

## Memory Programming Specification

This document includes the programming specifications for the following devices:

- PIC16F505

### 1.0 PROGRAMMING THE PIC16F505

The PIC16F505 is programmed using a serial method. The Serial mode will allow the PIC16F505 to be programmed while in the user's system. This allows for increased design flexibility. This programming specification applies to PIC16F505 devices in all packages.

### 1.1 Hardware Requirements

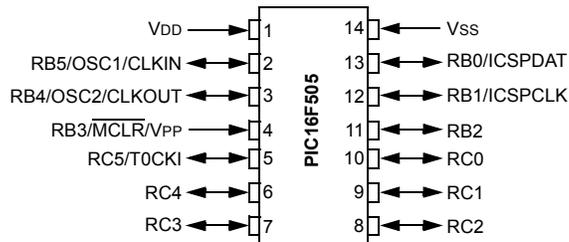
The PIC16F505 requires one power supply for VDD and one for VPP.

### 1.2 Program/Verify Mode

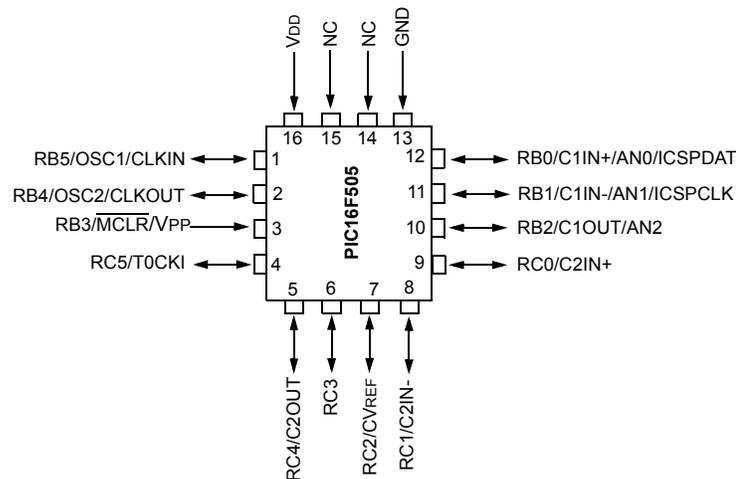
The Program/Verify mode for the PIC16F505 allows programming of user program memory, user ID locations, backup OSCCAL location and the Configuration Word.

### Pin Diagrams

#### PDIP, SOIC, TSSOP



#### QFN



# PIC16F505

---

**TABLE 1-1: PIN DESCRIPTIONS (DURING PROGRAMMING): PIC16F505**

| Pin Name | During Programming  |                  |   |
|----------|---------------------|------------------|---|
|          | Function            | Pin Type         | Pin Description                           |
| RB1      | ICSPCLK             | I                | Clock input – Schmitt Trigger input       |
| RB0      | ICSPDAT             | I/O              | Data input/output – Schmitt Trigger input |
| MCLR/VPP | Program/Verify mode | P <sup>(1)</sup> | Program Mode Select                       |
| VDD      | VDD                 | P                | Power Supply                              |
| VSS      | VSS                 | P                | Ground                                    |

**Legend:** I = Input, O = Output, P = Power

**Note 1:** In the PIC16F505, the programming high voltage is internally generated. To activate the Program/Verify mode, V<sub>IHH</sub> with current capability of I<sub>PP</sub> (see Table 6-1) needs to be applied to the MCLR input.

## 2.0 MEMORY MAPPING

### 2.1 User Program Memory Map

The user memory space extends from (0x000-0x3FF) on the PIC16F505. In Program/Verify mode, the program memory space extends from (0x000-0x7FF). The first half, (0x000-0x3FF), is user program memory. The second half, (0x400-0x7FF), is configuration memory. The PC will increment from (0x000 to 0x3FF), then to 0x400 (not to 0x000).

In the configuration memory space, 0x400-0x43F is physically implemented. However, only locations 0x400 through 0x403 are available. Other locations are reserved.

### 2.2 User ID Locations

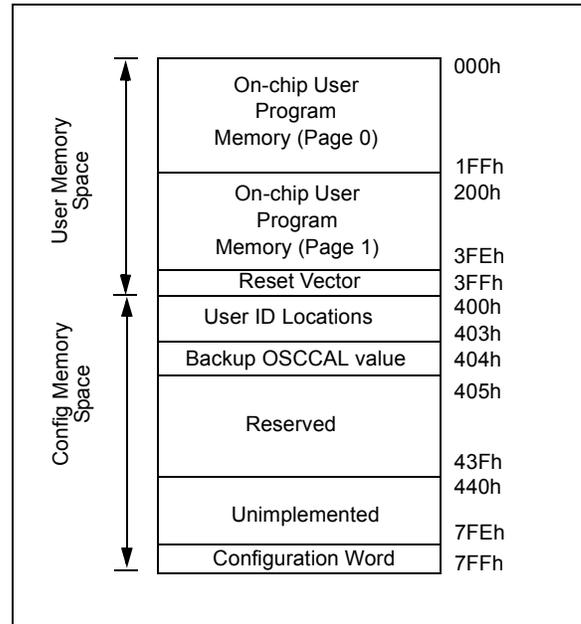
A user may store identification information (ID) in four user ID locations. The user ID locations are mapped in [0x400:0x403]. It is recommended that the user use only the four Least Significant bits (LSb) of each user ID. The user ID locations read out normally, even after code protection is enabled. The 12 bits may be programmed, but only the four LSbs are displayed by MPLAB® IDE. The xxxx's are "don't care" bits and are not read by MPLAB IDE. It is recommended that user ID locations are written as 'xxxx xxxx bbbb' where 'bbbb' is user ID information.

### 2.3 Configuration Word

The Configuration Word is physically located at 0x7FF. It is only available upon Program mode entry. Once an Increment Address command is issued, the Configuration Word is no longer accessible, regardless of the address of the program counter.

**Note:** By convention the Configuration Word is stored at the logical address location of 0xFFFF within the hex file generated for the PIC16F505. This logical address location may not reflect the actual physical address for the part itself. It is the responsibility of the programming software to retrieve the Configuration Word from the logical address within the hex file and translate the address to the proper physical location when programming.

**FIGURE 2-1: PIC16F505 PROGRAM MEMORY MAP**



### 2.4 Oscillator Calibration Bits

The oscillator Calibration bits are stored at the Reset vector as the operand of a MOVLW instruction. Programming interfaces must allow users to program the Calibration bits themselves for custom trimming of the INTOSC. Capability for programming the Calibration bits when programming the entire memory array must also be maintained for backwards compatibility.

### 2.5 Backup OSCCAL Value

The backup OSCCAL value at address 0x404 is a factory reserved location where the OSCCAL value is stored during testing of the INTOSC. This location is not erased during a standard Bulk Erase, but is erased if the PC is moved into configuration memory prior to invoking a Bulk Erase.

If this value is erased, it is the user's responsibility to rewrite it back to this location for future use.

**Note:** Default OSCCAL management given in Figure 3-9 always copies the back-up OSCCAL value to the program memory location.

# PIC16F505

## 3.0 COMMANDS AND ALGORITHMS

### 3.1 Program/Verify Mode

The Program/Verify mode is entered by holding pins ICSPCLK and ICSPDAT low while raising VDD pin from VIL to VDD. Then raise VPP from VIL to VIH. Once in this mode, the user program memory and configuration memory can be accessed and programmed in serial fashion. Clock and data are Schmitt Trigger input in this mode (see Figure 3-1).

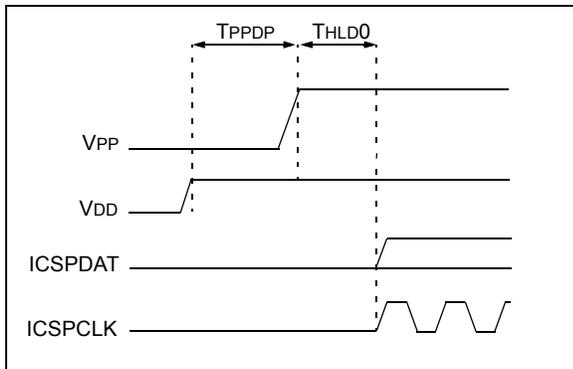
The sequence that enters the device into the Programming/Verify mode places all other logic into the Reset state (the MCLR pin was initially at VIL). This means that all I/O are in the Reset state (high-impedance inputs).

#### 3.1.1 PROGRAMMING

The programming sequence loads a word, programs, verifies and finally increments the PC.

Program/Verify mode entry will set the address to 0x7FF. The Increment Address command will increment the PC. The available commands are shown in Table 3-1.

**FIGURE 3-1: ENTERING HIGH VOLTAGE PROGRAM/VERIFY MODE**



**TABLE 3-1: COMMAND MAPPING FOR PIC16F505**

| Command                       | Mapping (MSb ... LSb) |   |   |   |   |   | Data             |
|-------------------------------|-----------------------|---|---|---|---|---|------------------|
| Load Data for Program Memory  | x                     | x | 0 | 0 | 1 | 0 | 0, data (14), 0  |
| Read Data from Program Memory | x                     | x | 0 | 1 | 0 | 0 | 0, data (14), 0  |
| Increment Address             | x                     | x | 0 | 1 | 1 | 0 |                  |
| Begin Programming             | x                     | x | 1 | 0 | 0 | 0 | Externally Timed |
| End Programming               | x                     | x | 1 | 1 | 1 | 0 |                  |
| Bulk Erase Program Memory     | x                     | x | 1 | 0 | 0 | 1 | Internally Timed |

#### 3.1.2 SERIAL PROGRAM/VERIFY OPERATION

The ICSPCLK pin is used for clock input and the ICSPDAT pin is used for data input/output during serial operation. To input a command, the clock pin is cycled six times. Each command bit is latched on the falling edge of the clock with the LSb of the command being input first. The data must adhere to the setup (TSET1) and hold (THLD1) times with respect to the falling edge of the clock (see Table 6-1).

Commands that do not have data associated with them are required to wait a minimum of TDLY2 measured from the falling edge of the last command clock to the rising edge of the next command clock (see Table 6-1). Commands that do have data associated with them (Read and Load) are also required to wait TDLY2 between the command and the data segment measured from the falling edge of the last command clock to the rising edge of the first data clock. The data segment, consisting of 16 clock cycles, can begin after this delay.

**Note:** After every End Programming command, a delay of TDIS must be delayed.

The first and last clock pulses during the data segment correspond to the Start and Stop bits, respectively. Input data is a "don't care" during the Start and Stop cycles. The 14 clock pulses between the Start and Stop cycles, clock the 14 bits of input/output data. Data is transferred LSb first.

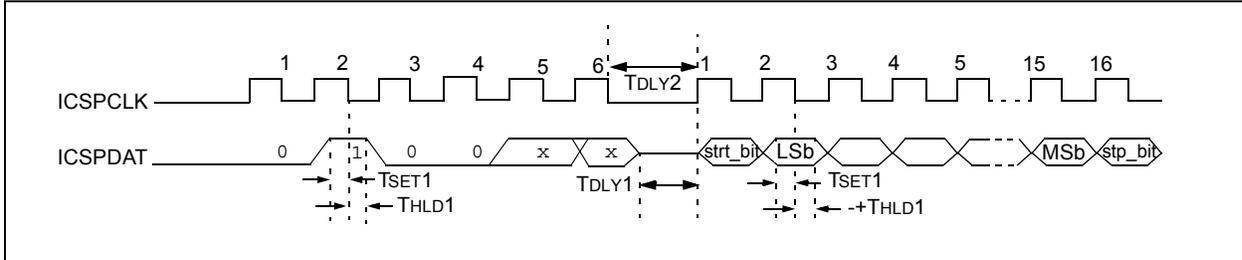
During Read commands, in which the data is output from the PIC16F505, the ICSPDAT pin transitions from the high-impedance input state to the low-impedance output state at the rising edge of the second data clock (first clock edge after the Start cycle). The ICSPDAT pin returns to the high-impedance state at the rising edge of the 16th data clock (first edge of the Stop cycle). See Figure 3-3.

The commands that are available are described in Table 3-1.

### 3.1.2.1 Load Data For Program Memory

After receiving this command, the chip will load in a 14-bit “data word” when 16 cycles are applied, as described previously. Because this is a 12-bit core, the two MSBs of the data word are ignored. A timing diagram for the Load Data command is shown in Figure 3-2.

**FIGURE 3-2: LOAD DATA COMMAND (PROGRAM/VERIFY)**

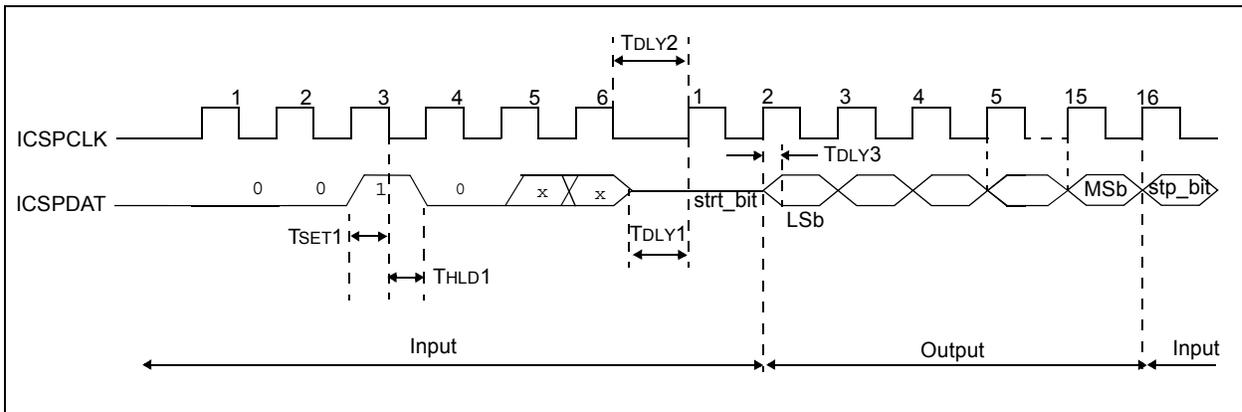


### 3.1.2.2 Read Data From Program Memory

After receiving this command, the chip will transmit data bits out of the program memory (user or configuration) currently addressed, starting with the second rising edge of the clock input. The data pin will go into Output mode on the second rising clock edge, and it will revert to Input mode (high-impedance) after the 16th rising edge. Because this is a 12-bit core, the two MSBs of the 14-bit word will be read as ‘0’s.

If the program memory is code-protected ( $\overline{CP} = 0$ ), portions of the program memory will be read as zeros. See **Section 5.0 “Code Protection”** for details.

**FIGURE 3-3: READ DATA FROM PROGRAM MEMORY COMMAND**



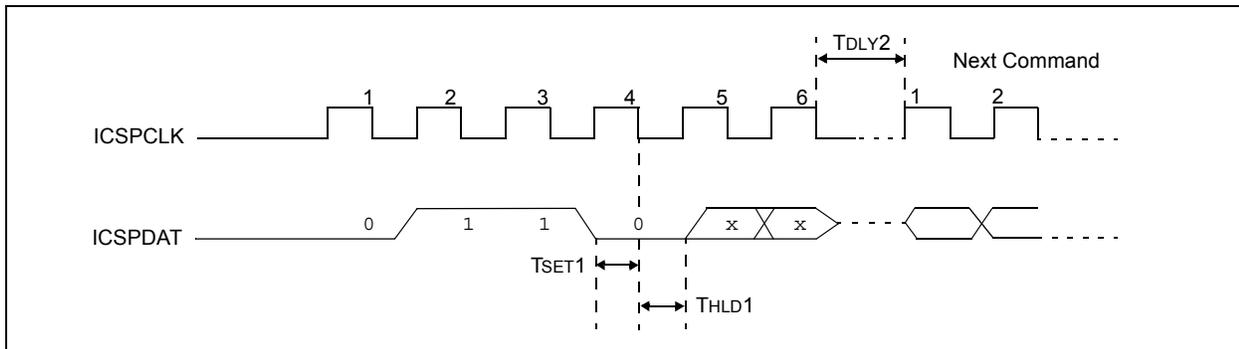
# PIC16F505

## 3.1.2.3 Increment Address

The PC is incremented when this command is received. A timing diagram of this command is shown in Figure 3-4.

It is not possible to decrement the address counter. To reset this counter, the user must either exit and re-enter Program/Verify mode or increment the PC from 0x7FF to 0x000.

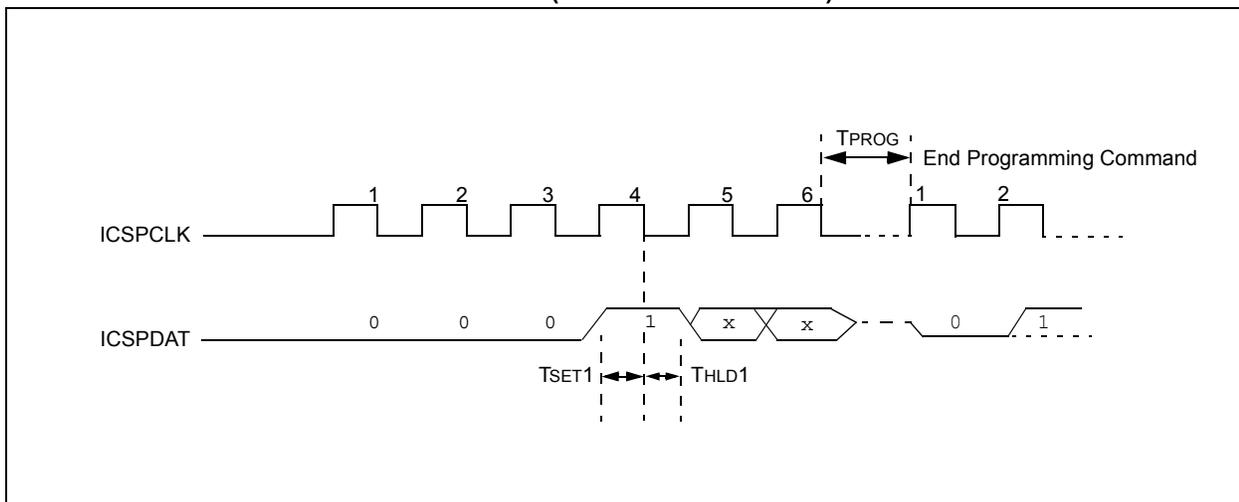
**FIGURE 3-4: INCREMENT ADDRESS COMMAND**



## 3.1.2.4 Begin Programming (Externally Timed)

A Load command must be given before every Begin Programming command. Programming will begin after this command is received and decoded. Programming requires (TPROG) time and is terminated using an End Programming command. This command programs the current location, no erase is performed.

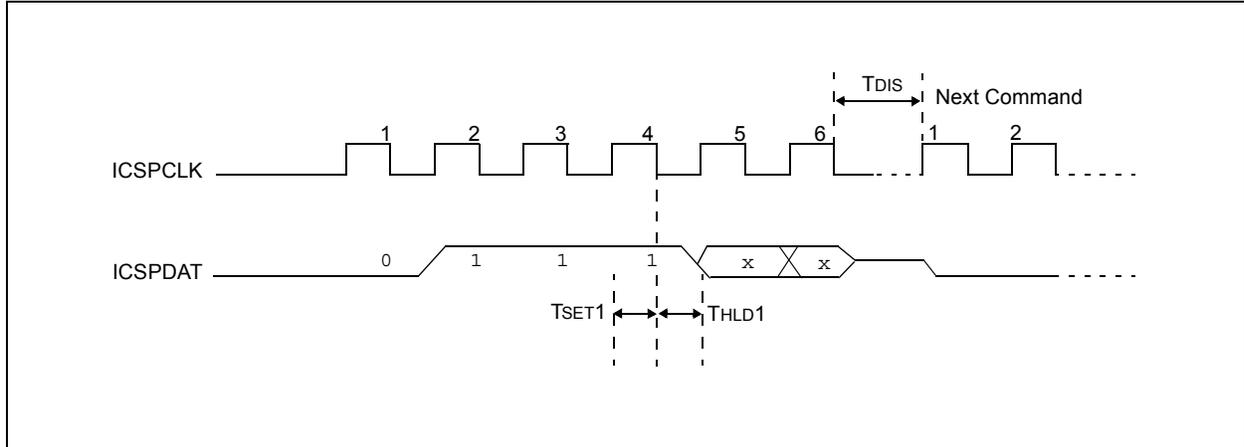
**FIGURE 3-5: BEGIN PROGRAMMING (EXTERNALLY TIMED)**



### 3.1.2.5 End Programming

The End Programming command terminates the program process. A delay of TDIS (see Table 6-1) is required before the next command to allow the internal programming voltage to discharge (see Figure 3-6).

**FIGURE 3-6: END PROGRAMMING (EXTERNALLY TIMED)**



### 3.1.2.6 Bulk Erase Program Memory

After this command is performed, the entire program memory and Configuration Word is erased.

- Note 1:** A fully erased part will read '1's in every program memory location.
- 2:** The oscillator Calibration bits are erased if a Bulk Erase is invoked. They must be read and saved prior to erasing the device and restored during the programming operation. Oscillator Calibration bits are stored at the Reset vector as the operand of a MOV<sub>LW</sub> instruction.

To perform a Bulk Erase of the program memory and configuration fuses, the following sequence must be performed (see Figure 3-12).

1. Read and save 0x3FF oscillator Calibration bits and 0x404 backup OSCCAL bits into computer/ programmer temporary memory.
2. Enter Program/Verify mode. PC is set to Configuration Word address.
3. Perform a Bulk Erase Program Memory command.
4. Wait TERA to complete Bulk Erase.
5. Restore OSCCAL bits.

To perform a full device Bulk Erase of the program memory, configuration fuses, user IDs and backup OSCCAL, the following sequence must be performed (see Figure 3-13).

1. Read and save 0x3FF oscillator Calibration bits and 0x404 backup OSCCAL bits into computer/ programmer temporary memory.
2. Enter Program/Verify mode.
3. Increment PC to 0x400 (first user ID location).
4. Perform a Bulk Erase command.
5. Wait TERA to complete Bulk Erase.
6. Restore OSCCAL bits.
7. Restore backup OSCCAL bits.

# PIC16F505

**TABLE 3-2: BULK ERASE RESULTS**

| PC =                                       | Program Memory Space |              | Configuration Memory Space |         |              |
|--|----------------------|--------------|----------------------------|---------|--------------|
|  | Program Memory       | Reset Vector | Configuration Word         | User ID | Backup OSCAL |
| Configuration Word or Program Memory Space | E                    | E            | E                          | U       | U            |
| First User ID Location                     | E                    | E            | E                          | E       | E            |

**Legend:** E = Erased, U = Unaffected

**FIGURE 3-7: BULK ERASE PROGRAM MEMORY COMMAND**

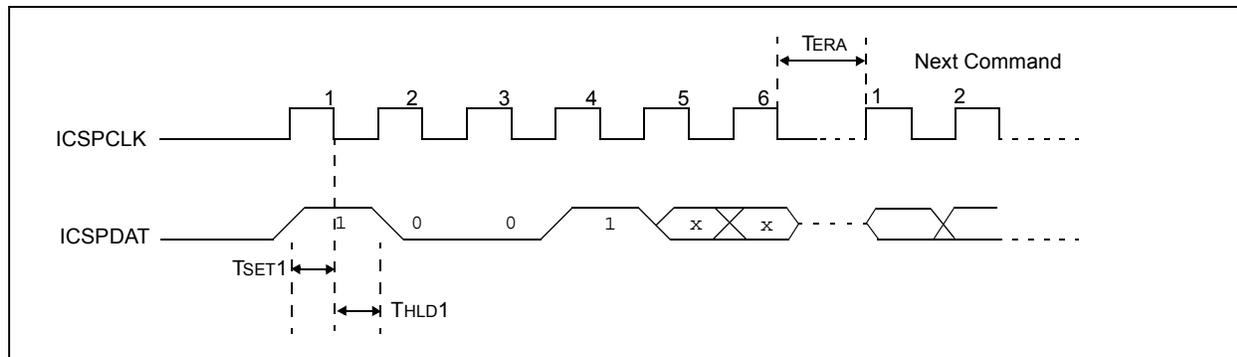
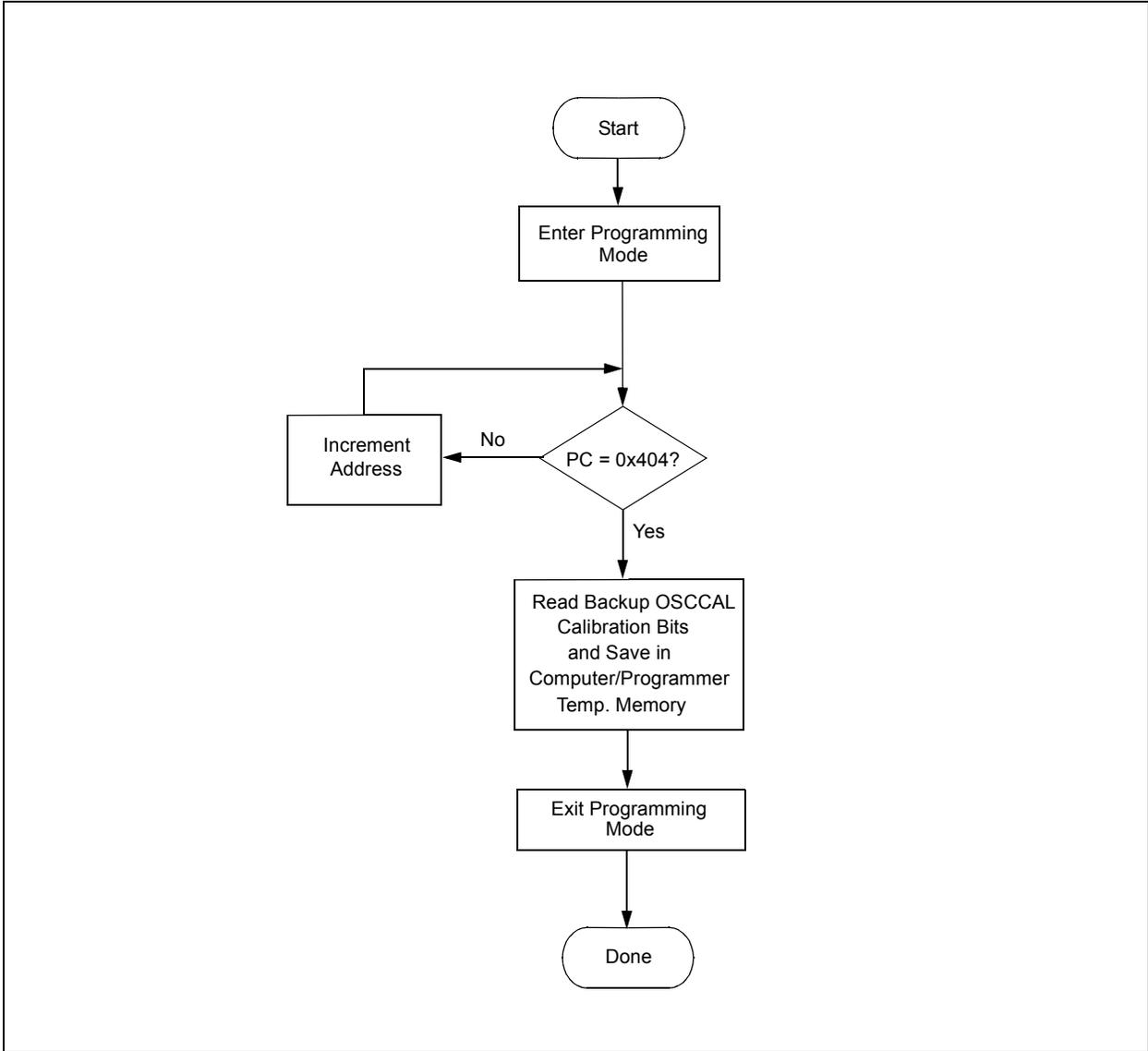
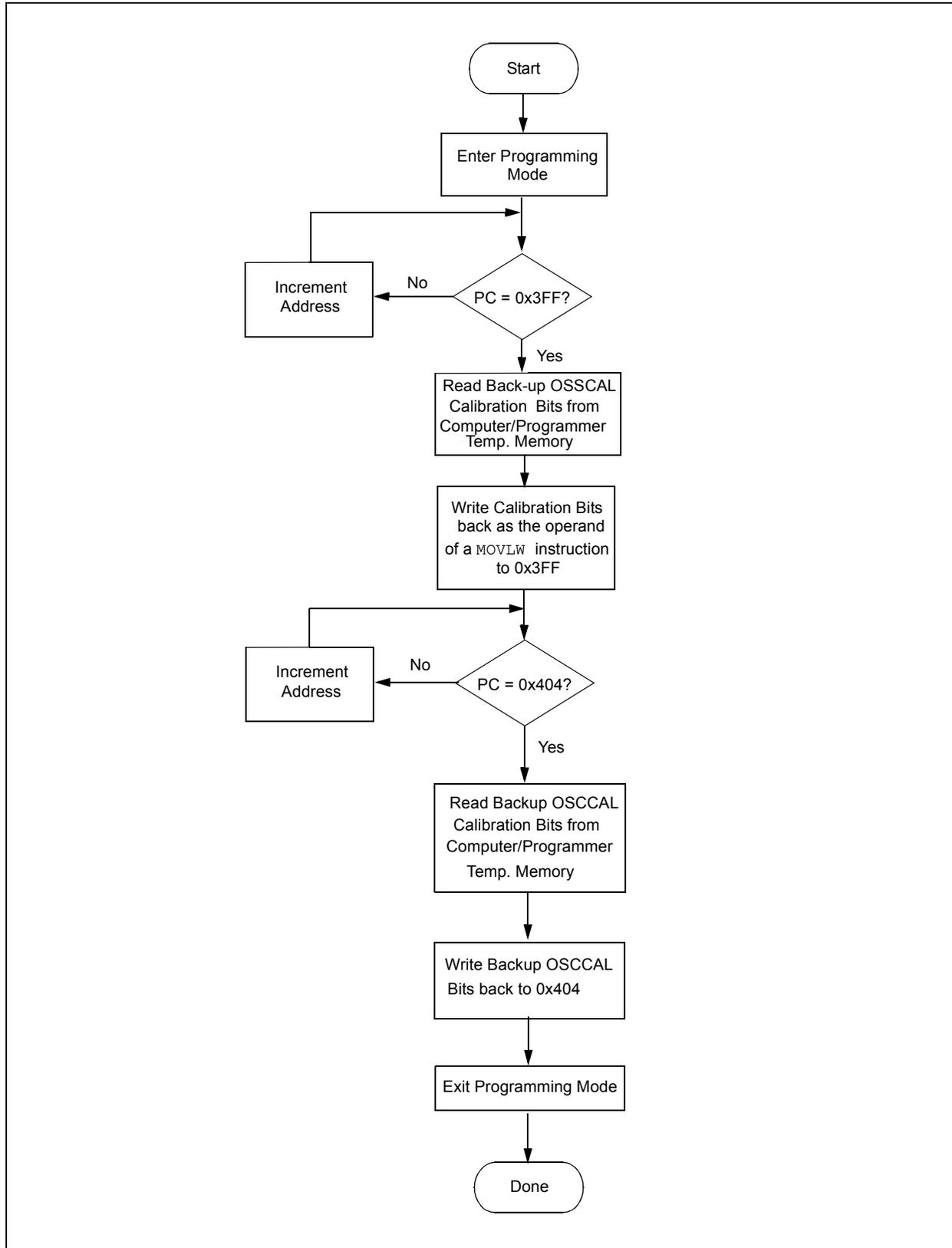


FIGURE 3-8: READING AND TEMPORARY SAVING OF THE OSCCAL CALIBRATION BITS

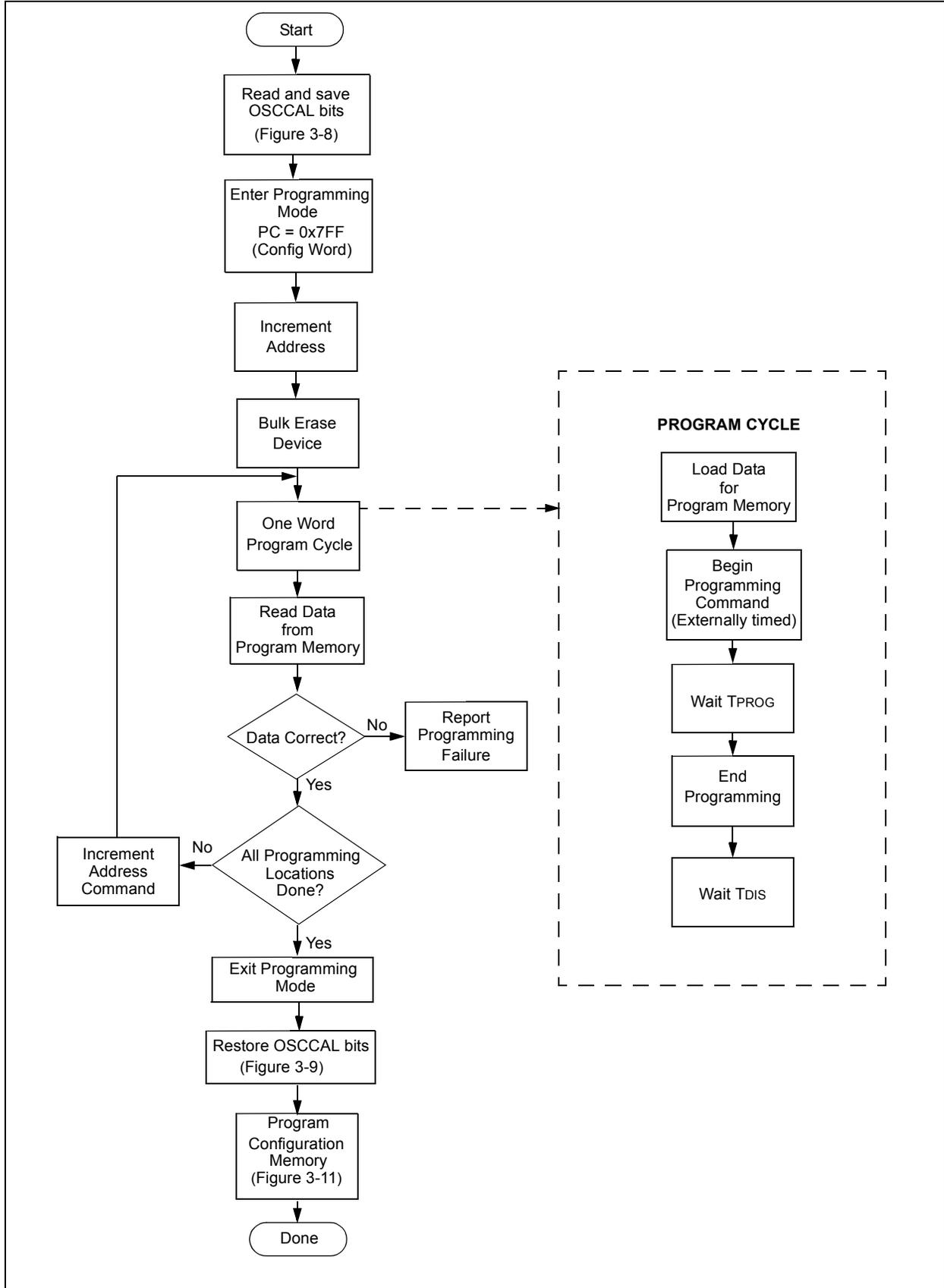


# PIC16F505

FIGURE 3-9: RESTORING/PROGRAMMING THE OSSCAL CALIBRATION BITS

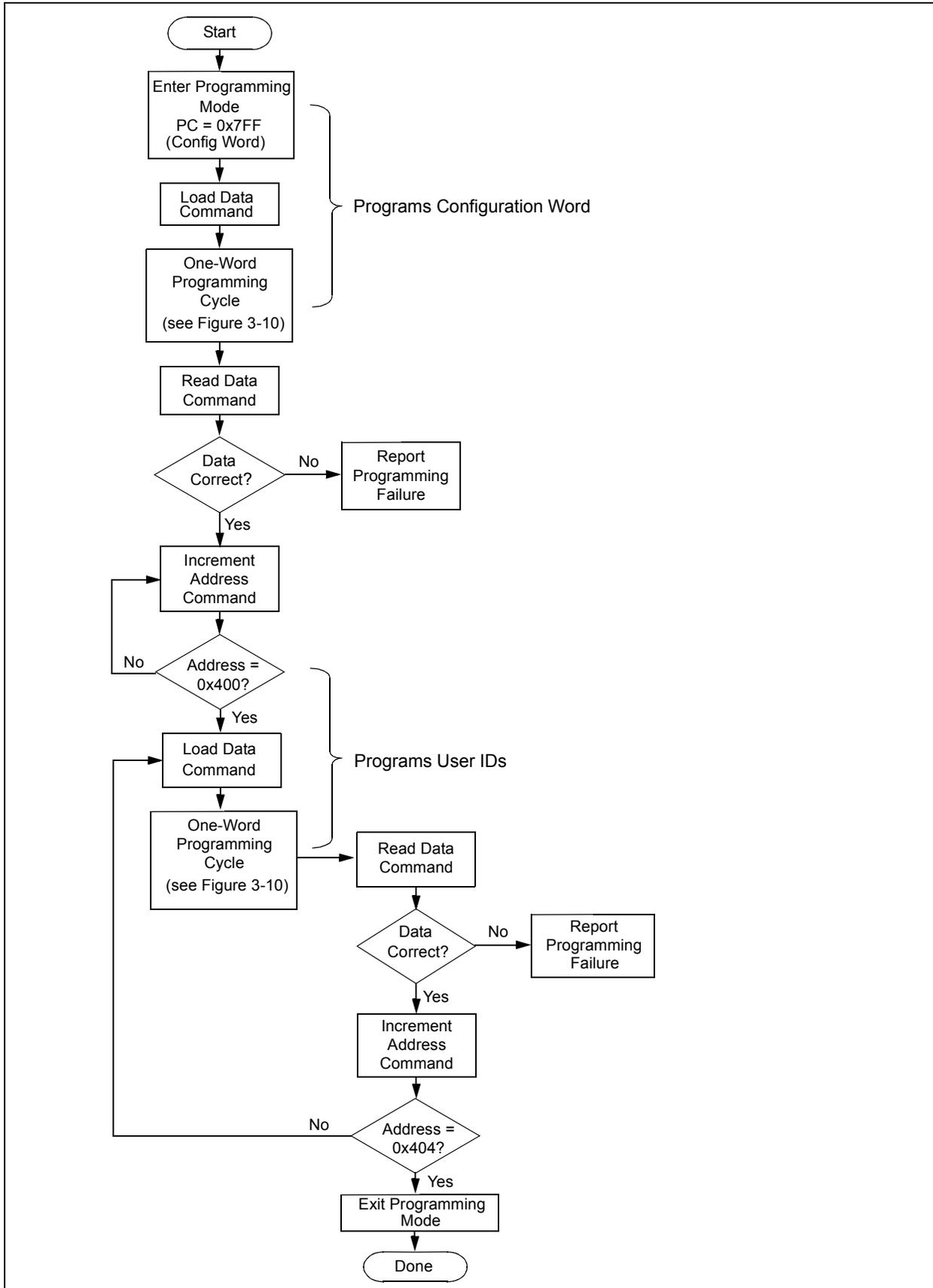


**FIGURE 3-10: PROGRAM FLOWCHART – PIC16F505 PROGRAM MEMORY**

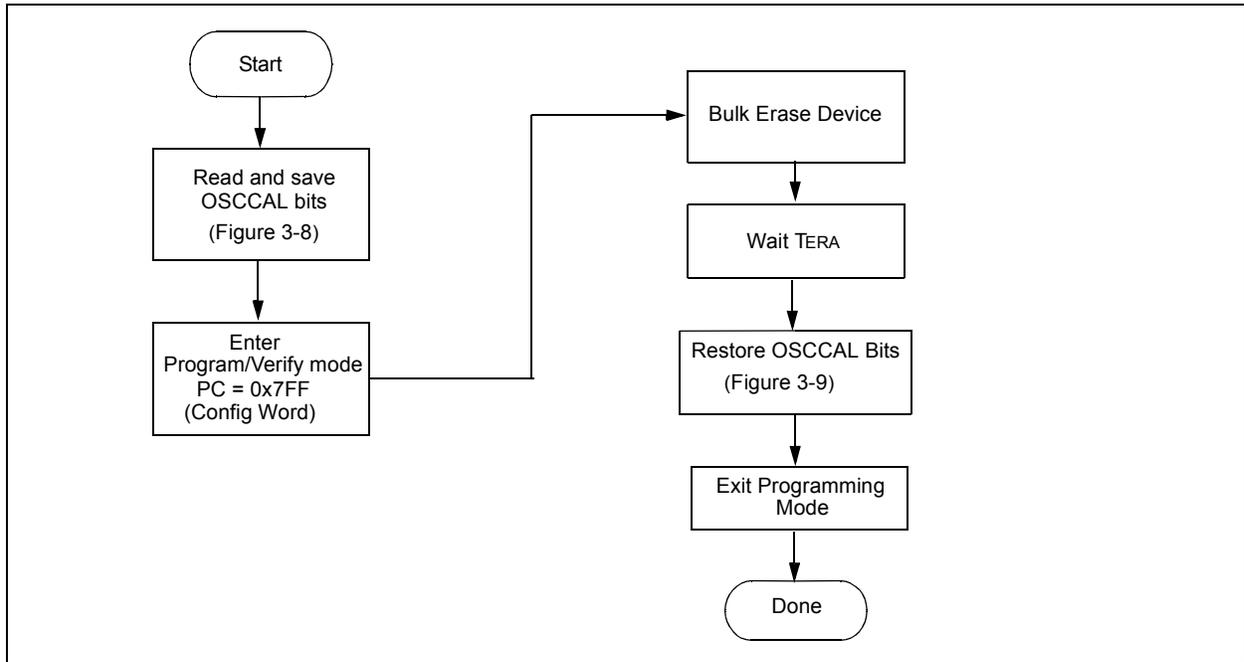


# PIC16F505

FIGURE 3-11: PROGRAM FLOWCHART – PIC16F505 CONFIGURATION MEMORY

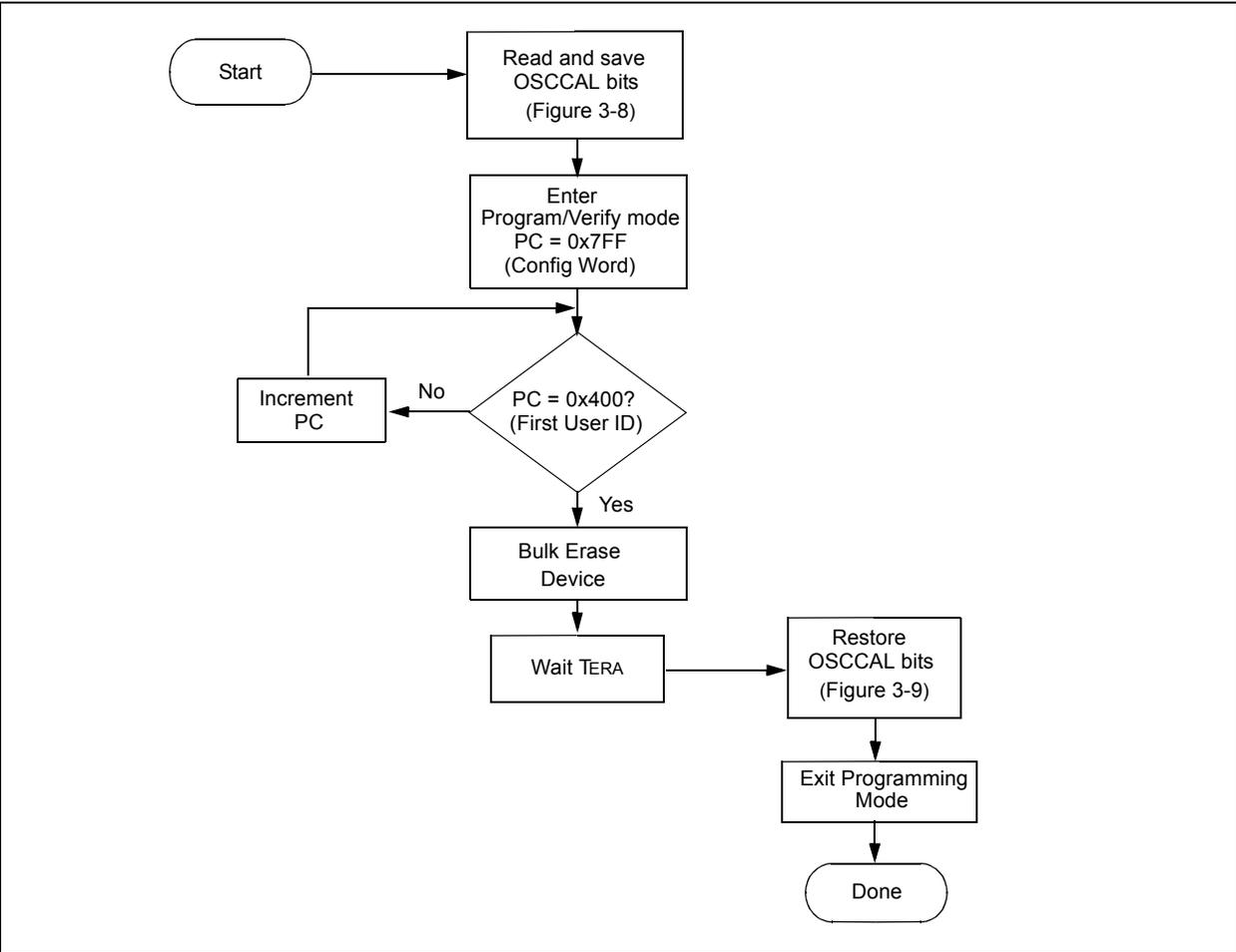


**FIGURE 3-12: PROGRAM FLOWCHART – ERASE PROGRAM MEMORY, CONFIGURATION WORD**



# PIC16F505

FIGURE 3-13: PROGRAM FLOWCHART – ERASE PROGRAM MEMORY, CONFIGURATION WORD AND USER ID



## 4.0 CONFIGURATION WORD

The PIC16F505 has several Configuration bits. These bits can be programmed (reads '0') or left unchanged (reads '1'), to select various device configurations.

### REGISTER 4-1: CONFIGURATION WORD – PIC16F505

|        |   |   |   |   |   |       |                        |      |       |       |       |
|--------|---|---|---|---|---|-------|------------------------|------|-------|-------|-------|
|        |   |   |   |   |   |       |                        |      |       |       |       |
| —      | — | — | — | — | — | MCLRE | $\overline{\text{CP}}$ | WDTE | FOSC2 | FOSC1 | FOSC0 |
| bit 11 |   |   |   |   |   | bit 0 |                        |      |       |       |       |

#### Legend:

|                   |                  |                                    |
|-------------------|------------------|------------------------------------|
| R = Readable bit  | W = Writable bit | U = Unimplemented bit, read as '0' |
| -n = Value at POR | '1' = Bit is set | '0' = Bit is cleared               |
|                   |                  | x = Bit is unknown                 |

bit 11-6 **Unimplemented:** Read as '1'

bit 5 **MCLRE:** Master Clear Enable bit

1 = RB3/ $\overline{\text{MCLR}}$  pin functions as  $\overline{\text{MCLR}}$

0 = RB3/ $\overline{\text{MCLR}}$  pin functions as RB3,  $\overline{\text{MCLR}}$  internally tied to VDD

bit 4  **$\overline{\text{CP}}$ :** Code Protection bit

1 = Code protection off

0 = Code protection on

bit 3 **WDTE:** Watchdog Timer Enable bit

1 = WDT enabled

0 = WDT disabled

bit 2-0 **FOSC2:FOSC0:** Oscillator Selection bits

000 = LP oscillator

001 = XT oscillator

010 = HS oscillator

011 = EC oscillator with RB4 function on RB4/OSC2/CLKOUT

100 = INTOSC with RB4 function on RB4/OSC2/CLKOUT

101 = INTOSC with CLKOUT function on RB4/OSC2/CLKOUT

110 = EXTRC with RB4 function on RB4/OSC2/CLKOUT

111 = EXTRC with CLKOUT function on RB4/OSC2/CLKOUT

# PIC16F505

## 5.0 CODE PROTECTION

For the PIC16F505, once code protection is enabled, program memory locations 0x040-0x3FE inclusive, read all '0's. Program memory locations 0x000-0x03F and 0x3FF are always unprotected. The user ID locations, backup OSCCAL location and the Configuration Word read out in an unprotected fashion. It is possible to program the user ID locations, backup OSCCAL location and the Configuration Word after code-protect is enabled.

### 5.1 Disabling Code Protection

It is recommended that the following procedure be performed before any other programming is attempted. It is also possible to turn code protection off ( $\overline{CP} = 1$ ) using this procedure. However, ***all data within the program memory will be erased when this procedure is executed, and thus, the security of the code is not compromised.***

To disable code-protect:

- a) Enter Program mode.
- b) Execute Bulk Erase Program Memory command (001001).
- c) Wait TERA.

### 5.2 Embedding Configuration Word and User ID Information in the Hex File

**Note:** To allow portability of code, the programmer is required to read the Configuration Word and user ID locations from the hex file when loading the hex file. If Configuration Word information was not present in the hex file, then a simple warning message may be issued. Similarly, while saving a hex file, Configuration Word and user ID information must be included. An option to not include this information may be provided. Microchip Technology Incorporated feels strongly that this feature is important for the benefit of the end customer.

## 5.3 Checksum Computation

### 5.3.1 CHECKSUM

Checksum is calculated by reading the contents of the PIC16F505 memory locations and adding up the opcodes up to the maximum user addressable location (e.g., 0x3FF for the PIC16F505). Any carry bits exceeding 16 bits are neglected. Finally, the Configuration Word (appropriately masked) is added to the checksum. Checksum computation for the PIC16F505 is shown in Table 5-1.

The checksum is calculated by summing the following:

- The contents of all program memory locations
- The Configuration Word, appropriately masked
- Masked ID locations (when applicable)

The Least Significant 16 bits of this sum is the checksum.

The following table describes how to calculate the checksum for each device.

**Note:** The checksum calculation differs depending on the code-protect setting. The Configuration Word and ID locations can always be read regardless of the code-protect settings.

**TABLE 5-1: CHECKSUM COMPUTATIONS – PIC16F505<sup>(1)</sup>**

| Device    | Code-Protect | Checksum*                              | Blank Value | 0x723 at 0 and Max Address |
|-----------|--------------|--|-------------|----------------------------|
| PIC16F505 | OFF          | SUM[0x000:0x3FE] + CFGW & 0x03F        | 0xEC40      | 0xDA88                     |
|           | ON           | SUM[0x00:0x3F] + CFGW & 0x03F + SUM_ID | 0xEC2F      | 0xD19B                     |

**Legend:** CFGW = Configuration Word

SUM[a:b] = [Sum of locations a to b inclusive]

SUM\_ID = User ID locations masked by 0xF then made into a 16-bit value with ID0 as the Most Significant nibble.

For example, ID0 = 0x1, ID1 = 0x2, ID2 = 0x3, ID3 = 0x4, then SUM\_ID = 0x1234.

\*Checksum = [Sum of all the individual expressions] MODULO [0xFFFF]

+ = Addition

& = Bitwise AND

**Note 1:** Checksum shown assumes that SUM\_ID contains the unprotected checksum.

# PIC16F505

## 6.0 PROGRAM/VERIFY MODE ELECTRICAL CHARACTERISTICS

**TABLE 6-1: AC/DC CHARACTERISTICS TIMING REQUIREMENTS FOR PROGRAM/VERIFY MODE**

| AC/DC CHARACTERISTICS        |  | Standard Operating Conditions (unless otherwise stated)<br>Operating Temperature $10^{\circ}\text{C} \leq T_A \leq 40^{\circ}\text{C}$<br>Operating Voltage $4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$ |      |                   |               |                         |
|------------------------------|--|--|------|-------------------|---------------|-------------------------|
| Sym.                         | Characteristics  | Min.   | Typ. | Max.              | Units         | Conditions/<br>Comments |
| <b>General</b>               |  |  |      |                   |               |                         |
| VDDPROG                      | VDD level for programming operations, program memory   | 4.5  | —    | 5.5               | V             |                         |
| VDDERA                       | VDD level for Bulk Erase operations, program memory  | 4.5  | —    | 5.5               | V             |                         |
| IDDPROG                      | IDD level for programming operations, program memory   | —  | —    | 0.5               | mA            |                         |
| IDDERA                       | IDD level for Bulk Erase operations, program memory  | —  | —    | 0.5               | mA            |                         |
| VIHH                         | High voltage on $\overline{\text{MCLR}}$ for Program/Verify mode entry   | 12.5   | —    | 13.5              | V             |                         |
| I <sub>PP</sub>              | $\overline{\text{MCLR}}$ pin current during Program/Verify mode  | —  | —    | 0.45              | mA            |                         |
| T <sub>VHHR</sub>            | $\overline{\text{MCLR}}$ rise time ( $V_{SS}$ to $V_{IHH}$ ) for Program/Verify mode entry                             | —  | —    | 1.0               | $\mu\text{s}$ |                         |
| T <sub>PPDP</sub>            | Hold time after $V_{PP}\uparrow$   | 5  | —    | —                 | $\mu\text{s}$ |                         |
| V <sub>IH1</sub>             | (ICSPCLK, ICSPDAT) input high-level  | 0.8 $V_{DD}$   | —    | —                 | V             |                         |
| V <sub>L1</sub>              | (ICSPCLK, ICSPDAT) input low-level   | —  | —    | 0.2 $V_{DD}$      | V             |                         |
| T <sub>SET0</sub>            | ICSPCLK, ICSPDAT setup time before $\overline{\text{MCLR}}\uparrow$ (Program/Verify mode selection pattern setup time) | 100  | —    | —                 | ns            |                         |
| T <sub>HLD0</sub>            | ICSPCLK, ICSPDAT hold time after $\overline{\text{MCLR}}\uparrow$ (Program/Verify mode selection pattern setup time)   | 5  | —    | —                 | $\mu\text{s}$ |                         |
| <b>Serial Program/Verify</b> |  |  |      |                   |               |                         |
| T <sub>SET1</sub>            | Data in setup time before clock $\downarrow$   | 100  | —    | —                 | ns            |                         |
| T <sub>HLD1</sub>            | Data in hold time after clock $\downarrow$   | 100  | —    | —                 | ns            |                         |
| T <sub>DLY1</sub>            | Data input not driven to next clock input (delay required between command/data or command/command)                     | 1.0  | —    | —                 | $\mu\text{s}$ |                         |
| T <sub>DLY2</sub>            | Delay between clock $\downarrow$ to clock $\uparrow$ of next command or data   | 1.0  | —    | —                 | $\mu\text{s}$ |                         |
| T <sub>DLY3</sub>            | Clock $\uparrow$ to data out valid (during Read Data)  | —  | —    | 80                | ns            |                         |
| T <sub>ERA</sub>             | Erase cycle time   | —  | —    | 10 <sup>(1)</sup> | ms            |                         |
| T <sub>PROG</sub>            | Programming cycle time (externally timed)  | —  | —    | 2 <sup>(1)</sup>  | ms            |                         |
| T <sub>DIS</sub>             | Time delay for internal programming voltage discharge  | 100  | —    | —                 | $\mu\text{s}$ |                         |
| T <sub>RESET</sub>           | Time between exiting Program mode with VDD and VPP at GND and then re-entering Program mode by applying VDD.           | —  | 10   | —                 | ms            |                         |

**Note 1:** Minimum time to ensure that function completes successfully over voltage, temperature and device variations.

## REVISION HISTORY

### Revision G (02/2010)

Revised Section 1.1; Added QFN pin diagram; Revised Note 1, Table 1-1; Revised Section 3.1, 1st para; Added Revision History.

# PIC16F505

---

NOTES:

---

**Note the following details of the code protection feature on Microchip devices:**

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as "unbreakable."

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip's code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

---

Information contained in this publication regarding device applications and the like is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION, INCLUDING BUT NOT LIMITED TO ITS CONDITION, QUALITY, PERFORMANCE, MERCHANTABILITY OR FITNESS FOR PURPOSE. Microchip disclaims all liability arising from this information and its use. Use of Microchip devices in life support and/or safety applications is entirely at the buyer's risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights.

**Trademarks**

The Microchip name and logo, the Microchip logo, dsPIC, KEELOQ, KEELOQ logo, MPLAB, PIC, PICmicro, PICSTART, PIC<sup>32</sup> logo, rPIC and UNI/O are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

FilterLab, Hampshire, HI-TECH C, Linear Active Thermistor, MXDEV, MXLAB, SEEVAL and The Embedded Control Solutions Company are registered trademarks of Microchip Technology Incorporated in the U.S.A.

Analog-for-the-Digital Age, Application Maestro, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, HI-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPASM, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Octopus, Omniscient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICTail, REAL ICE, rLAB, Select Mode, Total Endurance, TSHARC, UniWinDriver, WiperLock and ZENA are trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

SQTP is a service mark of Microchip Technology Incorporated in the U.S.A.

All other trademarks mentioned herein are property of their respective companies.

© 2010, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 Printed on recycled paper.

ISBN: 978-1-60932-033-1

*Microchip received ISO/TS-16949:2002 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.*

**QUALITY MANAGEMENT SYSTEM  
CERTIFIED BY DNV  
== ISO/TS 16949:2002 ==**



## WORLDWIDE SALES AND SERVICE

### AMERICAS

**Corporate Office**  
2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 480-792-7200  
Fax: 480-792-7277  
Technical Support:  
<http://support.microchip.com>  
Web Address:  
[www.microchip.com](http://www.microchip.com)

**Atlanta**  
Duluth, GA  
Tel: 678-957-9614  
Fax: 678-957-1455

**Boston**  
Westborough, MA  
Tel: 774-760-0087  
Fax: 774-760-0088

**Chicago**  
Itasca, IL  
Tel: 630-285-0071  
Fax: 630-285-0075

**Cleveland**  
Independence, OH  
Tel: 216-447-0464  
Fax: 216-447-0643

**Dallas**  
Addison, TX  
Tel: 972-818-7423  
Fax: 972-818-2924

**Detroit**  
Farmington Hills, MI  
Tel: 248-538-2250  
Fax: 248-538-2260

**Kokomo**  
Kokomo, IN  
Tel: 765-864-8360  
Fax: 765-864-8387

**Los Angeles**  
Mission Viejo, CA  
Tel: 949-462-9523  
Fax: 949-462-9608

**Santa Clara**  
Santa Clara, CA  
Tel: 408-961-6444  
Fax: 408-961-6445

**Toronto**  
Mississauga, Ontario,  
Canada  
Tel: 905-673-0699  
Fax: 905-673-6509

### ASIA/PACIFIC

**Asia Pacific Office**  
Suites 3707-14, 37th Floor  
Tower 6, The Gateway  
Harbour City, Kowloon  
Hong Kong  
Tel: 852-2401-1200  
Fax: 852-2401-3431

**Australia - Sydney**  
Tel: 61-2-9868-6733  
Fax: 61-2-9868-6755

**China - Beijing**  
Tel: 86-10-8528-2100  
Fax: 86-10-8528-2104

**China - Chengdu**  
Tel: 86-28-8665-5511  
Fax: 86-28-8665-7889

**China - Chongqing**  
Tel: 86-23-8980-9588  
Fax: 86-23-8980-9500

**China - Hong Kong SAR**  
Tel: 852-2401-1200  
Fax: 852-2401-3431

**China - Nanjing**  
Tel: 86-25-8473-2460  
Fax: 86-25-8473-2470

**China - Qingdao**  
Tel: 86-532-8502-7355  
Fax: 86-532-8502-7205

**China - Shanghai**  
Tel: 86-21-5407-5533  
Fax: 86-21-5407-5066

**China - Shenyang**  
Tel: 86-24-2334-2829  
Fax: 86-24-2334-2393

**China - Shenzhen**  
Tel: 86-755-8203-2660  
Fax: 86-755-8203-1760

**China - Wuhan**  
Tel: 86-27-5980-5300  
Fax: 86-27-5980-5118

**China - Xian**  
Tel: 86-29-8833-7252  
Fax: 86-29-8833-7256

**China - Xiamen**  
Tel: 86-592-2388138  
Fax: 86-592-2388130

**China - Zhuhai**  
Tel: 86-756-3210040  
Fax: 86-756-3210049

### ASIA/PACIFIC

**India - Bangalore**  
Tel: 91-80-3090-4444  
Fax: 91-80-3090-4123

**India - New Delhi**  
Tel: 91-11-4160-8631  
Fax: 91-11-4160-8632

**India - Pune**  
Tel: 91-20-2566-1512  
Fax: 91-20-2566-1513

**Japan - Yokohama**  
Tel: 81-45-471- 6166  
Fax: 81-45-471-6122

**Korea - Daegu**  
Tel: 82-53-744-4301  
Fax: 82-53-744-4302

**Korea - Seoul**  
Tel: 82-2-554-7200  
Fax: 82-2-558-5932 or  
82-2-558-5934

**Malaysia - Kuala Lumpur**  
Tel: 60-3-6201-9857  
Fax: 60-3-6201-9859

**Malaysia - Penang**  
Tel: 60-4-227-8870  
Fax: 60-4-227-4068

**Philippines - Manila**  
Tel: 63-2-634-9065  
Fax: 63-2-634-9069

**Singapore**  
Tel: 65-6334-8870  
Fax: 65-6334-8850

**Taiwan - Hsin Chu**  
Tel: 886-3-6578-300  
Fax: 886-3-6578-370

**Taiwan - Kaohsiung**  
Tel: 886-7-536-4818  
Fax: 886-7-536-4803

**Taiwan - Taipei**  
Tel: 886-2-2500-6610  
Fax: 886-2-2508-0102

**Thailand - Bangkok**  
Tel: 66-2-694-1351  
Fax: 66-2-694-1350

### EUROPE

**Austria - Wels**  
Tel: 43-7242-2244-39  
Fax: 43-7242-2244-393

**Denmark - Copenhagen**  
Tel: 45-4450-2828  
Fax: 45-4485-2829

**France - Paris**  
Tel: 33-1-69-53-63-20  
Fax: 33-1-69-30-90-79

**Germany - Munich**  
Tel: 49-89-627-144-0  
Fax: 49-89-627-144-44

**Italy - Milan**  
Tel: 39-0331-742611  
Fax: 39-0331-466781

**Netherlands - Drunen**  
Tel: 31-416-690399  
Fax: 31-416-690340

**Spain - Madrid**  
Tel: 34-91-708-08-90  
Fax: 34-91-708-08-91

**UK - Wokingham**  
Tel: 44-118-921-5869  
Fax: 44-118-921-5820

01/05/10