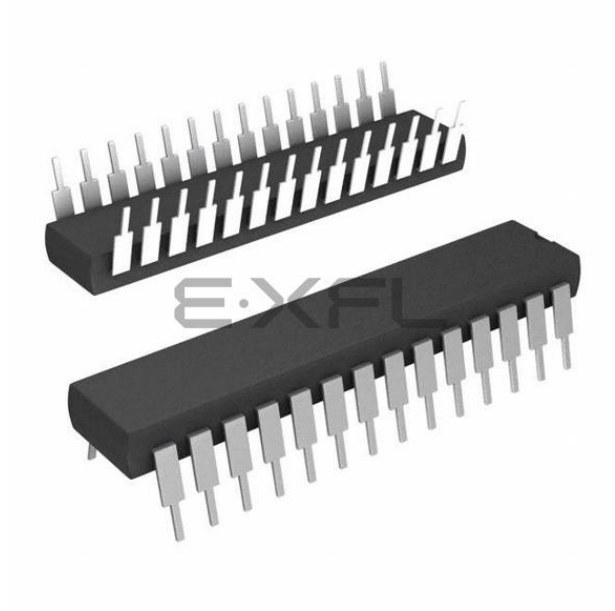


**Welcome to E-XFL.COM**

## 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 | 20MHz |
| 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) | 4.5V ~ 5.5V |
| Data Converters | - |
| Oscillator Type | External |
| Operating Temperature | 0°C ~ 70°C (TA) |
| Mounting Type | Through Hole |
| Package / Case | 28-DIP (0.300", 7.62mm) |
| Supplier Device Package | 28-SPDIP |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/pic16c57-hs-sp |

**NOTES:**

**TABLE 3-2: PINOUT DESCRIPTION - PIC16C55, PIC16C57, PIC16CR57**

| Pin Name | Pin Number | | | Pin Type | Buffer Type | Description |
|---|---|---|---|---|---|---|
| | DIP | SOIC | SSOP | | | |
| RA0 | 6 | 6 | 5 | I/O | TTL | Bi-directional I/O port |
| RA1 | 7 | 7 | 6 | I/O | TTL | |
| RA2 | 8 | 8 | 7 | I/O | TTL | |
| RA3 | 9 | 9 | 8 | I/O | TTL | |
| RB0 | 10 | 10 | 9 | I/O | TTL | Bi-directional I/O port |
| RB1 | 11 | 11 | 10 | I/O | TTL | |
| RB2 | 12 | 12 | 11 | I/O | TTL | |
| RB3 | 13 | 13 | 12 | I/O | TTL | |
| RB4 | 14 | 14 | 13 | I/O | TTL | |
| RB5 | 15 | 15 | 15 | I/O | TTL | |
| RB6 | 16 | 16 | 16 | I/O | TTL | |
| RB7 | 17 | 17 | 17 | I/O | TTL | |
| RC0 | 18 | 18 | 18 | I/O | TTL | Bi-directional I/O port |
| RC1 | 19 | 19 | 19 | I/O | TTL | |
| RC2 | 20 | 20 | 20 | I/O | TTL | |
| RC3 | 21 | 21 | 21 | I/O | TTL | |
| RC4 | 22 | 22 | 22 | I/O | TTL | |
| RC5 | 23 | 23 | 23 | I/O | TTL | |
| RC6 | 24 | 24 | 24 | I/O | TTL | |
| RC7 | 25 | 25 | 25 | I/O | TTL | |
| T0CKI | 1 | 1 | 2 | I | ST | Clock input to Timer0. Must be tied to $V_{SS}$ or $V_{DD}$, if not in use, to reduce current consumption. |
| $\overline{MCLR}$ | 28 | 28 | 28 | I | ST | Master clear (RESET) input. This pin is an active low RESET to the device. |
| OSC1/CLKIN | 27 | 27 | 27 | I | ST | Oscillator crystal input/external clock source input. |
| OSC2/CLKOUT | 26 | 26 | 26 | 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. |
| $V_{DD}$ | 2 | 2 | 3,4 | P | — | Positive supply for logic and I/O pins. |
| $V_{SS}$ | 4 | 4 | 1,14 | P | — | Ground reference for logic and I/O pins. |
| N/C | 3,5 | 3,5 | — | — | — | Unused, do not connect. |

Legend: I = input, O = output, I/O = input/output, P = power, — = Not Used, TTL = TTL input, ST = Schmitt Trigger input

## 3.1 Clocking Scheme/Instruction Cycle

The clock input (OSC1/CLKIN pin) is internally divided by four to generate four non-overlapping quadrature clocks, namely Q1, Q2, Q3 and Q4. Internally, the program counter is incremented every Q1 and the instruction is fetched from program memory and latched into the instruction register in Q4. It is decoded and executed during the following Q1 through Q4. The clocks and instruction execution flow are shown in Figure 3-2 and Example 3-1.

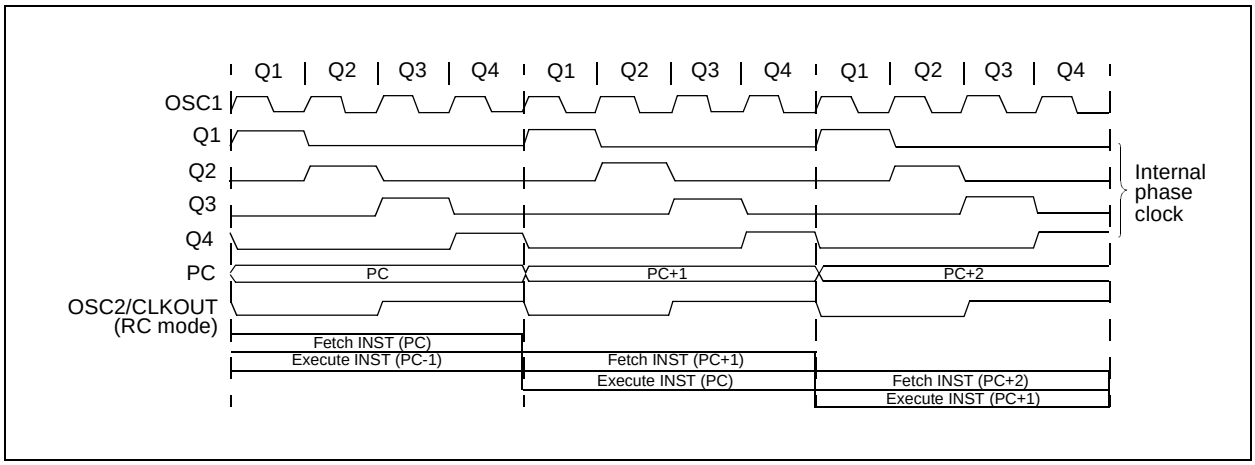
## 3.2 Instruction Flow/Pipelining

An Instruction Cycle consists of four Q cycles (Q1, Q2, Q3 and Q4). The instruction fetch and execute are pipelined such that fetch takes one instruction cycle, while decode and execute takes another instruction cycle. However, due to the pipelining, each instruction effectively executes in one cycle. If an instruction causes the program counter to change (e.g., `GOTO`), then two cycles are required to complete the instruction (Example 3-1).

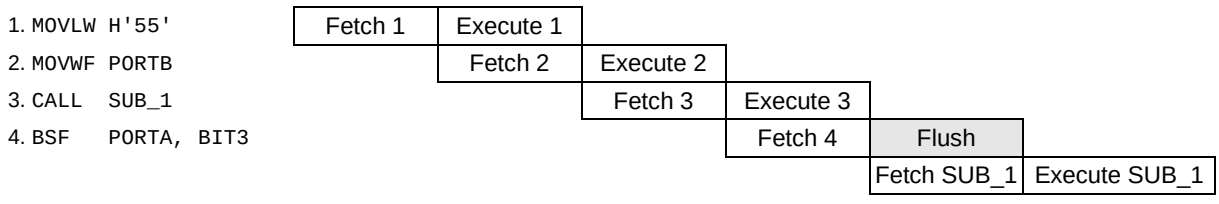
A fetch cycle begins with the program counter (PC) incrementing in Q1.

In the execution cycle, the fetched instruction is latched into the Instruction Register in cycle Q1. This instruction is then decoded and executed during the Q2, Q3 and Q4 cycles. Data memory is read during Q2 (operand read) and written during Q4 (destination write).

**FIGURE 3-2: CLOCK/INSTRUCTION CYCLE**




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




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

**NOTES:**

## 6.2 Data Memory Organization

Data memory is composed of registers, or bytes of RAM. Therefore, data memory for a device is specified by its register file. The register file is divided into two functional groups: Special Function Registers and General Purpose Registers.

The Special Function Registers include the TMR0 register, the Program Counter (PC), the Status Register, the I/O registers (ports) and the File Select Register (FSR). In addition, Special Purpose Registers are used to control the I/O port configuration and prescaler options.

The General Purpose Registers are used for data and control information under command of the instructions.

For the PIC16C54, PIC16CR54, PIC16C56 and PIC16CR56, the register file is composed of 7 Special Function Registers and 25 General Purpose Registers (Figure 6-4).

For the PIC16C55, the register file is composed of 8 Special Function Registers and 24 General Purpose Registers.

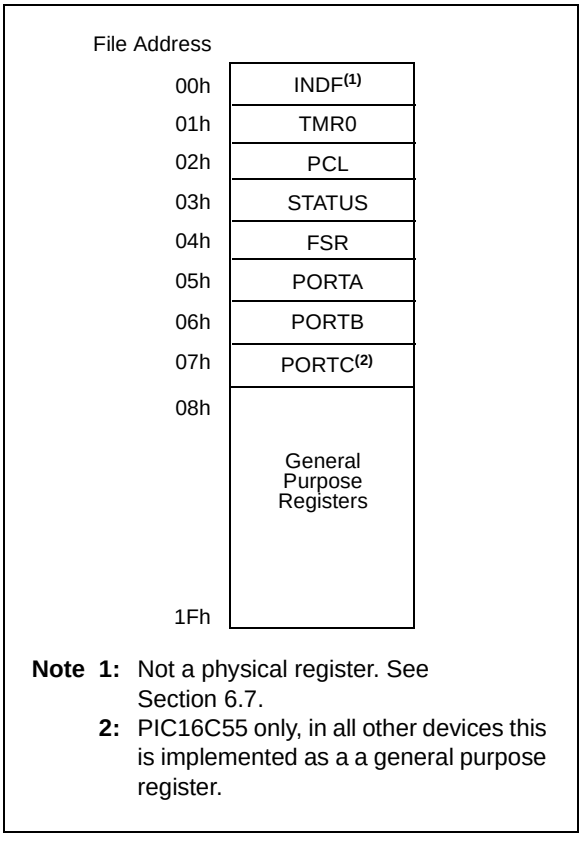
For the PIC16C57 and PIC16CR57, the register file is composed of 8 Special Function Registers, 24 General Purpose Registers and up to 48 additional General Purpose Registers that may be addressed using a banking scheme (Figure 6-5).

For the PIC16C58 and PIC16CR58, the register file is composed of 7 Special Function Registers, 25 General Purpose Registers and up to 48 additional General Purpose Registers that may be addressed using a banking scheme (Figure 6-6).

### 6.2.1 GENERAL PURPOSE REGISTER FILE

The register file is accessed either directly or indirectly through the File Select Register (FSR). The FSR Register is described in Section 6.7.

**FIGURE 6-4: PIC16C54, PIC16CR54, PIC16C55, PIC16C56, PIC16CR56 REGISTER FILE MAP**




**Note 1:** Not a physical register. See Section 6.7.

**2:** PIC16C55 only, in all other devices this is implemented as a a general purpose register.

**NOTES:**

**FIGURE 8-3: TIMER0 TIMING: INTERNAL CLOCK/NO PRESCALER**

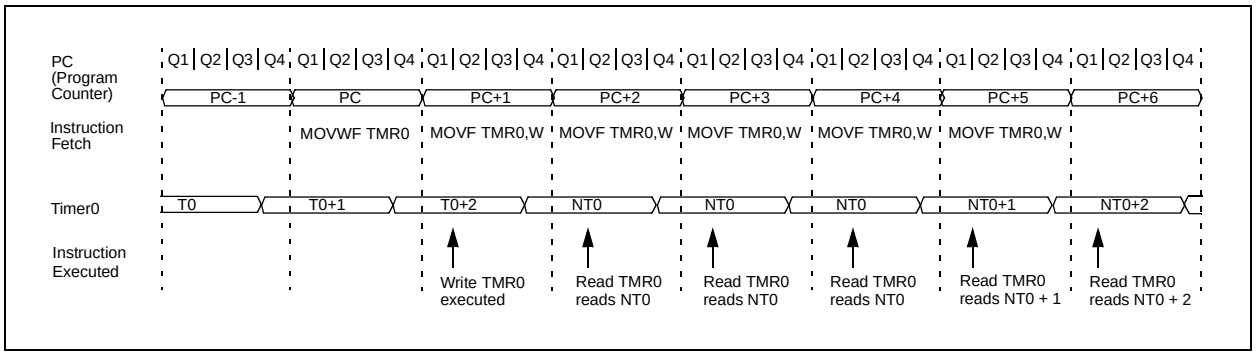



**FIGURE 8-4: TIMER0 TIMING: INTERNAL CLOCK/PRESCALER 1:2**

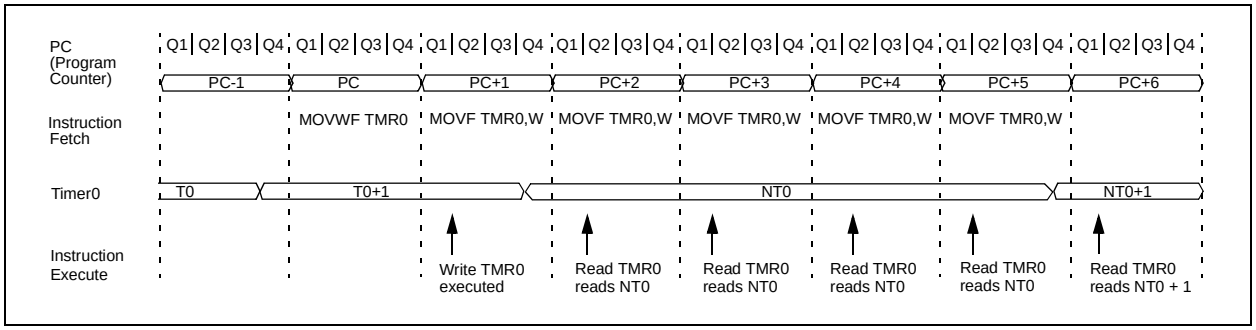



**TABLE 8-1: REGISTERS ASSOCIATED WITH TIMER0**

| 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 MCLR and WDT Reset |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 01h | TMR0 | Timer0 - 8-bit real-time clock/counter | | | | | | | | xxxx xxxx | uuuu uuuu |
| N/A | OPTION | — | — | T0CS | T0SE | PSA | PS2 | PS1 | PS0 | --11 1111 | --11 1111 |

Legend: x = unknown, u = unchanged, - = unimplemented. Shaded cells not used by Timer0.

**RLF** **Rotate Left f through Carry**

Syntax: [ *label* ] RLF f,d

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

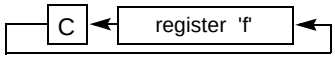
Operation: See description below

Status Affected: C

Encoding:

| 0011 | 01df | ffff |
|---|---|---|

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




Words: 1

Cycles: 1

Example:
```
RLF    REG1,0
```
Before Instruction
```
REG1    =  1110 0110
C       =  0
```
After Instruction
```
REG1    =  1110 0110
W       =  1100 1100
C       =  1
```

**RRF** **Rotate Right f through Carry**

Syntax: [ *label* ] RRF f,d

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

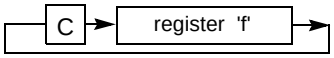
Operation: See description below

Status Affected: C

Encoding:

| 0011 | 00df | ffff |
|---|---|---|

Description: The contents of register 'f' are rotated one bit to the right through the Carry Flag (STATUS<0>). 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:
```
RRF    REG1,0
```
Before Instruction
```
REG1    =  1110 0110
C       =  0
```
After Instruction
```
REG1    =  1110 0110
W       =  0111 0011
C       =  0
```

**SLEEP** **Enter SLEEP Mode**

Syntax: [*label*] SLEEP

Operands: None

Operation: 00h → WDT;
0 → WDT prescaler; if assigned
1 → $\overline{TO}$;
0 → $\overline{PD}$

Status Affected: $\overline{TO}$, $\overline{PD}$

Encoding:

| 0000 | 0000 | 0011 |
|---|---|---|

Description: Time-out status bit ($\overline{TO}$) is set. The power-down status bit ($\overline{PD}$) is cleared. The WDT and its prescaler are cleared.
The processor is put into SLEEP mode with the oscillator stopped. See section on SLEEP for more details.

Words: 1

Cycles: 1

Example:
```
SLEEP
```

**SUBWF** **Subtract W from f**

Syntax: [*label*] SUBWF f,d

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

Operation: (f) – (W) → (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: $0 \le f \le 31$
$d \in [0,1]$

Operation: (f<3:0>) → (dest<7:4>);
(f<7:4>) → (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 or 7

Operation: (W) → TRIS register f

Status Affected: None

Encoding:

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

Description: TRIS register 'f' (f = 5, 6, 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

## 11.8 MPLAB ICD In-Circuit Debugger

Microchip's In-Circuit Debugger, MPLAB ICD, is a powerful, low cost, run-time development tool. This tool is based on the FLASH PIC MCUs and can be used to develop for this and other PIC microcontrollers. The MPLAB ICD utilizes the in-circuit debugging capability built into the FLASH devices. This feature, along with Microchip's In-Circuit Serial Programming™ protocol, offers cost-effective in-circuit FLASH debugging from the graphical user interface of the MPLAB Integrated Development Environment. This enables a designer to develop and debug source code by watching variables, single-stepping and setting break points. Running at full speed enables testing hardware in real-time.

## 11.9 PRO MATE II Universal Device Programmer

The PRO MATE II universal device programmer is a full-featured programmer, capable of operating in Stand-alone mode, as well as PC-hosted mode. The PRO MATE II device programmer is CE compliant.

The PRO MATE II device programmer has programmable $V_{DD}$ and $V_{PP}$ supplies, which allow it to verify programmed memory at $V_{DD}$ min and $V_{DD}$ max for maximum reliability. It has an LCD display for instructions and error messages, keys to enter commands and a modular detachable socket assembly to support various package types. In Stand-alone mode, the PRO MATE II device programmer can read, verify, or program PIC devices. It can also set code protection in this mode.

## 11.10 PICSTART Plus Entry Level Development Programmer

The PICSTART Plus development programmer is an easy-to-use, low cost, prototype programmer. It connects to the PC via a COM (RS-232) port. MPLAB Integrated Development Environment software makes using the programmer simple and efficient.

The PICSTART Plus development programmer supports all PIC devices with up to 40 pins. Larger pin count devices, such as the PIC16C92X and PIC17C76X, may be supported with an adapter socket. The PICSTART Plus development programmer is CE compliant.

## 11.11 PICDEM 1 Low Cost PIC MCU Demonstration Board

The PICDEM 1 demonstration board is a simple board which demonstrates the capabilities of several of Microchip's microcontrollers. The microcontrollers supported are: PIC16C5X (PIC16C54 to PIC16C58A), PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 and PIC17C44. All necessary hardware and software is included to run basic demo programs. The user can program the sample microcontrollers provided with the PICDEM 1 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The user can also connect the PICDEM 1 demonstration board to the MPLAB ICE in-circuit emulator and download the firmware to the emulator for testing. A prototype area is available for the user to build some additional hardware and connect it to the microcontroller socket(s). Some of the features include an RS-232 interface, a potentiometer for simulated analog input, push button switches and eight LEDs connected to PORTB.

## 11.12 PICDEM 2 Low Cost PIC16CXX Demonstration Board

The PICDEM 2 demonstration board is a simple demonstration board that supports the PIC16C62, PIC16C64, PIC16C65, PIC16C73 and PIC16C74 microcontrollers. All the necessary hardware and software is included to run the basic demonstration programs. The user can program the sample microcontrollers provided with the PICDEM 2 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer, and easily test firmware. The MPLAB ICE in-circuit emulator may also be used with the PICDEM 2 demonstration board to test firmware. A prototype area has been provided to the user for adding additional hardware and connecting it to the microcontroller socket(s). Some of the features include a RS-232 interface, push button switches, a potentiometer for simulated analog input, a serial EEPROM to demonstrate usage of the $I^2C$™ bus and separate headers for connection to an LCD module and a keypad.

## 12.0 ELECTRICAL CHARACTERISTICS - PIC16C54A

**Absolute Maximum Ratings(†)**

Ambient Temperature under bias ..... –55°C to +125°C

Storage Temperature ..... –65°C to +150°C

Voltage on $V_{DD}$ with respect to $V_{SS}$ .....0V to +7.5V

Voltage on $\overline{MCLR}$ with respect to $V_{SS}$(1) .....0V to +14V

Voltage on all other pins with respect to $V_{SS}$ .....–0.6V to ($V_{DD}$ + 0.6V)

Total power dissipation(2) ..... 800 mW

Max. current out of $V_{SS}$ pin ..... 150 mA

Max. current into $V_{DD}$ pin ..... 100 mA

Max. current into an input pin (T0CKI only)..... ±500 μA

Input clamp current, $I_{IK}$ ($V_I$ < 0 or $V_I$ > $V_{DD}$) ..... ±20 mA

Output clamp current, $I_{OK}$ ($V_O$ < 0 or $V_O$ > $V_{DD}$) ..... ±20 mA

Max. output current sunk by any I/O pin ..... 25 mA

Max. output current sourced by any I/O pin ..... 20 mA

Max. output current sourced by a single I/O port (PORTA, B or C) ..... 40 mA

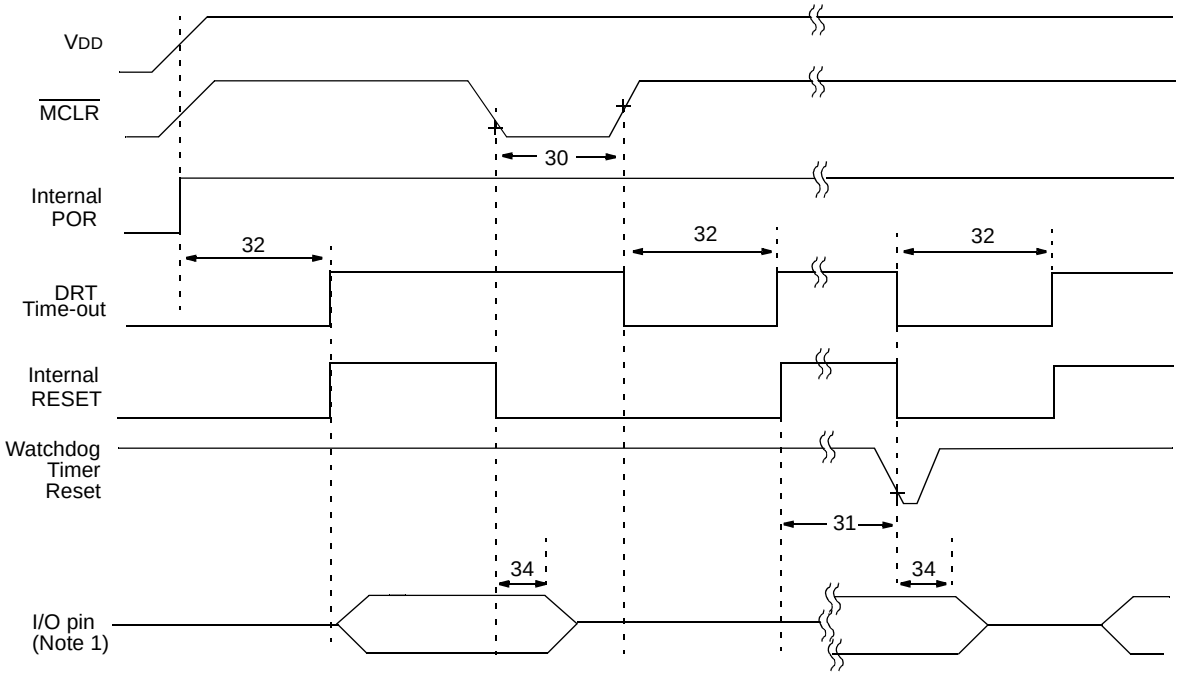
Max. output current sunk by a single I/O port (PORTA, B or C)..... 50 mA

**Note 1:** Voltage spikes below $V_{SS}$ at the $\overline{MCLR}$ pin, inducing currents greater than 80 mA, may cause latch-up. Thus, a series resistor of 50 to 100 Ω should be used when applying a "low" level to the $\overline{MCLR}$ pin rather than pulling this pin directly to $V_{SS}$.

**2:** Power Dissipation is calculated as follows: $P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$

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

**FIGURE 12-4: RESET, WATCHDOG TIMER, AND DEVICE RESET TIMER TIMING - PIC16C54/55/56/57**




**Note 1:** Please refer to Figure 12-1 for load conditions.

**TABLE 12-3: RESET, WATCHDOG TIMER, AND DEVICE RESET TIMER - PIC16C54/55/56/57**

| **AC Characteristics** | **Standard Operating Conditions (unless otherwise specified)**<br>Operating Temperature 0°C ≤ TA ≤ +70°C for commercial<br>–40°C ≤ TA ≤ +85°C for industrial<br>–40°C ≤ TA ≤ +125°C for extended | | | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Symbol** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| 30 | TmcL | $\overline{\text{MCLR}}$ Pulse Width (low) | 100* | — | — | ns | VDD = 5.0V |
| 31 | Twdt | Watchdog Timer Time-out Period (No Prescaler) | 9.0* | 18* | 30* | ms | VDD = 5.0V (Comm) |
| 32 | TDRT | Device Reset Timer Period | 9.0* | 18* | 30* | ms | VDD = 5.0V (Comm) |
| 34 | Tioz | I/O Hi-impedance from $\overline{\text{MCLR}}$ Low | — | — | 100* | ns | |

* 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 14-6: MAXIMUM IPD vs. VDD, WATCHDOG DISABLED**

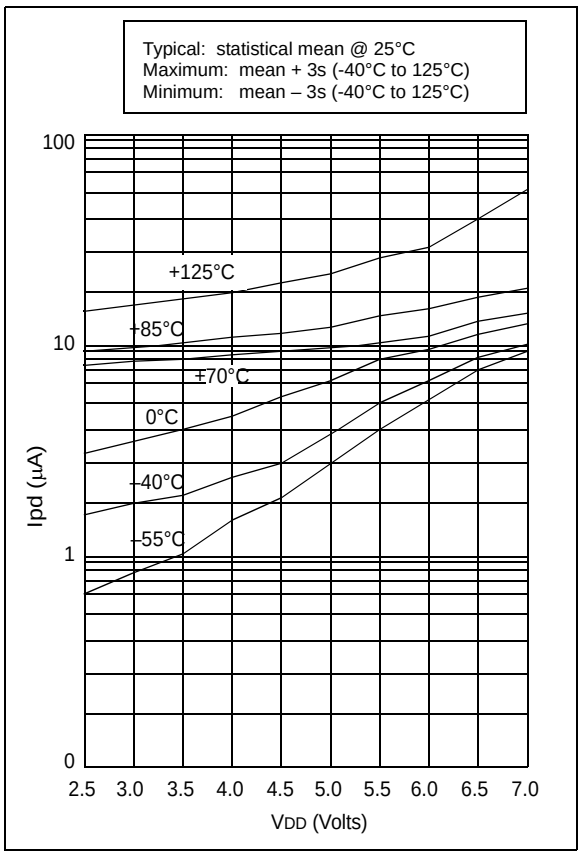



**FIGURE 14-7: TYPICAL IPD vs. VDD, WATCHDOG ENABLED**

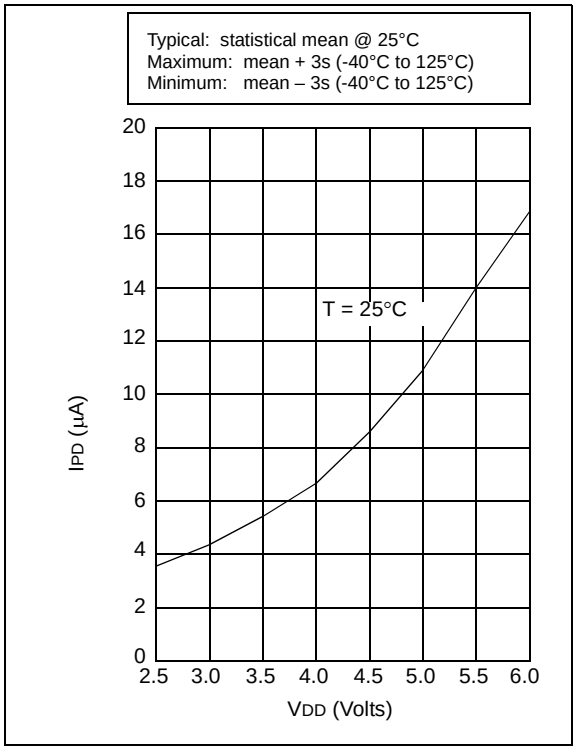
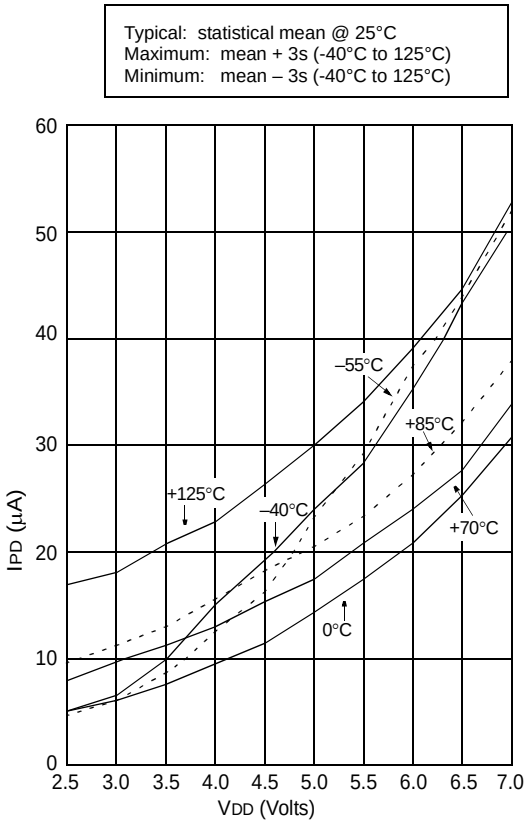



**FIGURE 14-8: MAXIMUM IPD vs. VDD, WATCHDOG ENABLED**




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

## 15.0 ELECTRICAL CHARACTERISTICS - PIC16C54A

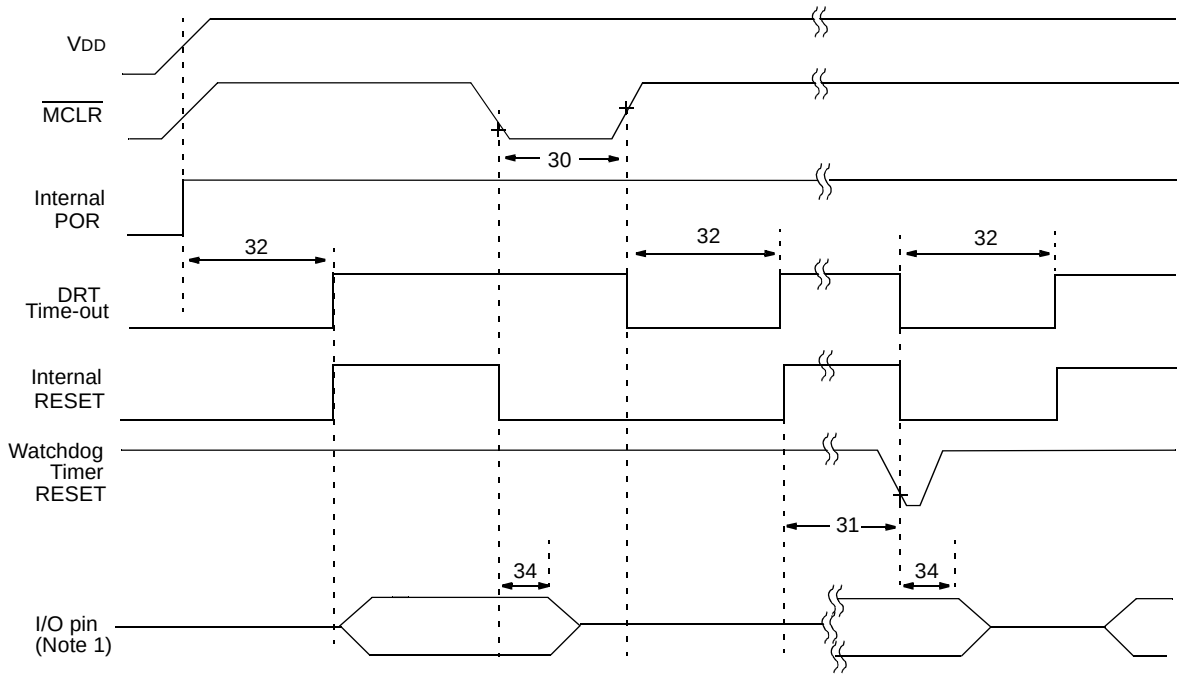
Absolute Maximum Ratings(†)

Ambient temperature under bias ..... –55°C to +125°C
Storage temperature ..... –65°C to +150°C
Voltage on $V_{DD}$ with respect to $V_{SS}$ ..... 0 to +7.5V
Voltage on $\overline{MCLR}$ with respect to $V_{SS}$ ..... 0 to +14V
Voltage on all other pins with respect to $V_{SS}$ ..... –0.6V to ($V_{DD}$ + 0.6V)
Total power dissipation(1) ..... 800 mW
Max. current out of $V_{SS}$ pin ..... 150 mA
Max. current into $V_{DD}$ pin ..... 100 mA
Max. current into an input pin (T0CKI only) ..... ±500 µA
Input clamp current, $I_{IK}$ ($V_I$ < 0 or $V_I$ > $V_{DD}$) ..... ±20 mA
Output clamp current, $I_{OK}$ ($V_O$ < 0 or $V_O$ > $V_{DD}$) ..... ±20 mA
Max. output current sunk by any I/O pin ..... 25 mA
Max. output current sourced by any I/O pin ..... 20 mA
Max. output current sourced by a single I/O port (PORTA or B) ..... 50 mA
Max. output current sunk by a single I/O port (PORTA or B) ..... 50 mA

**Note 1:** Power dissipation is calculated as follows: $Pdis = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD}-V_{OH}) \times I_{OH}\} + \sum(V_{OL} \times I_{OL})$

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

**FIGURE 15-4: RESET, WATCHDOG TIMER, AND DEVICE RESET TIMER TIMING - PIC16C54A**




**Note 1:** Please refer to Figure 15-1 for load conditions.

**TABLE 15-3: RESET, WATCHDOG TIMER, AND DEVICE RESET TIMER - PIC16C54A**

| AC Characteristics | Standard Operating Conditions (unless otherwise specified)<br>Operating Temperature 0°C ≤ TA ≤ +70°C for commercial<br>–40°C ≤ TA ≤ +85°C for industrial<br>–20°C ≤ TA ≤ +85°C for industrial - PIC16LV54A-02I<br>–40°C ≤ TA ≤ +125°C for extended | | | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Symbol** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| 30 | TmcL | $\overline{MCLR}$ Pulse Width (low) | 100*<br>1 | —<br>— | —<br>— | ns<br>µs | VDD = 5.0V<br>VDD = 5.0V (PIC16LV54A only) |
| 31 | Twdt | Watchdog Timer Time-out Period (No Prescaler) | 9.0* | 18* | 30* | ms | VDD = 5.0V (Comm) |
| 32 | TDRT | Device Reset Timer Period | 9.0* | 18* | 30* | ms | VDD = 5.0V (Comm) |
| 34 | Tioz | I/O Hi-impedance from $\overline{MCLR}$ Low | —<br>— | —<br>— | 100*<br>1µs | ns<br>— | <br>(PIC16LV54A only) |

* These parameters are characterized but not tested.

† 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.

**FIGURE 16-2: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD, CEXT = 20 pF, 25°C**

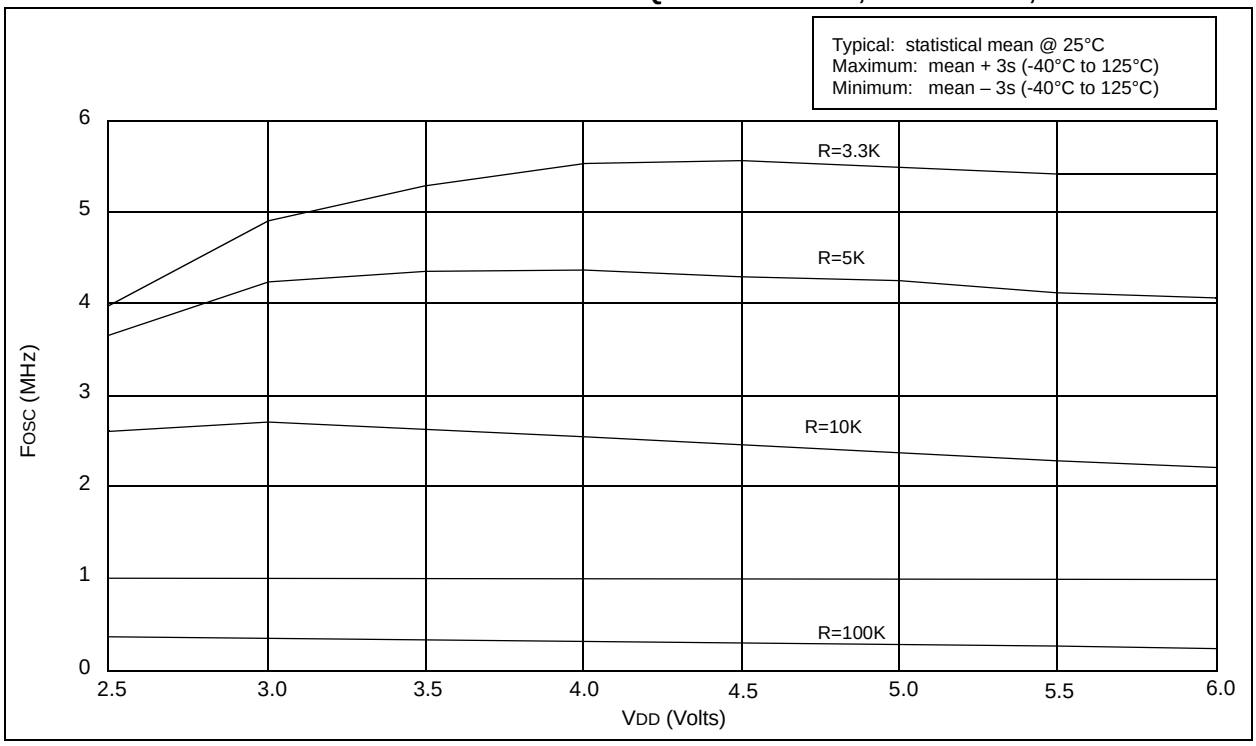



**FIGURE 16-3: TYPICAL RC OSCILLATOR FREQUENCY vs. VDD, CEXT = 100 pF, 25°C**

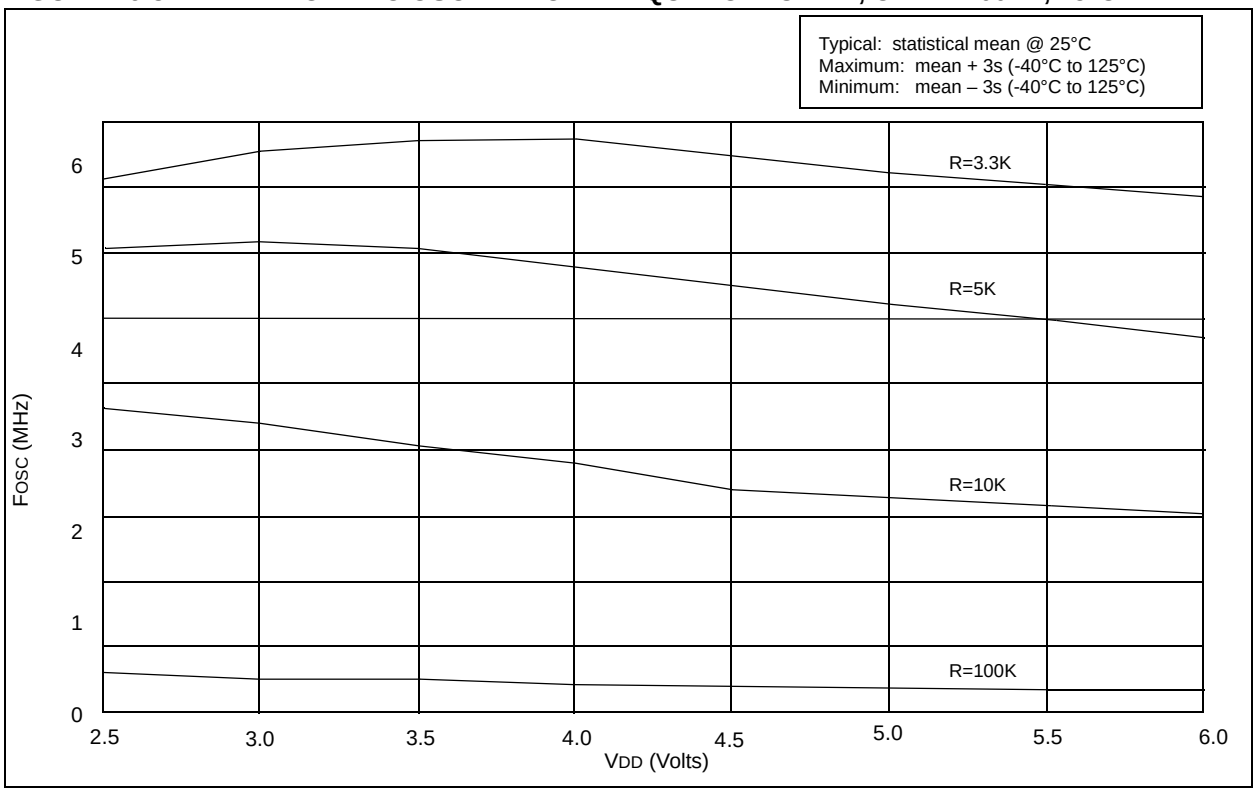
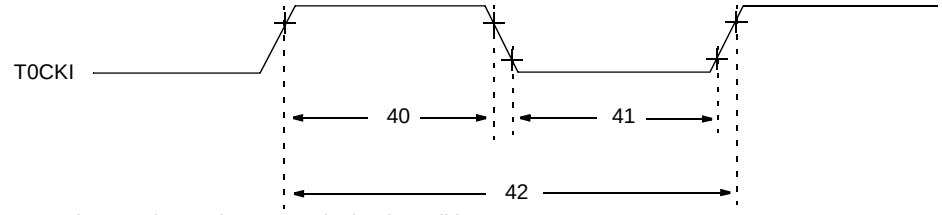


**FIGURE 17-9: TIMER0 CLOCK TIMINGS - PIC16C5X, PIC16CR5X**




**Note:** Please refer to Figure 17-5 for load conditions.

**TABLE 17-4: TIMER0 CLOCK REQUIREMENTS - PIC16C5X, PIC16CR5X**

| AC Characteristics | | | Standard Operating Conditions (unless otherwise specified)<br>Operating Temperature 0°C ≤ TA ≤ +70°C for commercial<br>–40°C ≤ TA ≤ +85°C for industrial<br>–40°C ≤ TA ≤ +125°C for extended | | | | |
|---|---|---|---|---|---|---|---|
| **Param No.** | **Symbol** | **Characteristic** | **Min** | **Typ†** | **Max** | **Units** | **Conditions** |
| 40 | Tt0H | T0CKI High Pulse Width<br>- No Prescaler | 0.5 TCY + 20* | — | — | ns | |
| | | - With Prescaler | 10* | — | — | ns | |
| 41 | Tt0L | T0CKI Low Pulse Width<br>- No Prescaler | 0.5 TCY + 20* | — | — | ns | |
| | | - With Prescaler | 10* | — | — | ns | |
| 42 | Tt0P | T0CKI Period | 20 or $\frac{T_{CY} + 40}{N}$* | — | — | ns | Whichever is greater.<br>N = Prescale Value (1, 2, 4,..., 256) |

* These parameters are characterized but not tested.

† 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.

**FIGURE 18-12: TYPICAL IDD vs. FREQUENCY (WDT DISABLED, RC MODE @ 100 pF, 25°C)**

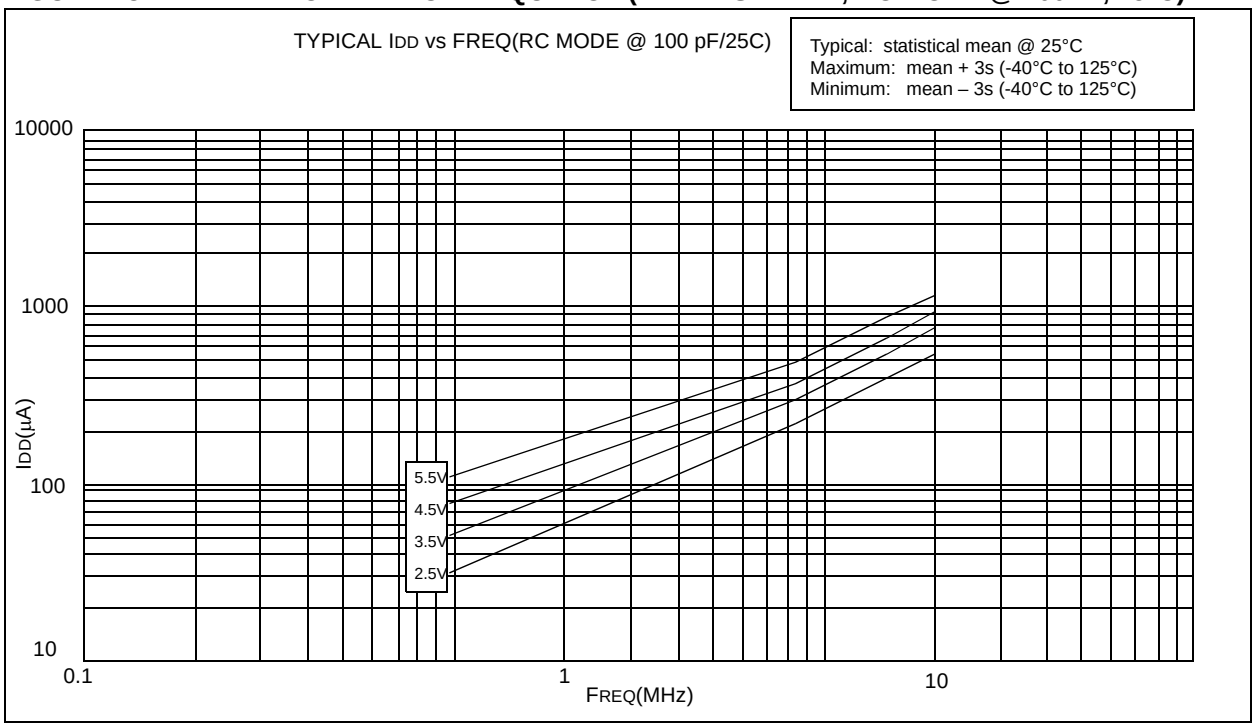



**FIGURE 18-13: TYPICAL IDD vs. FREQUENCY (WDT DISABLED, RC MODE @ 300 pF, 25°C)**

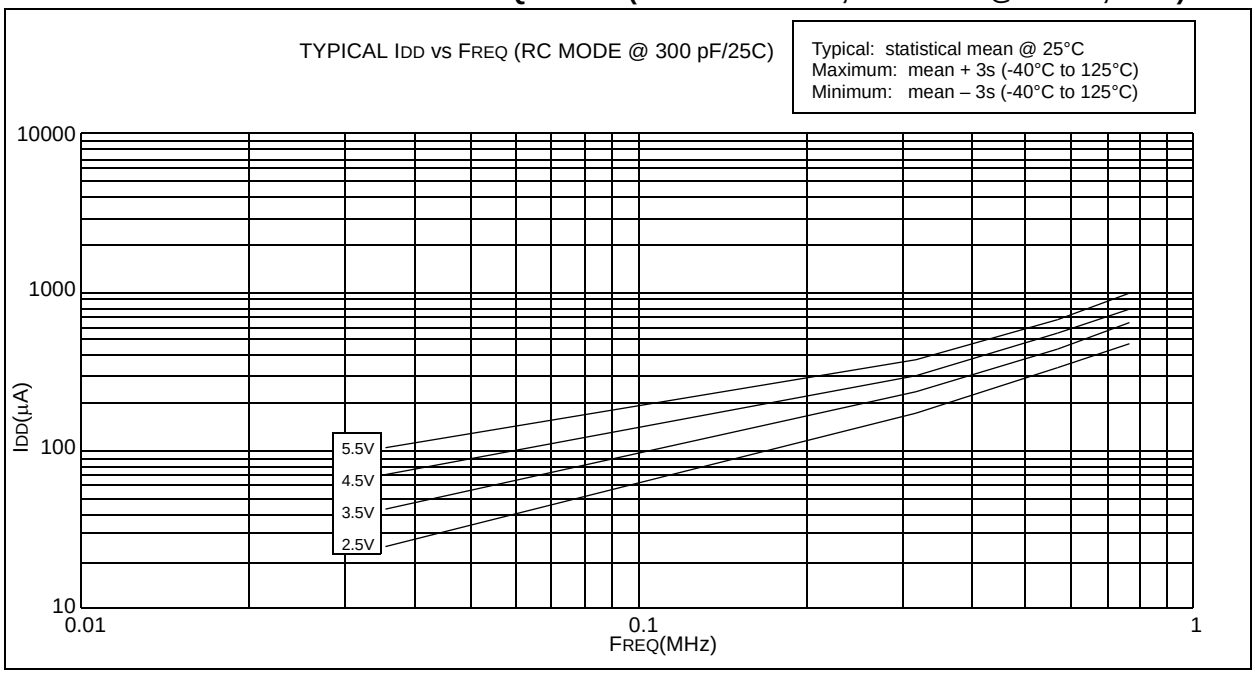
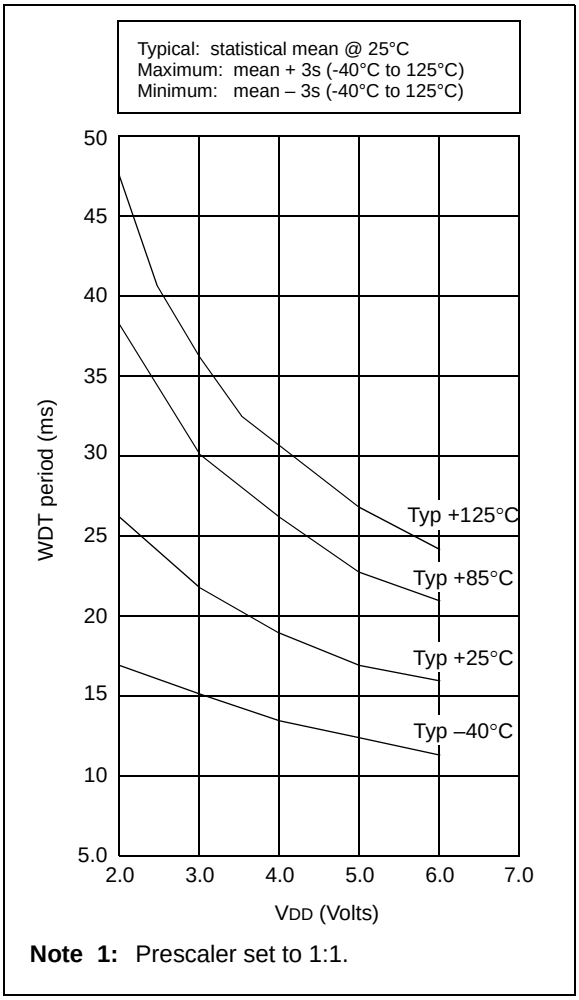


**FIGURE 20-7: WDT TIMER TIME-OUT PERIOD vs. $V_{DD}$(1)**




**Note 1:** Prescaler set to 1:1.

**FIGURE 20-8: $I_{OH}$ vs. $V_{OH}$, $V_{DD}$ = 5 V**

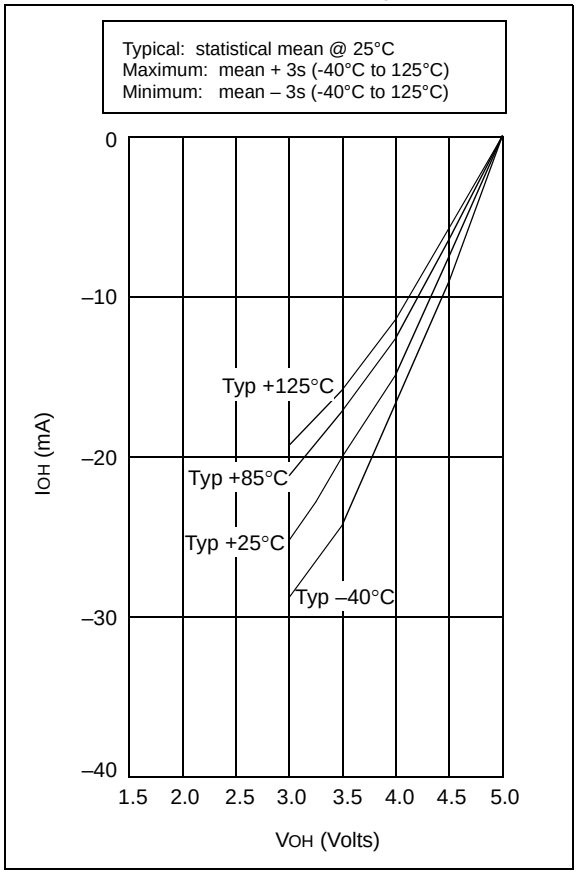



**TABLE 20-1: INPUT CAPACITANCE**

| Pin | Typical Capacitance (pF) | |
|---|---|---|
| | 18L PDIP | 18L SOIC |
| RA port | 5.0 | 4.3 |
| RB port | 5.0 | 4.3 |
| $\overline{MCLR}$ | 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 ±25% (three standard deviations) should be taken into account.

**NOTES:**