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™ 3 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.

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^{32} 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, CodeGuard, dsPICDEM, dsPICDEM.net, dsPICworks, dsSPEAK, ECAN, ECONOMONITOR, FanSense, HI-TIDE, In-Circuit Serial Programming, ICSP, Mindi, MiWi, MPASM, MPLAB Certified logo, MPLIB, MPLINK, mTouch, Octopus, Omniscient Code Generation, PICC, PICC-18, PICDEM, PICDEM.net, PICkit, PICtail, REAL ICE, 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.

Printed on recycled paper.

ISBN: 978-1-60932-373-8

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

Chapter 6. PICkit 3 Programmer-To-Go

Chapter 7. Troubleshooting First Steps

Chapter 8. Frequently Asked Questions (FAQs)

Chapter 9. Error Messages

Chapter 10. Debugger Function Summary

Appendix A. Hardware Specification

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 online help. Select the Help menu, and then Topics to open a list of available online help files.

INTRODUCTION

This chapter contains general information that will be useful to know before using the PICkit™ 3 programmer/debugger. Items discussed include:

- Document Layout
- Conventions Used in this Guide
- Warranty Registration
- Recommended Reading
- The Microchip Web Site
- Development Systems Customer Change Notification Service
- Customer Support
- Revision History

DOCUMENT LAYOUT

This document describes how to use the PICkit 3 as a development tool to emulate and debug firmware on a target board. The manual layout is as follows:

- **Chapter 1. Overview** – An overview of the PICkit 3 programmer/debugger.
- **Chapter 2. Theory of Operation** – A simplified description of how the PICkit 3 programmer/debugger works.
- **Chapter 3. Installation** – How to install the PICkit 3 programmer/debugger.
- **Chapter 4. General Setup** – Provides Instructions on how to get started using the PICkit 3 programmer/debugger to program supported devices.
- **Chapter 5. PICkit 3 Debug Express** – Provides basic information on using the PICkit™ 3 Debug Express.
- **Chapter 6. PICkit 3 Programmer-To-Go** – Provides instruction on using the PICkit 3 unit to program devices without being connected to a PC.
- **Chapter 7. Troubleshooting First Steps** – Begins the troubleshooting process by identifying first steps and common reasons for problems with debugging.
- **Chapter 8. Frequently Asked Questions (FAQs)** – Provides information on solving common problems.
- **Chapter 9. Error Messages** – Provides specific error messages and general corrective actions.
- **Chapter 10. Debugger Function Summary** – Summarizes the available debugging functions.
- **Appendix A. Hardware Specification** – Details hardware and electrical specifications for the PICkit 3.
- **Appendix B. PICkit 3 Schematics** – Provides hardware schematic diagrams for the PICkit 3 programmer/debugger.

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	<u>*File>Save*</u>
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 3 . Other useful documents are listed below. The following Microchip documents are available and recommended as supplemental reference resources.

In-Circuit Debugger Design Advisory (DS51764)

Please read this first! This document contains important information about operational issues that should be considered when using the PICkit 3 with your target design.

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 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 online 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™ 3 Debug Express

For the latest information on using the PICkit 3 Debug Express, read the “`Readme for PICkit 3.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.

PICkit™ 3 Debug Express C18 Lessons

These tutorials guide you through using the PICkit 3 Debug Express with the MPLAB C Compiler for PIC18 MCU’s. They are available on the MPLAB IDE CDROM and on the Microchip web site.

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, assemblers, linkers and other language tools. These include all MPLAB C compilers; all MPLAB assemblers (including MPASM™ assembler); all MPLAB linkers (including MPLINK™ object linker); and all MPLAB librarians (including MPLIB™ object librarian).
- **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 MPLAB ICD 3 in-circuit debuggers and PICkit™ 2 and 3 debug express.
- **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 production programmers such as MPLAB REAL ICE in-circuit emulator, MPLAB ICD 3 in-circuit debugger, and MPLAB PM3 device programmers. Also included are nonproduction development programmers such as PICSTART® Plus and PICkit 1, 2 and 3.

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.

Documentation errors or comments may be e-mailed to docerrors@microchip.com.

REVISION HISTORY

Revision A (January 2009)

This is the initial release of this document.

Revision B (July 2010)

Section 1.5 Device and Feature Support has been removed. It is available through the PICkit 3 online help in MPLAB IDE.

Section 2.4.4 Debugger Powered – third paragraph was replaced.

Section 2.7 Programming – modified to include the three ways to program a device.

Added a new Chapter 6. PICkit 3 Programmer-To-Go.

Table 10.6 Program Memory Options – added new option for "Use high voltage on MCLR" to table.

Section 10.5.8 Settings Dialog, Power Tab – added explanatory text regarding the "Power target circuit from PICkit 3" option.

Section 10.5.10 Setting Dialog, Programmer-to-go Tab has been added.

Renamed Chapter 11 Hardware Specification to Appendix A Hardware Specification.

Part 1 – Getting Started

NOTES:

Chapter 1. Overview

1.1 INTRODUCTION

An overview of the PICkit 3 programmer/debugger system is given.

- PICkit 3 Programmer/Debugger Defined
- How the PICkit 3 Programmer/Debugger Helps You
- PICkit 3 Programmer/Debugger Components

1.2 PICkit 3 PROGRAMMER/DEBUGGER DEFINED

The PICkit 3 programmer/debugger (see Figure 1-1) is a simple, low-cost in-circuit debugger that is controlled by a PC running MPLAB IDE (v8.20 or greater) software on a Windows® platform. The PICkit 3 programmer/debugger is an integral part of the development engineer’s toolsuite. The application usage can vary from software development to hardware integration.

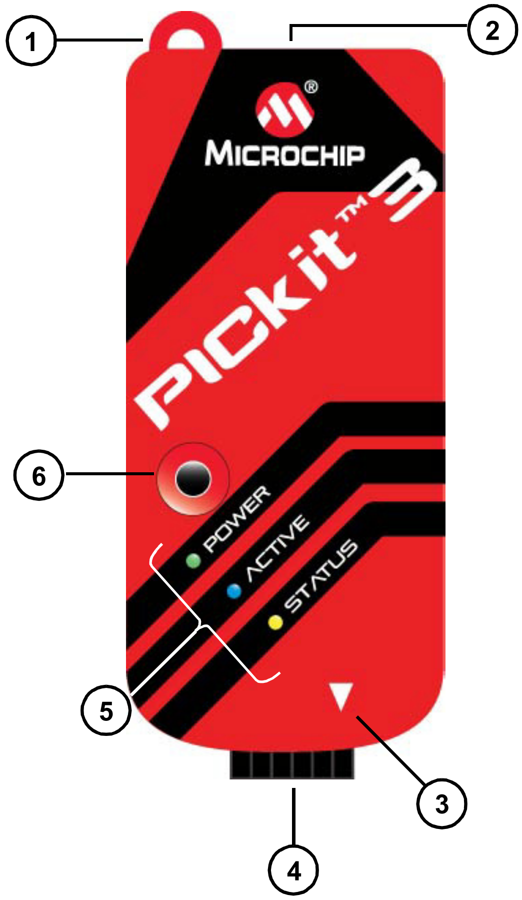
The PICkit 3 programmer/debugger is a debugger system used for hardware and software development of Microchip PIC® microcontrollers (MCUs) and dsPIC® Digital Signal Controllers (DSCs) that are based on In-Circuit Serial Programming™ (ICSP™) and Enhanced In-Circuit Serial Programming 2-wire serial interfaces. In addition to debugger functions, the PICkit 3 programmer/debugger system also may be used as a development programmer. The PICkit 3 programmer/debugger is not intended to be used as a production programmer.

The debugger system executes code like an actual device because it uses a device with built-in emulation circuitry, instead of a special debugger chip, for emulation. All available features of a given device are accessible interactively, and can be set and modified by the MPLAB IDE interface.

The PICkit 3 debugger was developed for emulating embedded processors with debug facilities. The PICkit 3 features include:

- Full-speed USB support using Windows standard drivers
- Real-time execution
- Processors run at maximum speeds
- Built-in over-voltage/short circuit monitor
- Low voltage to 5V (1.8-5V range)
- Diagnostic LEDs (power, active, status)
- Read/write program and data memory of microcontroller
- Erase of all memory types (EEPROM, ID, configuration and program) with verification
- Peripheral freeze at breakpoint

FIGURE 1-1: PICkit™ 3 MCU PROGRAMMER/DEBUGGER



Legend:

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

1.2.1 Lanyard Connection

A convenient lanyard connection is available on the programmer.

1.2.2 USB Port Connection

The USB port connection is a USB mini-B connector. Connect the PICkit 3 to the PC using the supplied USB cable.

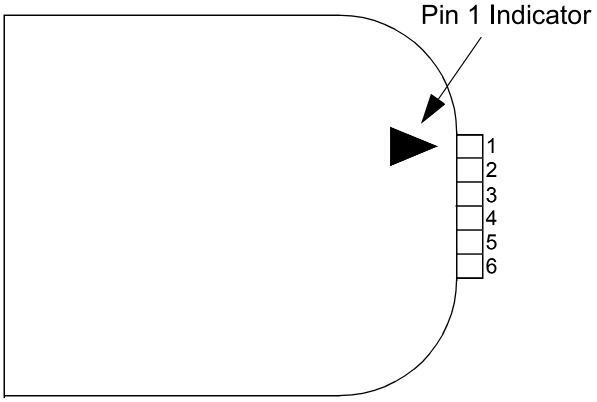
1.2.3 Pin 1 Marker

This marker designates the location of pin 1 for proper connector alignment.

1.2.4 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.

FIGURE 1-2: PICkit™ 3 PROGRAMMER CONNECTOR PINOUT



Pin Description*

1 = $\overline{\text{MCLR}}$/VPP

2 = VDD Target

3 = VSS (ground)

4 = PGD (ICSPDAT)

5 = PGC (ICSPCLK)

6 = PGM (LVP)

* The 6-pin header (0.100" spacing) accepts 0.025" square pins.

> **Note:** The programming connector pin functions are different for programming Serial EEPROM devices. See the ReadMe file for the PICkit 3 (*Help>Readme)* included with the MPLAB IDE software for these pinouts.

1.2.5 Status LEDs

The Status LEDs indicate the status of the PICkit 3.

1. **Power** (green) – Power is supplied to the PICkit 3 via the USB port.
2. **Active** (blue) – The PICkit 3 has connection to the PC USB port and the communication link is active.
3. **Status:**

   Busy (yellow) – The PICkit 3 is busy with a function in progress, such as programming.

   Error (red) – The PICkit 3 has encountered an error.

1.2.6 Push Button

The push button is used for Programmer-To-Go. See **Chapter 6. “PICkit 3 Programmer-To-Go”**.

1.3 HOW THE PICkit 3 PROGRAMMER/DEBUGGER HELPS YOU

The PICkit 3 programmer/debugger allows you to:

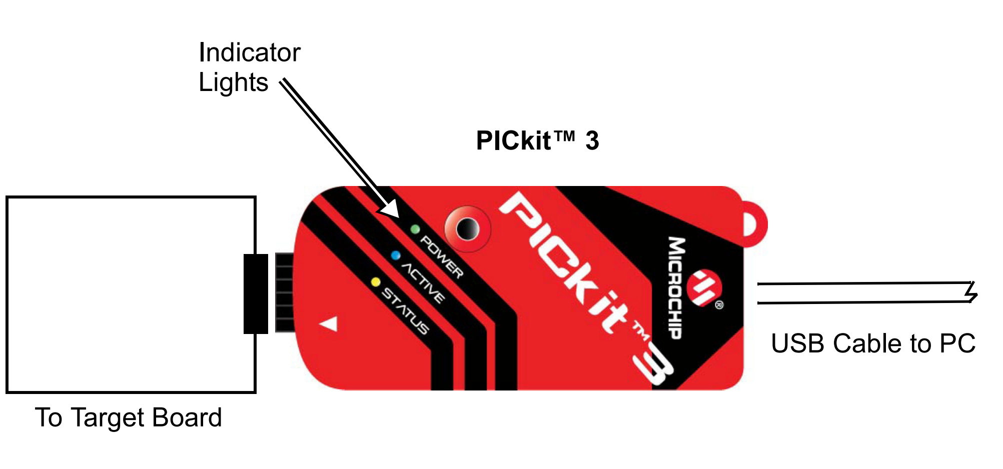
- Debug your application on your own hardware in real time
- Debug with hardware breakpoints
- Set breakpoints based on internal events
- Monitor internal file registers
- Emulate at full speed
- Program your device

1.4 PICkit 3 PROGRAMMER/DEBUGGER COMPONENTS

The components of the PICkit 3 programmer/debugger system are:

1. PICkit 3 with indicator lights for power, activity and status.
2. USB cable to provide communications between the debugger and a PC and to provide power to the debugger.
3. CD-ROM with MPLAB IDE software and online documentation.

FIGURE 1-3: BASIC DEBUGGER SYSTEM



Additional hardware that may be ordered separately:

- PICkit 3 Debug Express Kit which includes:
  - a 44-pin demo board with a PIC18F45K20 MCU
  - free version of MPLAB C Compiler for PIC18 MCUs
  - easy-to-understand lessons and tutorials
  - other software utilities, examples with source code and full documentation
- Transition socket
- ICD headers
- MPLAB IDE processor extension kits

Chapter 2. Theory of Operation

2.1 INTRODUCTION

A simplified description of how the PICkit 3 programmer/debugger system works is provided here. It is intended to provide enough information so a target board can be designed that is compatible with the debugger for both emulation and programming operations. The basic theory of in-circuit emulation and programming is described so that problems, if encountered, are quickly resolved.

- PICkit 3 vs. PICkit 2
- Debugger to Target Communication
- Communication Connections
- Debugging
- Requirements for Debugging
- Programming
- Resources Used by the Debugger

2.2 PICkit 3 VS. PICkit 2

The PICkit 3 programmer/debugger system is similar in function to the PICkit 2 in-circuit debugger system. Similarities of the two debuggers include:

- Powered via USB cable to PC
- Provides a programmable voltage power supply

The PICkit 3 differs from the PICkit 2 by providing:

- Extended EE program image space (512 Kbytes)
- True voltage reference
- Increased voltage range (1.8-5V V_{DD}; 1.8-14V V_{PP})

2.3 DEBUGGER TO TARGET COMMUNICATION

The debugger system configurations are discussed in the following sections.

CAUTION
Do not change hardware connections while the PICkit 3 or target is powered.

Standard ICSP Device Communication

The debugger system can be configured to use standard ICSP communication for both programming and debugging functions. This 6-pin connection is the same one used by the PICkit 2 programmer/debugger.

The modular cable can be either (1) inserted into a matching socket at the target, where the target device is on the target board (Figure 2-1), or (2) inserted into a standard adapter/header board combo (available as a Processor Pak), which is then plugged into the target board (Figure 2-2).

> **Note:** Older header boards used a 6-pin (RJ-11) connector instead of an 8-pin connector, so these headers may be connected to the debugger with the AC164110 ICSP adapter.

For more on standard communication, see **Appendix A. "Hardware Specification"**.

FIGURE 2-1: STANDARD DEBUGGER SYSTEM – DEVICE WITH ON-BOARD ICE CIRCUITRY

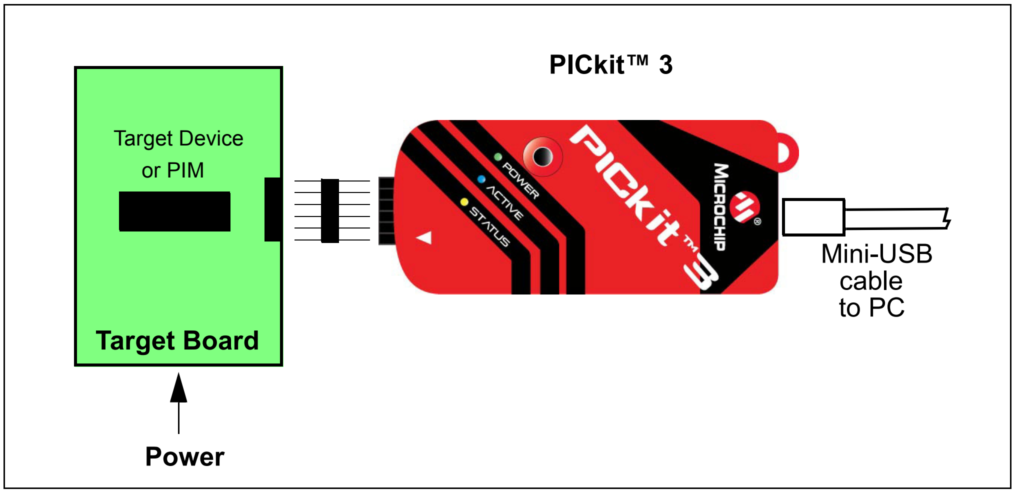
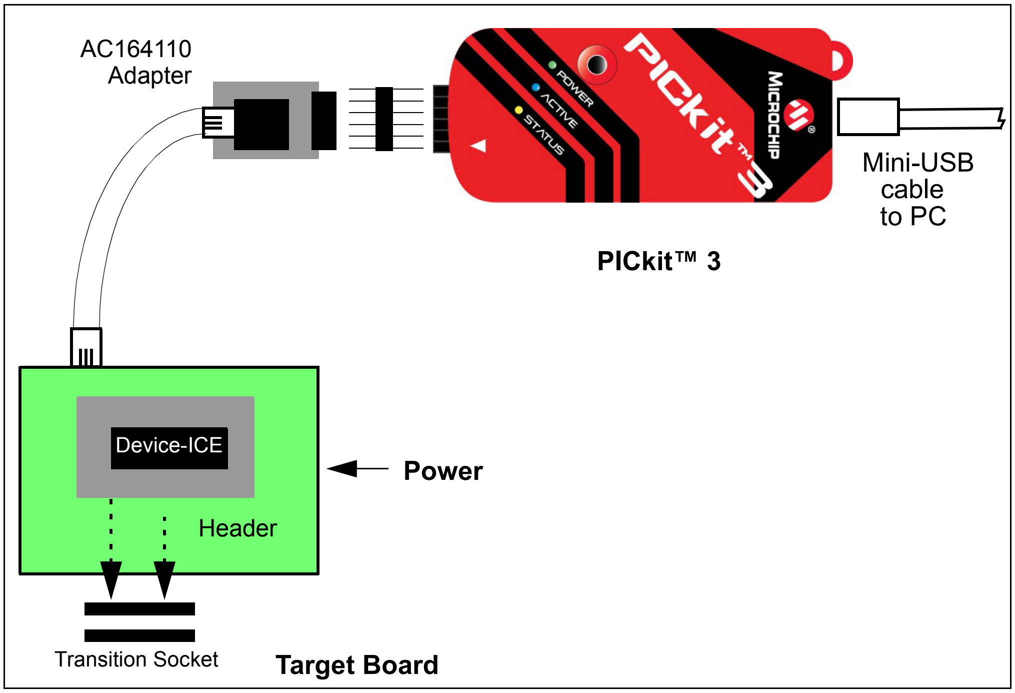


FIGURE 2-2: STANDARD DEBUGGER SYSTEM – ICE DEVICE



2.4 COMMUNICATION CONNECTIONS

2.4.1 Communication Target Connections

2.4.1.1 USING SINGLE IN-LINE CONNECTOR

Use the 6-pin in-line connector between the PICkit 3 programmer/debugger and the target board connector. See Figure 2-1. Also see Table 2-1 and **Section A.6 “Standard Communication Hardware”**.

TABLE 2-1: TARGET CONNECTOR PINOUT

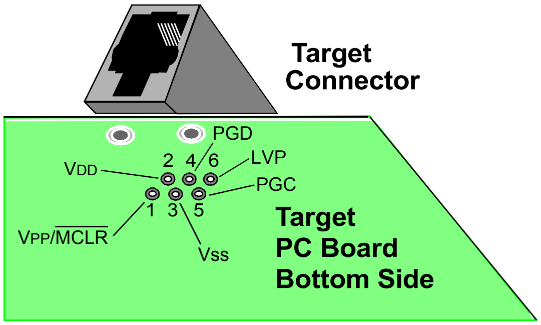
Connector Pin	Microcontroller Pin
1	$\overline{\text{MCLR}}$/VPP
2	VDD
3	Ground
4	PGD (ICSPDAT)
5	PGC (ICSPCLK)
6	PGM (LVP)

2.4.1.2 USING AN ADAPTER

Use the AC164110 adapter between the PICkit 3 programmer/debugger and the target device with the modular interface (six conductor) cable. The pin numbering for the connector is shown from the bottom of the target PC board in Figure 2-3.

> **Note:** Cable connections at the debugger and target are mirror images of each other, i.e., pin 1 on one end of the cable is connected to pin 6 on the other end of the cable. See **Section A.6.2.3 “Modular Cable Specification”**.

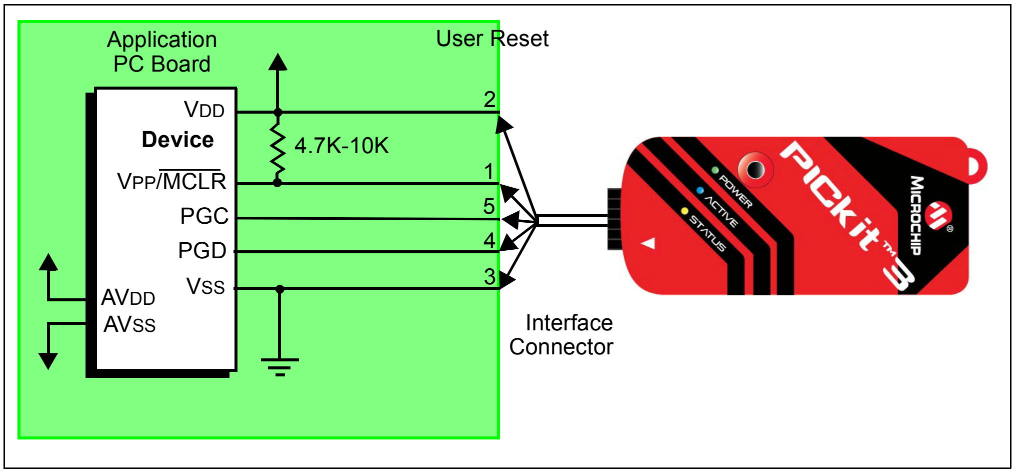
FIGURE 2-3: STANDARD RJ-11 CONNECTION AT TARGET



2.4.2 Target Connection Circuitry

Figure 2-4 shows the interconnections of the PICkit 3 programmer/debugger to the connector on the target board. The diagram also shows the wiring from the connector to a device on the target PC board. A pull-up resistor (usually around 10 kΩ) is recommended to be connected from the VPP/$\overline{\text{MCLR}}$ line to VDD so that the line may be strobed low to reset the device.

FIGURE 2-4: STANDARD CONNECTION TARGET CIRCUITRY



2.4.3 Target Powered

In the following descriptions, only three lines are active and relevant to core debugger operation: pins 1 (VPP/MCLR), 5 (PGC) and 4 (PGD). Pins 2 (VDD) and 3 (VSS) are shown on Figure 2-4 for completeness. PICkit 3 has two configurations for powering the target device: internal debugger and external target power.

The recommended source of power is external and derived from the target application. In this configuration, target VDD is sensed by the debugger to allow level translation for the target low voltage operation. If the debugger does not sense voltage on its VDD line (pin 2 of the interface connector), it will not operate.

2.4.4 Debugger Powered

The internal debugger power is limited to 30 mA. This may be of benefit for very small applications that have the device VDD separated from the rest of the application circuit for independent programming, but is not recommended for general usage as it imposes more current demands from the USB power system derived from the PC.

Not all devices have the AVDD and AVSS lines, but if they are present on the target device, all must be connected to the appropriate levels in order for the debugger to operate. They cannot be left floating.

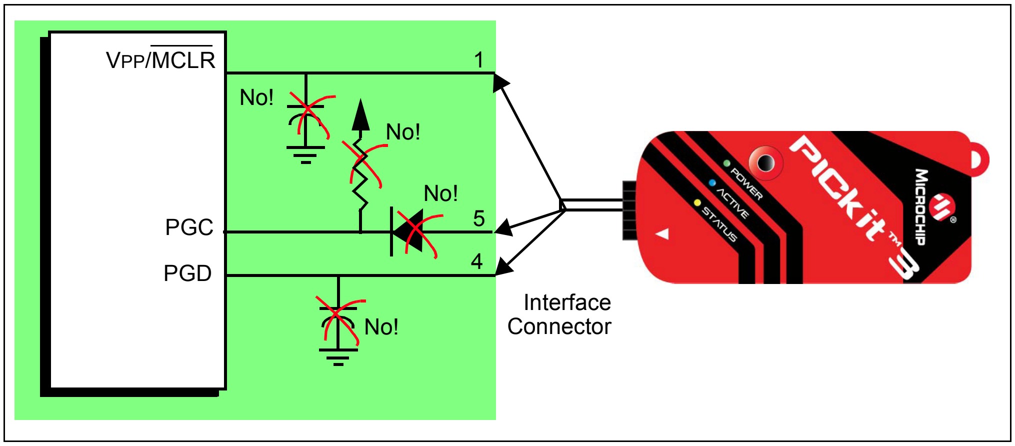
Also, devices with a VCAP line (PIC18FXXJ for example) should be connected to the appropriate capacitor or level.

Note: The interconnection is very simple. Any problems experienced are often caused by other connections or components on these critical lines that interfere with the operation of the PICkit 3 programmer/debugger, as discussed in the following section.

2.4.5 Circuits That Will Prevent the Debugger From Functioning

Figure 2-5 shows the active debugger lines with some components that will prevent the PICkit 3 debugger system from functioning.

FIGURE 2-5: IMPROPER CIRCUIT COMPONENTS



Specifically, these guidelines must be followed:

- Do not use pull-ups on PGC/PGD – they will disrupt the voltage levels, since these lines have 4.7 kΩ pull-down resistors in the debugger.
- Do not use capacitors on PGC/PGD – they will prevent fast transitions on data and clock lines during programming and debug communications.
- Do not use capacitors on $\overline{\text{MCLR}}$ – they will prevent fast transitions of VPP. A simple pull-up resistor is generally sufficient.
- Do not use diodes on PGC/PGD – they will prevent bidirectional communication between the debugger and the target device.

2.5 DEBUGGING

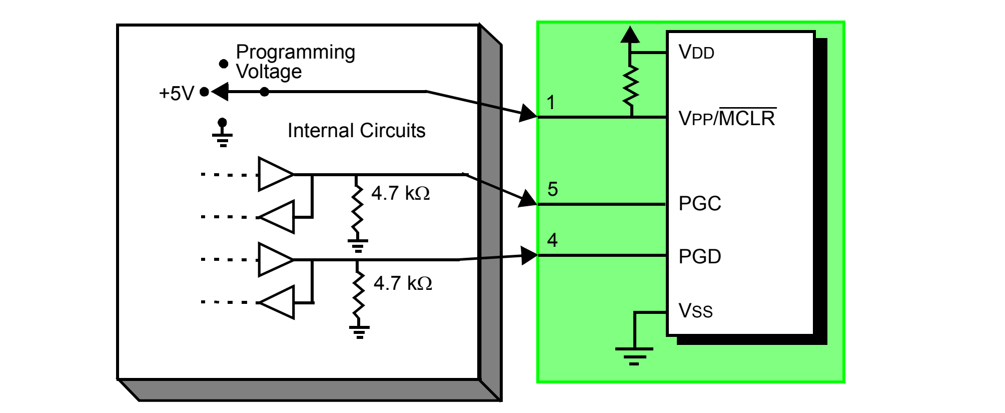
There are two steps to using the PICkit 3 programmer/debugger system as a debugger. The first requires that an application be programmed into the target device (usually with the PICkit 3 itself). The second uses the internal in-circuit debug hardware of the target Flash device to run and test the application program. These two steps are directly related to the MPLAB IDE operations:

1. Program the code into the target and activate special debug functions (see the next section for details).
2. Use the debugger to set breakpoints and run.

If the target device cannot be programmed correctly, the PICkit 3 programmer/debugger will not be able to debug.

Figure 2-6 shows the basic interconnections required for programming. Note that this is the same as Figure 2-4, but for the sake of clarity, the VDD and VSS lines from the debugger are not shown.

FIGURE 2-6: PROPER CONNECTIONS FOR PROGRAMMING



A simplified diagram of some of the internal interface circuitry of the PICkit 3 programmer/debugger is shown. For programming, no clock is needed on the target device, but power must be supplied. When programming, the debugger puts programming levels on VPP/MCLR, sends clock pulses on PGC and serial data via PGD. To verify that the part has been programmed correctly, clocks are sent to PGC and data is read back from PGD. This conforms to the ICSP protocol of the device under development.

2.6 REQUIREMENTS FOR DEBUGGING

To debug (set breakpoints, see registers, etc.) with the PICkit 3 programmer/debugger system, there are critical elements that must be working correctly:

- The debugger must be connected to a PC. It must be powered by the PC via the USB cable, and it must be communicating with the MPLAB IDE software via the USB cable. See **Chapter 3. "Installation"** for details.
- The debugger must be connected as shown to the VPP, PGC and PGD pins of the target device with the modular interface cable (or equivalent). VSS and VDD are also required to be connected between the debugger and target device.
- The target device must have power and a functional, running oscillator. If the target device does not run, for any reason, the PICkit 3 programmer/debugger cannot debug.
- The target device must have its Configuration Words programmed correctly:
  - The oscillator Configuration bits should correspond to RC, XT, etc., depending upon the target design.
  - For some devices, the Watchdog Timer is enabled by default and needs to be disabled.
  - The target device must not have code protection enabled.
  - The target device must not have table read protection enabled.
- PGM (LVP) should be disabled.

Once the above conditions are met, you may proceed to the following:

- Sequence of Operations Leading to Debugging
- Debugging Details

2.6.1 Sequence of Operations Leading to Debugging

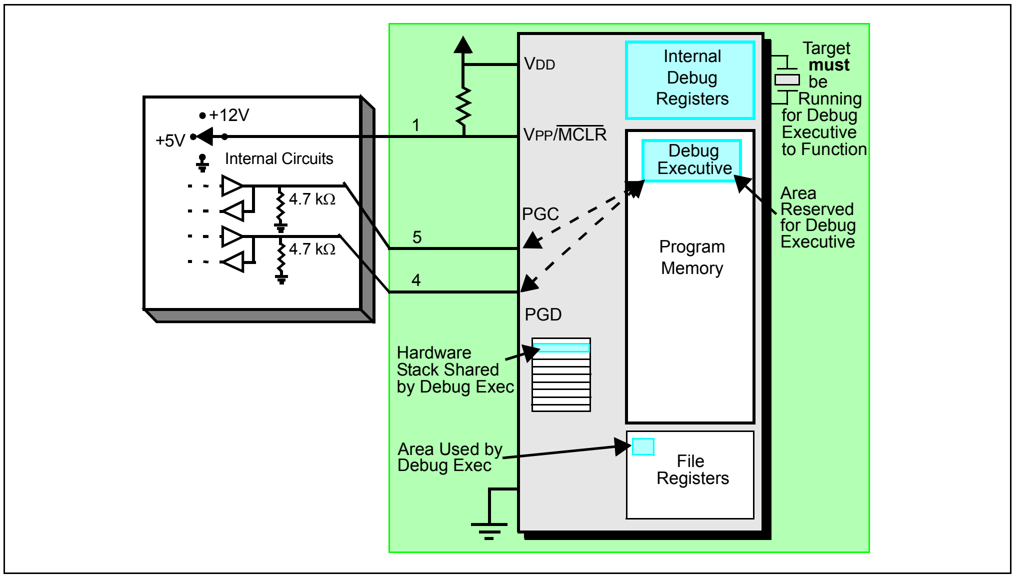
Given that the requirements for debugging (see previous section) are met, these actions can be performed when the PICkit 3 programmer/debugger is set as the current debugger from the MPLAB IDE menu (*Debugger>Select Tool>PICkit 3*):

- The application code is compiled/assembled by selecting *Project>Build Configuration>Debug*.
- When *Debugger>Program* is selected, the application code is programmed into the device's memory via the ICSP protocol as described above.
- A small "debug executive" program is loaded into the high area of program memory of the target device automatically by MPLAB IDE. Since the debug executive must reside in program memory, the application program must not use this reserved space. Some devices have special memory areas dedicated to the debug executive. Check your device data sheet for details.
- Special "in-circuit debug" registers in the target device are enabled. These allow the debug executive to be activated by the debugger.
- The target device is held in Reset by keeping the VPP/$\overline{\text{MCLR}}$ line low.

2.6.2 Debugging Details

Figure 2-7 illustrates the PICkit 3 programmer/debugger system when it is ready for debugging.

FIGURE 2-7: PICkit™ 3 DEBUGGER READY FOR DEBUGGING



Typically, in order to find out if an application program will run correctly, a breakpoint is set early in the program code. When a breakpoint is set from the user interface of MPLAB IDE, the address of the breakpoint is stored in the special internal debug registers of the target device. Commands on PGC and PGD communicate directly to these registers to set the breakpoint address.

Next, the *Debugger>Run* function or the Run icon (forward arrow) is usually pressed from MPLAB IDE. The debugger will then tell the debug executive to run. The target will start from the Reset vector and execute until the Program Counter reaches the breakpoint address previously stored in the internal debug registers.

After the instruction at the breakpoint address is executed, the in-circuit debug mechanism of the target device "fires" and transfers the device's Program Counter to the debug executive (much like an interrupt) and the user's application is effectively halted. The debugger communicates with the debug executive via PGC and PGD, gets the breakpoint status information and sends it back to MPLAB IDE. MPLAB IDE then sends a series of queries to the debugger to get information about the target device, such as file register contents and the state of the CPU. These queries are ultimately performed by the debug executive.

The debug executive runs just like an application in program memory. It uses some locations on the stack for its temporary variables. If the device does not run, for whatever reason, such as no oscillator, a faulty power supply connection, shorts on the target board, etc., then the debug executive cannot communicate to the PICkit 3 programmer/debugger and MPLAB IDE will issue an error message.

Another way to get a breakpoint is to press the MPLAB IDE's **Halt** button (the "pause" symbol to the right of the Run arrow). This toggles the PGC and PGD lines so that the in-circuit debug mechanism of the target device switches the Program Counter from the user's code in program memory to the debug executive. Again, the target application program is effectively halted, and MPLAB IDE uses the debugger communications with the debug executive to interrogate the state of the target device.

2.7 PROGRAMMING

There are three ways to program a device with the PICkit 3 unit:

- Through MPLAB IDE with the PICkit 3 connected to the PC.
- Through PICkit 3 Programmer-To-Go, after setting it up through MPLAB IDE. (See **Chapter 6. "PICkit 3 Programmer-To-Go"** for more information.)
- Through PICkit 3 Programmer Application, a software program that allows you to program devices with PICkit 3 without using MPLAB IDE. (See "*PICkit™ 3 Programmer Application User's Guide"* for instructions.)

Use the PICkit 3 programmer/debugger as a programmer to program an actual (non -ICE/-ICD) device, i.e., a device not on a header board. From the MPLAB IDE menu, select "PICkit 3" from *Programmer>Select Programmer* and compile/assemble your application code with the "Build Configuration" list box on the MPLAB IDE toolbar set to "Release". Also, it may be set by selecting *Project>Build Configuration>Release*.

All debug features are turned off or removed when the debugger is used as a programmer. When using the *Programmer>Program* selection to program a device, MPLAB IDE will disable the in-circuit debug registers so the PICkit 3 programmer/debugger will program only the target application code and the Configuration bits (and EEPROM data, if available and selected) into the target device. The debug executive will not be loaded. As a programmer, the debugger can only toggle the $\overline{\text{MCLR}}$ line to reset and start the target. A breakpoint cannot be set, and register contents cannot be seen or altered.

The PICkit 3 programmer/debugger system programs the target using ICSP. V_{PP}, PGC and PGD lines should be connected as described previously. No clock is required while programming, and all modes of the processor can be programmed, including code protection, Watchdog Timer and table read protection.

2.8 RESOURCES USED BY THE DEBUGGER

For a complete list of resources used by the debugger for your device, please see the online help file in MPLAB IDE for the PICkit 3 programmer/debugger.

Chapter 3. Installation

3.1 INTRODUCTION

How to install the PICkit 3 programmer/debugger system is discussed.

- Installing the Software
- Connecting the Target
- Setting Up the Target Board
- Setting Up MPLAB® IDE

3.2 INSTALLING THE SOFTWARE

To install the MPLAB IDE software, first acquire the latest MPLAB IDE installation executable (`MPxxxxx.exe`, where *xxxxx* represents the version of MPLAB IDE) from either the Microchip web site (www.microchip.com) or the MPLAB IDE CD-ROM (DS51123). Then run the executable and follow the screens to install MPLAB IDE.

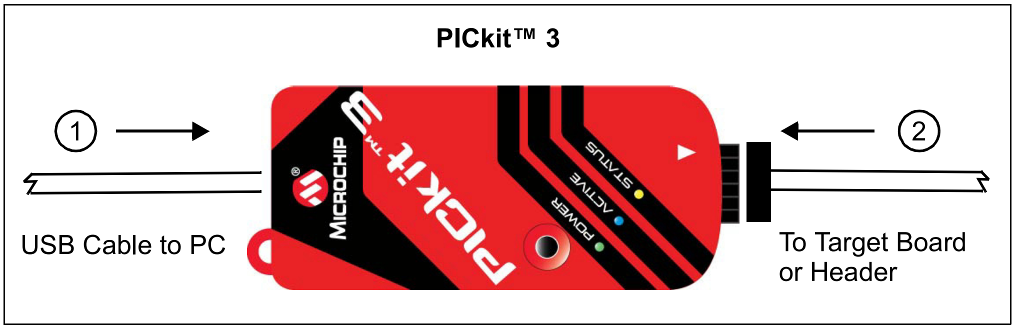
Note: MPLAB IDE v8.20 or greater is required to use the PICkit 3 programmer/debugger.

3.3 CONNECTING THE TARGET

A connection is built-in to select the type of communication with the target. See **Section 2.3 "Debugger to Target Communication"** for more details and a diagram.

1. Plug in the USB/power cable if not already connected.
2. Attach the communication cable(s) between debugger and target if using RJ11 plug or connect directly to a 6-pin inline header.

FIGURE 3-1: INSERT COMMUNICATIONS AND USB/POWER CABLES


PICkit™ 3
1
2
USB Cable to PC
To Target Board or Header
MICROCHIP
PICkit 3
STATUS
ACTIVE
POWER