-chip RC oscillator which does not require any external components. That means that the WDT will run, even if the clock on the OSC1/OSC2 pins of the device has been stopped, for example, by execution of a SLEEP instruction. A WDT time-out generates a device RESET. The WDT can be permanently disabled by programming a "zero" into a special EPROM fuse which is not part of the normal program memory EPROM. The development tools provide special commands to program this fuse.

10.1 WDT Period

The WDT has a nominal time-out period of 18 ms, (with no prescaler). The time-out periods vary with temperature, VDD and process variations from part to part (DC specifications). 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 2.5 seconds can be realized.

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

The status bit "TO" in file register f3 will be cleared upon a watchdog timer time-out.

The WDT period is a function of the supply voltage, operating temperature, and will also vary from unit to unit due to variations in the manufacturing process. Please refer to the graphs in Section 16.0 and DC specifications for more details.

10.2 WDT Programming Considerations

In a noisy application environment the OPTION register can get corrupted. The OPTION register should be updated at regular intervals.

It should also be taken in account that under worst case conditions (VDD = Minimum, Temperature = Maximum, maximum WDT prescaler) it may take several seconds before a WDT time-out occurs.

11.0 OSCILLATOR CIRCUITS

11.1 Oscillator Types

The MTA81010 series is available with three different oscillator options. On OTP and QTP devices, the oscillator configuration is programmed by the factory and the parts are tested only to the according specifications.

11.2 Crystal Oscillator

The MTA81010, -XT, or LP needs a crystal or ceramic resonator connected to the OSC1 and OSC2 pins to establish oscillation (Figure 11-1). XT = Standard crystal oscillator, LP = low power crystal oscillator. The series resistor RS may be required in XT mode with AT strip-cut type crystals to avoid overdriving.

11.3 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 (Rext) and capacitor (Cext) values, and the operation 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 Cext values. The user also needs to take into account variation due to tolerance of external R and C components used. Figure 11-3 shows how the R/C combination is connected to the MTA81010. For Rext values below 2.2 kΩ, the oscillator operation may become unstable, or stop completely. For very high Rext values (e.g., 1 MΩ), the oscillator becomes sensitive to noise, humidity and leakage. Thus, we recommend to keep Rext between 5 kΩ and 100 kΩ.

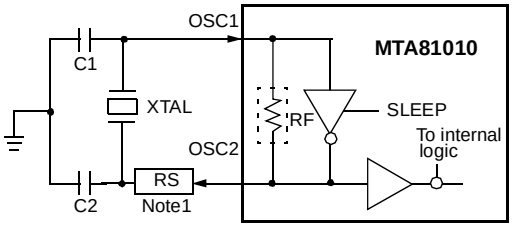
Although the oscillator will operate with no external capacitor (Cext = 0 pF), we recommend using values above 20 pF for noise and stability reasons. With no small external capacitance, the oscillation frequency can vary dramatically due to changes in external capacitances, such as PCB trace capacitance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

See Table 16-1 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 characteristics in Section 16.0 for variation of oscillator frequency due to V_{DD} for given Rext/Cext 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 (Figure 2-2 for timing).

FIGURE 11-1: CRYSTAL OPERATION OR CERAMIC RESONATOR (XT OR LP TYPES ONLY)



Rs may be required in XT mode for AT strip-cut crystals to avoid overdriving. See Table 11-1 and Table 11-2 for recommended values of C1, C2 per oscillator type and frequency.

TABLE 11-1: CAPACITOR SELECTION FOR CERAMIC RESONATORS

Oscillator Type	Resonator Frequency	Capacitor Range C1 = C2
XT	455 kHz	150 - 330 pF
	2.0 MHz	20 - 330 pF
	4.0 MHz	20 - 330 pF

Higher capacitance increases the stability of 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.

FIGURE 11-2: EXTERNAL CLOCK INPUT OPERATION (XT OR LP TYPES ONLY)

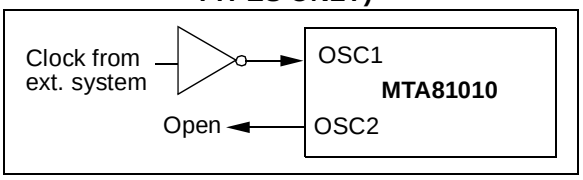
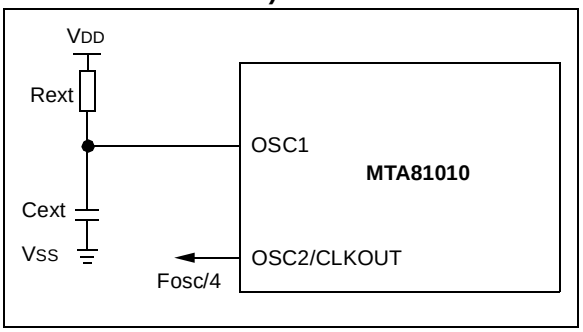


TABLE 11-2: CAPACITOR SELECTION FOR CRYSTAL OSCILLATOR

Osc Type	Freq	C1	C2
LP	32 kHz	15 pF	15 pF
XT	100 kHz	15 - 30 pF	200 -300 pF
	200 kHz	15 - 30 pF	100 -200 pF
	450 kHz	15 - 30 pF	15 - 100 pF
	1 MHz	15 - 30 pF	15 - 30 pF
	2 MHz	15 pF	15 pF
	4 MHz	15 pF	15 pF

Higher capacitance increases the stability of oscillator but also increases the start-up time. These values are for design guidance only. Rs may be required in 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.

FIGURE 11-3: RC OSCILLATOR (RC TYPE ONLY)



12.0 OSCILLATOR START-UP TIMER (OST)

Oscillator circuits based on crystals or ceramic resonators require a certain time after power-up to establish a stable oscillation. An on-chip oscillator start-up timer is provided which keeps the device in a RESET for approximately 18 ms after the voltage on the $\overline{MCLR}$ pin has reached a logic high (V_{IHMC}) level. Thus, external RC networks connected to the $\overline{MCLR}$ input are not required in most cases, allowing for savings in cost-sensitive and/or space restricted applications.

The OST will also be triggered upon a watchdog timer time-out. This is particularly important for applications using the WDT to awake the MTA81010 from SLEEP mode automatically.

The OST is not adequate for low frequency crystals which require much longer than 18 ms to start up and stabilize.

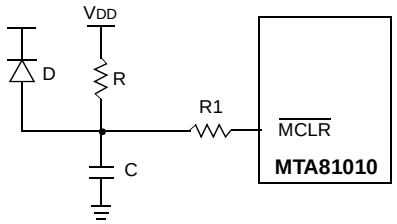
12.1 Power On Reset (POR)

The MTA81010 incorporates an on-chip Power On Reset (POR) circuitry which provides internal chip reset for most power-up situations. To use this feature the user merely needs to tie $\overline{MCLR}$ pin to MCV_{DD}. A simplified block diagram of the on-chip power on reset circuit is shown in Figure 12-4. The power on reset circuit and the oscillator start-up timer circuit are closely related. On power-up the reset latch is set and the start-up timer (Figure 12-4) is reset. The start-up timer begins counting once it detects $\overline{MCLR}$ to be high. After the time-out period, which is typically 18 ms, it will reset the reset-latch and thus end the on-chip reset signal.

Figure 12-5 and Figure 12-6 are two power-up situations with relatively fast rise time on MCV_{DD}. In Figure 12-5, MCV_{DD} is allowed to rise and stabilize before bringing $\overline{MCLR}$ high. The chip will actually come out of reset T_{OST} ms after $\overline{MCLR}$ goes high. In Figure 12-6, the on chip power-on reset feature is being utilized ($\overline{MCLR}$ and MCV_{DD} are tied together). The MCV_{DD} is stable before the start-up timer times out and there is no problem in getting a proper reset. Figure 12-7 depicts a potentially problematic situation where MCV_{DD} rises too slowly. In this situation, when the start-up timer times out, MCV_{DD} has not reached the MCV_{DD} (minimum) value and the chip is therefore not guaranteed to function correctly.

To summarize, the on-chip power on reset is guaranteed to work if the rate of rise of MCV_{DD} is no slower than 0.05 V/ms. It is also necessary that the MCV_{DD} starts from 0V. The on-chip power on reset is also not adequate for low frequency crystals which require much longer than 18 ms to start up and stabilize. For such situations, we recommend that external RC circuits are used for longer power on reset.

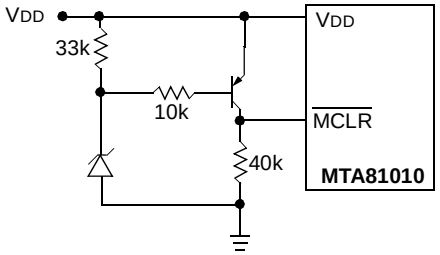
FIGURE 12-1: EXTERNAL POWER ON RESET CIRCUIT



Note:

1. External power-on reset circuit is required only if V_{DD} power-up slope is too slow. The diode D helps discharge the capacitor quickly when V_{DD} powers down.
2. R < 40 kΩ is recommended to make sure that voltage drop across R does not exceed 0.2V (max leakage current spec on $\overline{MCLR}$ pin is 5 μA). A larger voltage drop will degrade V_{IH} level on $\overline{MCLR}$ pin.
3. R1 = 100Ω to 1 kΩ will limit any current flowing into $\overline{MCLR}$ from external capacitor C in the event of $\overline{MCLR}$ pin breakdown due to ESD or EOS.

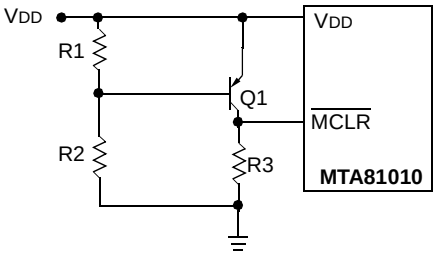
FIGURE 12-2: BROWN OUT PROTECTION CIRCUIT



Note:

1. This circuit will activate reset when V_{DD} goes below (Vz + 0.7V) where Vz = Zener voltage.

FIGURE 12-3: BROWN OUT PROTECTION CIRCUIT



Note:

1. This brown circuit is less expensive, albeit less accurate. Transistor Q1 turns off when V_{DD} is below a certain level such that:

$$V_{DD} \cdot \frac{R1}{R1 + R2} = 0.7V.$$

FIGURE 12-4: SIMPLIFIED POWER ON RESET BLOCK DIAGRAM

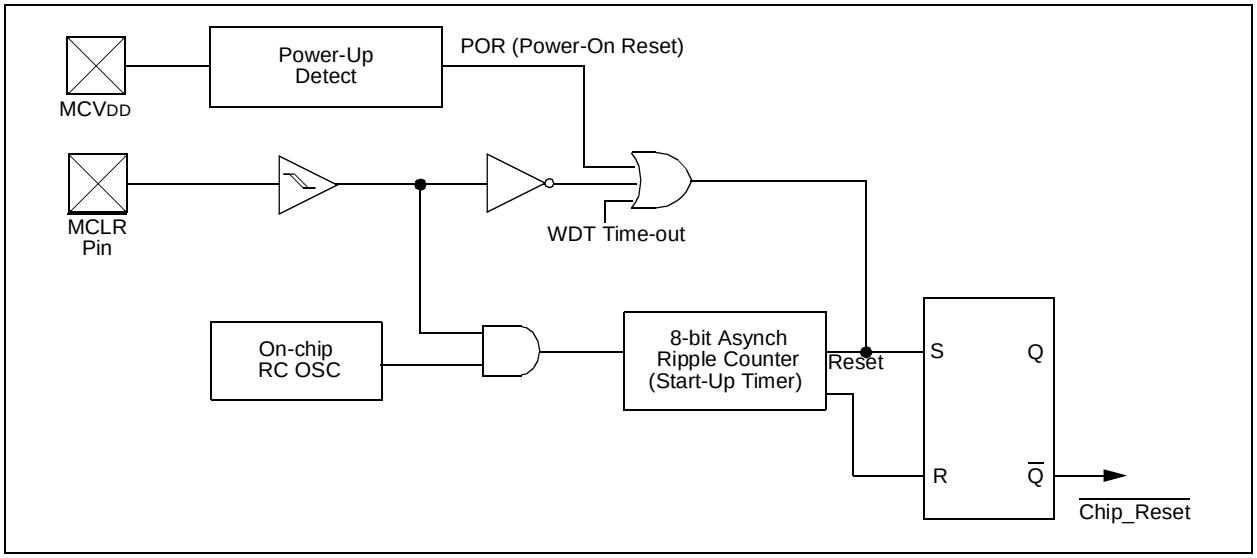
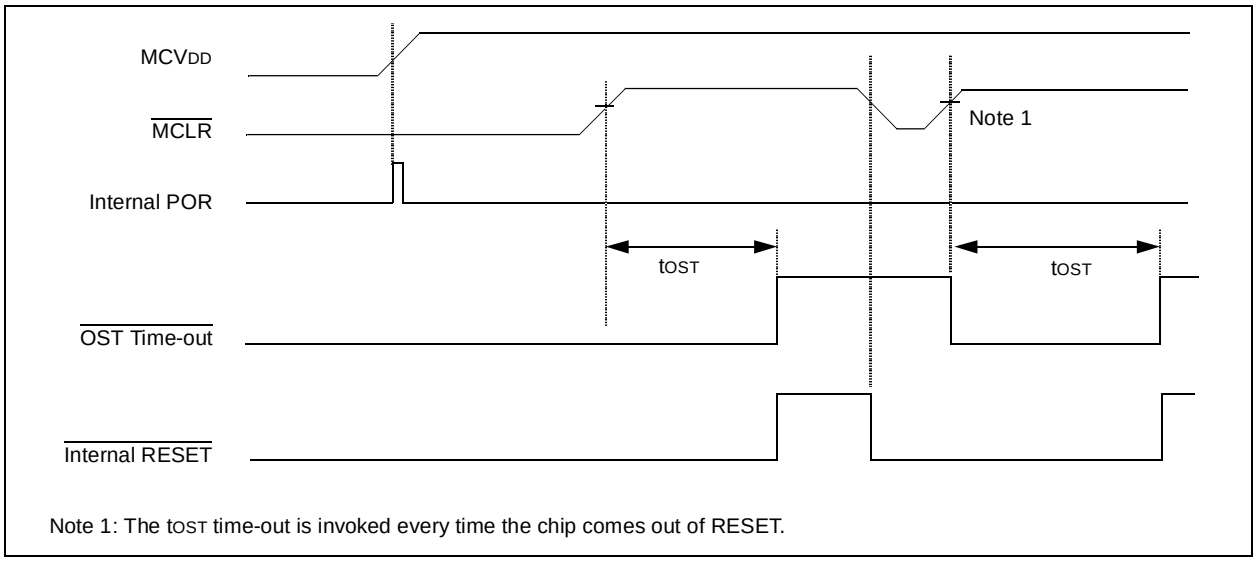


FIGURE 12-5: USING EXTERNAL RESET INPUT



Note 1: The tOST time-out is invoked every time the chip comes out of RESET.

FIGURE 12-6: USING ON-CHIP POR (FAST VDD RISE TIME)

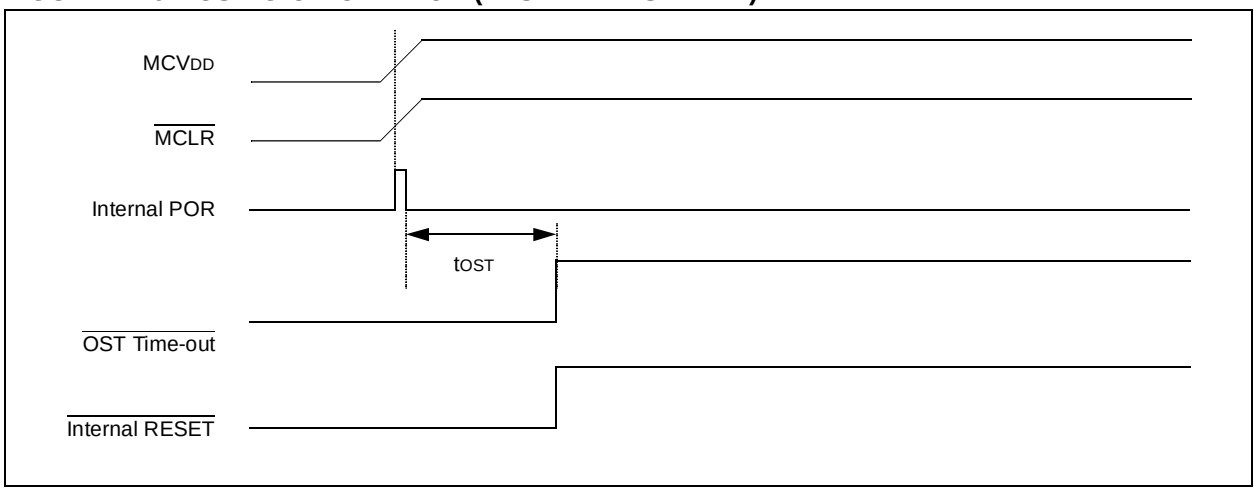
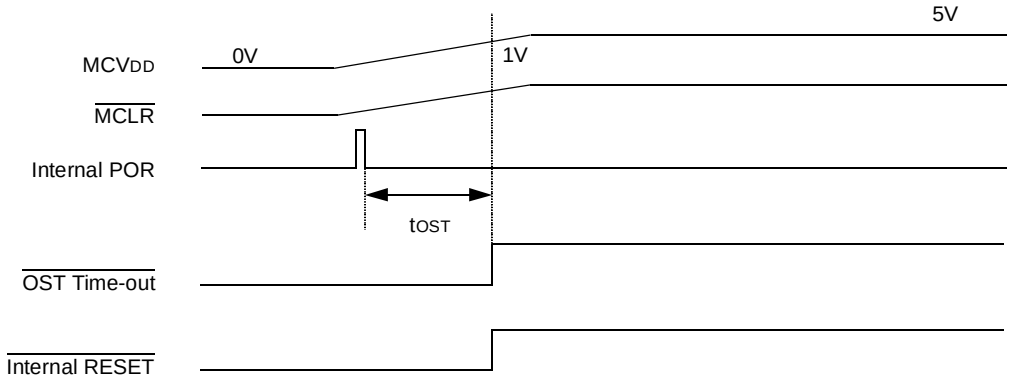


FIGURE 12-7: USING ON-CHIP POR (SLOW V_{DD} RISE TIME)



When V_{DD} rises slowly, the internal time-out period expires long before V_{DD} has reached its final value.
In this example, the chip will reset properly if, and only if, $1V \geq V_{DD}$ min.

13.0 POWER DOWN MODE (SLEEP)

The power down mode is entered by executing a SLEEP instruction.

If enabled, the watchdog timer will be cleared but keeps running, the bit PD in the status register (f3) is cleared, the TO bit is set, and the oscillator driver is turned off. The I/O ports maintain the status they had, before the SLEEP command was executed (driving high, low, or hi-impedance).

For lowest current consumption in this mode, all I/O pins should be either at V_{DD}, or V_{SS}, with no external circuitry drawing current from the I/O pin. I/O pins that are in the High-Z mode should be pulled high or low externally to avoid switching currents caused by floating inputs. The T0CKI input should also be at V_{DD} or V_{SS} for lowest current consumption.

The $\overline{MCLR}$ pin must be at V_{IHMC}.

13.1 Wake-Up

The device can be awakened by a watchdog timer time-out (if it is enabled) or an externally applied "low" pulse at the $\overline{MCLR}$ pin. In both cases the PIC16C54 will stay in RESET mode for one oscillator start-up timer period (triggered from rising edge on $\overline{MCLR}$ or WDT time-out) before normal program execution resumes.

The PD bit in the STATUS register, which is set to “1” during power on, but cleared by the SLEEP command, can be used to determine if the processor was powered up or if it was awakened from the power down mode (Figure 3-1). The TO bit in the Status register can be used to determine if the "wake up" was caused by an external $\overline{MCLR}$ signal or a watchdog timer time out.

Note: Some applications may require external R/C networks on the $\overline{MCLR}$ pin in order to allow for oscillator start-up times longer than one OST period. In this case, a WDT wake up from power down mode is not recommended, because a RESET generated by a WDT time out does not discharge the external capacitor, and the PIC16C54 will be in RESET only for the oscillator start-up timer period.

14.0 CONFIGURATION FUSES

The configuration EPROM consists of four EPROM fuses which are not part of the normal EPROM for program storage.

Two are for the selection of the oscillator type, one is the watchdog timer enable fuse, and one is the code protection fuse.

OTP or QTP devices have the oscillator configuration programmed by the factory and the parts are tested accordingly. The packages are marked with the suffixes XT, RC, HS or LP following the part number to identify the oscillator type and operating range.

14.1 Customer ID Code

The MTA81010 series has 16 special EPROM bits which are not part of the normal program memory. These bits are available to the user to store an identifier (ID) code, checksum, or other informative data. They cannot be accessed during normal program execution. Programmers designed for the MTA81010 provide special commands to read or write these ID bits.

14.2 Code Protection

The program code written into the EPROM can be protected by programming the code protection fuse with "0".

When code protected, the contents of the program EPROM cannot be read out in a way that the program code can be reconstructed. In addition, all memory locations starting at 040h and above are protected against programming.

It is still possible to program locations 000h - 03Fh, the ID locations and the configuration fuses.

Note that the configuration fuses and the ID bits can still be read, even if the code protection logic is active.

14.2.1 VERIFYING A CODE-PROTECTED PIC16C54

When code protected verifying any program memory location will read a scrambled output which looks like "00000000XXXX" (binary) where X is 1 or 0. To verify a device after code protection, follow this procedure:

a) First, program and verify a good device without code protecting it.
b) Next, blow its code protection fuse and then load its contents in a file.
c) Verify any code-protected PIC16C54 against this file.

15.0 ELECTRICAL CHARACTERISTICS

15.1 Absolute Maximum Ratings*

Ambient temperature under bias ..... -55°C to +125°C
Storage Temperature ..... -65°C to +150°C
Voltage on V_{DD} with respect to V_{SS} ..... 0 to +7.5V
Voltage on $\overline{MCLR}$ with respect to V_{SS} ..... 0 to +14V
Voltage on all other pins with respect to V_{SS} ..... -0.6V to (V_{DD} + 0.6V)
Total power Dissipation (Note 1) ..... 800 mW
Maximum Current out of V_{SS} pin ..... 150 mA
Maximum Current into V_{DD} pin ..... 50 mA
Maximum Current into an input pin (T0CKI only) ..... ±500 µA
Input clamp current, I_{IK} ($V_I < 0$ or $V_I > V_{DD}$) ..... ±20 mA
Output clamp current, I_{OK} ($V_0 < 0$ or $V_0 > V_{DD}$) ..... ±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 Output Current sourced by a single I/O port (PortA, B or C) ..... 40 mA
Maximum Output Current sunk by a single I/O port (PortA, B or C) ..... 50 mA

Note 1: Total power dissipation should not exceed 800 mW for the package. 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 "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	PIC16C5X-04
RC	V_{DD}: 3.0V to 6.25V I_{DD}: 2.4 mA Max. at 5.5V I_{PD}: 4 µA Max. at 3.0V WDT dis Freq: 4 MHz Max.
XT	V_{DD}: 3.0V to 6.25V I_{DD}: 2.4 mA Max. at 5.5V I_{PD}: 5 µA Max. at 3.0V WDT dis Freq: 4 MHz Max.
HS	V_{DD}: 4.5V to 5.5V I_{DD}: 1.8 mA typ. at 5.5V I_{PD}: 0.3 µA typ. at 3.0V WDT dis Freq: 4 MHz Max.
LP	V_{DD}: 4.5V to 5.5V I_{DD}: 17 µA typ. at 32 kHz, 3.0V I_{PD}: 0.3 µA typ. at 3.0V WDT dis Freq: 200 kHz typ.

Note: 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 guarantees the specifications required.

TABLE 15-2: DC CHARACTERISTICS: PIC16C5X-RC, XT, HS, LP (COMMERCIAL)

DC CHARACTERISTICS **Power Supply Pins**		**Standard Operating Conditions** Operating temperature: 0°C ≤ TA ≤ +70°C, unless otherwise stated Operating voltage: VDD = 3.0V to 5.5V, unless otherwise stated				
Characteristic	**Sym**	**Min**	**Typ** (Note 1)	**Max**	**Units**	**Conditions**
Supply Voltage						
PIC16C5X-XT	VDD	3.0		6.25	V	FOSC = DC to 4 MHz
PIC16C5X-RC		3.0		6.25	V	FOSC = DC to 4 MHz
PIC16C5X-HS		4.5		5.5	V	FOSC = DC to 20 MHz
PIC16C5X-LP		2.5		6.25	V	FOSC = DC to 40 MHz
RAM Data Retention Voltage (Note 3)	VDR		1.5		V	Device in SLEEP mode
VDD start voltage to guarantee Power-On Reset	VPOR		VSS		V	See Section 13.1 for details on Power-On Reset
VDD rise rate to guarantee Power-On Reset	SVDD	0.05*			V/ms	See Section 13.1 for details on Power-On Reset
Supply Current (Note 2)						
PIC16C5X-XT	IDD		1.8	3.3	mA	FOSC = 4 MHz, VDD = 5.5V
PIC16C5X-RC (Note 5)			1.8	3.3	mA	FOSC = 4 MHz, VDD = 5.5V
PIC16C5X-HS			4.8	10	mA	FOSC = 10 MHz, VDD = 5.5V
			9.0	20	mA	FOSC = 20 MHz, VDD = 5.5V
PIC16C5X-LP			15	32	µA	FOSC = 32 kHz, VDD = 3.0V, WDT disabled
Power Down Current (Note 4)						
PIC16C5X	IPD		4	12	µA	VDD = 3.0V, WDT enabled
			0.6	9	µA	VDD = 3.0V, WDT disabled

* These parameters are based on characterization and are not tested.

Note 1: Data in the column labeled "Typ" is based on characterization results at 25°C. This data is for design guidance only and is not tested for, or guaranteed by Microchip Technology.

Note 2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as bus loading, oscillator type, bus rate, internal code execution pattern, and temperature also have an impact on the current consumption.

a) The test conditions for all IDD measurements in active operation mode are: $\overline{\text{OSC1}}$= external square wave, from rail to rail; all I/O pins tristated, pulled to VDD, T0CKI = VDD, $\overline{\text{MCLR}}$ = VDD; WDT enabled/disabled as specified.

b) For standby current measurements, the conditions are the same, except that the device is in SLEEP mode.

Note 3: This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

Note 4: 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.

Note 5: Does not include current through Rext. The current through the resistor can be estimated by the formula IR = VDD/2Rext (mA) with Rext in kΩ.

TABLE 15-3: DC CHARACTERISTICS: PIC16C5X-RC, XT, HS, LP (INDUSTRIAL)

DC CHARACTERISTICS **Power Supply Pins**		**Standard Operating Conditions** Operating temperature: -40°C ≤ TA ≤ +85°C, unless otherwise stated Operating voltage: VDD = 3.5V to 5.5V, unless otherwise stated				
Characteristic	**Sym**	**Min**	**Typ** (Note 1)	**Max**	**Units**	**Conditions**
Supply Voltage						
PIC16C5X-XT	VDD	3.0		6.25	V	FOSC = DC to 4 MHz
PIC16C5X-RC		3.0		6.25	V	FOSC = DC to 4 MHz
PIC16C5X-HS		4.5		5.5	V	FOSC = DC to 20 MHz
PIC16C5X-LP		2.5		6.25	V	FOSC = DC to 40 MHz
RAM Data Retention Voltage (Note 3)	VDR		1.5		V	Device in SLEEP mode
VDD start voltage to guarantee Power-On Reset	VPOR		VSS		V	See Section 13.1 for details on Power-On Reset
VDD rise rate to guarantee Power-On Reset	SVDD	0.05*			V/ms	See Section 13.1 for details on Power-On Reset
Supply Current (Note 2)						
PIC16C5X-XT	IDD		1.8	3.3	mA	FOSC = 4 MHz, VDD = 5.5V
PIC16C5X-RC (Note 5)			1.8	3.3	mA	FOSC = 4 MHz, VDD = 5.5V
PIC16C5X-HS			4.8	10	mA	FOSC = 10 MHz, VDD = 5.5V
			9.0	20	mA	FOSC = 20 MHz, VDD = 5.5V
PIC16C5X-LP			19	40	µA	FOSC = 32 kHz, VDD = 3.0V, WDT disabled
Power Down Current (Note 4)						
PIC16C5X	IPD		5	14	µA	VDD = 3.0V, WDT enabled
			0.6	12	µA	VDD = 3.0V, WDT disabled

* These parameters are characterized but not tested.

Note 1: Data in the column labeled "Typ" is based on characterization results at 25°C. This data is for design guidance only and is not tested for, or guaranteed by Microchip Technology.

Note 2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as bus loading, oscillator type, bus rate, internal code execution pattern, and temperature also have an impact on the current consumption.

a) 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, T0CKI = VDD, $\overline{MCLR}$ = VDD; WDT enabled/disabled as specified.

b) For standby current measurements, the conditions are the same, except that the device is in SLEEP mode.

Note 3: This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

Note 4: 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.

Note 5: Does not include current through Rext. The current through the resistor can be estimated by the formula IR = VDD/2Rext (mA) with Rext in kΩ.

TABLE 15-4: DC CHARACTERISTICS: PIC16C5X-RC, XT, HS, LP (AUTOMOTIVE)

DC Characteristics Power Supply Pins		Standard Operating Conditions Operating Temperature -40°C ≤ TA ≤ +125°C, unless otherwise stated. Operating Voltage VDD = 3.5V to 5.5V, unless otherwise stated.				
Characteristic	**Sym**	**Min**	**Typ (Note 1)**	**Max**	**Units**	**Conditions**
Supply Voltage						
PIC16C5X-XT	VDD	3.25		6.0	V	FOSC = DC to 4 MHz
PIC16C5X-RC		3.25		6.0	V	FOSC = DC to 4 MHz
PIC16C5X-HS		4.5		5.5	V	FOSC = DC to 20 MHz
PIC16C5X-LP		2.5		6.0	V	FOSC = DC to 40 kHz
RAM Data Retention Voltage (Note 3)	VDR		1.5		V	Device in SLEEP mode
VDD Start Voltage to Guarantee Power-On Reset	VPOR		VSS		V	See Section 13.1 for details on Power-On Reset
VDD rise rate to guarantee Power-On Reset	SVDD	0.05*			V/ms	See Section 13.1 for details on Power-On Reset
Supply Current (Note 2)						
PIC16C5X-XT	IDD		1.8	3.3	mA	FOSC = 4 MHz, VDD = 5.5V
PIC16C5X-RC (Note 5)			1.8	3.3	mA	FOSC = 4 MHz, VDD = 5.5V
PIC16C5X-HS			4.8	10	mA	FOSC = 10 MHz, VDD = 5.5V
			9.0	20	mA	FOSC = 16 MHz, VDD = 5.5V
PIC16C5X-LP			25	55	µA	FOSC = 32 kHz, VDD = 3.25V, WDT disabled
Power Down Current (Note 4)						
PIC16C5X	IPD		5	22	µA	VDD = 3.25V, WDT enabled
			0.8	18	µA	VDD = 3.25V, WDT disabled

* These parameters are based on characterization and are not tested.

Note 1: Data in the column labeled “Typical” is based on characterization results at 25°C. This data is for design guidance only and is not tested for, or guaranteed by Microchip Technology.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors such as bus loading, oscillator type, bus rate, internal code execution pattern, and temperature also have an impact on the current consumption.

a) 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, T0CKI = VDD, $\overline{\text{MCLR}}$ = VDD; WDT enabled/disabled as specified.
b) For stand-by current measurements, the conditions are the same, except that the device is in SLEEP mode.

3: This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.

4: 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.

5: Does not include current through Rext. The current through the resistor can be estimated by the formula: IR = VDD/2Rext (mA) with Rext in kΩ.

TABLE 15-5: DC CHARACTERISTICS: PIC16C5X-RC, XT, HS, LP (COMMERCIAL) PIC16C5XI-RC, XT, HS, LP (INDUSTRIAL)

DC Characteristics All Pins Except Power Supply Pins	Standard Operating Conditions Operating Temperature -40°C ≤ TA ≤ +85°C (for industrial) 0°C ≤ TA ≤ +70°C (for commercial) Operating Voltage VDD range is described in Section 15-2 and Section 15-3.					
Characteristic	**Sym**	**Min**	**Typ (Note 1)**	**Max**	**Units**	**Conditions**
Input Low Voltage						
I/O ports	VIL	VSS		0.2 VDD	V	Pin at hi-impedance
$\overline{\text{MCLR}}$ (Schmitt Trigger)		VSS		0.15 VDD	V	
T0CKI (Schmitt Trigger)		VSS		0.15 VDD	V	
OSC1 (Schmitt Trigger)		VSS		0.15 VDD	V	PIC16C5X-RC only (Note 5)
OSC1		VSS		0.3 VDD	V	PIC16C5X-XT, HS, LP
Input High Voltage						
I/O ports	VIH	0.45 VDD		VDD	V	For all VDD (Note 6)
		2.0		VDD	V	4.0V < VDD ≤ 5.5V (Note 6)
		0.36 VDD		VDD	V	VDD > 5.5V
$\overline{\text{MCLR}}$ (Schmitt Trigger)		0.85 VDD		VDD	V	
T0CKI (Schmitt Trigger)		0.85 VDD		VDD	V	
OSC1 (Schmitt Trigger)		0.85 VDD		VDD	V	PIC16C5X-RC only (Note 5)
OSC1		0.7 VDD		VDD	V	PIC16C5X-XT, HS, LP
Input Leakage Current (Notes 3,4)						**For VDD ≤ 5.5V**
I/O ports	IIL	-1	0.5	+1	µA	VSS ≤ VPIN ≤ VDD, Pin at hi-impedance
$\overline{\text{MCLR}}$		-5			µA	VPIN = VSS + 0.25V
$\overline{\text{MCLR}}$			0.5	+5	µA	VPIN = VDD
T0CKI		-3	0.5	+3	µA	VSS ≤ VPIN ≤ VDD
OSC1		-3	0.5	+3	µA	VSS ≤ VPIN ≤ VDD PIC16C5X-XT, HS, LP
Output Low Voltage						
I/O ports	VOL			0.6	V	IOL = 8.7 mA, VDD = 4.5V
OSC2/CLKOUT (PIC16C5X-RC)				0.6	V	IOL = 1.6 mA, VDD = 4.5V
Output High Voltage						
I/O ports (Note 4)	VOH	VDD-0.7			V	IOH = -5.4 mA, VDD = 4.5V
OSC2/CLKOUT (PIC16C5X-RC)		VDD-0.7			V	IOH = -1.0 mA, VDD = 4.5V

Note 1: Data in the column labeled “Typical” is based on characterization results at 25°C. This data is for design guidance only and is not tested for, or guaranteed by Microchip Technology.

2: Total power dissipation as stated under absolute maximum ratings must not be exceeded.

3: The leakage current on the $\overline{\text{MCLR}}$/VPP 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 voltage.

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

5: For PIC16C5X-RC devices, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C5X be driven with external clock in RC mode.

6: The user may use the better of the two specifications.

TABLE 15-6: DC CHARACTERISTICS: PIC16C5X-RC, XT, HS, LP (AUTOMOTIVE)

DC Characteristics All Pins Except Power Supply Pins		Standard Operating Conditions Operating Temperature -40°C ≤ TA ≤ +125°C Operating Voltage VDD range is described in Section 15-2 and Section 15-3.				
Characteristic	**Sym**	**Min**	**Typ (Note 1)**	**Max**	**Units**	**Conditions**
Input Low Voltage						
I/O ports	VIL	VSS		0.15 VDD	V	Pin at hi-impedance
MCLR (Schmitt Trigger)		VSS		0.15 VDD	V	
T0CKI (Schmitt Trigger)		VSS		0.15 VDD	V	
OSC1 (Schmitt Trigger)		VSS		0.15 VDD	V	PIC16C5X-RC only (Note 5)
OSC1		VSS		0.3 VDD	V	PIC16C5X-XT, HS, LP
Input High Voltage						
I/O ports	VIH	0.45 VDD		VDD	V	For all VDD (Note 6)
		2.0		VDD	V	4.0V < VDD ≤ 5.5V (Note 6)
		0.36 VDD		VDD	V	VDD > 5.5V
MCLR (Schmitt Trigger)		0.85 VDD		VDD	V	
T0CKI (Schmitt Trigger)		0.85 VDD		VDD	V	
OSC1 (Schmitt Trigger)		0.85 VDD		VDD	V	PIC16C5X-RC only (Note 5)
OSC1		0.7 VDD		VDD	V	PIC16C5X-XT, HS, LP
Input Leakage Current (Notes 3,4)						**For VDD ≤ 5.5V**
I/O ports	IIL	-1	0.5	+1	µA	VSS ≤ VPIN ≤ VDD, Pin at hi-impedance
MCLR		-5			µA	VPIN = VSS + 0.25V
MCLR			0.5	+5	µA	VPIN = VDD
T0CKI		-3	0.5	+3	µA	VSS ≤ VPIN ≤ VDD
OSC1		-3	0.5	+3	µA	VSS ≤ VPIN ≤ VDD PIC16C5X-XT, HS, LP
Output Low Voltage						
I/O ports	VOL			0.6	V	IOL = 8.7 mA, VDD = 4.5V
OSC2/CLKOUT (PIC16C5X-RC)				0.6	V	IOL = 1.6 mA, VDD = 4.5V
Output High Voltage						
I/O ports (Note 4)	VOH	VDD-0.7			V	IOH = -5.4 mA, VDD = 4.5V
OSC2/CLKOUT (PIC16C5X-RC)		VDD-0.7			V	IOH = -1.0 mA, VDD = 4.5V

Note 1: Data in the column labeled Typical is based on characterization results at 25°C. This data is for design guidance only and is not tested for, or guaranteed by Microchip Technology.

2: Total power dissipation as stated under absolute maximum ratings must not be exceeded.

3: The leakage current on the MCLR/VPP 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 voltage.

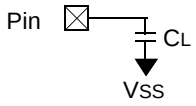
4: Negative current is defined as coming out of the pin.

5: For PIC16C5X-RC devices, the OSC1/CLKIN pin is a Schmitt Trigger input. It is not recommended that the PIC16C5X be driven with external clock in RC mode.

6: The user may use the better of the two specifications.

15.2 Timing Diagrams and Specifications

FIGURE 15-1: LOAD CONDITIONS



C_L = 50 pF for all pins except OSC2
15 pF for OSC2 in XT, HS or LP modes when external clock is used to drive OSC1

TABLE 15-7: EXTERNAL CLOCK TIMING

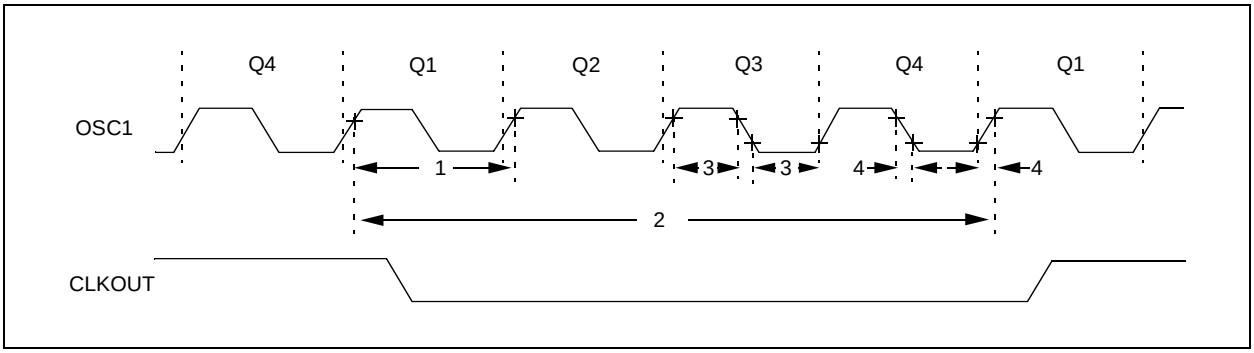


TABLE 15-8: EXTERNAL CLOCK TIMING REQUIREMENTS

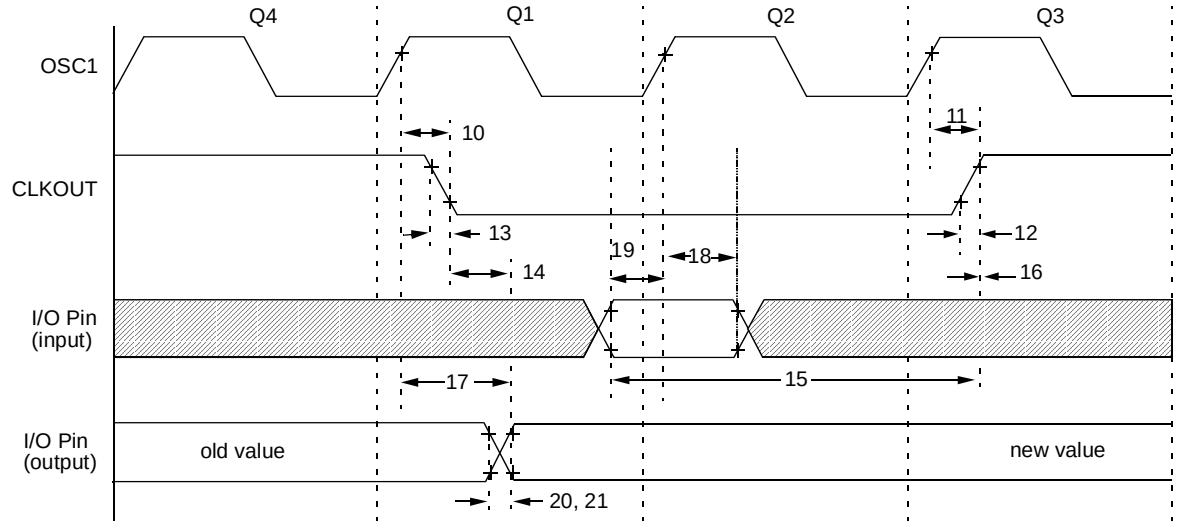
Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
	FOSC	External CLKIN Frequency (Note 1)	DC	—	4	MHz	RC osc mode
			DC	—	4	MHz	XT osc mode
			DC	—	20	MHz	HS osc mode (Comm/Indust)
			DC	—	16	MHz	HS osc mode (Automotive)
			DC	—	40	kHz	LP osc mode
		Oscillator Frequency (Note 1)	DC	—	4	MHz	RC osc mode
			0.1	—	4	MHz	XT osc mode
			4	—	20	MHz	HS osc mode (Comm/Indust)
			4	—	16	MHz	HS osc mode (Automotive)
			5	—	40	kHz	LP osc mode
1	TOSC	External CLKIN Period (Note 1)	250	—	—	ns	RC osc mode
			250	—	—	ns	XT osc mode
			50	—	—	ns	HS osc mode
			100	—	—	µs	LP osc mode
		Oscillator Period (Note 1)	250	—	—	ns	RC osc mode
			250	—	10,000	ns	XT osc mode
			62.5	—	250	ns	HS osc mode (Comm/Indust)
			50	—	250	ns	HS osc mode (Automotive)
			100	—	200	µs	LP osc mode
2	TCY	Instruction Cycle Time (Note 1)	1.0	—	DC	µs	
3	TosL, TosH	Clock in (OSC1) Low or High Time	50	—	—	ns	XT oscillator
			2.5	—	—	µs	LP oscillator
			10	—	—	ns	HS oscillator
4	TosR, TosF	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.

FIGURE 15-2: CLKOUT AND I/O TIMING



Note: All tests must be done with specified capacitive loads (see datasheet) 50 pF on I/O pins and CLKOUT.

TABLE 15-9: 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	—	—	0.5 TCY+20	ns	Note 1
15	TioV2ckH	Port in valid before CLKOUT↑	0.25 TCY+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	Note 2
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	—	10	25	ns	Note 2
21	TioF	Port output fall time	—	10	25	ns	Note 2

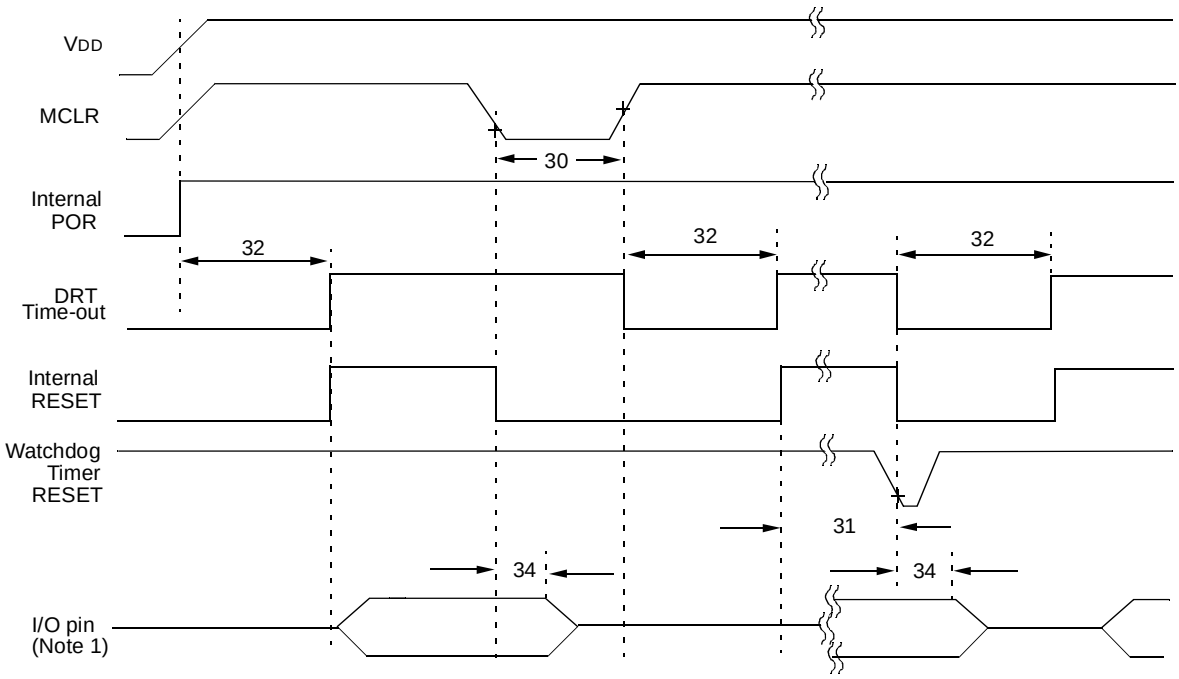
* These parameters are characterized but not tested.

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

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

2: See Figure 15-1 for loading conditions.

FIGURE 15-3: RESET, WATCHDOG TIMER, AND DEVICE RESET TIMER TIMING



Note 1: I/O pins must be taken out of hi-impedance mode by enabling the output drivers in software.

TABLE 15-10: RESET, WATCHDOG TIMER, AND DEVICE RESET TIMER

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30	TmcL	$\overline{\text{MCLR}}$ Pulse Width (low)	100	—	—	ns	VDD = 5V, -40°C to +125°C
31	Twdt	Watchdog Timer Time-out Period (No Prescaler)	9*	18	30*	ms	VDD = 5V, -40°C to +125°C
32	TDRT	Device Reset Timer Period	9*	18*	30*	ms	VDD = 5V, -40°C to +125°C
34	Tioz	I/O Hi-impedance from $\overline{\text{MCLR}}$ 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.

FIGURE 15-4: TIMER0 CLOCK TIMINGS

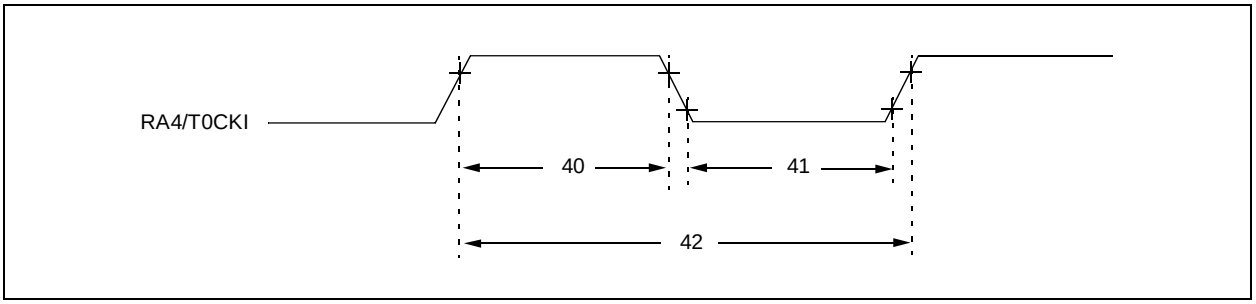


TABLE 15-11: TIMER0 CLOCK REQUIREMENTS

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		$\frac{T_{CY} + 40}{N}$*	—	—	ns	N = prescale value (1, 2, 4, ..., 256)

* 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.

16.0 DC AND AC CHARACTERISTICS

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 VDD range). This is for information only and devices are guaranteed to 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 16-1: TYPICAL RC OSCILLATOR FREQUENCY vs. TEMPERATURE

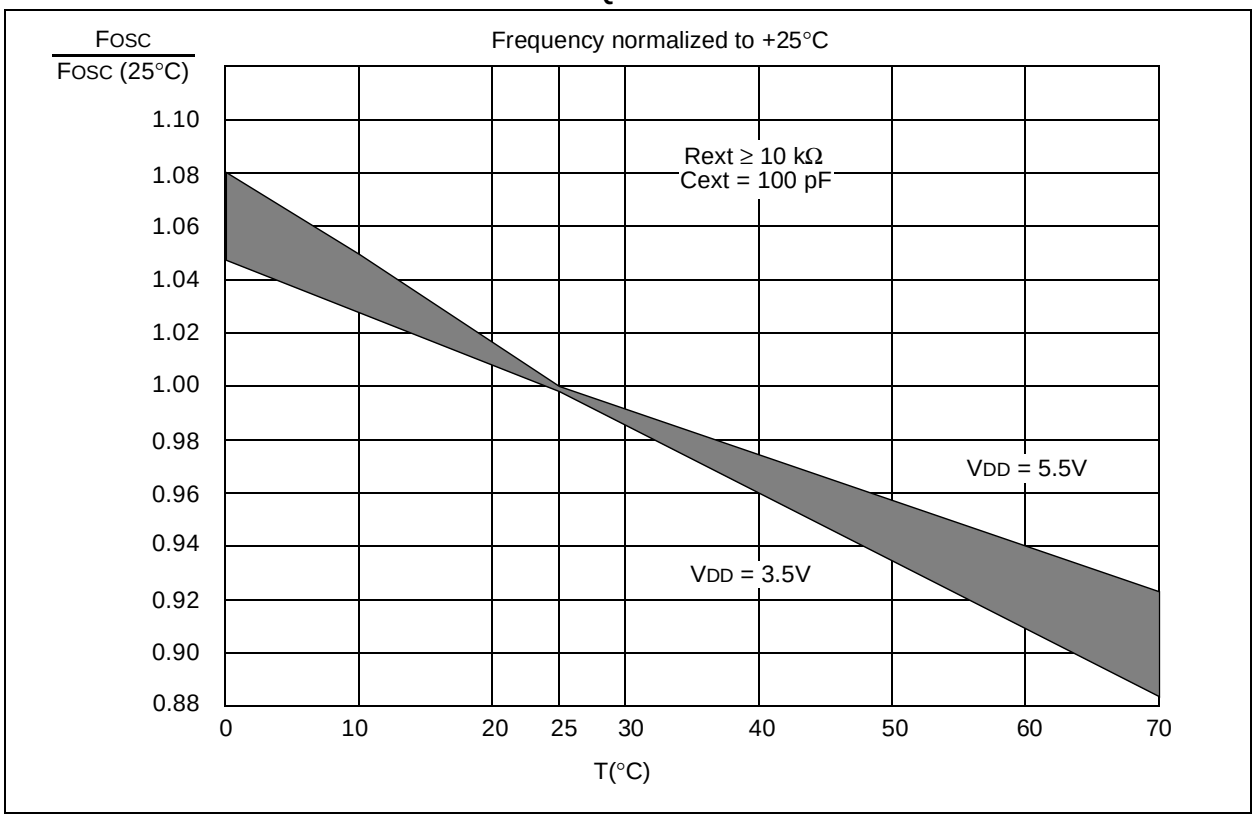


TABLE 16-1: RC OSCILLATOR FREQUENCIES

Cext	Rext	Average Fosc @ 5V, 25°C	
20 pF	3.3k	4.973 MHz	± 27%
	5k	3.82 MHz	± 21%
	10k	2.22 MHz	± 21%
	100k	262.15 kHz	± 31%
100 pF	3.3k	1.63 MHz	± 13%
	5k	1.19 MHz	± 13%
	10k	684.64 kHz	± 18%
	100k	71.56 kHz	± 25%
300 pF	3.3k	660.0 kHz	± 10%
	5.k	484.1 kHz	± 14%
	10k	267.63 kHz	± 15%
	160k	29.44 kHz	± 19%

Note 1: The frequencies are measured on DIP packages.

Note 2: 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 16-2: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

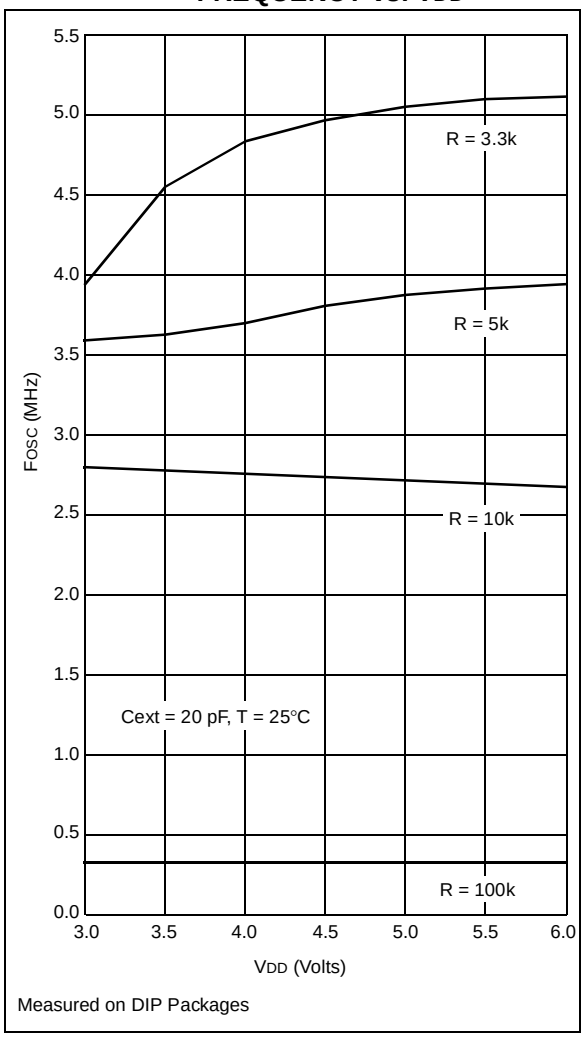


FIGURE 16-3: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

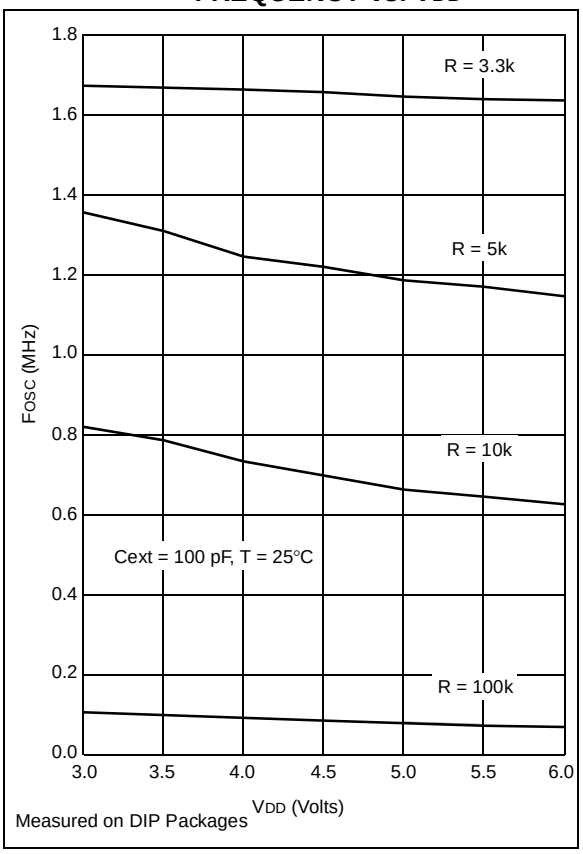


FIGURE 16-4: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD

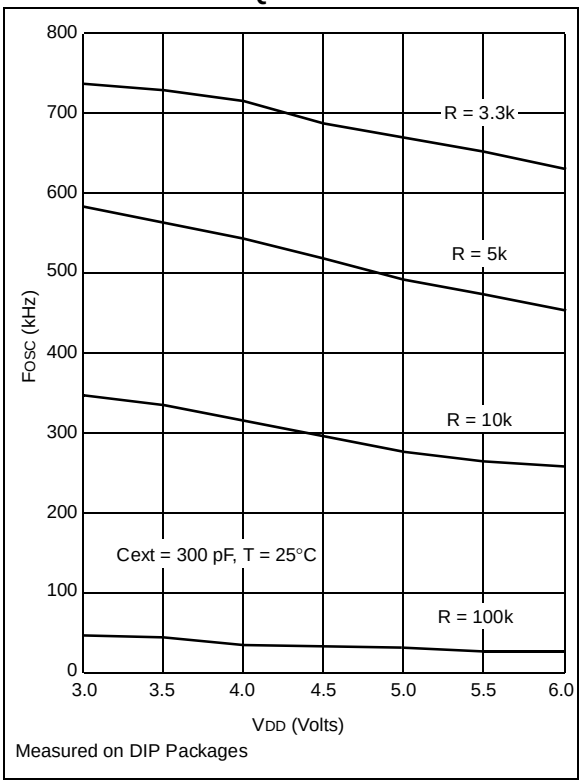


FIGURE 16-5: TYPICAL IPD vs. VDD WATCHDOG DISABLED 25°C

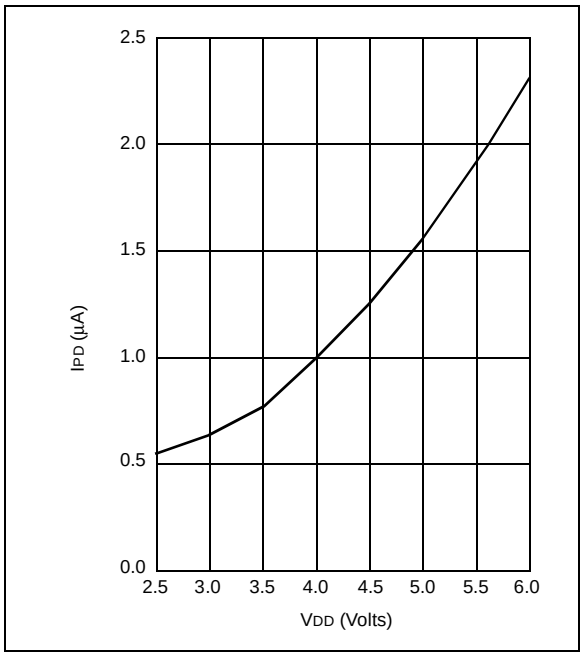


FIGURE 16-6: MAXIMUM IPD vs. VDD WATCHDOG DISABLED

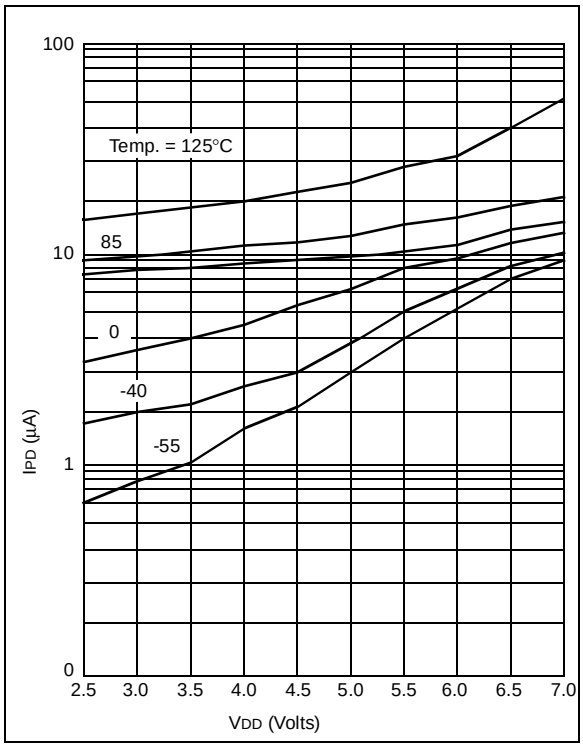


FIGURE 16-7: TYPICAL IPD vs. VDD WATCHDOG ENABLED 25°C

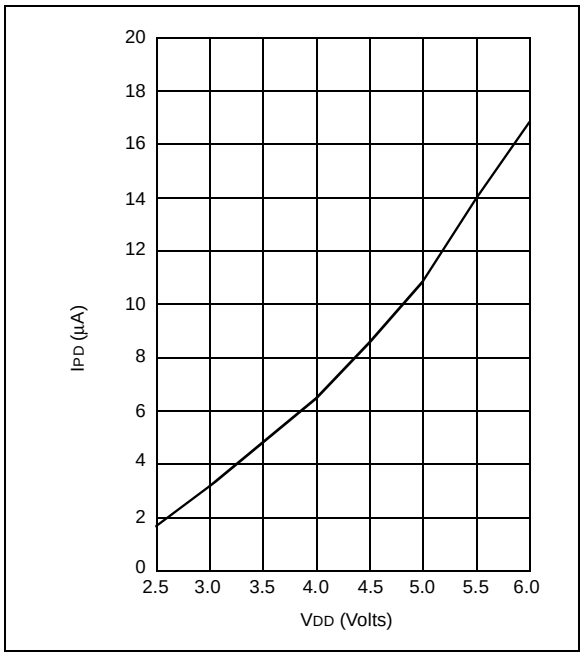
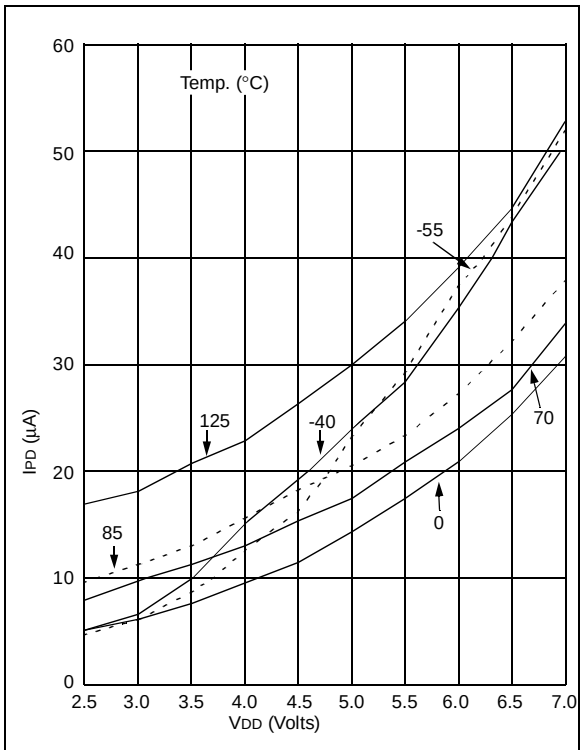


FIGURE 16-8: MAXIMUM IPD vs. VDD WATCHDOG ENABLED



IPD, with WDT enabled, has two components:
The leakage current which increases with higher temperature and the operating current of the WDT logic which increases with lower temperature. At -40°C, the latter dominates explaining the apparently anomalous behavior.

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

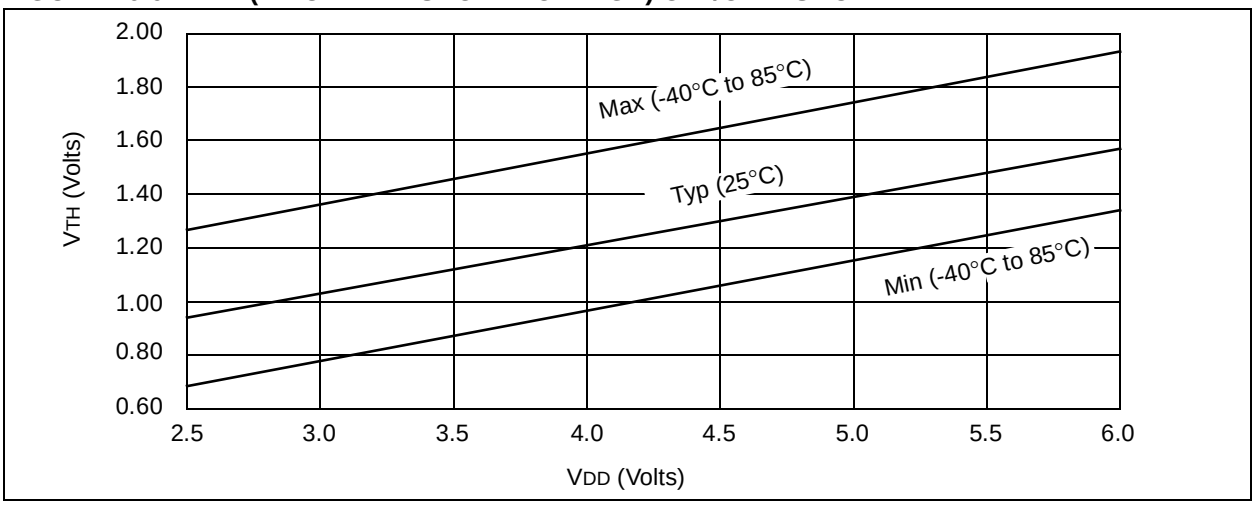
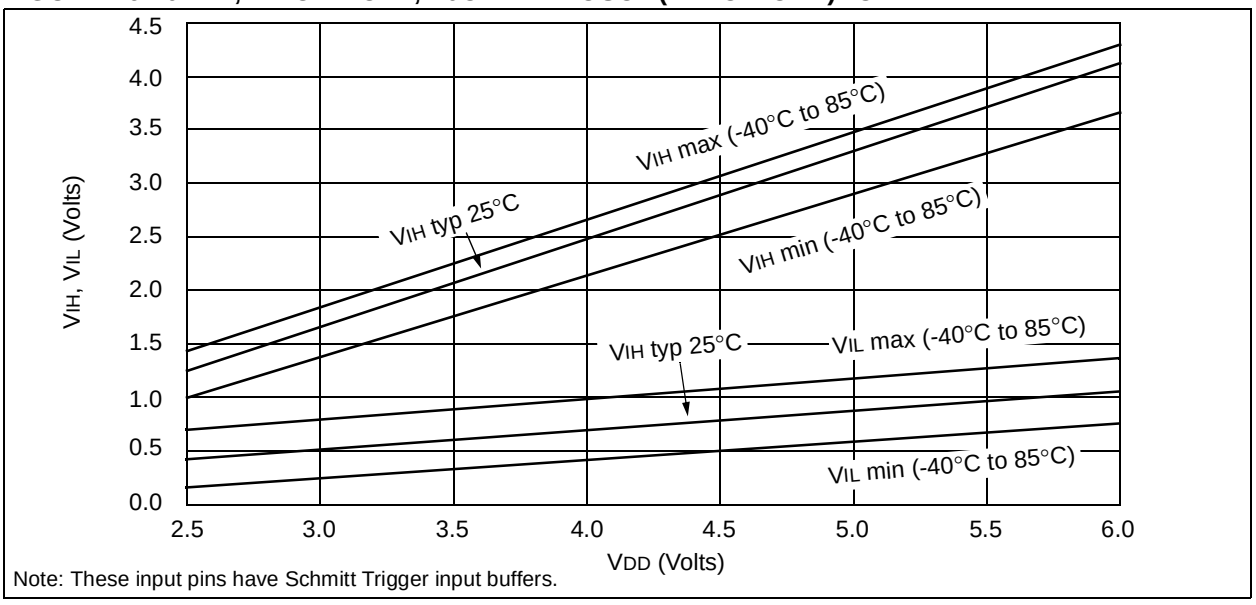


FIGURE 16-10: VIH, VIL OF $\overline{\text{MCLR}}$, T0CKI AND OSC1 (IN RC MODE) vs. VDD



Note: These input pins have Schmitt Trigger input buffers.

FIGURE 16-11: VTH (INPUT THRESHOLD VOLTAGE) OF I/O PINS vs. VDD

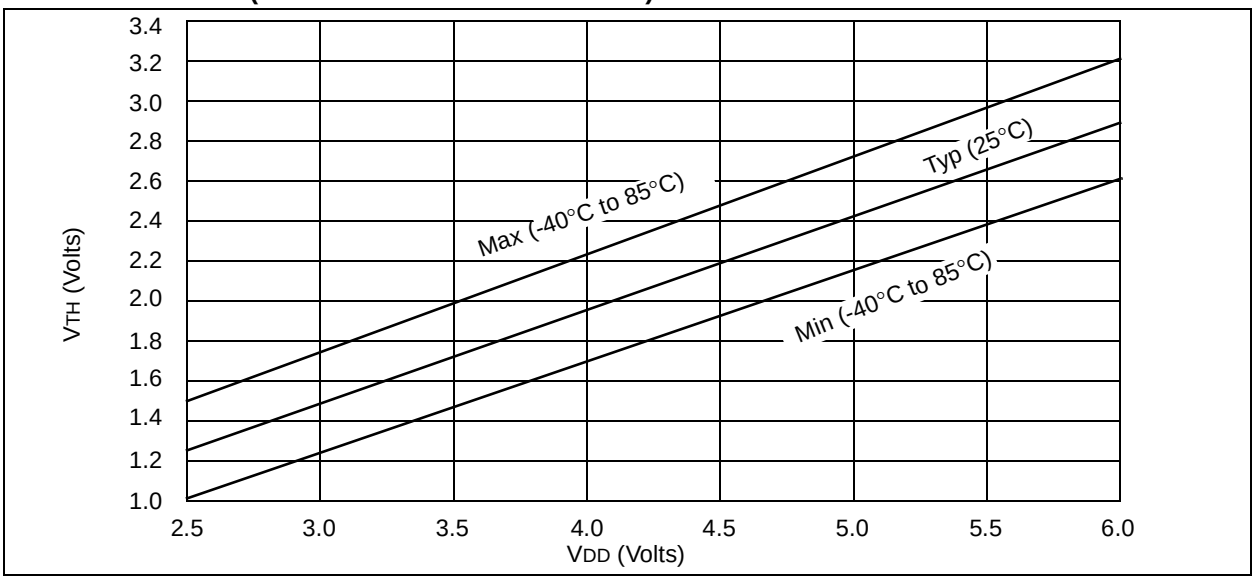


FIGURE 16-12: TYPICAL IDD vs. FREQUENCY (EXTERNAL CLOCK 25°C)

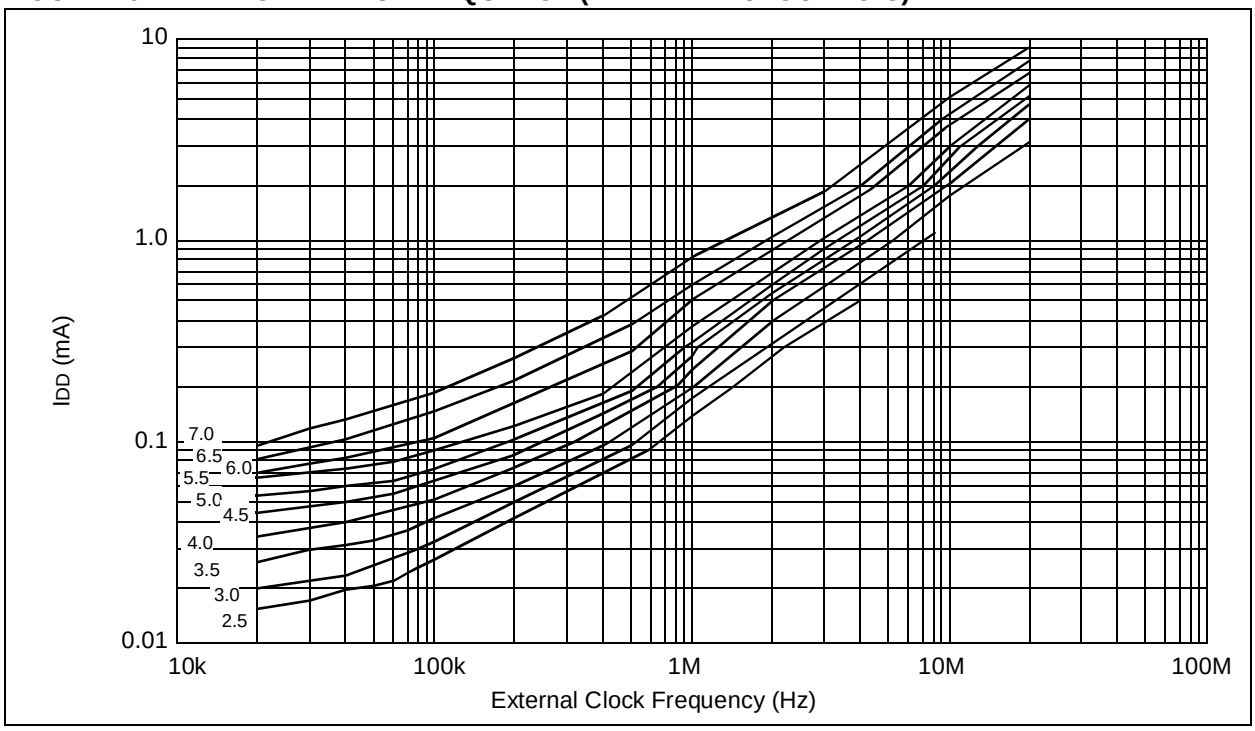


FIGURE 16-13: MAXIMUM IDD vs. FREQUENCY (EXTERNAL CLOCK -40°C TO +85°C)

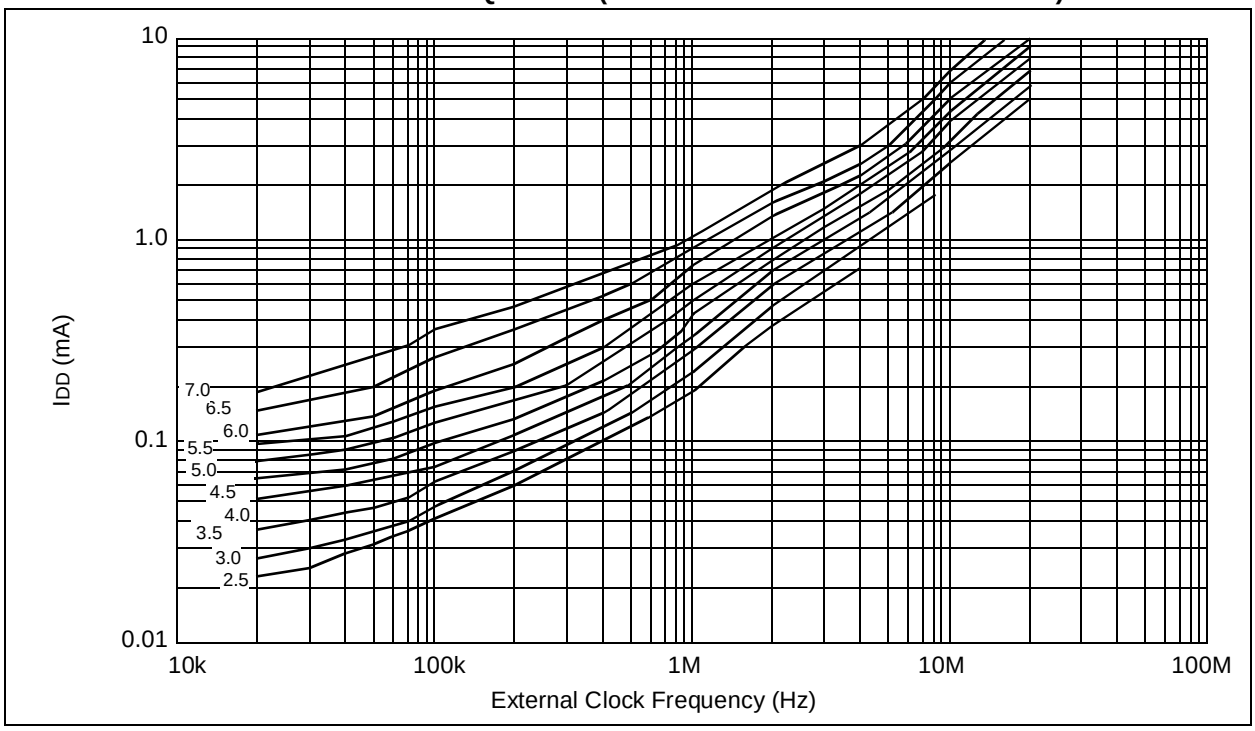


FIGURE 16-14: MAXIMUM IDD vs. FREQUENCY (EXTERNAL CLOCK -55°C TO +125°C)

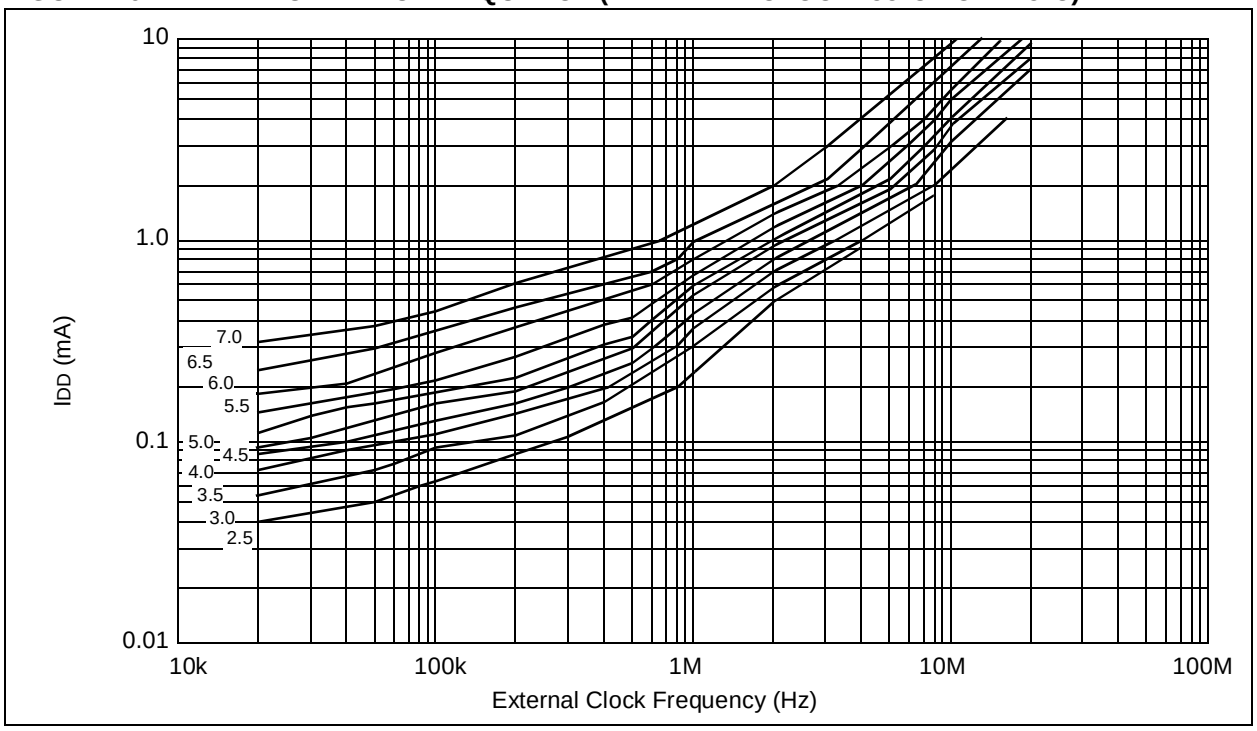


FIGURE 16-15: WDT TIMER TIME-OUT PERIOD vs. VDD

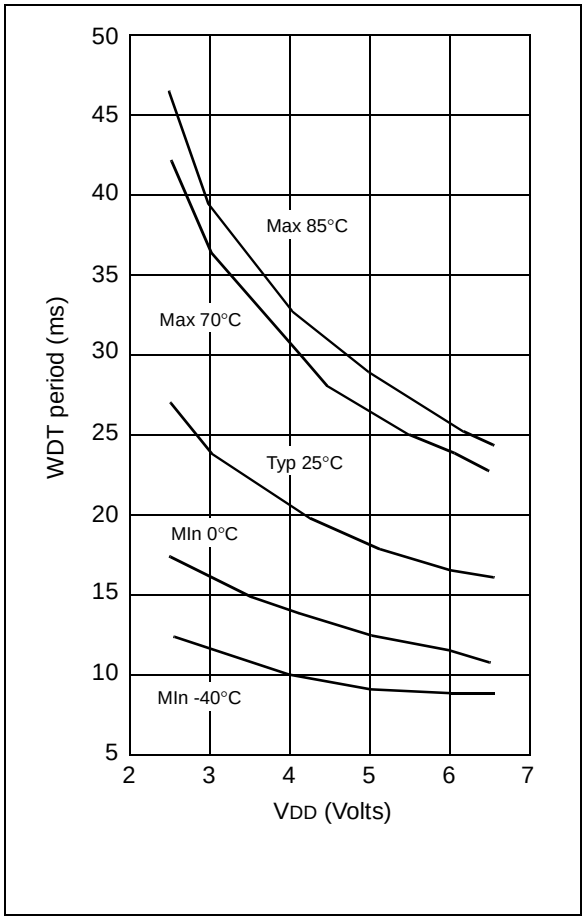


FIGURE 16-16: TRANSCONDUCTANCE (gm) OF HS OSCILLATOR vs. VDD

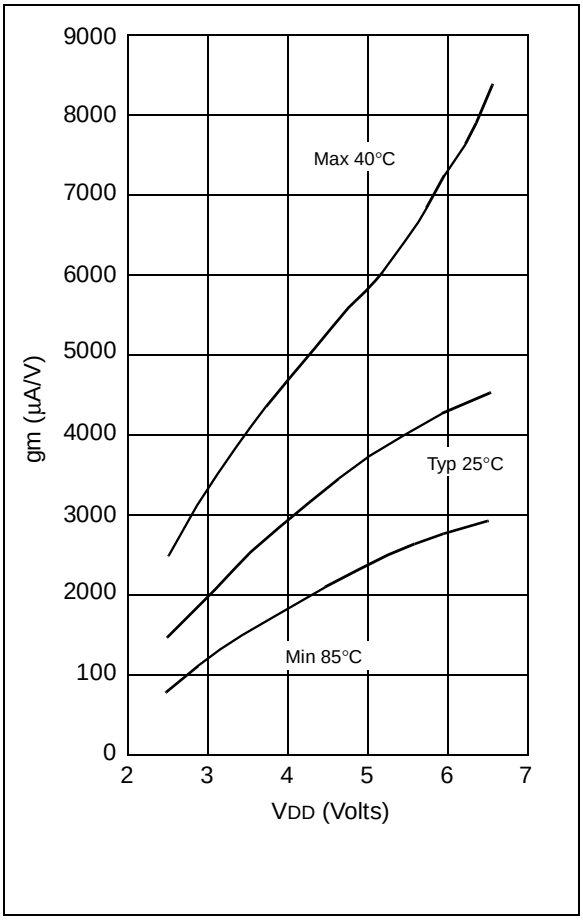


FIGURE 16-17: TRANSCONDUCTANCE (gm) OF LP OSCILLATOR vs. VDD

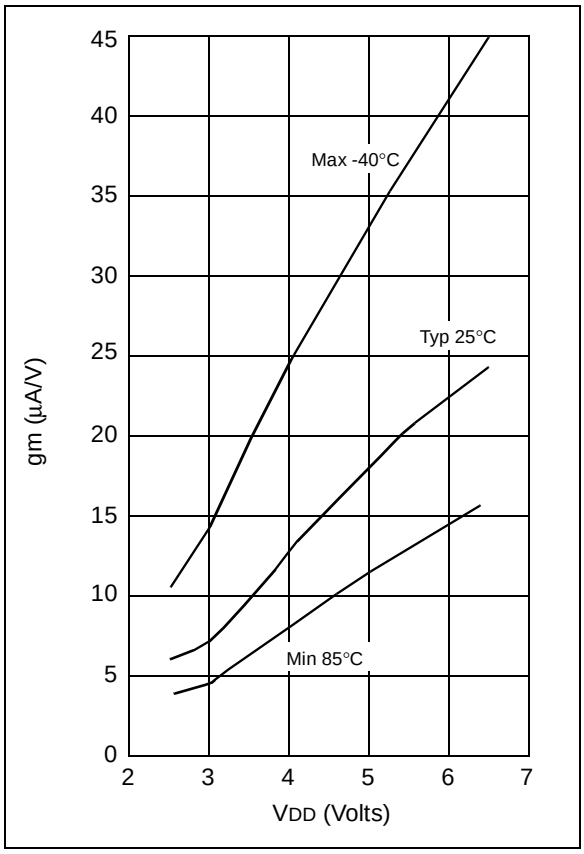


FIGURE 16-18: IOH vs. VOH, VDD = 3V

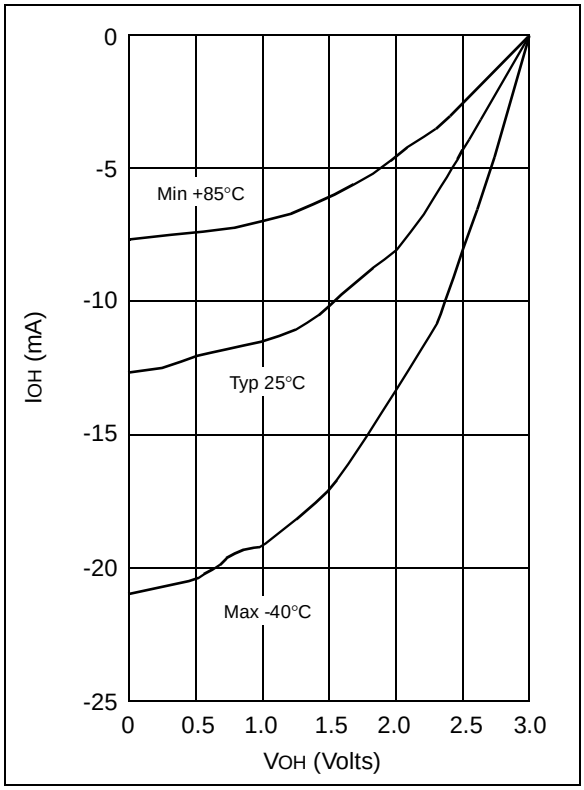


FIGURE 16-19: TRANSCONDUCTANCE (gm) OF XT OSCILLATOR vs. VDD

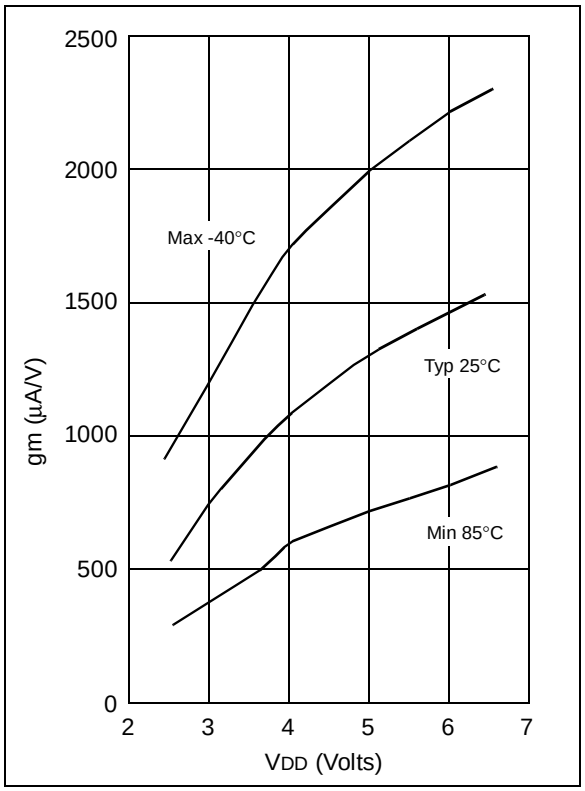


FIGURE 16-20: IOH vs. VOH, VDD = 5V

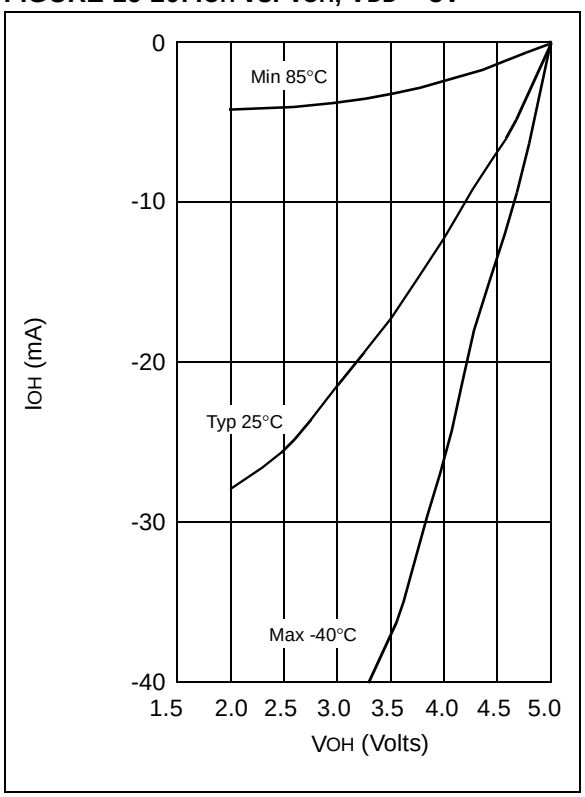


FIGURE 16-21: IOL vs. VOL, VDD = 3V

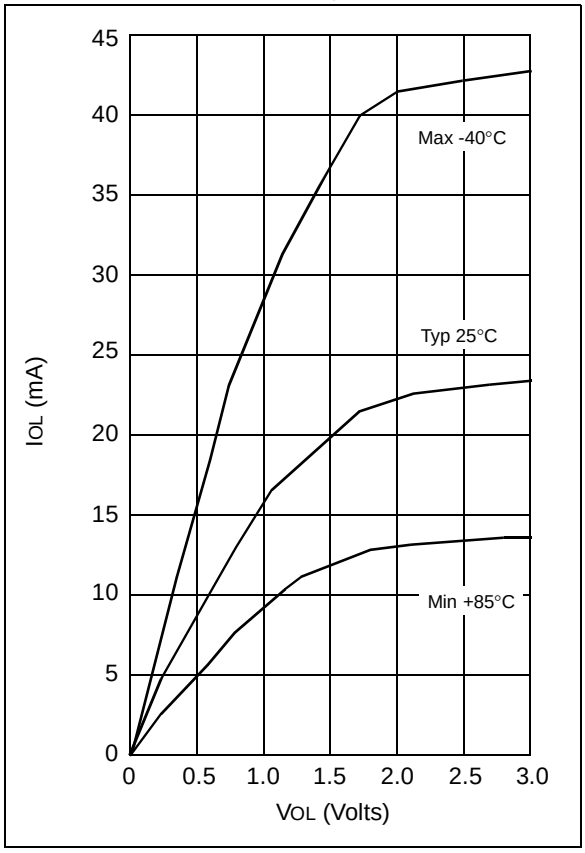
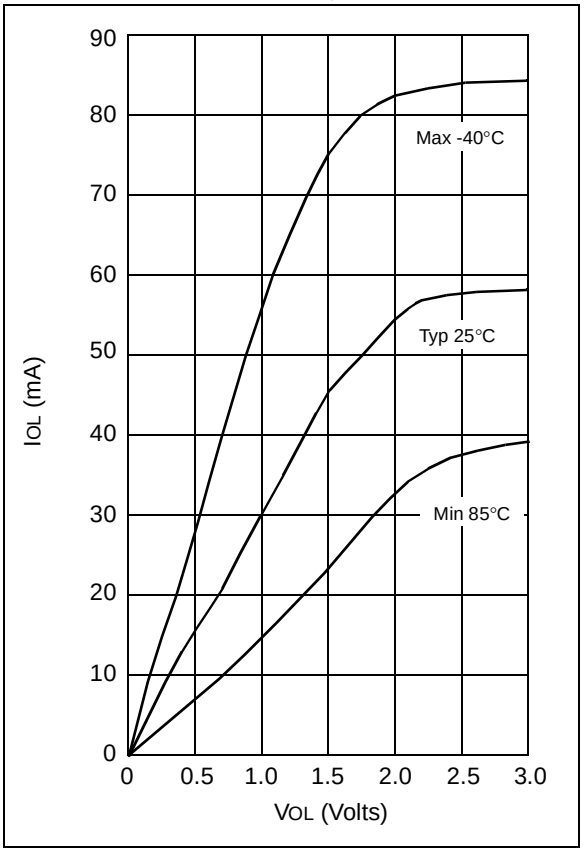


FIGURE 16-22: IOL vs. VOL, VDD = 5V



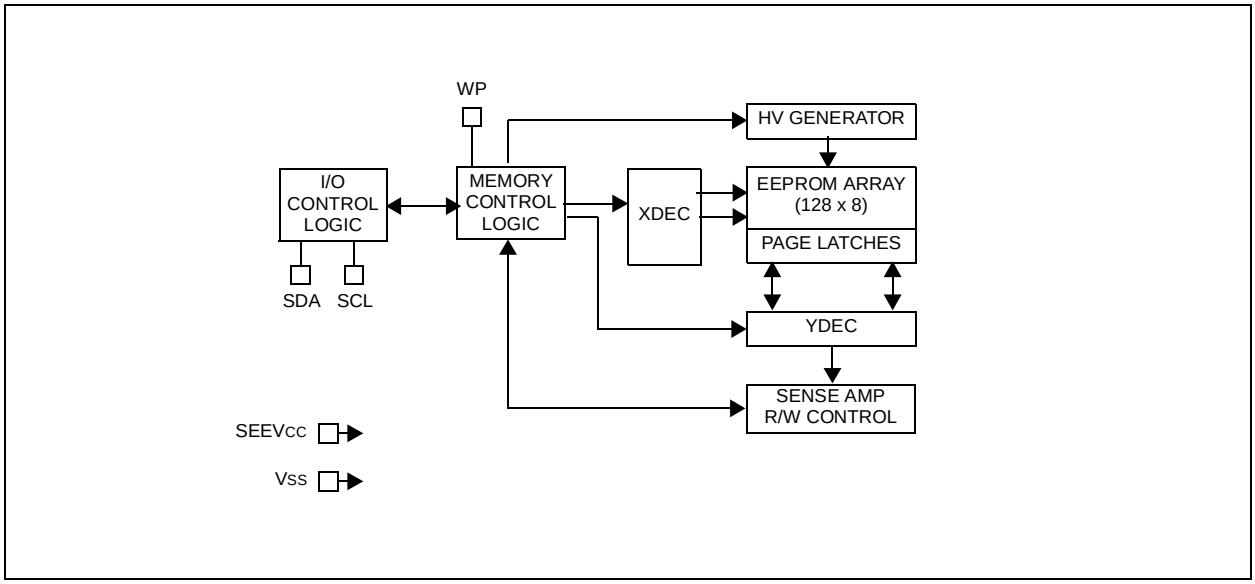
17.0 SERIAL EEPROM SECTION

The Microchip Technology Inc. MTA81010 contains a 24LC01B 1K-bit Electrically Erasable PROM. The SEEPROM is organized as a single block of 128 x 8-bit memory with a two-wire serial interface. Low voltage design permits operation down to 2.5V with a standby and active currents of only 5 μA and 1 mA respectively. The 24LC01B has page-write capability for up to 8 bytes of data.

TABLE 17-1: PIN FUNCTIONS

Name	Function
VSS	Ground
SDA	Serial Address/Data I/O
ACL	Serial Clock
WP	Write Protect Input
VCC	+2.5V to 5.5V Power Supply
A0, A1, A2	No Internal Connection

FIGURE 17-1: SERIAL EEPROM BLOCK DIAGRAM



18.0 SERIAL EEPROM FUNCTIONAL DESCRIPTION

The 24LC01B supports a bidirectional two wire bus and data transmission protocol. A device that sends data onto the bus is defined as transmitter, and a device receiving data as receiver. The bus has to be controlled by a master device which generates the serial clock (SCL), controls the bus access, and generates the START and STOP conditions, while the 24LC01B works as slave. The internal 24LC01B in the MTA81010 is hardwire configured with a device address of 0. Both, master and slave can operate as transmitter or receiver but the master device determines which mode is activated.

19.0 BUS CHARACTERISTICS

The following bus protocol has been defined:

- Data transfer may be initiated only when the bus is not busy.
- During data transfer, the data line must remain stable whenever the clock line is HIGH. Changes in the data line while the clock line is HIGH will be interpreted as a START or STOP.

Accordingly, the following bus conditions have been defined (Figure 19-1):

19.1 Bus not Busy (A)

Both data and clock lines remain HIGH.

19.2 Start Data Transfer (B)

A HIGH to LOW transition of the SDA line while the clock (SCL) is HIGH determines a START. All commands must be preceded by a START.

19.3 Stop Data Transfer (C)

A LOW to HIGH transition of the SDA line while the clock (SCL) is HIGH determines a STOP condition. All operations must be ended with a STOP condition.

19.4 Data Valid (D)

The state of the data line represents valid data when, after a START, the data line is stable for the duration of the HIGH period of the clock signal.

The data on the line must be changed during the LOW period of the clock signal. There is one clock pulse per bit of data.

Each data transfer is initiated with a START and terminated with a STOP. The number of the data bytes transferred between the START and STOP is determined by the master device and is theoretically unlimited, although only the last eight will be stored when doing a write operation. When an overwrite does occur it will replace data in a first in first out fashion.

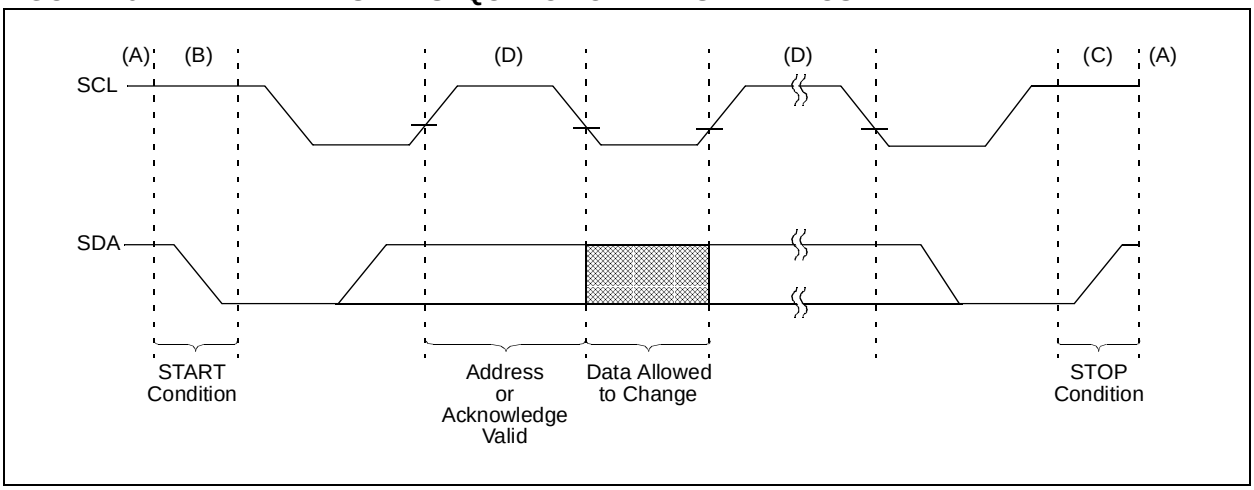
19.5 Acknowledge

Each receiving device, when addressed, is obliged to generate an acknowledge after the reception of each byte. The master device must generate an extra clock pulse which is associated with this acknowledge bit.

Note:	The 24LC01B does not generate any acknowledge bits if an internal programming cycle is in progress.

The device that acknowledges has to pull down the SDA line during the acknowledge clock pulse in such a way that the SDA line is stable LOW during the HIGH period of the acknowledge related clock pulse. Of course, setup and hold times must be taken into account. A master must signal an end of data to the slave by not generating an acknowledge bit on the last byte that has been clocked out of the slave. In this case, the slave must leave the data line HIGH to enable the master to generate the STOP.

FIGURE 19-1: DATA TRANSFER SEQUENCE ON THE SERIAL BUS



19.6 Slave Address

The chip address of the internal 24LC01B in the MTA81010 is hardwire configured with a device address of XXX.

After generating a START, the bus master transmits the slave address consisting of a 4-bit control code (1010) for the 24LC01B, followed by three don't care bits.

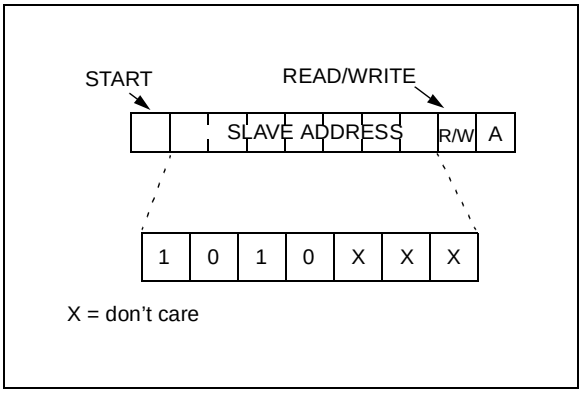
The eighth bit of slave address determines if the master device wants to read or write to the 24LC01B (Figure 19-2).

The 24LC01B monitors the bus for its corresponding control code all the time. It generates an acknowledge bit if the control code was true and it is not in a programming mode.

External devices (e.g., Serial EEPROMs) with the same control code as the internal 24LC01B must not be connected to the same serial bus or a conflict will occur. Devices with different control codes may be connected to the serial bus without risk of conflict.

Operation	Control Code	Chip Select	R/$\overline{W}$
Read	1010	XXX	1
Write	1010	XXX	0

FIGURE 19-2: CONTROL BYTE ALLOCATION



19.7 Byte Write

Following the start signal from the master, the device code (4-bits), the don't care bits (3-bits), and the R/W bit which is a logic low is placed onto the bus by the master transmitter. This indicates to the addressed slave receiver that a byte with a word address will follow after it has generated an acknowledge bit during the ninth clock cycle. Therefore the next byte transmitted by the master is the word address and will be written into the address pointer of the 24LC01B. After receiving another acknowledge signal from the 24LC01B the master device will transmit the data word to be written into the addressed memory location. The 24LC01B acknowledges again and the master generates a stop condition. This initiates the internal write cycle, and during this time the 24LC01B will not generate acknowledge signals (Figure 19-3).

19.8 Page Write

The write control byte, word address and the first data byte are transmitted to the 24LC01B in the same way as in a byte write. But instead of generating a stop condition the master transmits up to eight data bytes to the 24LC01B which are temporarily stored in the on-chip page buffer and will be written into the memory after the master has transmitted a stop condition. After the receipt of each word, the three lower order address pointer bits are internally incremented by one. The higher order five bits of the word address remains constant. If the master should transmit more than eight words prior to generating the stop condition, the address counter will roll over and the previously received data will be overwritten. As with the byte write operation, once the stop condition is received an internal write cycle will begin (Figure 19-4).

FIGURE 19-3: BYTE WRITE

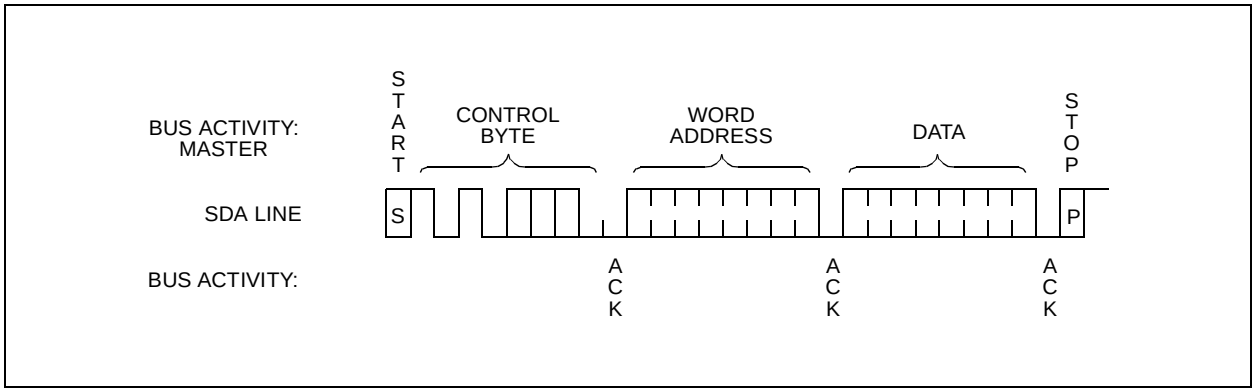


FIGURE 19-4: PAGE WRITE

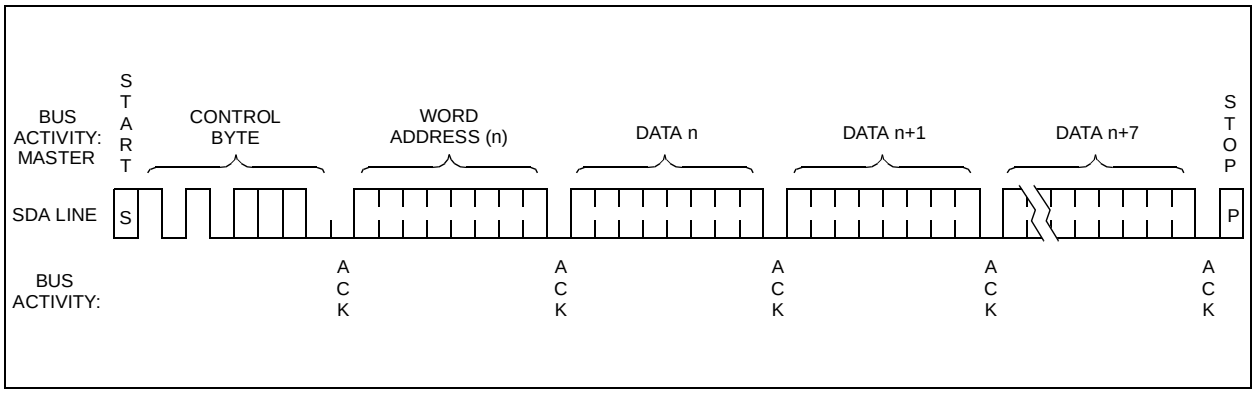


FIGURE 19-5: CURRENT ADDRESS READ

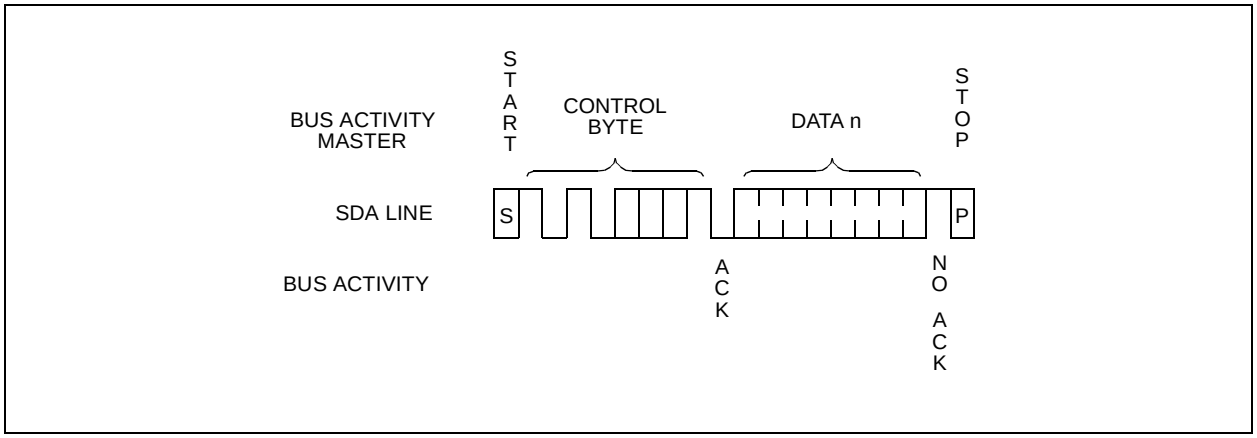


FIGURE 19-6: RANDOM READ

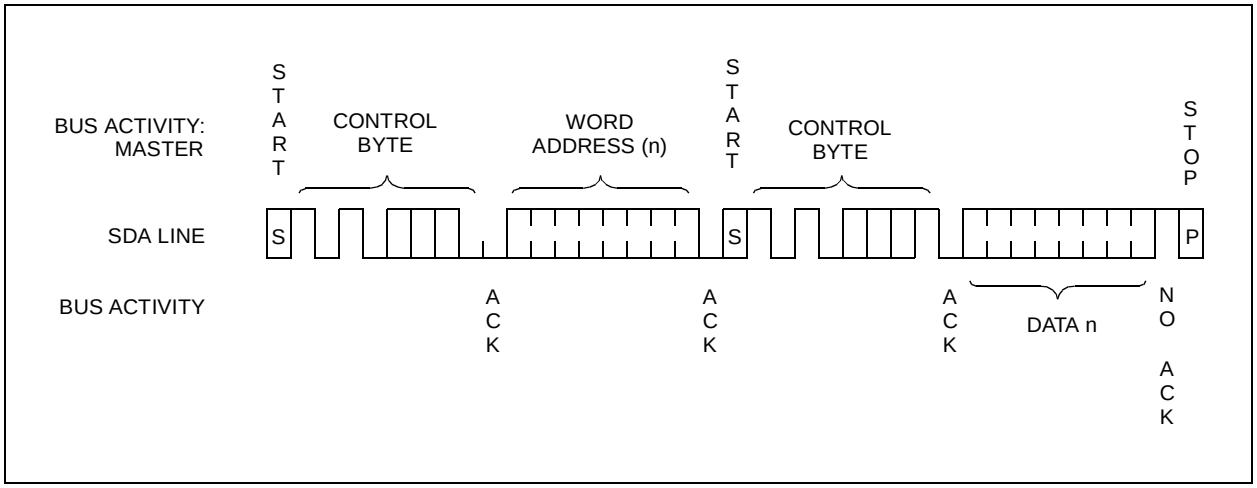
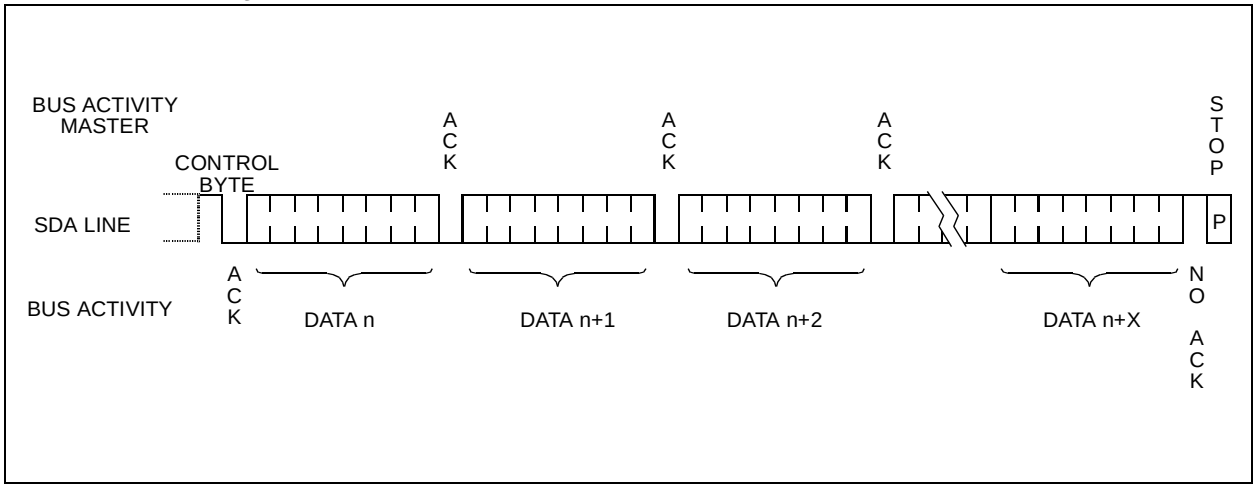


FIGURE 19-7: SEQUENTIAL READ



19.9 Write Protection

The 24LC01B can be used as a serial ROM when the WP pin is connected to SEEVDD. Programming will be inhibited and the entire memory will be write-protected.

19.10 Read Operation

Read operations are initiated in the same way as write operations with the exception that the R/W bit of the slave address is set to one. There are three basic types of read operations: current address read, random read, and sequential read.

19.10.1 CURRENT ADDRESS READ

The 24LC01B contains an address counter that maintains the address of the last word accessed, internally incremented by one. Therefore, if the previous access (either a read or write operation) was to address n, the next current address read operation would access data from address n + 1. Upon receipt of the slave address with R/W bit set to one, the 24LC01B issues an acknowledge and transmits the eight bit data word. The master will not acknowledge the transfer but does generate a stop condition and the 24LC01B discontinues transmission (Figure 19-5).

19.10.2 RANDOM READ

Random read operations allow the master 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 24LC01B as part of a write operation. After the word address is sent, the master generates a start condition following the acknowledge. This terminates the write operation, but not before the internal address pointer is set. Then the master issues the control byte again but with the R/W bit set to a one. The 24LC01B will then issue an acknowledge and transmits the eight bit data word. The master will not acknowledge the transfer but does generate a stop condition and the 24LC01B discontinues transmission (Figure 19-6).

19.10.3 SEQUENTIAL READ

Sequential reads are initiated in the same way as a random read except that after the 24LC01B transmits the first data byte, the master issues an acknowledge as opposed to a stop condition in a random read. This directs the 24LC01B to transmit the next sequentially addressed 8-bit word (Figure 19-7).

To provide sequential reads the 24LC01B contains an internal address pointer which is incremented by one at the completion of each operation. This address pointer allows the entire memory contents to be serially read during one operation.

19.10.4 NOISE PROTECTION

The 24LC01B employs a VDD threshold detector circuit which disables the internal erase/write logic if the VDD is below 1.5 volts at nominal conditions.

The SCL and SDA inputs have Schmitt trigger and filter circuits which suppress noise spikes to assure proper device operation even on a noisy bus.

20.0 ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings*

V_{CC} 7.0V

All inputs and outputs w.r.t. V_{SS} ..... -0.6V to V_{CC} +1.0V

Storage temperature ..... -65°C to +150°C

Ambient temperature with power applied ..... -65°C to +125°C

Soldering temperature of leads (10 seconds) ..... +300°C

ESD protection on all pins ..... ≥4 kV

*** Notice:** Stresses above those listed under “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 20-1: DC CHARACTERISTICS: 24LC01B

			V_{CC} = +2.5V to +5.5V Commercial (C): Tamb = 0°C to +70°C Industrial (I): Tamb = -40°C to +85°C		
Parameter	**Symbol**	**Min**	**Max**	**Units**	**Conditions**
WP, SCL and SDA pins: High level input voltage Low level input voltage Hysteresis of Schmitt trigger inputs Low level output voltage	 V_{IH} V_{IL} V_{HYS} V_{OL}	 0.7 V_{CC} — .05 V_{CC} —	 — 0.3 V_{DD} — .40	 V V V V	 Note 1 I_{OL} = 3.0 mA, V_{CC} = 2.5V
Input leakage current	I_{LI}	-10	10	µA	V_{IN} = 0.1V to 5.5V
Output leakage current	I_{LO}	-10	10	µA	V_{OUT} = 0.1V to 5.5V
Internal capacitance (all inputs/outputs)	C_{INT}	—	10	pF	V_{CC} = 5.0V (Note 1) Tamb = 25°C, F_{CLK} = 1 MHz
Operating current	I_{CC} Write I_{CC} Read	— —	3 1	mA mA	V_{CC} = 5.5V, SCL = 400 kHz
Standby current	I_{CCS}	— —	30 100	µA µA	V_{CC} = 3.0V, SDA = SCL =V_{CC} V_{CC} = 5.5V, SDA = SCL =V_{CC}

Note 1: This parameter is periodically sampled and not 100% tested.

FIGURE 20-1: BUS TIMING START/STOP

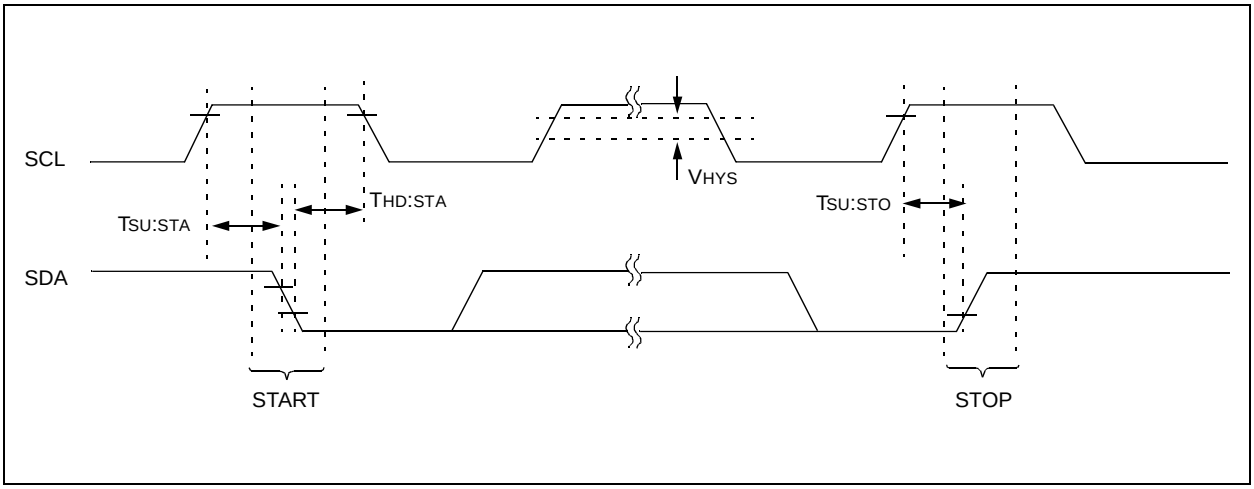


TABLE 20-2: AC CHARACTERISTICS: 24LC01B

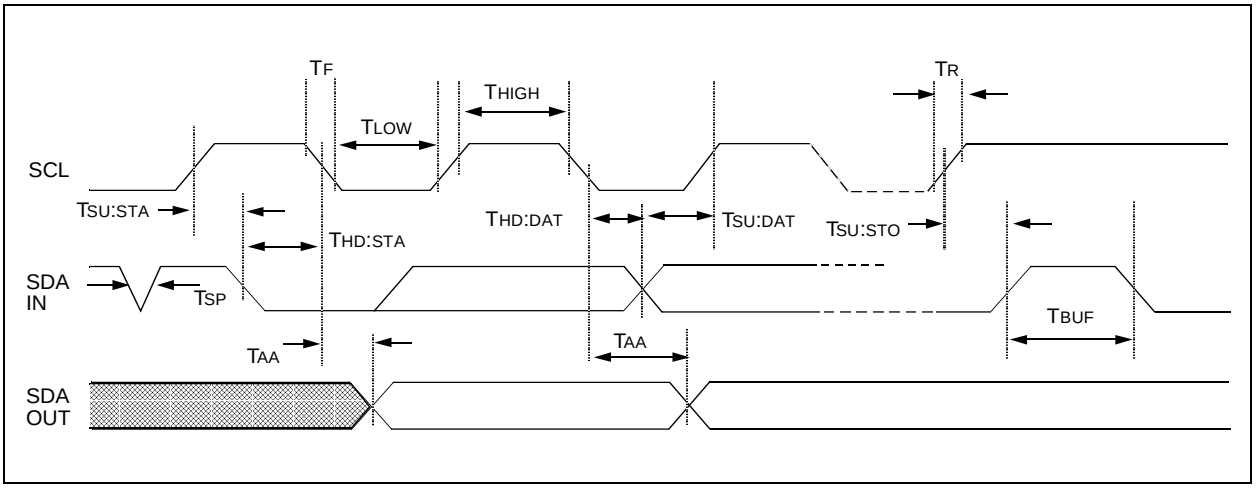
Parameter	Symbol	STANDARD MODE		Vcc = 4.5-5.5V FAST MODE		Units	Remarks
		Min	Max	Min	Max		
Clock frequency	FCLK	0	100	0	400	kHz	
Clock high time	THIGH	4000	—	600	—	ns	
Clock low time	TLOW	4700	—	1300	—	ns	
SDA and SCL rise time	TR	—	1000	20 + 0.1CB	300	ns	Note 2
SDA and SCL fall time	TF	—	300	20 + 0.1CB	300	ns	Note 2
START condition hold time	THD:STA	4000	—	600	—	ns	After this period the first clock pulse is generated
START condition setup time	TSU:STA	4700	—	600	—	ns	Only relevant for repeated START condition
Data input hold time	THD:DAT	0	—	0	—	ns	
Data input setup time	TSU:DAT	250	—	100	—	ns	
STOP condition setup time	TSU:STO	4000	—	600	—	ns	
Output valid from clock	TAA	—	3500	—	900	ns	Note 1
Bus free time	TBUF	4700	—	1300	—	ns	Time the bus must be free before a new transmission can start
Output fall time from VIH minimum to VIL maximum	TOF	—	250	20 + 0.1CB	250	ns	Note 2, CB ≤100 pF
Input filter spike suppression (SDA and SCL pins)	TSP	N/A	N/A	0	50	ns	Note 3
Write cycle time	TWR	—	10	—	10	ms	Byte or Page mode

Note 1: As a transmitter, the device must provide an internal minimum delay time to bridge the undefined region (minimum 300 ns) of the falling edge of SCL to avoid unintended generation of START or STOP conditions.

Note 2: Not 100 percent tested. CB = total capacitance of one bus line in pF.

Note 3: The combined TSP and VHYS specifications are due to new Schmitt trigger inputs which provide improved noise and spike suppression. This eliminates the need for a TI specification for standard operation.

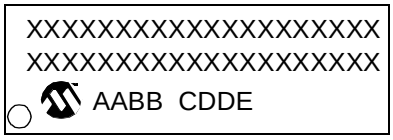
FIGURE 20-2: BUS TIMING DATA



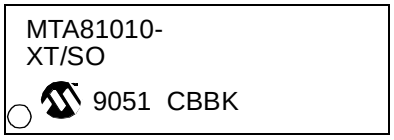
21.0 PACKAGING DIAGRAMS AND DIMENSIONS

21.1 PACKAGE MARKING INFORMATION

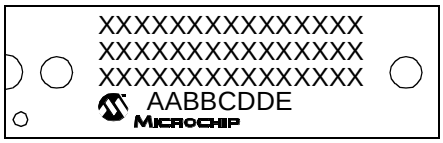
28-Lead SOIC



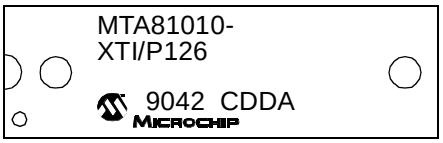
Example



28-Lead PDIP (.600 mil)



Example



Legend:	MM...M	Microchip part number information
	XX...X	Customer specific information*
	AA	Year code (last 22 digits of calender 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.
	D_1	Mask revision number for microcontroller
	D_2	Mask revision number for EEPROM
	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.

INDEX

List of Examples

Table of Figures

Table of Tables

CONNECTING TO MICROCHIP BBS

Connect worldwide to the Microchip BBS using the CompuServe® communications 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 BBS, except for a toll charge to the CompuServe access number, where applicable. You do not need to be a CompuServe member to take advantage of this connection (you never actually log in to CompuServe).

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 allows multiple users at baud rates up to 14400 bps.

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 **<ENTER↵>** and a garbage string will appear because CompuServe is expecting a 7E1 setting.
4. Type **+,** depress **<ENTER↵>** and `Host Name:` will appear.
5. Type **MCHIPBBS,** depress **< ENTER↵ >** and you will be connected to the Microchip BBS.

In the United States, to find CompuServe's 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 **< ENTER↵ >**
and follow CompuServe's directions.

For voice information (or calling from overseas), you may call (614) 457-1550 for your local CompuServe number.

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 area code (602) 786-7578.

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

To: Technical Publications Manager Total Pages Sent _____

RE: Reader Response

From: Name _____

Company _____

Address _____

City / State / ZIP / Country _____

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

Application (optional):

Would you like a reply? ___Y ___N

Device: MTA81010 Literature Number: DS39005D

Questions:

1. What are the best features of this document?

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

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

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

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

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

7. How would you improve this document?

8. How would you improve our software, systems, and silicon products?

NOTES:

NOTES:

MTA81010 Product Identification System

To order or to obtain information (e.g., on pricing or delivery), please use the listed part numbers, and refer to the factory or the listed sales offices.

PART NO. -XX X /XX XXX

Pattern:	3-Digit Pattern Code for QTP (blank for OTP)	
Package:	P = PDIP SO = 300 mil SOIC (Gull Wing lead)	
Temperature Range:	- = 0°C to +70°C (T for tape/reel) I = –40°C to +85°C (S for tape/reel)	
Oscillator Type:	RC XT LP	
Device:	MTA81010	

Examples:

a) MTA81010 - XT/PXXX = “XT” oscillator, commercial temp., PDIP package, QTP pattern

b) MTA81010 - XT/PXXX = “XT” oscillator, industrial temp., PDIP package, QTP pattern

AMERICAS

Corporate Office
Microchip Technology Inc.
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 602 786-7200 Fax: 602 786-7277

Atlanta
Microchip Technology Inc.
500 Sugar Mill Road, Suite 200B
Atlanta, GA 30350
Tel: 404 640-0034 Fax: 404 640-0307

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: 708 285-0071 Fax: 708 285-0075

Dallas
Microchip Technology Inc.
14651 Dallas Parkway, Suite 816
Dallas, TX 75240-8809
Tel: 214 991-7177 Fax: 214 991-8588

Dayton
Microchip Technology Inc.
35 Rockridge Road
Englewood, OH 45322
Tel: 513 832-2543 Fax: 513 832-2841

Los Angeles
Microchip Technology Inc.
18201 Von Karman, Suite 455
Irvine, CA 92715
Tel: 714 263-1888 Fax: 714 263-1338

New York
Microchip Technology Inc.
150 Motor Parkway, Suite 416
Hauppauge, NY 11788
Tel: 516 273-5305 Fax: 516 273-5335

AMERICAS (continued)

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

ASIA/PACIFIC

Hong Kong
Microchip Technology Inc.
Unit No. 3002-3004, Tower 1
Metroplaza
223 Hing Fong Road
Kwai Fong, N.T. Hong Kong
Tel: 852 2 401 1200 Fax: 852 2 401 3431

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

Singapore
Microchip Technology Taiwan
Singapore Branch
200 Middle Road
#10-03 Prime Centre
Singapore 0718
Tel: 65 334 8870 Fax: 65 334 8850

Taiwan
Microchip Technology Taiwan
10F-1C 207
Tung Hua North Road
Taipei, Taiwan, ROC
Tel: 886 2 717 7175 Fax: 886 2 545 0139

EUROPE

United Kingdom
Arizona Microchip Technology Ltd.
Unit 6, The Courtyard
Meadow Bank, Furlong Road
Bourne End, Buckinghamshire
SL8 5AJ
Tel: 44 0 1628 851077 Fax: 44 0 1628 850259

France
Arizona Microchip Technology SARL
2 Rue du Buisson aux Fraises
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 Muenchen, Germany
Tel: 49 89 627 144 0 Fax: 49 89 627 144 44

Italy
Arizona Microchip Technology SRL
Centro Direzionale Colleoni
Palazzo Pegaso Ingresso No. 2
Via Paracelso 23, 20041
Agrate Brianza (MI) Italy
Tel: 39 039 689 9939 Fax: 39 039 689 9883

JAPAN

Microchip Technology Intl. Inc.
Benex S-1 6F
3-18-20, Shin Yokohama
Kohoku-Ku, Yokohama
Kanagawa 222 Japan
Tel: 81 45 471 6166 Fax: 81 45 471 6122

MICROCHIP

Printed in the USA, 8/95