



Welcome to E-XFL.COM

What is "[Embedded - Microcontrollers](#)"?

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

Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	4MHz
Connectivity	-
Peripherals	POR, WDT
Number of I/O	20
Program Memory Size	3KB (2K x 12)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	72 x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 5.5V
Data Converters	-
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	28-SOIC
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16c57ct-04i-so

3.0 ARCHITECTURAL OVERVIEW

The high performance of the PIC16C5X family can be attributed to a number of architectural features commonly found in RISC microprocessors. To begin with, the PIC16C5X uses a Harvard architecture in which program and data are accessed on separate buses. This improves bandwidth over traditional von Neumann architecture where program and data are fetched on the same bus. Separating program and data memory further allows instructions to be sized differently than the 8-bit wide data word. Instruction opcodes are 12 bits wide making it possible to have all single word instructions. A 12-bit wide program memory access bus fetches a 12-bit instruction in a single cycle. A two-stage pipeline overlaps fetch and execution of instructions. Consequently, all instructions (33) execute in a single cycle except for program branches.

The PIC16C54/CR54 and PIC16C55 address 512 x 12 of program memory, the PIC16C56/CR56 address 1K x 12 of program memory, and the PIC16C57/CR57 and PIC16C58/CR58 address 2K x 12 of program memory. All program memory is internal.

The PIC16C5X can directly or indirectly address its register files and data memory. All special function registers including the program counter are mapped in the data memory. The PIC16C5X has a highly orthogonal (symmetrical) instruction set that makes it possible to carry out any operation on any register using any addressing mode. This symmetrical nature and lack of 'special optimal situations' make programming with the PIC16C5X simple yet efficient. In addition, the learning curve is reduced significantly.

The PIC16C5X device contains an 8-bit ALU and working register. The ALU is a general purpose arithmetic unit. It performs arithmetic and Boolean functions between data in the working register and any register file.

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

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

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

A simplified block diagram is shown in Figure 3-1, with the corresponding device pins described in Table 3-1 (for PIC16C54/56/58) and Table 3-2 (for PIC16C55/57).

5.0 RESET

PIC16C5X devices may be RESET in one of the following ways:

- Power-On Reset (POR)
- $\overline{\text{MCLR}}$ Reset (normal operation)
- $\overline{\text{MCLR}}$ Wake-up Reset (from SLEEP)
- WDT Reset (normal operation)
- WDT Wake-up Reset (from SLEEP)

Table 5-1 shows these RESET conditions for the PCL and STATUS registers.

Some registers are not affected in any RESET condition. Their status is unknown on POR and unchanged in any other RESET. Most other registers are reset to a "RESET state" on Power-On Reset (POR), $\overline{\text{MCLR}}$ or WDT Reset. A $\overline{\text{MCLR}}$ or WDT wake-up from SLEEP also results in a device RESET, and not a continuation of operation before SLEEP.

The $\overline{\text{TO}}$ and $\overline{\text{PD}}$ bits (STATUS <4:3>) are set or cleared depending on the different RESET conditions (Table 5-1). These bits may be used to determine the nature of the RESET.

Table 5-3 lists a full description of RESET states of all registers. Figure 5-1 shows a simplified block diagram of the On-chip Reset circuit.

TABLE 5-1: STATUS BITS AND THEIR SIGNIFICANCE

Condition	$\overline{\text{TO}}$	$\overline{\text{PD}}$
Power-On Reset	1	1
$\overline{\text{MCLR}}$ Reset (normal operation)	u	u
$\overline{\text{MCLR}}$ Wake-up (from SLEEP)	1	0
WDT Reset (normal operation)	0	1
WDT Wake-up (from SLEEP)	0	0

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

TABLE 5-2: SUMMARY OF REGISTERS ASSOCIATED WITH RESET

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on POR	Value on $\overline{\text{MCLR}}$ and WDT Reset
03h	STATUS	PA2	PA1	PA0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0001 1xxx	000q quuu

Legend: u = unchanged, x = unknown, q = see Table 5-1 for possible values.

5.1 Power-On Reset (POR)

The PIC16C5X family incorporates on-chip Power-On Reset (POR) circuitry which provides an internal chip RESET for most power-up situations. To use this feature, the user merely ties the $\overline{\text{MCLR}}/\text{VPP}$ pin to VDD . A simplified block diagram of the on-chip Power-On Reset circuit is shown in Figure 5-1.

The Power-On Reset circuit and the Device Reset Timer (Section 5.2) circuit are closely related. On power-up, the RESET latch is set and the DRT is RESET. The DRT timer begins counting once it detects $\overline{\text{MCLR}}$ to be high. After the time-out period, which is typically 18 ms, it will RESET the reset latch and thus end the on-chip RESET signal.

A power-up example where $\overline{\text{MCLR}}$ is not tied to VDD is shown in Figure 5-3. VDD is allowed to rise and stabilize before bringing $\overline{\text{MCLR}}$ high. The chip will actually come out of reset T_{DRT} msec after $\overline{\text{MCLR}}$ goes high.

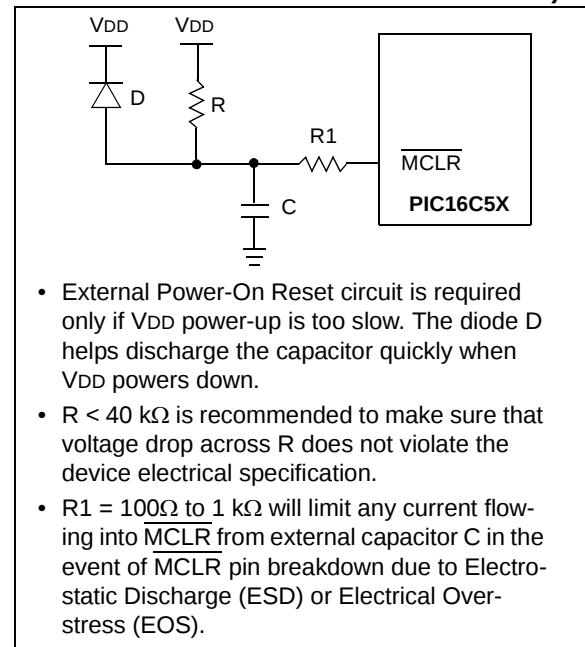
In Figure 5-4, the on-chip Power-On Reset feature is being used ($\overline{\text{MCLR}}$ and VDD are tied together). The VDD is stable before the start-up timer times out and there is no problem in getting a proper RESET. However, Figure 5-5 depicts a problem situation where VDD rises too slowly. The time between when the DRT senses a high on the $\overline{\text{MCLR}}/\text{VPP}$ pin, and when the $\overline{\text{MCLR}}/\text{VPP}$ pin (and VDD) actually reach their full value, is too long. In this situation, when the start-up timer times out, VDD has not reached the $\text{VDD}(\text{min})$ value and the chip is, therefore, not guaranteed to function correctly. For such situations, we recommend that external RC circuits be used to achieve longer POR delay times (Figure 5-2).

Note: 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.

For more information on PIC16C5X POR, see *Power-Up Considerations* - AN522 in the [Embedded Control Handbook](#).

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

FIGURE 5-2: EXTERNAL POWER-ON RESET CIRCUIT (FOR SLOW VDD POWER-UP)



6.0 MEMORY ORGANIZATION

PIC16C5X memory is organized into program memory and data memory. For devices with more than 512 bytes of program memory, a paging scheme is used. Program memory pages are accessed using one or two STATUS Register bits. For devices with a data memory register file of more than 32 registers, a banking scheme is used. Data memory banks are accessed using the File Selection Register (FSR).

6.1 Program Memory Organization

The PIC16C54, PIC16CR54 and PIC16C55 have a 9-bit Program Counter (PC) capable of addressing a 512 x 12 program memory space (Figure 6-1). The PIC16C56 and PIC16CR56 have a 10-bit Program Counter (PC) capable of addressing a 1K x 12 program memory space (Figure 6-2). The PIC16C57, PIC16C58 and PIC16CR58 have an 11-bit Program Counter (PC) capable of addressing a 2K x 12 program memory space (Figure 6-3). Accessing a location above the physically implemented address will cause a wraparound.

A NOP at the RESET vector location will cause a restart at location 000h. The RESET vector for the PIC16C54, PIC16CR54 and PIC16C55 is at 1FFh. The RESET vector for the PIC16C56 and PIC16CR56 is at 3FFh. The RESET vector for the PIC16C57, PIC16CR57, PIC16C58, and PIC16CR58 is at 7FFh. See Section 6.5 for additional information using CALL and GOTO instructions.

FIGURE 6-1: PIC16C54/CR54/C55 PROGRAM MEMORY MAP AND STACK

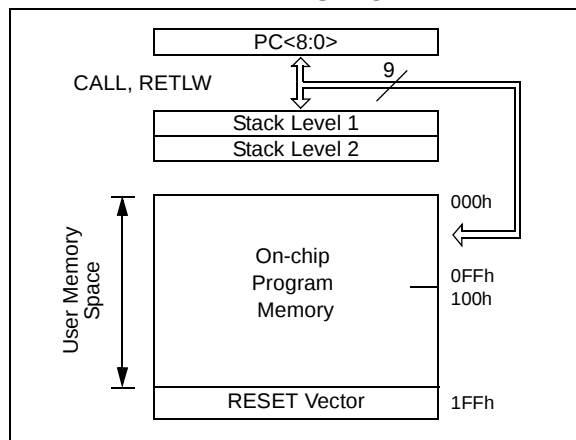


FIGURE 6-2: PIC16C56/CR56 PROGRAM MEMORY MAP AND STACK

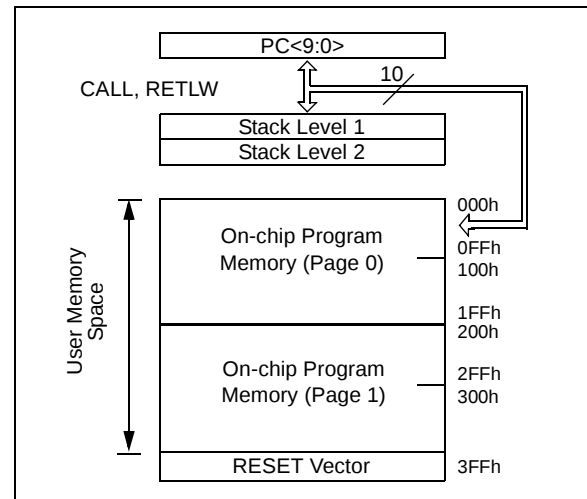
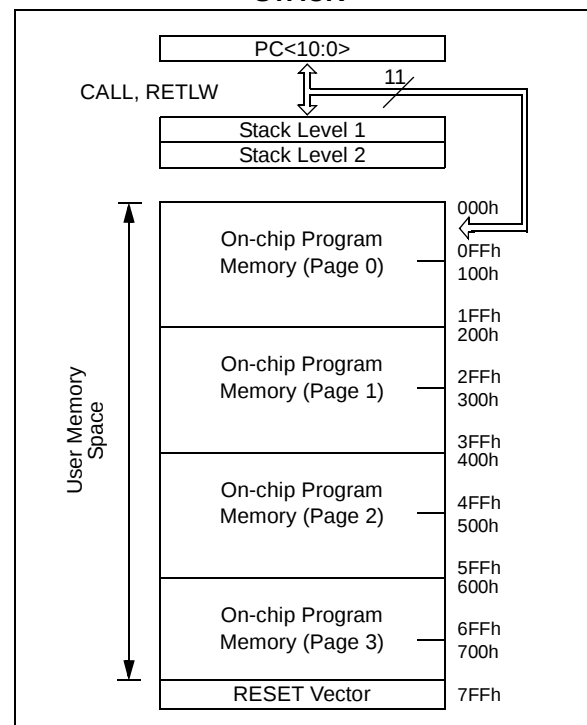


FIGURE 6-3: PIC16C57/CR57/C58/CR58 PROGRAM MEMORY MAP AND STACK



GOTO Unconditional Branch

Syntax: [*label*] GOTO *k*

Operands: $0 \leq k \leq 511$

Operation: $k \rightarrow PC<8:0>;$
 $STATUS<6:5> \rightarrow PC<10:9>$

Status Affected: None

Encoding:

101k	kkkk	kkkk
------	------	------

Description: GOTO is an unconditional branch. The 9-bit immediate value is loaded into PC bits <8:0>. The upper bits of PC are loaded from STATUS<6:5>. GOTO is a two-cycle instruction.

Words: 1

Cycles: 2

Example: GOTO THERE

After Instruction
 PC = address (THERE)

INCF Increment f

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

Operands: $0 \leq f \leq 31$
 $d \in [0,1]$

Operation: $(f) + 1 \rightarrow (\text{dest})$

Status Affected: Z

Encoding:

0010	10df	ffff
------	------	------

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

Words: 1

Cycles: 1

Example: INCF CNT, 1

Before Instruction

CNT = 0xFF
 Z = 0

After Instruction

CNT = 0x00
 Z = 1

INCFSZ Increment f, Skip if 0

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

Operands: $0 \leq f \leq 31$
 $d \in [0,1]$

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

Status Affected: None

Encoding:

0011	11df	ffff
------	------	------

Description: The contents of register 'f' are incremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'. If the result is 0, then the next instruction, which is already fetched, is discarded and a NOP is executed instead making it a two-cycle instruction.

Words: 1

Cycles: 1(2)

Example: HERE INCFSZ CNT, 1
 GOTO LOOP
 CONTINUE •
 •
 •

Before Instruction

PC = address (HERE)

After Instruction

CNT = CNT + 1;
 if CNT = 0,
 PC = address (CONTINUE);
 if CNT ≠ 0,
 PC = address (HERE + 1)

SUBWF Subtract W from f

Syntax: `[label] SUBWF f,d`

Operands: $0 \leq f \leq 31$
 $d \in [0,1]$

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

Status Affected: C, DC, Z

Encoding:

0000	10df	ffff
------	------	------

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

Words: 1

Cycles: 1

Example 1: SUBWF REG1, 1

Before Instruction
 REG1 = 3
 W = 2
 C = ?

After Instruction
 REG1 = 1
 W = 2
 C = 1 ; result is positive

Example 2:

Before Instruction
 REG1 = 2
 W = 2
 C = ?

After Instruction
 REG1 = 0
 W = 2
 C = 1 ; result is zero

Example 3:

Before Instruction
 REG1 = 1
 W = 2
 C = ?

After Instruction
 REG1 = 0xFF
 W = 2
 C = 0 ; result is negative

SWAPF Swap Nibbles in f

Syntax: `[label] SWAPF f,d`

Operands: $0 \leq f \leq 31$
 $d \in [0,1]$

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

Status Affected: None

Encoding:

0011	10df	ffff
------	------	------

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

Words: 1

Cycles: 1

Example SWAPF REG1, 0

Before Instruction
 REG1 = 0xA5

After Instruction
 REG1 = 0xA5
 W = 0x5A

TRIS Load TRIS Register

Syntax: `[label] TRIS f`

Operands: $f = 5, 6 \text{ or } 7$

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

Status Affected: None

Encoding:

0000	0000	0fff
------	------	------

Description: TRIS register 'f' ($f = 5, 6, \text{ or } 7$) is loaded with the contents of the W register.

Words: 1

Cycles: 1

Example TRIS PORTB

Before Instruction
 W = 0xA5

After Instruction
 TRISB = 0xA5

11.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK object linker combines relocatable objects created by the MPASM assembler and the MPLAB C17 and MPLAB C18 C compilers. It can also link relocatable objects from pre-compiled libraries, using directives from a linker script.

The MPLIB object librarian is a librarian for pre-compiled code to be used with the MPLINK object linker. When a routine from a library is called from another source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications. The MPLIB object librarian manages the creation and modification of library files.

The MPLINK object linker features include:

- Integration with MPASM assembler and MPLAB C17 and MPLAB C18 C compilers.
- Allows all memory areas to be defined as sections to provide link-time flexibility.

The MPLIB object librarian features include:

- Easier linking because single libraries can be included instead of many smaller files.
- Helps keep code maintainable by grouping related modules together.
- Allows libraries to be created and modules to be added, listed, replaced, deleted or extracted.

11.5 MPLAB SIM Software Simulator

The MPLAB SIM software simulator allows code development in a PC-hosted environment by simulating the PIC series microcontrollers on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a file, or user-defined key press, to any of the pins. The execution can be performed in single step, execute until break, or trace mode.

The MPLAB SIM simulator fully supports symbolic debugging using the MPLAB C17 and the MPLAB C18 C compilers and the MPASM assembler. The software simulator offers the flexibility to develop and debug code outside of the laboratory environment, making it an excellent multi-project software development tool.

11.6 MPLAB ICE High Performance Universal In-Circuit Emulator with MPLAB IDE

The MPLAB ICE universal in-circuit emulator is intended to provide the product development engineer with a complete microcontroller design tool set for PIC microcontrollers (MCUs). Software control of the MPLAB ICE in-circuit emulator is provided by the MPLAB Integrated Development Environment (IDE), which allows editing, building, downloading and source debugging from a single environment.

The MPLAB ICE 2000 is a full-featured emulator system with enhanced trace, trigger and data monitoring features. Interchangeable processor modules allow the system to be easily reconfigured for emulation of different processors. The universal architecture of the MPLAB ICE in-circuit emulator allows expansion to support new PIC microcontrollers.

The MPLAB ICE in-circuit emulator system has been designed as a real-time emulation system, with advanced features that are generally found on more expensive development tools. The PC platform and Microsoft® Windows environment were chosen to best make these features available to you, the end user.

11.7 ICEPIC In-Circuit Emulator

The ICEPIC low cost, in-circuit emulator is a solution for the Microchip Technology PIC16C5X, PIC16C6X, PIC16C7X and PIC16CXXX families of 8-bit One-Time-Programmable (OTP) microcontrollers. The modular system can support different subsets of PIC16C5X or PIC16CXXX products through the use of interchangeable personality modules, or daughter boards. The emulator is capable of emulating without target application circuitry being present.

FIGURE 13-4: RESET, WATCHDOG TIMER, AND DEVICE RESET TIMER TIMING - PIC16CR54A

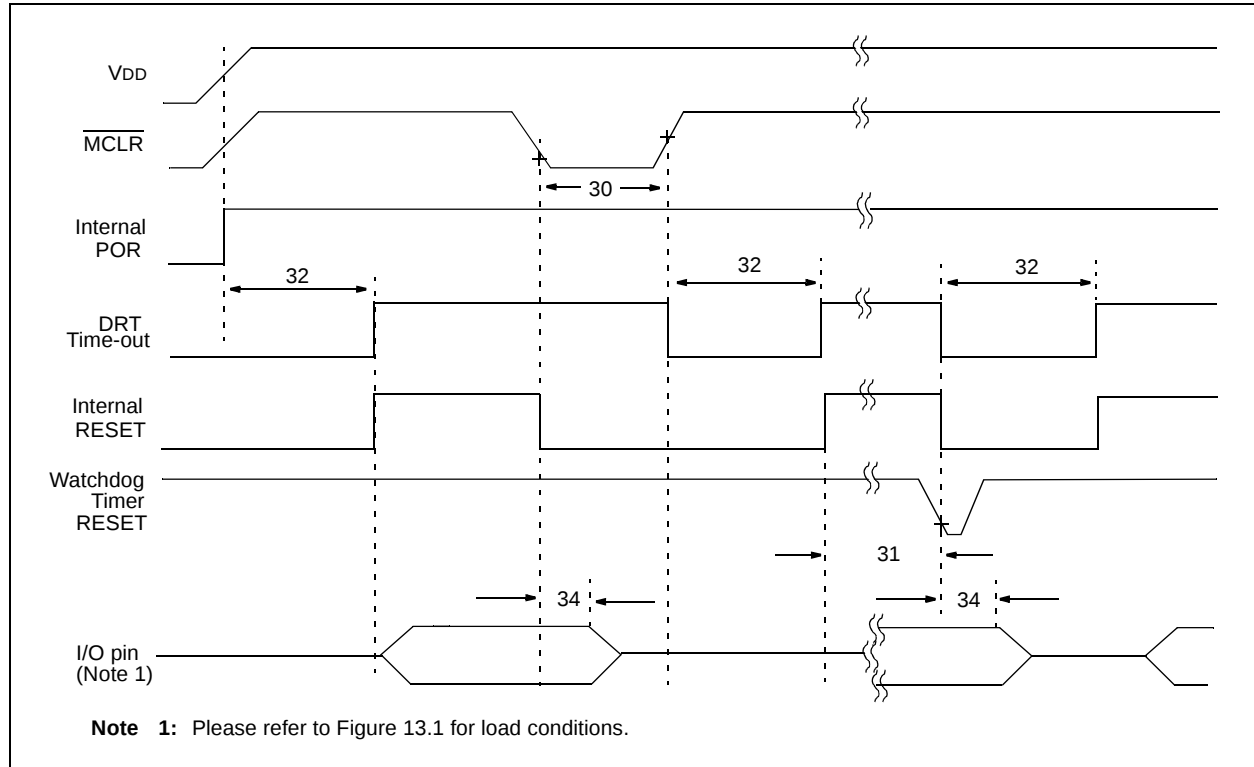


TABLE 13-3: RESET, WATCHDOG TIMER, AND DEVICE RESET TIMER - PIC16CR54A

Standard Operating Conditions (unless otherwise specified)							
AC Characteristics		Operating Temperature					
		0°C ≤ TA ≤ +70°C for commercial					
		–40°C ≤ TA ≤ +85°C for industrial					
		–40°C ≤ TA ≤ +125°C for extended					
Param No.	Symbol	Characteristic	Min	Typ†	Max	Units	Conditions
30	TmCL	MCLR Pulse Width (low)	1.0*	—	—	μs	VDD = 5.0V
31	Twdt	Watchdog Timer Time-out Period (No Prescaler)	7.0*	18*	40*	ms	VDD = 5.0V (Comm)
32	TDRT	Device Reset Timer Period	7.0*	18*	30*	ms	VDD = 5.0V (Comm)
34	Tioz	I/O Hi-impedance from MCLR Low	—	—	1.0*	μs	

* These parameters are characterized but not tested.

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

FIGURE 14-9: V_{TH} (INPUT THRESHOLD VOLTAGE) OF I/O PINS vs. V_{DD}

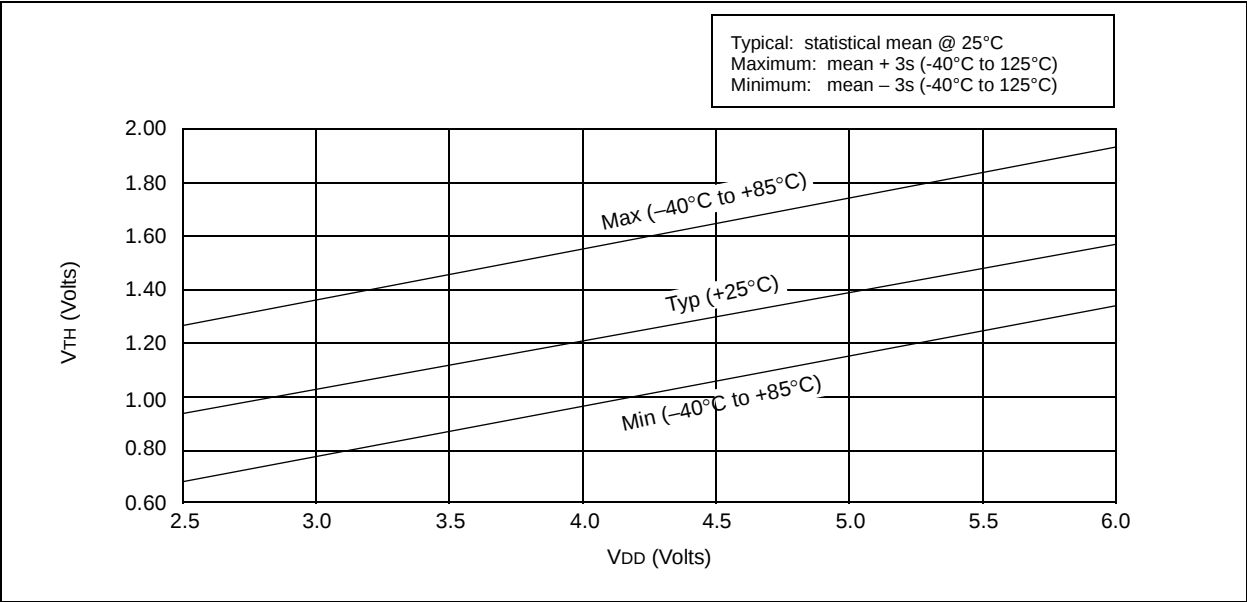
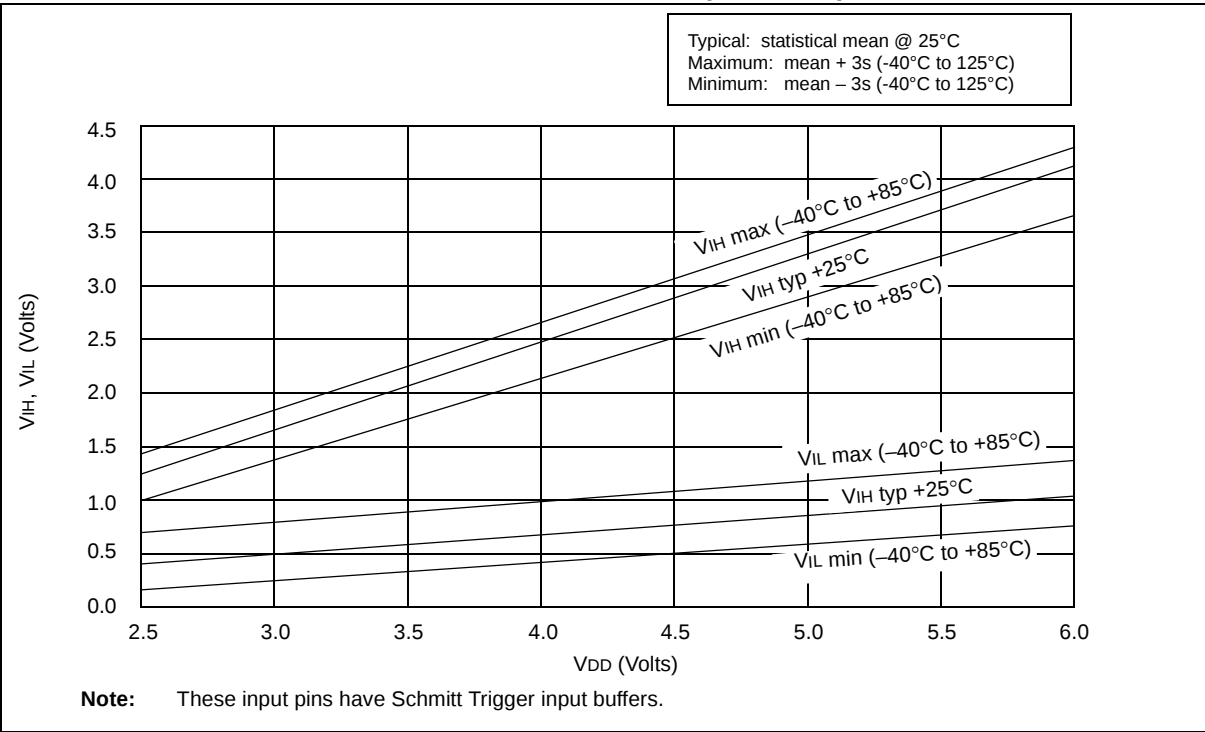


FIGURE 14-10: V_{IH} , V_{IL} OF MCLR, T0CKI AND OSC1 (RC MODE) vs. V_{DD}



PIC16C5X

**TABLE 14-2: INPUT CAPACITANCE FOR
PIC16C54/56**

Pin	Typical Capacitance (pF)	
	18L PDIP	18L SOIC
RA port	5.0	4.3
RB port	5.0	4.3
$\overline{\text{MCLR}}$	17.0	17.0
OSC1	4.0	3.5
OSC2/CLKOUT	4.3	3.5
T0CKI	3.2	2.8

All capacitance values are typical at 25°C. A part-to-part variation of $\pm 25\%$ (three standard deviations) should be taken into account.

**TABLE 14-3: INPUT CAPACITANCE FOR
PIC16C55/57**

Pin	Typical Capacitance (pF)	
	28L PDIP (600 mil)	28L SOIC
RA port	5.2	4.8
RB port	5.6	4.7
RC port	5.0	4.1
$\overline{\text{MCLR}}$	17.0	17.0
OSC1	6.6	3.5
OSC2/CLKOUT	4.6	3.5
T0CKI	4.5	3.5

All capacitance values are typical at 25°C. A part-to-part variation of $\pm 25\%$ (three standard deviations) should be taken into account.

15.6 Timing Diagrams and Specifications

FIGURE 15-2: EXTERNAL CLOCK TIMING - PIC16C54A

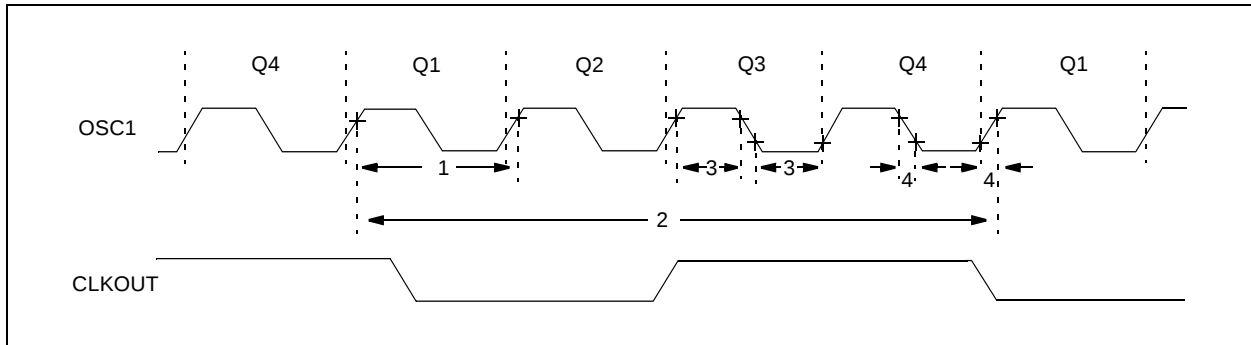


TABLE 15-1: EXTERNAL CLOCK TIMING REQUIREMENTS - PIC16C54A

Standard Operating Conditions (unless otherwise specified)							
AC Characteristics							
Operating Temperature 0°C ≤ TA ≤ +70°C for commercial -40°C ≤ TA ≤ +85°C for industrial -20°C ≤ TA ≤ +85°C for industrial - PIC16LV54A-02I -40°C ≤ TA ≤ +125°C for extended							
Param No.	Symbol	Characteristic	Min	Typ†	Max	Units	Conditions
	Fosc	External CLKIN Frequency ⁽¹⁾	DC	—	4.0	MHz	XT osc mode
			DC	—	2.0	MHz	XT osc mode (PIC16LV54A)
			DC	—	4.0	MHz	HS osc mode (04)
			DC	—	10	MHz	HS osc mode (10)
			DC	—	20	MHz	HS osc mode (20)
			DC	—	200	kHz	LP osc mode
		Oscillator Frequency ⁽¹⁾	DC	—	4.0	MHz	RC osc mode
			DC	—	2.0	MHz	RC osc mode (PIC16LV54A)
			0.1	—	4.0	MHz	XT osc mode
			0.1	—	2.0	MHz	XT osc mode (PIC16LV54A)
			4.0	—	4.0	MHz	HS osc mode (04)
			4.0	—	10	MHz	HS osc mode (10)
			4.0	—	20	MHz	HS osc mode (20)
			5.0	—	200	kHz	LP osc mode

* These parameters are characterized but not tested.

† Data in the Typical ("Typ") column is based on characterization results at 25°C. This data is for design guidance only and is not tested.

Note 1: 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.

When an external clock input is used, the "max" cycle time limit is "DC" (no clock) for all devices.

2: Instruction cycle period (Tcy) equals four times the input oscillator time base period.

FIGURE 16-7: V_{TH} (INPUT THRESHOLD VOLTAGE) OF I/O PINS - V_{DD}

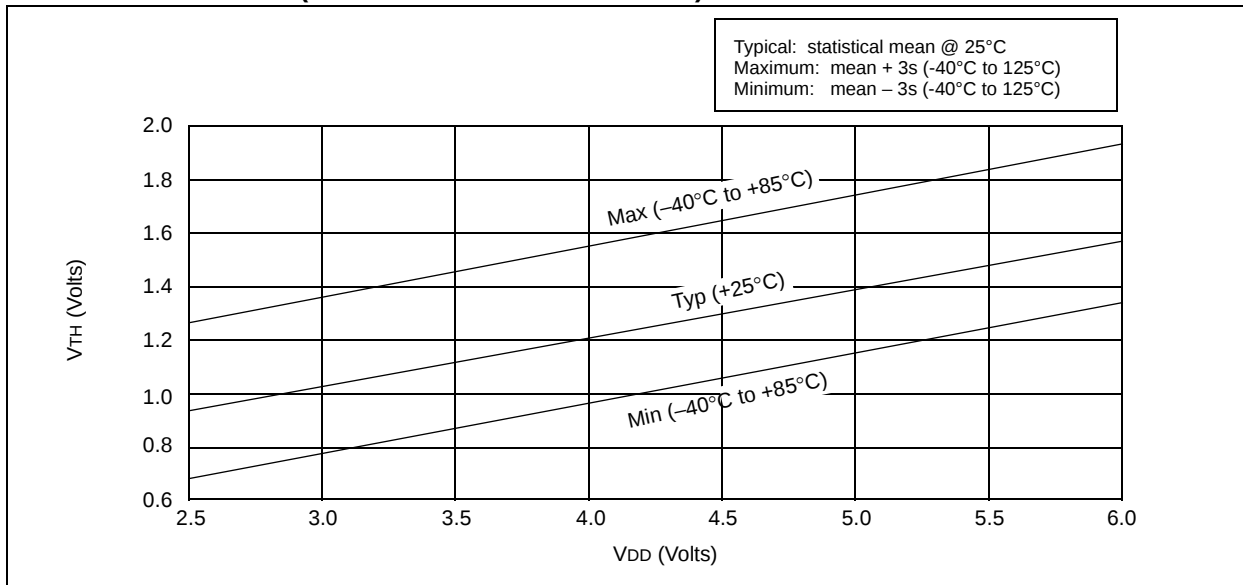


FIGURE 16-8: V_{TH} (INPUT THRESHOLD VOLTAGE) OF OSC1 INPUT (IN XT, HS, AND LP MODES) vs. V_{DD}

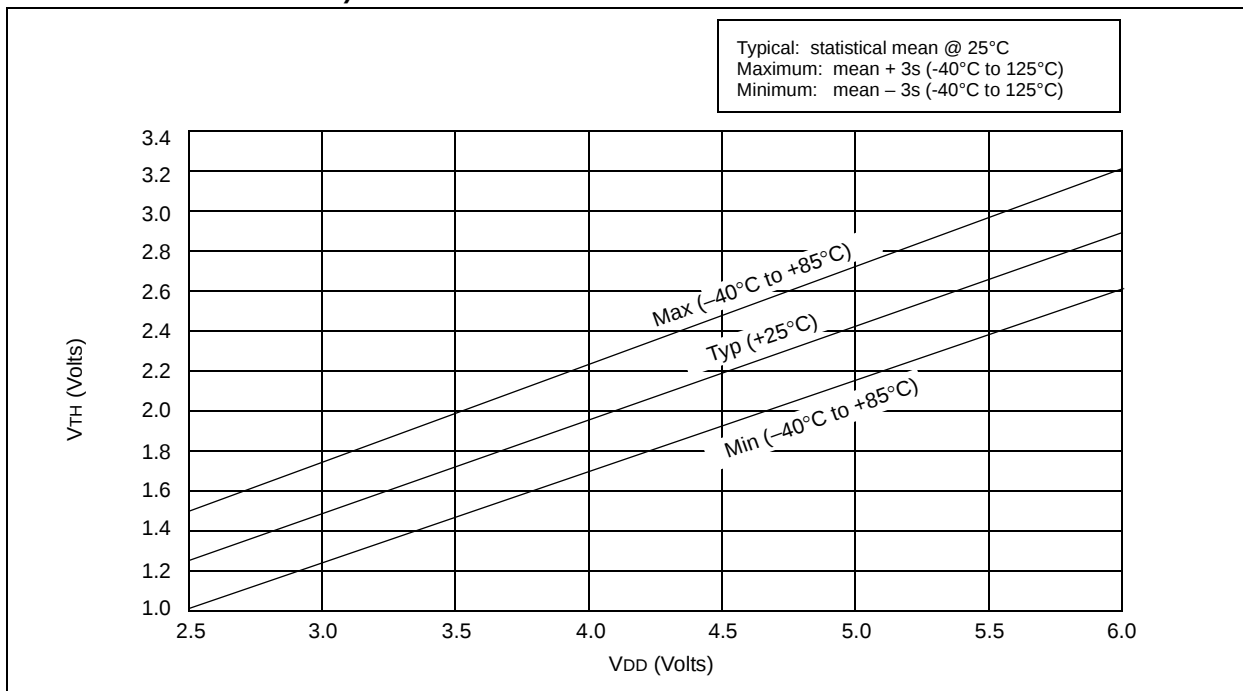


FIGURE 16-14: TYPICAL I_{DD} vs. FREQUENCY (WDT DISABLED, RC MODE @ 300 pF, 25°C)

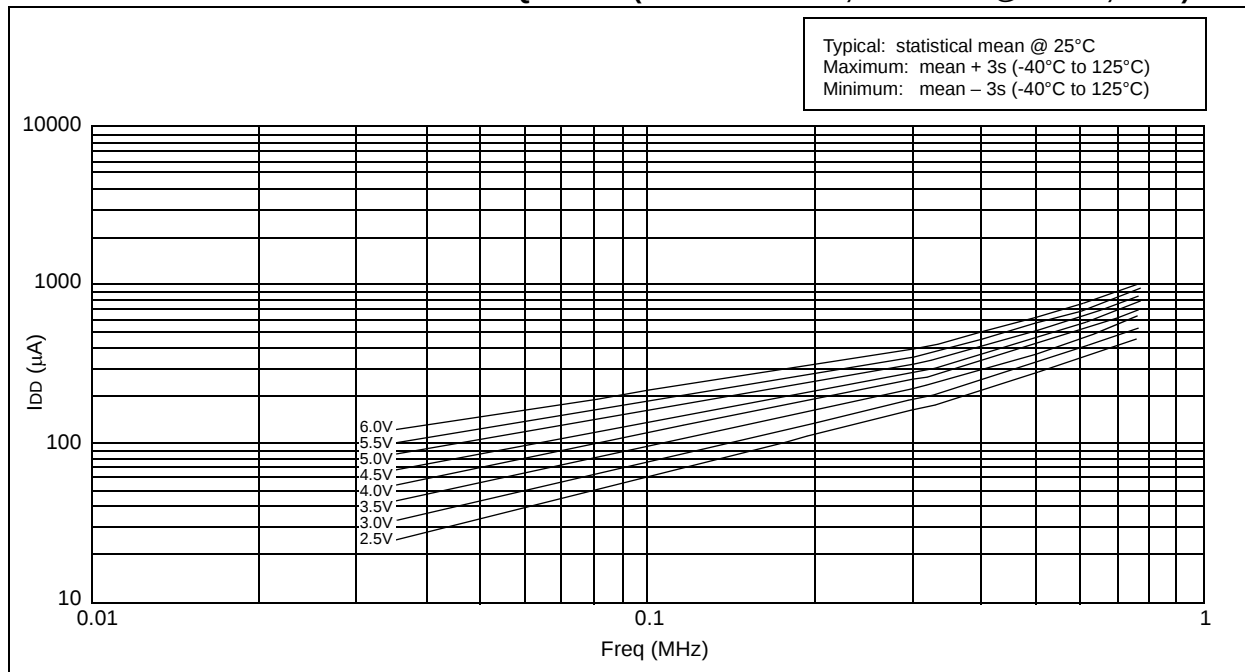


FIGURE 16-15: MAXIMUM I_{DD} vs. FREQUENCY (WDT DISABLED, RC MODE @ 300 pF, -40°C to +85°C)

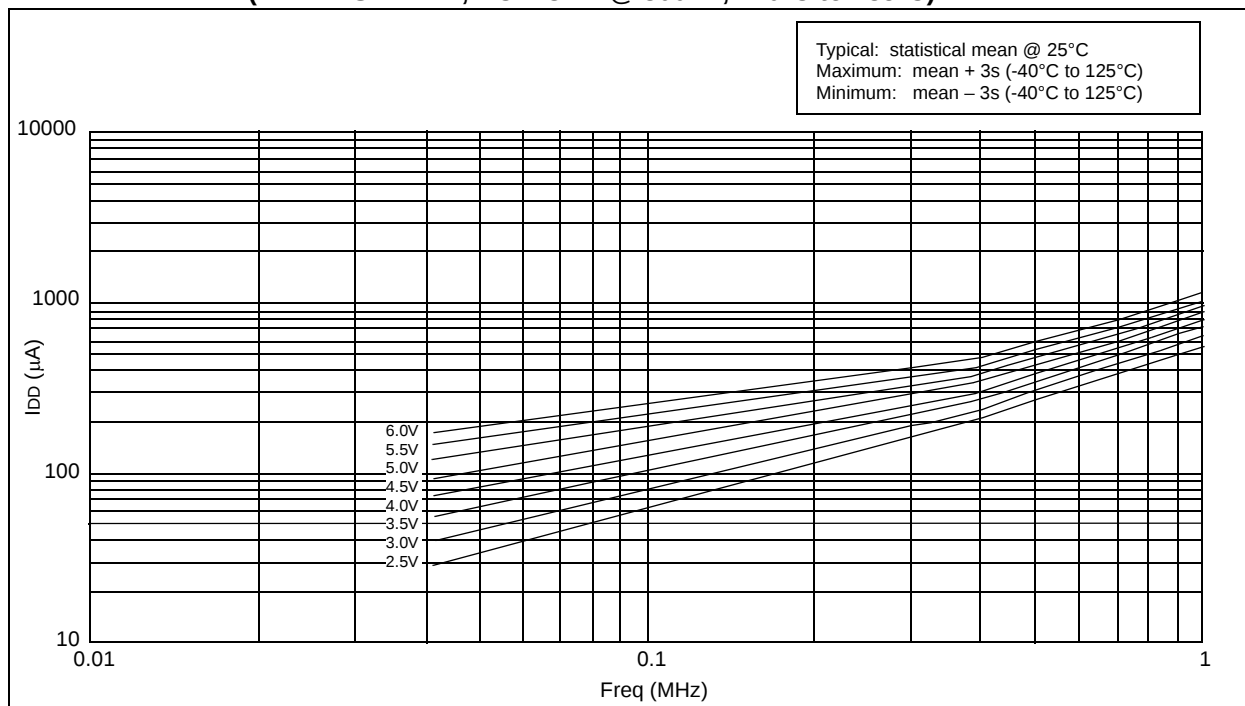


FIGURE 16-20: PORTA, B AND C I_{OH} vs. V_{OH} , $V_{DD} = 3V$

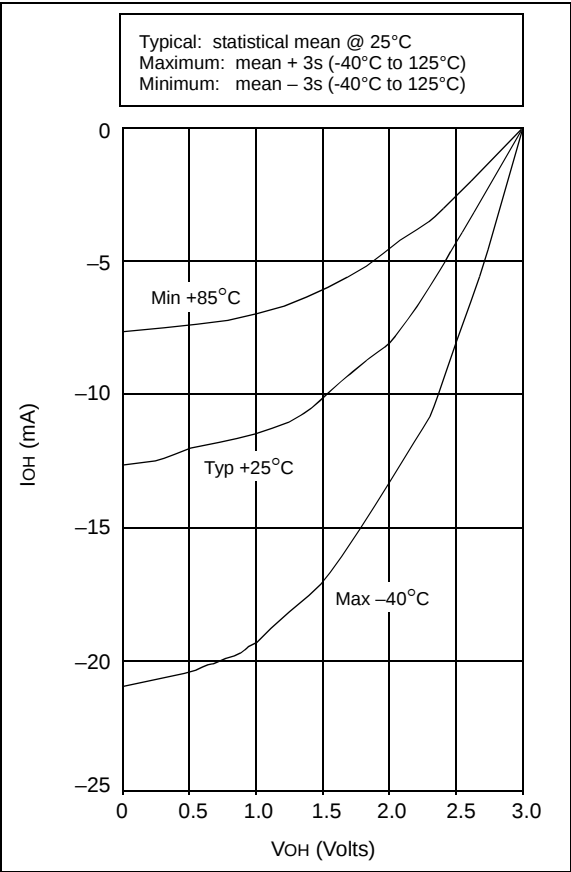


FIGURE 16-21: PORTA, B AND C I_{OH} vs. V_{OH} , $V_{DD} = 5V$

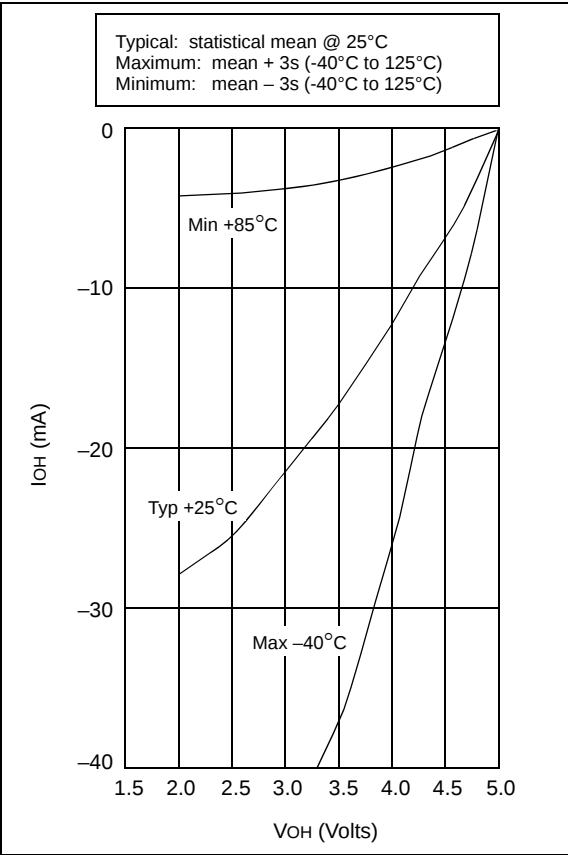


FIGURE 18-14: WDT TIMER TIME-OUT PERIOD vs. VDD⁽¹⁾

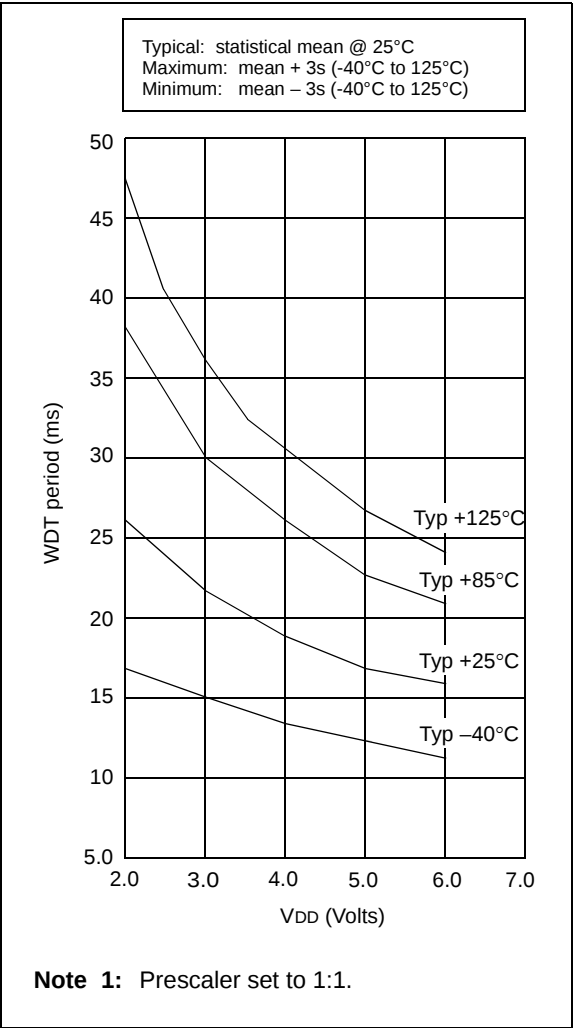


FIGURE 18-15: PORTA, B AND C IOH vs. VOH, VDD = 3 V

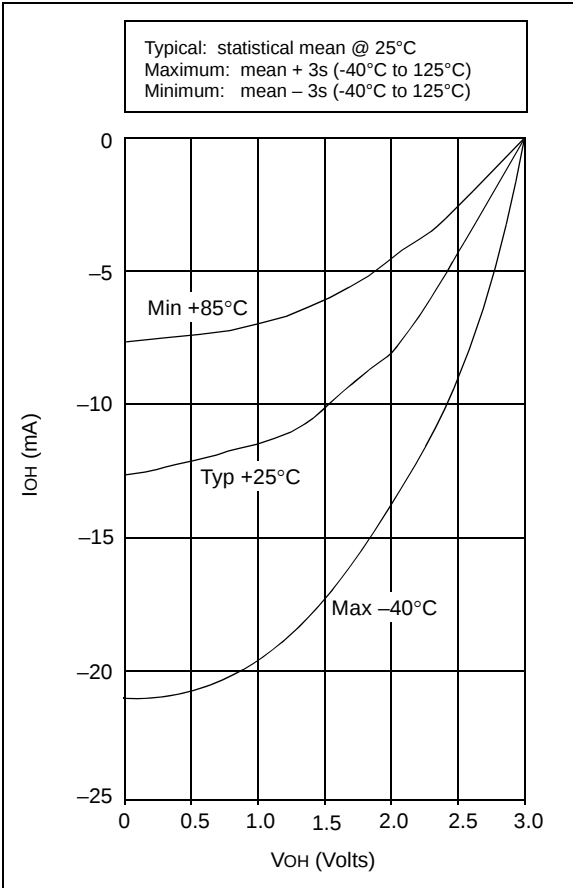


FIGURE 19-4: CLKOUT AND I/O TIMING - PIC16C5X-40

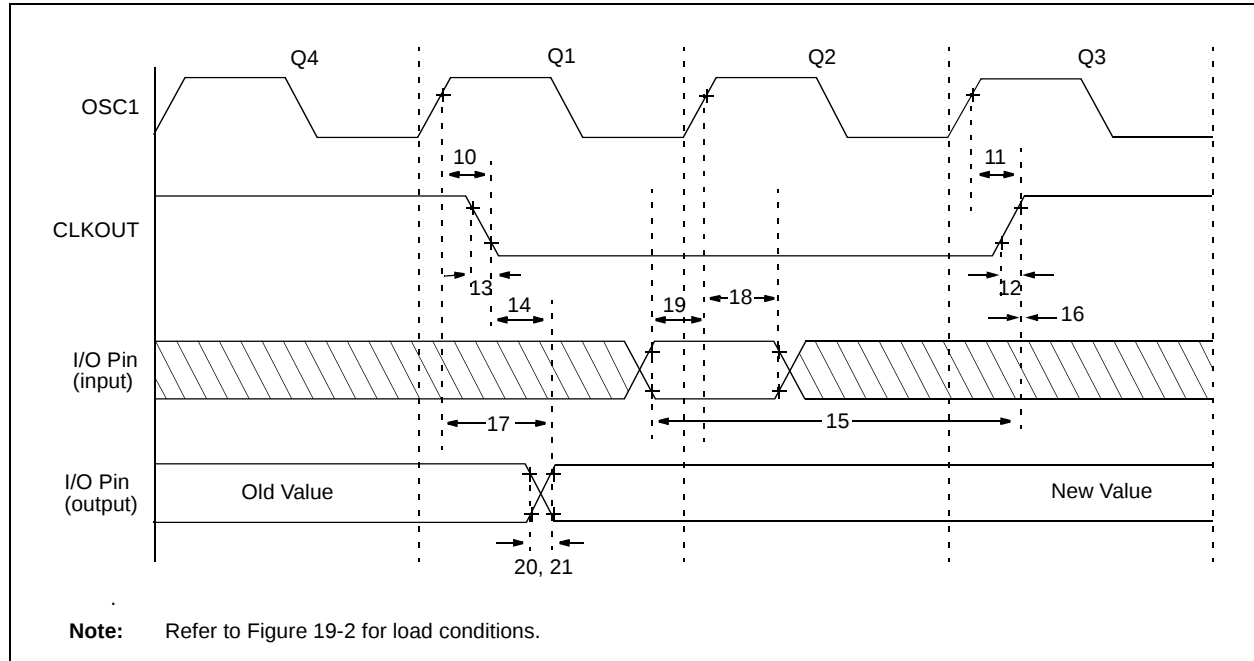


TABLE 19-2: CLKOUT AND I/O TIMING REQUIREMENTS - PIC16C5X-40

AC Characteristics		Standard Operating Conditions (unless otherwise specified)				
		Operating Temperature 0°C ≤ TA ≤ +70°C for commercial				
Param No.	Symbol	Characteristic	Min	Typ†	Max	Units
10	TosH2ckL	OSC1↑ to CLKOUT↓ ^(1,2)	—	15	30**	ns
11	TosH2ckH	OSC1↑ to CLKOUT↑ ^(1,2)	—	15	30**	ns
12	TckR	CLKOUT rise time ^(1,2)	—	5.0	15**	ns
13	TckF	CLKOUT fall time ^(1,2)	—	5.0	15**	ns
14	TckL2ioV	CLKOUT↓ to Port out valid ^(1,2)	—	—	40**	ns
15	TioV2ckH	Port in valid before CLKOUT↑ ^(1,2)	0.25 TCY+30*	—	—	ns
16	TckH2ioI	Port in hold after CLKOUT↑ ^(1,2)	0*	—	—	ns
17	TosH2ioV	OSC1↑ (Q1 cycle) to Port out valid ⁽²⁾	—	—	100	ns
18	TosH2ioI	OSC1↑ (Q2 cycle) to Port input invalid (I/O in hold time)	TBD	—	—	ns
19	TioV2osH	Port input valid to OSC1↑ (I/O in setup time)	TBD	—	—	ns
20	TioR	Port output rise time ⁽²⁾	—	10	25**	ns
21	TioF	Port output fall time ⁽²⁾	—	10	25**	ns

* These parameters are characterized but not tested.

** These parameters are design targets and are not tested. No characterization data available at this time.

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

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

2: Refer to Figure 19-2 for load conditions.

FIGURE 20-4: V_{TH} (INPUT THRESHOLD TRIP POINT VOLTAGE) OF I/O PINS vs. V_{DD}

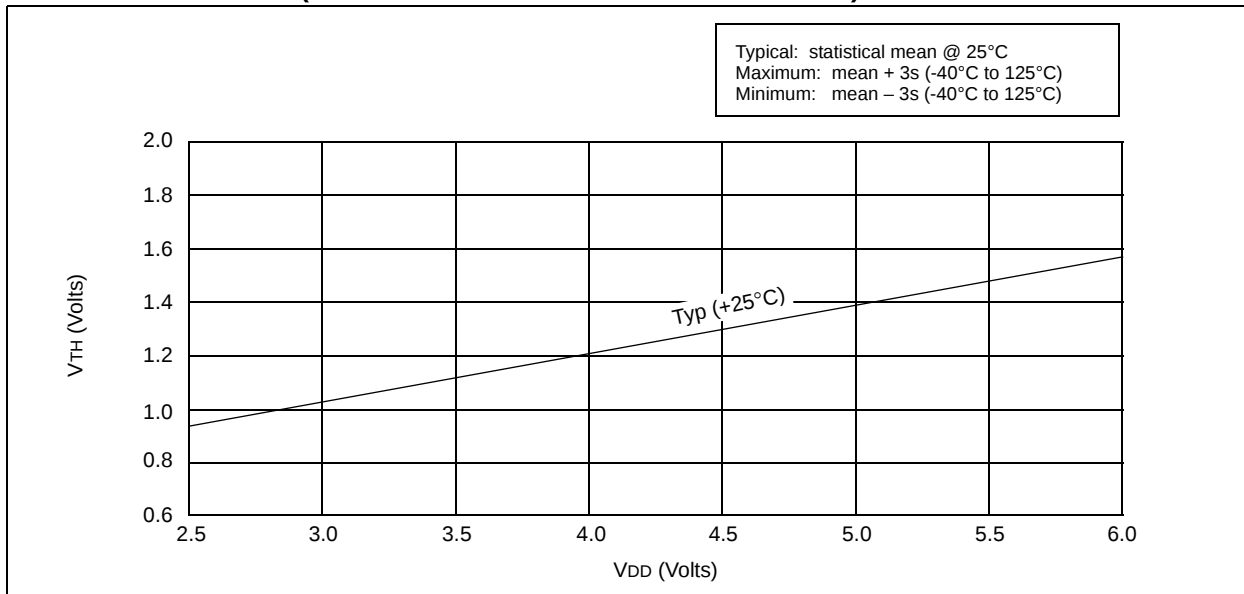


FIGURE 20-5: V_{TH} (INPUT THRESHOLD TRIP POINT VOLTAGE) OF OSC1 INPUT (HS MODE) vs. V_{DD}

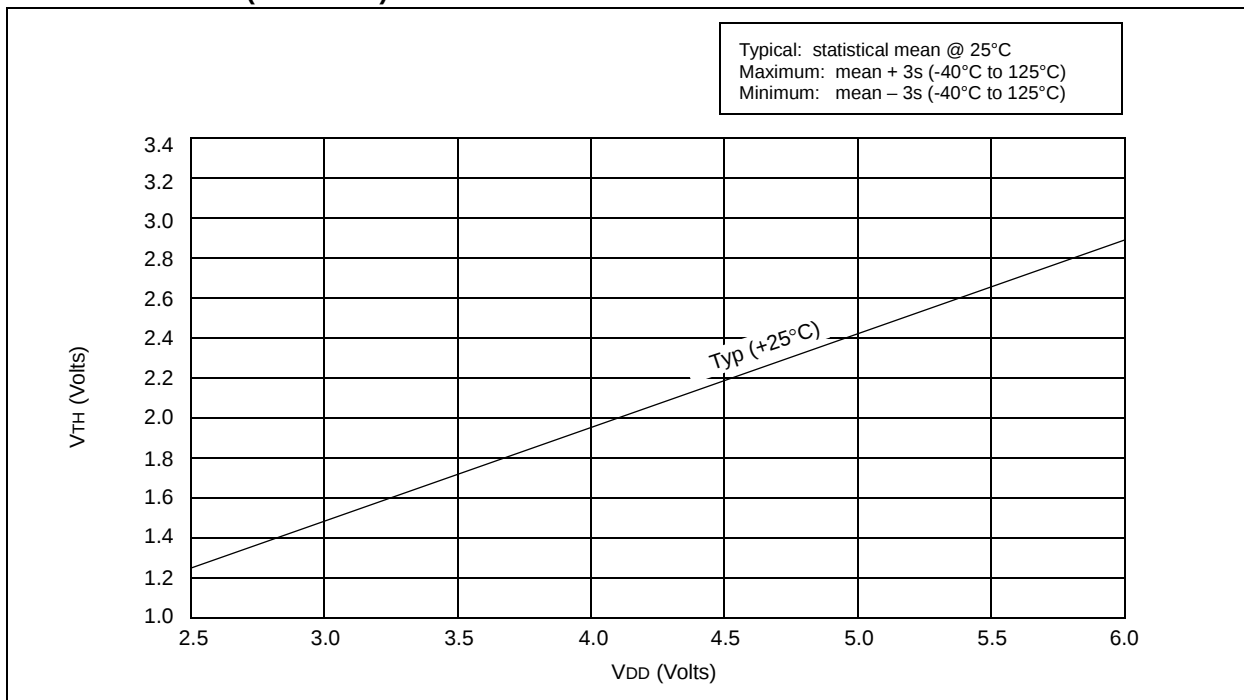
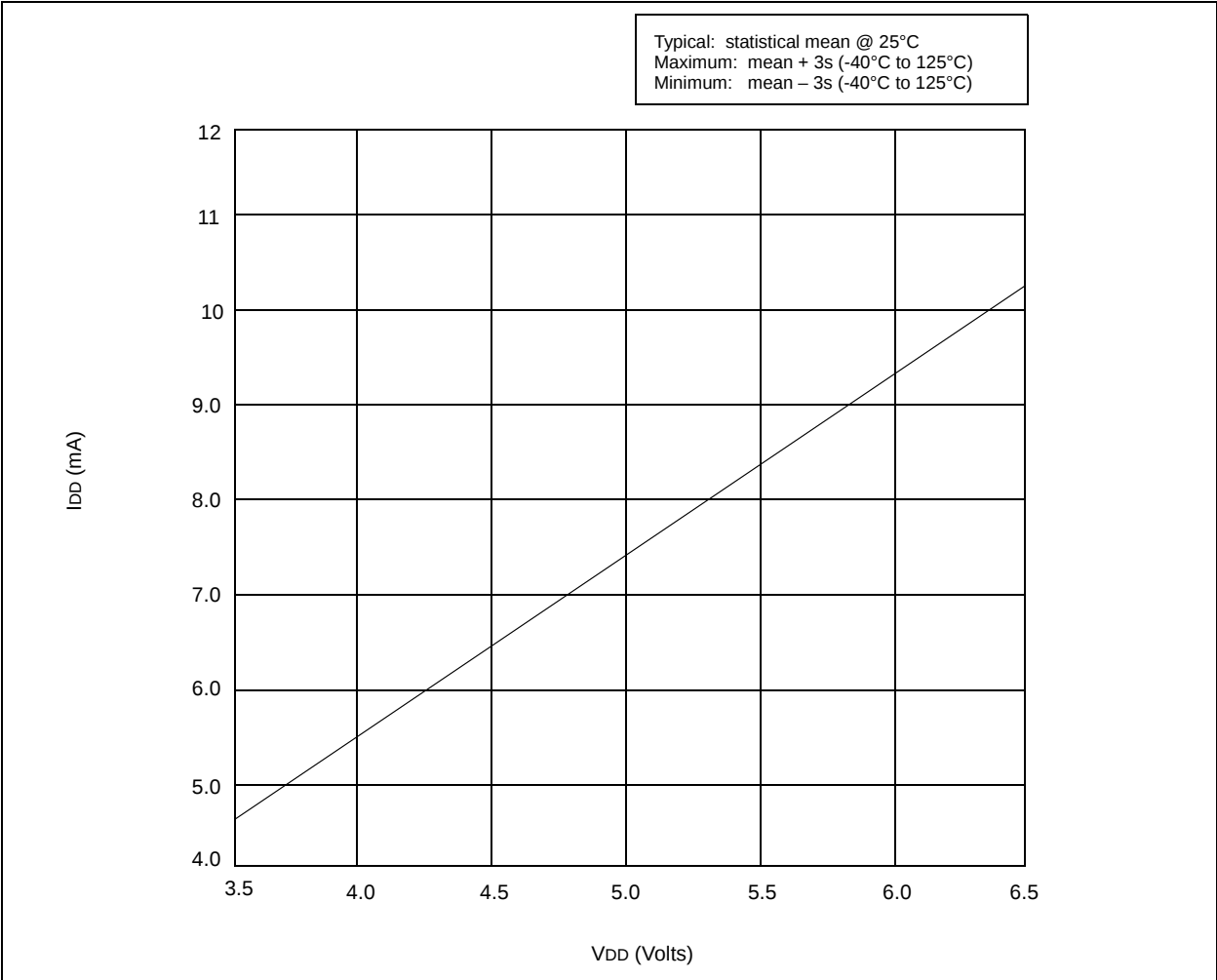


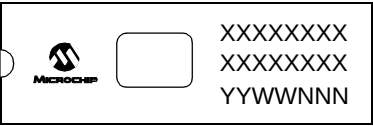
FIGURE 20-6: TYPICAL I_{DD} vs. V_{DD} (40 MHZ, WDT DISABLED, HS MODE, 70°C)



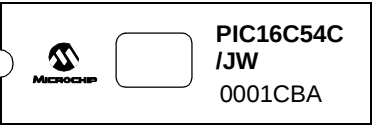
PIC16C5X

Package Marking Information (Cont'd)

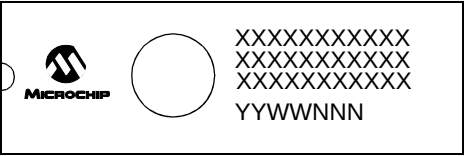
18-Lead Cerdip Windowed



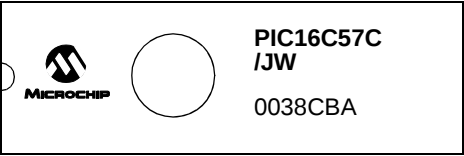
Example



28-Lead Cerdip Windowed



Example



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
	(e3)	Pb-free JEDEC designator for Matte Tin (Sn)
	*	This package is Pb-free. The Pb-free JEDEC designator (e3) can be found on the outer packaging for this package.

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.

ON-LINE SUPPORT

Microchip provides on-line support on the Microchip World Wide Web (WWW) site.

The web site is used by Microchip as a means to make files and information easily available to customers. To view the site, the user must have access to the Internet and a web browser, such as Netscape or Microsoft Explorer. Files are also available for FTP download from our FTP site.

Connecting to the Microchip Internet Web Site

The Microchip web site is available by using your favorite Internet browser to attach to:

www.microchip.com

The file transfer site is available by using an FTP service to connect to:

<ftp://ftp.microchip.com>

The web site and file transfer site provide a variety of services. Users may download files for the latest Development Tools, Data Sheets, Application Notes, User's Guides, Articles and Sample Programs. A variety of Microchip specific business information is also available, including listings of Microchip sales offices, distributors and factory representatives. Other data available for consideration is:

- Latest Microchip Press Releases
- Technical Support Section with Frequently Asked Questions
- Design Tips
- Device Errata
- Job Postings
- Microchip Consultant Program Member Listing
- Links to other useful web sites related to Microchip Products
- Conferences for products, Development Systems, technical information and more
- Listing of seminars and events