

---

**PIC16(L)F145X Memory Programming Specification**

---

**This document includes the programming specifications for the following devices:**

- PIC16F1454    • PIC16LF1454
- PIC16F1455    • PIC16LF1455
- PIC16F1459    • PIC16LF1459

## **1.0 OVERVIEW**

The devices can be programmed using either the high-voltage In-Circuit Serial Programming™ (ICSP™) method or the low-voltage ICSP™ method.

When using low-voltage ICSP™ programming (LVP = 1), the ICSPDAT/ICSPCLK functions are additionally enabled on the RA0/RA1 port pins. This legacy programming feature provides compatibility support for existing PIC18F1XK50 designs. For new designs, using the ICSPDAT/ICSPCLK functions on the RC0/RC1 port pins is recommended.

For Legacy Programming Support, refer to [Section 4.2.1 “Legacy ICSP Pinout Programming”](#).

## **1.1 Hardware Requirements**

### **1.1.1 HIGH-VOLTAGE ICSP PROGRAMMING**

In High-Voltage ICSP™ mode, these devices require two programmable power supplies: one for VDD and one for the MCLR/VPP pin.

### **1.1.2 LOW-VOLTAGE ICSP PROGRAMMING**

In Low-Voltage ICSP™ mode, these devices can be programmed using a single VDD source in the operating range. The MCLR/VPP pin does not have to be brought to a different voltage, but can instead be left at the normal operating voltage.

#### **1.1.2.1 Single-Supply ICSP Programming**

The LVP bit in Configuration Word 2 enables single-supply (low-voltage) ICSP programming. The LVP bit defaults to a '1' (enabled) from the factory. The LVP bit may only be programmed to '0' by entering the High-Voltage ICSP mode, where the MCLR/VPP pin is raised to VIH. Once the LVP bit is programmed to a '0', only the High-Voltage ICSP mode is available and only the High-Voltage ICSP mode can be used to program the device.

**Note 1:** The High-Voltage ICSP mode is always available, regardless of the state of the LVP bit, by applying VIH to the MCLR/VPP pin.

**2:** While in Low-Voltage ICSP mode, MCLR is always enabled, regardless of the MCLRE bit, and the port pin can no longer be used as a general purpose input.

# PIC16(L)F145X

## 1.2 Pin Utilization

Five pins are needed for ICSP™ programming. It is recommended to use the pins listed in [Table 1-1](#). For PIC18F1XK50 legacy designs refer to [Section 4.2.1 “Legacy ICSP Pinout Programming”](#).

**TABLE 1-1: RECOMMENDED PIN DESCRIPTIONS DURING PROGRAMMING**

Pin Name	During Programming		
	Function	Pin Type	Pin Description
RC1	ICSPCLK	I	Clock Input – Schmitt Trigger Input
RC0	ICSPDAT	I/O	Data Input/Output – Schmitt Trigger Input
MCLR/VPP/RA3	Program/Verify mode	P <sup>(1)</sup>	Program Mode Select/Programming Power Supply
VDD	VDD	P	Power Supply
VSS	VSS	P	Ground

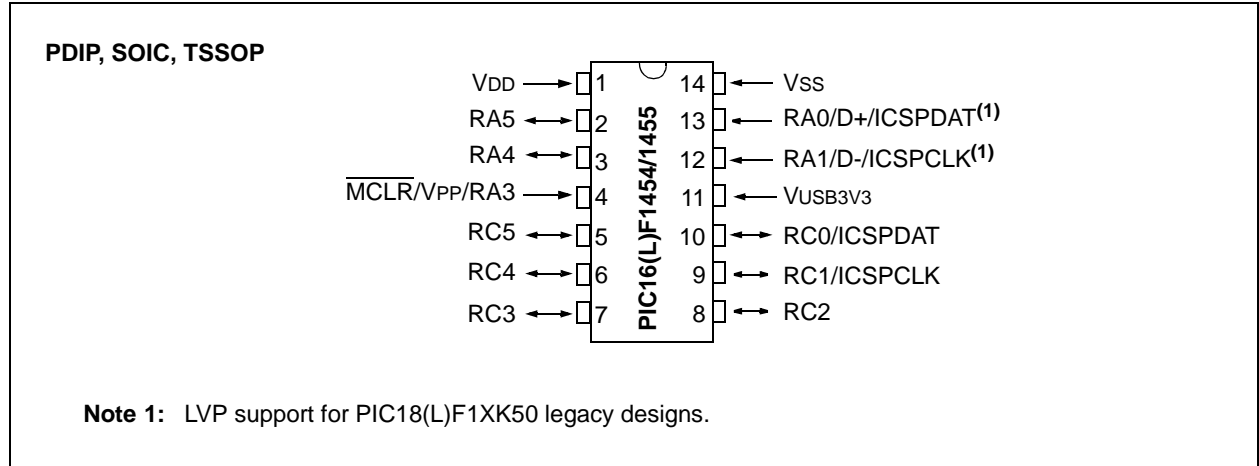
**Legend:** I = Input, O = Output, P = Power

**Note 1:** The programming high voltage is internally generated. To activate the Program/Verify mode, high voltage needs to be applied to MCLR input. Since the MCLR is used for a level source, MCLR does not draw any significant current.

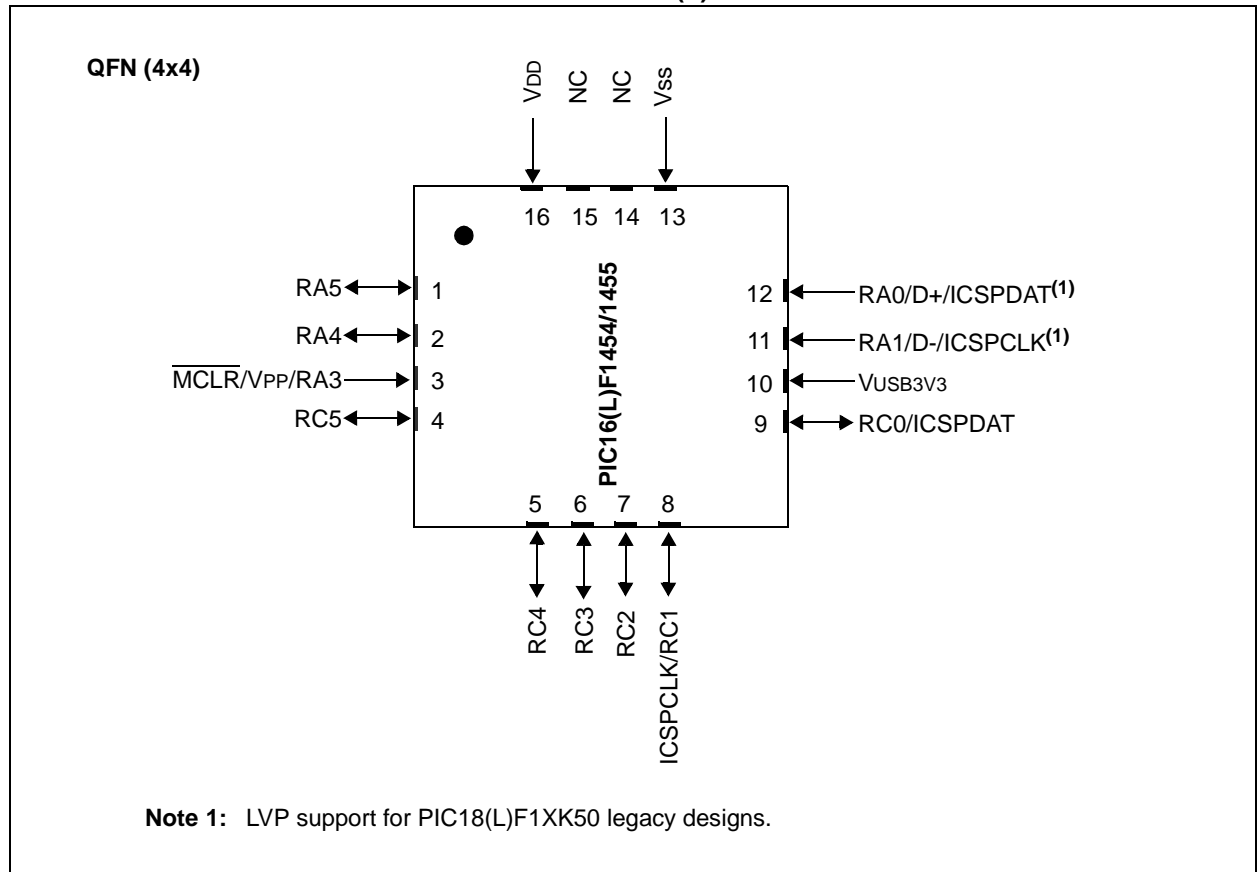
## 2.0 DEVICE PINOUTS

The pin diagrams are shown in [Figure 2-1](#) through [Figure 2-4](#). The pins that are required for programming are listed in [Table 1-1](#) and shown in bold lettering in the pin diagrams.

**FIGURE 2-1: 14-PIN PDIP, SOIC, TSSOP DIAGRAM FOR PIC16(L)F1454/1455**

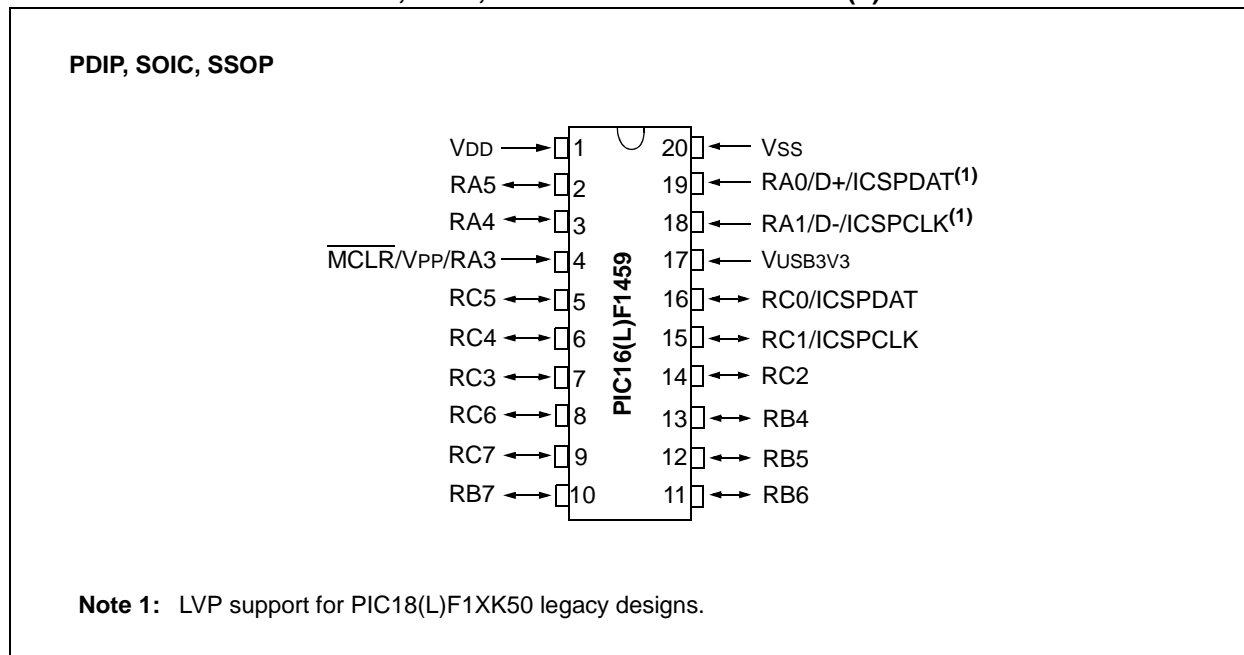


**FIGURE 2-2: 16-PIN QFN DIAGRAM FOR PIC16(L)F1454/1455**

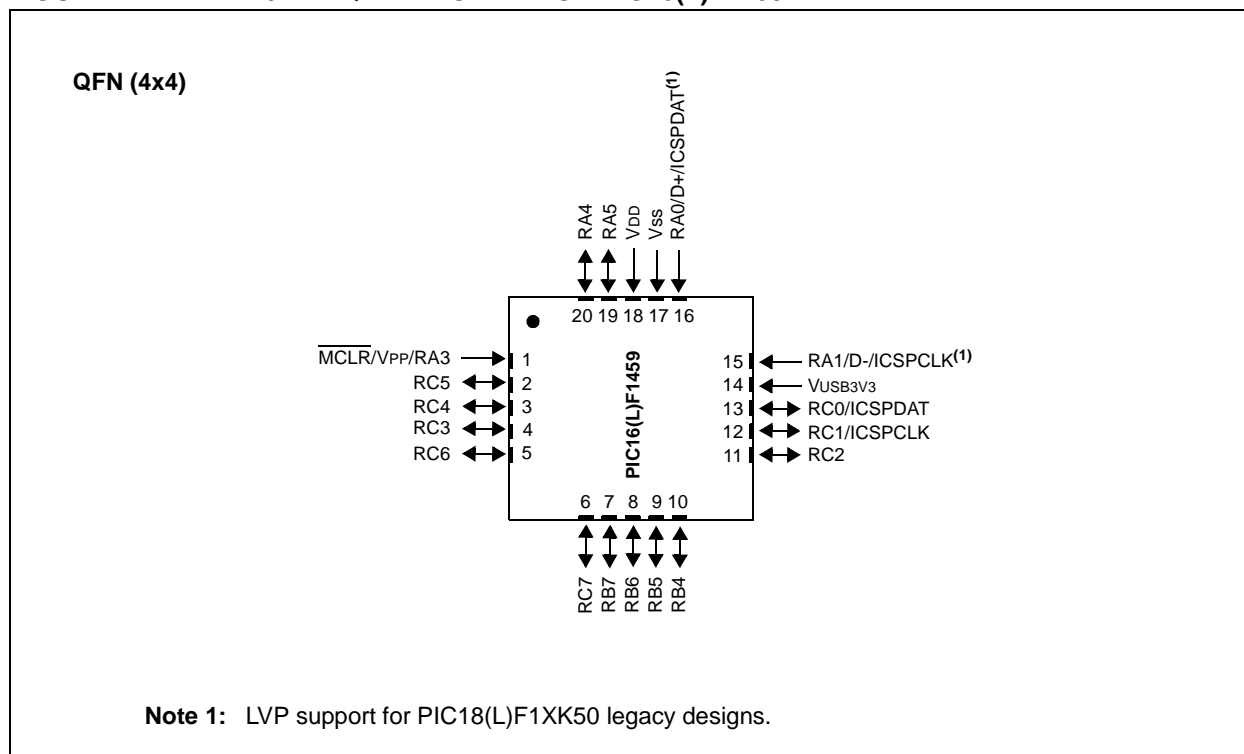


# PIC16(L)F145X

**FIGURE 2-3: 20-PIN PDIP, SOIC, SSOP DIAGRAM FOR PIC16(L)F1459**



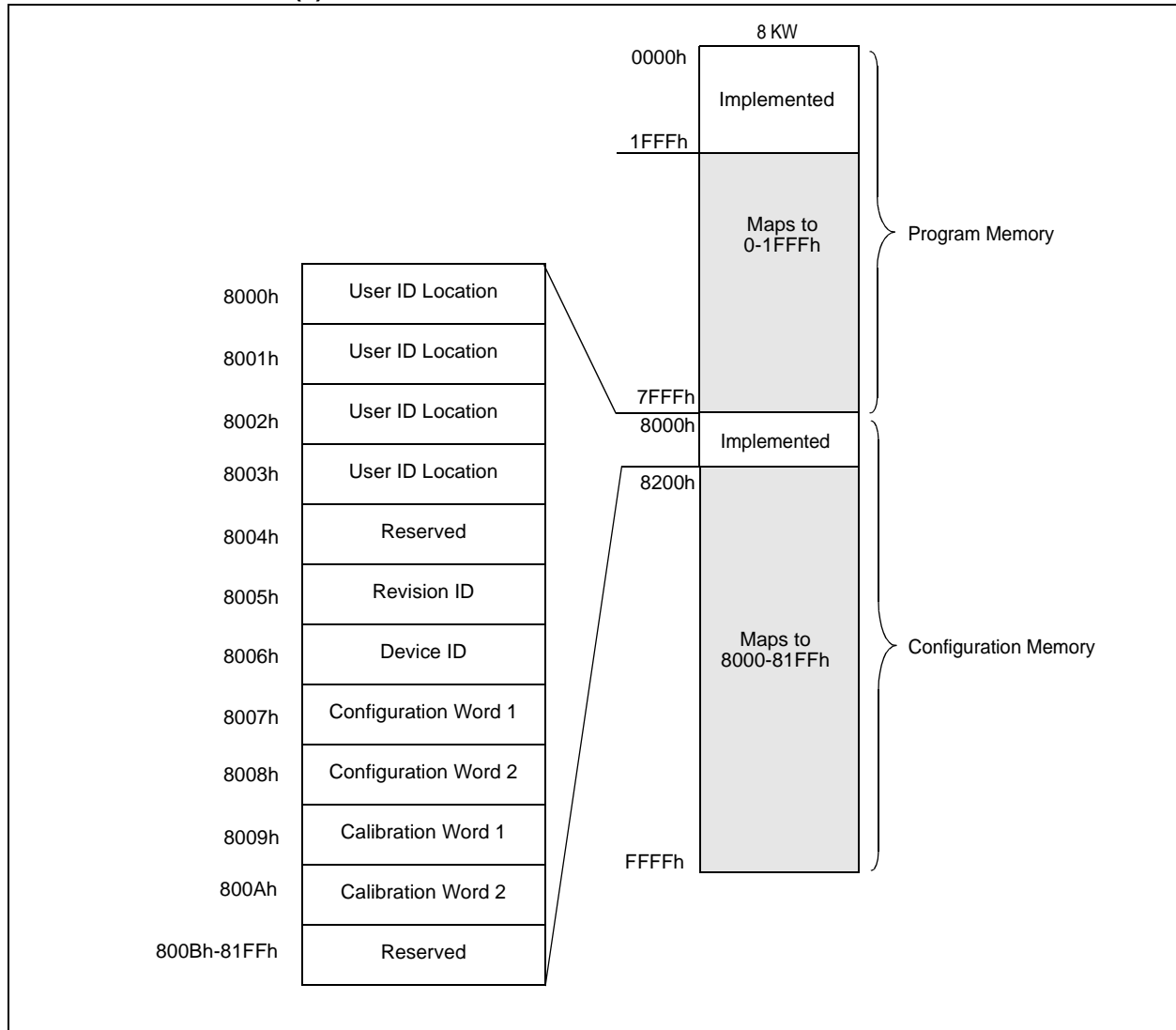
**FIGURE 2-4: 20-PIN QFN DIAGRAM FOR PIC16(L)F1459**



## 3.0 MEMORY MAP

The memory is broken into two sections: program memory and configuration memory. Only the size of the program memory changes between devices, the configuration memory remains the same.

**FIGURE 3-1: PIC16(L)F1454/1455/1459 PROGRAM MEMORY MAPPING**



# PIC16(L)F145X

## 3.1 User ID Location

A user may store identification information (user ID) in four designated locations. The user ID locations are mapped to 8000h-8003h. Each location is 14 bits in length. Code protection has no effect on these memory locations. Each location may be read with code protection enabled or disabled.

**Note:** MPLAB® IDE only displays the 7 Least Significant bits (LSb) of each user ID location, the upper bits are not read. It is recommended that only the 7 LSbs be used if MPLAB IDE is the primary tool used to read these addresses.

## 3.2 Revision ID

The Revision ID word is located at 8005h. This location is read-only and cannot be erased or modified.

**REGISTER 3-1: REVISION ID – 8005h<sup>(1)</sup>**

R	R	R	R	R	R
REV<13:8>					
bit 13			bit 8		

R	R	R	R	R	R	R	R
REV<7:0>							
bit 7				bit 0			

**Legend:** R = Readable bit

bit 13-0 **REV<13:0>**: Revision ID bits  
These bits are used to identify the revision.

**Note 1:** This location cannot be written.

## 3.3 Device ID

The device ID word is located at 8006h. This location is read-only and cannot be erased or modified.

**REGISTER 3-2: DEVICE ID: DEVICE ID REGISTER<sup>(1)</sup>**

R	R	R	R	R	R
DEV<13:8>					
bit 13			bit 8		

R	R	R	R	R	R	R	R
DEV<7:0>							
bit 7				bit 0			

**Legend:** R = Readable bit

bit 13-0 **DEV<13:0>**: Device ID bits  
These bits are used to identify the part number.

**Note 1:** This location cannot be written.

**TABLE 3-1: DEVICE ID VALUES**

DEVICE	DEVICE ID VALUES
	DEV<13:0>
PIC16F1454	0011 0000 0010 0000 (3020h)
PIC16LF1454	0011 0000 0010 0100 (3024h)
PIC16F1455	0011 0000 0010 0001 (3021h)
PIC16LF1455	0011 0000 0010 0101 (3025h)
PIC16F1459	0011 0000 0010 0011 (3023h)
PIC16LF1459	0011 0000 0010 0111 (3027h)

## 3.4 Configuration Words

There are two Configuration Words, Configuration Word 1 (8007h) and Configuration Word 2 (8008h). The individual bits within these Configuration Words are used to enable or disable device functions such as the Brown-out Reset, code protection and Power-up Timer.

## 3.5 Calibration Words

The internal calibration values are factory calibrated and stored in Calibration Words 1 and 2 (8009h, 800Ah).

The Calibration Words do not participate in erase operations. The device can be erased without affecting the Calibration Words.



## REGISTER 3-3: CONFIGURATION WORD 1

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	U-1
FCMEN	IESO	CLKOUTEN	BOREN<1:0>		—
bit 13					bit 8

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
CP	MCLR	PWRT	WDTE<1:0>		FOSC<2:0>		
bit 7							bit 0

<b>Legend:</b>	R = Readable bit	P = Programmable Bit	U = Unimplemented bit
	-n = Value after Bulk Erase	'1' = Bit is set	'0' = Bit is cleared

- bit 13 **FCMEN:** Fail-Safe Clock Monitor Enable bit  
 1 = Fail-Safe Clock Monitor is enabled  
 0 = Fail-Safe Clock Monitor is disabled
- bit 12 **IESO:** Internal/External Switchover bit  
 1 = Internal/External Switchover mode is enabled  
 0 = Internal/External Switchover mode is disabled
- bit 11 **CLKOUTEN:** Clock Out Enable bit  
 1 = CLKOUT function is disabled. I/O or oscillator function on CLKOUT pin.  
 0 = CLKOUT function is enabled on CLKOUT pin
- bit 10-9 **BOREN<1:0>:** Brown-out Reset Enable bits<sup>(1)</sup>  
 11 = BOR enabled  
 10 = BOR enabled during operation and disabled in Sleep  
 01 = BOR controlled by SBOREN bit of the PCON register  
 00 = BOR disabled
- bit 8 **Unimplemented:** Read as '1'
- bit 7 **CP:** Code Protection bit<sup>(2)</sup>  
 1 = Program memory code protection is disabled  
 0 = Program memory code protection is enabled
- bit 6 **MCLR:** MCLR/VPP Pin Function Select bit  
 If LVP bit = 1:  
 This bit is ignored.  
 If LVP bit = 0:  
 1 = MCLR/VPP pin function is MCLR; Weak pull-up enabled.  
 0 = MCLR/VPP pin function is digital input; MCLR internally disabled; Weak pull-up under control of WPUA register.
- bit 5 **PWRT:** Power-up Timer Enable bit<sup>(1)</sup>  
 1 = PWRT disabled  
 0 = PWRT enabled
- bit 4-3 **WDTE<1:0>:** Watchdog Timer Enable bit  
 11 = WDT enabled  
 10 = WDT enabled while running and disabled in Sleep  
 01 = WDT controlled by the SWDTEN bit in the WDTCON register  
 00 = WDT disabled
- bit 2-0 **FOSC<2:0>:** Oscillator Selection bits  
 111 = ECH: External Clock, High-Power mode: on CLKIN pin  
 110 = ECM: External Clock, Medium-Power mode: on CLKIN pin  
 101 = ECL: External Clock, Low-Power mode: on CLKIN pin  
 100 = INTOSC oscillator: I/O function on OSC1 pin  
 011 = EXTRC oscillator: RC function on CLKIN pin  
 010 = HS oscillator: High-speed crystal/resonator on OSC1 and OSC2 pins  
 001 = XT oscillator: Crystal/resonator on OSC1 and OSC2 pins  
 000 = LP oscillator: Low-power crystal on OSC1 and OSC2 pins

**Note 1:** Enabling Brown-out Reset does not automatically enable Power-up Timer.  
**Note 2:** Once enabled (CP = 0), code protection can only be disabled by bulk erasing the device.

# PIC16(L)F145X

## REGISTER 3-4: CONFIGURATION WORD 2

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
LVP	DEBUG	LPBOR	BORV	STVREN	PLLEN
bit 13					bit 8

R/P-1	R/P-1	R/P-1	R/P-1	U-1	U-1	R/P-1	R/P-1
PLLMULT	USBSLCLK	CPUDIV<1:0>	—	—	—	WRT<1:0>	—
bit 7							bit 0

<b>Legend:</b>	R = Readable bit -n = Value after Bulk Erase	P = Programmable Bit '1' = Bit is set	U = Unimplemented bit '0' = Bit is cleared
----------------	-------------------------------------------------	------------------------------------------	-----------------------------------------------

- bit 13 **LVP:** Low-Voltage Programming Enable bit<sup>(1), (2)</sup>  
1 = Low-voltage programming enabled  
0 = HV on MCLR/VPP must be used for programming
- bit 12 **DEBUG:** Debugger mode bit  
1 = In-Circuit Debugger disabled, ICSPCLK and ICSPDAT pins are general purpose I/O pins  
0 = In-Circuit Debugger enabled, ICSPCLK and ICSPDAT pins are dedicated to the debugger
- bit 11 **LPBOR:** Low-Power BOR bit  
1 = Low-Power BOR is disabled  
0 = Low-Power BOR is enabled
- bit 10 **BORV:** Brown-out Reset Voltage Selection bit  
1 = Brown-out Reset Voltage (VBOR), low trip point selected  
0 = Brown-out Reset Voltage (VBOR), high trip point selected
- bit 9 **STVREN:** Stack Overflow/Underflow Reset Enable bit  
1 = Stack Overflow or Underflow will cause a Reset  
0 = Stack Overflow or Underflow will not cause a Reset
- bit 8 **PLLEN:** PLL Enable bit  
1 = PLL enabled  
0 = PLL disabled
- bit 7 **PLLMULT:** PLL Multiplier Selection bit  
1 = 3x PLL Output Frequency is selected  
0 = 4x PLL Output Frequency is selected
- bit 6 **USBSLCLK:** USB Low-Speed Clock Selection bit  
1 = USB Clock divide-by 8, (48 MHz System input clock expected)  
0 = USB Clock divide-by 4, (24 MHz System input clock expected)
- bit 5-4 **CPUDIV<1:0>:** CPU System Clock Selection bits  
11 = CPU system clock divided by 6  
10 = CPU system clock divided by 3  
01 = CPU system clock divided by 2  
00 = No CPU system clock divide.
- bit 3-2 **Unimplemented:** Read as '1'
- bit 1-0 **WRT<1:0>:** Flash Memory Self-Write Protection bits  
8 kW Flash memory (PIC16(L)F1454/1455/1459):  
11 = Write protection off  
10 = 0000h to 01FFh write-protected, 0200h to 1FFFh may be modified by PMCON control  
01 = 0000h to 0FFFh write-protected, 1000h to 1FFFh may be modified by PMCON control  
00 = 0000h to 1FFFh write-protected, no addresses may be modified by PMCON control

- Note 1:** The LVP bit cannot be programmed to '0' when Programming mode is entered via LVP.
- Note 2:** When LVP = 1, the ICSP functions are enabled on the ICSPDAT/D+/RA0 and ICSPCLK/D-/RA1 pins.

## 4.0 PROGRAM/VERIFY MODE

In Program/Verify mode, the program memory and the configuration memory can be accessed and programmed in serial fashion. ICSPDAT and ICSPCLK are used for the data and the clock, respectively. All commands and data words are transmitted LSb first. Data changes on the rising edge of the ICSPCLK and latched on the falling edge. In Program/Verify mode both the ICSPDAT and ICSPCLK are Schmitt Trigger inputs. The sequence that enters the device into Program/Verify mode places all other logic into the Reset state. Upon entering Program/Verify mode, all I/Os are automatically configured as high-impedance inputs and the address is cleared.

### 4.1 High-Voltage Program/Verify Mode Entry and Exit

There are two different methods of entering Program/Verify mode via high-voltage:

- VPP – First entry mode
- VDD – First entry mode

#### 4.1.1 VPP – FIRST ENTRY MODE

To enter Program/Verify mode via the VPP-first method the following sequence must be followed:

1. Hold ICSPCLK and ICSPDAT low. All other pins should be unpowered.
2. Raise the voltage on  $\overline{\text{MCLR}}$  from 0V to  $V_{IH}$ .
3. Raise the voltage on VDD from 0V to the desired operating voltage.

The VPP-first entry prevents the device from executing code prior to entering Program/Verify mode. For example, the device will execute code when Configuration Word 1 has  $\overline{\text{MCLR}}$  disabled ( $\text{MCLRE} = 0$ ), the Power-up Timer is disabled ( $\text{PWRTS} = 0$ ), the internal oscillator is selected ( $\text{FOSC} = 100$ ), and ICSPCLK and ICSPDAT pins are driven by the user application. Since this may prevent entry, VPP-first Entry mode is strongly recommended. See the timing diagram in [Figure 8-2](#).

#### 4.1.2 VDD – FIRST ENTRY MODE

To enter Program/Verify mode via the VDD-first method the following sequence must be followed:

1. Hold ICSPCLK and ICSPDAT low.
2. Raise the voltage on VDD from 0V to the desired operating voltage.
3. Raise the voltage on  $\overline{\text{MCLR}}$  from VDD or below to  $V_{IH}$ .

The VDD-first method is useful when programming the device when VDD is already applied, for it is not necessary to disconnect VDD to enter Program/Verify mode. See the timing diagram in [Figure 8-1](#).

#### 4.1.3 PROGRAM/VERIFY MODE EXIT

To exit Program/Verify mode take  $\overline{\text{MCLR}}$  to VDD or lower ( $V_{IL}$ ). See [Figures 8-3](#) and [8-4](#).

### 4.2 Low-Voltage Programming (LVP) Mode

The Low-Voltage Programming mode allows devices to be programmed using VDD only, without high voltage. When the LVP bit of Configuration Word 2 register is set to '1', the low-voltage ICSP programming entry is enabled. To disable the Low-Voltage ICSP mode, the LVP bit must be programmed to '0'. This can only be done while in the High-Voltage Entry mode.

Entry into the Low-Voltage ICSP Program/Verify modes requires the following steps:

1.  $\overline{\text{MCLR}}$  is brought to  $V_{IL}$ .
2. A 32-bit key sequence is presented on ICSPDAT, while clocking ICSPCLK.

The key sequence is a specific 32-bit pattern, '0100 1101 0100 0011 0100 1000 0101 0000' (more easily remembered as MCHP in ASCII). The device will enter Program/Verify mode only if the sequence is valid. The Least Significant bit of the Least Significant nibble must be shifted in first.

Once the key sequence is complete,  $\overline{\text{MCLR}}$  must be held at  $V_{IL}$  for as long as Program/Verify mode is to be maintained.

For low-voltage programming timing, see [Figure 8-8](#) and [Figure 8-9](#).

Exiting Program/Verify mode is done by no longer driving  $\overline{\text{MCLR}}$  to  $V_{IL}$ . See [Figure 8-8](#) and [Figure 8-9](#).

**Note:** To enter LVP mode, the LSB of the Least Significant nibble must be shifted in first. This differs from entering the key sequence on other parts.

# PIC16(L)F145X

## 4.2.1 LEGACY ICSP PINOUT PROGRAMMING

The legacy ICSP pinout is for use with designs based on previous PIC18F1XK50 20-pin parts. This allows the programming through the same pinout layout. If LVP = 1, the PIC<sup>®</sup> device can be programmed through the recommended pins or the legacy pins (refer to Table 4-1).

When LVP is set, entry into the Low-Voltage ICSP Program/Verify mode, as well as the programming method, is the same as discussed in Section 4.2 “Low-Voltage Programming (LVP) Mode”. However, both the legacy and recommended ICSPDAT are

monitored while clocking ICSPCLK. Whichever ICSPDAT first receives the 32-bit key sequence will be used for programming with its corresponding ICSPCLK.

**Note:** The V<sub>IH</sub> voltage levels on port pins RA0/ICSPDAT/D+ and RA1/ICSPCLK/D- must be limited to 3.3V maximum, due to USB circuitry. The device must not be attached to a USB host and the USB module must be disabled. Refer to Figure 4-1, Figure 4-2 and Figure 4-3.

**TABLE 4-1: LEGACY PIN DESCRIPTIONS DURING PROGRAMMING**

Pin Name	During Programming		
	Function	Pin Type	Pin Description
RA1	ICSPCLK	I	Clock Input – Schmitt Trigger Input
RA0	ICSPDAT	I	Data Input – Schmitt Trigger Input
MCLR/VPP/RA3	Program/Verify mode	P <sup>(1)</sup>	Program Mode Select/Programming Power Supply
VDD	VDD	P	Power Supply
VSS	VSS	P	Ground

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

**Note 1:** The programming high voltage is internally generated. To activate the Program/Verify mode, high voltage needs to be applied to MCLR input. Since the MCLR is used for a level source, MCLR does not draw any significant current.

**FIGURE 4-1: IN-CIRCUIT PROGRAMMING (ICSP) RECOMMENDED METHOD**

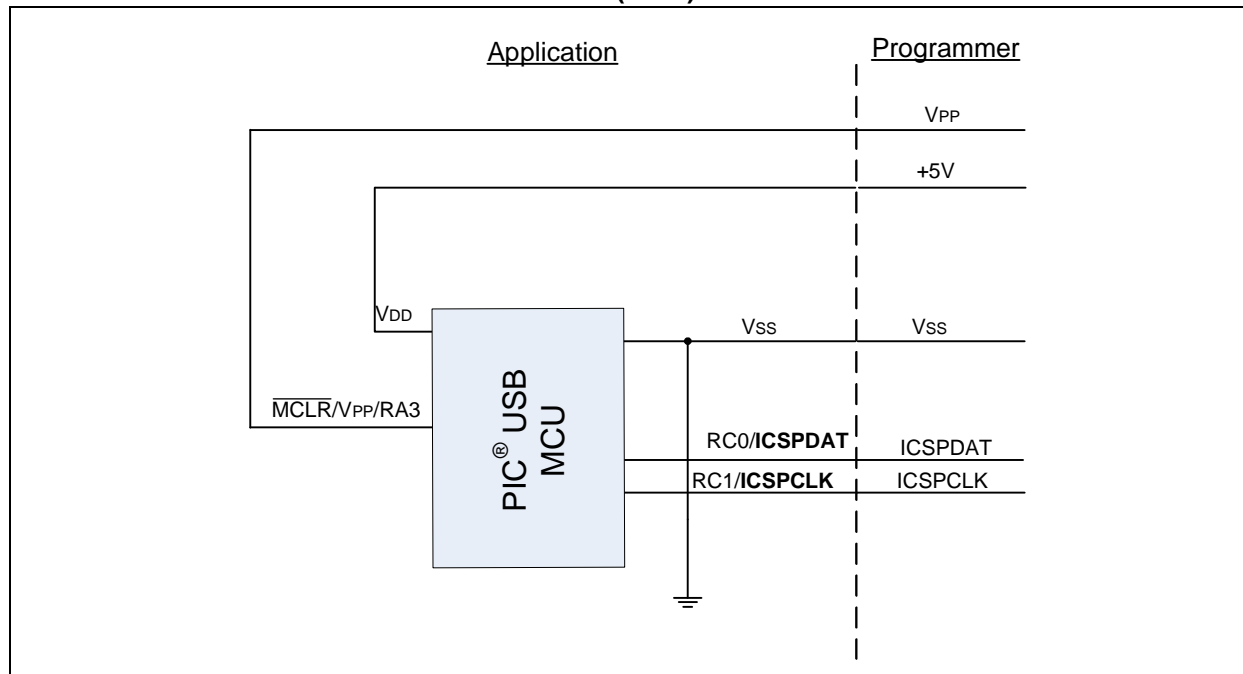


FIGURE 4-2: IN-CIRCUIT PROGRAMMING (ICSP) LEGACY WITH TRANSLATOR

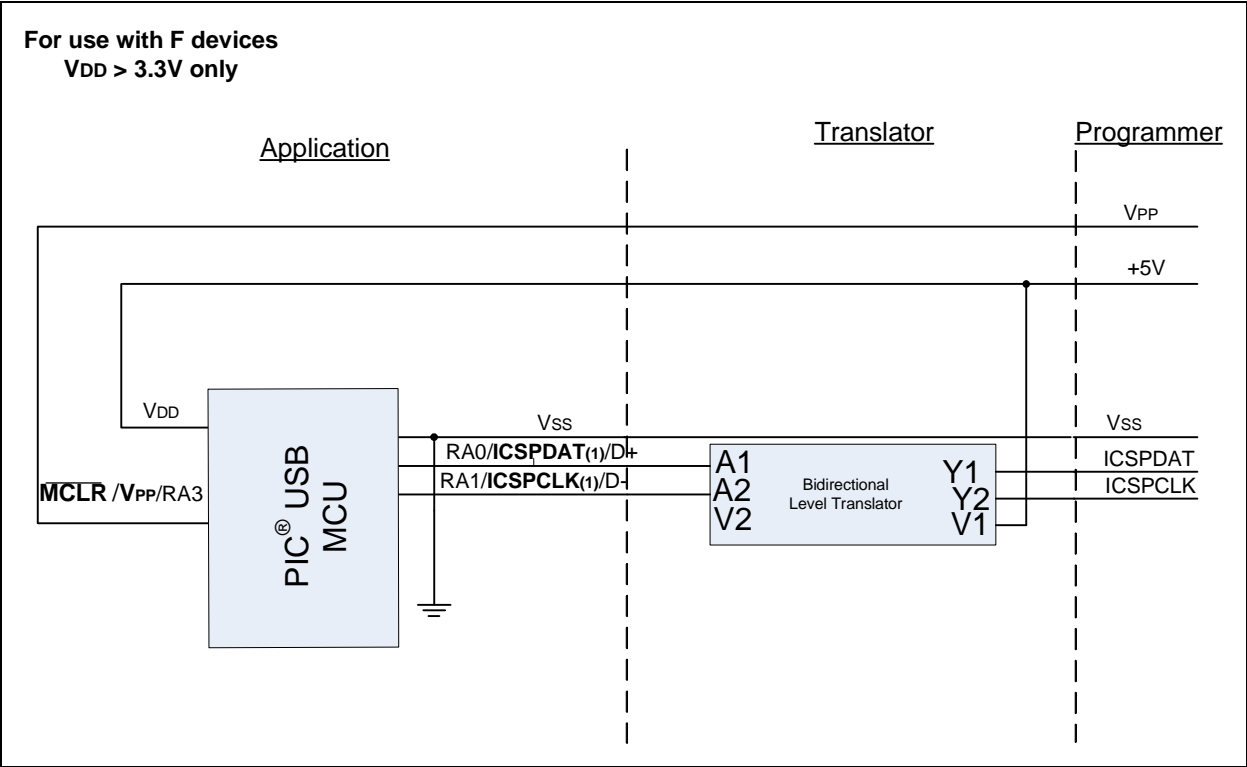
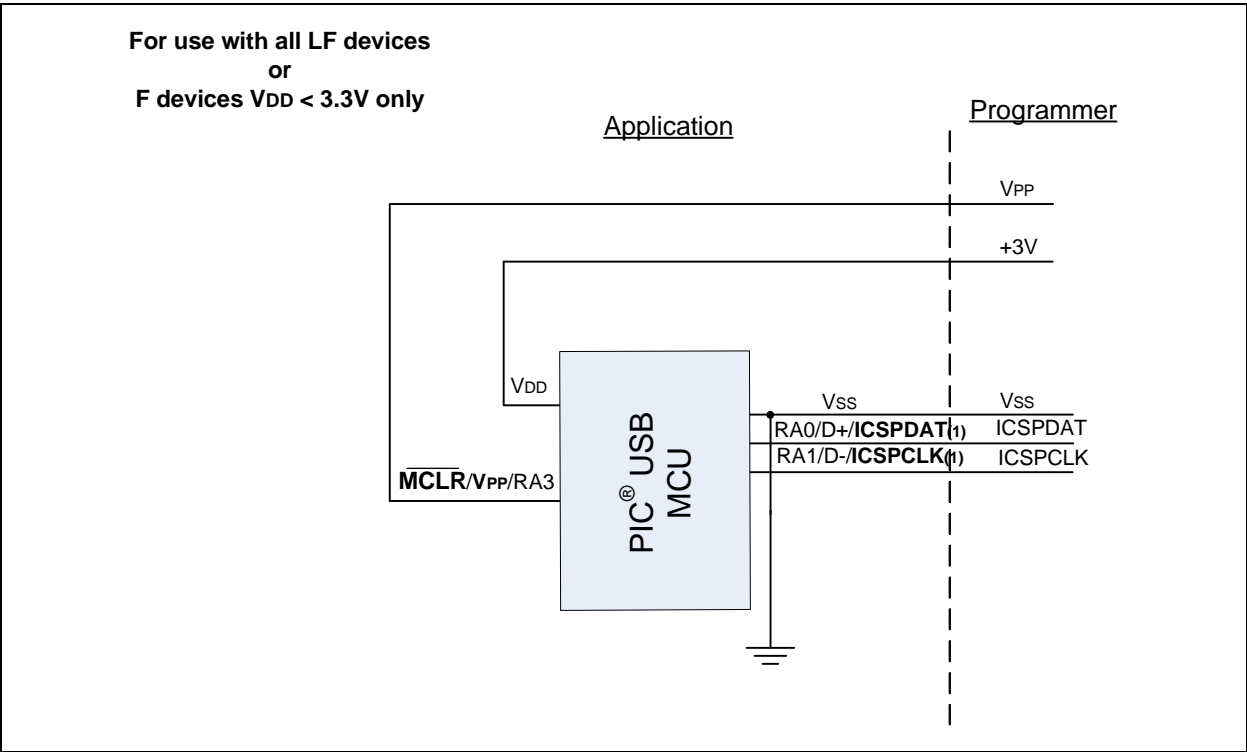


FIGURE 4-3: IN-CIRCUIT PROGRAMMING (ICSP) LEGACY



# PIC16(L)F145X

## 4.3 Program/Verify Commands

The devices implement 10 programming commands; each six bits in length. The commands are summarized in [Table 4-2](#).

Commands that have data associated with them are specified to have a minimum delay of TDLY between the command and the data. After this delay 16 clocks are required to either clock in or clock out the 14-bit data word. The first clock is for the Start bit and the last clock is for the Stop bit.

**TABLE 4-2: COMMAND MAPPING**

Command	Mapping							Data/Note
	Binary (MSb ... LSb)						Hex	
Load Configuration	x	0	0	0	0	0	00h	0, data (14), 0
Load Data For Program Memory	x	0	0	0	1	0	02h	0, data (14), 0
Read Data From Program Memory	x	0	0	1	0	0	04h	0, data (14), 0
Increment Address	x	0	0	1	1	0	06h	—
Reset Address	x	1	0	1	1	0	16h	—
Begin Internally Timed Programming	x	0	1	0	0	0	08h	—
Begin Externally Timed Programming	x	1	1	0	0	0	18h	—
End Externally Timed Programming	x	0	1	0	1	0	0Ah	—
Bulk Erase Program Memory	x	0	1	0	0	1	09h	Internally Timed
Row Erase Program Memory	x	1	0	0	0	1	11h	Internally Timed

### 4.3.1 LOAD CONFIGURATION

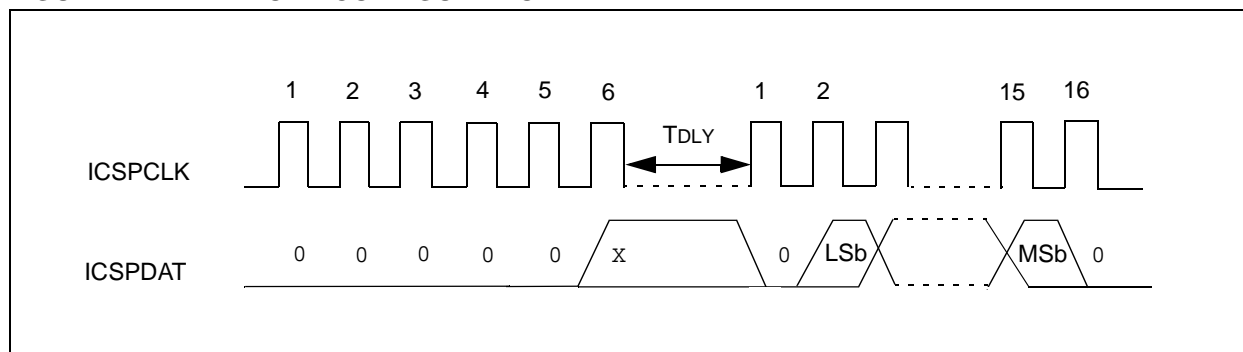
The Load Configuration command is used to access the configuration memory (user ID locations, Configuration Words, Calibration Words). The Load Configuration command sets the address to 8000h and loads the data latches with one word of data (see [Figure 4-4](#)).

After issuing the Load Configuration command, use the Increment Address command until the proper address to be programmed is reached. The address is then programmed by issuing either the Begin Internally Timed Programming or Begin Externally Timed Programming command.

**Note:** Externally timed writes are not supported for Configuration and Calibration bits. Any externally timed write to the Configuration or Calibration Word will have no effect on the targeted word.

The only way to get back to the program memory (address 0) is to exit Program/Verify mode or issue the Reset Address command after the configuration memory has been accessed by the Load Configuration command.

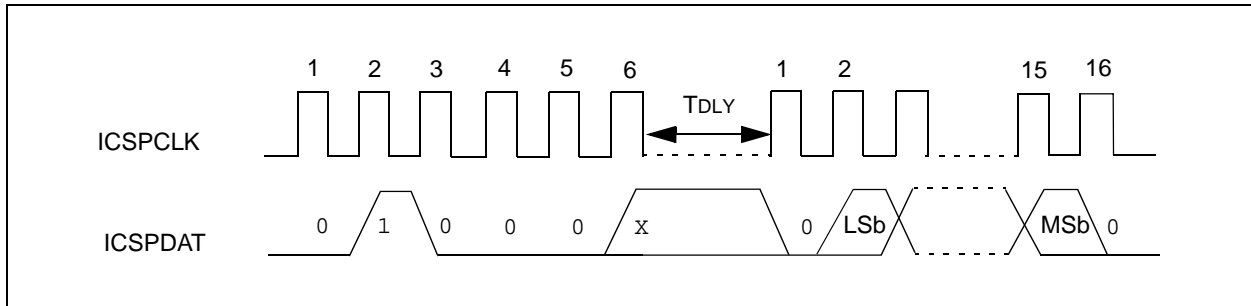
**FIGURE 4-4: LOAD CONFIGURATION**



## 4.3.2 LOAD DATA FOR PROGRAM MEMORY

The Load Data for Program Memory command is used to load one 14-bit word into the data latches. The word programs into program memory after the Begin Internally Timed Programming or Begin Externally Timed Programming command is issued (see [Figure 4-5](#)).

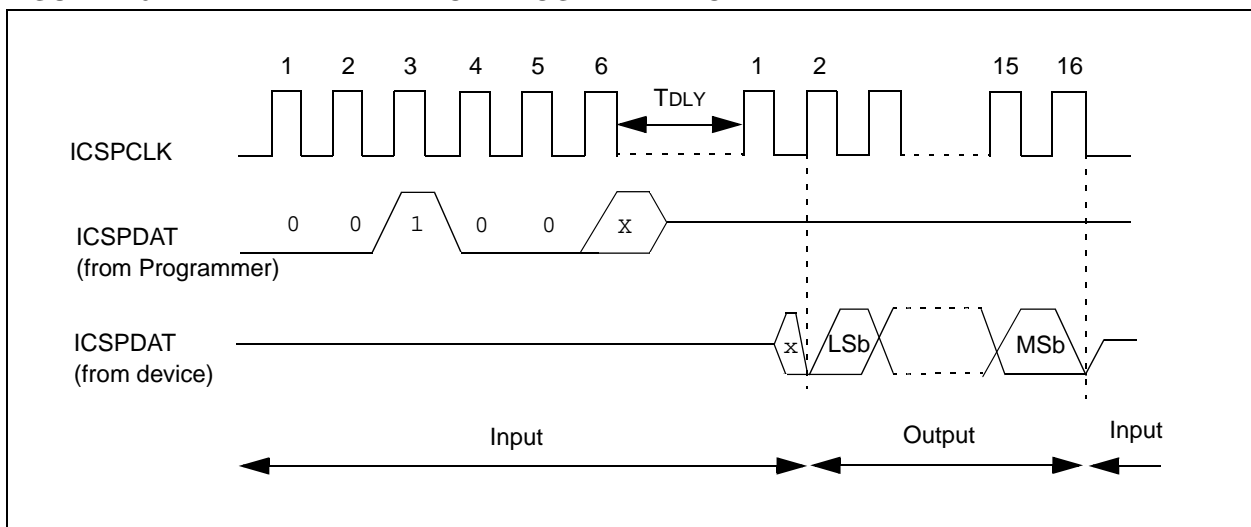
**FIGURE 4-5: LOAD DATA FOR PROGRAM MEMORY**



## 4.3.3 READ DATA FROM PROGRAM MEMORY

The Read Data from Program Memory command will transmit data bits out of the program memory map currently accessed, starting with the second rising edge of the clock input. The ICSPDAT pin will go into Output mode on the first falling clock edge, and it will revert to Input mode (high-impedance) after the 16th falling edge of the clock. If the program memory is code-protected ( $\overline{CP}$ ), the data will be read as zeros (see [Figure 4-6](#)).

**FIGURE 4-6: READ DATA FROM PROGRAM MEMORY**

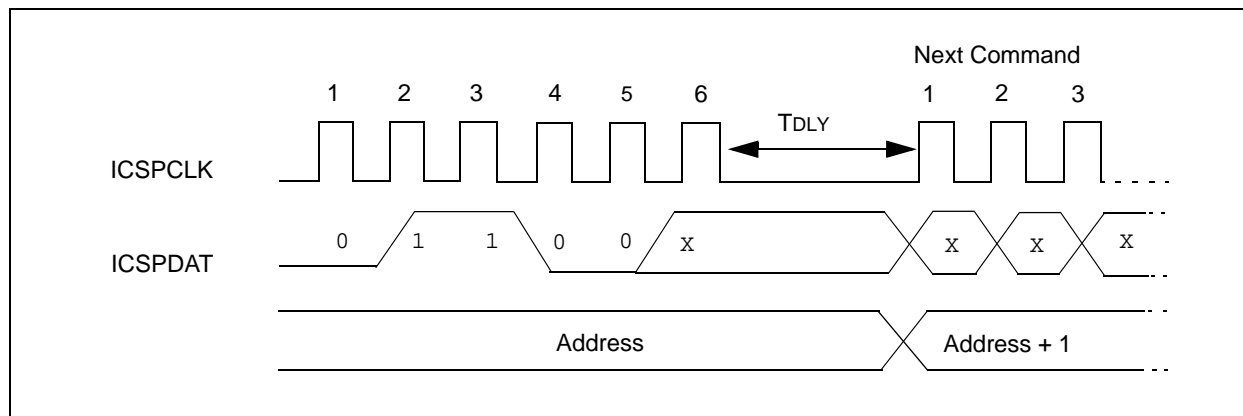


# PIC16(L)F145X

## 4.3.4 INCREMENT ADDRESS

The address is incremented when this command is received. It is not possible to decrement the address. To reset this counter, the user must use the Reset Address command or exit Program/Verify mode and re-enter it. If the address is incremented from address 7FFFh, it will wrap-around to location 0000h. If the address is incremented from FFFFh, it will wrap-around to location 8000h.

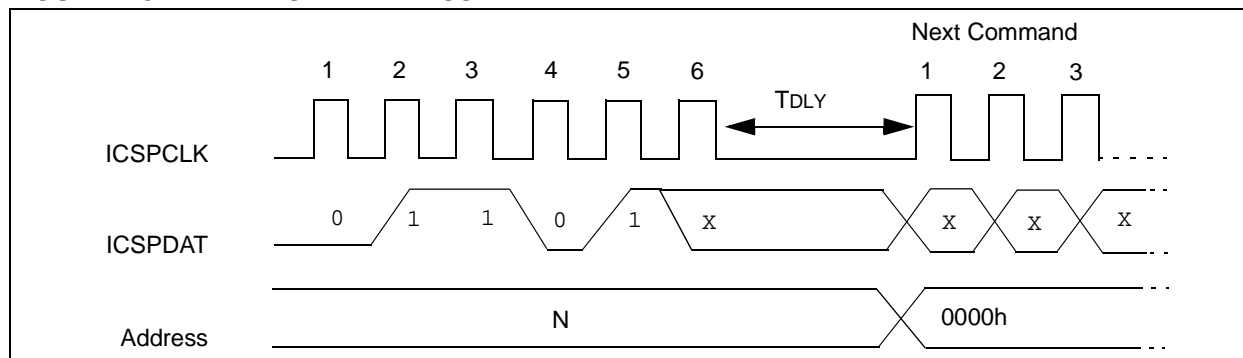
**FIGURE 4-7: INCREMENT ADDRESS**



## 4.3.5 RESET ADDRESS

The Reset Address command will reset the address to 0000h, regardless of the current value. The address is used in program memory or the configuration memory.

**FIGURE 4-8: RESET ADDRESS**





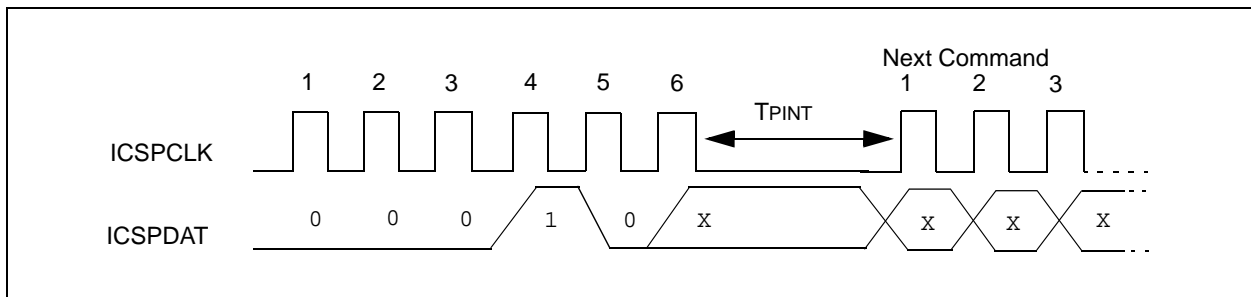
## 4.3.6 BEGIN INTERNALLY TIMED PROGRAMMING

A Load Configuration or Load Data for Program Memory command must be given before every Begin Programming command. Programming of the addressed memory will begin after this command is received. An internal timing mechanism executes the write. The user must allow for the program cycle time,  $T_{PINT}$ , for the programming to complete.

The End Externally Timed Programming command is not needed when the Begin Internally Timed Programming is used to start the programming.

The program memory address that is being programmed is not erased prior to being programmed.

**FIGURE 4-9: BEGIN INTERNALLY TIMED PROGRAMMING**

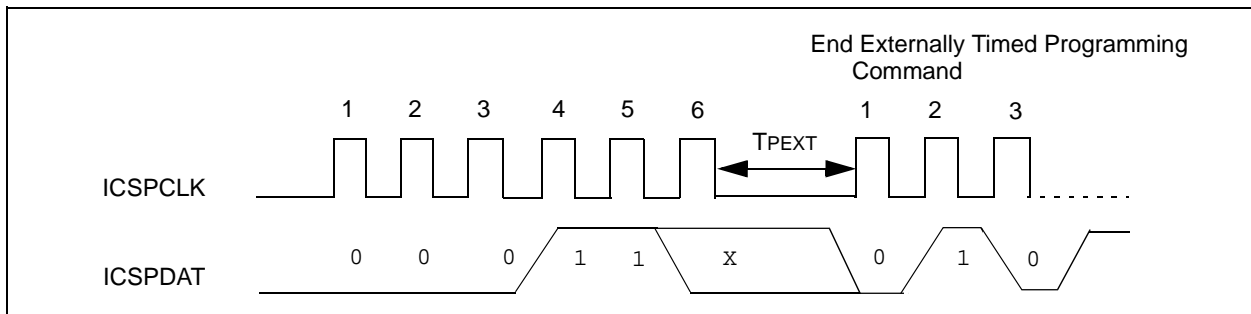


## 4.3.7 BEGIN EXTERNALLY TIMED PROGRAMMING

A Load Configuration or Load Data for Program Memory command must be given before every Begin Programming command. Programming of the addressed memory will begin after this command is received. To complete the programming the End Externally Timed Programming command must be sent in the specified time window defined by  $T_{PEXT}$  (see [Figure 4-10](#)).

Externally timed writes are not supported for Configuration and Calibration bits. Any externally timed write to the Configuration or Calibration Word will have no effect on the targeted word.

**FIGURE 4-10: BEGIN EXTERNALLY TIMED PROGRAMMING**



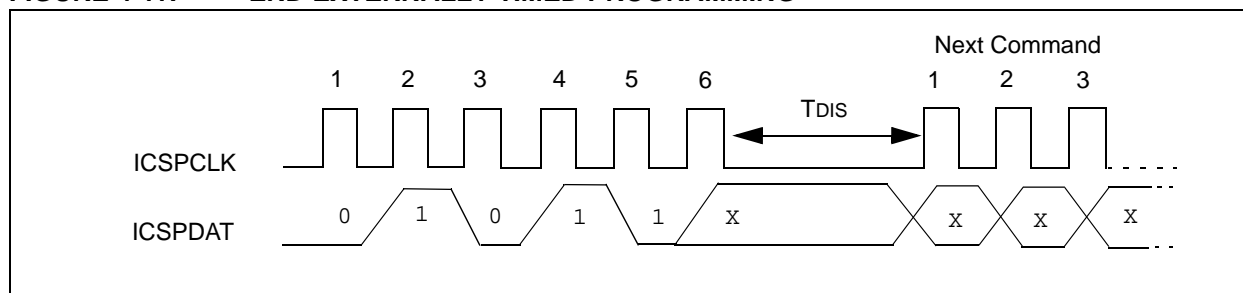
# PIC16(L)F145X

## 4.3.8 END EXTERNALLY TIMED PROGRAMMING

This command is required after a Begin Externally Timed Programming command is given. This command must be sent within the time window specified by  $T_{PEXT}$  after the Begin Externally Timed Programming command is sent.

After sending the End Externally Timed Programming command, an additional delay ( $T_{DIS}$ ) is required before sending the next command. This delay is longer than the delay ordinarily required between other commands (see Figure 4-11).

**FIGURE 4-11: END EXTERNALLY TIMED PROGRAMMING**



## 4.3.9 BULK ERASE PROGRAM MEMORY

The Bulk Erase Program Memory command performs two different functions dependent on the current state of the address.

Address 0000h-7FFFh:

- Program Memory is erased
- Configuration Words are erased

Address 8000h-8008h:

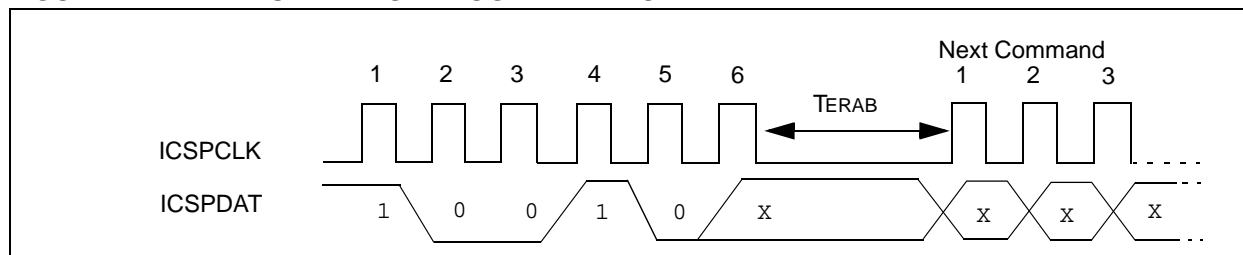
- Program Memory is erased
- Configuration Words are erased
- User ID Locations are erased

A Bulk Erase Program Memory command should not be issued when the address is greater than 8008h.

After receiving the Bulk Erase Program Memory command the erase will not complete until the time interval,  $T_{ERAB}$ , has expired.

**Note:** The code protection Configuration bit (CP) has no effect on the Bulk Erase Program Memory command.

**FIGURE 4-12: BULK ERASE PROGRAM MEMORY**



4.3.10 ROW ERASE PROGRAM MEMORY

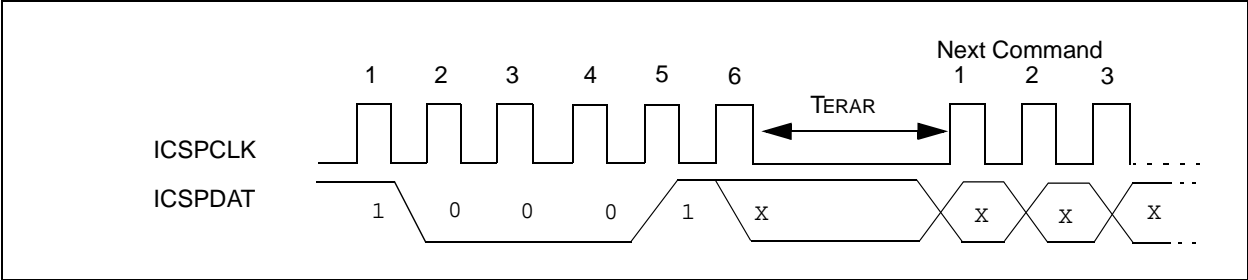
The Row Erase Program Memory command will erase an individual row. Refer to [Table 4-3](#) for row sizes of specific devices and the PC bits used to address them. If the program memory is code-protected, the Row Erase Program Memory command will be ignored. When the address is 8000h-8008h, the Row Erase Program Memory command will only erase the user ID locations, regardless of the setting of the  $\overline{CP}$  Configuration bit.

After receiving the Row Erase Program Memory command, the erase will not complete until the time interval, TERAR, has expired.

TABLE 4-3: PROGRAMMING ROW SIZE AND LATCHES

Devices	PC	Row Size	Number of Latches
PIC16(L)F1454	<15:5>	32	32
PIC16(L)F1455	<15:5>	32	32
PIC16(L)F1459	<15:5>	32	32

FIGURE 4-13: ROW ERASE PROGRAM MEMORY



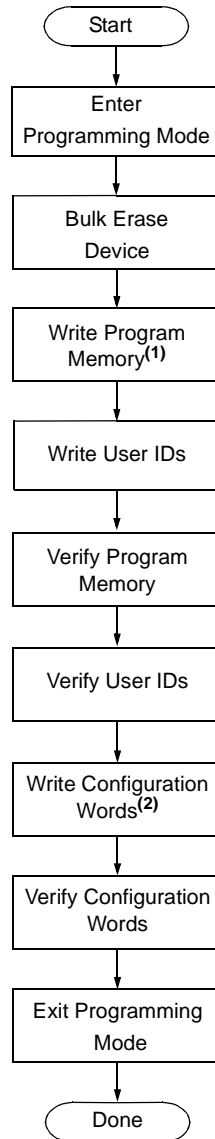
## 5.0 PROGRAMMING ALGORITHMS

The devices use internal latches to temporarily store the 14-bit words used for programming. Refer to [Table 4-3](#) for specific latch information. The data latches allow the user to write the program words with a single Begin Externally Timed Programming or Begin Internally Timed Programming command. The Load Program Data or the Load Configuration command is used to load a single data latch. The data latch will hold the data until the Begin Externally Timed Programming or Begin Internally Timed Programming command is given.

The data latches are aligned with the LSbs of the address. The PC's address at the time the Begin Externally Timed Programming or Begin Internally Timed Programming command is given will determine which location(s) in memory are written. Writes cannot cross the physical boundary. For example, with the PIC16F1455, attempting to write from address 0002h-0009h will result in data being written to 0008h-000Fh.

If more than the maximum number of data latches are written without a Begin Externally Timed Programming or Begin Internally Timed Programming command, the data in the data latches will be overwritten. The following figures show the recommended flowcharts for programming.

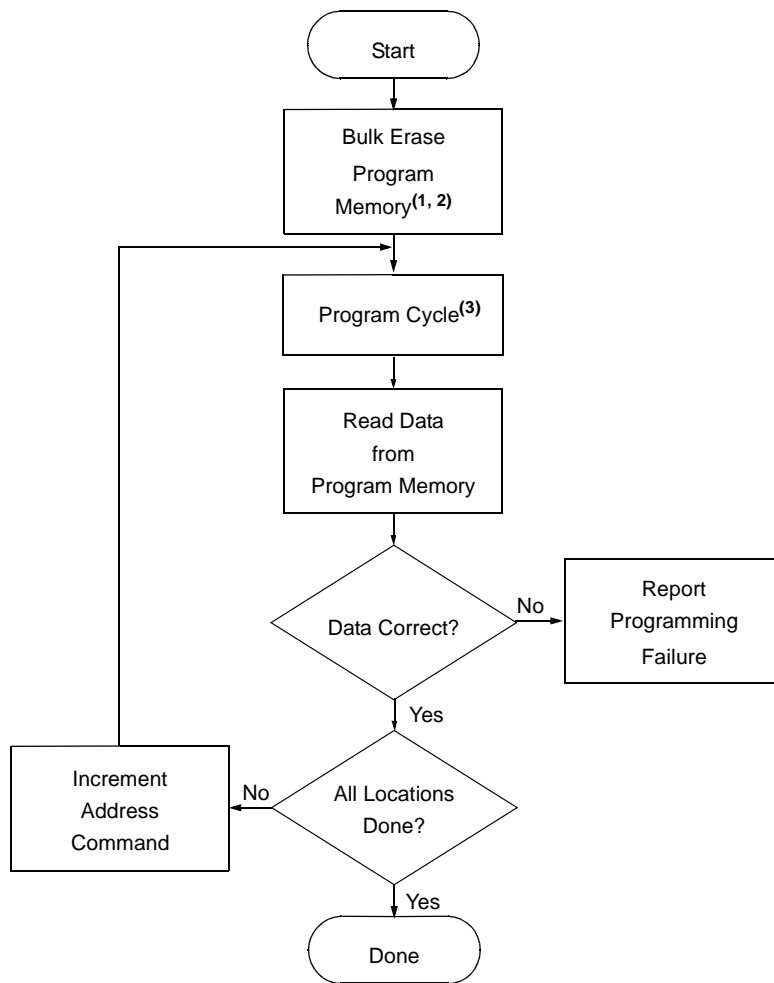
**FIGURE 5-1: DEVICE PROGRAM/VERIFY FLOWCHART**



**Note 1:** See [Figure 5-2](#).

**2:** See [Figure 5-5](#).

FIGURE 5-2: PROGRAM MEMORY FLOWCHART



- Note 1:** This step is optional if the device has already been erased or has not been previously programmed.
- Note 2:** If the device is code-protected or must be completely erased, then Bulk Erase the device per [Figure 5-6](#).
- Note 3:** See [Figure 5-3](#) or [Figure 5-4](#).

**FIGURE 5-3: ONE-WORD PROGRAM CYCLE**

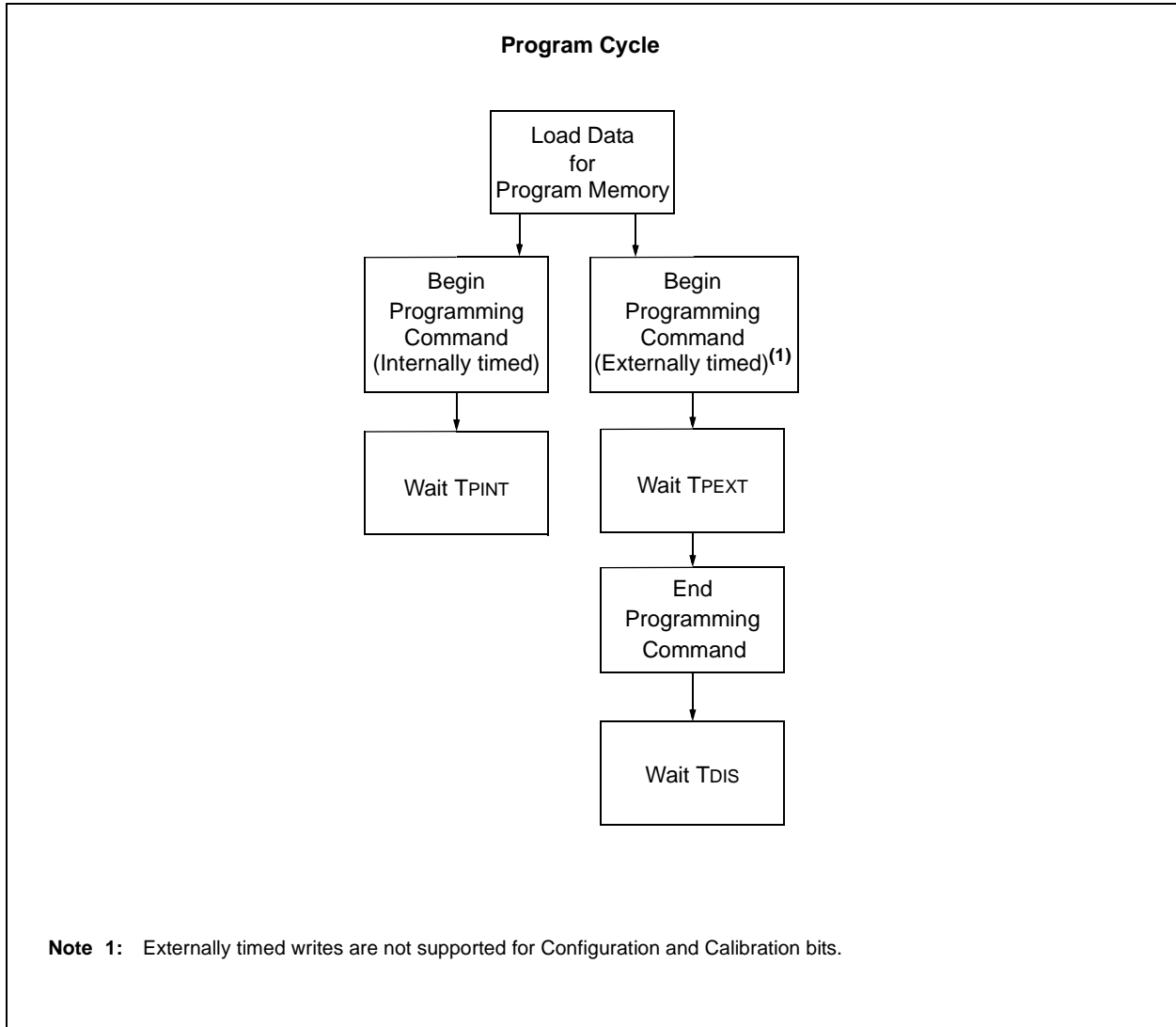
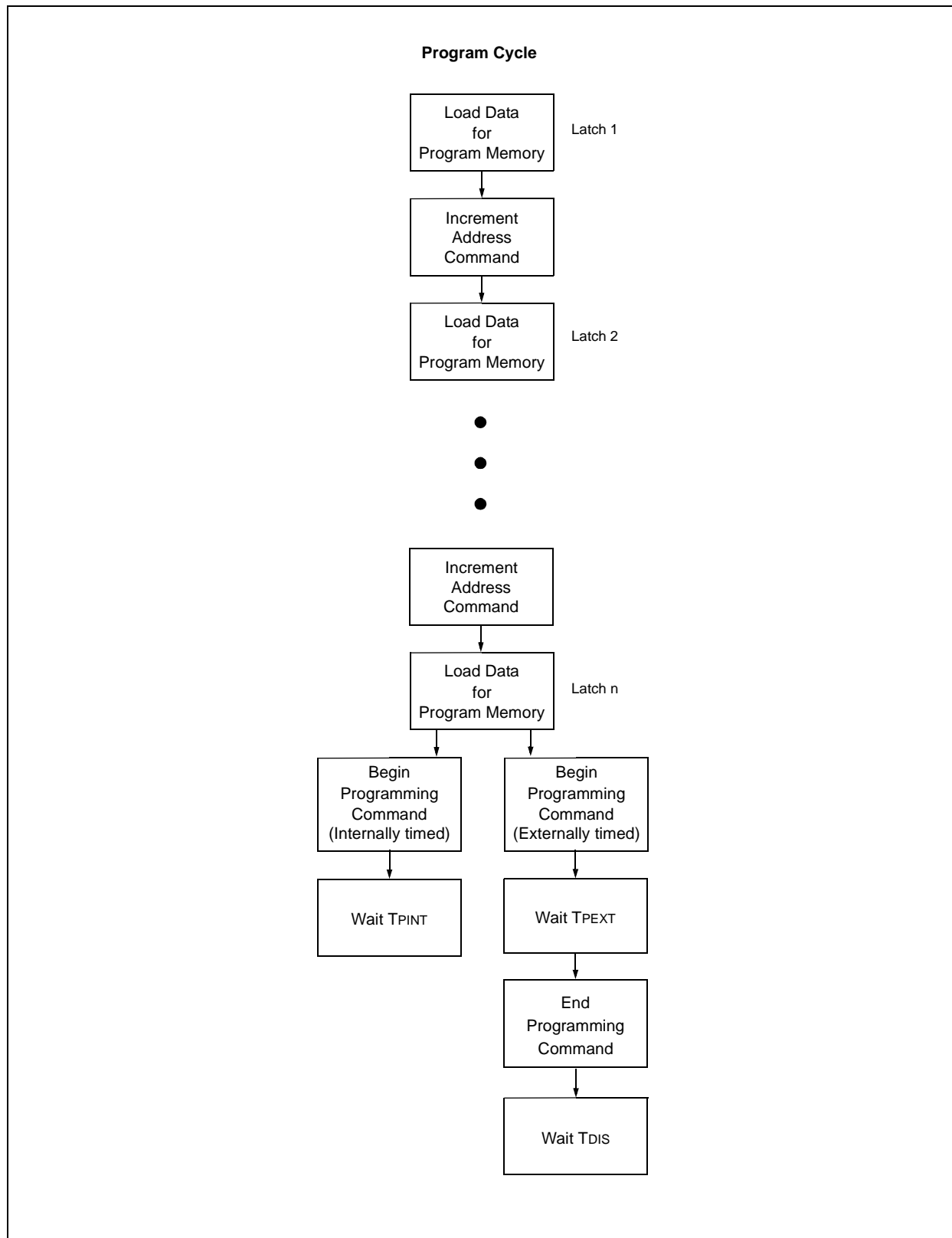
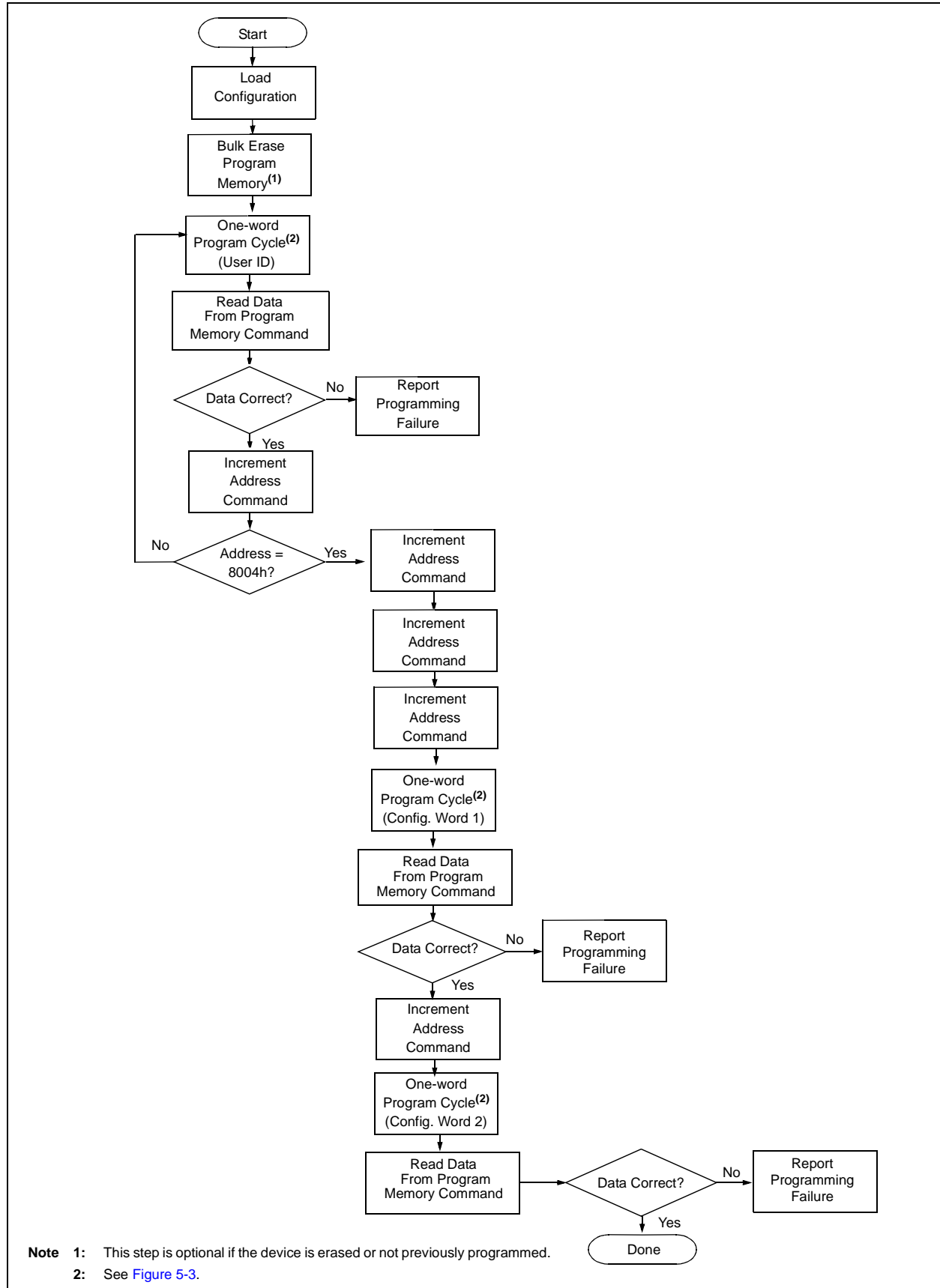


FIGURE 5-4: MULTIPLE-WORD PROGRAM CYCLE

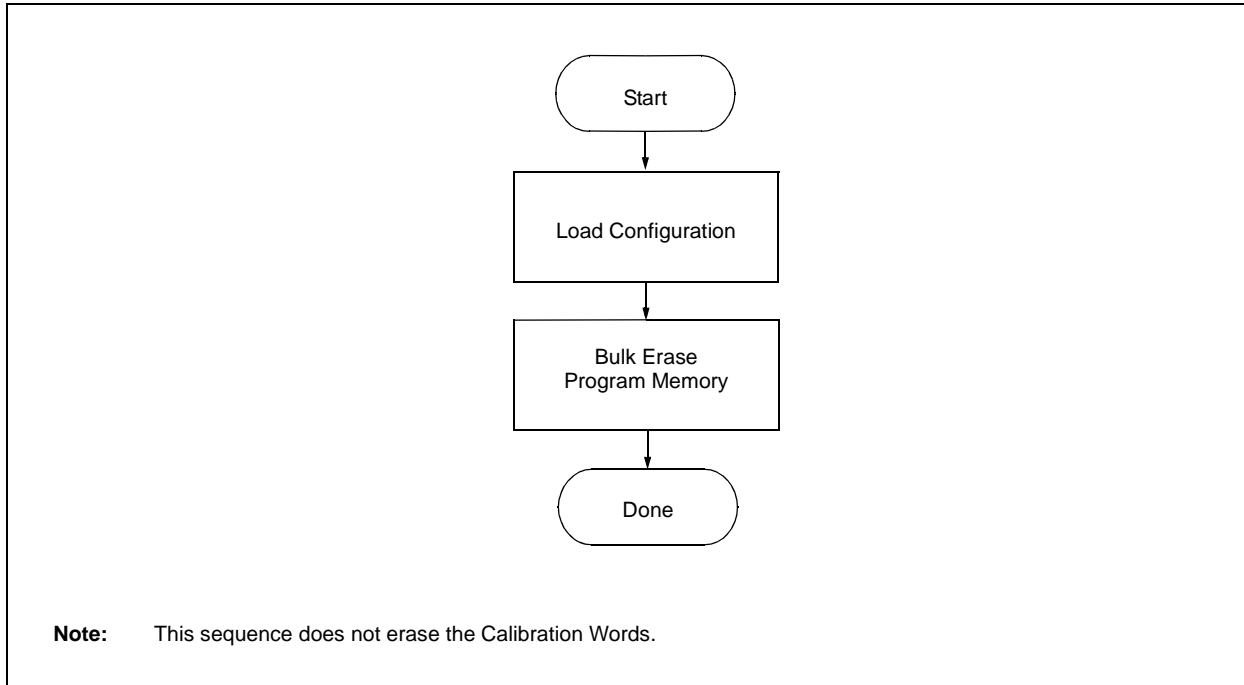




**FIGURE 5-5: CONFIGURATION MEMORY PROGRAM FLOWCHART**



**FIGURE 5-6: ERASE FLOWCHART**



## 6.0 CODE PROTECTION

Code protection is controlled using the  $\overline{\text{CP}}$  bit in Configuration Word 1. When code protection is enabled, all program memory locations (0000h-7FFFh) read as '0'. Further programming is disabled for the program memory (0000h-7FFFh).

The user ID locations and Configuration Words can be programmed and read out regardless of the code protection settings.

### 6.1 Program Memory

Code protection is enabled by programming the  $\overline{\text{CP}}$  bit in Configuration Word 1 register to '0'.

The only way to disable code protection is to use the Bulk Erase Program Memory command.

## 7.0 HEX FILE USAGE

In the hex file there are two bytes per program word stored in the Intel® INHX32 hex format. Data is stored LSB first, MSB second. Because there are two bytes per word, the addresses in the hex file are 2x the address in program memory. (Example: Configuration Word 1 is stored at 8007h on the PIC16(L)F1458. In the hex file this will be referenced as 1000Eh-1000Fh).

### 7.1 Configuration Word

To allow portability of code, it is strongly recommended that the programmer is able to read the Configuration Words and user ID locations from the hex file. If the Configuration Words information was not present in the hex file, a simple warning message may be issued. Similarly, while saving a hex file, Configuration Words and user ID information should be included.

### 7.2 Device ID

If a device ID is present in the hex file at 1000Ch-1000Dh (8006h on the part), the programmer should verify the device ID against the value read from the part. On a mismatch condition the programmer should generate a warning message.

# PIC16(L)F145X

## 7.3 Checksum Computation

The checksum is calculated by two different methods dependent on the setting of the  $\overline{CP}$  Configuration bit.

**TABLE 7-1: CONFIGURATION WORD MASK VALUES**

Device	Config. Word 1 Mask	Config. Word 2 Mask
PIC16(L)F1454	3EFFh	3FF3h
PIC16(L)F1455	3EFFh	3FF3h
PIC16(L)F1459	3EFFh	3FF3h

### 7.3.1 PROGRAM CODE PROTECTION DISABLED

With the program code protection disabled, the checksum is computed by reading the contents of the program memory locations and adding up the program memory data starting at address 0000h, up to the maximum user addressable location. Any Carry bit exceeding 16 bits are ignored. Additionally, the relevant bits of the Configuration Words are added to the checksum. All unimplemented Configuration bits are masked to '0'.

**EXAMPLE 7-1: CHECKSUM COMPUTED WITH PROGRAM CODE PROTECTION DISABLED (CP = 1), PIC16F1459, BLANK DEVICE**

PIC16F1459	Sum of Memory addresses 0000h-1FFFh	E000h <sup>(1)</sup>
	Configuration Word 1	3FFFh <sup>(2)</sup>
	Configuration Word 1 mask	3EFFh <sup>(3)</sup>
	Configuration Word 2	3FFFh <sup>(4)</sup>
	Configuration Word 2 mask	3FF3h <sup>(5)</sup>
	Checksum	= E000h + (3FFFh and 3EFFh) + (3FFFh and 3FF3h) <sup>(6)</sup> = E000h + 3EFFh + 3FF3h = 5EF2h

- Note 1:** This value is obtained by taking the total number of program memory locations (0x000 to 0x1FFFh which is E000h) and multiplying it by the blank memory value of 0x3FFF to get the sum of 1FF F800h. Then, truncate to 16 bits, thus having a final value of F800h.
- 2:** This value is obtained by making all bits of the Configuration Word 1 a '1', then converting it to hex, thus having a value of 3FFFh.
- 3:** This value is obtained by making all used bits of the Configuration Word 1 a '1', then converting it to hex, thus having a value of 3EFFh.
- 4:** This value is obtained by making all bits of the Configuration Word 2 a '1', then converting it to hex, thus having a value of 3FFFh.
- 5:** This value is obtained by making all used bits of the Configuration Word 2 a '1', then converting it to hex, thus having a value of 3FF3h.
- 6:** This value is obtained by ANDing the Configuration Word value with the Configuration Word Mask Value and adding it to the sum of memory addresses: (3FFFh and 3EFFh) + (3FFFh and 3FF3h) + E000h = 1 34FEh. Then, truncate to 16 bits, thus having a final value of 5EF2h.

**EXAMPLE 7-2: CHECKSUM COMPUTED WITH PROGRAM CODE PROTECTION DISABLED (CP = 1), PIC16LF1459, 00AAh AT FIRST AND LAST ADDRESS**

PIC16LF1459	Sum of Memory addresses 0000h-1FFFh	6156h <sup>(1)</sup>
	Configuration Word 1	3FFFh <sup>(2)</sup>
	Configuration Word 1 mask	3EFFh <sup>(3)</sup>
	Configuration Word 2	3FFFh <sup>(4)</sup>
	Configuration Word 2 mask	3FF3h <sup>(5)</sup>
	Checksum = 6156h + (3FFFh and 3EFFh) + (3FFFh and 3FF3h) <sup>(6)</sup>	
		= 6156h + 3EFFh + 3FF3h
		= E048h

- Note 1:** This value is obtained by taking the total number of program memory locations (0x000 to 0x1FFFh which is 2000h) subtracting 2h which yields 1EFFh, then multiplying it by the blank memory value of 0x3FFF to get the sum of 7FF 6002h. Then, truncate to 16 bits the value of 6002h. Now add 00AAh (00AAh + 00AAh) to 6002h to get the final value of 6156h.
- 2:** This value is obtained by making all bits of the Configuration Word 1 a '1', then converting it to hex, thus having a value of 3FFFh.
- 3:** This value is obtained by making all used bits of the Configuration Word 1 a '1', then converting it to hex, thus having a value of 3EFFh.
- 4:** This value is obtained by making all bits of the Configuration Word 2 a '1', then converting it to hex, thus having a value of 3FFFh.
- 5:** This value is obtained by making all used bits of the Configuration Word 2 a '1', then converting it to hex, thus having a value of 3FF3h.
- 6:** This value is obtained by ANDing the Configuration Word value with the Configuration Word Mask Value and adding it to the sum of memory addresses: (3FFFh and 3EFFh) + (3FFFh and 3FF3h) + 6156h = E048h. Then, truncate to 16 bits, thus having a final value of E048h.

# PIC16(L)F145X

## 7.3.2 PROGRAM CODE PROTECTION ENABLED

With the program code protection enabled, the checksum is computed in the following manner: The Least Significant nibble of each user ID is used to create a 16-bit value. The masked value of user ID location 8000h is the Most Significant nibble. This sum of user IDs is summed with the Configuration Words (all unimplemented Configuration bits are masked to '0').

### EXAMPLE 7-3: CHECKSUM COMPUTED WITH PROGRAM CODE PROTECTION ENABLED (CP = 0), PIC16F1459, BLANK DEVICE

PIC16F1459	Configuration Word 1	3F7Fh <sup>(1)</sup>
	Configuration Word 1 mask	3E7Fh <sup>(2)</sup>
	Configuration Word 2	3FFFh <sup>(3)</sup>
	Configuration Word 2 mask	3FF3h <sup>(4)</sup>
	User ID (8000h)	0006h <sup>(5)</sup>
	User ID (8001h)	0007h <sup>(5)</sup>
	User ID (8002h)	0001h <sup>(5)</sup>
	User ID (8003h)	0002h <sup>(5)</sup>
	Sum of User IDs = (0006h and 000Fh) << 12 + (0007h and 000Fh) << 8 + (0001h and 000Fh) << 4 + (0002h and 000Fh) <sup>(6)</sup>	
		= 6000h + 0700h + 0010h + 0002h
		= 6712h
	Checksum = (3F7Fh and 3E7Fh) + (3FFFh and 3FF3h) + Sum of User IDs <sup>(7)</sup>	
		= 3E7Fh + 3FF3h + 6712h
		= E584h

- Note 1:** This value is obtained by making all bits of the Configuration Word 1 a '1', but the code-protect bit is '0' (thus, enabled), then converting it to hex, thus having a value of 3F7Fh.
- 2:** This value is obtained by making all used bits of the Configuration Word 1 a '1', but the code-protect bit is '0' (thus, enabled), then converting it to hex, thus having a value of 3E7Fh.
- 3:** This value is obtained by making all bits of the Configuration Word 2 a '1', then converting it to hex, thus having a value of 3FFFh.
- 4:** This value is obtained by making all used bits of the Configuration Word 2 a '1', then converting it to hex, thus having a value of 3FF3h.
- 5:** These values are picked at random for this example; they could be any 16-bit value.
- 6:** In order to calculate the sum of user IDs, take the 16-bit value of the first user ID location (0006h), AND the address to (000Fh), thus masking the MSB. This gives you the value 0006h, then shift left 12 bits, giving you 6000h. Do the same procedure for the 16-bit value of the second user ID location (0007h), except shift left 8 bits. Also, do the same for the third user ID location (0001h), except shift left 4 bits. For the fourth user ID location do not shift. Finally, add up all four user ID values to get the final sum of user IDs of 6712h.
- 7:** This value is obtained by ANDing the Configuration Word value with the Configuration Word Mask Value and adding it to the sum of user IDs: (3F7Fh AND 3E7Fh) + (3FFFh AND 3FF3h) + 6712h = E584h. Then, truncate to 16 bits, thus having a final value of E584h.

**EXAMPLE 7-4: CHECKSUM COMPUTED WITH PROGRAM CODE PROTECTION ENABLED (CP = 0), PIC16LF1459, 00AAh AT FIRST AND LAST ADDRESS**

PIC16LF1459	Configuration Word 1	3F7Fh <sup>(1)</sup>
	Configuration Word 1 mask	3E7Fh
	Configuration Word 2	3FFFh <sup>(3)</sup>
	Configuration Word 2 mask	3FF3h <sup>(4)</sup>
	User ID (8000h)	000Eh <sup>(5)</sup>
	User ID (8001h)	0008h <sup>(5)</sup>
	User ID (8002h)	0005h <sup>(5)</sup>
	User ID (8003h)	0008h <sup>(5)</sup>
	Sum of User IDs	$= (000Eh \text{ and } 000Fh) \ll 12 + (0008h \text{ and } 000Fh) \ll 8 +$ $(0005h \text{ and } 000Fh) \ll 4 + (0008h \text{ and } 000Fh)^{(6)}$ $= E000h + 0800h + 0050h + 0008h$ $= E858h$
	Checksum	$= (3F7Fh \text{ and } 3E7Fh) + (3FFFh \text{ and } 3FF3h) + \text{Sum of User IDs}^{(7)}$ $= 3E7Fh + 3FF3h + E858h$ $= 66C9h$

**Note 1:** This value is obtained by making all bits of the Configuration Word 1 a '1', but the code-protect bit is '0' (thus, enabled), then converting it to hex, thus having a value of 3F7Fh.

**2:** This value is obtained by making all used bits of the Configuration Word 1 a '1', but the code-protect bit is '0' (thus, enabled), then converting it to hex, thus having a value of 3E7Fh.

**3:** This value is obtained by making all bits of the Configuration Word 2 a '1', then converting it to hex, thus having a value of 3FFFh.

**4:** This value is obtained by making all used bits of the Configuration Word 2 a '1', then converting it to hex, thus having a value of 3FF3h.

**5:** These values are picked at random for this example; they could be any 16-bit value.

**6:** In order to calculate the sum of user IDs, take the 16-bit value of the first user ID location (000Eh), AND the address to (000Fh), thus masking the MSB. This gives you the value 000Eh, then shift left 12 bits, giving you E000h. Do the same procedure for the 16-bit value of the second user ID location (0008h), except shift left 8 bits. Also, do the same for the third user ID location (0005h), except shift left 4 bits. For the fourth user ID location do not shift. Finally, add up all four user ID values to get the final sum of user IDs of E858h.

**7:** This value is obtained by ANDing the Configuration Word value with the Configuration Word Mask Value and adding it to the sum of user IDs: (3F7Fh AND 3E7Fh) + (3FFFh AND 3FF3h) + E858h = 66CAh. Then, truncate to 16 bits, thus having a final value of 66CAh.

# PIC16(L)F145X

## 8.0 ELECTRICAL SPECIFICATIONS

Refer to the device specific data sheet for absolute maximum ratings.

**TABLE 8-1: AC/DC CHARACTERISTICS TIMING REQUIREMENTS FOR PROGRAM/VERIFY MODE**

AC/DC CHARACTERISTICS			Standard Operating Conditions Production tested at 25°C				
Sym.	Characteristics		Min.	Typ.	Max.	Units	Conditions/Comments
Programming Supply Voltages and Currents							
VDD	Supply Voltage (VDDMIN, VDDMAX)	PIC16LF145X	1.80	—	3.60	V	Fosc <= 16 MHz
			2.70	—	3.60	V	Fosc <= 48 MHz
		PIC16F145X	2.30	—	5.50	V	Fosc <= 16 MHz
			2.70	—	5.50	V	Fosc <= 48 MHz
VPEW	Read/Write and Row Erase operations		VDDMIN	—	VDDMAX	V	
VBE	Bulk Erase operations		2.7	—	VDD max.	V	
IDDI	Current on VDD, Idle		—	—	1.0	mA	
IDDP	Current on VDD, Programming		—	—	3.0	mA	
IPP	VPP						
	Current on MCLR/VPP		—	—	600	μA	
VIHH	High voltage on MCLR/VPP for Program/Verify mode entry		8.0	—	9.0	V	
TVHHR	MCLR rise time (VIL to VIHH) for Program/Verify mode entry		—	—	1.0	μs	
	I/O pins						
VIH	(ICSPCLK, ICSPDAT, MCLR/VPP) input high level		0.8 VDD	—	—	V	
VIL	(ICSPCLK, ICSPDAT, MCLR/VPP) input low level		—	—	0.2 VDD	V	
VOH	ICSPDAT output high level		VDD-0.7 VDD-0.7 VDD-0.7	—	—	V	IOH = 3.5 mA, VDD = 5V IOH = 3 mA, VDD = 3.3V IOH = 2 mA, VDD = 1.8V
VOL	ICSPDAT output low level		—	—	VSS+0.6 VSS+0.6 VSS+0.6	V	IOH = 8 mA, VDD = 5V IOH = 6 mA, VDD = 3.3V IOH = 3 mA, VDD = 1.8V
VBOR	Brown-out Reset Voltage						
	BORV = 0 (high trip)		2.55 2.30	2.70 2.40	2.85 2.55	V V	PIC16F145X PIC16LF145X
	BORV = 1 (low trip)		1.80	1.90	2.05	V	PIC16(L)F145X
Programming Mode Entry and Exit							
TENTS	Programing mode entry setup time: ICSPCLK, ICSPDAT setup time before VDD or MCLR↑		100	—	—	ns	
TENTH	Programing mode entry hold time: ICSPCLK, ICSPDAT hold time after VDD or MCLR↑		250	—	—	μs	
Serial Program/Verify							
TCKL	Clock Low Pulse Width		100	—	—	ns	
TCKH	Clock High Pulse Width		100	—	—	ns	
TDS	Data in setup time before clock↓		100	—	—	ns	
TDH	Data in hold time after clock↓		100	—	—	ns	
Tco	Clock↑ to data out valid (during a Read Data command)		0	—	80	ns	
TLZD	Clock↓ to data low-impedance (during a Read Data command)		0	—	80	ns	
THZD	Clock↓ to data high-impedance (during a Read Data command)		0	—	80	ns	
TDLY	Data input not driven to next clock input (delay required between command/data or command/command)		1.0	—	—	μs	

**Note 1:** Externally timed writes are not supported for Configuration and Calibration bits.



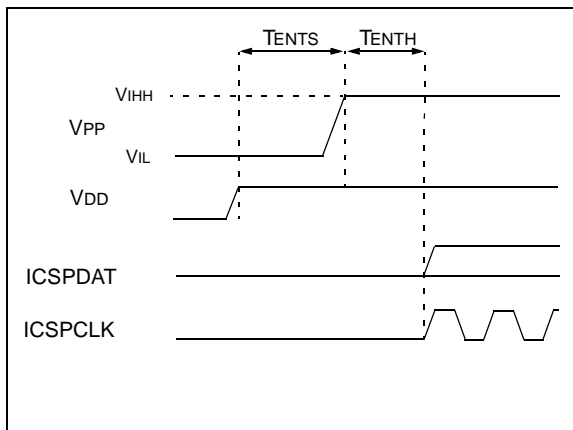
**TABLE 8-1: AC/DC CHARACTERISTICS TIMING REQUIREMENTS FOR PROGRAM/VERIFY**

AC/DC CHARACTERISTICS		Standard Operating Conditions Production tested at 25°C				
Sym.	Characteristics	Min.	Typ.	Max.	Units	Conditions/Comments
TERAB	Bulk Erase cycle time	—	—	5	ms	
TERAR	Row Erase cycle time	—	—	2.5	ms	
TPINT	Internally timed programming operation time	—	—	2.5 5	ms ms	Program memory Configuration Words
TPEXT	Externally timed programming pulse	1.0	—	2.1	ms	<b>Note 1</b>
TDIS	Time delay from program to compare (HV discharge time)	300	—	—	μs	
TEXT	Time delay when exiting Program/Verify mode	1	—	—	μs	

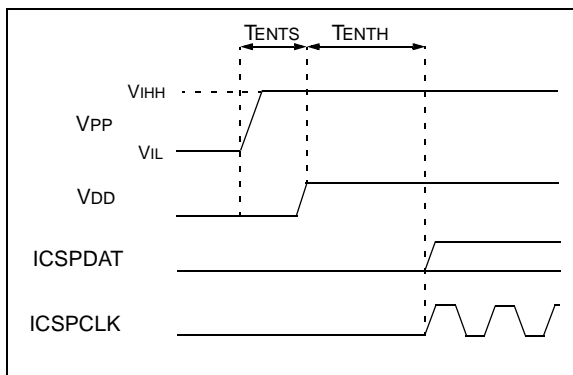
**Note 1:** Externally timed writes are not supported for Configuration and Calibration bits.

## 8.1 AC Timing Diagrams

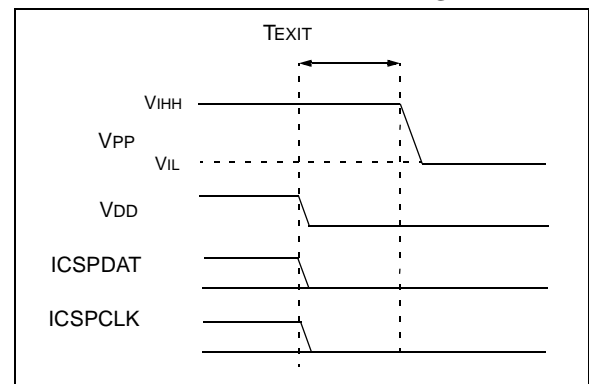
**FIGURE 8-1: PROGRAMMING MODE ENTRY – VDD FIRST**



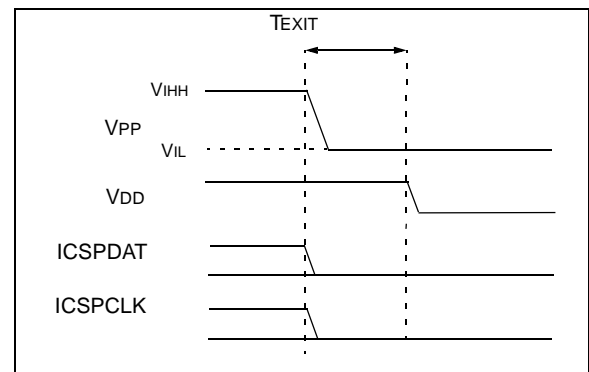
**FIGURE 8-2: PROGRAMMING MODE ENTRY – VPP FIRST**



**FIGURE 8-3: PROGRAMMING MODE EXIT – VPP LAST**

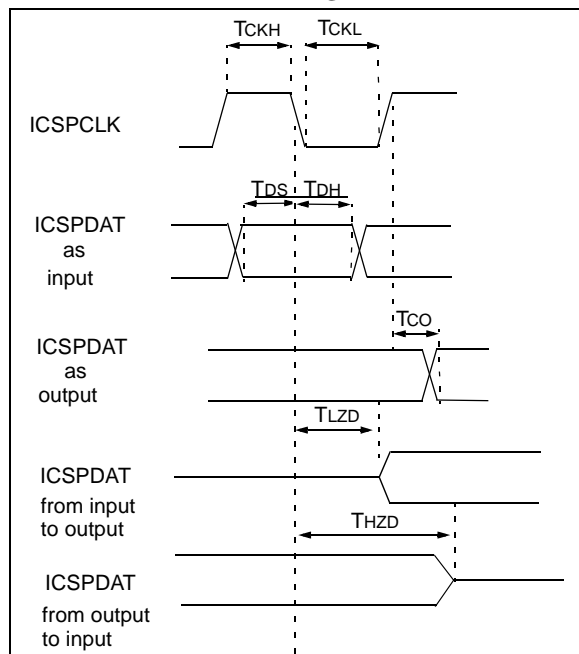


**FIGURE 8-4: PROGRAMMING MODE EXIT – VDD LAST**

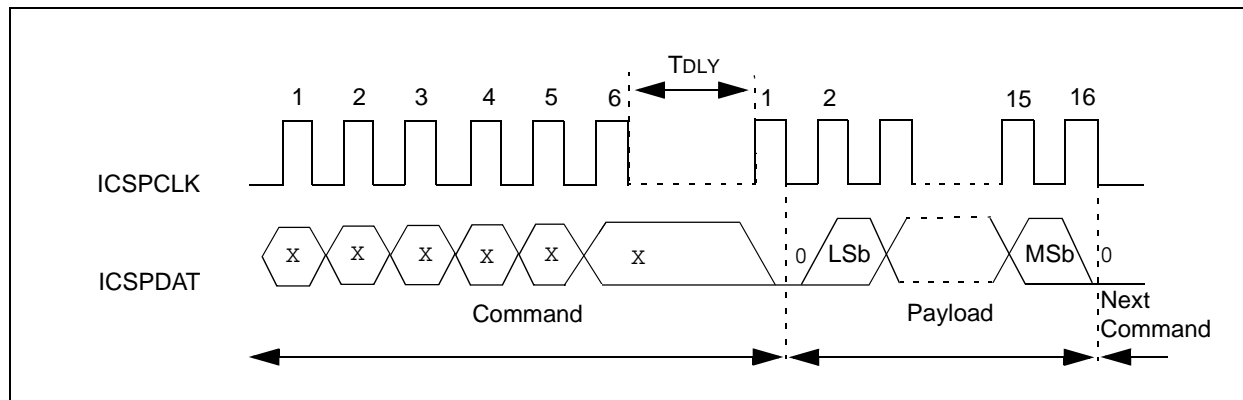


# PIC16(L)F145X

**FIGURE 8-5: CLOCK AND DATA TIMING**



**FIGURE 8-6: WRITE COMMAND-PAYLOAD TIMING**



**FIGURE 8-7: READ COMMAND-PAYLOAD TIMING**

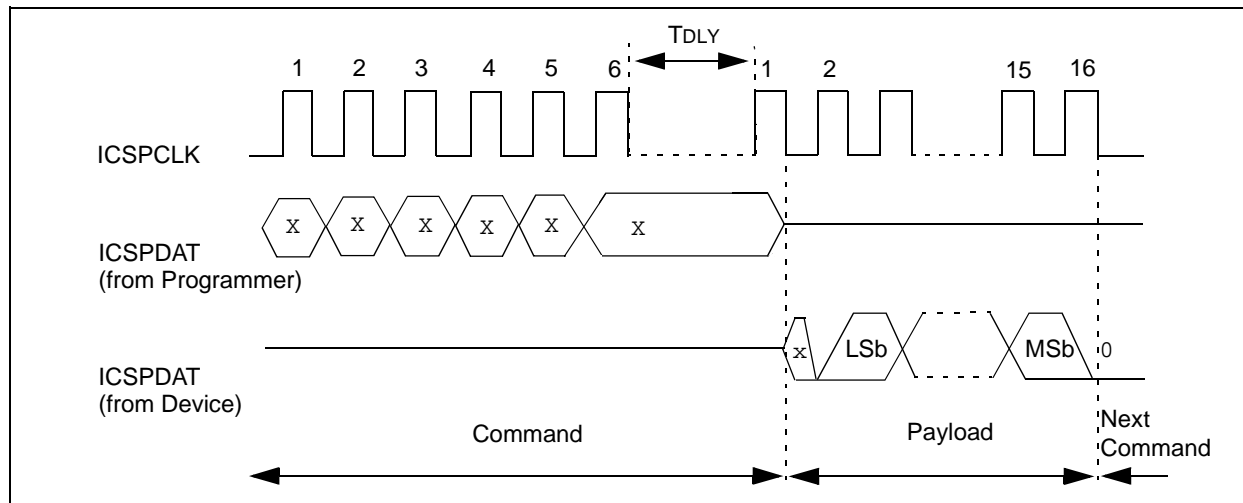


FIGURE 8-8: LVP ENTRY (POWERED)

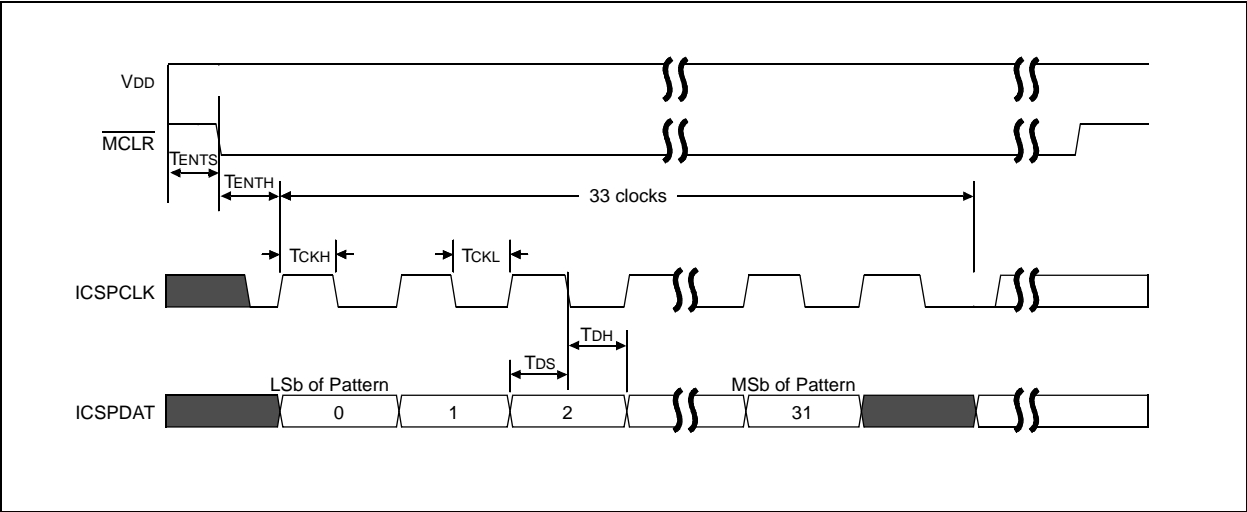
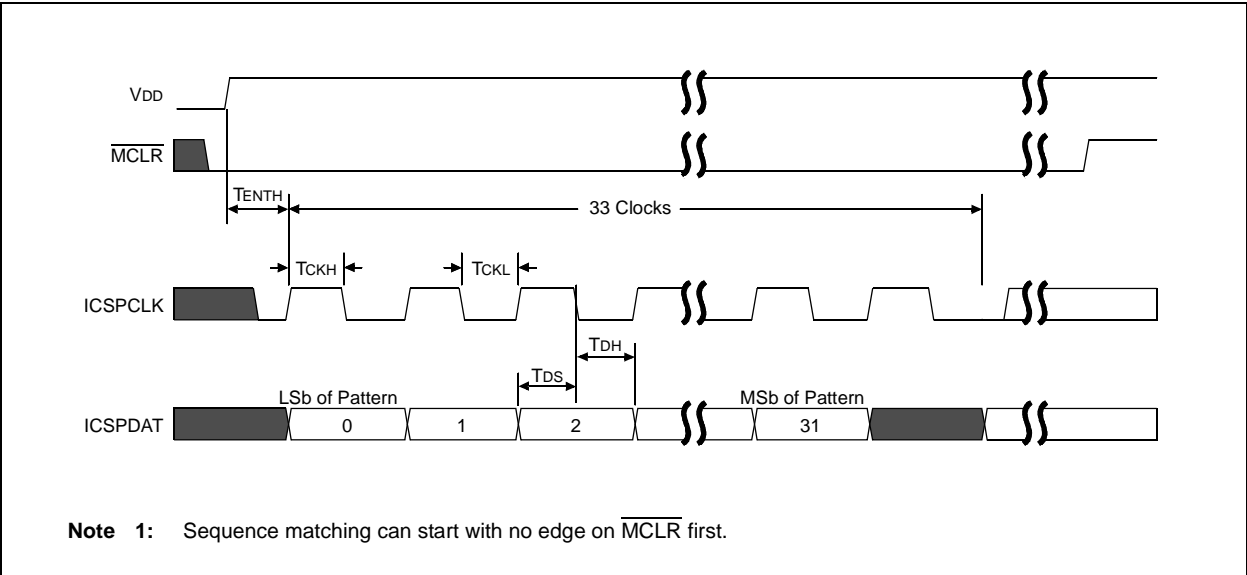


FIGURE 8-9: LVP ENTRY (POWERING UP)



# PIC16(L)F145X

---

## APPENDIX A: REVISION HISTORY

### Revision A (12/2011)

Original release of this document.

### Revision B (04/2012)

Added PIC16(L)F1454 devices; Removed PIC16(L)F1458 devices; Removed Figure 3-1; Updated Table 3-1 and Register 3-4; Updated Figures 4-1, 4-2 and 4-3; Updated Table 4-3 and Table 7-1; Other minor corrections.

---

**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, rfPIC 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, chipKIT, chipKIT logo, 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, Omniscent Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICKit, PICtail, REAL ICE, rfLAB, 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.

© 2011-2012, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

 Printed on recycled paper.

ISBN: 9781620762431

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

*Microchip received ISO/TS-16949:2009 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.*



## 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://www.microchip.com/support>  
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

**Indianapolis**  
Noblesville, IN  
Tel: 317-773-8323  
Fax: 317-773-5453

**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: 86-10-8569-7000  
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 - Hangzhou**  
Tel: 86-571-2819-3187  
Fax: 86-571-2819-3189

**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 - Osaka**  
Tel: 81-66-152-7160  
Fax: 81-66-152-9310

**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: 886-3-5778-366  
Fax: 886-3-5770-955

**Taiwan - Kaohsiung**  
Tel: 886-7-536-4818  
Fax: 886-7-330-9305

**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

11/29/11