

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) | 3V ~ 5.5V |
| Data Converters | - |
| Oscillator Type | External |
| Operating Temperature | -40°C ~ 85°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/pic16c57c-20i-sp |

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.

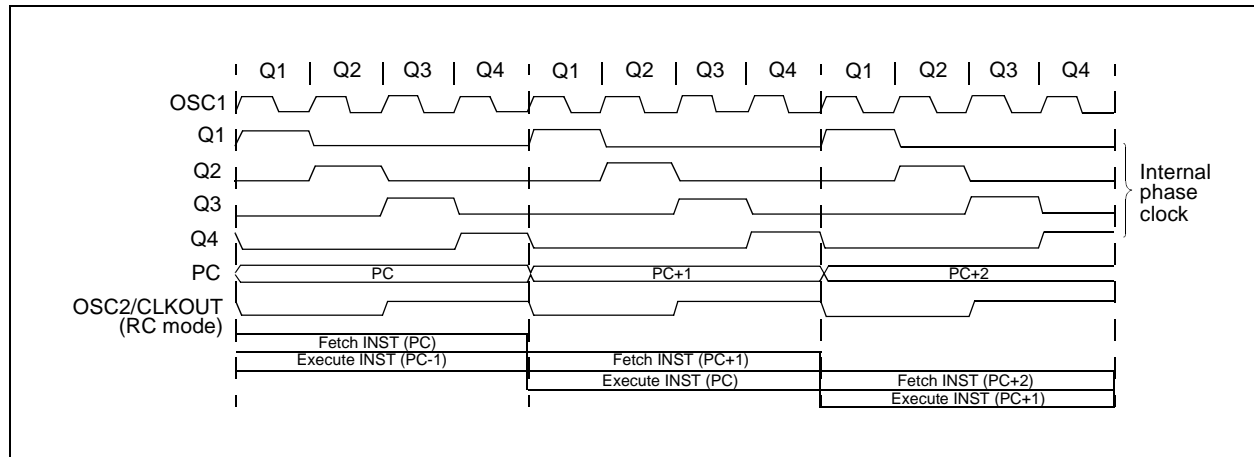
3.2 Instruction Flow/Pipelining

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).

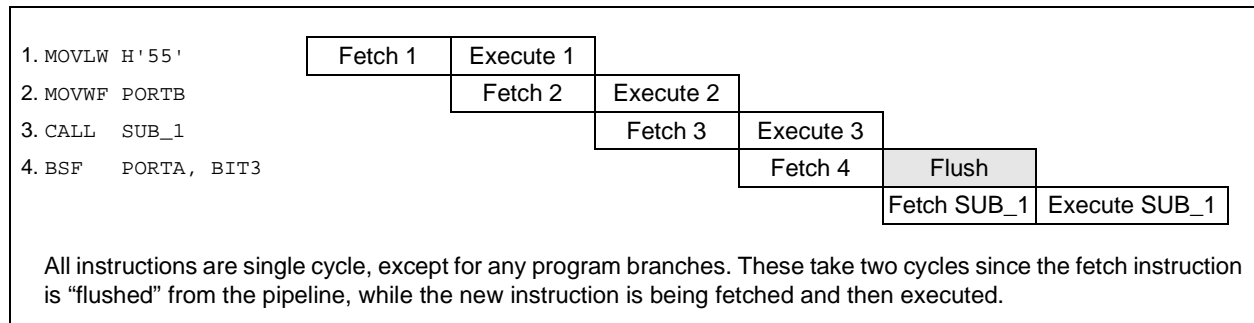
A fetch cycle begins with the program counter (PC) incrementing in Q1.

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



4.3 External Crystal Oscillator Circuit

Either a prepackaged oscillator or a simple oscillator circuit with TTL gates can be used as an external crystal oscillator circuit. Prepackaged oscillators provide a wide operating range and better stability. A well-designed crystal oscillator will provide good performance with TTL gates. Two types of crystal oscillator circuits can be used: one with parallel resonance, or one with series resonance.

Figure 4-3 shows an implementation example of a parallel resonant oscillator circuit. The circuit is designed to use the fundamental frequency of the crystal. The 74AS04 inverter performs the 180-degree phase shift that a parallel oscillator requires. The 4.7 k Ω resistor provides the negative feedback for stability. The 10 k Ω potentiometers bias the 74AS04 in the linear region. This circuit could be used for external oscillator designs.

FIGURE 4-3: EXAMPLE OF EXTERNAL PARALLEL RESONANT CRYSTAL OSCILLATOR CIRCUIT (USING XT, HS OR LP OSCILLATOR MODE)

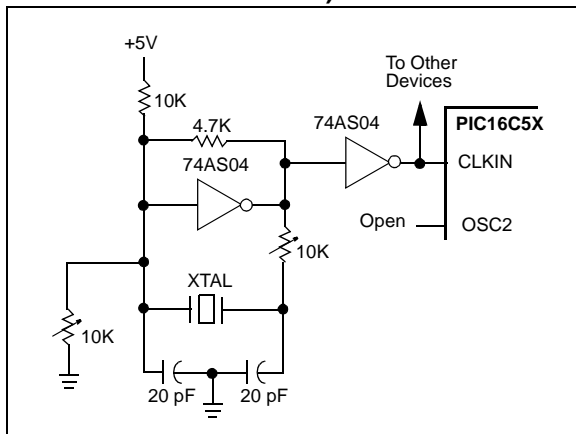
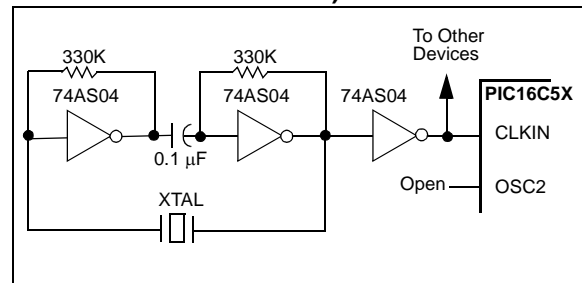


Figure 4-4 shows a series resonant oscillator circuit. This circuit is also designed to use the fundamental frequency of the crystal. The inverter performs a 180-degree phase shift in a series resonant oscillator circuit. The 330 k Ω resistors provide the negative feedback to bias the inverters in their linear region.

FIGURE 4-4: EXAMPLE OF EXTERNAL SERIES RESONANT CRYSTAL OSCILLATOR CIRCUIT (USING XT, HS OR LP OSCILLATOR MODE)



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

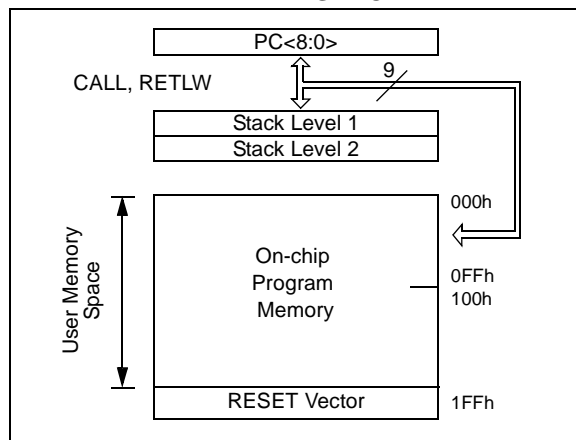


FIGURE 6-2: PIC16C56/CR56 PROGRAM MEMORY MAP AND STACK

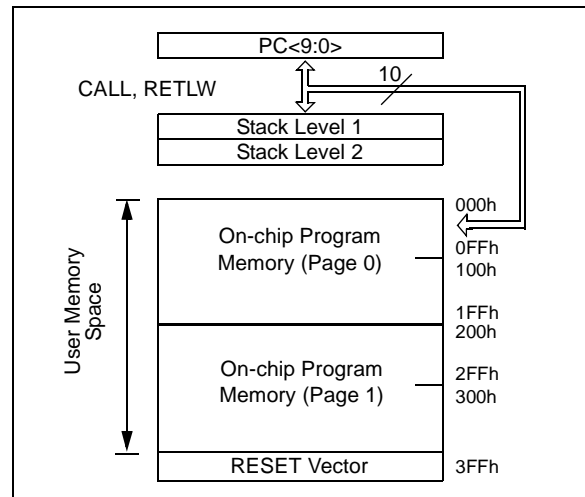
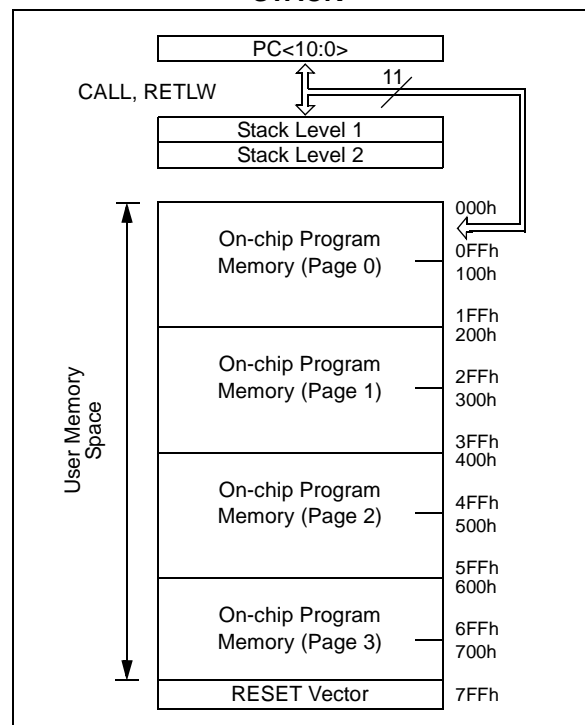


FIGURE 6-3: PIC16C57/CR57/C58/CR58 PROGRAM MEMORY MAP AND STACK



PIC16C5X

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

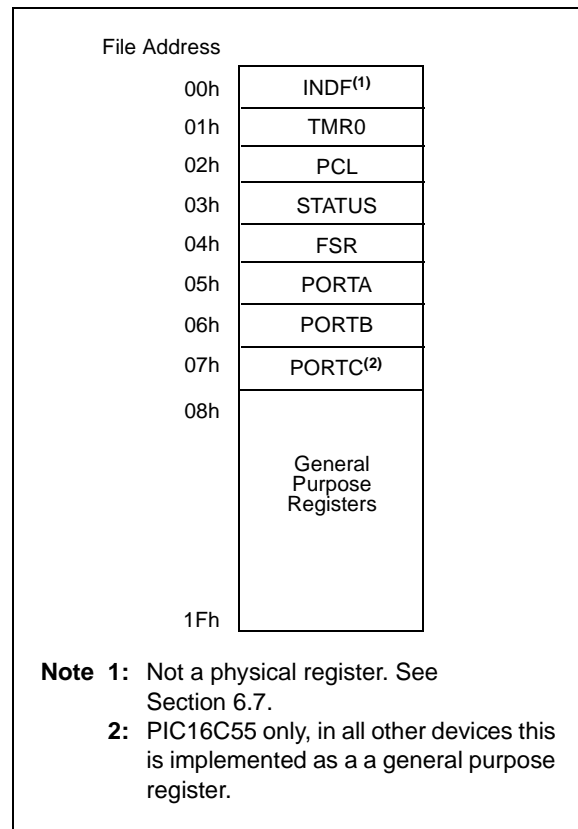
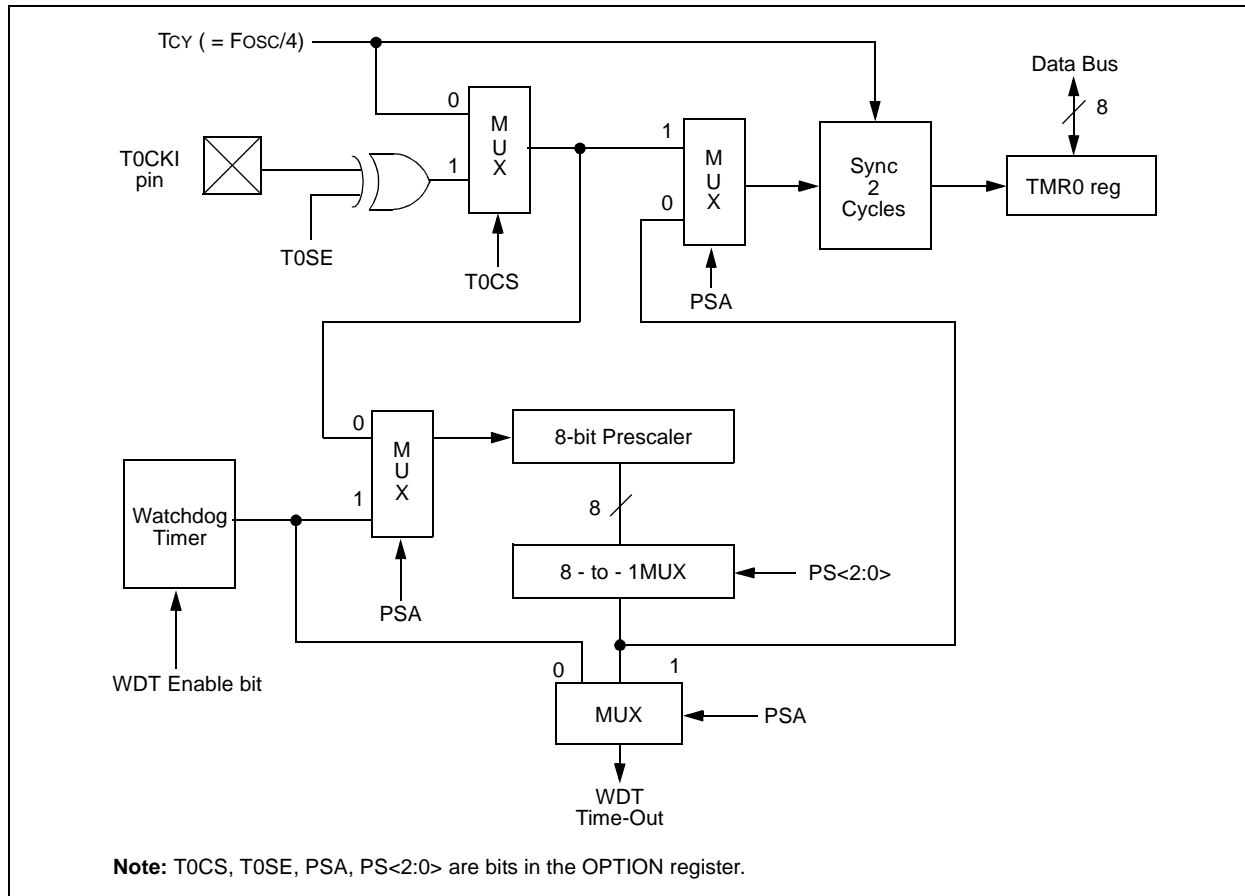


FIGURE 8-6: BLOCK DIAGRAM OF THE TIMER0/WDT PRESCALER



10.0 INSTRUCTION SET SUMMARY

Each PIC16C5X instruction is a 12-bit word divided into an OPCODE, which specifies the instruction type and one or more operands which further specify the operation of the instruction. The PIC16C5X instruction set summary in Table 10-2 groups the instructions into byte-oriented, bit-oriented, and literal and control operations. Table 10-1 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 is used to specify which one of the 32 file registers in that bank is to be used by the instruction.

The destination designator specifies where the result of the operation is to be placed. If 'd' is '0', the result is placed in the W register. If 'd' is '1', the result is placed in the file register specified in the instruction.

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

For **literal and control** operations, 'k' represents an 8 or 9-bit constant or literal value.

TABLE 10-1: OPCODE FIELD DESCRIPTIONS

| Field | Description |
|----------------|---|
| f | Register file address (0x00 to 0x1F) |
| 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 |
| label | Label name |
| TOS | Top of Stack |
| PC | Program Counter |
| WDT | Watchdog Timer Counter |
| TO | Time-out bit |
| PD | Power-down bit |
| dest | Destination, either the W register or the specified register file location |
| [] | Options |
| () | Contents |
| → | Assigned to |
| < > | Register bit field |
| ∈ | In the set of |
| <i>italics</i> | User defined term (font is courier) |

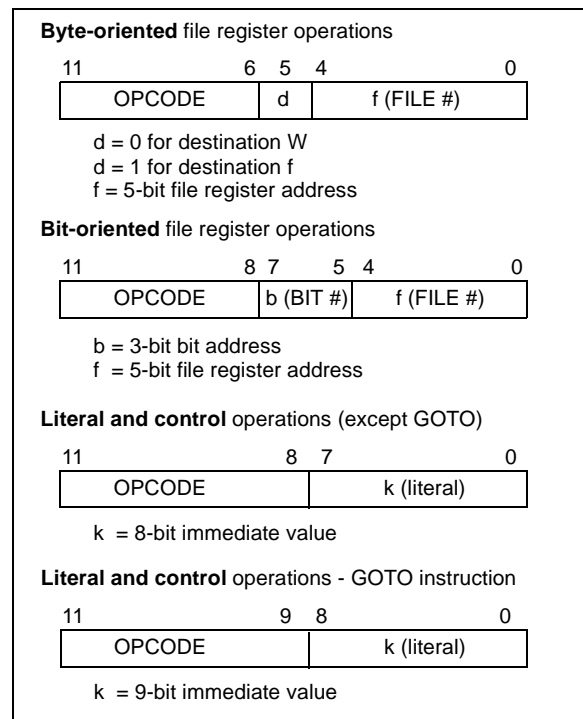
All instructions are executed within one single instruction cycle, unless a conditional test is true or the program counter is changed as a result of an instruction. In this case, the execution takes two instruction cycles. One instruction cycle consists of four oscillator periods. Thus, for an oscillator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

Figure 10-1 shows the three general formats that the instructions can have. All examples in the figure use the following format to represent a hexadecimal number:

0xhhh

where 'h' signifies a hexadecimal digit.

FIGURE 10-1: GENERAL FORMAT FOR INSTRUCTIONS



PIC16C5X

TABLE 10-2: INSTRUCTION SET SUMMARY

| Mnemonic, Operands | | Description | Cycles | 12-Bit Opcode | | | Status Affected | Notes |
|---------------------------------------|------|------------------------------|--------|---------------|------|------|--------------------|---------|
| | | | | MSb | LSb | | | |
| ADDWF | f, d | Add W and f | 1 | 0001 | 11df | ffff | C, DC, Z | 1, 2, 4 |
| ANDWF | f, d | AND W with f | 1 | 0001 | 01df | ffff | Z | 2, 4 |
| CLRF | f | Clear f | 1 | 0000 | 011f | ffff | Z | 4 |
| CLRW | — | Clear W | 1 | 0000 | 0100 | 0000 | Z | |
| COMF | f, d | Complement f | 1 | 0010 | 01df | ffff | Z | |
| DECF | f, d | Decrement f | 1 | 0000 | 11df | ffff | Z | 2, 4 |
| DECFSZ | f, d | Decrement f, Skip if 0 | 1 (2) | 0010 | 11df | ffff | None | 2, 4 |
| INCF | f, d | Increment f | 1 | 0010 | 10df | ffff | Z | 2, 4 |
| INCFSZ | f, d | Increment f, Skip if 0 | 1 (2) | 0011 | 11df | ffff | None | 2, 4 |
| IORWF | f, d | Inclusive OR W with f | 1 | 0001 | 00df | ffff | Z | 2, 4 |
| MOVF | f, d | Move f | 1 | 0010 | 00df | ffff | Z | 2, 4 |
| MOVWF | f | Move W to f | 1 | 0000 | 001f | ffff | None | 1, 4 |
| NOP | — | No Operation | 1 | 0000 | 0000 | 0000 | None | |
| RLF | f, d | Rotate left f through Carry | 1 | 0011 | 01df | ffff | C | 2, 4 |
| RRF | f, d | Rotate right f through Carry | 1 | 0011 | 00df | ffff | C | 2, 4 |
| SUBWF | f, d | Subtract W from f | 1 | 0000 | 10df | ffff | C, DC, Z | 1, 2, 4 |
| SWAPF | f, d | Swap f | 1 | 0011 | 10df | ffff | None | 2, 4 |
| XORWF | f, d | Exclusive OR W with f | 1 | 0001 | 10df | ffff | Z | 2, 4 |
| BIT-ORIENTED FILE REGISTER OPERATIONS | | | | | | | | |
| BCF | f, b | Bit Clear f | 1 | 0100 | bbbf | ffff | None | 2, 4 |
| BSF | f, b | Bit Set f | 1 | 0101 | bbbf | ffff | None | 2, 4 |
| BTFSC | f, b | Bit Test f, Skip if Clear | 1 (2) | 0110 | bbbf | ffff | None | |
| BTFSS | f, b | Bit Test f, Skip if Set | 1 (2) | 0111 | bbbf | ffff | None | |
| LITERAL AND CONTROL OPERATIONS | | | | | | | | |
| ANDLW | k | AND literal with W | 1 | 1110 | kkkk | kkkk | Z | 1 |
| CALL | k | Call subroutine | 2 | 1001 | kkkk | kkkk | None | |
| CLRWDT | k | Clear Watchdog Timer | 1 | 0000 | 0000 | 0100 | TO, PD | |
| GOTO | k | Unconditional branch | 2 | 101k | kkkk | kkkk | None | |
| IORLW | k | Inclusive OR Literal with W | 1 | 1101 | kkkk | kkkk | Z | |
| MOVLW | k | Move Literal to W | 1 | 1100 | kkkk | kkkk | None | |
| OPTION | k | Load OPTION register | 1 | 0000 | 0000 | 0010 | None | |
| RETLW | k | Return, place Literal in W | 2 | 1000 | kkkk | kkkk | None | |
| SLEEP | — | Go into standby mode | 1 | 0000 | 0000 | 0011 | TO, PD | |
| TRIS | f | Load TRIS register | 1 | 0000 | 0000 | 0fff | None | 3 |
| XORLW | k | Exclusive OR Literal to W | 1 | 1111 | kkkk | kkkk | Z | |

Note 1: The 9th bit of the program counter will be forced to a '0' by any instruction that writes to the PC except for GOTO (see Section 6.5 for more on program counter).

- 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'.
- The instruction `TRIS f`, where $f = 5, 6$ or 7 causes the contents of the W register to be written to the tristate latches of PORTA, B or C respectively. A '1' forces the pin to a hi-impedance state and disables the output buffers.
- If this instruction is executed on the TMR0 register (and, where applicable, $d = 1$), the prescaler will be cleared (if assigned to TMR0).

PIC16C5X

BSF Bit Set f

Syntax: [*label*] BSF f,b
 Operands: $0 \leq f \leq 31$
 $0 \leq b \leq 7$
 Operation: $1 \rightarrow (f)$
 Status Affected: None
 Encoding:

| | | |
|------|-------|------|
| 0101 | bbbbf | ffff |
|------|-------|------|

 Description: Bit 'b' in register 'f' is set.
 Words: 1
 Cycles: 1
 Example: BSF FLAG_REG, 7

Before Instruction
 FLAG_REG = 0x0A
 After Instruction
 FLAG_REG = 0x8A

BTFSC Bit Test f, Skip if Clear

Syntax: [*label*] BTFSC f,b
 Operands: $0 \leq f \leq 31$
 $0 \leq b \leq 7$
 Operation: skip if $(f) = 0$
 Status Affected: None
 Encoding:

| | | |
|------|-------|------|
| 0110 | bbbbf | ffff |
|------|-------|------|

 Description: If bit 'b' in register 'f' is 0 then the next instruction is skipped.
 If bit 'b' is 0 then the next instruction fetched during the current instruction execution is discarded, and a NOP is executed instead, making this a 2-cycle instruction.
 Words: 1
 Cycles: 1(2)
 Example: HERE BTFSC FLAG, 1
 FALSE GOTO PROCESS_CODE
 TRUE •
 •
 •

Before Instruction
 PC = address (HERE)
 After Instruction
 if FLAG<1> = 0,
 PC = address (TRUE);
 if FLAG<1> = 1,
 PC = address (FALSE)

BTFSS Bit Test f, Skip if Set

Syntax: [*label*] BTFSS f,b
 Operands: $0 \leq f \leq 31$
 $0 \leq b < 7$
 Operation: skip if $(f) = 1$
 Status Affected: None
 Encoding:

| | | |
|------|-------|------|
| 0111 | bbbbf | ffff |
|------|-------|------|

 Description: If bit 'b' in register 'f' is '1' then the next instruction is skipped.
 If bit 'b' is '1', then the next instruction fetched during the current instruction execution, is discarded and a NOP is executed instead, making this a 2-cycle instruction.
 Words: 1
 Cycles: 1(2)
 Example: HERE BTFSS FLAG, 1
 FALSE GOTO PROCESS_CODE
 TRUE •
 •
 •

Before Instruction
 PC = address (HERE)
 After Instruction
 If FLAG<1> = 0,
 PC = address (FALSE);
 if FLAG<1> = 1,
 PC = address (TRUE)

PIC16C5X

COMF Complement f

Syntax: [*label*] COMF f,d

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

Operation: $(f) \rightarrow (dest)$

Status Affected: Z

Encoding:

| | | |
|------|------|------|
| 0010 | 01df | ffff |
|------|------|------|

Description: The contents of register 'f' are complemented. 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: COMF REG1, 0

Before Instruction
 REG1 = 0x13
 After Instruction
 REG1 = 0x13
 W = 0xEC

DECFSZ Decrement f

Syntax: [*label*] DECFSZ f,d

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

Operation: $(f) - 1 \rightarrow (dest)$

Status Affected: Z

Encoding:

| | | |
|------|------|------|
| 0000 | 11df | ffff |
|------|------|------|

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'.

Words: 1

Cycles: 1

Example: DECFSZ CNT, 1

Before Instruction
 CNT = 0x01
 Z = 0
 After Instruction
 CNT = 0x00
 Z = 1

DECFSZ Decrement f, Skip if 0

Syntax: [*label*] DECFSZ f,d

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

Operation: $(f) - 1 \rightarrow d$; skip if result = 0

Status Affected: None

Encoding:

| | | |
|------|------|------|
| 0010 | 11df | ffff |
|------|------|------|

Description: The contents of register 'f' are decremented. If 'd' is 0 the result is placed in the W register. If 'd' is 1 the result is placed back in register 'f'. If the result is 0, the next instruction, 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 DECFSZ CNT, 1
 GOTO LOOP
 CONTINUE •
 •
 •

Before Instruction
 PC = address (HERE)
 After Instruction
 CNT = CNT - 1;
 if CNT = 0,
 PC = address (CONTINUE);
 if CNT \neq 0,
 PC = address (HERE+1)

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

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 \neq 0,
PC = address (HERE + 1)

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.

11.13 PICDEM 3 Low Cost PIC16CXXX Demonstration Board

The PICDEM 3 demonstration board is a simple demonstration board that supports the PIC16C923 and PIC16C924 in the PLCC package. It will also support future 44-pin PLCC microcontrollers with an LCD Module. 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 3 demonstration board on a PRO MATE II device programmer, or a PICSTART Plus development programmer with an adapter socket, and easily test firmware. The MPLAB ICE in-circuit emulator may also be used with the PICDEM 3 demonstration board to test firmware. A prototype area has been provided to the user for adding 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 thermistor and separate headers for connection to an external LCD module and a keypad. Also provided on the PICDEM 3 demonstration board is a LCD panel, with 4 commons and 12 segments, that is capable of displaying time, temperature and day of the week. The PICDEM 3 demonstration board provides an additional RS-232 interface and Windows software for showing the demultiplexed LCD signals on a PC. A simple serial interface allows the user to construct a hardware demultiplexer for the LCD signals.

11.14 PICDEM 17 Demonstration Board

The PICDEM 17 demonstration board is an evaluation board that demonstrates the capabilities of several Microchip microcontrollers, including PIC17C752, PIC17C756A, PIC17C762 and PIC17C766. All necessary hardware is included to run basic demo programs, which are supplied on a 3.5-inch disk. A programmed sample is included and the user may erase it and program it with the other sample programs using the PRO MATE II device programmer, or the PICSTART Plus development programmer, and easily debug and test the sample code. In addition, the PICDEM 17 demonstration board supports downloading of programs to and executing out of external FLASH memory on board. The PICDEM 17 demonstration board is also usable with the MPLAB ICE in-circuit emulator, or the PICMASTER emulator and all of the sample programs can be run and modified using either emulator. Additionally, a generous prototype area is available for user hardware.

11.15 KEELoQ Evaluation and Programming Tools

KEELOQ evaluation and programming tools support Microchip's HCS Secure Data Products. The HCS evaluation kit includes a LCD display to show changing codes, a decoder to decode transmissions and a programming interface to program test transmitters.

13.1 DC Characteristics: PIC16CR54A-04, 10, 20, PIC16LCR54A-04 (Commercial) PIC16CR54A-04I, 10I, 20I, PIC16LCR54A-04I (Industrial)

| PIC16LCR54A-04 PIC16LCR54A-04I (Commercial, Industrial) | | | Standard Operating Conditions (unless otherwise specified) Operating Temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial | | | | |
|---|--------|---|---|------|------|-------|--------------------------------------|
| PIC16CR54A-04, 10, 20 PIC16CR54A-04I, 10I, 20I (Commercial, Industrial) | | | Standard Operating Conditions (unless otherwise specified) Operating Temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial | | | | |
| Param No. | Symbol | Characteristic/Device | Min | Typ† | Max | Units | Conditions |
| | IPD | Power-down Current⁽²⁾ | | | | | |
| D006 | | PIC16LCR54A-Commercial | — | 1.0 | 6.0 | μA | V _{DD} = 2.5V, WDT disabled |
| | | | — | 2.0 | 8.0* | μA | V _{DD} = 4.0V, WDT disabled |
| | | | — | 3.0 | 15 | μA | V _{DD} = 6.0V, WDT disabled |
| | | | — | 5.0 | 25 | μA | V _{DD} = 6.0V, WDT enabled |
| D006A | | PIC16CR54A-Commercial | — | 1.0 | 6.0 | μA | V _{DD} = 2.5V, WDT disabled |
| | | | — | 2.0 | 8.0* | μA | V _{DD} = 4.0V, WDT disabled |
| | | | — | 3.0 | 15 | μA | V _{DD} = 6.0V, WDT disabled |
| | | | — | 5.0 | 25 | μA | V _{DD} = 6.0V, WDT enabled |
| D007 | | PIC16LCR54A-Industrial | — | 1.0 | 8.0 | μA | V _{DD} = 2.5V, WDT disabled |
| | | | — | 2.0 | 10* | μA | V _{DD} = 4.0V, WDT disabled |
| | | | — | 3.0 | 20* | μA | V _{DD} = 4.0V, WDT enabled |
| | | | — | 3.0 | 18 | μA | V _{DD} = 6.0V, WDT disabled |
| | | | — | 5.0 | 45 | μA | V _{DD} = 6.0V, WDT enabled |
| D007A | | PIC16CR54A-Industrial | — | 1.0 | 8.0 | μA | V _{DD} = 2.5V, WDT disabled |
| | | | — | 2.0 | 10* | μA | V _{DD} = 4.0V, WDT disabled |
| | | | — | 3.0 | 20* | μA | V _{DD} = 4.0V, WDT enabled |
| | | | — | 3.0 | 18 | μA | V _{DD} = 6.0V, WDT disabled |
| | | | — | 5.0 | 45 | μA | V _{DD} = 6.0V, WDT enabled |

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

* These parameters are characterized but not tested.

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

Note 1: This is the limit to which V_{DD} can be lowered in SLEEP mode without losing RAM data.

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

a) The test conditions for all I_{DD} measurements in active Operation mode are: OSC1 = external square wave, from rail-to-rail; all I/O pins tristated, pulled to V_{SS}, T_{0CKI} = V_{DD}, MCLR = V_{DD}; WDT enabled/disabled as specified.

b) For standby current measurements, the conditions are the same, except that the device is in SLEEP mode. The power-down current in SLEEP mode does not depend on the oscillator type.

3: Does not include current through R_{EXT}. The current through the resistor can be estimated by the formula: I_R = V_{DD}/2R_{EXT} (mA) with R_{EXT} in kΩ.

15.1 DC Characteristics: PIC16C54A-04, 10, 20 (Commercial) PIC16C54A-04I, 10I, 20I (Industrial) PIC16LC54A-04 (Commercial) PIC16LC54A-04I (Industrial)

| PIC16LC54A-04 PIC16LC54A-04I (Commercial, Industrial) | | | Standard Operating Conditions (unless otherwise specified) Operating Temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial | | | | |
|---|--------|-----------------------|---|------|-----|---------------|---|
| PIC16C54A-04, 10, 20 PIC16C54A-04I, 10I, 20I (Commercial, Industrial) | | | Standard Operating Conditions (unless otherwise specified) Operating Temperature $0^{\circ}\text{C} \leq T_A \leq +70^{\circ}\text{C}$ for commercial $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ for industrial | | | | |
| Param No. | Symbol | Characteristic/Device | Min | Typ† | Max | Units | Conditions |
| D006 | IPD | PIC16LC5X | Power-down Current⁽²⁾ | | | | |
| | | | — | 2.5 | 12 | μA | $V_{DD} = 2.5\text{V}$, WDT enabled, Commercial |
| | | | — | 0.25 | 4.0 | μA | $V_{DD} = 2.5\text{V}$, WDT disabled, Commercial |
| | | | — | 2.5 | 14 | μA | $V_{DD} = 2.5\text{V}$, WDT enabled, Industrial |
| D006A | | PIC16C5X | — | 0.25 | 5.0 | μA | $V_{DD} = 2.5\text{V}$, WDT disabled, Industrial |
| | | | — | 4.0 | 12 | μA | $V_{DD} = 3.0\text{V}$, WDT enabled, Commercial |
| | | | — | 0.25 | 4.0 | μA | $V_{DD} = 3.0\text{V}$, WDT disabled, Commercial |
| | | | — | 5.0 | 14 | μA | $V_{DD} = 3.0\text{V}$, WDT enabled, Industrial |
| | | | — | 0.3 | 5.0 | μA | $V_{DD} = 3.0\text{V}$, WDT disabled, Industrial |

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

* These parameters are characterized but not tested.

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

Note 1: This is the limit to which V_{DD} can be lowered in SLEEP mode without losing RAM data.

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

a) The test conditions for all I_{DD} measurements in active Operation mode are: $OSC1$ = external square wave, from rail-to-rail; all I/O pins tristated, pulled to V_{SS} , $T0CKI = V_{DD}$, $MCLR = V_{DD}$; WDT enabled/disabled as specified.

b) For standby current measurements, the conditions are the same, except that the device is in SLEEP mode. The power-down current in SLEEP mode does not depend on the oscillator type.

3: Does not include current through R_{EXT} . The current through the resistor can be estimated by the formula: $I_R = V_{DD}/2R_{EXT}$ (mA) with R_{EXT} in $k\Omega$.

PIC16C5X

15.2 DC Characteristics: PIC16C54A-04E, 10E, 20E (Extended) PIC16LC54A-04E (Extended)

| PIC16LC54A-04E (Extended) | | Standard Operating Conditions (unless otherwise specified) Operating Temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended | | | | | |
|---------------------------------------|--------|---|------------|--------|--------------|--------|--|
| PIC16C54A-04E, 10E, 20E (Extended) | | Standard Operating Conditions (unless otherwise specified) Operating Temperature $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ for extended | | | | | |
| Param No. | Symbol | Characteristic | Min | Typ† | Max | Units | Conditions |
| D001 | VDD | Supply Voltage | | | | | |
| | | PIC16LC54A | 3.0 2.5 | — — | 6.25 6.25 | V V | XT and RC modes LP mode |
| D001A | | PIC16C54A | 3.5 4.5 | — — | 5.5 5.5 | V V | RC and XT modes HS mode |
| D002 | VDR | RAM Data Retention Voltage ⁽¹⁾ | — | 1.5* | — | V | Device in SLEEP mode |
| D003 | VPOR | VDD Start Voltage to ensure Power-on Reset | — | VSS | — | V | See Section 5.1 for details on Power-on Reset |
| D004 | SVDD | VDD Rise Rate to ensure Power-on Reset | 0.05* | — | — | V/ms | See Section 5.1 for details on Power-on Reset |
| D010 | IDD | Supply Current ⁽²⁾ | | | | | |
| | | PIC16LC54A | — | 0.5 | 25 | mA | FOSC = 4.0 MHz, VDD = 5.5V, RC ⁽³⁾ and XT modes |
| | | | — | 11 | 27 | μA | FOSC = 32 kHz, VDD = 2.5V, LP mode, Commercial |
| | | | — | 11 | 35 | μA | FOSC = 32 kHz, VDD = 2.5V, LP mode, Industrial |
| | | | — | 11 | 37 | μA | FOSC = 32 kHz, VDD = 2.5V, LP mode, Extended |
| D010A | | PIC16C54A | — | 1.8 | 3.3 | mA | FOSC = 4.0 MHz, VDD = 5.5V, RC ⁽³⁾ and XT modes |
| | | | — | 4.8 | 10 | mA | FOSC = 10 MHz, VDD = 5.5V, HS mode |
| | | | — | 9.0 | 20 | mA | FOSC = 20 MHz, VDD = 5.5V, HS mode |

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

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

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

a) The test conditions for all IDD measurements in active Operation mode are: OSC1 = external square wave, from rail-to-rail; all I/O pins tristated, pulled to VSS, T0CKI = VDD, MCLR = VDD; WDT enabled/disabled as specified.

b) For standby current measurements, the conditions are the same, except that the device is in SLEEP mode. The power-down current in SLEEP mode does not depend on the oscillator type.

Note 3: Does not include current through REXT. The current through the resistor can be estimated by the formula: $I_R = V_{DD}/2R_{EXT}$ (mA) with REXT in kΩ.

PIC16C5X

FIGURE 16-12: TYPICAL I_{DD} vs. FREQUENCY (WDT DISABLED, RC MODE @ 100 pF, 25°C)

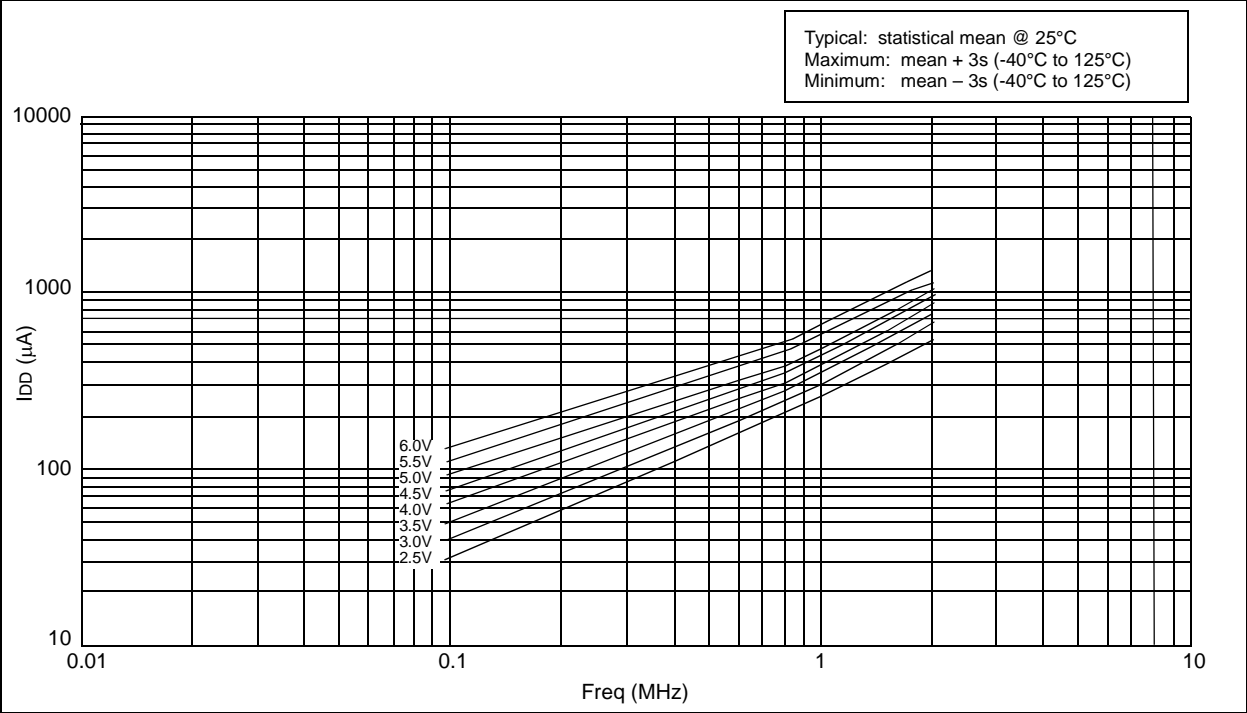


FIGURE 16-13: MAXIMUM I_{DD} vs. FREQUENCY (WDT DISABLED, RC MODE @ 100 pF, -40°C to +85°C)

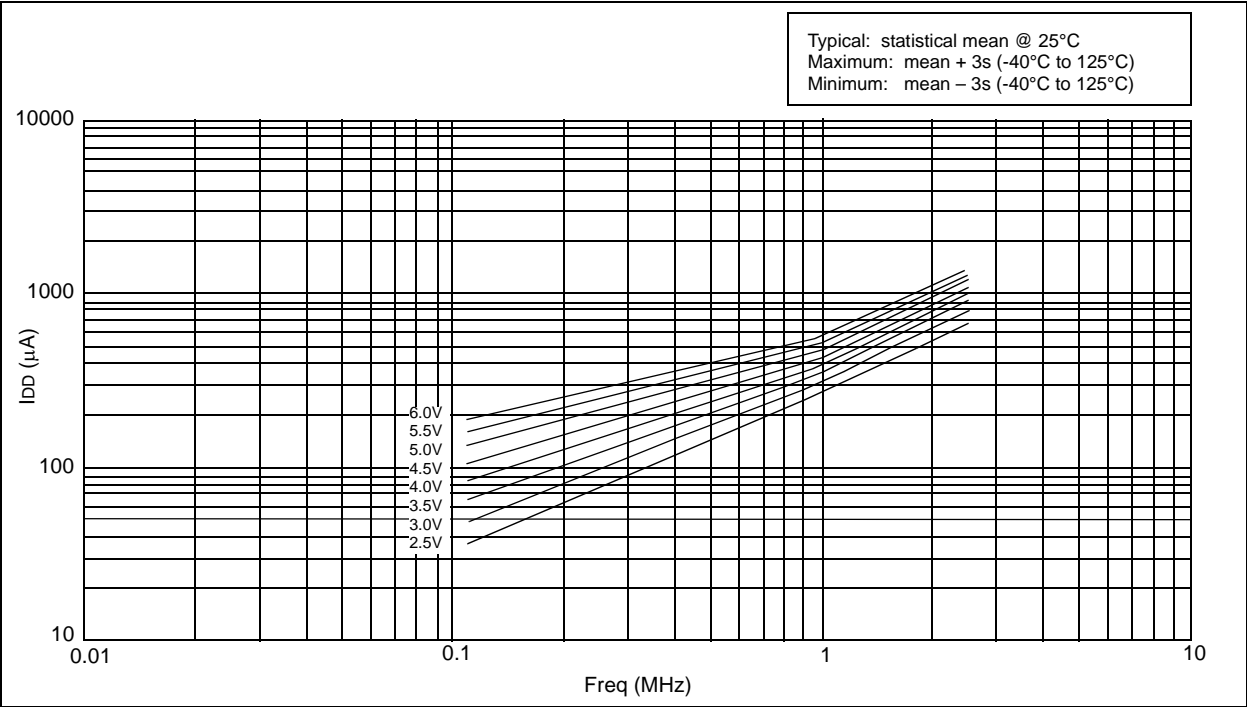


FIGURE 18-14: WDT TIMER TIME-OUT PERIOD vs. VDD⁽¹⁾

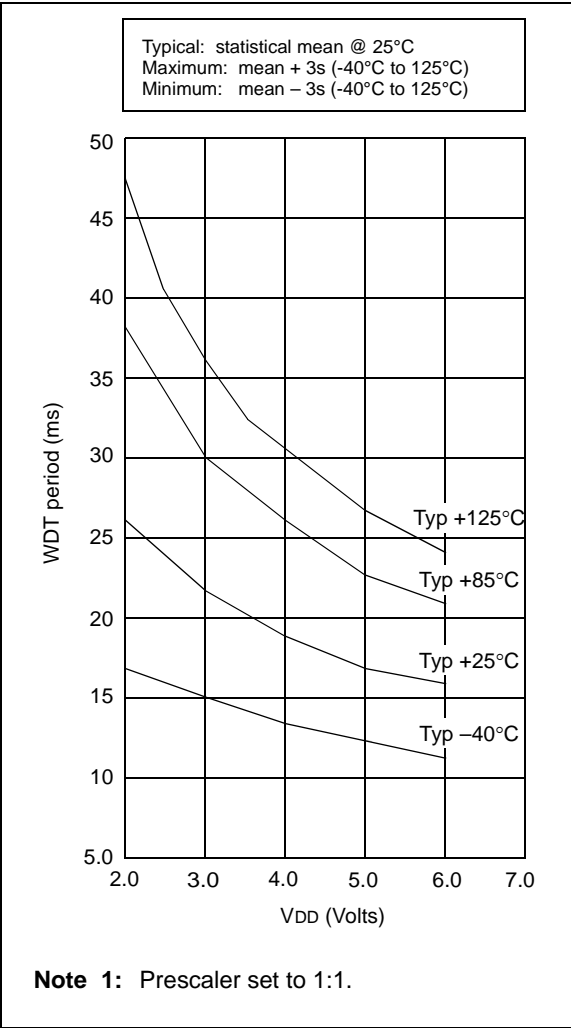
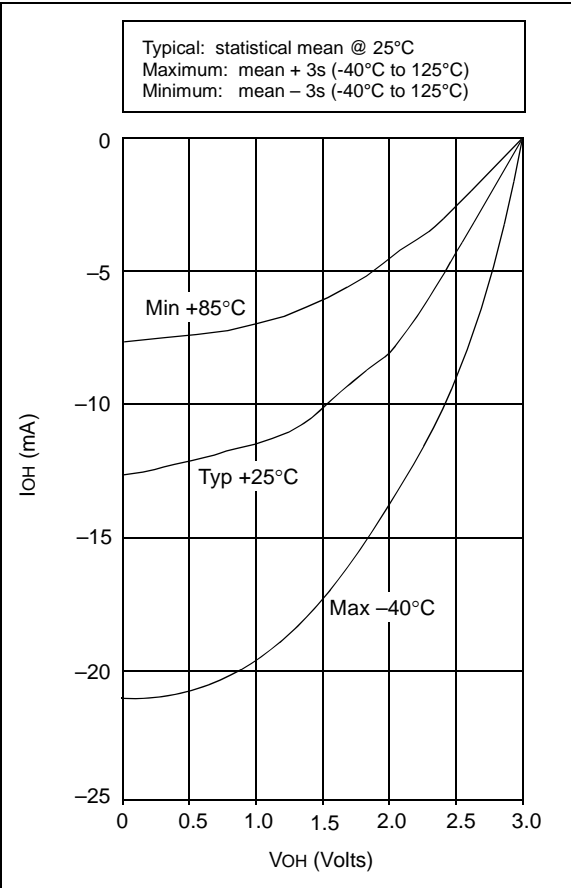


FIGURE 18-15: PORTA, B AND C I_{OH} vs. V_{OH}, VDD = 3 V



19.0 ELECTRICAL CHARACTERISTICS - PIC16LC54C 40MHz

Absolute Maximum Ratings^(†)

| | |
|--|-----------------------|
| Ambient temperature under bias | –55°C to +125°C |
| Storage temperature | –65°C to +150°C |
| Voltage on VDD with respect to VSS | 0 to +7.5V |
| Voltage on $\overline{\text{MCLR}}$ with respect to VSS..... | 0 to +14V |
| Voltage on all other pins with respect to VSS | –0.6V to (VDD + 0.6V) |
| Total power dissipation ⁽¹⁾ | 800 mW |
| Max. current out of Vss pin | 150 mA |
| Max. current into VDD pin | 100 mA |
| Max. current into an input pin (T0CKI only) | ±500 μ A |
| Input clamp current, I _{IK} (V _I < 0 or V _I > VDD)..... | ±20 mA |
| Output clamp current, I _{OK} (V _O < 0 or V _O > VDD) | ±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 A, B or C) | 50 mA |
| Max. output current sunk by a single I/O (Port A, B or C)..... | 50 mA |

Note 1: 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.

19.1 DC Characteristics: PIC16C54C/C55A/C56A/C57C/C58B-40 (Commercial)⁽¹⁾

| PIC16C54C/C55A/C56A/C57C/C58B-40 (Commercial) | | | Standard Operating Conditions (unless otherwise specified) | | | | |
|--|--------|---|--|------------|------------|----------|---|
| | | | Operating Temperature 0°C ≤ TA ≤ +70°C for commercial | | | | |
| Param No. | Symbol | Characteristic | Min | Typ† | Max | Units | Conditions |
| D001 | VDD | Supply Voltage | 4.5 | — | 5.5 | V | HS mode from 20 - 40 MHz |
| D002 | VDR | RAM Data Retention Voltage ⁽²⁾ | — | 1.5* | — | V | Device in SLEEP mode |
| D003 | VPOR | VDD Start Voltage to ensure Power-on Reset | — | VSS | — | V | See Section 5.1 for details on Power-on Reset |
| D004 | SVDD | VDD Rise Rate to ensure Power-on Reset | 0.05* | — | — | V/ms | See Section 5.1 for details on Power-on Reset |
| D010 | IDD | Supply Current ⁽³⁾ | — | 5.2 6.8 | 12.3 16 | mA mA | FOSC = 40 MHz, VDD = 4.5V, HS mode FOSC = 40 MHz, VDD = 5.5V, HS mode |
| D020 | IPD | Power-down Current ⁽³⁾ | — | 1.8 9.8 | 7.0 27* | μA μA | VDD = 5.5V, WDT disabled, Commercial VDD = 5.5V, WDT enabled, Commercial |

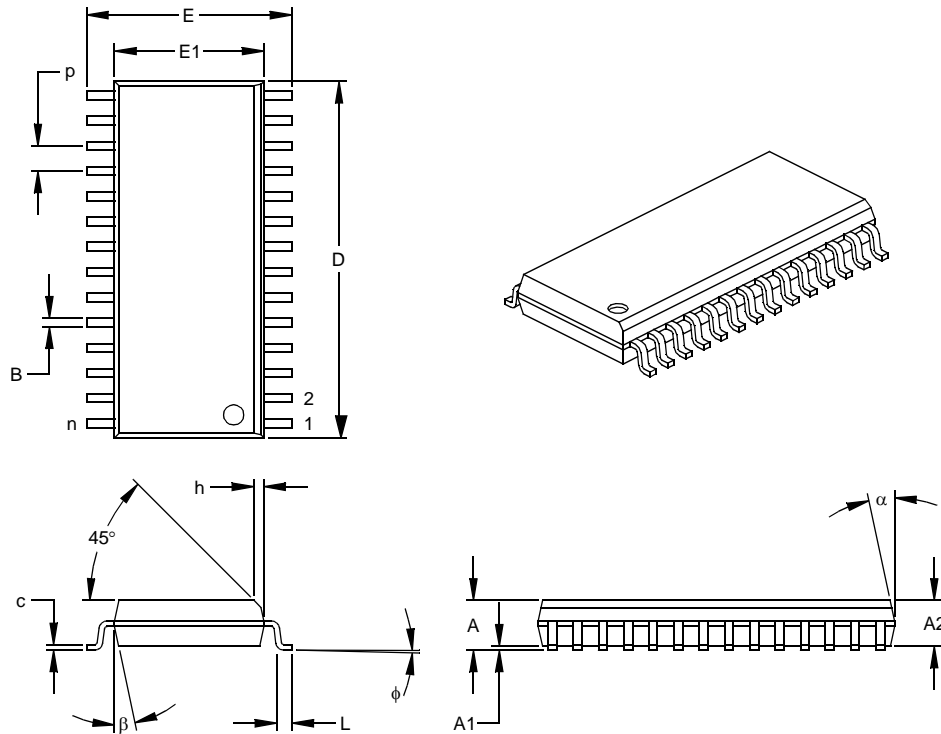
* These parameters are characterized but not tested.

† 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:** Device operation between 20 MHz to 40 MHz requires the following: VDD between 4.5V to 5.5V, OSC1 pin externally driven, OSC2 pin not connected, HS oscillator mode and commercial temperatures. For operation between DC and 20 MHz, See Section 19.1.
- 2:** This is the limit to which VDD can be lowered in SLEEP mode without losing RAM data.
- 3:** The supply current is mainly a function of the operating voltage and frequency. Other factors such as bus loading, oscillator type, bus rate, internal code execution pattern and temperature also have an impact on the current consumption.
- The test conditions for all IDD measurements in active Operation mode are: OSC1 = external square wave, from rail-to-rail; all I/O pins tristated, pulled to VSS, T0CKI = VDD, MCLR = VDD; WDT enabled/disabled as specified.
 - For standby current measurements, the conditions are the same, except that the device is in SLEEP mode. The power-down current in SLEEP mode does not depend on the oscillator type.

28-Lead Plastic Small Outline (SO) – Wide, 300 mil (SOIC)

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



| Units | | INCHES* | | | MILLIMETERS | | |
|--------------------------|----|---------|------|------|-------------|-------|-------|
| Dimension Limits | | MIN | NOM | MAX | MIN | NOM | MAX |
| Number of Pins | n | | 28 | | | 28 | |
| 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 | .288 | .295 | .299 | 7.32 | 7.49 | 7.59 |
| Overall Length | D | .695 | .704 | .712 | 17.65 | 17.87 | 18.08 |
| 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 Top | φ | 0 | 4 | 8 | 0 | 4 | 8 |
| Lead Thickness | c | .009 | .011 | .013 | 0.23 | 0.28 | 0.33 |
| 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 |

* 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.

JEDEC Equivalent: MS-013

Drawing No. C04-052