NEAR EAST UNIVERSITY

Faculty of Engineering

Department of Electrical and Electronic Engineering

MOVING MESSAGE WITH PIC

Graduation Project
EE-400

Students: Remzi Uzun (20031692)

Supervisor: Asst. Prof. Kadri BÜRÜNCÜK

Lefkoşa - 2007

TABLE OF CONTENTS

ACKNOWLEDGMENTS

First I want to thank Asist. Prof. Dr. Kadri BÜRÜNCÜK to be my advisor. Under this guidance, I succesfully overcome many difficulties and learn a lot about Pic Programming. In each discussion, he showed from where I can take help, and I felt my quick progress from his advises. He always helps me a lot either in my study or my life. I asked him many questions in elektronics and electrics and he always answered my questions guickly and in details. Also I want to thank Asst. Prof. Dr. Özgür Cemal ÖZERDEM to be my supervisor. He helped me as much as Mr. Kadri.

Special thanks to Eren Electronics. With their help, I could find every kind of materials and documents easily.

I also want to thank my friends helped me at this project in NEU: Armağan and Uğur. And thanks to Prof. Dr. Doğan IBRAHIM.

Finally, I want to thank my family, especially my parents. Without their endless support and love for me, I would never achieve my current position. I wish my mother and my father lives happily always.

ABSTRACT

As the technology age has affected every aspect of our life, the need for electronic systems has raised. Electronics systems started to use in every application. So that electronic is in developmental stage.

This Project is concerned about using programmable chip systems to help to smooth our needs in life. These kind of projects are used in electronic advertisement, robotic projects, control the devices with telephon systems, clock and calendar applications and etc.

Big electronic boards with too much electronic components change-place with small elecronic boards and small programmable chips which needs only a few electronic components.

INTRODUCTION

The programmable chips are most popular in microelectronics. There are a lot of types of these microelectronic devices and they have different usage area depends of their production types. PIC is the one type of these chips. Also PIC's have a lot of types with different configurations.

PIC's can be programmable easily and these made a huge usage area to pics. A lot of program languages can be used to program pics. Compilers can turn the different languages to language which pic can be understand.

Also LCD monitors have high technology and they used with pics in kinds of applications like showing the every measurable values, clock and calendar applications, showing text messages and all storable information.

The Project application is moving message on lcd screen with using pic. Also we can use group of leds which are placed in columns and rows to show the message. These kind of applications mostly used in signboards. Messages can be placed on signboard from left to right, right to left, up to down, down to up with some changes of the software.

LCD monitors shows the messages with computer servers except the pics. It is mostly used when the message signal often have to be changed. But it is costly apllication. Because of that for unalterable messages pics are used. They made the system cheaper and simple than computer server systems.

1. PIC16F84A 18-PIN ENHANCED FLASH/EEPROM 8-BIT MICROCONTROLLER

1.1. High Performance RISC CPU Features:

- Only 35 single word instructions to learn
- All instructions single-cycle except for program branches which are two-cycle
- Operating speed: DC - 20 MHz clock input
  DC - 200 ns instruction cycle
- 1024 words of program memory
- 68 bytes of Data RAM
- 64 bytes of Data EEPROM
- 14-bit wide instruction words
- 8-bit wide data bytes
- 15 Special Function Hardware registers
- Eight-level deep hardware stack
- Direct, indirect and relative addressing modes
- Four interrupt sources:
  - -External RB0/INT pin
  - -TMR0 timer overflow
  - -PORTB<7:4> interrupt-on-change
  - -Data EEPROM write complete

1.2. Peripheral Features:

- 13 I/O pins with individual direction control
- High current sink/source for direct LED drive
  - -25 mA sink max. per pin
  - -25 mA source max. per pin
- TMR0: 8-bit timer/counter with 8-bit programmable prescaler

1.3. Special Microcontroller Features:

- 10,000 erase/write cycles Enhanced FLASH Program memory typical
- 10,000,000 typical erase/write cycles EEPROM Data memory typical
- EEPROM Data Retention > 40 years
- In-Circuit Serial Programming™ (ICSP™) - via two pins
- Power-on Reset (POR), Power-up Timer (PWRT), Oscillator Start-up Timer (OST)
- Watchdog Timer (WDT) with its own On-Chip RC Oscillator for reliable operation
- Code protection
- Power saving SLEEP mode
- Selectable oscillator options

1.4. Pin Diagrams

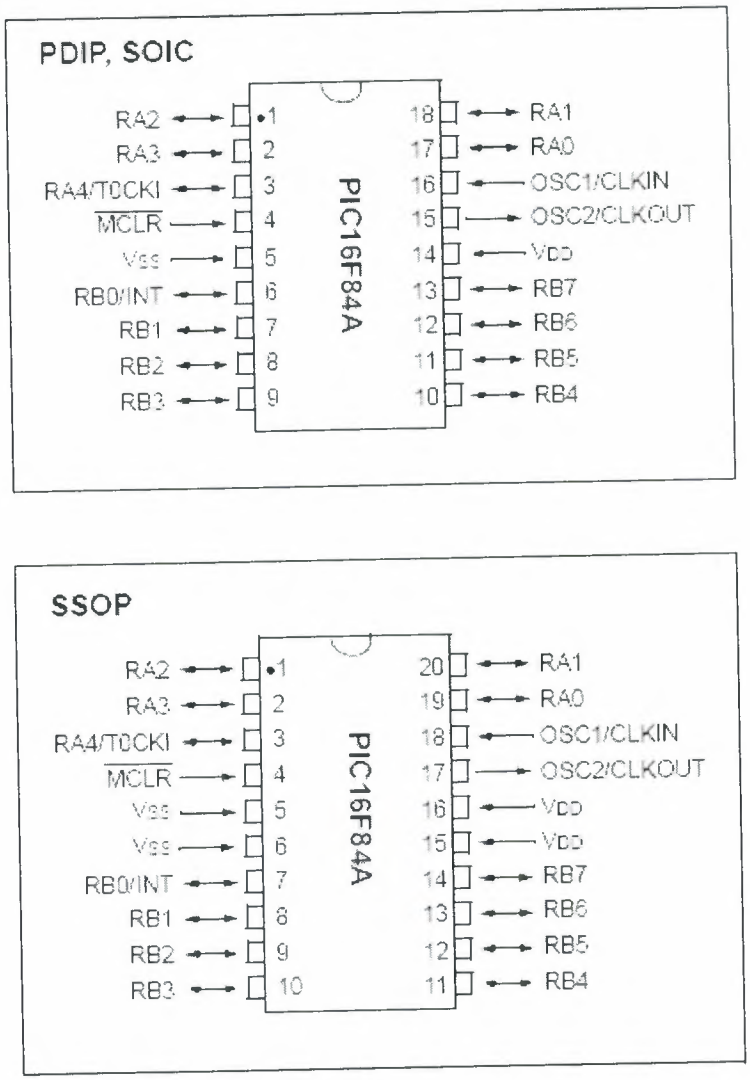


Figure 1.1 Pin diagrams

1.5. CMOS Enhanced FLASH/EEPROM Technology:

- Low power, high speed technology
- Fully static design
- Wide operating voltage range:
  - -Commercial: 2.0V to 5.5V
  - -Industrial: 2.0V to 5.5V
- Low power consumption:
  - -< 2 mA typical at 5V, 4 MHz
  - -15 µA typical at 2V, 32 kHz
  - -< 0.5 µA typical standby current at 2V

2. DEVICE OVERVIEW

The PIC16F84A belongs to the mid-range family of the PICmicro® microcontroller devices. A block diagram of the device is shown in **Figure 2.1**.

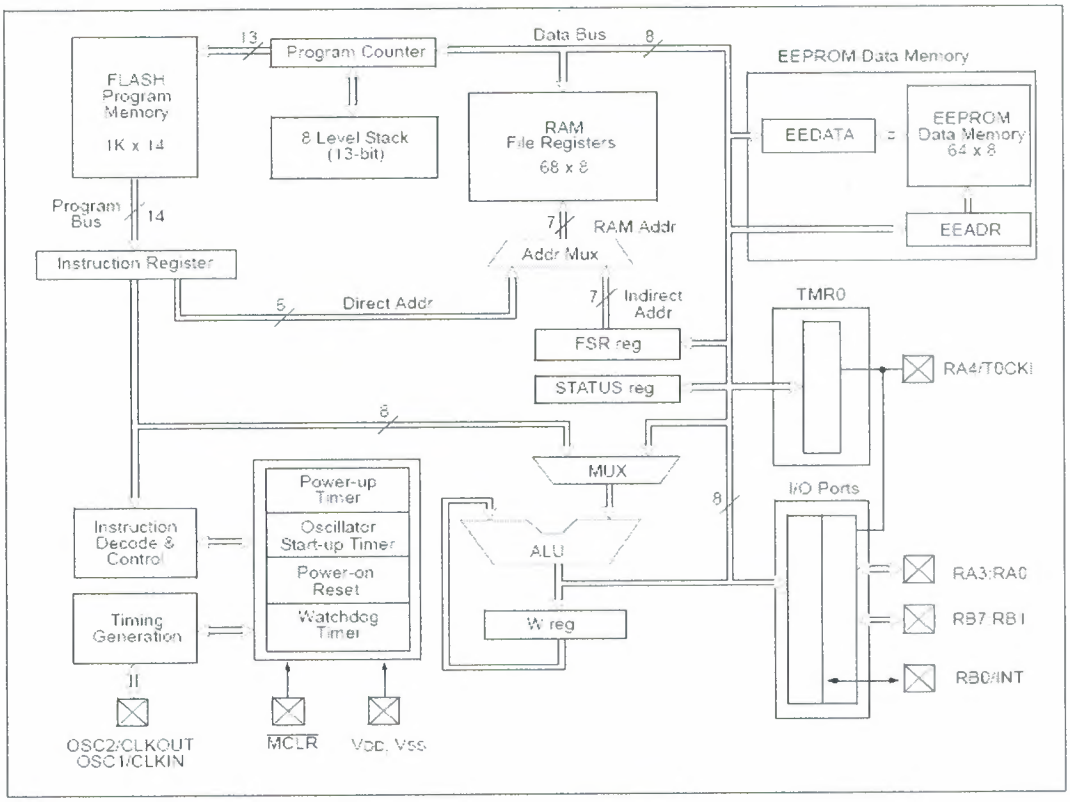


Figure 2.1 PIC 16F84A block diagram

The program memory contains 1K words, which translates to 1024 instructions, since each 14-bit program memory word is the same width as each device instruction. The data memory (RAM) contains 68 bytes. Data EEPROM is 64 bytes.

There are also 13 I/O pins that are user-configured on a pin-to-pin basis. Some pins are multiplexed with other device functions. These functions include:

- External interrupt
- Change on PORTB interrupt
- Timer0 clock input

Table 2.1 shows details the pinout of the device with descriptions and details for each pin.

Pin Name	PDIP No.	SOIC No.	SSOP No.	I/O/P Type	Buffer Type	Description
OSC1/CLKIN	16	16	18	I	ST/CMOS(3)	Oscillator crystal input/external clock source input.
OSC2/CLKOUT	15	15	19	O	—	Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode. In RC mode, OSC2 pin outputs CLKOUT, which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate.
MCLR	4	4	4	I/P	ST	Master Clear (Reset) input/programming voltage input. This pin is an active low RESET to the device.
						PORTA is a bi-directional I/O port.
RA0	17	17	19	I/O	TTL	
RA1	18	18	20	I/O	TTL	
RA2	1	1	1	I/O	TTL	
RA3	2	2	2	I/O	TTL	
RA4/T0CKI	3	3	3	I/O	ST	Can also be selected to be the clock input to the TMR0 timer/counter. Output is open drain type.
						PORTB is a bi-directional I/O port. PORTB can be software programmed for internal weak pull-up on all inputs.
RB0/INT	6	6	7	I/O	TTL/ST(1)	RB0/INT can also be selected as an external interrupt pin.
RB1	7	7	8	I/O	TTL	
RB2	8	8	9	I/O	TTL	
RB3	9	9	10	I/O	TTL	
RB4	10	10	11	I/O	TTL	Interrupt-on-change pin.
RB5	11	11	12	I/O	TTL	Interrupt-on-change pin.
RB6	12	12	13	I/O	TTL/ST(2)	Interrupt-on-change pin. Serial programming clock.
RB7	13	13	14	I/O	TTL/ST(2)	Interrupt-on-change pin. Serial programming data.
Vss	5	5	5,6	P	—	Ground reference for logic and I/O pins.
VDD	14	14	15,16	P	—	Positive supply for logic and I/O pins.

Table 2.1 PIC16F84A pinout description

Legend: I= input O = Output I/O = Input/Output P = Power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured in RC oscillator mode and a CMOS input otherwise.

3. MEMORY ORGANIZATION

There are two memory blocks in the PIC16F84A. These are the program memory and the data memory. Each block has its own bus, so that access to each block can occur during the same oscillator cycle.

The data memory can further be broken down into the general purpose RAM and the Special Function Registers (SFRs). The operation of the SFRs that control the "core" are described here. The SFRs used to control the peripheral modules are described in the section discussing each individual peripheral module.

The data memory area also contains the data EEPROM memory. This memory is not directly mapped into the data memory, but is indirectly mapped. That is, an indirect address pointer specifies the address of the data EEPROM memory to read/write. The 64 bytes of data EEPROM memory have the address range 0h-3Fh. More details on the EEPROM memory can be found in Section 4.

3.1. Program Memory Organization

The PIC16FXX has a 13-bit program counter capable of addressing an 8K x 14 program memory space. For the PIC16F84A, the first 1K x 14 (0000h-03FFh) are physically

implemented (**Figure 3.1**). Accessing a location above the physically implemented address will cause a wrap around. For example, for locations 20h, 420h, 820h, C20h, 1020h, 1420h, 1820h, and 1C20h, the instruction will be the same.

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

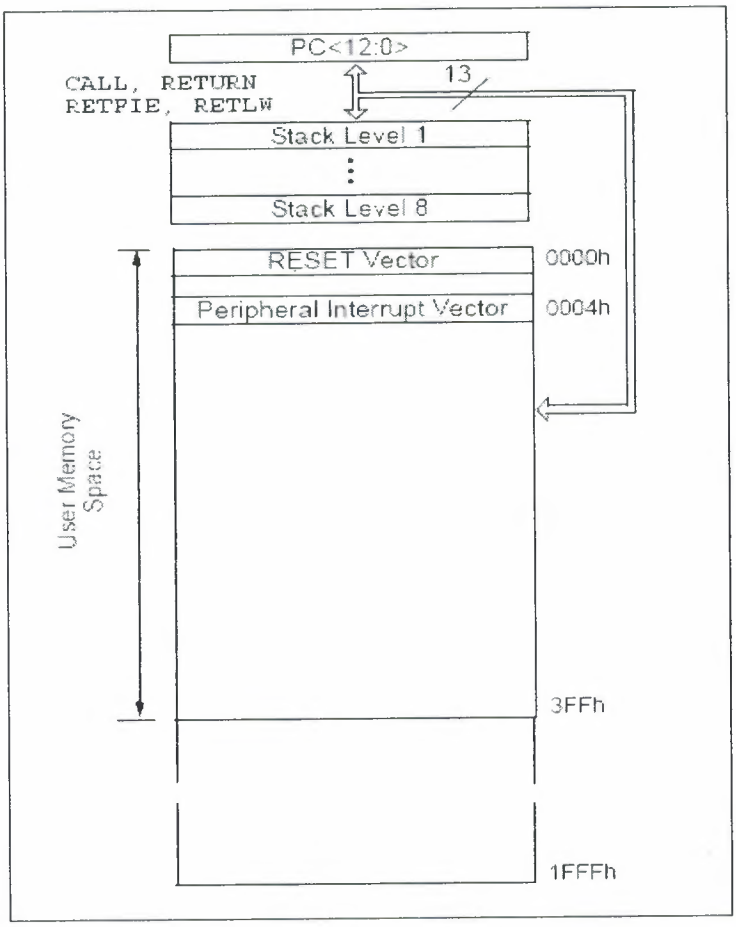


Figure 3.1 Program memory map and stack – PIC16F84A

3.2. Data Memory Organization

The data memory is partitioned into two areas. The first is the Special Function Registers (SFR) area, while the second is the General Purpose Registers (GPR) area. The SFRs control the operation of the device.

Portions of data memory are banked. This is for both the SFR area and the GPR area. The GPR area is banked to allow greater than 116 bytes of general purpose RAM. The banked areas of the SFR are for the registers that control the peripheral functions. Banking requires the use of control bits for bank selection. These control bits are located in the STATUS Register. **Figure 3.2** shows the data memory map organization.

Instructions MOVWF and MOVF can move values from the W register to any location in the register file ("F"), and vice-versa.

The entire data memory can be accessed either directly using the absolute address of each register file or indirectly through the File Select Register (FSR) (Section 3.5). Indirect addressing uses the present value of the RP0 bit for access into the banked areas of data memory.

Data memory is partitioned into two banks which contain the general purpose registers and the special function registers. Bank 0 is selected by clearing the RP0 bit (STATUS<5>). Setting the RP0 bit selects Bank 1. Each Bank extends up to 7Fh (128 bytes). The first twelve locations of each Bank are reserved for the Special Function Registers. The remainder are General Purpose Registers, implemented as static RAM.

3.2.1. General Purpose Register File

Each General Purpose Register (GPR) is 8-bits wide and is accessed either directly or indirectly through the FSR (Section 3.5).

The GPR addresses in Bank 1 are mapped to addresses in Bank 0. As an example, addressing location 0Ch or 8Ch will access the same GPR.

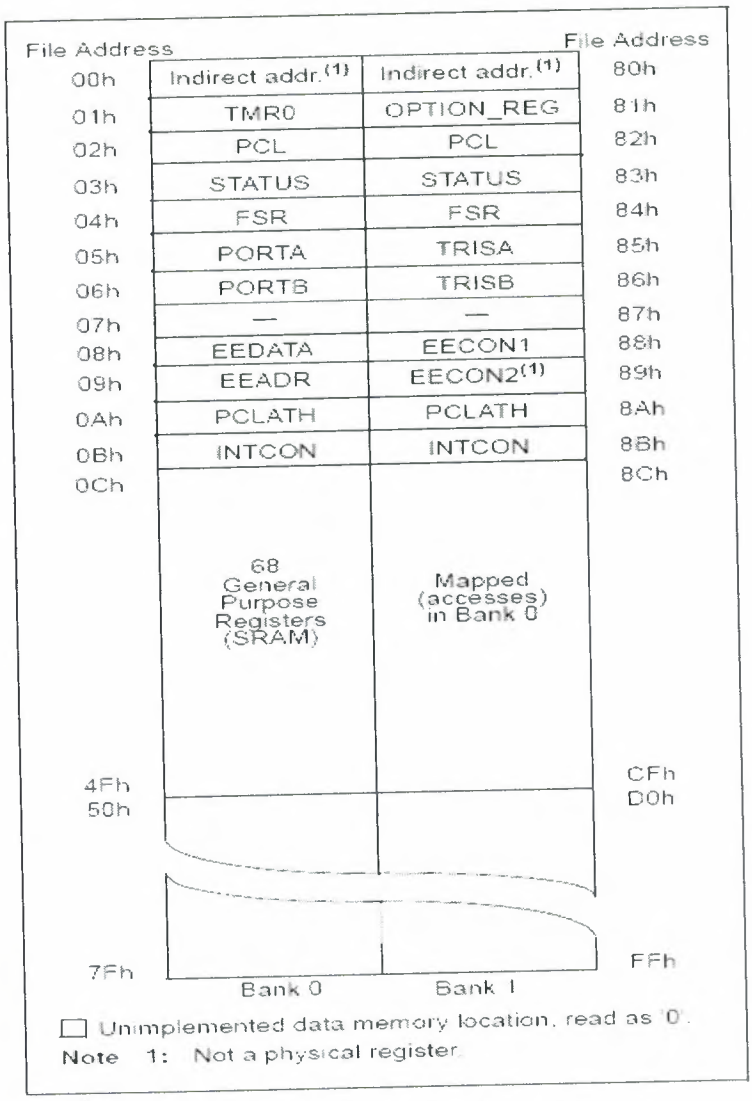


Figure 3.2 Register file map PIC16F84A

3.3. Special Function Registers

The special function registers can be classified into two sets, core and peripheral. Those associated with the The Special Function Registers (**Figure 3.2** and core functions are described in this section. Those **Table 3.1**) are used by the CPU and Peripheral related to the operation of the peripheral features are functions to control the device operation. These described in the section for that specific feature. registers are static RAM.

The special function registers can be classified into two sets, core and peripheral. Those associates with the core functions are described in this section. Those related to the operation of the peripheral features are described in the section fort hat specific feature.

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-on RESET	Details on page
Bank 0											
00h	INDF	Uses contents of FSR to address Data Memory (not a physical register)								---- ----	11
01h	TMR0	8-bit Real-Time Clock/Counter								xxxx xxxx	20
02h	PCL	Low Order 8 bits of the Program Counter (PC)								0000 0000	11
03h	STATUS(2)	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	8
04h	FSR	Indirect Data Memory Address Pointer 0								xxxx xxxx	11
05h	PORTA(4)	—	—	—	RA4/T0CKI	RA3	RA2	RA1	RA0	---x xxxx	16
06h	PORTB(5)	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0/INT	xxxx xxxx	18
07h	—	Unimplemented location, read as '0'								—	—
08h	EEDATA	EEPROM Data Register								xxxx xxxx	13,14
09h	EEADR	EEPROM Address Register								xxxx xxxx	13,14
0Ah	PCLATH	—	—	—	Write Buffer for upper 5 bits of the PC(1)					---0 0000	11
0Bh	INTCON	GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	10
Bank 1											
80h	INDF	Uses Contents of FSR to address Data Memory (not a physical register)								---- ----	11
81h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	9
82h	PCL	Low order 8 bits of Program Counter (PC)								0000 0000	11
83h	STATUS (2)	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	8
84h	FSR	Indirect data memory address pointer 0								xxxx xxxx	11
85h	TRISA	—	—	—	PORTA Data Direction Register					---1 1111	16
86h	TRISB	PORTB Data Direction Register								1111 1111	18
87h	—	Unimplemented location, read as '0'								—	—
88h	EECON1	—	—	—	EEIF	WRERR	WREN	WR	RD	---0 x000	13
89h	EECON2	EEPROM Control Register 2 (not a physical register)								---- ----	14
0Ah	PCLATH	—	—	—	Write buffer for upper 5 bits of the PC(1)					---0 0000	11
0Bh	INTCON	GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	10

Table 3.1 Special function register file summary

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

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a slave register for PC<12:8>. The contents of PCLATH can be transferred to the upper byte of the program counter, but the contents of PC<12:8> are never transferred to PCLATH.

2: The TO and PD status bits in the STATUS register are not affected by a MCLR Reset.

3: Other (non power-up) RESETS include: external RESET through MCLR and

the Watchdog Timer Reset.

4: On any device RESET, these pins are configured as inputs.

5: This is the value that will be in the port output latch.

3.3.1. Status Register

The STATUS register contains the arithmetic status of the ALU, the RESET status and the bank select bit for data memory.

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

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

Only the BCF, BSF, SWAPF and MOVWF instructions should be used to alter the STATUS register (**Table 7-2**), because these instructions do not affect any status bit.

Note 1: The IRP and RP1 bits (STATUS<7:6>) are not used by the PIC16F84A and should be programmed as cleared. Use of these bits as general purpose R/W bits is NOT recommended, since this may affect upward compatibility with future products.

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

3: When the STATUS register is the destination for an instruction that affects the Z, DC or C bits, then the write to these three bits is disabled. The specified bit(s) will be updated according to device logic.

REGISTER 3-1: STATUS REGISTER (ADDRESS 03h, 83h)

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C
bit 7							bit 0

bit 7-6 Unimplemented: Maintain as '0'

bit 5 RP0: Register Bank Select bits (used for direct addressing)

01 = Bank 1 (80h - FFh)

00 = Bank 0 (00h - 7Fh)

bit 4 TO: Time-out bit

1= After power-up, CLRWDT instruction, or SLEEP instruction

0= A WDT time-out occurred

bit 3 PD: Power-down bit

1= After power-up or by the CLRWDT instruction

0= By execution of the SLEEP instruction

bit 2 Z: Zero bit

1= The result of an arithmetic or logic operation is zero

0= The result of an arithmetic or logic operation is not zero

bit 1 DC: Digit carry/borrow bit (ADDWF, ADDLW,SUBLW,SUBWF instructions) (for borrow, the polarity is reversed)

1= A carry-out from the 4th low order bit of the result occurred

0= No carry-out from the 4th low order bit of the result

bit 0 C: Carry/borrow bit (ADDWF, ADDLW,SUBLW,SUBWF instructions) (for borrow, the polarity is reversed)

1= A carry-out from the Most Significant bit of the result occurred

0= No carry-out from the Most Significant bit of the result occurred

Note: A subtraction is executed by adding the two's complement of the second operand. For rotate (RRF, RLF) instructions, this bit is loaded with either the high or low order bit of the source register.

Legend: R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

- n = Value at PORT

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

3.3.2. Option Register

The OPTION register is a readable and writable register which contains various control bits to configure the TMR0/WDT prescaler, the external INT interrupt, TMR0, and the weak pull-ups on PORTB.

Note: When the prescaler is assigned to the WDT (PSA = '1'), TMR0 has a 1:1 prescaler assignment.

REGISTER 3-2: OPTION REGISTER (ADDRESS 81h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

bit 7 RBPU: PORTB Pull-up Enable bit

1= PORTB pull-ups are disabled

0= PORTB pull-ups are enabled by individual port latch values

bit 6 INTEDG: Interrupt Edge Select bit

1= Interrupt on rising edge of RB0/INT pin

0= Interrupt on falling edge of RB0/INT pin

bit 5 T0CS: TMR0 Clock Source Select bit

1= Transition on RA4/T0CKI pin

0= Internal instruction cycle clock (CLKOUT)

bit 4 T0SE: TMR0 Source Edge Select bit

1= Increment on high-to-low transition on RA4/T0CKI pin

0= Increment on low-to-high transition on RA4/T0CKI pin

bit 3 PSA: Prescaler Assignment bit

1= Prescaler is assigned to the WDT

0= Prescaler is assigned to the Timer0 module

bit 2-0 PS2:PS0: Prescaler Rate Select bits

Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

Legend: R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

- n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

3.3.3. Intcon Register

The INTCON register is a readable and writable register that contains the various enable bits for all interrupt sources.

Note: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the global enable bit, GIE (INTCON<7>).

REGISTER 3-3: INTCON REGISTER (ADDRESS 0Bh, 8Bh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF
bit 7							bit 0

bit 7 GIE: Global Interrupt Enable bit

1 = Enables all unmasked interrupts

0 = Disables all interrupts

bit 6 EEIE: EE Write Complete Interrupt Enable bit

1 = Enables the EE Write Complete interrupts

0= Disables the EE Write Complete interrupt

bit 5 T0IE: TMR0 Overflow Interrupt Enable bit

1 = Enables the TMR0 interrupt

0 = Disables the TMR0 interrupt

bit 4 INTE: RB0/INT External Interrupt Enable bit

1 = Enables the RB0/INT external interrupt

0 = Disables the RB0/INT external interrupt

bit 3 RBIE: RB Port Change Interrupt Enable bit

1 = Enables the RB port change interrupt

0 = Disables the RB port change interrupt

bit 2 T0IF: TMR0 Overflow Interrupt Flag bit

1 = TMR0 register has overflowed (must be cleared in software)

0 = TMR0 register did not overflow

bit 1 INTF: RB0/INT External Interrupt Flag bit

1 = The RB0/INT external interrupt occurred (must be cleared in software)

0 = The RB0/INT external interrupt did not occur

bit 0 RBIF: RB Port Change Interrupt Flag bit

1 = At least one of the RB7:RB4 pins changed state (must be cleared in software)

0 = None of the RB7:RB4 pins have changed state

Legend: R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

-n = Value at POR

'1' = Bit is set

'0' = Bit is cleared x = Bit is unknown

3.4. PCL and PCLATH

The program counter (PC) specifies the address of the instruction to fetch for execution. The PC is 13 bits wide. The low byte is called the PCL register. This register is readable and writable. The high byte is called the PCH register. This register contains the PC<12:8> bits and is not directly readable or writable. If the program counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP. All updates to the PCH register go through the PCLATH register.

3.4.1. STACK

The stack allows a combination of up to 8 program calls and interrupts to occur. The stack contains the return address from this branch in program execution.

Mid-range devices have an 8 level deep x 13-bit wide hardware stack. The stack space is not part of either program or data space and the stack pointer is not readable or writable. The PC is PUSHed onto the stack when a CALL instruction is executed or an interrupt causes a branch. The stack is POPed in the event of a RETURN, RETLW or a RETFIE instruction execution. PCLATH is not modified when the stack is PUSHed or POPed.

After the stack has been PUSHed eight times, the ninth push overwrites the value that was stored from the first push. The tenth push overwrites the second push (and so on).

3.5. Indirect Addressing; INDF and FSR Registers

The INDF register is not a physical register. Addressing INDF actually addresses the register whose address is contained in the FSR register (FSR is a pointer). This is indirect addressing.

EXAMPLE 3-1: INDIRECT ADDRESSING

- Register file 05 contains the value 10h
- Register file 06 contains the value 0Ah
- Load the value 05 into the FSR register
- A read of the INDF register will return the value of 10h
- Increment the value of the FSR register by one (FSR = 06)
- A read of the INDF register now will return the value of 0Ah.

Reading INDF itself indirectly (FSR = 0) will produce 00h. Writing to the INDF register indirectly results in a no-operation (although STATUS bits may be affected).

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

EXAMPLE 3-2: HOW TO CLEAR RAM USING INDIRECT ADDRESSING

```
        movlw   0x20    ;initialize pointer
        movwf   FSR     ;to RAM
NEXT    clrf    INDF    ;clear INDF register
        İncf    FSR     ;inc pointer
        btfss   FSR,4   ;all done?
        goto    NEXT    ;NO, clear next
CONTINUE
        :               ;YES, continue
```

An effective 9-bit address is obtained by concatenating the 8-bit FSR register and the IRP bit (STATUS<7>), as shown in **Figure 3.3**. However, IRP is not used in the PIC16F84A.

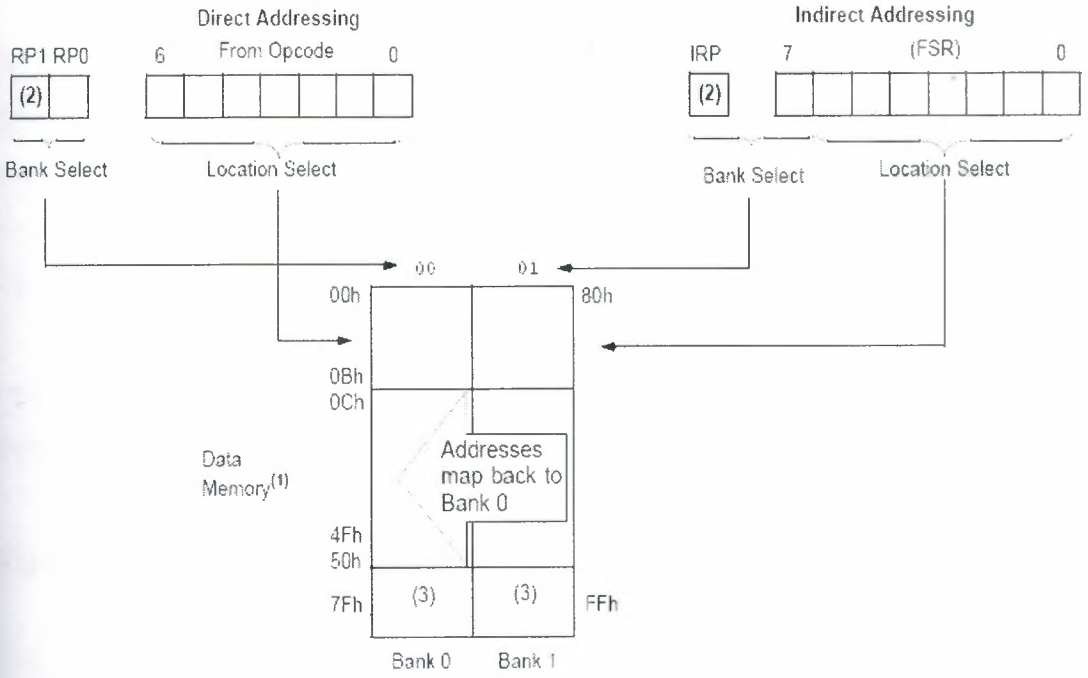


Figure 3.3 Direct/Indirect addressing

Note 1: For memory map detail, see **Figure 3-2.**

2: Maintain as clear for upward compatibility with future products.

3: Not implemented.

4. DATA EEPROM MEMORY

The EEPROM data memory is readable and writable during normal operation (full VDD range). This memory is not directly mapped in the register file space. Instead it is indirectly addressed through the Special Function Registers. There are four SFRs used to read and write this memory. These registers are:

- EECON1
- EECON2 (not a physically implemented register)
- EEDATA
- EEADR

EEDATA holds the 8-bit data for read/write, and EEADR holds the address of the EEPROM location being accessed. PIC16F84A devices have 64 bytes of data EEPROM with an address range from 0h to 3Fh.

The EEPROM data memory allows byte read and write. A byte write automatically erases the location and writes the new data (erase before write). The EEPROM data memory is rated for high erase/write cycles. The write time is controlled by an on-chip timer. The write-time will vary with voltage and temperature as well as from chip to chip. Please refer to AC specifications for exact limits.

When the device is code protected, the CPU may continue to read and write the data EEPROM memory. The device programmer can no longer access this memory.

REGISTER 3-1: EECON1 REGISTER (ADDRESS 88h)

U-0	U-0	U-0	R/W-0	R/W-x	R/W-0	R/S-0	R/S-0
—	—	—	EEIF	WRERR	WREN	WR	RD
bit 7							bit 0

bit 7-5 Unimplemented: Read as '0'

bit 4 EEIF: EEPROM Write Operation Interrupt Flag bit

1 = The write operation completed (must be cleared in software)

0 = The write operation is not complete or has not been started

bit 3 WRERR: EEPROM Error Flag bit

1= A write operation is prematurely terminated

(any MCLR Reset or any WDT Reset during normal operation)

0= The write operation completed

bit 2 WREN: EEPROM Write Enable bit

1 = Allows write cycles

0 = Inhibits write to the EEPROM

bit 1 WR: Write Control bit

1 = Initiates a write cycle. The bit is cleared by hardware once write is complete. The WR bit can only be set (not cleared) in software.

0 = Write cycle to the EEPROM is complete

bit 0 RD: Read Control bit

1 = Initiates an EEPROM read RD is cleared in hardware. The RD bit can only be set (not cleared) in software.

0 = Does not initiate an EEPROM read

Legend: R = Readable bit

W = Writable bit

U = Unimplemented bit, read as '0'

- n = Value at POR

'1' = Bit is set

'0' = Bit is cleared

x = Bit is unknown

4.1. Reading the EEPROM Data Memory

To read a data memory location, the user must write the address to the EEADR register and then set control bit RD (EECON1<0>). The data is available, in the very next cycle, in the EEDATA register; therefore, it can be read in the next instruction. EEDATA will hold this value until another read or until it is written to by the user (during a write operation).

EXAMPLE 4-1: DATA EEPROM READ

```
BCF     STATUS,   RP0   ; Bank 0
MOVLW   CONFIG_ADDR     ;
MOVWF   EEADR           ; Address to read
BSF     STATUS,   RP0   ; Bank 1
BSF     EECON1,   RD    ; EE Read
BCF     STATUS,   RP0   ; Bank 0
MOVF    EEDATA,   W     ; W = EEDATA
```

4.2. Writing to the EEPROM Data Memory

To write an EEPROM data location, the user must first write the address to the EEADR register and the data to the EEDATA register. Then the user must follow a specific sequence to initiate the write for each byte.

EXAMPLE 4-2: DATA EEPROM WRITE

```
BSF     STATUS, RP0     ; Bank 1
BCF     INTCON, GIE     ; Disable INTs.
BSF     EECON1, WREN    ; Enable Write
MOVLW   55h             ;
MOVWF   EECON2          ; Write 55h
MOVLW   AAh             ;
MOVWF   EECON2          ; Write AAh
BSF     EECON1,WR       ; Set WR bit
                        ; begin write
BSF     INTCON, GIE     ; Enable INTs.
```

The write will not initiate if the above sequence is not exactly followed (write 55h to EECON2, write AAh to EECON2, then set WR bit) for each byte. We strongly recommend that interrupts be disabled during this code segment.

Additionally, the WREN bit in EECON1 must be set to enable write. This mechanism prevents accidental writes to data EEPROM due to errant (unexpected) code execution (i.e., lost programs). The user should keep the WREN bit clear at all times, except when updating EEPROM. The WREN bit is not cleared by hardware.

After a write sequence has been initiated, clearing the WREN bit will not affect this write cycle. The WR bit will be inhibited from being set unless the WREN bit is set.

At the completion of the write cycle, the WR bit is cleared in hardware and the EE Write Complete Interrupt Flag bit (EEIF) is set. The user can either enable this interrupt or poll this bit. EEIF must be cleared by software.

4.3. Write Verify

Depending on the application, good programming practice may dictate that the value written to the Data EEPROM should be verified (Example 4-3) to the desired value to

be written. This should be used in applications where an EEPROM bit will be stressed near the specification limit.

Generally, the EEPROM write failure will be a bit which was written as a '0', but reads back as a '1' (due to leakage off the bit).

EXAMPLE 4-3: WRITE VERIFY

```
        BCF     STATUS,RP0      ; Bank 0
        :                       ; Any code
        :                       ; can go here
        MOVF    EEDATA,W        ; Must be in Bank 0
        BSF     STATUS,RP0      ; Bank 1
READ
        BSF     EECON1, RD      ; YES, Read the
                                ; value written
        BCF     STATUS, RP0     ; Bank 0
                                ;
                                ; Is the value written
                                ; (in W reg) and
                                ; read (in EEDATA)
                                ; the same?
                                ;
        SUBWF EEDATA, W         ;
        BTFSS   STATUS, Z       ; Is difference 0?
        GOTO    WRITE_ERR       ; NO, Write error
```

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-on Reset	Value on all other RESETS
08h	EEDATA	EEPROM Data Register								xxxx xxxx	uuuu uuuu
09h	EEADR	EEPROM Address Register								xxxx xxxx	uuuu uuuu
88h	EECON1	—	—	—	EEIF	WRERR	WREN	WR	RD	---0 x000	---0 q000
89h	EECON2	EEPROM Control Register 2								---- ----	---- ----

Table 4.1 Registers/Bits associated with data EEPROM

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

Shaded cells are not used by data EEPROM.

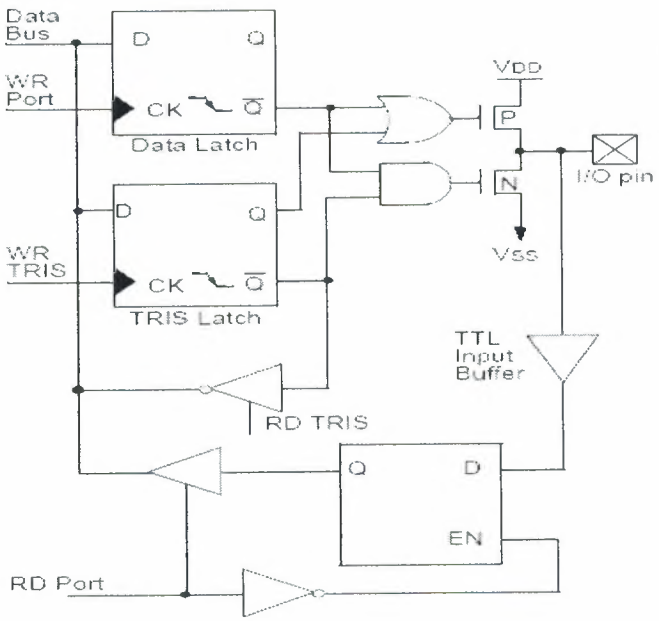
5. I/O PORTS

Some pins for these I/O ports are multiplexed with an alternate function for the peripheral features on the device. In general, when a peripheral is enabled, that pin may not be used as a general purpose I/O pin.

5.1. PORTA and TRISA Registers

PORTA is a 5-bit wide, bi-directional port. The corresponding data direction register is TRISA. Setting a TRISA bit (= 1) will make the corresponding PORTA pin an input (i.e., put the corresponding output driver in a Hi-Impedance mode). Clearing a TRISA bit (= 0) will make the corresponding PORTA pin an output (i.e., put the contents of the output latch on the selected pin).

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



Fifure 5.1 Block diagram of pins RA3:RA0

Note: I/O pins have protection diodes to VDD and VSS.

Reading the PORTA register reads the status of the pins, whereas writing to it will write to the port latch. All write operations are read-modify-write operations. Therefore, a write to a port implies that the port pins are read. This value is modified and then written to the port data latch.

Pin RA4 is multiplexed with the Timer0 module clock input to become the RA4/T0CKI pin. The RA4/T0CKI pin is a Schmitt Trigger input and an open drain output. All other RA port pins have TTL input levels and full CMOS output drivers.

EXAMPLE 5-1: INITIALIZING PORTA

```
BCF     STATUS, RP0 ;
CLRF    PORTA       ; Initialize PORTA by
                    ; clearing output
                    ; data latches
BSF     STATUS, RP0 ; Select Bank 1
MOVLW   0x0F        ; Value used to
                    ; initialize data
                    ; direction
MOVWF   TRISA       ; Set RA<3:0> as inputs
                    ; RA4 as output
                    ; TRISA<7:5> are always
                    ; read as '0'.
```

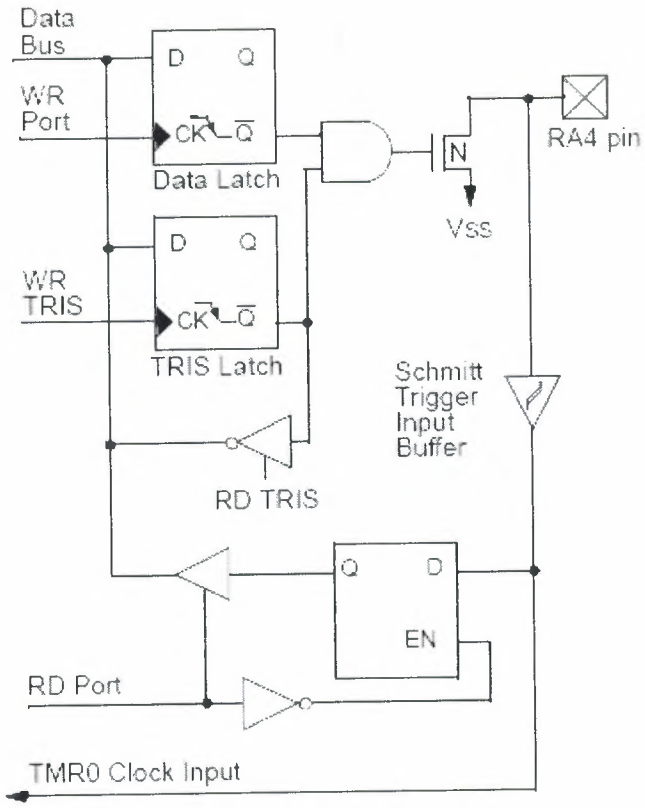


Figure 5.2 Block diagram of pin RA4

Note: I/O pins have protection diodes to VDD and VSS.

Name	Bit0	Buffer Type	Function
RA0	bit0	TTL	Input/output
RA1	bit1	TTL	Input/output
RA2	bit2	TTL	Input/output
RA3	bit3	TTL	Input/output
RA4/T0CKI	bit4	ST	Input/output or external clock input for TMR0. Output is open drain type.

Table 5.1 PORTA functions

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

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-on Reset	Value on all other RESETS
05h	PORTA	—	—	—	RA4/T0CKI	RA3	RA2	RA1	RA0	---x xxxx	---u uuuu
85h	TRISA	—	—	—	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	---1 1111	---1 1111

Table 5.2 Summary of registers associated with PORTA

Legend: x = unknown, u = unchanged, -= unimplemented, read as '0'. Shaded cells are unimplemented, read as '0'.

5.2. PORTB and TRISB Registers

PORTB is an 8-bit wide, bi-directional port. The corresponding data direction register is TRISB. Setting a TRISB bit (= 1) will make the corresponding PORTB pin an input (i.e., put the corresponding output driver in a Hi-Impedance mode). Clearing a TRISB bit (= 0) will make the corresponding PORTB pin an output (i.e., put the contents of the output latch on the selected pin).

EXAMPLE 5-2: INITIALIZING PORTB

```
BCF     STATUS, RP0     ;
CLRF    PORTB           ; Initialize PORTB by
                        ; clearing output
                        ; data latches
BSF     STATUS, RP0     ; Select Bank 1
MOVLW   0xCF            ; Value used to
                        ; initialize data
                        ; direction
MOVWF   TRISB           ; Set RB<3:0> as inputs
                        ; RB<5:4> as outputs
                        ; RB<7:6> as inputs
```

Each of the PORTB pins has a weak internal pull-up. A single control bit can turn on all the pull-ups. This is performed by clearing bit RBPU (OPTION<7>). The weak pull-up is automatically turned off when the port pin is configured as an output. The pull-ups are disabled on a Power-on Reset.

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

"mismatch" outputs of RB7:RB4 are OR'ed together to generate the RB Port Change Interrupt with flag bit RBIF (INTCON<0>).

This interrupt can wake the device from SLEEP. The user, in the Interrupt Service Routine, can clear the interrupt in the following manner:

a) Any read or write of PORTB. This will end the mismatch condition.
b) Clear flag bit RBIF.

A mismatch condition will continue to set flag bit RBIF. Reading PORTB will end the mismatch condition and allow flag bit RBIF to be cleared.

The interrupt-on-change feature is recommended for wake-up on key depression operation and operations where PORTB is only used for the interrupt-on-change feature. Polling of PORTB is not recommended while using the interrupt-on-change feature.

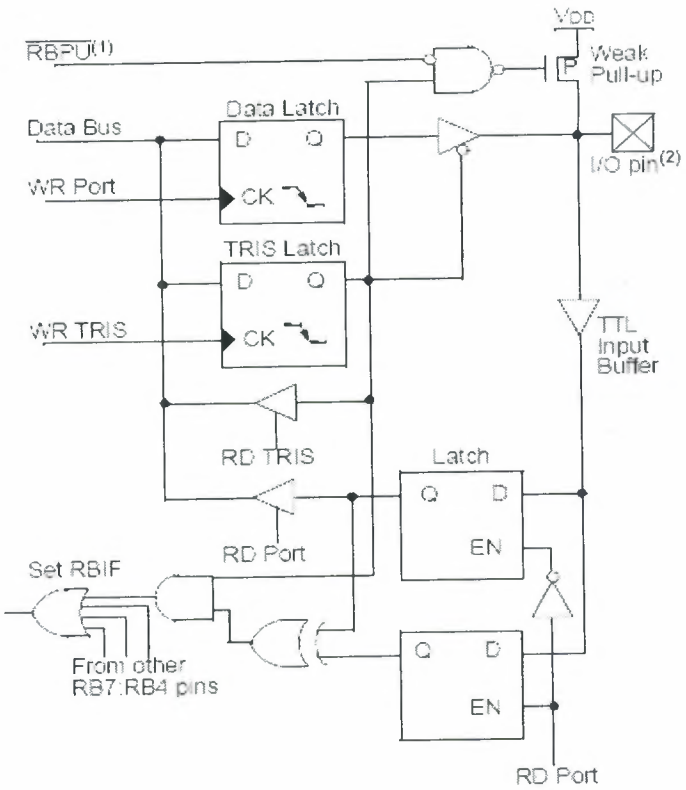


Figure 5.3 Block diagram of pins RB7:RB4

Note 1: TRISB = '1' enables weak pull-up
(if RBPU = '0' in the OPTION_REG register).
2: I/O pins have diode protection to VDD and VSS.

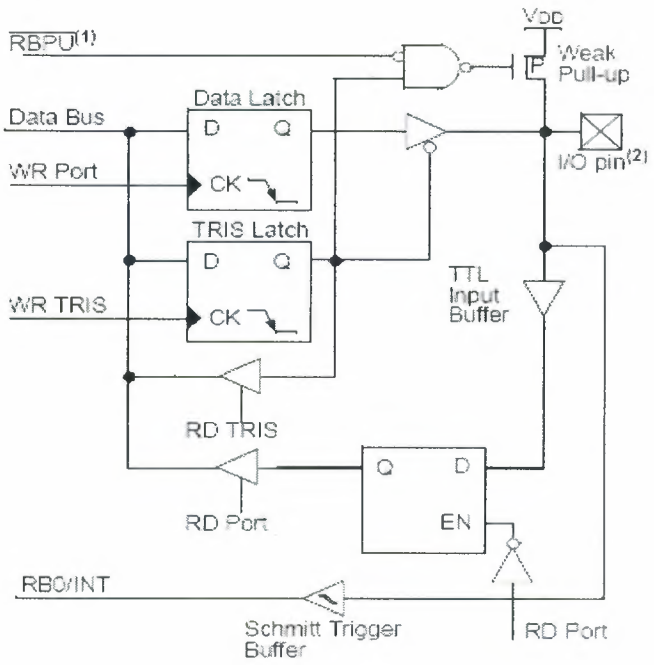


Figure 5.4 Block diagram of pins RB3:RB0

Note 1: TRISB = '1' enables weak pull-up
(if RBPU = '0' in the OPTION_REG register).
2: I/O pins have diode protection to VDD and VSS.

Name	Bit	Buffer Type	I/O Consistency Function
RB0/INT	bit0	TTL/ST(1)	Input/output pin or external interrupt input. Internal software programmable weak pull-up.
RB1	bit1	TTL	Input/output pin. Internal software programmable weak pull-up.
RB2	bit2	TTL	Input/output pin. Internal software programmable weak pull-up.
RB3	bit3	TTL	Input/output pin. Internal software programmable weak pull-up.
RB4	bit4	TTL	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up.
RB5	bit5	TTL	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up.
RB6	bit6	TTL/ST(2)	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up. Serial programming clock.
RB7	bit7	TTL/ST(2)	Input/output pin (with interrupt-on-change). Internal software programmable weak pull-up. Serial programming data.

Table 5.3 PORTB functions

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

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.

2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-on Reset	Value on all other RESETS
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0/INT	xxxx xxxx	uuuu uuuu
86h	TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
81h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
0Bh,8Bh	INTCON	GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u

Table 5.4 Summary of registers associated with PORTB

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

6. TIMER0 MODULE

The Timer0 module timer/counter has the following features:

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

Figure 6.1 is a simplified block diagram of the Timer0 module.

6.1. Timer0 Operation

Timer0 can operate as a timer or as a counter.

Timer mode is selected by clearing bit T0CS (OPTION_REG<5>). In Timer mode, the Timer0 module will increment every instruction cycle (without prescaler). If the TMR0 register is written, the increment is inhibited for the following two instruction cycles. The user can work around this by writing an adjusted value to the TMR0 register.

Counter mode is selected by setting bit T0CS (OPTION_REG<5>). In Counter mode, Timer0 will increment, either on every rising or falling edge of pin RA4/T0CKI. The incrementing edge is determined by the Timer0 Source Edge Select bit, T0SE (OPTION_REG<4>). Clearing bit T0SE selects the rising edge. Restrictions on the external clock input are discussed below.

When an external clock input is used for Timer0, it must meet certain requirements. The requirements ensure the external clock can be synchronized with the internal phase clock (TOSC). Also, there is a delay in the actual incrementing of Timer0 after synchronization.

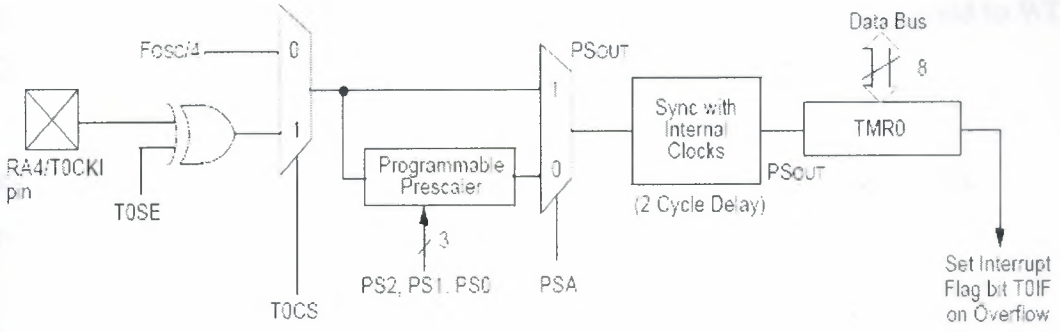


Figure 6.1 TIMER0 block diagram

Note 1: T0CS, T0SE, PSA, PS2:PS0 (OPTION_REG<5:0>).

2: The prescaler is shared with Watchdog Timer (refer to **Figure 6.2** for detailed block diagram).

6.2. Prescaler

An 8-bit counter is available as a prescaler for the Timer0 module, or as a postscaler for the Watchdog Timer, respectively (**Figure 6.2**). For simplicity, this counter is being referred to as "prescaler" throughout this data sheet. Note that there is only one

prescaler available which is mutually exclusively shared between the Timer0 module and the Watchdog Timer. Thus, a prescaler assignment for the Timer0 module means that there is no prescaler for the Watchdog Timer, and vice-versa.

The prescaler is not readable or writable.

The PSA and PS2:PS0 bits (OPTION_REG<3:0>) determine the prescaler assignment and prescale ratio.

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

Setting bit PSA will assign the prescaler to the Watchdog Timer (WDT). When the prescaler is assigned to the WDT, prescale values of 1:1, 1:2, ..., 1:128 are selectable.

When assigned to the Timer0 module, all instructions writing to the TMR0 register (e.g., CLRF 1, MOVWF 1, BSF 1,etc.) will clear the prescaler. When assigned to WDT, a CLRWDT instruction will clear the prescaler along with the WDT.

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

6.2.1. Switching Prescaler Assignment

The prescaler assignment is fully under software control (i.e., it can be changed "on the fly" during program execution).

Note: To avoid an unintended device RESET, a specific instruction sequence must be executed when changing the prescaler assignment from Timer0 to the WDT. This sequence must be followed even if the WDT is disabled.

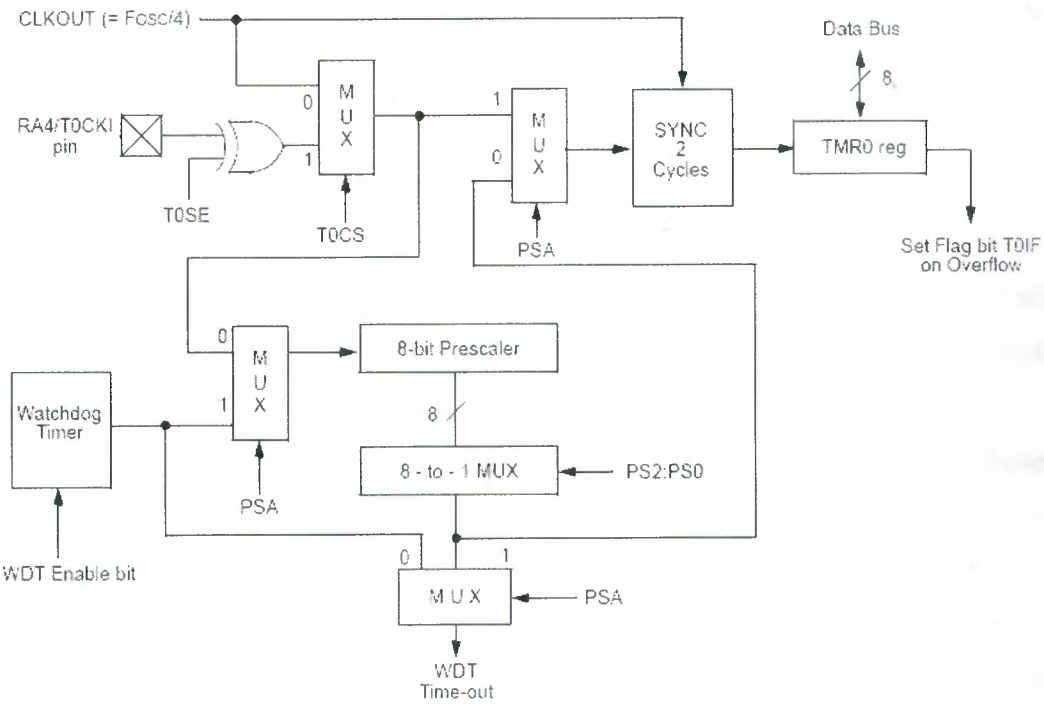


Figure 6.2 Block diagram of the TIMER0/WDT PRESCALER

Note: T0CS, T0SE, PSA, PS2:PS0 are (OPTION_REG<5:0>).

6.3. Timer0 Interrupt

The TMR0 interrupt is generated when the TMR0 register overflows from FFh to 00h. This overflow sets bit T0IF (INTCON<2>). The interrupt can be masked by clearing bit T0IE (INTCON<5>). Bit T0IF must be cleared in software by the Timer0 module Interrupt Service Routine before re-enabling this interrupt. The TMR0 interrupt cannot awaken the processor from SLEEP since the timer is shut-off during SLEEP.

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR, BOR	Value on all other RESETS
01h	TMR0	Timer0 Module Register								xxxx xxxx	uuuu uuuu
0Bh,8Bh	INTCON	GIE	EEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
81h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
85h	TRISA	—	—	—	PORTA Data Direction Register					---1 1111	---1 1111

Table 6.1 Registers associated with TIMER0

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

7. SPECIAL FEATURES OF THE CPU

What sets a microcontroller apart from other processors are special circuits to deal with the needs of real time applications. The PIC16F84A has a host of such features intended to maximize system reliability, minimize cost through elimination of external components, provide power saving operating modes and offer code protection. These features are:

- OSC Selection
- RESET
    - -Power-on Reset (POR)
    - -Power-up Timer (PWRT)
    - -Oscillator Start-up Timer (OST)
- Interrupts
- Watchdog Timer (WDT)
- SLEEP
- Code Protection
- ID Locations
- In-Circuit Serial Programming™ (ICSP™)

The PIC16F84A has a Watchdog Timer which can be shut-off only through configuration bits. It runs off its own RC oscillator for added reliability. There are two timers that offer necessary delays on power-up. One is the Oscillator Start-up Timer (OST), intended to keep the chip in RESET until the crystal oscillator is stable. The other is the Power-up Timer (PWRT), which provides a fixed delay of 72 ms (nominal) on power-up only. This design keeps the device in RESET while the power supply stabilizes. With these two timers on-chip, most applications need no external RESET circuitry.

SLEEP mode offers a very low current power-down mode. The user can wake-up from SLEEP through external RESET, Watchdog Timer Time-out or through an interrupt. Several oscillator options are provided to allow the part to fit the application. The RC oscillator option saves system cost while the LP crystal option saves power. A set of configuration bits are used to select the various options.

7.1. Configuration Bits

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

Address 2007h is beyond the user program memory space and it belongs to the special test/configuration memory space (2000h - 3FFFh). This space can only be accessed during programming.

REGISTER 7-1: PIC16F84A CONFIGURATION WORD

R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u	R/P-u
CP	CP	CP	CP	CP	CP	CP	CP	CP	CP	$\overline{\text{PWRTE}}$	WDTE	FOSC1	FOSC0
bit13													bit0

bit 13-4 CP: Code Protection bit

1 = Code protection disabled

0 = All program memory is code protected

bit 3 PWRTE: Power-up Timer Enable bit

1 = Power-up Timer is disabled

0 = Power-up Timer is enabled

bit 2 WDTE: Watchdog Timer Enable bit

1 = WDT enabled

0 = WDT disabled

bit 1-0 FOSC1:FOSC0: Oscillator Selection bits

11 = RC oscillator

10 = HS oscillator

01 = XT oscillator

00 = LP oscillator

7.2. Oscillator Configurations

7.2.1. Oscillator Types

The PIC16F84A can be operated in four different oscillator modes. The user can program two configuration bits (FOSC1 and FOSC0) to select one of these four modes:

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

7.2.2. Crystal Oscillator/Ceramic Resonators

In XT, LP, or HS modes, a crystal or ceramic resonator is connected to the OSC1/CLKIN and OSC2/CLKOUT pins to establish oscillation (**Figure 7.1**).

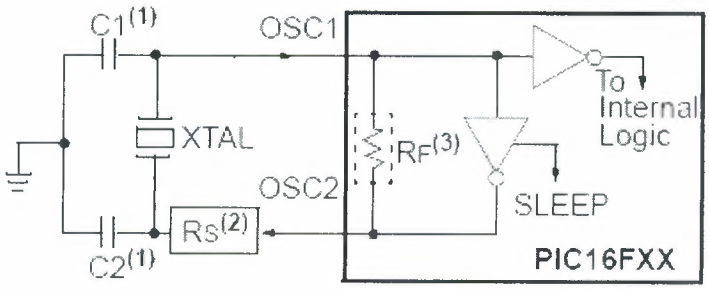


Figure 7.1 Crystal/Ceramic resonator operation (HS, XT OR LP OSC configuration)

Note 1: See **Table 7.1** for recommended values of C1 and C2.

2: A series resistor (RS) may be required for AT strip cut crystals.

The PIC16F84A oscillator design requires the use of a parallel cut crystal. Use of a series cut crystal may give a frequency out of the crystal manufacturers specifications. When in XT, LP, or HS modes, the device can have an external clock source to drive the OSC1/CLKIN pin (**Figure 7.2**).

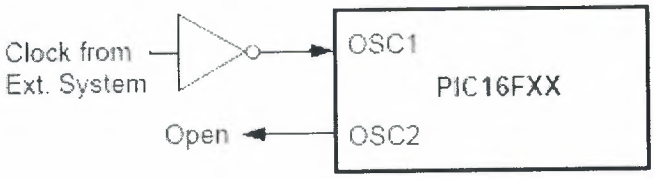


Figure 7.2 External clock input operation (HS, XT OR LP OSC configuration)

Ranges Tested:			
Mode	**Freq**	**OSC1/C1**	**OSC2/C2**
XT	455 kHz	47 - 100 pF	47 - 100 pF
	2.0 MHz	15 - 33 pF	15 - 33 pF
	4.0 MHz	15 - 33 pF	15 - 33 pF
HS	8.0 MHz	15 - 33 pF	15 - 33 pF
	10.0 MHz	15 - 33 pF	15 - 33 pF

Table 7.1 Capacitor selection for ceramic resonators

Note: Recommended values of C1 and C2 are identical to the ranges tested in this table. Higher capacitance increases the stability of the oscillator, but also increases the start-up time. These values are for design guidance only. Since each resonator has its own characteristics, the user should consult the resonator manufacturer for the appropriate values of external components.

Note: When using resonators with frequencies above 3.5 MHz, the use of HS mode rather than XT mode, is recommended. HS mode may be used at any VDD for which the controller is rated.

Mode	Freq	OSC1/C1	OSC2/C2
LP	32 kHz 200 kHz	68 - 100 pF 15 - 33 pF	68 - 100 pF 15 - 33 pF
XT	100 kHz 2 MHz 4 MHz	100 - 150 pF 15 - 33 pF 15 - 33 pF	100 - 150 pF 15 - 33 pF 15 - 33 pF
HS	4 MHz 20 MHz	15 - 33 pF 15 - 33 pF	15 - 33 pF 15 - 33 pF

Table 7.2 Capacitor selection for crystal oscillator

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

For VDD > 4.5V, C1 = C2 » 30 pF is recommended.

7.2.3. 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) values, capacitor (CEXT) values, and the operating temperature. In addition to this, the oscillator frequency will vary from unit to unit due to normal process parameter variation. Furthermore, the difference in lead frame capacitance between package types also affects the oscillation frequency, especially for low CEXT values. The user needs to take into account variation, due to tolerance of the external R and C components. **Figure 7.3** shows how an R/C combination is connected to the PIC16F84A.

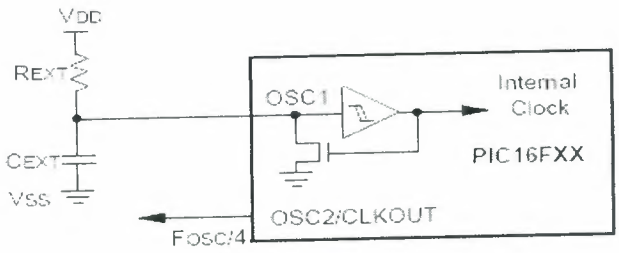


Figure 7.3 RC oscillator mode

Recommended values: 5 kW <= REXT <= 100 kW CEXT > 20pF

3. RESET

he PIC16F84A differentiates between various kinds of RESET:

Power-on Reset (POR)
MCLR during normal operation
MCLR during SLEEP
WDT Reset (during normal operation)
WDT Wake-up (during SLEEP)

Figure 7.4 shows a simplified block diagram of the On-Chip RESET Circuit. The MCLR Reset path has a noise filter to ignore small pulses. The electrical specifications state the pulse width requirements for the MCLR pin.

Some registers are not affected in any RESET condition; their status is unknown on a POR and unchanged in any other RESET. Most other registers are reset to a "RESET state" on POR, MCLR or WDT Reset during normal operation and on MCLR during SLEEP. They are not affected by a WDT Reset during SLEEP, since this RESET is viewed as the resumption of normal operation.

Table 7.3 gives a description of RESET conditions for the program counter (PC) and the STATUS register. **Table 7.4** gives a full description of RESET states for all registers.

The TO and PD bits are set or cleared differently in different RESET situations (Section 7.7). These bits are used in software to determine the nature of the RESET.

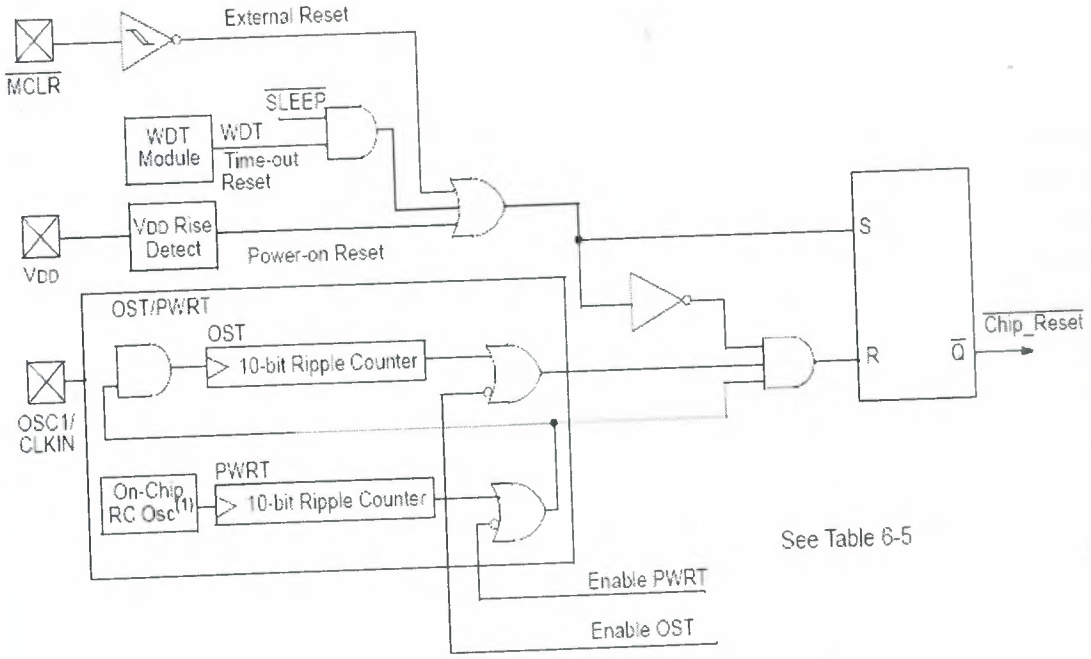


Figure 7.4 Simplified block diagram of on-chip reset circuit

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

2: See **Table 7.5.**

Condition	Program Counter	STATUS Register
Power-on Reset	000h	0001 1xxx
MCLR during normal operation	000h	000u uuuu
MCLR during SLEEP	000h	0001 0uuu
WDT Reset (during normal operation)	000h	0000 1uuu
WDT Wake-up	PC + 1	uuu0 0uuu
Interrupt wake-up from SLEEP	PC + 1(1)	uuu1 0uuu

Table 7.3 Reset condition for program counter and the status register

Legend: u = unchanged, x = unknown

Note 1: When the wake-up is due to an interrupt and the GIE bit is set, the PC is loaded with the interrupt vector (0004h).

Register	Address	Power-on Reset	$\overline{MCLR}$ during: – normal operation – SLEEP WDT Reset during normal operation	Wake-up from SLEEP: – through interrupt – through WDT Time-out
W	—	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	00h	---- ----	---- ----	---- ----
TMR0	01h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02h	0000 0000	0000 0000	PC + 1(2)
STATUS	03h	0001 1xxx	000q quuu(3)	uuuq quuu(3)
FSR	04h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA(4)	05h	---x xxxx	---u uuuu	---u uuuu
PORTB(5)	06h	xxxx xxxx	uuuu uuuu	uuuu uuuu
EEDATA	08h	xxxx xxxx	uuuu uuuu	uuuu uuuu
EEADR	09h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCLATH	0Ah	---0 0000	---0 0000	---u uuuu
INTCON	0Bh	0000 000x	0000 000u	uuuu uuuu(1)
INDF	80h	---- ----	---- ----	---- ----
OPTION_REG	81h	1111 1111	1111 1111	uuuu uuuu
PCL	82h	0000 0000	0000 0000	PC + 1(2)
STATUS	83h	0001 1xxx	000q quuu(3)	uuuq quuu(3)
FSR	84h	xxxx xxxx	uuuu uuuu	uuuu uuuu
TRISA	85h	---1 1111	---1 1111	---u uuuu
TRISB	86h	1111 1111	1111 1111	uuuu uuuu
EECON1	88h	---0 x000	---0 q000	---0 uuuu
EECON2	89h	---- ----	---- ----	---- ----
PCLATH	8Ah	---0 0000	---0 0000	---u uuuu
INTCON	8Bh	0000 000x	0000 000u	uuuu uuuu(1)

Table 7.4 Reset conditions for all registers

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

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

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

3: **Table 6.3** lists the RESET value for each specific condition.

4: On any device RESET, these pins are configured as inputs.

5: This is the value that will be in the port output latch.

7.4. Power-on Reset (POR)

A Power-on Reset pulse is generated on-chip when VDD rise is detected (in the range of 1.2V - 1.7V). To take advantage of the POR, just tie the MCLR pin directly (or through a resistor) to VDD. This will eliminate external RC components usually needed to create Power-on Reset. A minimum rise time for VDD must be met for this to operate properly. See Electrical Specifications for details.

When the device starts normal operation (exits the RESET condition), device operating parameters (voltage, frequency, temperature, etc.) must be met to ensure operation. If these conditions are not met, the device must be held in RESET until the operating conditions are met.

The POR circuit does not produce an internal RESET when VDD declines.

7.5. Power-up Timer (PWRT)

The Power-up Timer (PWRT) provides a fixed 72 ms nominal time-out (TPWRT) from POR (**Figures 7.6** through **7.9**). The Power-up Timer operates on an internal RC oscillator. The chip is kept in RESET as long as the PWRT is active. The PWRT delay allows the VDD to rise to an acceptable level (possible exception shown in **Figure 7.9**).

A configuration bit, PWRTE, can enable/disable the PWRT. See Register 7-1 for the operation of the PWRTE bit for a particular device.

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

7.6. Oscillator Start-up Timer (OST)

The Oscillator Start-up Timer (OST) provides a 1024 oscillator cycle delay (from OSC1 input) after the PWRT delay ends (**Figure 7.6**, **Figure 7.7**, **Figure 7.8** and **Figure 7.9**). This ensures the crystal oscillator or resonator has started and stabilized.

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

When VDD rises very slowly, it is possible that the TPWRT time-out and TOST time-out will expire before VDD has reached its final value. In this case (**Figure 7.9**), an external Power-on Reset circuit may be necessary (**Figure 7.5**).

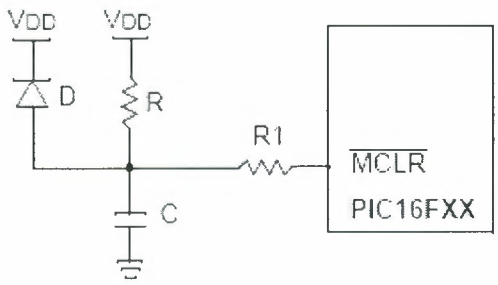


Figure 7.5 External power-on reset circuit (for slow VDD power-up)

Note 1: External Power-on Reset circuit is required only if VDD power-up rate is too slow. The diode D helps discharge the capacitor quickly when VDD powers down.

2: R < 40 kW is recommended to make sure that voltage drop across R does not exceed 0.2V (max leakage current spec on MCLR pin is 5 mA). A larger voltage drop will degrade VIH level on the MCLR pin.

3: R1 = 100W to 1 kW will limit any current flow-ing into MCLR from external capacitor C, in the event of a MCLR pin breakdown due to ESD or EOS.

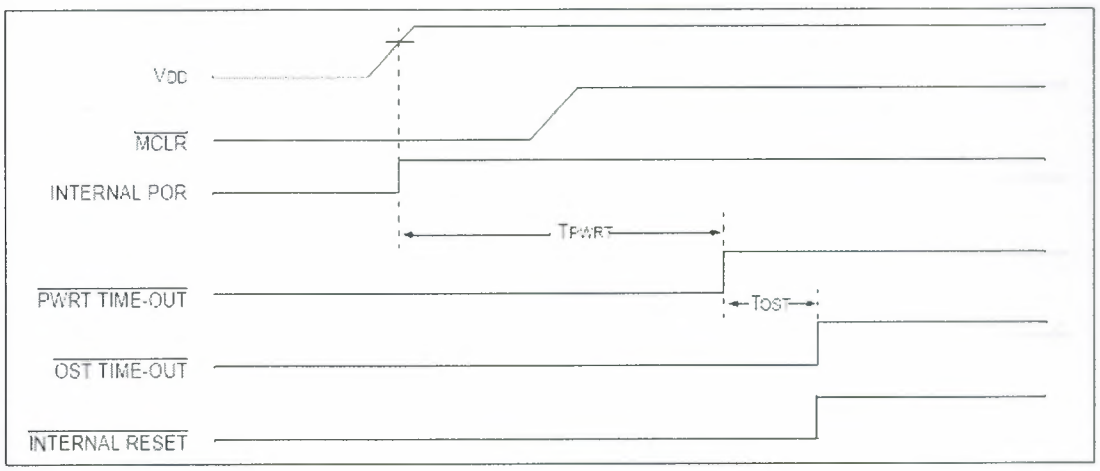
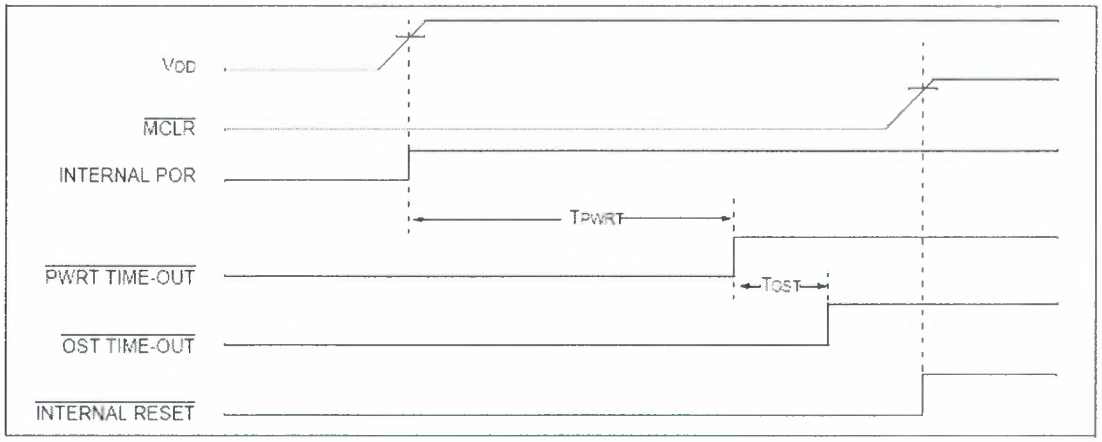


Figure 7.6 Time-out sequence on power-up (MCLR not tied to VDD): case 1



Fifure 7.7 Time-out sequence on power-up (MCLR not tied to VDD): case 2

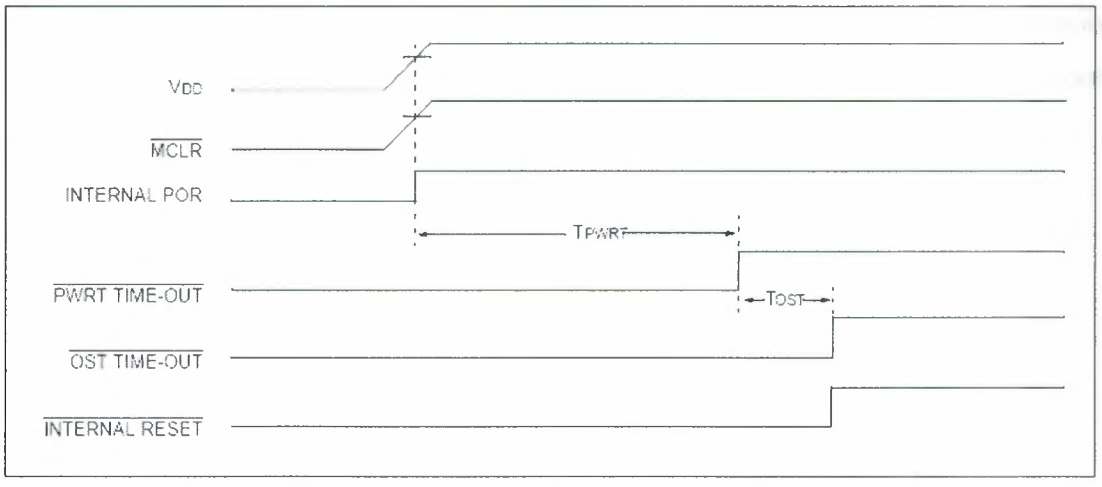


Figure 7.8 Time-out sequence on power-up (MCLR tied to VDD): fast VDD rise time

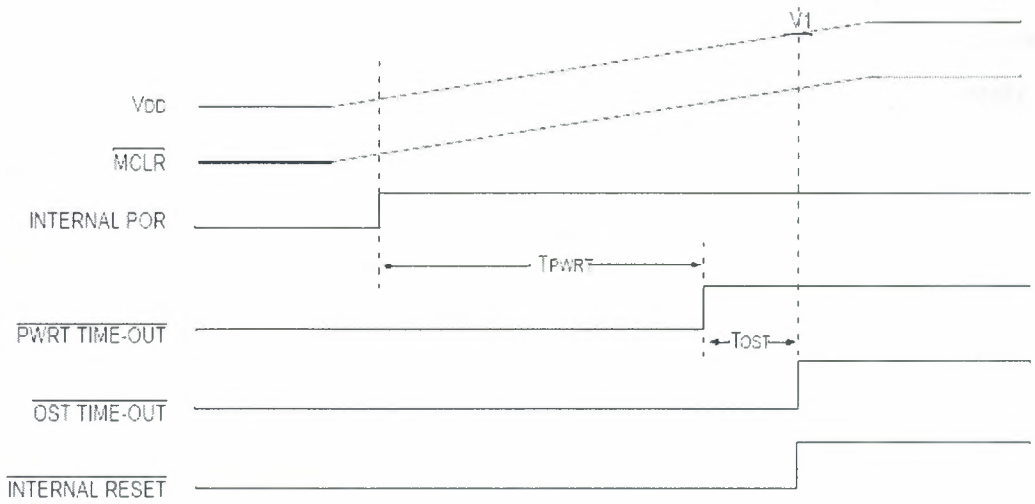


Figure 7.9 Time-out sequence on power-up (MCLR tied to VDD): slow VDD rise time

When VDD rises very slowly, it is possible that the TPWRT time-out and TOST time-out will expire before VDD has reached its final value. In this example, the chip will reset properly if, and only if, V1 ³ VDD min.

7.7. Time-out Sequence and Power-down Status Bits (TO/PD)

On power-up (**Figures 7-6** through **7-9**), the time-out sequence is as follows:

1. PWRT time-out is invoked after a POR has expired.
2. Then, the OST is activated.

The total time-out will vary based on oscillator configuration and PWRTE configuration bit status. For example, in RC mode with the PWRT disabled, there will be no time-out at all.

Oscillator Configuration	Power-up		Wake-up from SLEEP
	PWRT Enabled	PWRT Disabled	
XT, HS, LP	72 ms + 1024Tosc	1024Tosc	1024Tosc
RC	72 ms	—	—

Table 7.5 Time-out in various situations

Since the time-outs occur from the POR pulse, if MCLR is kept low long enough, the time-outs will expire. Then bringing MCLR high, execution will begin immediately (**Figure 7.6**). This is useful for testing purposes or to synchronize more than one PIC16F84A device when operating in parallel.

Table 7.6 shows the significance of the TO and PD bits. **Table 7.3** lists the RESET conditions for some special registers, while **Table 7.4** lists the RESET conditions for all the registers.

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

Table 7.6 Status bits and their significance

7.8. Interrupts

The PIC16F84A has 4 sources of interrupt:

- External interrupt RB0/INT pin
- TMR0 overflow interrupt
- PORTB change interrupts (pins RB7:RB4)
- Data EEPROM write complete interrupt

The interrupt control register (INTCON) records individual interrupt requests in flag bits. It also contains the individual and global interrupt enable bits.

The global interrupt enable bit, GIE (INTCON<7>), enables (if set) all unmasked interrupts or disables (if cleared) all interrupts. Individual interrupts can be disabled through their corresponding enable bits in INTCON register. Bit GIE is cleared on RESET.

The "return from interrupt" instruction, RETFIE, exits interrupt routine as well as sets the GIE bit, which re-enables interrupts.

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

When an interrupt is responded to, the GIE bit is cleared to disable any further interrupt, the return address is pushed onto the stack and the PC is loaded with 0004h. For external interrupt events, such as the RB0/INT pin or PORTB change interrupt, the interrupt latency will be three to four instruction cycles. The exact latency depends when the interrupt event occurs. The latency is the same for both one and two cycle instructions. Once in the Interrupt Service Routine, the source(s) of the interrupt can be determined by polling the interrupt flag bits. The interrupt flag bit(s) must be cleared in software before re-enabling interrupts to avoid infinite interrupt requests.

Note: Individual interrupt flag bits are set regardless of the status of their corresponding mask bit or the GIE bit.

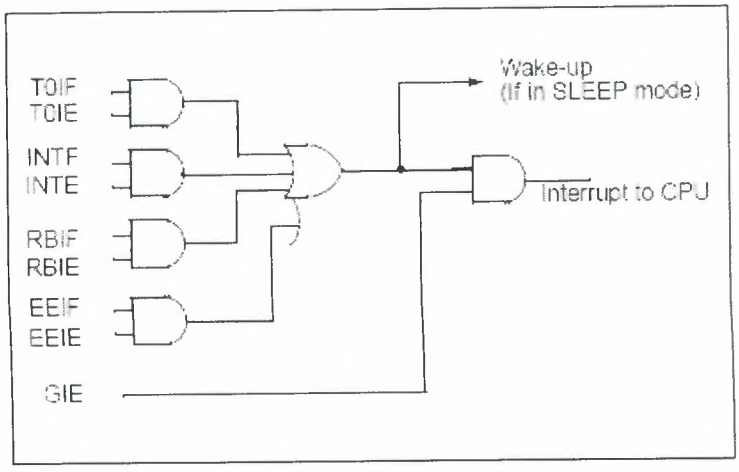


Figure 7.10 Interrupt logic

7.8.1. INT Interrupt

External interrupt on RB0/INT pin is edge triggered: either rising if INTEDG bit (OPTION_REG<6>) is set, or falling if INTEDG bit is clear. When a valid edge appears on the RB0/INT pin, the INTF bit (INTCON<1>) is set. This interrupt can be disabled by clearing control bit INTE (INTCON<4>). Flag bit INTF must be cleared in software via the Interrupt Service Routine before re-enabling this interrupt. The INT interrupt can wake the processor from SLEEP (Section 7.11) only if the INTE bit was set prior to going into SLEEP. The status of the GIE bit decides whether the processor branches to the interrupt vector following wake-up.

7.8.2. TMR0 Interrupt

An overflow (FFh → 00h) in TMR0 will set flag bit T0IF (INTCON<2>). The interrupt can be enabled/disabled by setting/clearing enable bit T0IE (INTCON<5>) (Section 6).

7.8.3. PORTB Interrupt

An input change on PORTB<7:4> sets flag bit RBIF (INTCON<0>). The interrupt can be enabled/disabled by setting/clearing enable bit RBIE (INTCON<3>) (Section 5.2).

Note: For a change on the I/O pin to be recognized, the pulse width must be at least TCY wide.

7.8.4. DATA EEPROM Interrupt

At the completion of a data EEPROM write cycle, flag bit EEIF (EECON1<4>) will be set. The interrupt can be enabled/disabled by setting/clearing enable bit EEIE (INTCON<6>) (Section 4.0).

7.9. Context Saving During Interrupts

During an interrupt, only the return PC value is saved on the stack. Typically, users wish to save key register values during an interrupt (e.g., W register and STATUS register). This is implemented in software.

The code in Example 7-1 stores and restores the STATUS and W register's values. The user defined registers, W_TEMP and STATUS_TEMP are the temporary storage locations for the W and STATUS registers values.

EXAMPLE 7-1 DOES THE FOLLOWINGS:

a) Stores the W register.
b) Stores the STATUS register in STATUS_TEMP.
c) Executes the Interrupt Service Routine code.

d) Restores the STATUS (and bank select bit) register.

e) Restores the W register.

EXAMPLE 7-1: SAVING STATUS AND W REGISTERS IN RAM

```
PUSH MOVWF    W_TEMP          ; Copy W to TEMP register,
     SWAPF    STATUS, W       ; Swap status to be saved into W
     MOVWF    STATUS_TEMP     ; Save status to STATUS_TEMP register
ISR  :                        ;
     :                        ; Interrupt Service Routine
     :                        ; should configure Bank as required
     :                        ;
POP  SWAPF    STATUS_TEMP,W   ; Swap nibbles in STATUS_TEMP register
                              ; and place result into W
     MOVWF    STATUS          ; Move W into STATUS register
                              ; (sets bank to original state)
     SWAPF    W_TEMP, F       ; Swap nibbles in W_TEMP and place result
                              ; in W_TEMP
     SWAPF    W_TEMP, W       ; Swap nibbles in W_TEMP and place result
                              ;into W
```

7.10. Watchdog Timer (WDT)

The Watchdog Timer is a free running On-Chip RC Oscillator which does not require any external components. This RC oscillator is separate from the RC oscillator of the OSC1/CLKIN pin. That means that the WDT will run even if the clock on the OSC1/CLKIN and OSC2/CLKOUT pins of the device has been stopped, for example, by execution of a SLEEP instruction. During normal operation, a WDT time-out generates a device RESET. If the device is in SLEEP mode, a WDT wake-up causes the device to wake-up and continue with normal operation. The WDT can be permanently disabled by programming configuration bit WDTE as a '0' (Section 7.1).

7.10.1. 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 (see DC specs). If longer time-out periods are desired, a prescaler with a division ratio of up to 1:128 can be assigned to the WDT under software control by writing to the OPTION_REG register. Thus, time-out periods up to 2.3 seconds can be realized.

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

The TO bit in the STATUS register will be cleared upon a WDT time-out.

7.10.2. WDT Programming Considerations

It should also be taken into account that under worst case conditions (VDD = Min., Temperature = Max., Max. WDT Prescaler), it may take several seconds before a WDT time-out occurs.

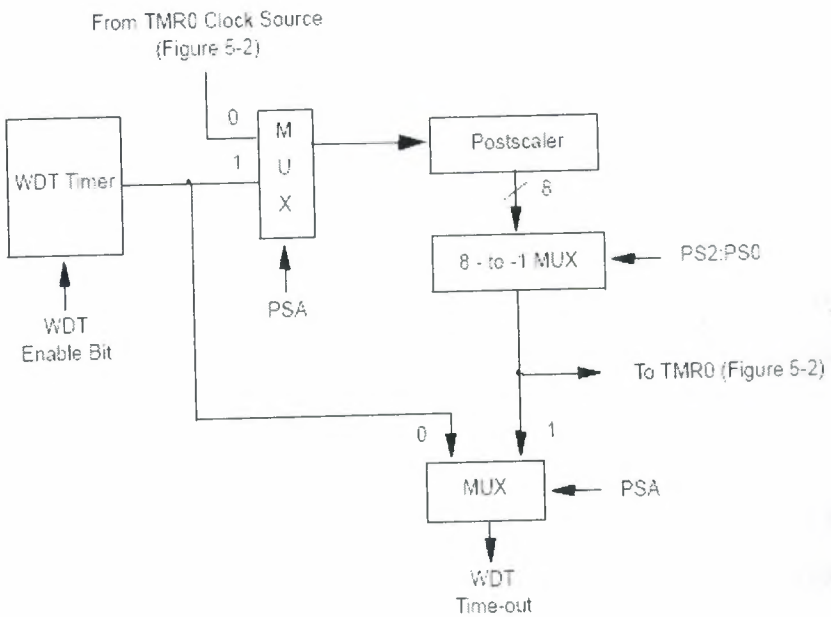


Figure 7.11 WATCHDOG timer block diagram

Note: PSA and PS2:PS0 are bits in the OPTION_REG register.

Addr	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on Power-on Reset	Value on all other RESETS
2007h	Config. bits	(2)	(2)	(2)	(2)	$\overline{\text{PWRTE}}$[1]	WDTE	FOSC1	FOSC0	(2)	
81h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

Table 7.7 Summary of registers associated with the WATCHDOG timer

Legend: x = unknown. Shaded cells are not used by the WDT.

Note 1: See Register 7-1 for operation of the PWRTE bit.

2: See Register 7-1 and Section 7.12 for operation of the code and data protection bits.

7.11. Power-down Mode (SLEEP)

A device may be powered down (SLEEP) and later powered up (wake-up from SLEEP).

7.11.1. SLEEP

The Power-down mode is entered by executing the SLEEP instruction.

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

For the lowest current consumption in SLEEP mode, place all I/O pins at either VDD or VSS, with no external circuitry drawing current from the I/O pins, and disable external clocks. I/O pins that are hi-impedance inputs should be pulled high or low externally to avoid switching currents caused by floating inputs. The T0CKI input should also be at VDD or VSS. The contribution from on-chip pull-ups on PORTB should be considered.

The MCLR pin must be at a logic high level (VIHMC).

It should be noted that a RESET generated by a WDT time-out does not drive the MCLR pin low.

7.11.2. WAKE-UP From SLEEP

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

1. External RESET input on MCLR pin.
2. WDT wake-up (if WDT was enabled).
3. Interrupt from RB0/INT pin, RB port change, or data EEPROM write complete.

Peripherals cannot generate interrupts during SLEEP, since no on-chip Q clocks are present.

The first event (MCLR Reset) will cause a device RESET. The two latter events are considered a continuation of program execution. The TO and PD bits can be used to determine the cause of a device RESET. The PD bit, which is set on power-up, is cleared when SLEEP is invoked. The TO bit is cleared if a WDT time-out occurred (and caused wake-up).

While the SLEEP instruction is being executed, the next instruction (PC + 1) is pre-fetched. For the device to wake-up through an interrupt event, the corresponding interrupt enable bit must be set (enabled). Wake-up occurs regardless of the state of the GIE bit. If the GIE bit is clear (disabled), the device continues execution at the instruction after the SLEEP instruction. If the GIE bit is set (enabled), the device executes the instruction after the SLEEP instruction and then branches to the interrupt address (0004h). In cases where the execution of the instruction following SLEEP is not desirable, the user should have a NOP after the SLEEP instruction.

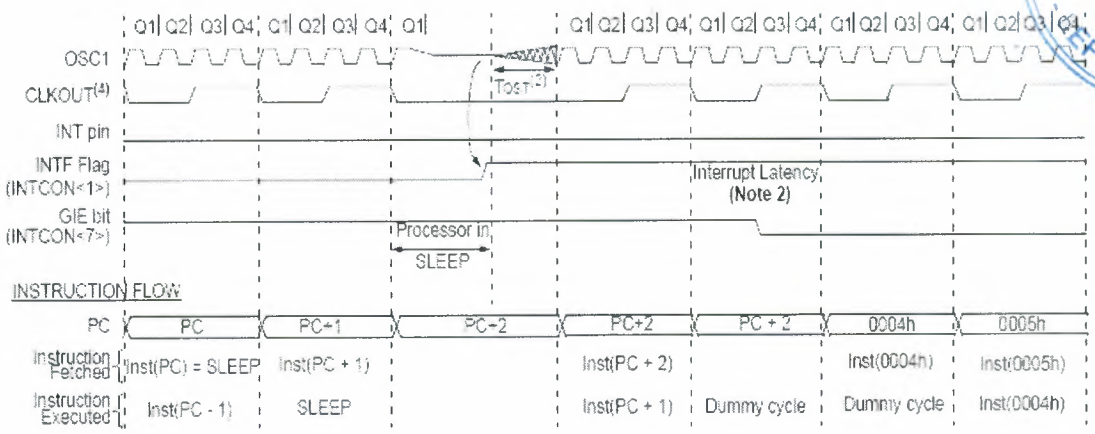


Figure 7.12 Wake-up from sleep through interrupt

Note 1: XT, HS, or LP oscillator mode assumed.

2: TOST = 1024TOSC (drawing not to scale). This delay will not be there for RC osc mode.

3: GIE = '1' assumed. In this case after wake-up, the processor jumps to the interrupt routine. If GIE = '0', execution will continue in-line.

4: CLKOUT is not available in these osc modes, but shown here for timing reference.

7.11.3. WAKE-UP Using INTERRUPTS

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

• If the interrupt occurs before the execution of a SLEEP instruction, the SLEEP instruction will complete as a NOP. Therefore, the WDT and WDT postscaler will not be cleared, the TO bit will not be set and PD bits will not be cleared.

• If the interrupt occurs during or after the execution of a SLEEP instruction, the device will immediately wake-up from SLEEP. The SLEEP instruction will be completely executed before the wake-up. Therefore, the WDT and WDT postscaler will be cleared, the TO bit will be set and the PD bit will be cleared. Even if the flag bits were checked before executing a SLEEP instruction, it may be possible for flag bits to become set before the SLEEP instruction completes. To determine whether a SLEEP instruction

executed, test the PD bit. If the PD bit is set, the SLEEP instruction was executed as a NOP.

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

7.12. Program Verification/Code Protection

If the code protection bit(s) have not been programmed, the on-chip program memory can be read out for verification purposes.

7.13. ID Locations

Four memory locations (2000h - 2004h) are designated as ID locations to store checksum or other code identification numbers. These locations are not accessible during normal execution but are readable and writable only during program/verify. Only the four Least Significant bits of ID location are usable.

7.14. In-Circuit Serial Programming

PIC16F84A microcontrollers can be serially programmed while in the end application circuit. This is simply done with two lines for clock and data, and three other lines for power, ground, and the programming voltage. Customers can manufacture boards with unprogrammed devices, and then program the microcontroller just before shipping the product, allowing the most recent firmware or custom firmware to be programmed.

Juniour pic programmer is used to load the program to the PIC16F84A. It is good at programming pic f series. Also it is cheap and no need any other connection. RS232 board makes it easy to add a serial port. It shown in **Figure 7.13**. Only one chip must be found on it when programming. Programming the chip is impractical with laptops.

Figure 7.13 Pic and Eproms can programmed

8. INSTRUCTION SET SUMMARY

Each PIC16CXX instruction is a 14-bit word, divided into an OPCODE which specifies the instruction type and one or more operands which further specify the operation of the instruction. The PIC16CXX instruction set summary in **Table 8.2** lists byte-oriented, bit-oriented, and literal and control operations. **Table 8.1** shows the opcode field descriptions.

For byte-oriented instructions, 'f' represents a file register designator and 'd' represents a destination designator. The file register designator specifies which file register is to be used by the instruction.

The destination designator specifies where the result of the operation is to be placed. If 'd' is zero, the result is placed in the W register. If 'd' is one, the result is placed in the file register specified in the instruction.

For bit-oriented instructions, 'b' represents a bit field designator which selects the number of the bit affected by the operation, while 'f' represents the address of the file in which the bit is located.

For literal and control operations, 'k' represents an eight or eleven bit constant or literal value.

Field	Description
f	Register file address (0x00 to 0x7F)
w	Working register (accumulator)
b	Bit address within an 8-bit file register
k	Literal field, constant data or label
x	Don't care location (= 0 or 1) The assembler will generate code with x = 0. It is the recommended form of use for compatibility with all Microchip software tools.
d	Destination select; d = 0: store result in W, d = 1: store result in file register f. Default is d = 1
PC	Program Counter
TO	Time-out bit
PD	Power-down bit

Table 8.1

The instruction set is highly orthogonal and is grouped into three basic categories:

- Byte-oriented operations
- Bit-oriented operations
- Literal and control operations

All instructions are executed within one single instruction cycle, unless a conditional test is true or the program counter is changed as a result of an instruction. In this case, the execution takes two instruction cycles with the second cycle executed as a NOP. One instruction cycle consists of four oscillator periods. Thus, for an oscillator frequency of 4 MHz, the normal instruction execution time is 1 ms. If a conditional test is true or the program counter is changed as a result of an instruction, the instruction execution time is 2 ms.

Table 8.2 lists the instructions recognized by the MPASM™ Assembler.

Figure 8.1 shows the general formats that the instructions can have.

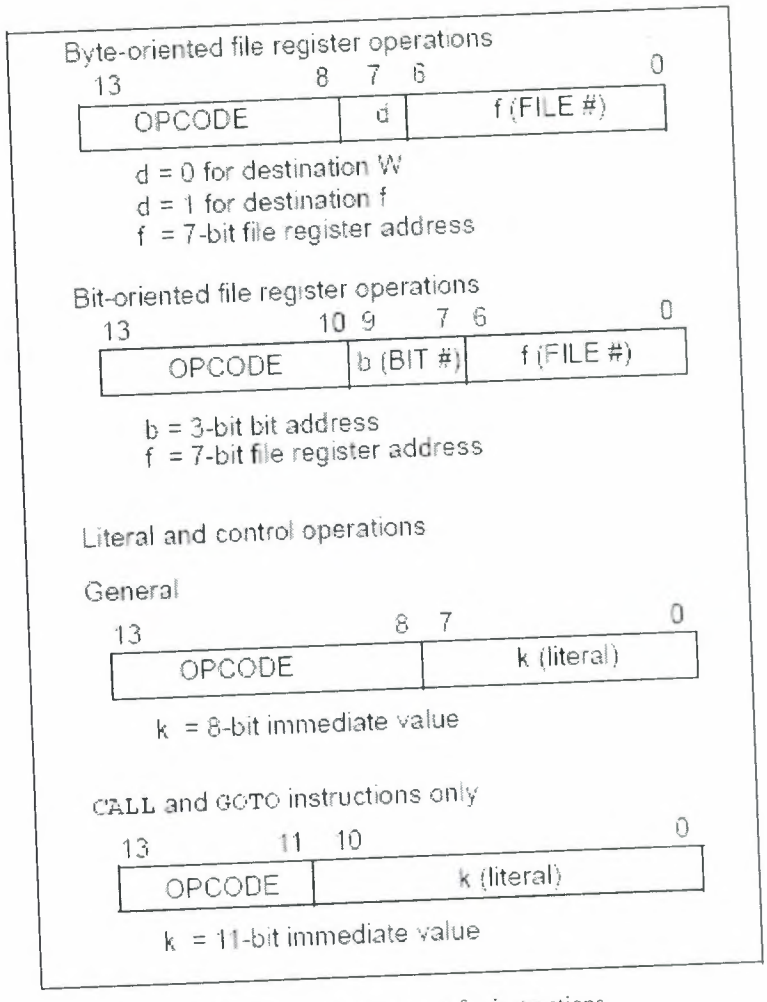


Figure 8.1 general format for instructions

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

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

0xhh where h signifies a hexadecimal digit.

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb LSb	Status Affected	Notes
BYTE-ORIENTED FILE REGISTER OPERATIONS						
ADDWF	f, d	Add W and f	1	00 0111 dfff ffff	C,DC,Z	1,2
ANDWF	f, d	AND W with f	1	00 0101 dfff ffff	Z	1,2
CLRF	f	Clear f	1	00 0001 lfff ffff	Z	2
CLRW	-	Clear W	1	00 0001 0xxx xxxx	Z	
COMF	f, d	Complement f	1	00 1001 dfff ffff	Z	1,2
DECF	f, d	Decrement f	1	00 0011 dfff ffff	Z	1,2
DECFSZ	f, d	Decrement f, Skip if 0	1 (2)	00 1011 dfff ffff		1,2,3
INCF	f, d	Increment f	1	00 1010 dfff ffff	Z	1,2
INCFSZ	f, d	Increment f, Skip if 0	1 (2)	00 1111 dfff ffff		1,2,3
IORWF	f, d	Inclusive OR W with f	1	00 0100 dfff ffff	Z	1,2
MOVF	f, d	Move f	1	00 1000 dfff ffff	Z	1,2
MOVWF	f	Move W to f	1	00 0000 lfff ffff		
NOP	-	No Operation	1	00 0000 0xx0 0000		
RLF	f, d	Rotate Left f through Carry	1	00 1101 dfff ffff	C	1,2
RRF	f, d	Rotate Right f through Carry	1	00 1100 dfff ffff	C	1,2
SUBWF	f, d	Subtract W from f	1	00 0010 dfff ffff	C,DC,Z	1,2
SWAPF	f, d	Swap nibbles in f	1	00 1110 dfff ffff		1,2
XORWF	f, d	Exclusive OR W with f	1	00 0110 dfff ffff	Z	1,2
BIT-ORIENTED FILE REGISTER OPERATIONS						
BCF	f, b	Bit Clear f	1	01 00bb bfff ffff		1,2
BSF	f, b	Bit Set f	1	01 01bb bfff ffff		1,2
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01 10bb bfff ffff		3
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01 11bb bfff ffff		3
LITERAL AND CONTROL OPERATIONS						
ADDLW	k	Add literal and W	1	11 111x kkkk kkkk	C,DC,Z	
ANDLW	k	AND literal with W	1	11 1001 kkkk kkkk	Z	
CALL	k	Call subroutine	2	10 0kkk kkkk kkkk		
CLRWDT	-	Clear Watchdog Timer	1	00 0000 0110 0100	$\overline{TO}$,$\overline{PD}$	
GOTO	k	Go to address	2	10 1kkk kkkk kkkk		
IORLW	k	Inclusive OR literal with W	1	11 1000 kkkk kkkk	Z	
MOVLW	k	Move literal to W	1	11 00xx kkkk kkkk		
RETFIE	-	Return from interrupt	2	00 0000 0000 1001		
RETLW	k	Return with literal in W	2	11 01xx kkkk kkkk		
RETURN	-	Return from Subroutine	2	00 0000 0000 1000		
SLEEP	-	Go into standby mode	1	00 0000 0110 0011	$\overline{TO}$,$\overline{PD}$	
SUBLW	k	Subtract W from literal	1	11 110x kkkk kkkk	C,DC,Z	
XORLW	k	Exclusive OR literal with W	1	11 1010 kkkk kkkk	Z	

Table 8.2 PIC16CXXX instruction set

Note 1: When an I/O register is modified as a function of itself (e.g., MOVF PORTB, 1), the value used will be that value present on the pins themselves. For example, if the data latch is '1' for a pin configured as input and is driven low by an external device, the data will be written back with a '0'.

2: If this instruction is executed on the TMR0 register (and, where applicable, d = 1), the prescaler will be cleared if assigned to the Timer0 Module.

3: If Program Counter (PC) is modified or a conditional test is true, the instruction requires two cycles. The second cycle is executed as a NOP.

8.1. Instruction Descriptions

ADDLW Add Literal and W k

Syntax: [label] ADDLW

Operands: $0 \leq k \leq 255$

Operation: (W) + k → (W)

Status Affected: C, DC, Z

Description: The contents of the W register are added to the eight-bit literal 'k' and the result is placed in the W register.

ADDWF Add W and f

Syntax: [label] ADDWF f,d

Operands: $0 \leq f \leq 127$

d € [0,1]

Operation: (W) + (f) → (destination)

Status Affected: C, DC, Z

Description: Add the contents of the W register with register 'f'. If 'd' is 0, the result is stored in the W register. If 'd' is 1, the result is stored back in register 'f'.

ANDLW AND Literal with W

Syntax: [label] ANDLW k

Operands: $0 \leq k \leq 255$

Operation: (W) .AND. (k) → (W)

Status Affected: Z

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

ANDWF AND W with f

Syntax: [label] ANDWF f,d

Operands: $0 \leq f \leq 127$

d € [0,1]

Operation: (W) .AND. (f) → (destination)

Status Affected: Z

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

BCF Bit Clear f

Syntax: [label] BCF f,b

Operands: $0 \leq f \leq 127$

$0 \leq b \leq 7$

Operation: $0 \rightarrow (f<b>)$

Status Affected: None

Description: Bit 'b' in register 'f' is cleared.

BSF Bit Set f

Syntax: [label] BSF f,b

Operands: $0 \leq f \leq 127$

$0 \leq b \leq 7$

Operation: $1 \rightarrow (f<b>)$

Status Affected: None

Description: Bit 'b' in register 'f' is set.

BTFSS Bit Test f, Skip if Set

Syntax: [label] BTFSS f,b

Operands: $0 \leq f \leq 127$

$0 \leq b < 7$

Operation: skip if $(f<b>) = 1$

Status Affected: None

Description: If bit 'b' in register 'f' is '0', the next instruction is executed. If bit 'b' is '1', then the next instruction is discarded and a NOP is executed instead, making this a 2TCY instruction.

BTFSC Bit Test, Skip if Clear

Syntax: [label] BTFSC f,b

Operands: $0 \leq f \leq 127$

$0 \leq b \leq 7$

Operation: skip if $(f<b>) = 0$

Status Affected: None

Description: If bit 'b' in register 'f' is '1', the next instruction is executed. If bit 'b' in register 'f' is '0', the next instruction is discarded, and a NOP is executed instead, making this a 2TCY instruction.

CALL Call Subroutine

Syntax: [label] CALL k

Operands: $0 \leq k \leq 2047$

Operation: (PC)+ 1→ TOS,

k → PC<10:0>,

(PCLATH<4:3>) → PC<12:11>

Status Affected: None

Description: Call Subroutine. First, return address (PC+1) is pushed onto the stack. The eleven-bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALLis a two-cycle instruction.

CLRF Clear f

Syntax: [label] CLRF f

Operands: $0 \leq f \leq 127$

Operation: 00h → (f)

1 → Z

Status Affected: Z

Description: The contents of register 'f' are cleared and the Z bit is set.

CLRW Clear W

Syntax: [label] CLRW

Operands: None

Operation: 00h → (W)

1 → Z

Status Affected: Z

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

CLRWDT Clear Watchdog Timer

Syntax: [label] CLRWDT

Operands: None

Operation: 00h → WDT

0 → WDT prescaler,

1 → TO

1 → PD

Status Affected: TO, 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: [label] COMF f,d

Operands: $0 \leq f \leq 127$

d € [0,1]

Operation: (f) → (destination)

Status Affected: Z

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

DECF Decrement f

Syntax: [label] DECF f,d

Operands: $0 \leq f \leq 127$

d € [0,1]

Operation: (f) - 1 → (destination)

Status Affected: Z

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

DECFSZ Decrement f, Skip if 0

Syntax: [label] DECFSZ f,d

Operands: $0 \leq f \leq 127$

d € [0,1]

Operation: (f) - 1 → (destination);

skip if result = 0

Status Affected: None

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

If the result is 1, the next instruction is executed. If the result is 0, then a NOP is executed instead, making it a 2TCY instruction.

GOTO Unconditional Branch

Syntax: [label] GOTO k

Operands: $0 \leq k \leq 2047$

Operation: k → PC<10:0>

PCLATH<4:3> → PC<12:11>

Status Affected: None

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

INCF Increment f

Syntax: [label] INCF f,d

Operands: $0 \leq f \leq 127$

d € [0,1]

Operation: (f) + 1 → (destination)

Status Affected: Z

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

INCFSZ Increment f, Skip if 0

Syntax: [label] INCFSZ f,d

Operands: $0 \leq f \leq 127$

d € [0,1]

Operation: (f) + 1 → (destination),

skip if result = 0

Status Affected: None

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

If the result is 1, the next instruction is executed. If the result is 0, a NOPis executed instead, making it a 2TCY instruction.

IORLW Inclusive OR Literal with W

Syntax: [label] IORLW k

Operands: $0 \leq k \leq 255$

Operation: (W) .OR. k → (W)

Status Affected: Z

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

IORWF Inclusive OR W with f

Syntax: [label] IORWF f,d

Operands: $0 \leq f \leq 127$

d € [0,1]

Operation: (W) .OR. (f) → (destination)

Status Affected: Z

Description: Inclusive OR the W register with register 'f'. If 'd' is 0, the result is placed in the W register. If 'd' is 1, the result is placed back in register 'f'.

MOVF Move f

Syntax: [label] MOVF f,d

Operands: $0 \leq f \leq 127$

d € [0,1]

Operation: (f) → (destination)

Status Affected: Z

Description: The contents of register f are moved to a destination dependant upon the status of d. If d = 0, destination is W register. If d = 1, the destination is file register f itself. d = 1 is useful to test a file register, since status flag Z is affected.

MOVLW Move Literal to W

Syntax: [label] MOVLW k

Operands: $0 \leq k \leq 255$

Operation: k → (W)

Status Affected: None

Description: The eight-bit literal 'k' is loaded into W register. The don't cares will assemble as 0's.

MOVWF Move W to f

Syntax: [label] MOVWF f

Operands: $0 \leq f \leq 127$

Operation: (W) → (f)

Status Affected: None

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

NOP No Operation

Syntax: [label] NOP

Operands: None

Operation: No operation

Status Affected: None

Description: No operation.

RETFIE Return from Interrupt

Syntax: [label] RETFIE

Operands: None

Operation: TOS → PC,

1 → GIE

Status Affected: None

RETLW Return with Literal in W

Syntax: [label] RETLW k

Operands: $0 \leq k \leq 255$

Operation: k → (W);

TOS → PC

Status Affected: None

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

RETURN Return from Subroutine

Syntax: [label] RETURN

Operands: None

Operation: TOS → PC

Status Affected: None

Description: Return from subroutine. The stack is POPed and the top of the stack (TOS) is loaded into the program counter. This is a two-cycle instruction.

RLF Rotate Left f through Carry

Syntax: [label] RLF f,d

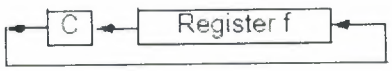
Operands: $0 \leq f \leq 127$

$d \in [0,1]$

Operation: See description below

Status Affected: C

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



RRF Rotate Right f through Carry

Syntax: [label] RRF f,d

Operands: $0 \leq f \leq 127$

$d \in [0,1]$

Operation: See description below

Status Affected: C

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

SLEEP

Syntax: [label]SLEEP

Operands: None

Operation: 00h → WDT,

0 → WDT prescaler,

1 → TO,

0 → PD

Status Affected: 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.

SUBLW Subtract W from Literal

Syntax: [label] SUBLW k

Operands: $0 \leq k \leq 255$

Operation: k - (W) → (W)

Status Affected: C, DC, Z

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

SUBWF Subtract W from f

Syntax: [label] SUBWF f,d

Operands: $0 \leq f \leq 127$

d € [0,1]

Operation: (f) - (W) → (destination)

Status Affected: C, DC, Z

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

SWAPF Swap Nibbles in f

Syntax: [label] SWAPF f,d

Operands: $0 \leq f \leq 127$

$d \in [0,1]$

Operation: (f<3:0>) → (destination<7:4>),

(f<7:4>) → (destination<3:0>)

Status Affected: 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'.

XORLW Exclusive OR Literal with W

Syntax: [label] XORLW k

Operands: $0 \leq k \leq 255$

Operation: (W) .XOR. k → (W)

Status Affected: Z

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

XORWF Exclusive OR W with f

Syntax: [label] XORWF f,d

Operands: $0 \leq f \leq 127$

$d \in [0,1]$

Operation: (W) .XOR. (f) → (destination)

Status Affected: Z

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

9. DEVELOPMENT SUPPORT

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

- Integrated Development Environment

-MPLAB® IDE Software

- Assemblers/Compilers/Linkers

-MPASMTM Assembler

-MPLAB C17 and MPLAB C18 C Compilers

-MPLINKTM Object Linker/ MPLIBTM Object Librarian

• Simulators

-MPLAB SIM Software Simulator

•Emulators

-MPLAB ICE 2000 In-Circuit Emulator

-ICEPIC™ In-Circuit Emulator

• In-Circuit Debugger

-MPLAB ICD

• Device Programmers

-PRO MATE® II Universal Device Programmer

-PICSTART® Plus Entry-Level Development Programmer

• Low Cost Demonstration Boards

-PICDEMTM 1 Demonstration Board

-PICDEM 2 Demonstration Board

-PICDEM 3 Demonstration Board

-PICDEM 17 Demonstration Board

-KEELOQ® Demonstration Board

		PIC12CXXX	PIC14000	PIC16C5X	PIC16C6X	PIC16CXXX	PIC16F6XX	PIC16C7X	PIC16C7XX	PIC16C8X	PIC16F8XX	PIC16C9XX	PIC17C4X	PIC17C7XX	PIC18CXX2	PIC18FXXX	24CXX/ 25CXX/ 93CXX	HCSXXX	MCRFXXX	MCP2510
Software Tools	MPLAB® Integrated Development Environment	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
	MPLAB® C17 C Compiler												✓	✓						
	MPLAB® C18 C Compiler														✓	✓				
	MPASM™ Assembler/ MPLINK™ Object Linker	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		
Emulators	MPLAB® ICE In-Circuit Emulator	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
	ICEPIC™ In-Circuit Emulator	✓		✓	✓	✓	✓	✓	✓	✓		✓								
Debugger	MPLAB® ICD In Circuit Debugger				✓			✓			✓									
Programmers	PICSTART® Plus Entry Level Development Programmer	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
	PRO MATE® II Universal Device Programmer	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓		
Demo Boards and Eval Kits	PICDEM™ 1 Demonstration Board			✓		✓		✓		✓			✓							
	PICDEM™ 2 Demonstration Board							✓								✓				
	PICDEM™ 3 Demonstration Board											✓								
	PICDEM™ 14A Demonstration Board		✓																	
	PICDEM™ 17 Demonstration Board													✓						
	KEELOQ® Evaluation Kit																	✓		
	KEELOQ® Transponder Kit																	✓		
	microID™ Programmer's Kit																		✓	
	125 kHz microID™ Developer's Kit																		✓	
	125 kHz Anticollision microID™ Developer's Kit																		✓	
	13.56 MHz Anticollision microID™ Developer's Kit																		✓	
	MCP2510 CAN Developer's Kit																			✓

* Contact the Microchip Technology Inc. web site at www.microchip.com for information on how to use the MPLAB® ICD In-Circuit Debugger (DV164001) with PIC16C63, 63A, 64, 64A, 65, 65A, 72, 73, 73A, 74, 74A, 76, 77.
** Contact Microchip Technology Inc. for availability date
† Development tool is available on select devices.

Table 9.1 Development tools

10. ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings †

Ambient temperature under bias -55°C to +125°C

Storage temperature -65°C to +150°C

Voltage on any pin with respect to VSS (except VDD, MCLR, and RA4)

-0.3V to (VDD + 0.3V)

Voltage on VDD with respect to VSS	-0.3 to +7.5V
Voltage on MCLR with respect to VSS(1)	-0.3 to +14V
Voltage on RA4 with respect to VSS	-0.3 to +8.5V
Total power dissipation(2)	800 mW
Maximum current out of VSS pin	150 mA
Maximum current into VDD pin	100 mA
Input clamp current, IIK (VI < 0 or VI > VDD)	± 20 mA
Output clamp current, IOK (VO < 0 or VO > VDD)	± 20 mA
Maximum output current sunk by any I/O pin	25 mA
Maximum output current sourced by any I/O pin	25 mA
Maximum current sunk by PORTA	80 mA
Maximum current sourced by PORTA	50 mA
Maximum current sunk by PORTB	150 mA
Maximum current sourced by PORTB	100 mA

Note 1: Voltage spikes below VSS at the MCLR pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50-100W should be used when applying a "low" level to the MCLR pin rather than pulling this pin directly to VSS.

2: Power dissipation is calculated as follows: Pdis = VDD x {IDD -å IOH} + å {(VDD-VOH) x IOH} + å(VOl x IOL).

† NOTICE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at those or any other conditions above those indicated in the operation listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

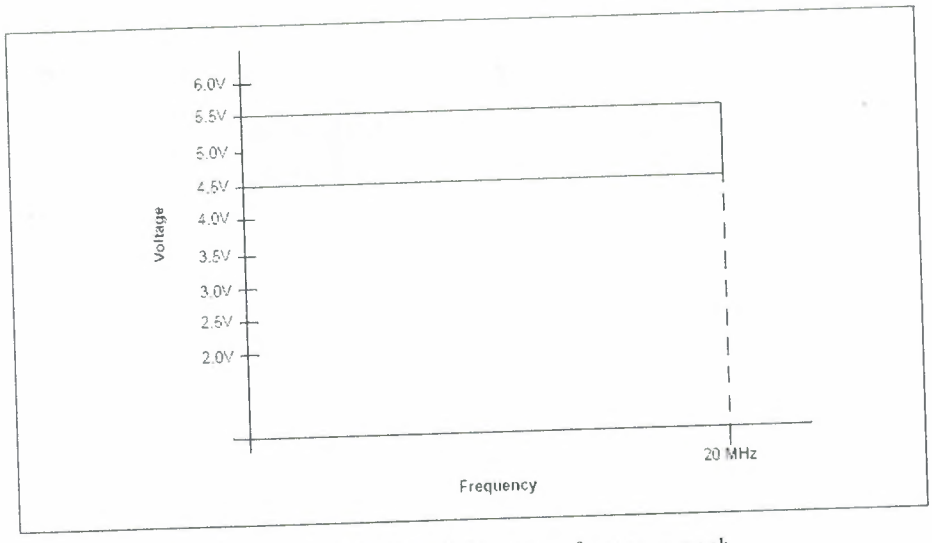


Figure 10.1 PIC16F84A-20 voltage-frequency graph

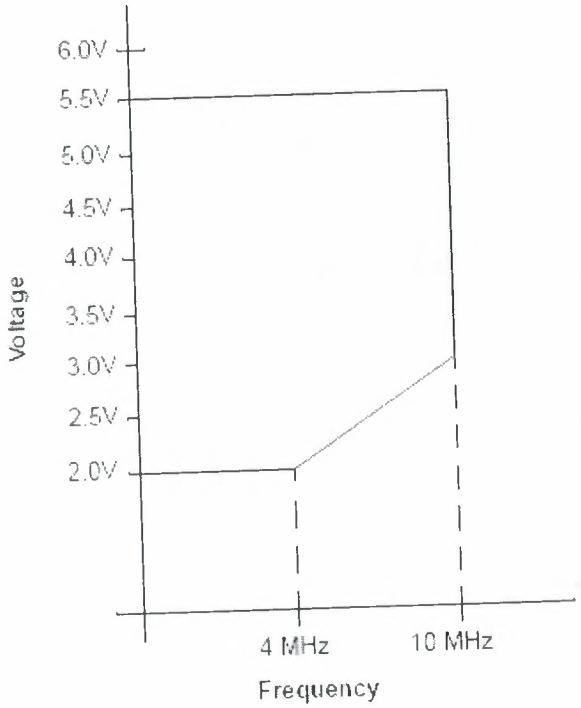


Figure 10.2 PIC16FL84A voltage-frequency graph

FMAX = (6.0 MHz/V) (VDDAPPMIN - 2.0V) + 4 MHz

Note 1: VDDAPPMIN is the minimum voltage of the PICmicro® device in the application.

2: FMAX has a maximum frequency of 10 MHz.

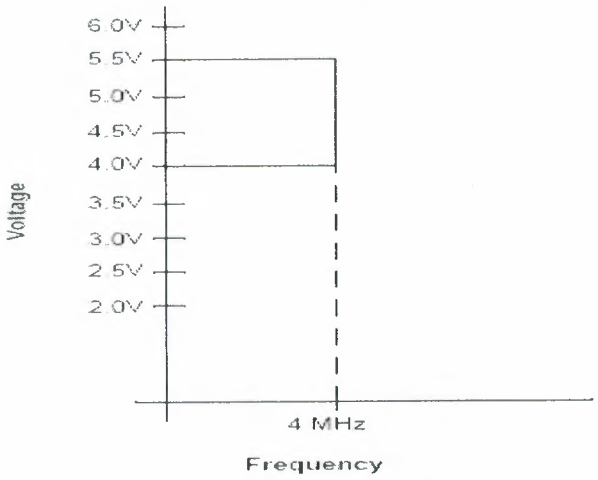


Figure 10.3 PIC16F84A-04 voltage-frequancy graph

10.1. DC Characteristics

PIC16LF84A-04 (Commercial, Industrial)			**Standard Operating Conditions (unless otherwise stated)** Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended)				
PIC16F84A-04 (Commercial, Industrial, Extended) **PIC16F84A-20** (Commercial, Industrial, Extended)			**Standard Operating Conditions (unless otherwise stated)** Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended)				
Param No.	Symbol	Characteristic	Min	Typ†	Max	Units	Conditions
	VDD	**Supply Voltage**					
D001		16LF84A	2.0	—	5.5	V	XT, RC, and LP osc configuration
D001		16F84A	4.0	—	5.5	V	XT, RC and LP osc configuration
D001A			4.5	—	5.5	V	HS osc configuration
D002	VDR	**RAM Data Retention Voltage (Note 1)**	1.5	—	—	V	Device in SLEEP mode
D003	VPOR	**VDD Start Voltage** to ensure internal Power-on Reset signal	—	Vss	—	V	See section on Power-on Reset for details
D004	SVDD	**VDD Rise Rate** to ensure internal Power-on Reset signal	0.05	—	—	V/ms	
	IDD	**Supply Current (Note 2)**					
D010		16LF84A	—	1	4	mA	RC and XT osc configuration **(Note 4)** FOSC = 2.0 MHz, VDD = 5.5V
D010		16F84A	—	1.8	4.5	mA	RC and XT osc configuration **(Note 4)** FOSC = 4.0 MHz, VDD = 5.5V
D010A			—	3	10	mA	RC and XT osc configuration **(Note 4)** FOSC = 4.0 MHz, VDD = 5.5V (During FLASH programming)
D013			—	10	20	mA	HS osc configuration (PIC16F84A-20) FOSC = 20 MHz, VDD = 5.5V
D014		16LF84A	—	15	45	µA	LP osc configuration FOSC = 32 kHz, VDD = 2.0V, WDT disabled

Table 10.1: DC characteristics

Legend: Rows with standard voltage device data only are shaded for improved readability.

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

Note 1: This is the limit to which VDD can be lowered without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern, and temperature also have an impact on the current consumption. The test conditions for all IDD measurements in active operation mode are: OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD, T0CKI = VDD, MCLR = VDD; WDT enabled/disabled as specified.

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

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

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

10.1. DC Characteristics (Continued)

PIC16LF84A-04 (Commercial, Industrial)			Standard Operating Conditions (unless otherwise stated) Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended)				
PIC16F84A-04 (Commercial, Industrial, Extended) PIC16F84A-20 (Commercial, Industrial, Extended)			Standard Operating Conditions (unless otherwise stated) Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial) -40°C ≤ TA ≤ +125°C (extended)				
Param No.	Symbol	Characteristic	Min	Typ†	Max	Units	Conditions
	IPD	**Power-down Current (Note 3)**					
D020		16LF84A					
D020		16F84A-20 16F84A-04					
D021A		16LF84A	—	0.4	1.0	µA	VDD = 2.0V, WDT disabled, industrial
D021A		16F84A-20	—	1.5	3.5	µA	VDD = 4.5V, WDT disabled, industrial
		16F84A-04	—	1.0	3.0	µA	VDD = 4.0V, WDT disabled, industrial
D021B		16F84A-20	—	1.5	5.5	µA	VDD = 4.5V, WDT disabled, extended
		16F84A-04	—	1.0	5.0	µA	VDD = 4.0V, WDT disabled, extended
		Module Differential Current (Note 5)					
D022	ΔIWDT	Watchdog Timer	—	.20	16	µA	VDD = 2.0V, Industrial, Commercial
			—	3.5	20	µA	VDD = 4.0V, Commercial
			—	3.5	28	µA	VDD = 4.0V, Industrial, Extended
			—	4.8	25	µA	VDD = 4.5V, Commercial
			—	4.8	30	µA	VDD = 4.5V, Industrial, Extended

Table 10.2 : DC characteristics (continued)

Legend: Rows with standard voltage device data only are shaded for improved readability.

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

Note 1: This is the limit to which VDD can be lowered without losing RAM data.

2: The supply current is mainly a function of the operating voltage and frequency. Other factors, such as I/O pin loading and switching rate, oscillator type, internal code execution pattern,and temperature also have an impact on the current consumption. The test conditions for all IDD measurements in active operation mode are: OSC1 = external square wave, from rail-to-rail; all I/O pins tri-stated, pulled to VDD, T0CKI = VDD, MCLR = VDD; WDT enabled/disabled as specified.

3: The power-down current in SLEEP mode does not depend on the oscillator

type. Power-down current is measured with the part in SLEEP mode, with all I/O pins in hi-impedance state and tied to VDD and VSS.

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

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

10.2. DC Characteristics:

PIC16F84A-04 (Commercial, Industrial)

PIC16F84A-20 (Commercial, Industrial)

PIC16LF84A-04 (Commercial, Industrial)

DC Characteristics All Pins Except Power Supply Pins			Standard Operating Conditions (unless otherwise stated) Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial) Operating voltage VDD range as described in DC specifications (Section 9.1)				
Param No.	Symbol	Characteristic	Min	Typ†	Max	Units	Conditions
	VIL	**Input Low Voltage**					
		I/O ports:					
D030		with TTL buffer	VSS	—	0.8	V	4.5V ≤ VDD ≤ 5.5V **(Note 4)**
D030A			VSS	—	0.16VDD	V	Entire range **(Note 4)**
D031		with Schmitt Trigger buffer	VSS	—	0.2VDD	V	Entire range
D032		$\overline{MCLR}$, RA4/T0CKI	VSS	—	0.2VDD	V	
D033		OSC1 (XT, HS and LP modes)	VSS	—	0.3VDD	V	**(Note 1)**
D034		OSC1 (RC mode)	VSS	—	0.1VDD	V	
	VIH	**Input High Voltage**					
		I/O ports:		—			
D040		with TTL buffer	2.0	—	VDD	V	4.5V ≤ VDD ≤ 5.5V **(Note 4)**
D040A			0.25VDD+0.8	—	VDD	V	Entire range **(Note 4)**
D041		with Schmitt Trigger buffer	0.8 VDD	—	VDD		Entire range
D042		$\overline{MCLR}$,	0.8 VDD	—	VDD	V	
D042A		RA4/T0CKI	0.8 VDD	—	8.5	V	
D043		OSC1 (XT, HS and LP modes)	0.8 VDD	—	VDD	V	**(Note 1)**
D043A		OSC1 (RC mode)	0.9 VDD		VDD	V	
D050	VHYS	**Hysteresis of Schmitt Trigger Inputs**	—	0.1	—	V	
D070	IPURB	**PORTB Weak Pull-up Current**	50	250	400	µA	VDD = 5.0V, VPIN = VSS
	IIL	**Input Leakage Current (Notes 2, 3)**					
D060		I/O ports	—	—	±1	µA	VSS ≤ VPIN ≤ VDD, Pin at hi-impedance
D061		$\overline{MCLR}$, RA4/T0CKI	—	—	±5	µA	VSS ≤ VPIN ≤ VDD
D063		OSC1	—	—	±5	µA	VSS ≤ VPIN ≤ VDD, XT, HS and LP osc configuration

Table 10.3 : DC charecteristics

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

Note 1: In RC oscillator configuration, the OSC1 pin is a Schmitt Trigger input. Do not drive the PIC16F84A with an external clock while the device is in RC mode, or chip damage may result.

2: The leakage current on the MCLR pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

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

4: The user may choose the better of the two specs.

10.2. DC Characteristics: (Continued)

DC Characteristics All Pins Except Power Supply Pins			Standard Operating Conditions (unless otherwise stated) Operating temperature 0°C ≤ TA ≤ +70°C (commercial) -40°C ≤ TA ≤ +85°C (industrial) Operating voltage VDD range as described in DC specifications (Section 9.1)				
Param No.	Symbol	Characteristic	Min	Typ†	Max	Units	Conditions
	VOL	**Output Low Voltage**					
D080		I/O ports	—	—	0.6	V	IOL = 8.5 mA, VDD = 4.5V
D083		OSC2/CLKOUT	—	—	0.6	V	IOL = 1.6 mA, VDD = 4.5V, (RC mode only)
	VOH	**Output High Voltage**					
D090		I/O ports (Note 3)	VDD-0.7	—	—	V	IOH = -3.0 mA, VDD = 4.5V
D092		OSC2/CLKOUT (Note 3)	VDD-0.7	—	—	V	IOH = -1.3 mA, VDD = 4.5V (RC mode only)
	VOD	**Open Drain High Voltage**					
D150		RA4 pin	—	—	8.5	V	
		Capacitive Loading Specs on Output Pins					
D100	COSC2	OSC2 pin	—	—	15	pF	In XT, HS and LP modes when external clock is used to drive OSC1
D101	CIO	All I/O pins and OSC2 (RC mode)	—	—	50	pF	
		Data EEPROM Memory					
D120	ED	Endurance	1M	10M	—	E/W	25°C at 5V
D121	VDRW	VDD for read/write	VMIN	—	5.5	V	VMIN = Minimum operating voltage
D122	TDEW	Erase/Write cycle time	—	4	8	ms	
		Program FLASH Memory					
D130	EP	Endurance	1000	10K	—	E/W	
D131	VPR	VDD for read	VMIN	—	5.5	V	VMIN = Minimum operating voltage
D132	VPEW	VDD for erase/write	4.5	—	5.5	V	
D133	TPEW	Erase/Write cycle time	—	4	8	ms	

Table 10.4 : DC characteristics

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

Note 1: In RC oscillator configuration, the OSC1 pin is a Schmitt Trigger input. Do not drive the PIC16F84A with an external clock while the device is in RC mode, or chip damage may result.

2: The leakage current on the MCLR pin is strongly dependent on the applied voltage level. The specified levels represent normal operating conditions. Higher leakage current may be measured at different input voltages.

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

4: The user may choose the better of the two specs.

10.3. AC (Timing) Characteristics

10.3.1. Timing Parameter Symbology

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

1. TppS2ppS
2. TppS

T			
	F Frequency		T Time
Lowercase letters (pp) and their meanings:			
pp			
	2 to		os, osc OSC1
	ck CLKOUT		ost oscillator start-up timer
	cy cycle time		pwrt power-up timer
	io I/O port		rbt RBx pins
	inp INT pin		t0 T0CKI
	mp $\overline{\text{MCLR}}$		wdt watchdog timer
Uppercase letters and their meanings:			
S			
	F Fall		P Period
	H High		R Rise
	I Invalid (high impedance)		V Valid
	L Low		Z High Impedance

Table 10.5 : Timing parameter symbology

10.3.2. Timing Conditions

The temperature and voltages specified in **Table 10.6** apply to all timing specifications unless otherwise noted. All timings are measured between high and low measurement points as indicated in **Figure 10.4**. **Figure 10.5** specifies the load conditions for the timing specifications.

AC CHARACTERISTICS	Standard Operating Conditions (unless otherwise stated)
	Operating temperature 0°C ≤ TA ≤ +70°C for commercial
	-40°C ≤ TA ≤ +85°C for industrial
	Operating voltage VDD range as described in DC specifications (Section 9.1)

Table 10.6 Temperature and voltage specifications -AC

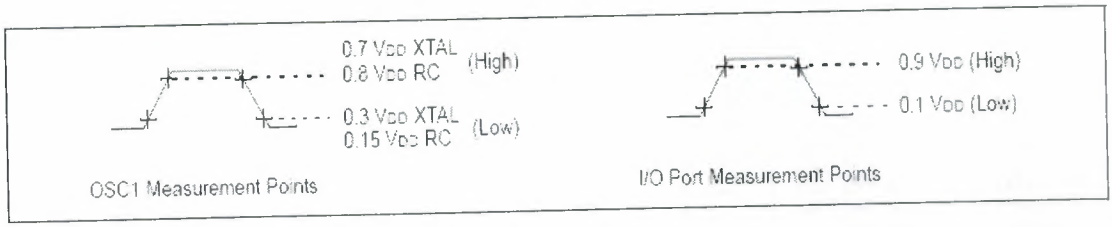


Figure 10.4 Parameter measurement information

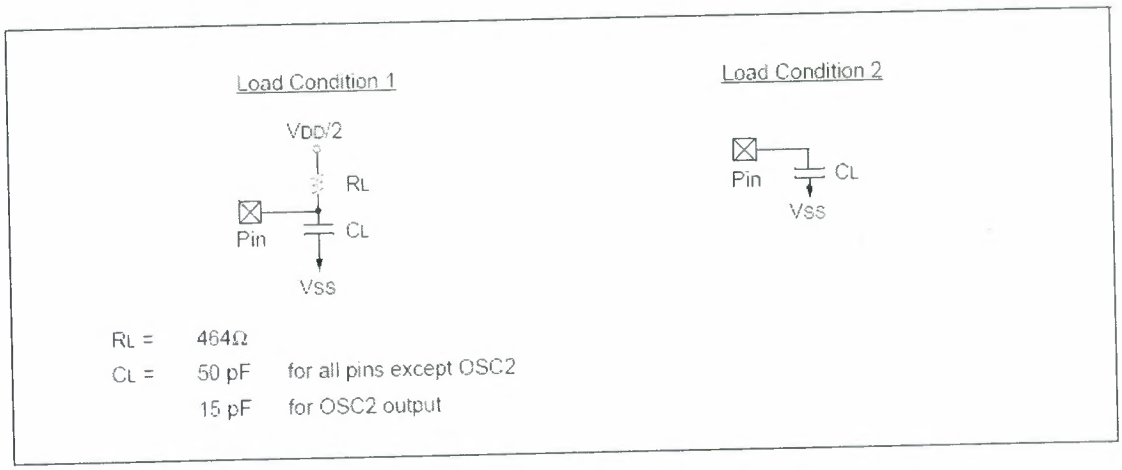


Figure 10.5 Load conditions

10.3.3. Timing Diagrams and Specifications

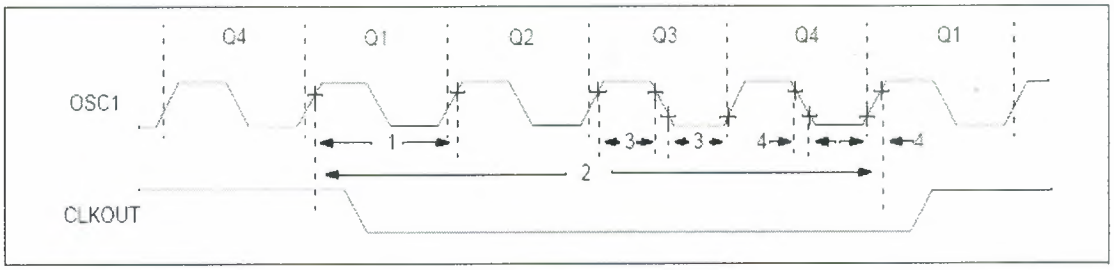


Figure 10.6 External clock timing

Param No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
	FOSC	External CLKIN Frequency(1)	DC	—	2	MHz	XT, RC osc (-04, LF)
			DC	—	4	MHz	XT, RC osc (-04)
			DC	—	20	MHz	HS osc (-20)
			DC	—	200	kHz	LP osc (-04, LF)
		Oscillator Frequency(1)	DC	—	2	MHz	RC osc (-04, LF)
			DC	—	4	MHz	RC osc (-04)
			0.1	—	2	MHz	XT osc (-04, LF)
			0.1	—	4	MHz	XT osc (-04)
			1.0	—	20	MHz	HS osc (-20)
			DC	—	200	kHz	LP osc (-04, LF)
1	TOSC	External CLKIN Period(1)	500	—	—	ns	XT, RC osc (-04, LF)
			250	—	—	ns	XT, RC osc (-04)
			50	—	—	ns	HS osc (-20)
			5.0	—	—	µs	LP osc (-04, LF)
		Oscillator Period(1)	500	—	—	ns	RC osc (-04, LF)
			250	—	—	ns	RC osc (-04)
			500	—	10,000	ns	XT osc (-04, LF)
			250	—	10,000	ns	XT osc (-04)
			50	—	1,000	ns	HS osc (-20)
			5.0	—	—	µs	LP osc (-04, LF)
2	TCY	Instruction Cycle Time(1)	0.2	4/FOSC	DC	µs	
3	TOSL, TOSH	Clock in (OSC1) High or Low Time	80	—	—	ns	XT osc (-04, LF)
			50	—	—	ns	XT osc (-04)
			2.0	—	—	µs	LP osc (-04, LF)
			17.5	—	—	ns	HS osc (-20)
4	TOSR, TOSF	Clock in (OSC1) Rise or Fall Time	25	—	—	ns	XT osc (-04)
			50	—	—	ns	LP osc (-04, LF)
			7.5	—	—	ns	HS osc (-20)

Table 10.7 External clock timing requirement

† Data in "Typ" column is at 5.0V, 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 pin. When an external clock input is used, the "Max." cycle time limit is "DC" (no clock) for all devices.

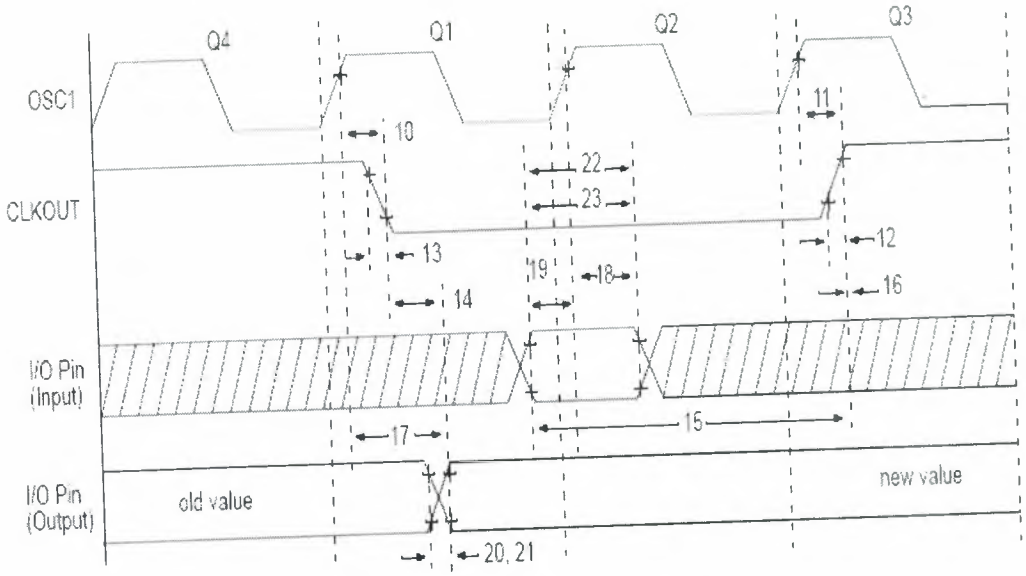


Figure 10.7 CLKOUT and I/0 timing

Note: All tests must be done with specified capacitive loads (**Figure 10.5**) 50 pF on I/O pins and CLKOUT.

Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
10 10A	TosH2ckL	OSC1↑ to CLKOUT↓	Standard	—	15	30	ns	(Note 1)
			Extended (LF)	—	15	120	ns	(Note 1)
11 11A	TosH2ckH	OSC1↑ to CLKOUT↑	Standard	—	15	30	ns	(Note 1)
			Extended (LF)	—	15	120	ns	(Note 1)
12 12A	TckR	CLKOUT rise time	Standard	—	15	30	ns	(Note 1)
			Extended (LF)	—	15	100	ns	(Note 1)
13 13A	TckF	CLKOUT fall time	Standard	—	15	30	ns	(Note 1)
			Extended (LF)	—	15	100	ns	(Note 1)
14	TckL2ioV	CLKOUT ↓ to Port out valid		—	—	0.5Tcy +20	ns	(Note 1)
15	TioV2ckH	Port in valid before CLKOUT ↑	Standard	0.30Tcy + 30	—	—	ns	(Note 1)
			Extended (LF)	0.30Tcy + 80	—	—	ns	(Note 1)
16	TckH2iol	Port in hold after CLKOUT ↑		0	—	—	ns	(Note 1)
17	TosH2ioV	OSC1↑ (Q1 cycle) to Port out valid	Standard	—	—	125	ns	
			Extended (LF)	—	—	250	ns	
18	TosH2iol	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	Standard	10	—	—	ns	
			Extended (LF)	10	—	—	ns	
19	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)	Standard	-75	—	—	ns	
			Extended (LF)	-175	—	—	ns	
20 20A	TioR	Port output rise time	Standard	—	10	35	ns	
			Extended (LF)	—	10	70	ns	
21 21A	TioF	Port output fall time	Standard	—	10	35	ns	
			Extended (LF)	—	10	70	ns	
22 22A	TINP	INT pin high or low time	Standard	20	—	—	ns	
			Extended (LF)	55	—	—	ns	
23 23A	TRBP	RB7:RB4 change INT high or low time	Standard	Tosc§	—	—	ns	
			Extended (LF)	Tosc§	—	—	ns	

Table 10.8 CLKOUT and I/0 timing requirements

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

§ By design.

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

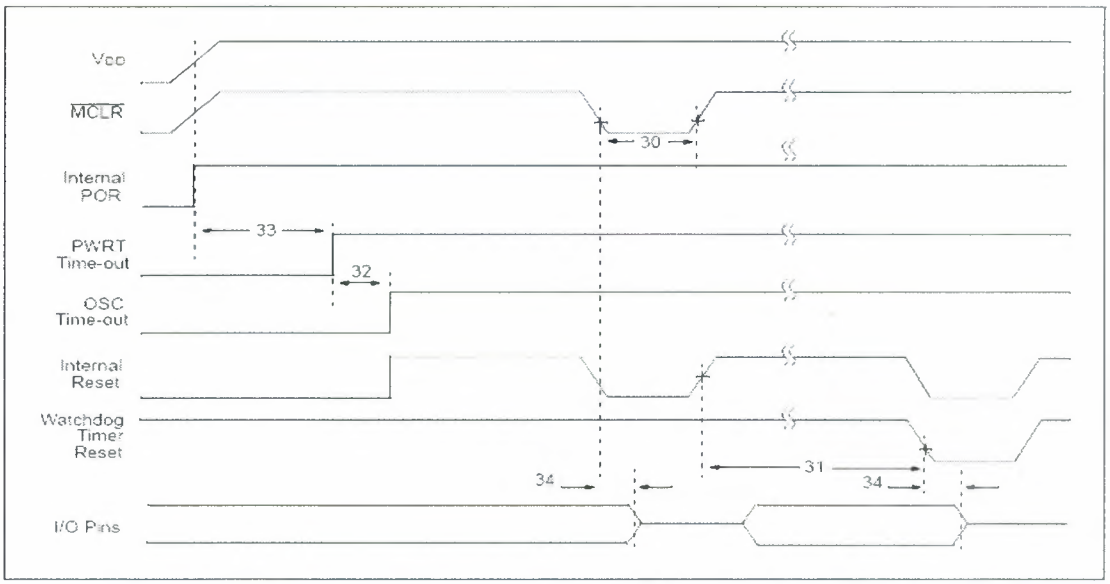


Figure 10.8 Reset, WATCHDOG timer, oscillator start-up timer and power-up timer timing

Parameter No.	Sym	Characteristic	Min	Typ†	Max	Units	Conditions
30	TmcL	$\overline{MCLR}$ Pulse Width (low)	2	—	—	μs	VDD = 5.0V
31	TWDT	Watchdog Timer Time-out Period (No Prescaler)	7	18	33	ms	VDD = 5.0V
32	TOST	Oscillation Start-up Timer Period		1024Tosc		ms	Tosc = OSC1 period
33	TPWRT	Power-up Timer Period	28	72	132	ms	VDD = 5.0V
34	TIOZ	I/O hi-impedance from $\overline{MCLR}$ Low or RESET	—	—	100	ns	

Table 10.9 Reset, WATCHDOG timer, oscillator start-up timer and power-up timer requirements

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

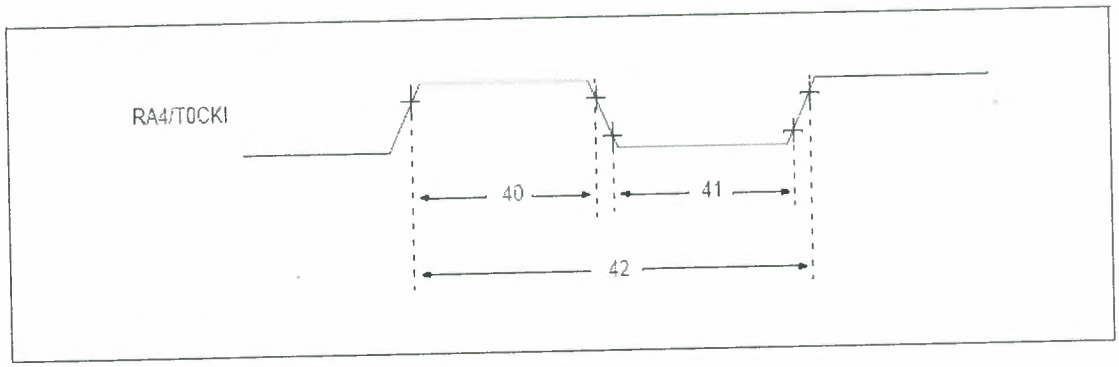


Figure 10.9 TIMER0 clock timing

Parameter No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
40	TtOH	T0CKI High Pulse Width	No Prescaler	0.5TCY + 20	—	—	ns	
			With Prescaler	50	—	—	ns	2.0V ≤ VDD ≤ 3.0V
				30	—	—	ns	3.0V ≤ VDD ≤ 6.0V
41	TtOL	T0CKI Low Pulse Width	No Prescaler	0.5TCY + 20	—	—	ns	
			With Prescaler	50	—	—	ns	2.0V ≤ VDD ≤ 3.0V
				20	—	—	ns	3.0V ≤ VDD ≤ 6.0V
42	TtOP	T0CKI Period		TCY + 40 / N	—	—	ns	N = prescale value (2, 4, ..., 256)

Table 10.10 TIMER0 clock requirements

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

11. DC/AC CHARACTERISTIC GRAPHS

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

In some graphs, the data presented are outside specified operating range (i.e., outside specified VDD range). This is for information only and devices are ensured 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 and matrix samples. 'Typical' represents the mean of the distribution at 25°C. 'Max' or 'Min' represents (mean + 3s) or (mean -3s), respectively, where s is a standard deviation over the whole temperature range.

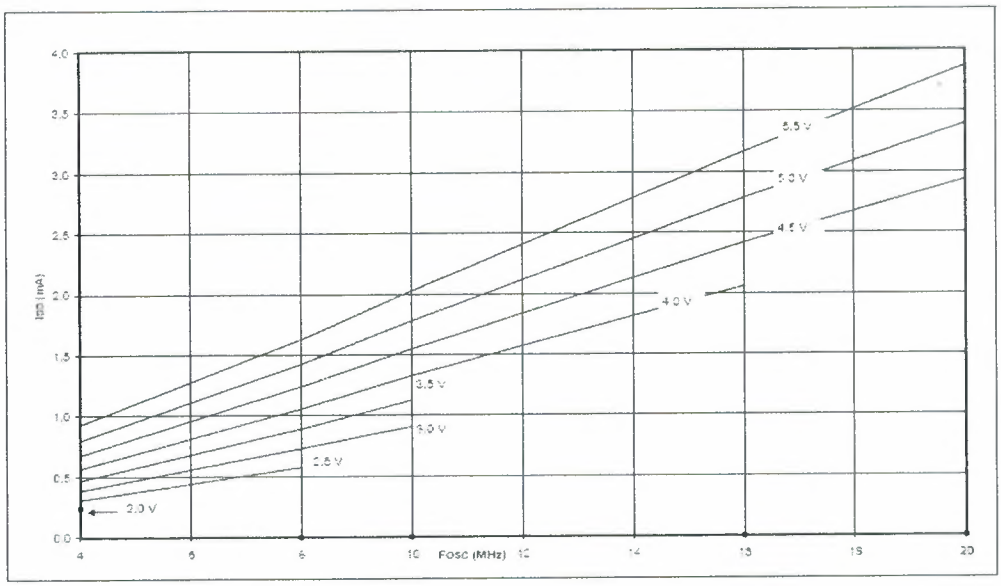


Figure 11.1 Typical IDD vs. FOSC over VDD (HS mode, 25°C)

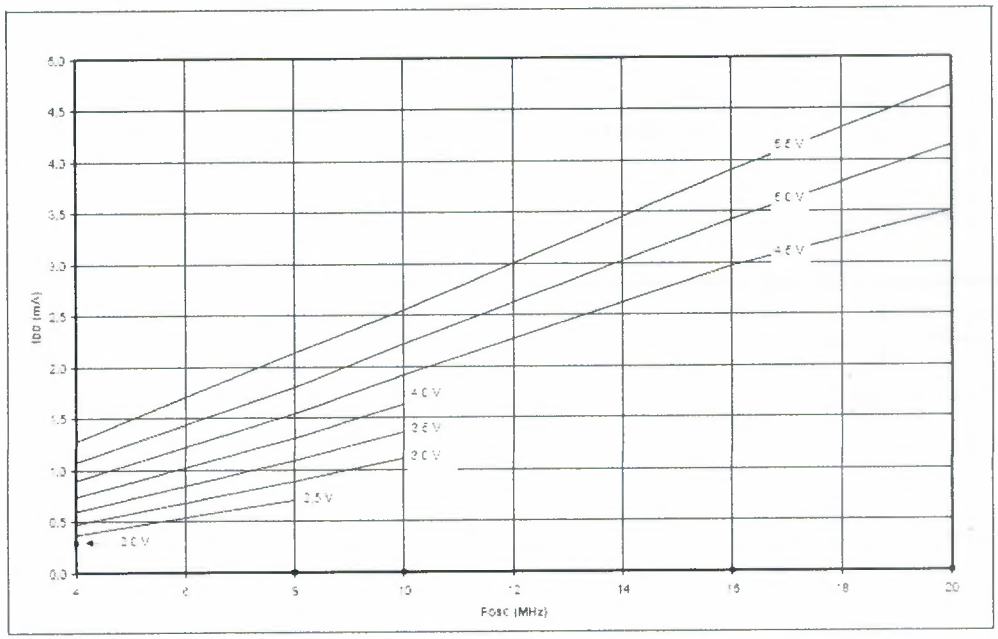


Figure 11.2 Maximum IDD vs. FOSC over VDD (HS mode, -40° to +125°C)

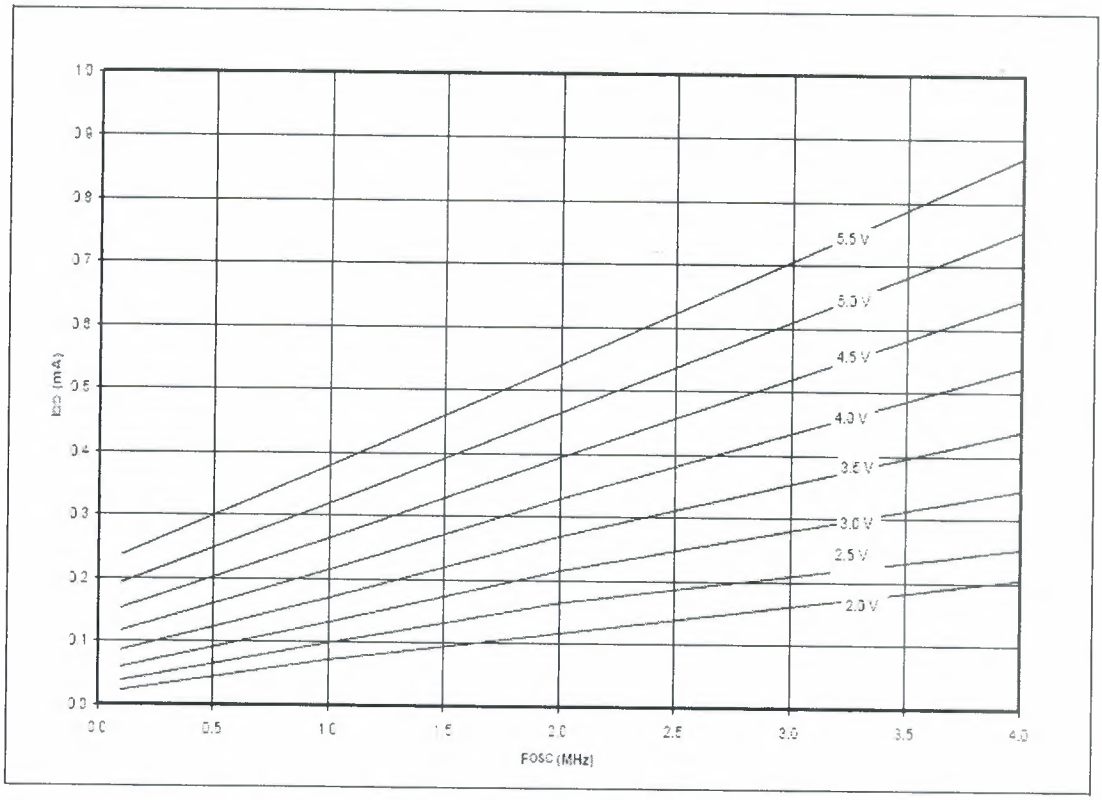


Figure 11.3 Typical IDD vs. FOSC over VDD (XT mode, 25°C)

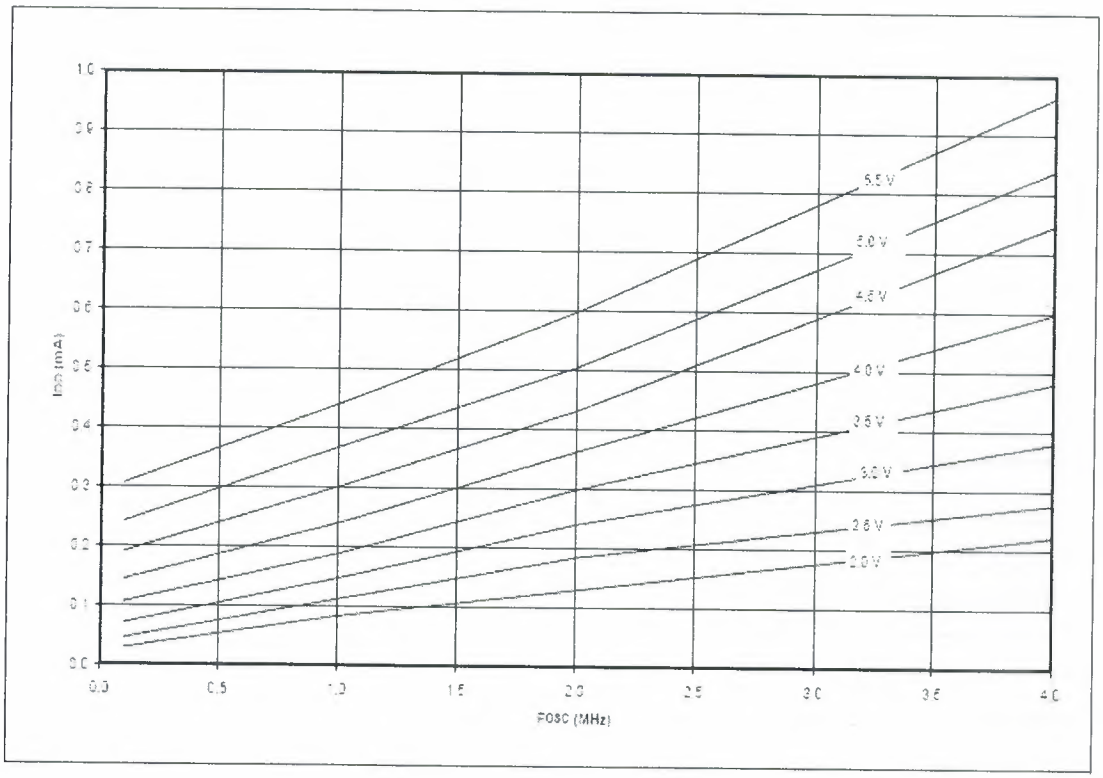


Figure 11.4 Maximum IDD vs. FOSC over VDD (XT mode, -40° to +125°C)

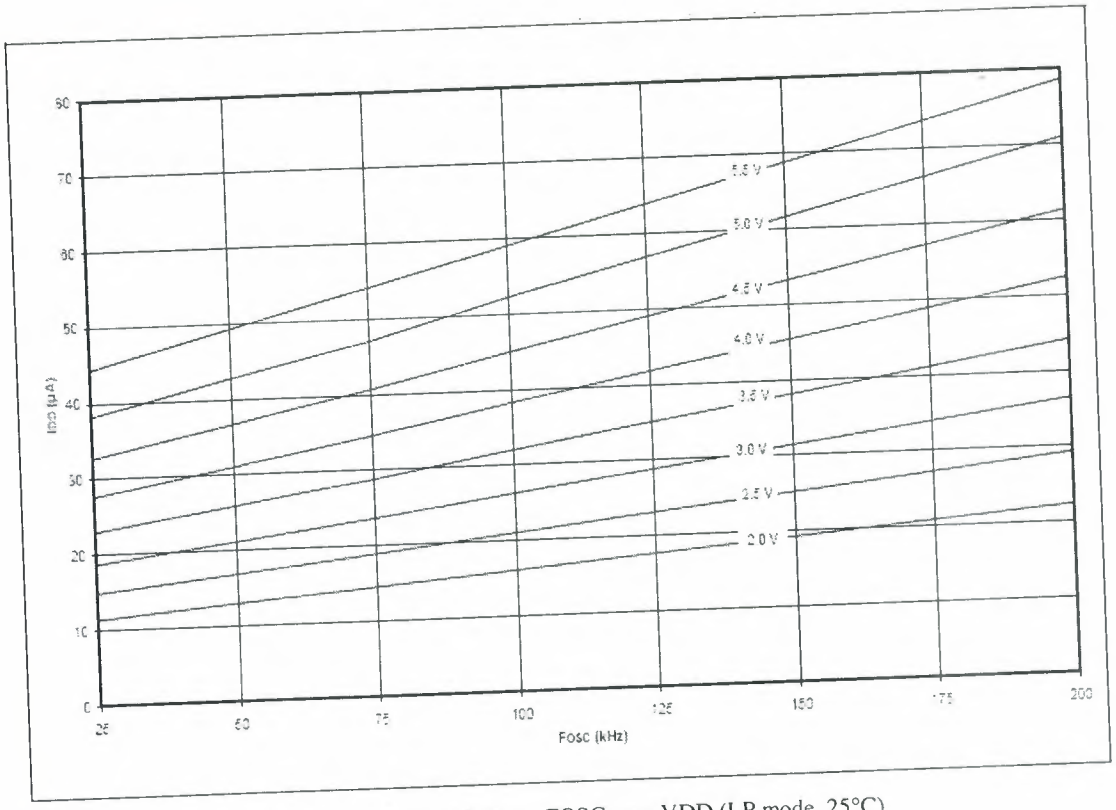


Figure 11.5 Typical IDD vs. FOSC over VDD (LP mode, 25°C)

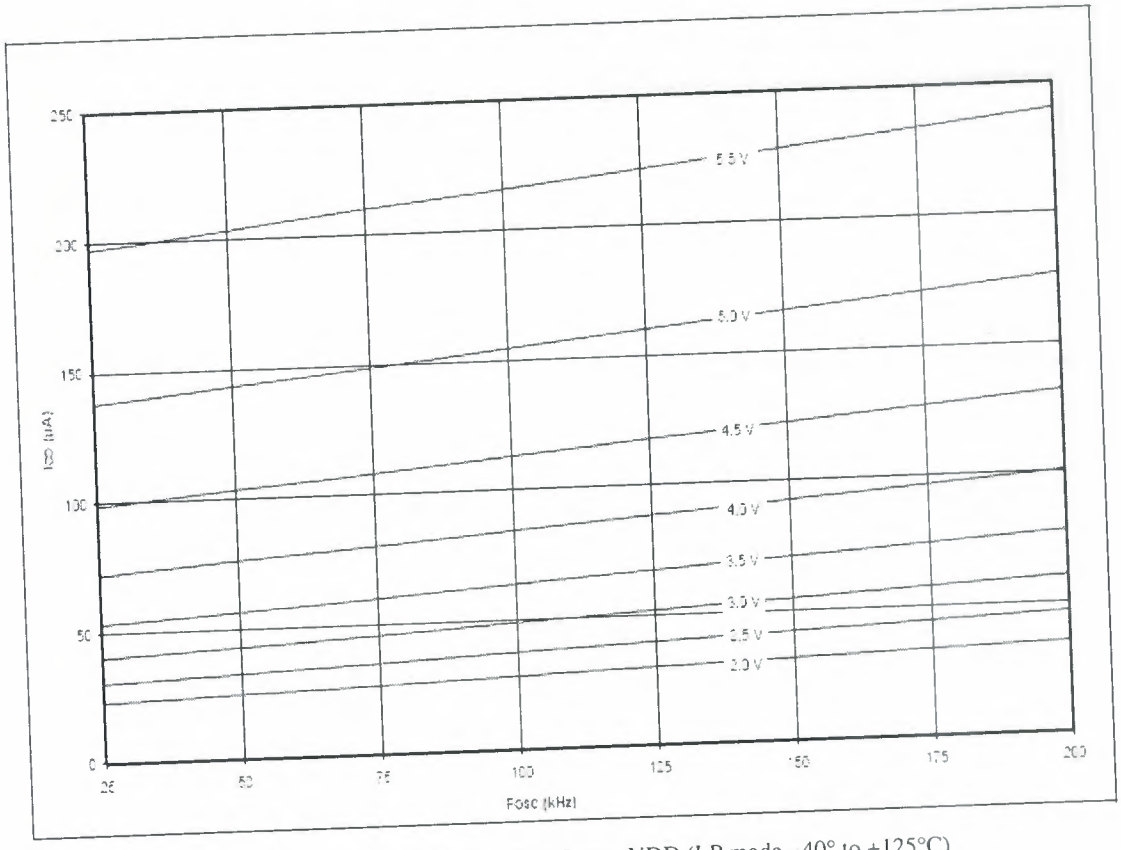


Figure 11.6 Maximum IDD vs. FOSC over VDD (LP mode, -40° to +125°C)

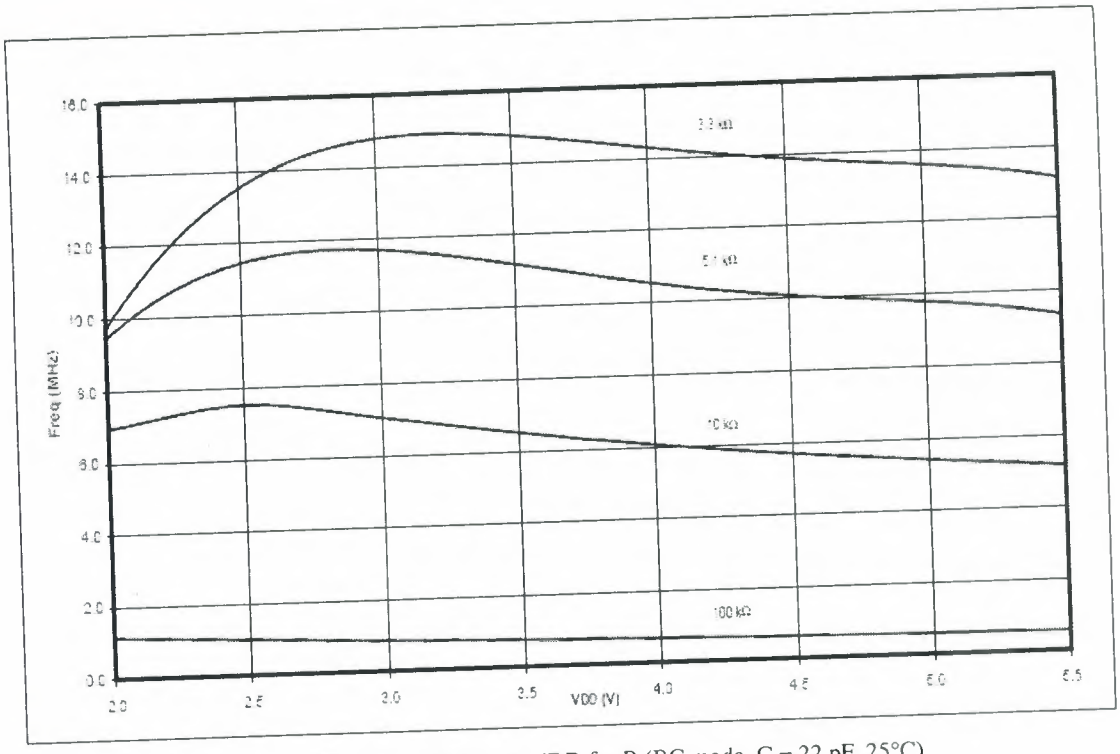


Figure 11.7 Average FOSC vs. VDD for R (RC mode, C = 22 pF, 25°C)

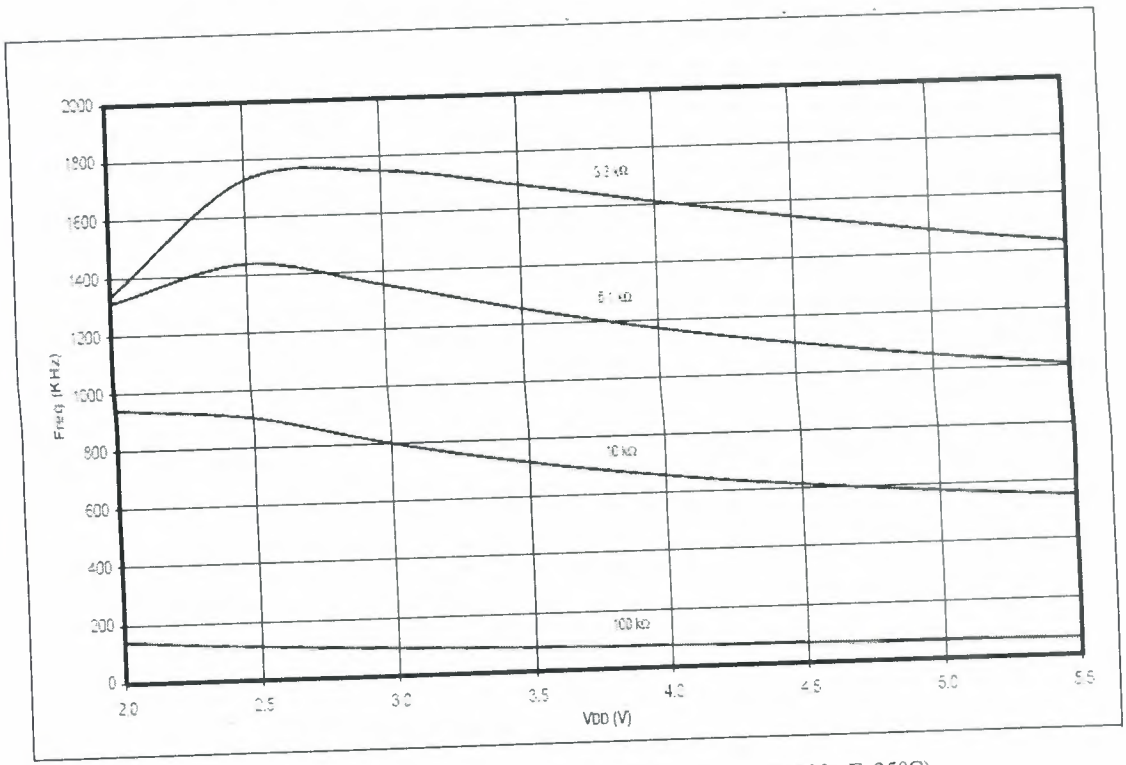


Figure 11.8 Average FOSC vs. VDD for R (RC mode, C=100 pF, 25°C)

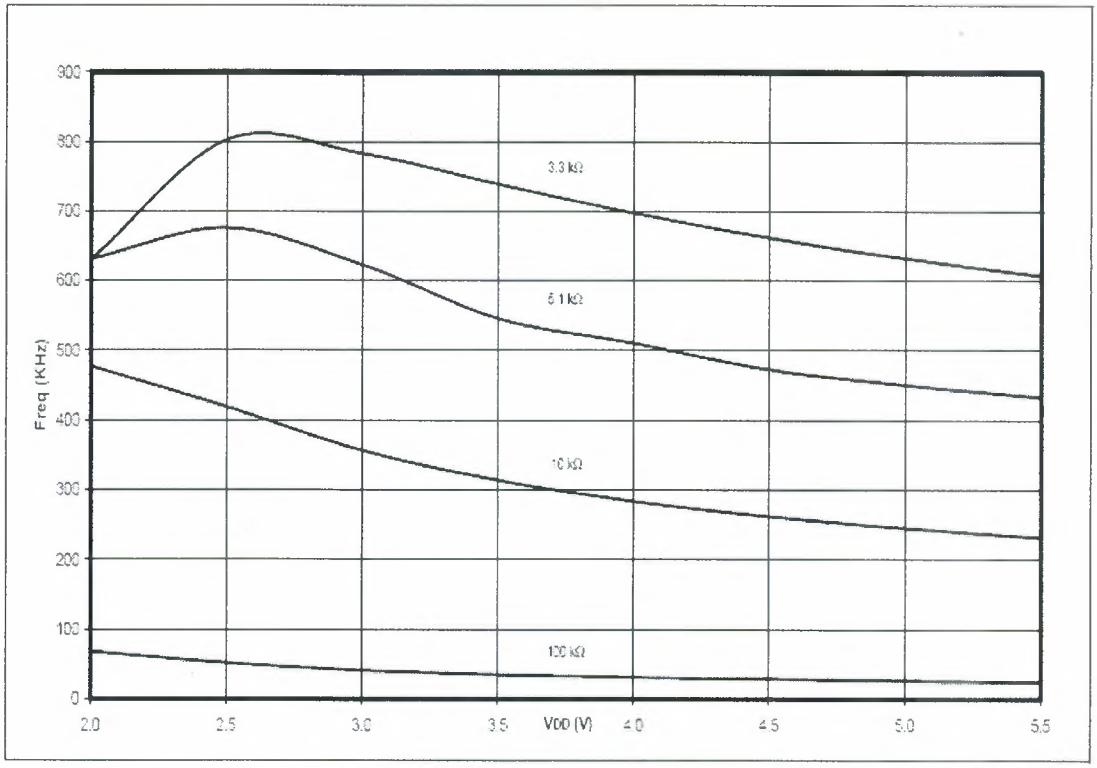


Figure 11.9 Average FOSC vs. VDD for R (RC mode, C = 300 pF, 25°C)

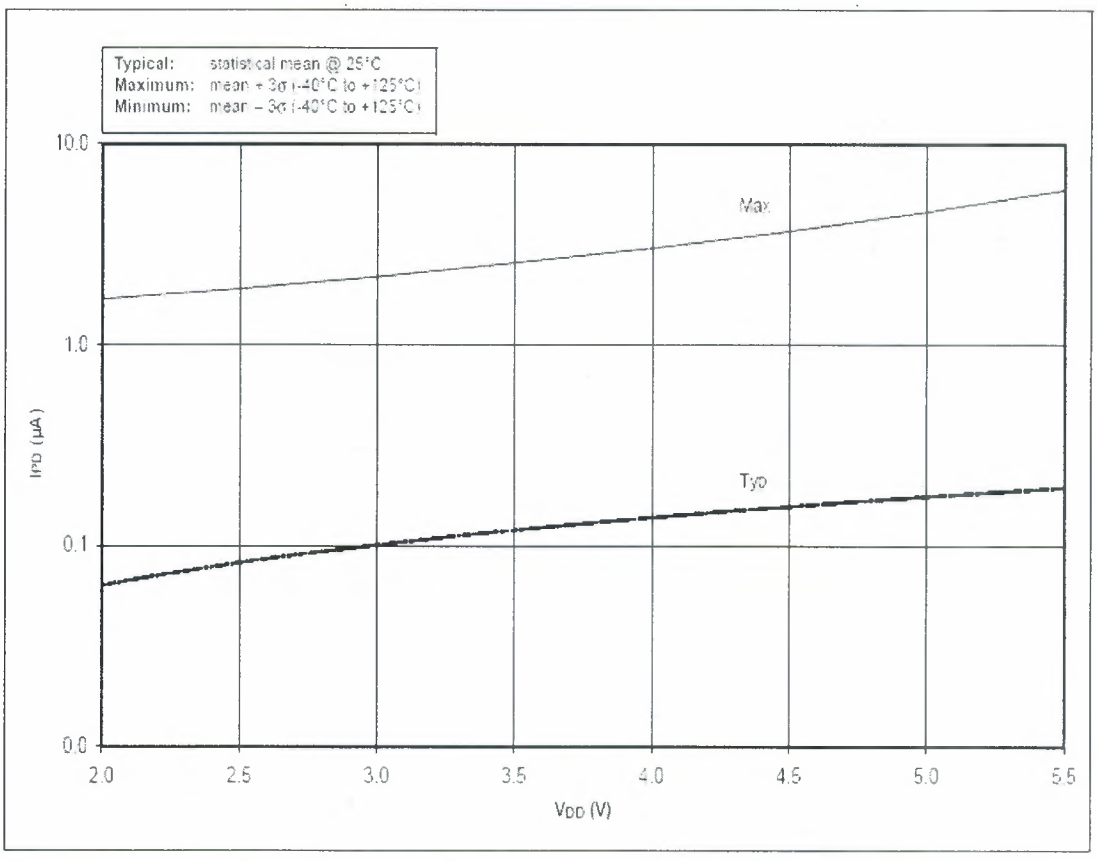


Figure 11.10 IPD vs. VDD (sleep mode, all peripherals disabled)

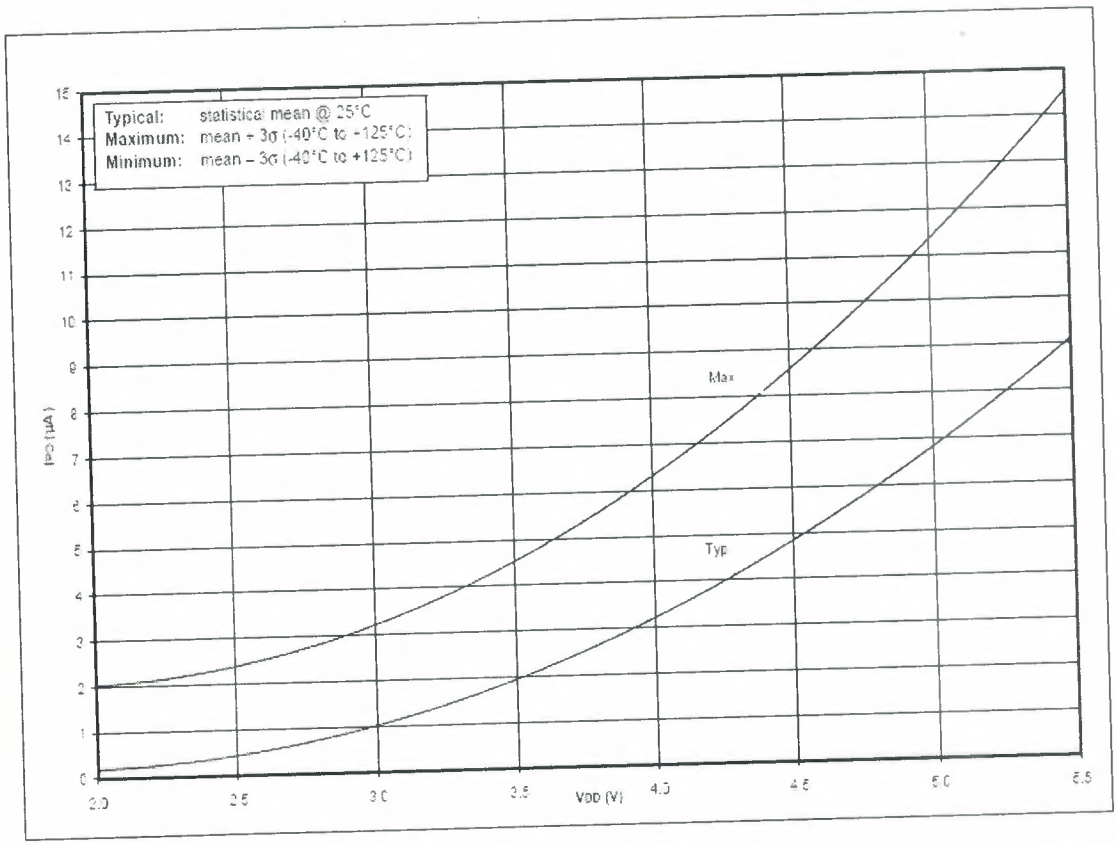


Figure 11.11 IPD vs. VDD (WDT mode)

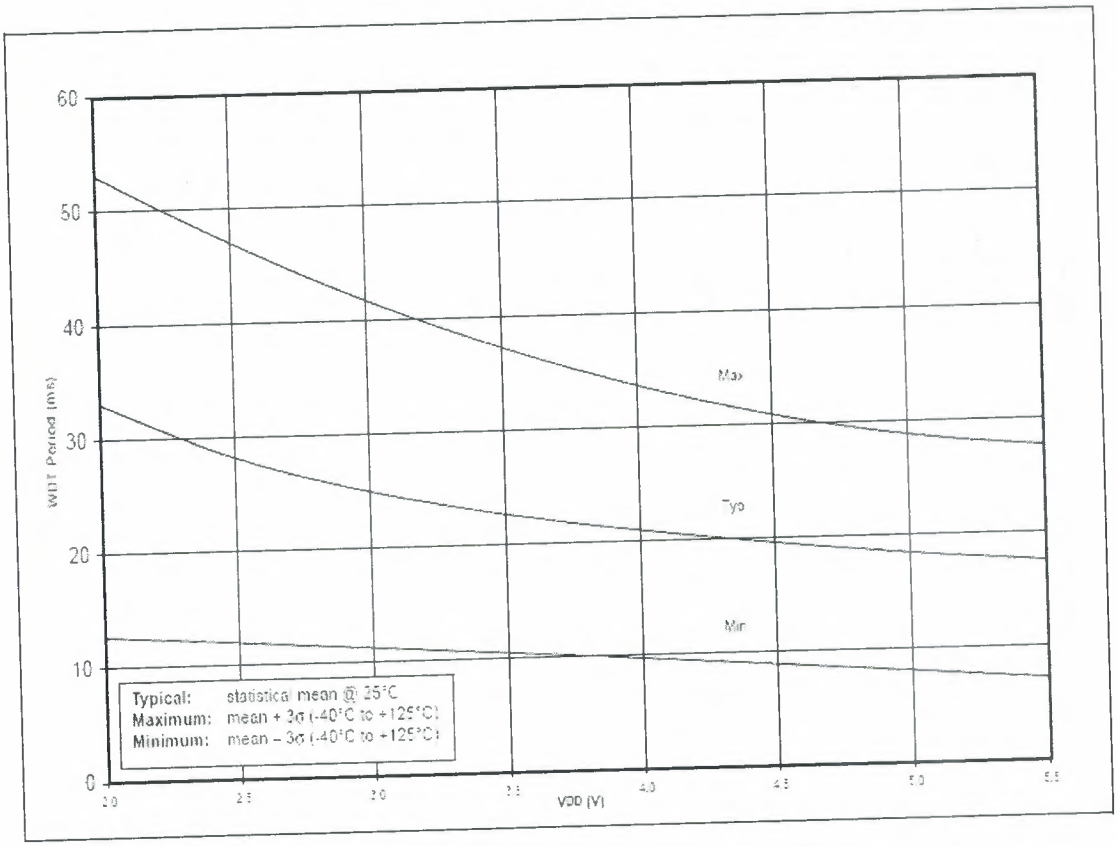


Figure 11.12 Typical, minimum, and maximum WDT period vs. VDD over temp.

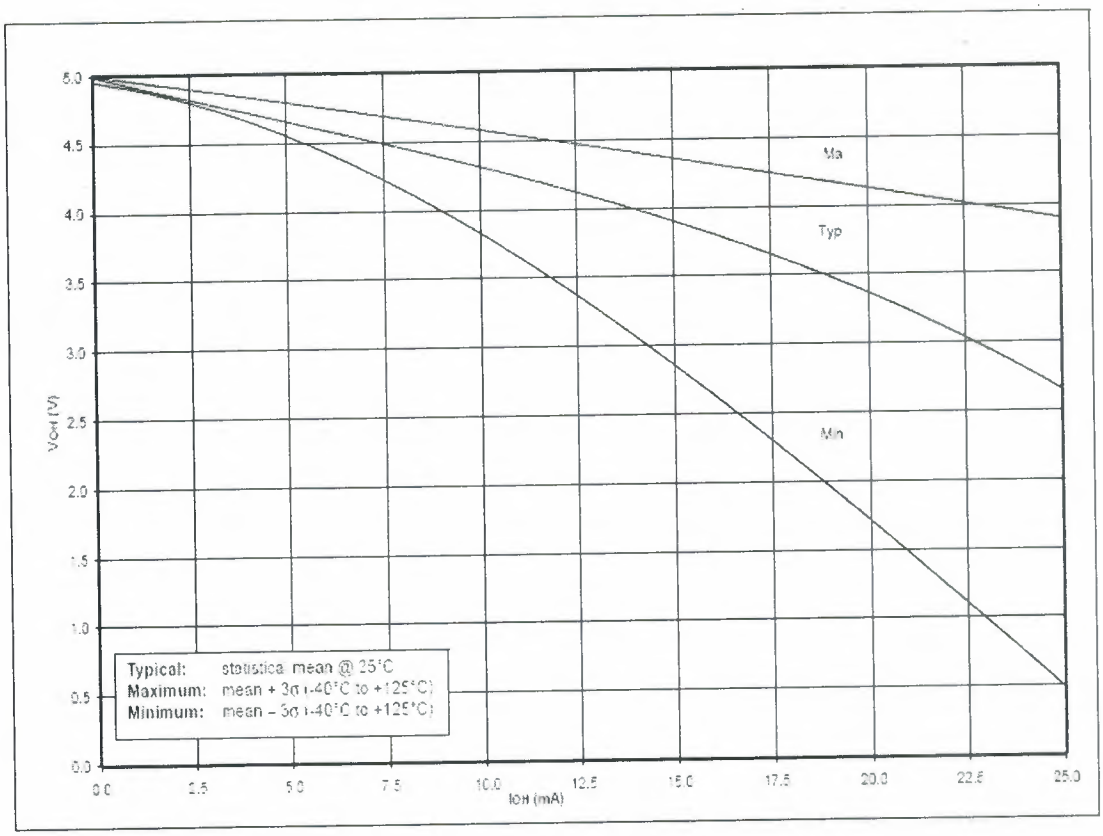


Figure 11.13 Typical, minimum and maximum VOH vs. IOH (VDD = 5V, -40°C to +125°C)

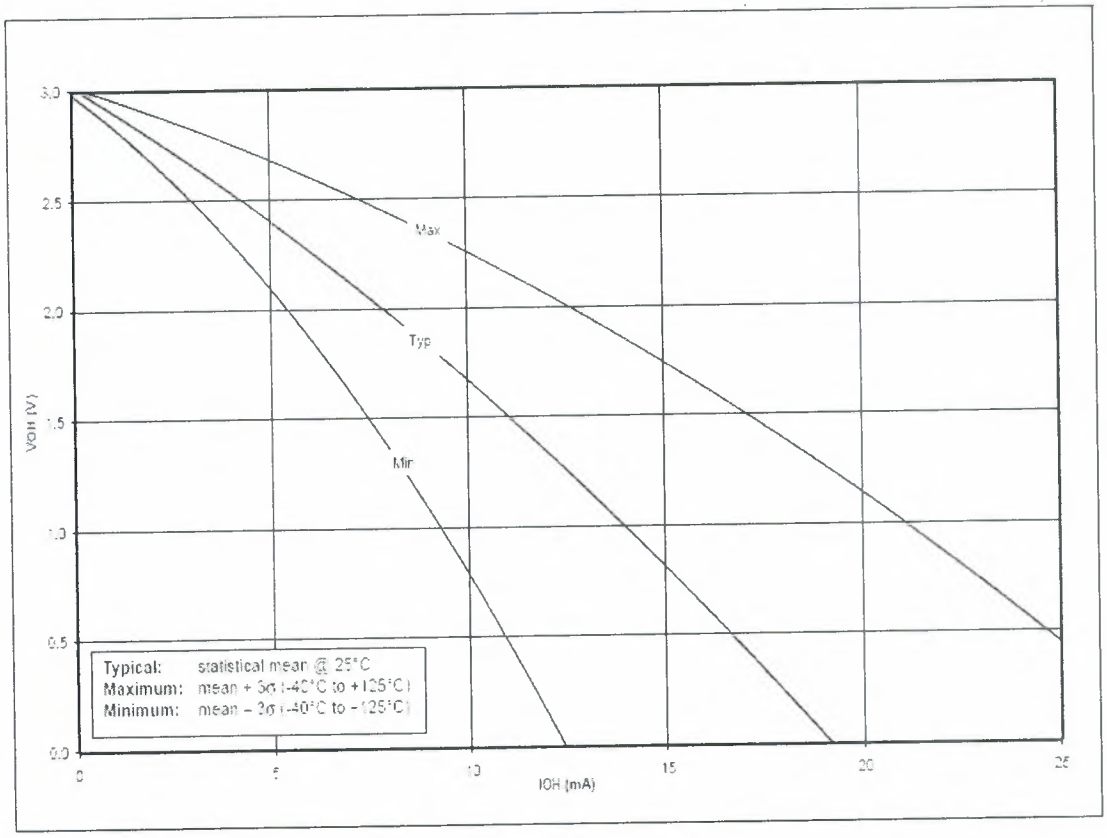


Figure 11.14 Typical, minimum and maximum VOH vs. IOH (VDD = 3V, -40°C to +125°C)

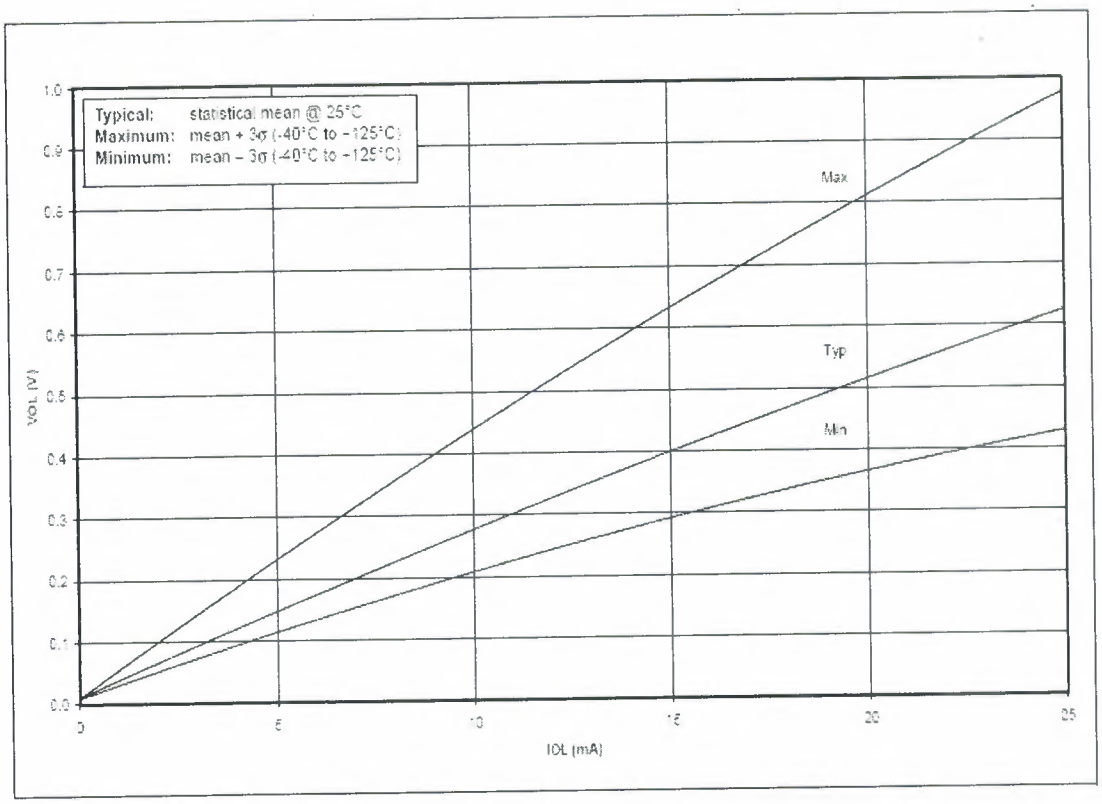


Figure 11.15 Typical, minimum and maximum VOL vs. IOL (VDD = 5V, -40°C to +125°C)

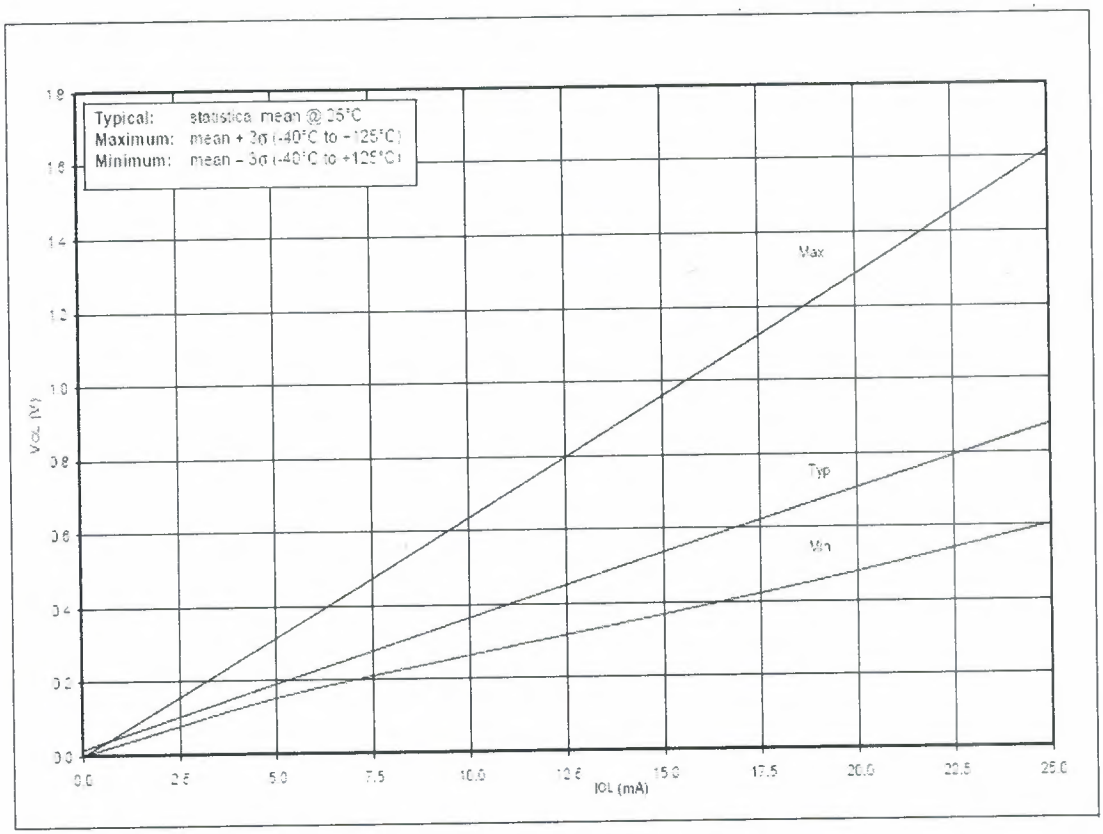


Figure 11.16 Typical, minimum and maximum VOL vs. IOL (VDD = 3V, -40°C to +125°C)

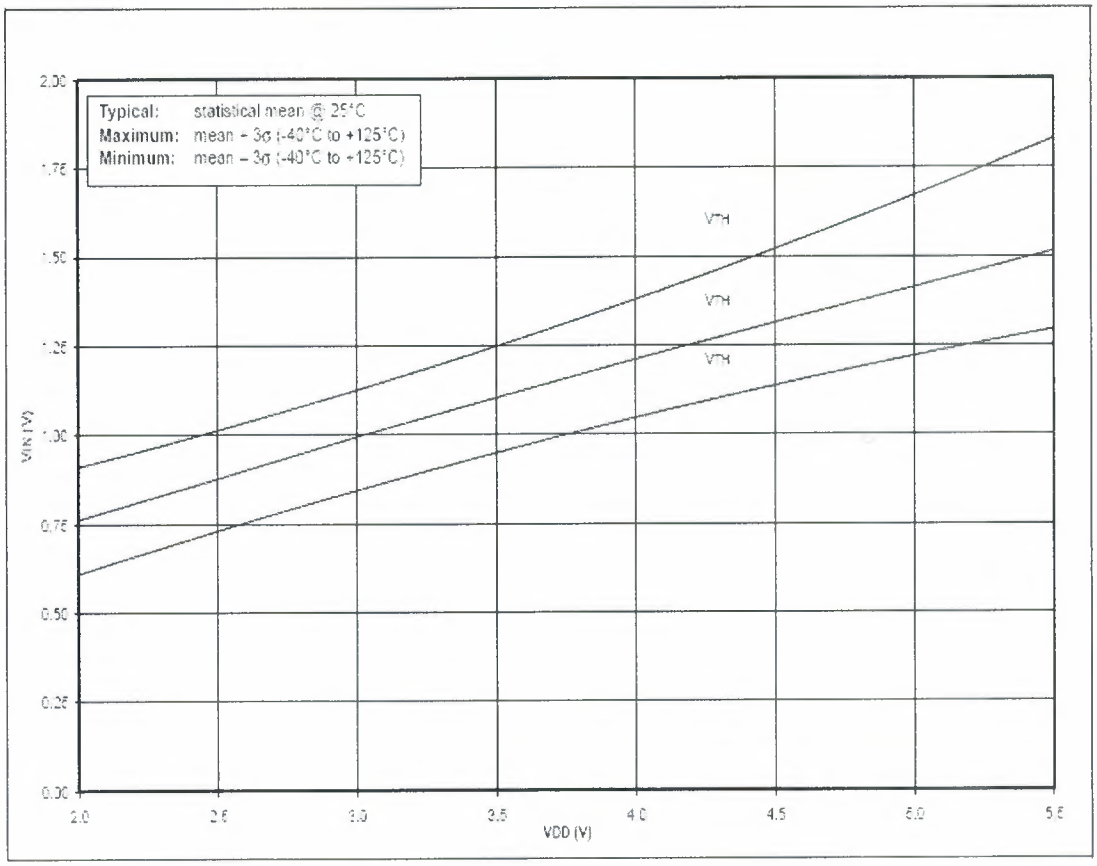


Figure 11.17 Minimum and maximum VIN vs. VDD, (TTL input, -40°C to +125°C)

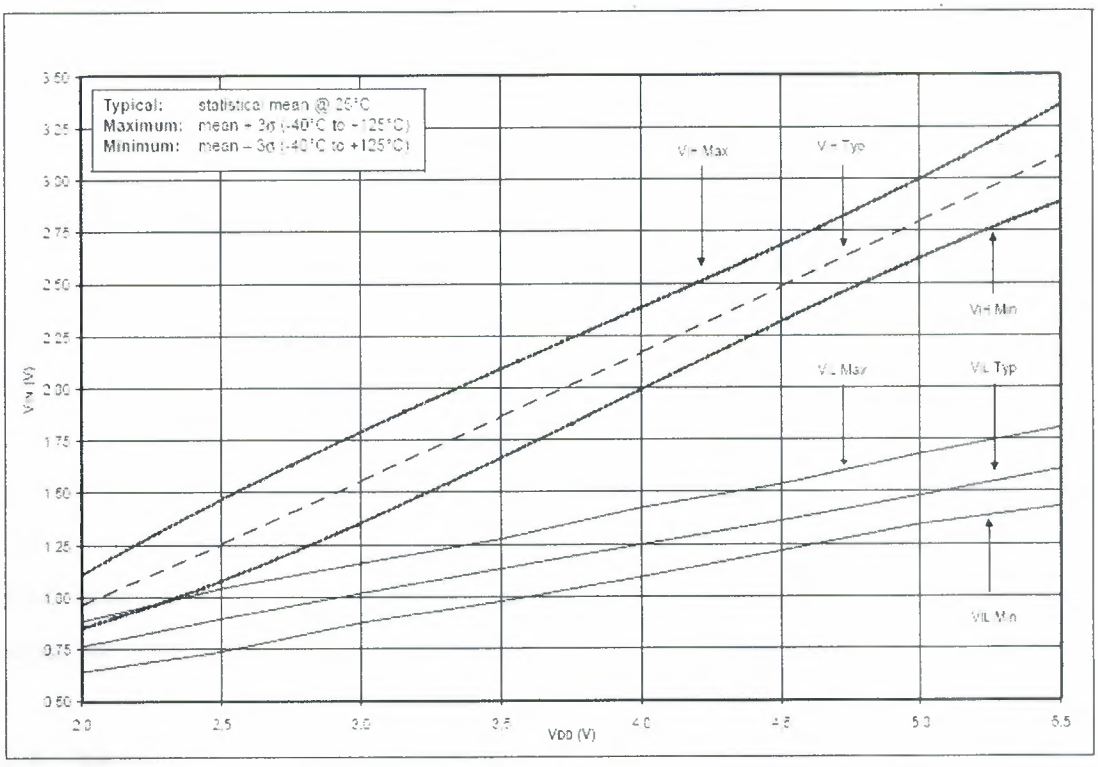


Figure 11.18 Minimum and maximum VIN vs. VDD (ST input, -40°C to +125°C)

12. PACKAGING INFORMATION

12.1. Package Marking Information

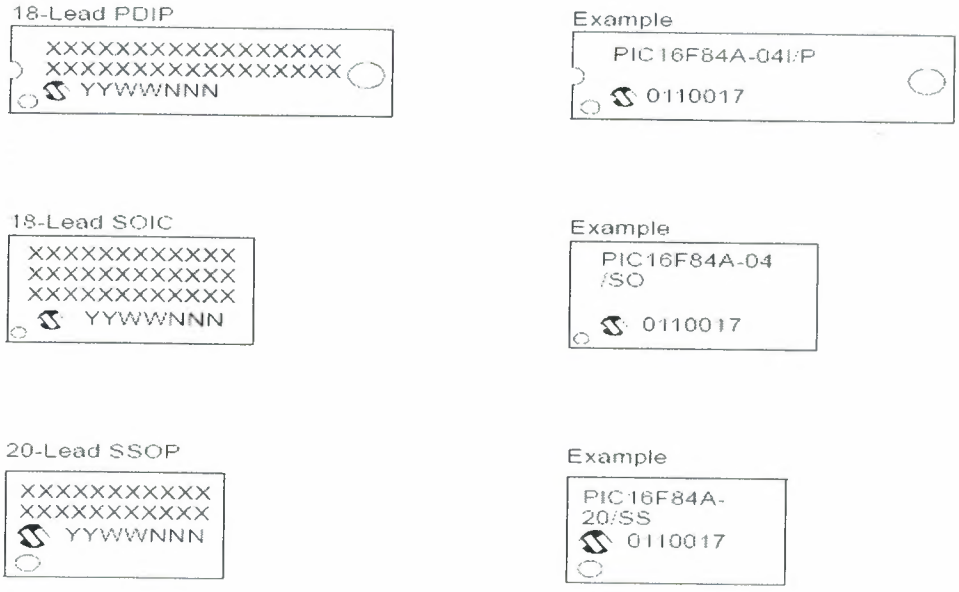


Figure 12.1 :Package marking information

Legend: XX...X Customer specific information*

Y Year code (last digit of calendar year)

YY Year code (last 2 digits of calendar year)

WW Week code (week of January 1 is week '01')

NNN Alphanumeric traceability code

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 PICmicro device marking consists of Microchip part number, year code, week code, and traceability code. For PICmicro device 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.

12.2. 18-Lead Plastic Dual In-line (P) – 300 mil (PDIP)

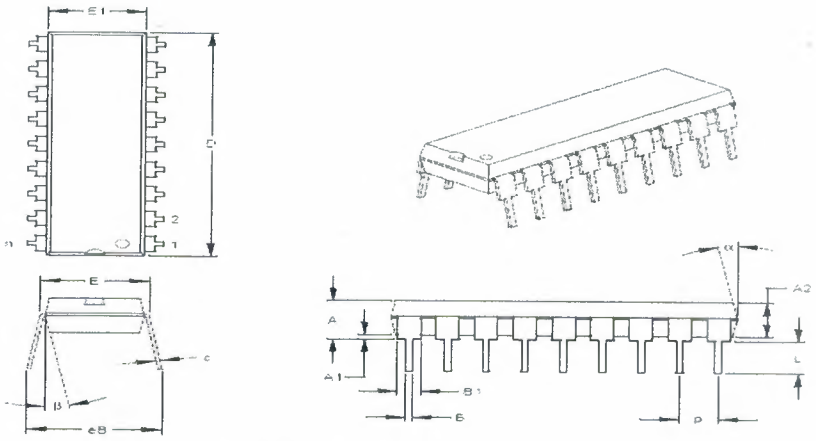


Figure 12.2 : 18-Lead Plastic Dual In-line (P) – 300 mil (PDIP)

Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		18			18	
Pitch	p		.100			2.54	
Top to Seating Plane	A	.140	.155	.170	3.56	3.94	4.32
Molded Package Thickness	A2	.115	.130	.145	2.92	3.30	3.68
Base to Seating Plane	A1	.015			0.38		
Shoulder to Shoulder Width	E	.300	.313	.325	7.62	7.94	8.26
Molded Package Width	E1	.240	.250	.260	6.10	6.35	6.60
Overall Length	D	.890	.898	.905	22.61	22.80	22.99
Tip to Seating Plane	L	.125	.130	.135	3.18	3.30	3.43
Lead Thickness	c	.008	.012	.015	0.20	0.29	0.38
Upper Lead Width	B1	.045	.058	.070	1.14	1.46	1.78
Lower Lead Width	B	.014	.018	.022	0.36	0.46	0.56
Overall Row Spacing §	eB	.310	.370	.430	7.87	9.40	10.92
Mold Draft Angle Top	α	5	10	15	5	10	15
Mold Draft Angle Bottom	β	5	10	15	5	10	15

Table 12.1 : Parameters

* Controlling Parameter

§ Significant Characteristic

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

12.3. 18-Lead Plastic Small Outline (SO) – Wide, 300 mil (SOIC)

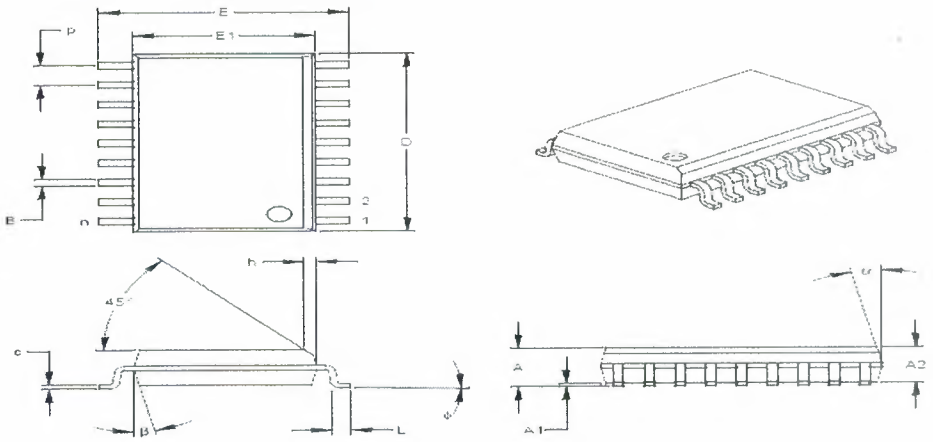


Figure 12.3 : 18-Lead Plastic Small Outline (SO) – Wide, 300 mil (SOIC)

Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		18			18	
Pitch	p		.050			1.27	
Overall Height	A	.093	.099	.104	2.36	2.50	2.64
Molded Package Thickness	A2	.088	.091	.094	2.24	2.31	2.39
Standoff §	A1	.004	.008	.012	0.10	0.20	0.30
Overall Width	E	.394	.407	.420	10.01	10.34	10.67
Molded Package Width	E1	.291	.295	.299	7.39	7.49	7.59
Overall Length	D	.446	.454	.462	11.33	11.53	11.73
Chamfer Distance	h	.010	.020	.029	0.25	0.50	0.74
Foot Length	L	.016	.033	.050	0.41	0.84	1.27
Foot Angle	ϕ	0	4	8	0	4	8
Lead Thickness	c	.009	.011	.012	0.23	0.27	0.30
Lead Width	B	.014	.017	.020	0.36	0.42	0.51
Mold Draft Angle Top	α	0	12	15	0	12	15
Mold Draft Angle Bottom	β	0	12	15	0	12	15

Table 12.2 : parameters

* Controlling Parameter

§ Significant Characteristic

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

12.4. 20-Lead Plastic Shrink Small Outline (SS) – 209 mil, 5.30 mm (SSOP)

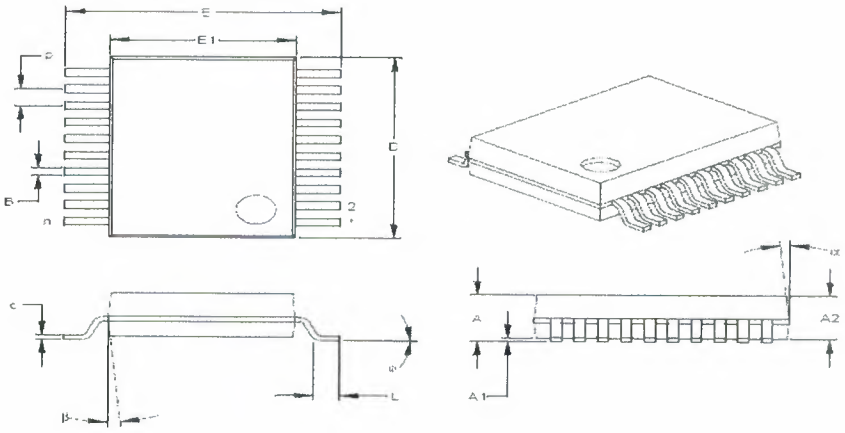


Figure 12.4 : 20-Lead Plastic Shrink Small Outline (SS) – 209 mil, 5.30 mm (SSOP)

Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		20			20	
Pitch	p		.026			0.65	
Overall Height	A	.068	.073	.078	1.73	1.85	1.98
Molded Package Thickness	A2	.064	.068	.072	1.63	1.73	1.83
Standoff §	A1	.002	.006	.010	0.05	0.15	0.25
Overall Width	E	.299	.309	.322	7.59	7.85	8.18
Molded Package Width	E1	.201	.207	.212	5.11	5.25	5.38
Overall Length	D	.278	.284	.289	7.06	7.20	7.34
Foot Length	L	.022	.030	.037	0.56	0.75	0.94
Lead Thickness	c	004	.007	.010	0.10	0.18	0.25
Foot Angle	ϕ	0	4	8	0.00	101.60	203.20
Lead Width	B	010	.013	.015	0.25	0.32	0.38
Mold Draft Angle Top	α	0	5	10	0	5	10
Mold Draft Angle Bottom	β	0	5	10	0	5	10

Table 12.3 : Parameters

* Controlling Parameter

§ Significant Characteristic

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

13. MOVING MESSAGE SIGNAL ON LCD WITH PIC 16F84A

PIC16F84A can be programmable with kinds of programmer circuits and kinds of PC software programs. Nowadays PC software programs are forward softwares. These softwares can use kind of programming languages to program the pics. Here we will use PIC C and ASSEMBLY languages. Also PIC BASIC can be used.

13.1. LCDs

LCD (Liquid Crystal Display) 's firt used in 1960's and one-of-these-days they used too much in the applications and nearly every kind of systems they can be used.

They are high technologic systems and have two different types. One of these types is normal, only shows the writing characters and they have green or blue colours on their screans. Other type is GLCD (Grapich Liquid Crystal Display). This kind of LCD's are colourfull.

LCD's have huge usage area. Example of the usage areas are mobile phones, calculators and televisions. They used to show speed, humidity, heat, pressure and every kind of measurable values.

LCD's are build by 5x7 dotmatrix tables. Selling LCD's in the world are 1x8, 2x8, 2x16, 4x16, 4x20, 4x24 characters. First number is chopper number and last number is the character number. For example 2x16 LCD have 2 choppers 16 characters like in the **figure 13.1.**

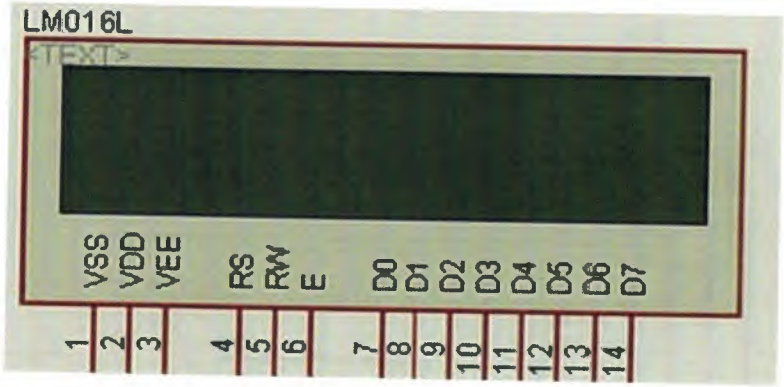


figure 13.1 : General shown of 2x16 LCD in Proteus electronic program

13.1.1. Pins of LCD

1 Vss, ground, logic Vss, logic ground

2 Vcc, +5 Volt, logic Vcc, logic power

3 VLc, VBias, Bias, Contrast

4	RS,	Register select
5	R/W,Read/Write	
6	E,	Enable, Strobe
7-14	D0-D7 (Data entrance)	
15	Led+, A, Backlight Anode, +5 Volt	
16	Led-, K, Backlight Cathode	

Note : 15 and 16 pins not found in some of displays. These pins are used for lighten the panel.

13.2. Connection of the Compenents in Circuit

Moving or stable messages like text, graphich and shapes can made with using this components of **figure 13.2.**

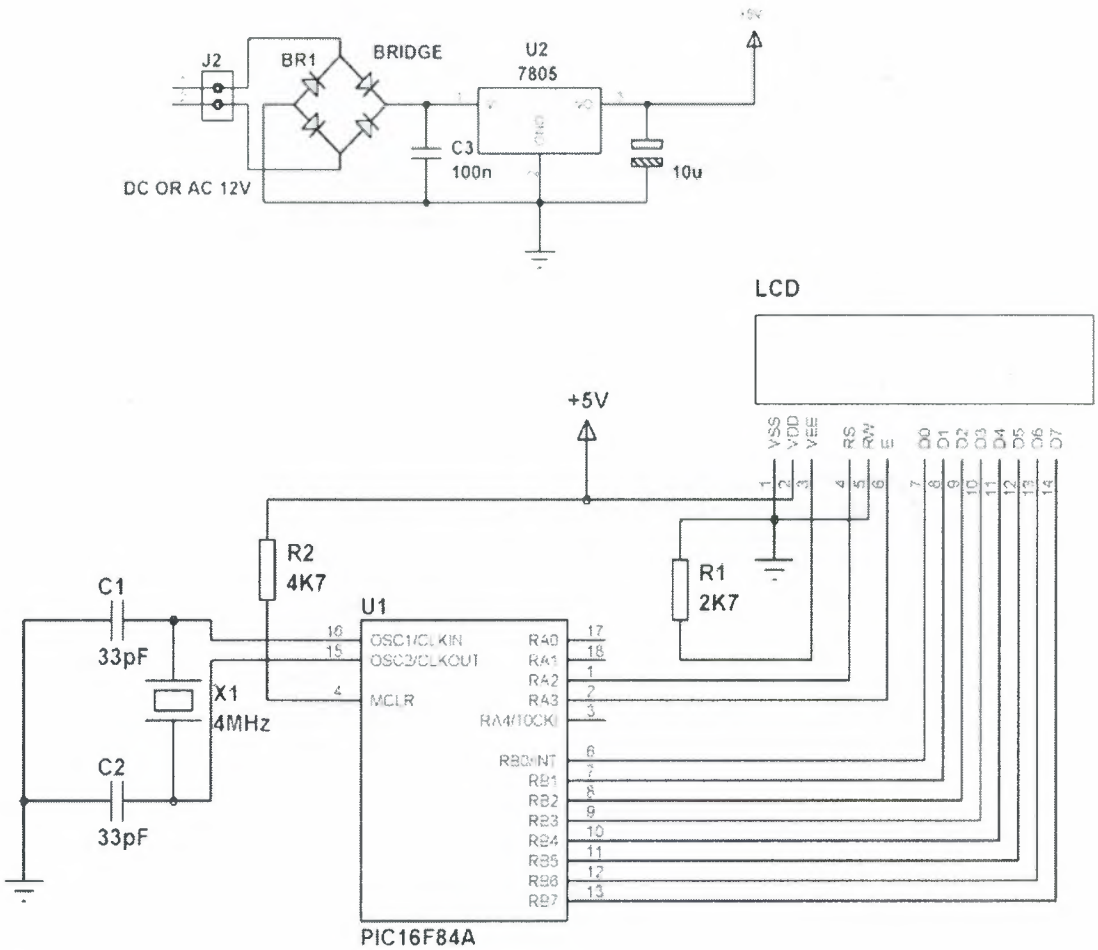


Figure 13.2 Circuit diagram and components of PIC16F84A moving message circuit.

The positioning of circuit component on electronic circuit board seen in **Figure13.3**

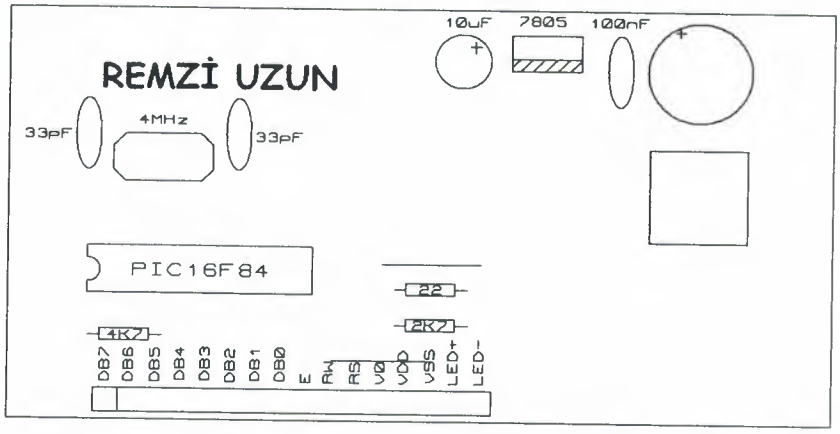


Figure 13.3 Positions of electronic components on board

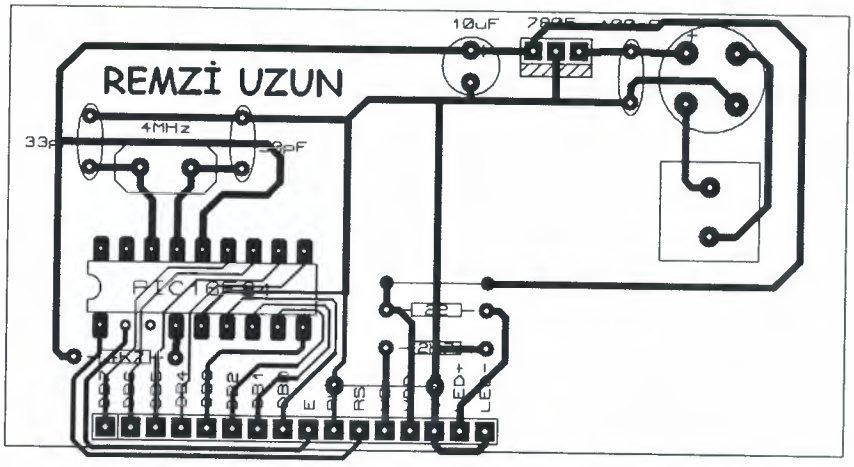


Figure 13.4 Prophyl shown

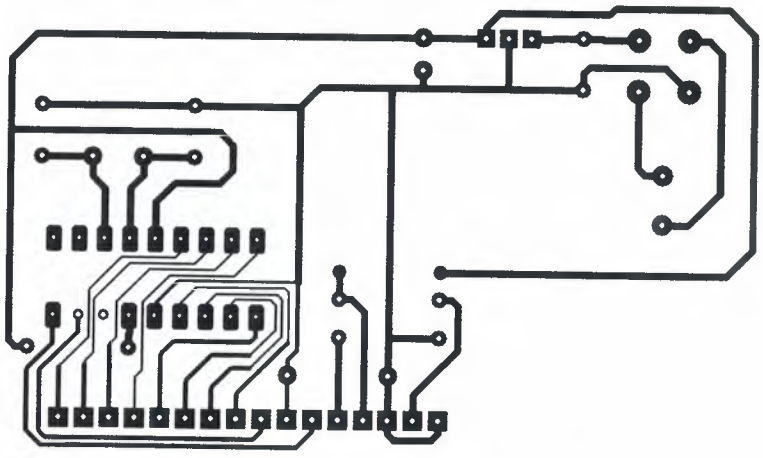


Figure 13.5 Printing shown

13.3. Working Explanation of the Circuit

First lcd starts to work from descriptions of program when the energy is given to the circuit. Waiting time is done 200ms to make the lcd ready at first working. Than LCDOUT instructions are used to write the text at first and second chopper. For-next loop used inside of while-wend loop in the program to make the moving text.

0001 1000 send from PORTB and with make RS=0 to make E downstep corner (first 1 than 0).

13.3.1. Power Unit

Connecting voltage first given to bridge diode to make it full wave d.c.. This full wave d.c. voltage given to chip 7805. Chip 7805 is a regulator and gives pozitif 5V. Than 5V d.c. used to sustenance LCD and PIC16F84A. 100nF and 10µF condensers used like filters at input and output of chip 7805.

13.3.2. Connection of LCD

There is used 2x16 LCD in project. LCD's can be conrtolled from 1, 4 and 8 data bits. E is render of datas or instructions, RS used to control if coming signal is data or intruction, RW used to control if write signal to LCD or read signal from LCD. RW connected to the ground in the project. Because will be written text on LCD. VEE used to control contrasts of screen.

CONCLUSION

We all know that electric and elecronic play a very important part in our life. Electronic components which we use in every sight shows that we lives a technological life. Electronic had a big developed period with programmable chips.

These devices are used like microcontrollers in every electronic, electric, communication and computer systems. So that every systems are differ from each other and needs different types of microcontrollers. You are given special ideas about pics which are used mostly in electric and electronics. Some usage areas are to control motors, to show every kind of measurable values, to show text like advertisement or information and robotic applications.

There are explanations of pic's performance, features, pin diagrams, memories, I/0 ports, instruction sets, electrical characteristics and lcds. In the end there is an project depends of above explanations.

The project is moving the loaded texts on a lcd with pic microcontrollers. Program loaded to pic and the pic connected to circuit. Moving message is making a possibility to read the longer text than lcd.

REFERENCES

[1] Ibrahim D., PIC Programlama ve Ileri PIC Projeleri

[2] Ibrahim D., PIC C ile PIC Programlama

[3] Ibrahim D., PIC Basic

[4} Microchip Family, PICmicro™ Mid-Range Reference Manual (DS33023), 2005 Microchip Technology Inc., USA

[5] Microchip Family, MPLAB® IDE User's Guide (DS51519), 2005 Microchip Technology Inc., April, 5, 2007. U.S.A.

[6] Microchip Family, Application Maestro Software User's Guide, 2005 Microchip Technology Inc., April, 5, 2007. U.S.A.

[7] Çatak M., 16F84 ile LCD Kayan Yazı, "http://www.teknomerkez.net/", Retrived April, 12, 2007. TURKEY

[8] Akyan M., 216 lcd kayan yazı, "http://www.elektrotekno.com/post-38194.html", Retrived April, 20, 2007. TURKEY

[9] Electronic Portal, "http://www.elektronikciler.com/Yazilar.asp?goster=kat&kategori=22", Retrived April, 15, 2007. TURKEY

[10] "http://ari.cankaya.edu.tr/~dere/picproje.htm", TURKEY

[11] Form, "http://www.forumilk.com/", Retrived March,29,2007. TURKEY

[12] How to Control a LCD, "http://home.iae.nl/users/pouweha/lcd/lcd.shtml", Retrived March, 15, 2007. ENGLAND

[13] Microchip website , "http://www.microchip.com/cn" , Retrived March,22,2007. U.S.A.

[14] HI-TECH Software Pty. Ltd., HI-TECH PICC Compiler, "http://www.htsoft.com", Retrived September, 27, 2007. AUSTRALIA

APPENDIXES

APPENDIX A: CONVERSION CONSIDERATIONS

Considerations for converting from one PIC16X8X device to another are listed in Table 1.

Difference	PIC16C84	PIC16F83/F84	PIC16CR83/ CR84	PIC16F84A
Program Memory Size	1K x 14	512 x 14 / 1K x 14	512 x 14 / 1K x 14	1K x 14
Data Memory Size	36 x 8	36 x 8 / 68 x 8	36 x 8 / 68 x 8	68 x 8
Voltage Range	2.0V - 6.0V (-40°C to +85°C)	2.0V - 6.0V (-40°C to +85°C)	2.0V - 6.0V (-40°C to +85°C)	2.0V - 5.5V (-40°C to +125°C)
Maximum Operating Frequency	10 MHz	10 MHz	10 MHz	20 MHz
Supply Current (I_{DD}). See parameter # D014 in the electrical specs for more detail.	I_{DD} (typ) = 60 µA I_{DD} (max) = 400 µA (LP osc, Fosc = 32 kHz, V_{DD} = 2.0V, WDT disabled)	I_{DD} (typ) = 15 µA I_{DD} (max) = 45 µA (LP osc, Fosc = 32 kHz, V_{DD} = 2.0V, WDT disabled)	I_{DD} (typ) = 15 µA I_{DD} (max) = 45 µA (LP osc, Fosc = 32 kHz, V_{DD} = 2.0V, WDT disabled)	I_{DD} (typ) = 15 µA I_{DD} (max) = 45 µA (LP osc, Fosc = 32 kHz, V_{DD} = 2.0V, WDT disabled)
Power-down Current (I_{PD}). See parameters # D020, D021, and D021A in the electrical specs for more detail.	I_{PD} (typ) = 26 µA I_{PD} (max) = 100 µA (V_{DD} = 2.0V, WDT disabled, industrial)	I_{PD} (typ) = 0.4 µA I_{PD} (max) = 9 µA (V_{DD} = 2.0V, WDT disabled, industrial)	I_{PD} (typ) = 0.4 µA I_{PD} (max) = 6 µA (V_{DD} = 2.0V, WDT disabled, industrial)	I_{PD} (typ) = 0.4 µA I_{PD} (max) = 1 µA (V_{DD} = 2.0V, WDT disabled, industrial)
Input Low Voltage (V_{IL}). See parameters # D032 and D034 in the electrical specs for more detail.	V_{IL} (max) = 0.2V_{DD} (OSC1, RC mode)	V_{IL} (max) = 0.1V_{DD} (OSC1, RC mode)	V_{IL} (max) = 0.1V_{DD} (OSC1, RC mode)	V_{IL} (max) = 0.1V_{DD} (OSC1, RC mode)
Input High Voltage (V_{IH}). See parameter # D040 in the electrical specs for more detail.	V_{IH} (min) = 0.36V_{DD} (I/O Ports with TTL, 4.5V ≤ V_{DD} ≤ 5.5V)	V_{IH} (min) = 2.4V (I/O Ports with TTL, 4.5V ≤ V_{DD} ≤ 5.5V)	V_{IH} (min) = 2.4V (I/O Ports with TTL, 4.5V ≤ V_{DD} ≤ 5.5V)	V_{IH} (min) = 2.4V (I/O Ports with TTL, 4.5V ≤ V_{DD} ≤ 5.5V)
Data EEPROM Memory Erase/Write cycle time (T_{DEW}). See parameter # D122 in the electrical specs for more detail.	T_{DEW} (typ) = 10 ms T_{DEW} (max) = 20 ms	T_{DEW} (typ) = 10 ms T_{DEW} (max) = 20 ms	T_{DEW} (typ) = 10 ms T_{DEW} (max) = 20 ms	T_{DEW} (typ) = 4 ms T_{DEW} (max) = 8 ms
Port Output Rise/Fall time (TioR, TioF). See parameters #20, 20A, 21, and 21A in the electrical specs for more detail	TioR, TioF (max) = 25 ns (C84) TioR, TioF (max) = 60 ns (LC84)	TioR, TioF (max) = 35 ns (C84) TioR, TioF (max) = 70 ns (LC84)	TioR, TioF (max) = 35 ns (C84) TioR, TioF (max) = 70 ns (LC84)	TioR, TioF (max) = 35 ns (C84) TioR, TioF (max) = 70 ns (LC84)
$\overline{MCLR}$ on-chip filter. See parameter #30 in the electrical specs for more detail	No	Yes	Yes	Yes
PORTA and crystal oscillator values less than 500 kHz	For crystal oscillator configurations operating below 500 kHz, the device may generate a spurious internal Q-clock when PORTA<0> switches state.	N/A	N/A	N/A
RB0/INT pin	TTL	TTL/ST* (*Schmitt Trigger)	TTL/ST* (*Schmitt Trigger)	TTL/ST* (*Schmitt Trigger)

Table 1: Conversion considerations - PIC16C84, PIC16F83/F84, PIC16CR83/CR84, PIC16F84A

Difference	PIC16C84	PIC16F83/F84	PIC16CR83/ CR84	PIC16F84A
EEADR<7:6> and IDD	It is recommended that the EEADR<7:6> bits be cleared. When either of these bits is set, the maximum IDD for the device is higher than when both are cleared.	N/A	N/A	N/A
The polarity of the $\overline{\text{PWRTE}}$ bit	PWRTE	$\overline{\text{PWRTE}}$	$\overline{\text{PWRTE}}$	$\overline{\text{PWRTE}}$
Recommended value of REXT for RC oscillator circuits	REXT = 3kΩ - 100kΩ	REXT = 5kΩ - 100kΩ	REXT = 5kΩ - 100kΩ	REXT = 3kΩ - 100kΩ
GIE bit unintentional enable	If an interrupt occurs while the Global Interrupt Enable (GIE) bit is being cleared, the GIE bit may unintentionally be re-enabled by the user's Interrupt Service Routine (the RETFIE instruction).	N/A	N/A	N/A
Packages	PDIP, SOIC	PDIP, SOIC	PDIP, SOIC	PDIP, SOIC, SSOP
Open Drain High Voltage (VDD)	14V	12V	12V	8.5V

Table 1: Conversion considerations - PIC16C84, PIC16F83/F84, PIC16CR83/CR84, PIC16F84A (continued)

APPENDIX B: MIGRATION FROM BASELINE TO MID-RANGE DEVICES

This section discusses how to migrate from a baseline device (i.e., PIC16C5X) to a mid-range device (i.e., PIC16CXXX).

The following is the list of feature improvements over the PIC16C5X microcontroller family:

1. Instruction word length is increased to 14-bits. This allows larger page sizes, both in program memory (2K now as opposed to 512K before) and the register file (128 bytes now versus 32 bytes before).
2. A PC latch register (PCLATH) is added to handle program memory paging. PA2, PA1 and PA0 bits are removed from the STATUS register and placed in the OPTION register.
3. Data memory paging is redefined slightly. The STATUS register is modified.
4. Four new instructions have been added: RETURN, RETFIE, ADDLW, and SUBLW.

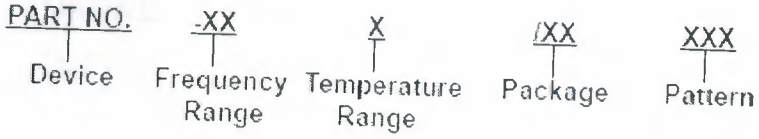
Two instructions, TRIS and OPTION, are being phased out, although they are kept for compatibility with PIC16C5X.

5. OPTION and TRIS registers are made addressable.
6. Interrupt capability is added. Interrupt vector is at 0004h.
7. Stack size is increased to eight-deep.
8. RESET vector is changed to 0000h.
9. RESET of all registers is revisited. Five different RESET (and wake-up) types are recognized. Registers are reset differently.
10. Wake-up from SLEEP through interrupt is added.
11. Two separate timers, the Oscillator Start-up Timer (OST) and Power-up Timer (PWRT), are included for more reliable power-up. These timers are invoked selectively to avoid unnecessary delays on power-up and wake-up.
12. PORTB has weak pull-ups and interrupt-onchange features.
13. T0CKI pin is also a port pin (RA4/T0CKI).
14. FSR is a full 8-bit register.
15. "In system programming" is made possible. The user can program PIC16CXX devices using only five pins: VDD, VSS, VPP, RB6 (clock) and RB7 (data in/out).

To convert code written for PIC16C5X to PIC16F84A, the user should take the following steps:

1. Remove any program memory page select operations (PA2, PA1, PA0 bits) for CALL, GOTO.
2. Revisit any computed jump operations (write to PC or add to PC, etc.) to make sure page bits are set properly under the new scheme.
3. Eliminate any data memory page switching. Redefine data variables for reallocation.
4. Verify all writes to STATUS, OPTION, and FSR registers since these have changed.
5. Change RESET vector to 0000h.

PIC16F84A PRODUCT IDENTIFICATION SYSTEM



Device PIC16F84A(1), PIC16F84AT(2) PIC16LF84A(1), PIC16LF84AT(2)

Frequency Range 04 = 4 MHz

20 = 20 MHz

Temperature Range -= 0°C to +70°C

I = -40°C to +85°C

Package P = PDIP

SO = SOIC (Gull Wing, 300 mil body)

SS = SSOP

Pattern QTP, SQTP, ROM Code (factory specified) or Special Requirements . Blank for OTP and Windowed devices.

Examples:

a) PIC16F84A -04/P 301 = Commercial temp., PDIP package, 4 MHz, normal VDD limits, QTP pattern #301.

b) PIC16LF84A - 04I/SO = Industrial temp., SOIC package, 200 kHz, Extended VDD limits.

c) PIC16F84A - 20I/P = Industrial temp., PDIP package, 20 MHz, normal VDD limits.

Note 1: F = Standard VDD range

LF = Extended VDD range

2: T = in tape and reel - SOIC and SSOP packages only.

APPENDIX C : MOVING MESSAGE PIC BASIC PRO CODES

```
DEFINE     LCD_DREG  PORTB
DEFINE     LCD_DBIT   0
DEFINE     LCD_RSREG PORTA        'RS connected to PORTA
DEFINE     LCD_RSBIT  2
DEFINE     LCD_EREG   PORTA       'E connected to PORTA
DEFINE     LCD_EBIT   3
DEFINE     LCD_BITS   8           '8 data pins of LCD are used
DEFINE     LCD_LINES  2

N  VAR BYTE
```

```
      PAUSE  200                              '200MS waited for LCD
WHILE 1>0
  LCDOUT  $FE,1,"          REMZI UZUN"
  LCDOUT  $FE,$C0,"        ELECTRICAL ENGINEER"
  FOR N=1 TO 35
  PORTB=%00011000     'The message send to the LCD to PORT B
  LOW PORTA.2         'PORTA.2 = 0 .
  HIGH  PORTA.3       'PORTA.3 = 1 .
  PAUSEUS 5           '5microsecond waited
  LOW PORTA.3
  PAUSE  1000         '1000ms (ONE SECOND) waited.
  NEXT N
WEND
     END          'PROGRAM END
```