Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

PICkit™ 2 Programmer/Debugger User's Guide

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.

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

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.

Table of Contents

Preface

NOTICE TO CUSTOMERS

All documentation becomes dated, and this manual is no exception. Microchip tools and documentation are constantly evolving to meet customer needs, so some actual dialogs and/or tool descriptions may differ from those in this document. Please refer to our web site (www.microchip.com) to obtain the latest documentation available.

Documents are identified with a "DS" number. This number is located on the bottom of each page, in front of the page number. The numbering convention for the DS number is "DSXXXXXA", where "XXXXX" is the document number and "A" is the revision level of the document.

For the most up-to-date information on development tools, see the MPLAB® IDE on-line help. Select the Help menu, and then Topics to open a list of available on-line help files.

INTRODUCTION

This chapter contains general information that will be useful to know before using PICkit™ 2 . Items discussed include:

DOCUMENT LAYOUT

This document describes how to use the PICkit 2 as a development tool to emulate and debug firmware on a target board. The manual layout is as follows:

- **Chapter 1. PICkit 2 Programmer/Debugger Overview** – Provides an overview of the PICkit 2 Programmer/Debugger.
- **Chapter 2. PICkit 2 Programmer Getting Started** – Provides Instructions on how to get started using the PICkit 2 Programmer/Debugger to program Flash-based PIC® Microcontroller Units (MCUs).
- **Chapter 3. Using In-Circuit Serial Programming™ (ICSP™)** – Describes programming with the PICkit 2 Programmer/Debugger using In-Circuit Serial Programming™ (ICSP™).
- **Chapter 4. PICkit 2 Debug Express** – Provides a tutorial on using the PICkit™ 2 Debug Express debugger program.
- **Chapter 5. Troubleshooting** – Provides information on solving common problems and Debug Express errors.
- **Chapter 6. Updating the PICkit 2 Operating System** – Provides instruction on how to update your PICkit 2 Programmer/Debugger's Operating System.
- **Chapter 7. PICkit 2 UART Tool** – Provides information on using the PICkit 2 as a serial UART terminal interface for communicating with a PIC microcontroller
- **Appendix A. MPLAB IDE Reference** – Describes how the PICkit 2 Programmer/Debugger works with MPLAB IDE.
- **Appendix B. PICkit 2 Schematics** – Illustrates the PICkit 2 Programmer/Debugger hardware schematic diagrams.

CONVENTIONS USED IN THIS GUIDE

The following conventions may appear in this documentation:

DOCUMENTATION CONVENTIONS

Description	Represents	Examples
Arial font:		
Italic	Referenced books	*MPLAB® IDE User's Guide*
	Emphasized text	...is the *only* compiler...
Initial caps	A window	the Output window
	A dialog	the Settings dialog
	A menu selection	select Enable Programmer
Quotes	A field name in a window or dialog	"Save project before build"
Underlined, italic with right angle bracket	A menu path	*File>Save*
Bold characters	A dialog button	Click **OK**
	A tab	Click the **Power** tab
Text in angle brackets < >	A key on the keyboard	Press <Enter>, <F1>
Courier New font:		
Plain	Sample source code	`#define START`
	Filenames	`autoexec.bat`
	File paths	`c:\mcc18\h`
	Keywords	`_asm, _endasm, static`
	Command-line options	`-Opa+, -Opa-`
	Bit values	`0, 1`
	Constants	`0xFF,'A'`
Italic	A variable argument	*`file`*`.o`, where *`file`* can be any valid filename
Square brackets [ ]	Optional arguments	`mpasmwin [options] file [options]`
Curly brackets and pipe character: { | }	Choice of mutually exclusive arguments; an OR selection	`errorlevel {0|1}`
Ellipses...	Replaces repeated text	`var_name [, var_name...]`
	Represents code supplied by user	`void main (void) { ... }`

WARRANTY REGISTRATION

Please complete the enclosed Warranty Registration Card and mail it promptly. Sending in the Warranty Registration Card entitles users to receive new product updates. Interim software releases are available at the Microchip web site.

RECOMMENDED READING

This user's guide describes how to use PICkit 2 . Other useful documents are listed below. The following Microchip documents are available and recommended as supplemental reference resources.

44-Pin Demo Board User's Guide (DS41296)

Consult this document for instructions on how to use the 44-Pin demo board as a development tool to emulate and debug firmware on a target board.

Low Pin Count Demo Board User's Guide (DS51556)

Consult this document for instructions on how to use Microchip Technology's low pin count device (8-pin, 14-pin and 20-pin). This document includes a series of tutorials.

MPLAB® IDE Quick Start Guide (DS51281)

Describes how to set up the MPLAB IDE software and use it to create projects and program devices.

MPLAB® IDE User's Guide/Help (DS51519)

Consult this document for more information pertaining to the installation and features of the MPLAB Integrated Development Environment (IDE) software. An on-line Help version is also available.

In-Circuit Serial Programmer™ (ICSP™) Guide (DS30277)

This document contains helpful design guidelines for successful ICSP programming. It includes application notes on hardware designs and the ICSP programming specifications.

MPASM™ Assembler, MPLINK™ Object Linker, MPLIB™ Object Librarian User's Guide (DS33014)

Describes how to use the Microchip PIC® MCU assembler (MPASM assembler), linker (MPLINK linker), and librarian (MPLIB librarian).

README for PICkit™ 2 Debug Express

For the latest information on using the PICkit 2 Debug Express, read the "`Readme for PICkit 2.htm`" file (an HTML file) in the Readmes subdirectory of the MPLAB IDE installation directory. The Readme file contains updated information and known issues that may not be included in this user's guide.

Readme Files

For the latest information on using other tools, read the tool-specific Readme files in the Readmes subdirectory of the MPLAB IDE installation directory. The Readme files contain updated information and known issues that may not be included in this user's guide.

THE MICROCHIP WEB SITE

Microchip provides online support via our web site at www.microchip.com. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQs), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

DEVELOPMENT SYSTEMS CUSTOMER CHANGE NOTIFICATION SERVICE

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at www.microchip.com, click on Customer Change Notification and follow the registration instructions.

The Development Systems product group categories are:

- **Compilers** – The latest information on Microchip C compilers and other language tools. These include the MPLAB C18 and MPLAB C30 C compilers; MPASM™ and MPLAB ASM30 assemblers; MPLINK™ and MPLAB LINK30 object linkers; and MPLIB™ and MPLAB LIB30 object librarians.
- **Emulators** – The latest information on Microchip in-circuit emulators.This includes the MPLAB REAL ICE™ and MPLAB ICE 2000 in-circuit emulators.
- **In-Circuit Debuggers** – The latest information on the Microchip in-circuit debuggers. This includes the MPLAB ICD 2 and PICkit™ 2.
- **MPLAB® IDE** – The latest information on Microchip MPLAB IDE, the Windows® Integrated Development Environment for development systems tools. This list is focused on the MPLAB IDE, MPLAB IDE Project Manager, MPLAB Editor and MPLAB SIM simulator, as well as general editing and debugging features.
- **Programmers** – The latest information on Microchip programmers. These include the MPLAB PM3 device programmer and the PICSTART® Plus, PICkit 1 and PICkit 2 development programmers.

CUSTOMER SUPPORT

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or field application engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document. See our web site for a complete, up-to-date listing of sales offices.

Technical support is available through the web site at: http://support.microchip.com.

Chapter 1. PICkit 2 Programmer/Debugger Overview

1.1 INTRODUCTION

This chapter introduces the PICkit 2 Development Programmer/Debugger features and PICkit 2 Programmer application menu functions.

This chapter discusses:

- PICkit 2 Development Programmer/Debugger Contents
- PICkit 2 Development Programmer/Debugger
- PICkit 2 Programmer Application

1.2 PICkit 2 DEVELOPMENT PROGRAMMER/DEBUGGER CONTENTS

The PICkit 2 Development Programmer/Debugger kit contains the following items:

1. The PICkit 2 Development Programmer/Debugger
2. USB cable
3. PICkit Starter Kit and MPLAB IDE CD-ROMs

Additionally, the PICkit 2 Starter Kit and PICkit 2 Debug Express kit both contain a demo board with a PIC microcontroller device.

1.3 PICkit 2 DEVELOPMENT PROGRAMMER/DEBUGGER

The PICkit 2 Development Programmer/Debugger is a low-cost development programmer. It is capable of programming most of Microchip's Flash microcontrollers and serial EEPROM devices. For specific device support, see the `README` file included on the PICkit 2 Starter Kit CD-ROM.

> **Note:** The PICkit 2 is intended for development programming. For production programming, please consider the MPLAB PM3 device programmer or other third party programmers designed for a production environment.

New device support can be added by updating the programming software. The latest software is available on Microchip's web site page for the PICkit 2: www.microchip.com/pickit2.

The PICkit 2 also may be used to debug selected devices. See **Chapter 4. "PICkit 2 Debug Express"** for more details.

The PICkit 2 unit is shown in Figure 1-1.

FIGURE 1-1: PICkit™ 2 MCU PROGRAMMER/DEBUGGER

Legend:

1 – Status LEDs	3 – Lanyard Connection	5 – Pin 1 Marker
2 – Push Button	4 – USB Port Connection	6 – Programming Connector

1.3.1 USB Port Connection

The USB port connection is a USB mini-B connector. Connect the PICkit 2 to the PC using the supplied USB cable.

1.3.2 Status LEDs

The Status LEDs indicate the status of the PICkit 2.

1. **Power** (green) – Power is applied to the PICkit 2 via the USB port.
2. **Target** (yellow) – The PICkit 2 is powering the target device.
3. **Busy** (red) – The PICkit 2 is busy with a function in progress, such as programming.

1.3.3 Push Button

The push button may be used to initiate the Write Device programming function when *Programmer>Write on PICkit Button* is checked on the PICkit 2 Programmer application menu (see item labeled 2 in Figure 1-1.)

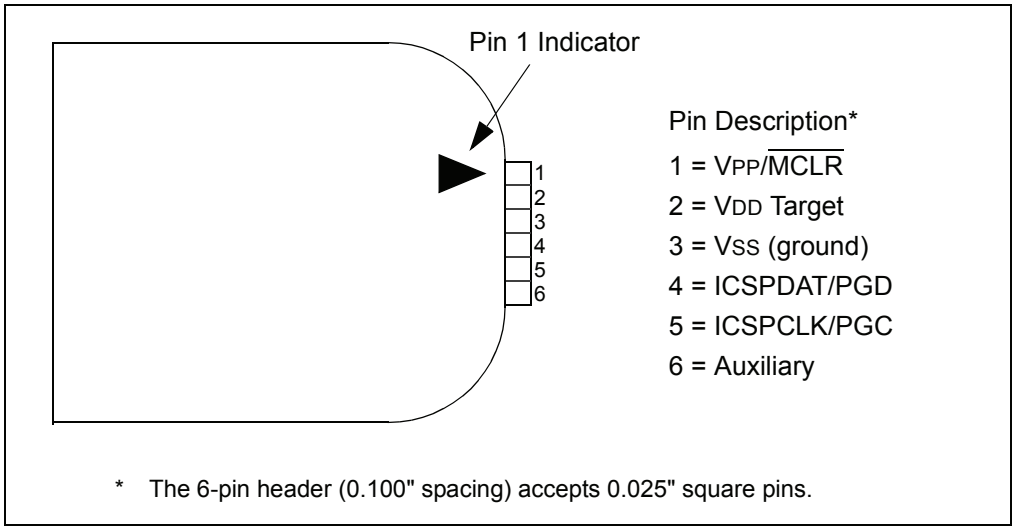
The push button may also be used to put the PICkit 2 unit operating system firmware into Bootloader mode. For more information on this feature, see **Chapter 6. "Updating the PICkit 2 Operating System"**.

1.3.4 Programming Connector

The programming connector is a 6-pin header (0.100" spacing) that connects to the target device. See the pinout specification in Figure 1-2.

For more information on how to use the PICkit 2 with In-Circuit Serial Programming (ICSP), refer to **Chapter 3. "Using In-Circuit Serial Programming™ (ICSP™)"**.

FIGURE 1-2: PICkit™ 2 PROGRAMMER CONNECTOR PINOUT



Note: The programming connector pin functions are different for programming Serial EEPROMS and HCS devices. See the ReadMe file (*Help>Readme)* included with the PICkit 2 programming software for these pinouts.

1.3.5 Lanyard Connection

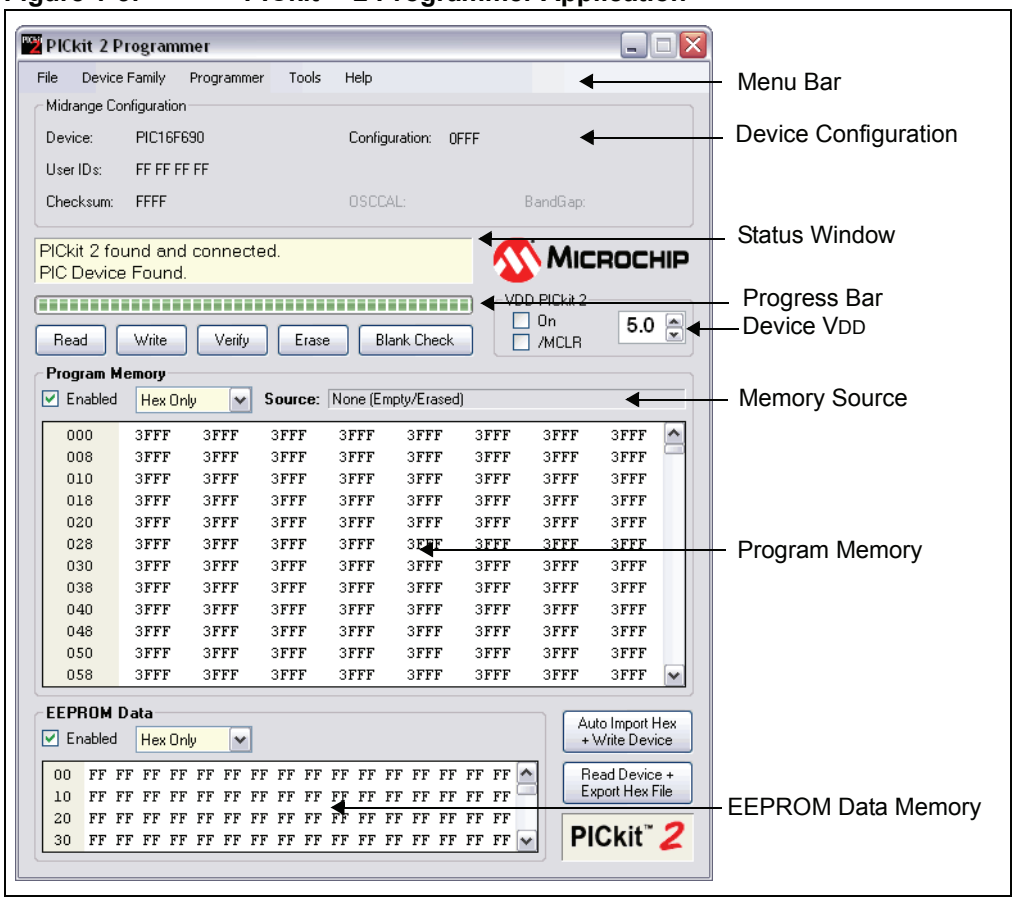
To help prevent possible loss of the PICkit 2, a convenient lanyard connection is available on the programmer.

1.4 PICkit 2 PROGRAMMER APPLICATION

The PICkit 2 Programmer application allows you to program all supported devices listed in the PICkit 2 Readme file. The programming interface appears as shown in Figure 1-3. Its controls are listed in the following sections.

For more information on how to install and use the PICkit 2 Programmer application, see **Chapter 2. "PICkit 2 Programmer Getting Started"**.

Figure 1-3: PICkit™ 2 Programmer Application



1.4.1 Menu Bar

The menu bar selects various functions of the PICkit 2 Programmer application. A summary of the functions are:

File

- Import Hex – Import a hex file for programming. The hex file format INHX32 is supported.
- Export Hex – Export a hex file read from a device.The hex file is created in the INHX32 format.
- File History – Up to the last four hex files opened are displayed with their filepath. These recent hex files may be selected to quickly import them. Note that the file history will initially be blank on a new installation until a hex file is imported.
- Exit – Exit the program.

Device Family

Select a device family to search for a connected device in that family. Selecting the device family of the current part will clear all device data.

Some families which cannot be auto-detected (such as Baseline) will bring up a drop down box from which supported devices may be selected.

Programmer

- Read Device – Reads program memory, data EEPROM memory, ID locations and Configuration bits.
- Write Device – Writes program memory, data EEPROM memory, ID locations and Configuration bits.
- Verify – Verifies program memory, data EEPROM memory, ID locations and Configuration bits read from the target MCU against the code stored in the programming application.
- Erase – Performs a Bulk Erase of the target MCU. OSCCAL and band gap values are preserved on parts with these features.
- Blank Check – Performs a Blank Check of program memory, data EEPROM memory, ID locations and Configuration bits.
- Verify on Write – When checked, the device will be immediately verified after programming on a Write (recommended). When unchecked, the device will be programmed but not verified on a Write.
- Hold Device in Reset – When checked, the $\overline{\text{MCLR}}$ (V_{PP}) pin is held low (asserted). When unchecked, the pin is released (tri-stated), allowing an external pull-up to bring the device out of Reset.
- Write on PICkit Button – When checked, a Write operation will be initiated by pressing the PICkit 2 push button.

Tools

- Enable Code Protect – Enables code protection features of the microcontroller on future Write operations.

Note:	To disable code protect, *all* device memory must be erased and rewritten.

- Enable Data Protect – Enables data protection feature of microcontrollers with data EEPROM memory on future Write operations.

Note:	To disable data protect, *all* device memory must be erased and rewritten.

- Set OSCCAL – Allows the OSCCAL value to be changed for devices where it is stored in the last location of Program Memory.
- Target V_{DD} Source
  - Auto-Detect – The PICkit 2 will automatically detect whether the target device has its own power supply or needs to be powered by the programmer on each operation.
  - Force PICkit 2 – The PICkit 2 will always attempt to supply V_{DD} to the target device.
  - Force Target – The PICkit 2 will always assume the target has its own power supply.
- Calibrate V_{DD} & Set Unit ID – Opens a wizard that steps the user through calibrating the PICkit 2 V_{DD} supplied voltage so it is more accurate, and optionally assigning a Unit ID to identify between multiple PICkit 2 devices.
- Use V_{PP} First Program Entry – When checked, it allows the PICkit 2 to connect to and program devices with configurations and code that interferes with the ICSP signal pins, preventing PICkit 2 from detecting them. Using this feature requires that the PICkit 2 supplies V_{DD} to the target.
- Fast Programming – When checked, the PICkit 2 will attempt to program the device as fast as possible. When unchecked, the PICkit 2 will slow down ICSP communication. This may be helpful for targets with loaded ICSP lines.
- Check Communication – Verifies USB communication with the PICkit 2 and ICSP communication with a target device by attempting to identify the connected device by its device ID.
- UART Tool… - Puts the PICkit 2 in UART Mode and opens a terminal-like interface for communicating with a PIC MCU device program through the USART pins. See **Chapter 7. "PICkit 2 UART Tool"** for more information.
- Troubleshoot... – Opens a wizard to help with troubleshooting connectivity from the PICkit 2 to the target device. This is most useful where the programmer is unable to detect the target device at all.
- Download PICkit 2 Programmer Operating System – Performs a download of the PICkit 2 operating system (firmware).

Help

- PICkit 2 User's Guide – Attempts to launch the user's guide PDF (Adobe® Reader must be installed).
- 44-Pin Demo Board Guide – Attempts to launch the 44-Pin Demo Board User's Guide PDF (Adobe Reader must be installed).
- LPC Demo Board Guide – Attempts to launch the Low Pin Count Demo Board User's Guide PDF (Adobe Reader must be installed).
- PICkit 2 Programmer on the web – Opens www.microchip.com/pickit2 in the default web browser.
- Readme – Opens the PICkit 2 `Readme.txt` file.
- About – Opens a dialog with the PICkit 2 Programmer application version, device file version and firmware version.

1.4.2 Device Configuration

The Device Configuration window displays the device, User ID, Configuration Word and Checksum. It also displays OSCCAL and Band Gap for parts with those features.

For baseline (12-bit core) devices, serial EEPROM devices, and KeeLoq® HCS devices, you must select the device from the Device drop-down menu.

All other part family devices will be detected by their device ID and the part name will be displayed on the Device line.

1.4.3 Status Window

The status window displays text status of the operations in progress. If an operation is successful, the status window will display a green background. If an operation fails, the status window will display red. If an operation alerts a caution, the status window will display yellow.

1.4.4 Progress Bar

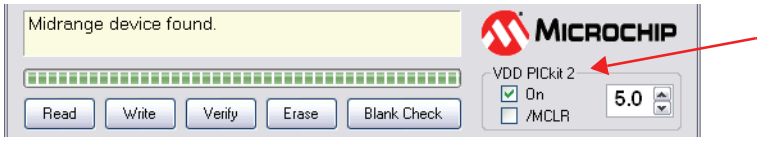
The progress bar displays the progress of an operation.

1.4.5 Device VDD

The PICkit 2 VDD may be turned on and off by clicking the checkbox “On”. The voltage may be set in the box on the right either by typing it directly or using the up/down arrows to adjust it a tenth of a volt at a time. The maximum and minimum allowed voltages will vary depending on the target device.

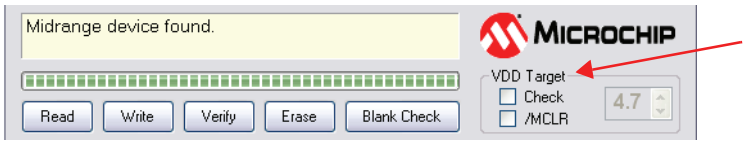
If the “On” checkbox is unchecked, PICkit 2 will automatically turn on the VDD at the set voltage during any requested programming operation.

FIGURE 1-4: PICKIT™ 2 SUPPLIED VDD



If the target device has its own power supply, then the PICkit 2 will display the detected VDD voltage in the box on the right, which will be grayed out to prevent being changed. The checkbox text changes to “check”, and clicking on the checkbox will update the detected VDD voltage value. If *Target VDD>Auto-Detect* is selected, clicking on the checkbox will revert the VDD mode back to PICkit 2 supplied VDD if a target power supply is no longer detected.

FIGURE 1-5: TARGET SUPPLIED VDD



1.4.6 Device $\overline{\text{MCLR}}$ State

The “/MCLR” checkbox shown in Figure 1-4 and Figure 1-5 has the same functionality as the menu selection *Programmer>Hold Device in Reset*. When the box is checked the target device will be held in Reset. When unchecked, the target circuit is allowed to pull $\overline{\text{MCLR}}$ up to VDD to release the device from Reset. This function can be used to prevent a device from executing code before and after programming.

Note: If the target device allows the $\overline{\text{MCLR}}$ pin to be configured as an input port, and it is configured as such, PICkit 2 will not be able to hold the device in Reset.

1.4.7 Memory Source

The Source bar displays the source of the currently loaded device data. If read from a hex file, it will display the hex file name. If read from a device, it will display the part name. `None (Empty/Erased)` indicates the buffers are empty, and it will display `Edited` once Program Memory or Data EEPROM Memory has been edited in the window.

1.4.8 Program Memory

Program code can be loaded into the PICkit 2 Programmer application by selecting *File>Import HEX* to import a hex file or by clicking **Read** to read the device memory. The origin of the code is displayed in the Source block. The Program Memory window displays the program code in hexadecimal. The code may be edited in the window.

The checkbox next to the Program Memory window is only available on devices with EEPROM data memory. If the box is checked, then Program Memory, User IDs, and Configuration Words are written to, read from, and verified on the device. If the box is unchecked, then Program Memory, User IDs, and Configuration Words will not be erased or altered during a Write Device operation, and will not be read or verified. The checkbox does not affect Erase Device or Blank Check operations. Both memory window checkboxes may not be cleared at the same time.

For supported serial EEPROM devices, the device contents are displayed in the Program Memory window instead of the Data EEPROM Memory window for easier viewing in the larger display area.

1.4.9 Data EEPROM Memory

Similar to Program Memory above, data EEPROM code can be loaded into the PICkit 2 Programmer application by selecting *File>Import HEX* to import a hex file or by clicking **Read** to read the device memory. The origin of the code is displayed in the Source block. The Data EEPROM Memory window displays the program code in hexadecimal. The code may be edited in the window.

The check box next to the EEPROM Data window controls whether the EEPROM Data memory is written, read and verified. If the box is checked, then the device EEPROM will be overwritten with the window data. If the box is not checked, then the device EEPROM will not be erased or altered during a Write Device operation. The checkbox does not affect Erase Device or Blank Check operations. Both memory window checkboxes may not be cleared at the same time.

Chapter 2. PICkit 2 Programmer Getting Started

2.1 INTRODUCTION

This chapter gives instruction on how to get started using the PICkit 2 Development Programmer/Debugger to program Flash-based PIC microcontroller units.

For information on how to use the PICkit 2 with In-Circuit Serial Programming™ (ICSP™), refer to **Chapter 3. "Using In-Circuit Serial Programming™ (ICSP™)"**.

For information on how to update the PICkit 2 operating system (firmware), refer to **Chapter 6. "Updating the PICkit 2 Operating System"**.

For information on using the PICkit 2 as a debugger in MPLAB IDE, see **Chapter 4. "PICkit 2 Debug Express"**.

- Installing the PICkit 2 Hardware
- Installing and Launching the PICkit 2 Programmer Application
- Connecting to the Device
- Selecting Target Power
- Importing a Hex File
- Writing the Program to the Device
- Verifying the Device
- Reading Device Memory
- Code Protecting the Device
- Erasing and Blank Checking the Device
- Automating Write/Read Procedures

2.2 INSTALLING THE PICkit 2 HARDWARE

To install the PICkit 2 hardware:

- Plug one end of the USB cable into PICkit 2 USB connector. Plug the other end into a USB port on your PC.
- Connect the PICkit 2 to a target board via a 6-pin connector. The target board can be the included demo board or any target equipped with the appropriate 6-pin connector.
- Do not connect the PICkit 2 to a target board that has its own power supply if it is not connected to a powered USB port.
- To connect the PICkit 2 to a target with an MPLAB ICD 2 style RJ-11 connector, the AC164110 RJ-11 to ICSP Adapter kit is required.

When plugging the PICkit 2 into the USB, it is recommended to disconnect it from any target board first. Similarly, when starting up or rebooting the host PC, ensure it is disconnected from a target.

For more information about the PICkit 2 hardware, see **Section 1.3 "PICkit 2 Development Programmer/Debugger"**.

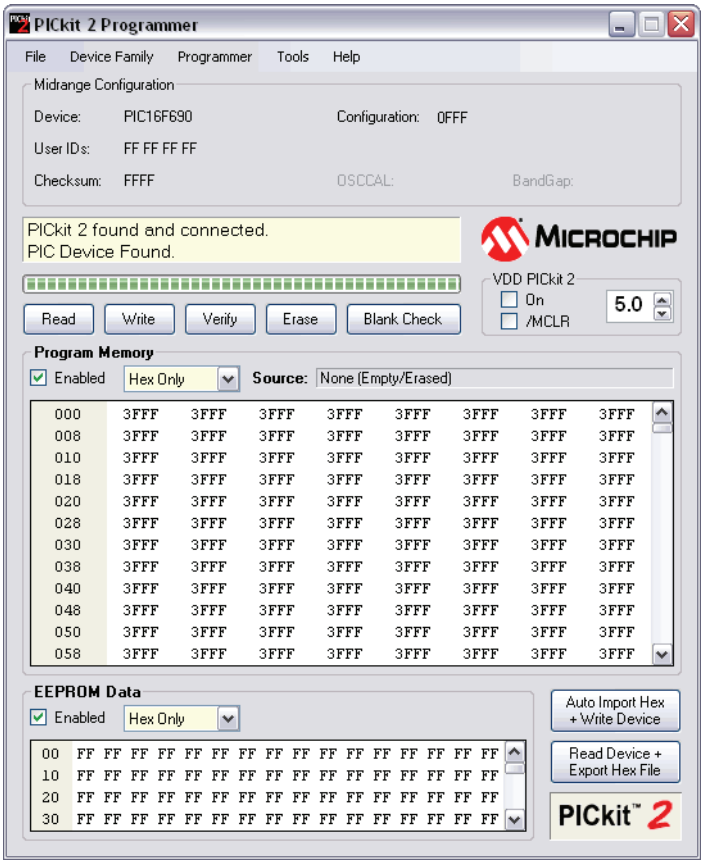
2.3 INSTALLING AND LAUNCHING THE PICkit 2 PROGRAMMER APPLICATION

Insert the PICkit 2 Starter Kit CD-ROM into the CD-ROM drive. In a few moments, the introductory screen should be displayed. Follow the directions on the screen for installing the PICkit 2 Programmer application.

If the introductory screen does not appear, browse to the CD-ROM directory and open the `PICkit_Starter_Kit_Welcome.htm` file.

Once installed, start the PICkit 2 Programmer application by selecting *Start>Programs>Microchip>PICkit 2*. The programming interface appears as shown in Figure 2-1. A listing of its features and functions may be found in **Section 1.4 "PICkit 2 Programmer Application"**.

FIGURE 2-1: PICkit™ 2 PROGRAMMING APPLICATION

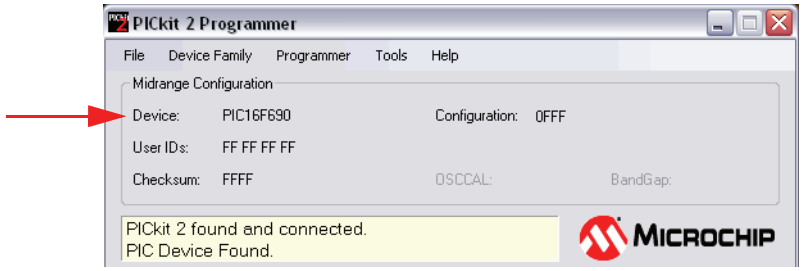


2.4 CONNECTING TO THE DEVICE

The PICkit 2 is capable of programming a variety of Flash-based Microchip PIC micro-controllers and serial EEPROM devices. Supported devices are listed in the PICkit 2 Readme file on the CD-ROM, which can also be viewed by selecting *Help>Readme*.

When the PICkit 2 Programmer application is first opened, it will attempt to identify the connected device by the device ID and display it in the Configuration window as shown in Figure 2-2.

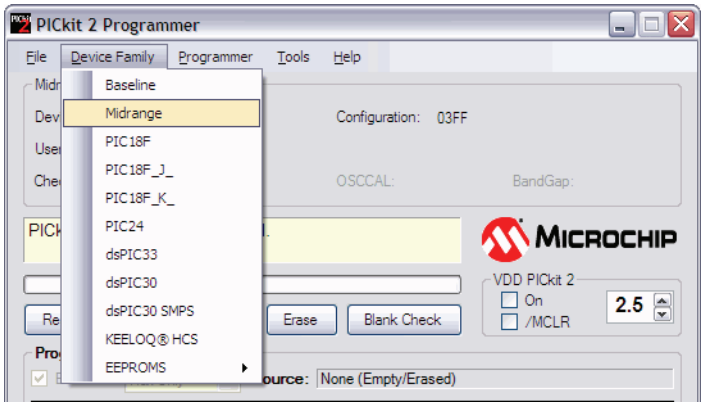
FIGURE 2-2: IDENTIFY DEVICE



If the device on the target is not correctly identified, check the target power (**Section 2.5 "Selecting Target Power"**) and device ICSP connections before attempting to reselect or change the device.

At any time, the device family may be selected to search for connectivity to a device in that family. To connect to a device once the application is already running, select the device family by clicking on the Device Family menu as shown in Figure 2-3.

FIGURE 2-3: SELECT DEVICE FAMILY

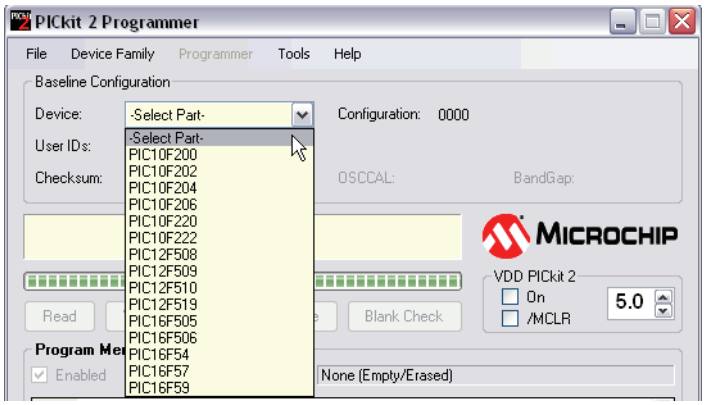


If the Baseline (12-bit core), KEELOQ® HCS or EEPROMs device family is selected, you must select the specific device from the device drop-down box as shown in Figure 2-4. These devices do not have a device ID and do not support automatic detection.

CAUTION

Ensure that the correct Baseline has been selected. These devices do not contain a device ID to confirm device selection. Choosing the wrong Baseline may cause an erasing of the OSCCAL value stored in the last memory location.

FIGURE 2-4: SELECT BASELINE FLASH DEVICE



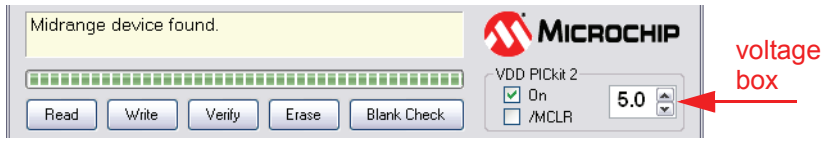
2.5 SELECTING TARGET POWER

The PICkit 2 can supply power to the target or the target may be powered externally.

2.5.1 Target Powered from PICkit 2

If you are going to power the target board from the PICkit 2, do not attach a power supply to the target or the PICkit 2 will sense it and not give you the option to use PICkit 2 power. For a target board not connected to an external power supply, you will see the options displayed in Figure 2-5.

FIGURE 2-5: ENABLE POWER FROM PICkit™ 2



To enable power to the target device, check the VDD PICkit 2 "On" checkbox as shown. The default setting is "Off", i.e., the checkbox is unchecked.

Note: If a target power supply is not detected, the PICkit 2 will always supply power to the target during programming, regardless of the VDD PICkit 2 "On" checkbox state.

The voltage supplied to the target may be adjusted before or after enabling power by adjusting the VDD PICkit 2 voltage box (Figure 2-5).

If a short or heavy current load is detected on the programmer-supplied VDD, then you will receive an error and VDD will be automatically disabled. Refer to Figure 2-6.

CAUTION

The USB port current limit is set to 100 mA. If the target plus the PICkit 2 exceed this current limit, the USB port may turn off. The target may be powered externally if more power is required.

FIGURE 2-6: VDD ERROR

To avoid heavy current load errors, it is recommended to keep the target current consumption below 25 mA. Large VDD capacitances should also be avoided as they slow down the VDD risetime. The allowed VDD rise time is 500 µs or less.

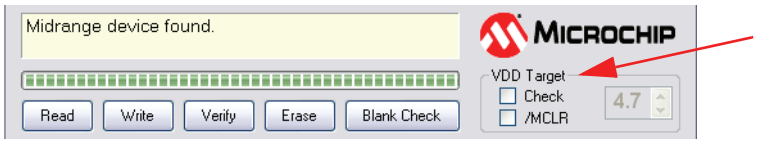
2.5.2 Target Powered from External Supply

The target device may also be powered externally. By default, the PICkit 2 will automatically detect an externally powered board. The heading "VDD PICkit 2" will be changed to "VDD Target", the "On" checkbox will be replaced by a checkbox named "Check", and the detected VDD voltage is displayed in the grayed out voltage box as in Figure 2-7.

Clicking the "Check" checkbox will update the detected VDD voltage displayed in the voltage box. If no VDD voltage is detected when the checkbox is clicked, then PICkit 2 will return to supplying VDD power to the target device.

Note: The maximum external VDD that may be used with the PICkit 2 is 5.0 Volts. The minimum external VDD that may be used with the PICkit 2 is 2.5 Volts.

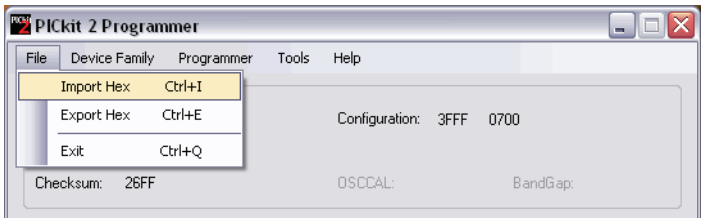
FIGURE 2-7: EXTERNALLY POWERED TARGET



2.6 IMPORTING A HEX FILE

To import a compiled program (hex file) to be programmed into the target device, select *File>Import HEX* as shown in Figure 2-8.

FIGURE 2-8: IMPORT HEX FILE

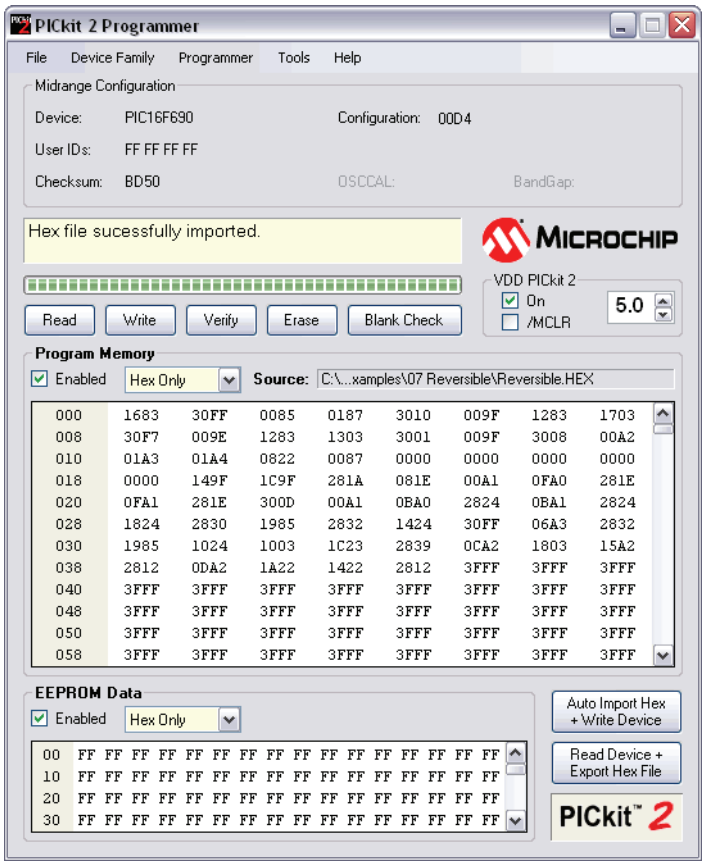


Example source code and hex files may be found under the `Install\Lessons\` directory for the appropriate kit demo board on the PICkit 2 Starter Kit CD-ROM. The hex file `Reversible.hex` from the folder `07 Reversible` will be used.

Browse for the hex file and click **Open**. The code is displayed in the Program Memory and EEPROM Data windows. The name of the hex file is displayed in the Source block under Program Memory.

Note: The example lesson code is meant to be installed to the local hard drive from the CD-ROM using the Setup files in the CD-ROM directory `Install/Lessons/`. The lessons are covered in the user's guide for the included kit Demo Board, and this guide should be read before using the lesson projects and source code.

FIGURE 2-9: EXAMPLE HEX FILE IMPORTED



The PICkit 2 Programmer application will warn you if the hex file does not contain any Configuration Words. You can be sure these are included in the hex file by selecting *File>Export* in MPLAB IDE to save a hex file including configuration memory. (For MPLAB IDE usage, see **Chapter 4. "PICkit 2 Debug Express"**.)

You will also be warned that the hex file is larger than the selected device if the hex file contains memory locations that do not exist in the current device. Any data for non-existent locations will not be imported.

2.7 WRITING THE PROGRAM TO THE DEVICE

After a device family has been selected and a hex file has been imported, the target device can be programmed by clicking **Write** (Figure 2-10). The device will be erased and programmed with the hex code previously imported.

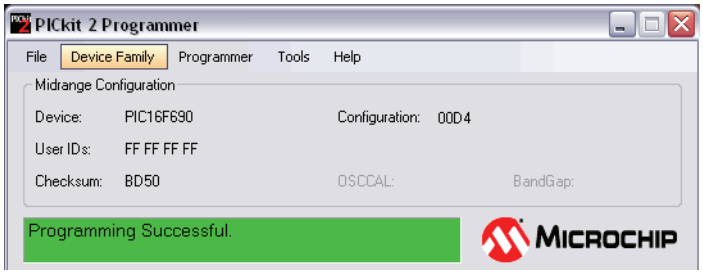
When erasing the device during programming, a Bulk Erase method is used. All Baseline, Mid-Range, and many dsPIC30F and PIC18F devices require a minimum VDD for the Bulk Erase. Some of these devices support a low voltage row erase method that can be used at lower voltages, but this method takes longer to erase the device. See the Readme file under *Help>Readme* for devices that support this feature. If a device does not support row erasing, a dialog will pop up to warn you if the device VDD is below the minimum required for a Bulk Erase.

Note: If any Code Protect, Data Protect, Write Protect, or Read Protect configuration bits are currently set in the device, the Bulk Erase method must be used prior to programming. The lower voltage row erase procedure will not succeed.

FIGURE 2-10: BUTTONS – WRITE

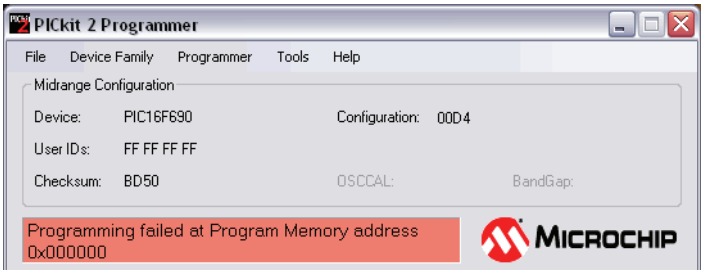
The status of the Write operation is displayed in the status bar located under the Device Configuration window. If the write is successful, the status bar turns green and displays “Programming Successful”, as shown in Figure 2-11.

FIGURE 2-11: WRITE SUCCESSFUL STATUS



If the write fails, the status bar turns red and displays “Programming Failed”, as shown in Figure 2-12. This error indicates that the data was corrupted during the programming sequence. If this error is displayed, try writing the program to the device again. If the error continues, see **Chapter 5. “Troubleshooting”** for assistance.

FIGURE 2-12: WRITE ERROR STATUS



Other write issues may be displayed as warnings and will turn the status bar yellow as in Figure 2-13. In this case, the PICkit 2 and demo board had become disconnected.