PICDEM™ FS USB DEMONSTRATION BOARD 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.

Trademarks

Printed on recycled paper.

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

NOTES:

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 the Chapter Name. Items discussed in this chapter include:

- About This Guide
- Warranty Registration
- Recommended Reading
- Troubleshooting
- The Microchip Web Site
- Development Systems Customer Change Notification Service
- Customer Support

ABOUT THIS GUIDE

Document Layout

This document describes how to use the PICDEM FS USB demonstration board as a development tool for creating full-speed USB applications. The manual layout is as follows:

- **Chapter 1: Introduction to the PICDEM™ FS USB Board** describes the hardware of the demonstration board, and how it can be used in creating new USB solutions.
- **Chapter 2: Getting Started with the PICDEM™ FS USB Board** describes how to connect the demonstration board to a host system, and how to install the Demonstration Tool software.
- **Chapter 3: Using the Demo Tool Software** describes how to use the application in both Demo and Bootload modes.
- **Chapter 4: Using the Microchip USB Firmware Framework** refers to the *"MCHPFSUSB Firmware User's Guide"* (DS51679), which describes how to design USB solutions using the reference firmware.
- **Chapter 5: Reconfiguring the PICDEM™ FS USB Hardware** describes how to tailor the demonstration board's hardware to your application.
- **Chapter 6: Troubleshooting** discusses some common questions about using the demonstration board.
- **Appendix A: PICDEM™ FS USB Board Technical Information** provides the schematics and other technical details about the demonstration board.

CONVENTIONS USED IN THIS GUIDE

This manual uses the following documentation conventions:

DOCUMENTATION CONVENTIONS

Description	Represents	Examples
Arial font:		
Italic characters	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 text with right angle bracket	A menu path	<u>*File>Save*</u>
Bold characters	A dialog button	Click **OK**
	A tab	Click the **Power** tab
N'Rnnnn	A number in verilog format, where N is the total number of digits, R is the radix and n is a digit.	4'b0010, 2'hF1
Text in angle brackets < >	A key on the keyboard	Press <Enter>, <F1>
Courier New font:		
Plain Courier New	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 Courier New	A variable argument	*`file`*`.o`, where *`file`* can be any valid filename
Square brackets [ ]	Optional arguments	`mcc18 [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 the PICDEM™ FS USB Demo Board. Other useful documents are listed below. The following documents are available and recommended as supplemental reference resources.

MCHPFSUSB Release Notes

The MCHPFSUSB release notes contain up to date information regarding specific release versions of the MCHPFSUSB USB framework.

MCHPFSUSB Firmware User's Guide (DS51679)

The *“MCHPFSUSB Firmware User’s Guide”* includes important information useful in understanding how the reference USB firmware works. This framework can be found on the Microchip USB design center: http://www.microchip.com/usb.

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 update information and known issues that may not be included in this user’s guide.

PIC18F2445/2550/4445/4550 Device Data Sheet (DS39632)

This is the comprehensive reference for Microchip’s enhanced microcontroller with Full-Speed USB. For users already familiar with the USB protocol, the data sheet provides the basic information needed for designing the hardware and firmware for a Microchip-based USB solution.

USB Specification, Revision 2.0 (USB Implementers Forum, Inc., www.usb.org)

For developers creating a USB application from the ground up, this is the comprehensive reference on the Universal Serial Bus protocol. All features of USB, from physical and electrical specifications to data and communication protocols to device management are defined here. Chapters 5, 8 and 9 are especially useful for those interested in understanding and developing USB peripheral devices.

TROUBLESHOOTING

See **Chapter 6. “Troubleshooting”** for information on common problems.

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 ICE 2000 and MPLAB ICE 4000.
- **In-Circuit Debuggers** – The latest information on the Microchip in-circuit debugger, MPLAB ICD 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 SIM simulator, MPLAB IDE Project Manager and general editing and debugging features.
- **Programmers** – The latest information on Microchip programmers. These include the MPLAB PM3 and PRO MATE® II device programmers and the PICSTART® Plus and PICkit™ 1 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.

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

Chapter 1. Introduction to the PICDEM™ FS USB Board

1.1 INTRODUCTION

The PICDEM FS USB Demonstration Kit is designed as an easy-to-use evaluation platform for the PIC18F2455/2550/4455/4550 family and similar PIC18 USB 2.0 full-speed microcontrollers. The Demonstration Kit provides all of the hardware and software needed to demonstrate and develop a complete USB communication solution. Items discussed in this chapter include:

- PICDEM FS USB Demonstration Kit Contents
- Overview of the PICDEM FS USB Board
- PICDEM FS USB Board Features
- Demonstration Tool Software

1.2 PICDEM FS USB DEMONSTRATION KIT CONTENTS

The Demonstration Kit contains the following items:

1. The PICDEM FS USB demonstration board, preprogrammed with USB bootloader and demonstration firmware.
2. A standard USB cable for use in communicating with the board.
3. The PICDEM FS USB Starter Kit CD-ROM, containing the USB driver, Demo Tool application and development tools.

1.3 OVERVIEW OF THE PICDEM FS USB DEMONSTRATION BOARD

The microcontroller for the PICDEM FS USB board is the PIC18F4550, the superset (largest memory and pin count) device of the PIC18F2455/2550/4455/4550 family. All of these devices implement power-saving nanoWatt Technology, Enhanced Flash program memory, and feature USB modules with the following:

- USB 2.0 compliance
- Full-speed (12 Mbit/s) and low-speed (1.5 Mbit/s) operation
- Support of control, interrupt, bulk and isochronous transfers
- Support of up to 32 endpoints
- 1 Kbyte of dual access RAM for USB
- On-chip features for a true single chip USB implementation, including
  - USB transceiver
  - USB voltage regulator
  - USB pull-up resistors
- Interface for off-chip USB transceiver

The included reference designs and examples use the Microchip USB Firmware Framework. This provides the services for handling lower level USB protocol management, control transfer, USB interrupt handling, hardware register control and management. Reference designs included with the demonstration kit cover Human Interface Devices (HID), the Communication Device Class (CDC) and the Microchip General Purpose USB device class.

The software consists of the Microchip General Purpose USB Windows operating system driver, `mchpusb.sys`, which allows a PC application to communicate directly with the device’s endpoints. A set of user-friendly programming interfaces, the MPUSBAPI Library, is also provided to allow easy application development.

The board comes preprogrammed with a default demo application and is ready for evaluation right out of the box. New programs can be downloaded to the PIC18F4550 microcontroller using a preprogrammed bootloader via the USB interface. A PC-based application, the PICDEM FS USB Demo Tool, facilitates the bootloading process and serves as the host application in the default demonstration.

1.3.1 Benefits of Using the PICDEM FS USB Board

The PICDEM FS USB demonstration board provides two simple advantages:

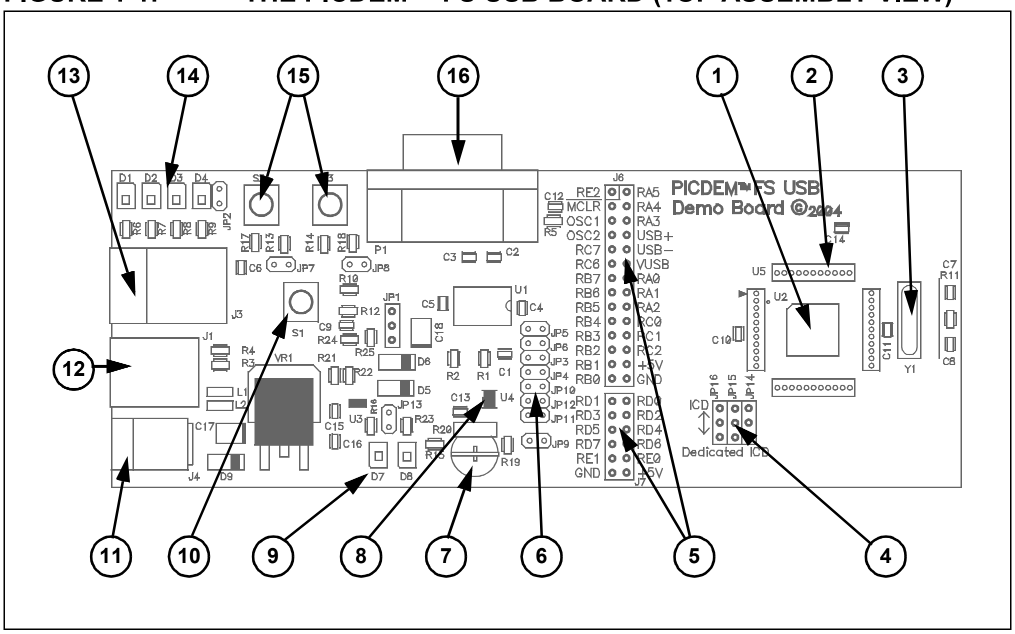
A READY-TO-USE DEMONSTRATION: As delivered, the board is ready for full-speed USB operation; all that is required is the proper driver (included) and the Demo Tool application. Users can also reprogram the board with different applications, included on the Starter Kit CD, to evaluate other USB solutions that can be tailored to their needs.

AN EXPANDABLE PLATFORM: Users can also expand the hardware capabilities of the board through its expansion headers, and even interface it with available PICtail™ demonstration and evaluation daughter boards.

1.4 PICDEM FS USB BOARD HARDWARE FEATURES

The overall layout of the board is shown in Figure 1-1, with a list of the main features following. The more complex features are discussed in detail later in this section.

FIGURE 1-1: THE PICDEM™ FS USB BOARD (TOP ASSEMBLY VIEW)



1. **Microcontroller:** The 44-pin TQFP PIC18F4550 microcontroller (U2) is the heart of the demonstration board, and provides all USB functionality on one chip. Refer to the device data sheet (DS39632) for a complete discussion of the microcontroller and its feature set.
2. **ICE Interface Riser:** The microcontroller is surrounded by a 44-pin pad (U5), arranged as four groups of 11 on each side. These locations can be used to mount a riser for interfacing with Microchip’s MPLAB ICE 2000/4000 emulator system.

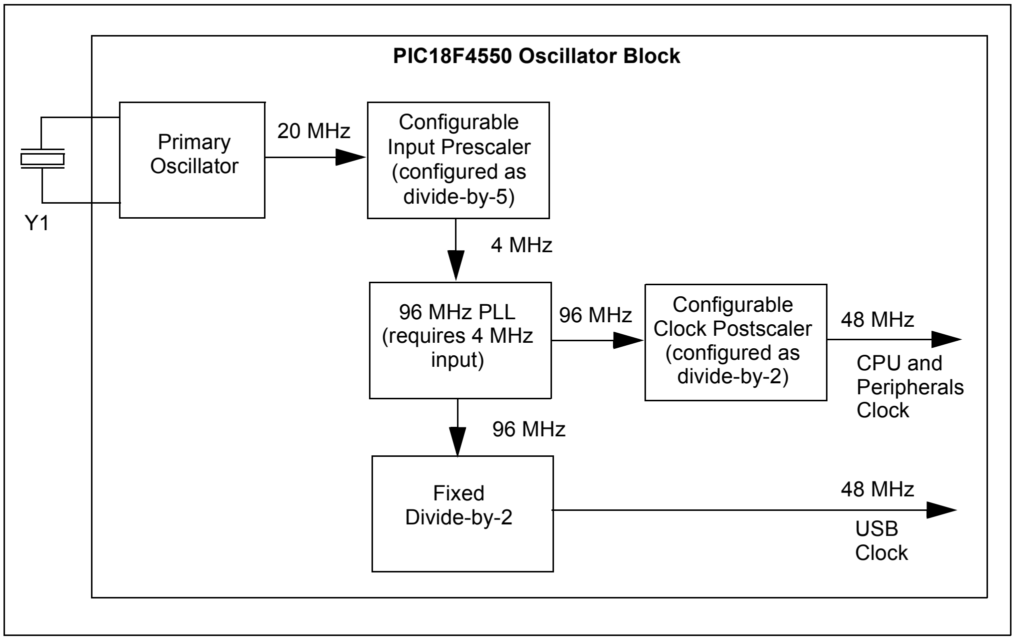
3. **Oscillator:** The demonstration board uses a 20 MHz crystal oscillator (Y1) as the primary clock service. The PIC18F4550 uses this oscillator to generate the necessary clock signals for both the USB Serial Interface Engine (SIE) and the core processor.
4. **ICD Configuration Jumpers:** These three unpopulated jumper positions allow the user to choose either legacy or dedicated ICSP™ and ICD ports for the controller. By default, the board is hard-configured for the legacy port. The configuration and use of these jumpers is detailed in **Section 5.2.4 “Adding In-Circuit Emulation Capability”**.
5. **Expansion and PICtail Daughter Board Headers:** The pads at J6 and J7 are provided for users to install the header and directly access the microcontroller’s I/O port signals. In addition, the 14 even numbered pins of J6 (those on the right side as viewed from the top) serve as the interface for Microchip’s PICtail daughter boards. This allows the PICDEM FS USB board to be used as a test platform and USB communications interface for the PICtail daughter boards.
6. **Configuration Jumpers:** A total of 13 unpopulated jumper positions are provided across the board; these allow users to modify the board by configuring its hardware to suit their needs. By default, all jumpers are bridged and all features are enabled. The configuration and use of these jumpers is detailed in **Section 5.2 “Configuring the Demonstration Board Options”**.
7. **Potentiometer:** The potentiometer (R20) simulates an analog input for the controller. Its real-time value can also displayed by the Demo Tool host software.
8. **Temperature Sensor:** A Microchip TC77 digital temperature sensor (U4) continuously monitors the board’s ambient temperature. Data is transmitted to the controller via a 3-wire SPI interface, and is displayed in real time by the Demo Tool software.
9. **Power LEDs (Green):** These light to show that power is being supplied to the board, and to indicate how the board is being powered. LED D7 indicates that the board is being powered from the bus, while D8 indicates the board is being powered from a separate power supply.
10. **Reset Pushbutton:** This switch (S1) is tied to the $\overline{\text{MCLR}}$ pin of the PIC18F4550 controller; pressing it causes a hard device Reset.
11. **Power Connector:** Power (9 VDC) can be supplied to the board from an external power adapter through a mini barrel jack. Using an external supply is optional, as all examples provided with the demonstration board can use power from the USB cable.
12. **USB Connector:** This is a standard USB series “B” receptacle. The USB port is the primary channel for controlling and communicating with the demonstration board.
13. **ICD Connector:** This 6-wire RJ11 connector provides the standard interface used by Microchip development and demonstration boards for programming and debugging applications, using MPLAB ICD 2 and other development tools.
14. **Status LED Bank:** A bank of four green LEDs is used to show the operational status of the board. Two LEDs (D1 and D2) are used by the application firmware to indicate the status of the USB connection. The other LEDs (D3 and D4) can be defined by the user; they are directly controllable through the Demo Tool software.
15. **User-Defined Pushbuttons:** These two switches (S2 and S3) are provided to simulate digital control inputs. Pressing either button causes its port to read as ‘0’.
16. **RS-232 (DB9F) Port:** A standard D-shell connector, along with a standard level shifter (U1), provides an RS-232 serial connection to the demonstration board.

1.4.1 Oscillator and Operating Frequency

The USB module of the PIC18F4550 requires a specific clock frequency input to operate correctly; specifically, 48 MHz is required when operating in Full-Speed mode, or 6 MHz when operating in Low-Speed mode. The PICDEM FS USB board uses a 20 MHz crystal oscillator as an external clock, and derives the necessary internal clock frequencies from the 96 MHz PLL module. The clock configuration used on the demonstration board is shown in Figure 1-2. A detailed explanation of how to set up the clock configuration is explained in Chapter 2.0 of the PIC18F4550 device data sheet (DS39632).

The PICDEM FS USB board is configured to run in Full-Speed USB mode, generating an internal system clock of 48 MHz (equivalent to 12 MIPS) from the external 20 MHz crystal.

FIGURE 1-2: PIC18F4550 CLOCK SETUP FOR PICDEM™ FS USB



1.4.2 Power

The PICDEM FS USB demonstration board operates at 5V. Power can be drawn directly from the USB bus or from an external power supply. There are no jumpers to select which power source to use. Instead, the power circuitry automatically selects the external power source when both power sources are available. Two LEDs, D7 and D8, are used to indicate the active power source. D7 indicates that the board is operating in Bus Power mode; D8 indicates Self-Power mode (i.e., an external power supply).

> **Note:** There are no power conditions that will cause both D7 and D8 to light at the same time.

Like most USB peripherals, the PICDEM FS USB board can be powered from the 5V available from the USB cable. A minimum of 100 mA is always available on the bus for a device; a maximum of 500 mA can be requested and used if available.

A barrel-type power supply connector (2.5 mm diameter) is also provided to run the board from a 9-18 VDC power supply. A transformer is not supplied in the kit because every example included will run from USB bus power.

A USB host may send a query to a USB device to determine if it is currently self-powered. Without an ability to sense the output of the 5V regulator (VR1), there would be no way to determine the status of the power supply. The PICDEM FS USB board uses two of the microcontroller's I/O pins (PORTA<2:1>) to sense which supply is available: PORTA<2> monitors the regulator, while PORTA<1> senses the USB cable. When a port is read as '1', its corresponding power source is active. When a port is read as '0', its source is disconnected. The combinations of PORTA states are shown in Table 1-1.

For more details on power requirement and management, refer to Microchip application note *AN950, "Power Management for PIC18 USB Microcontrollers with nanoWatt Technology"* (DS00950).

TABLE 1-1: PORTA<2:1> STATE COMBINATIONS AND THEIR MEANINGS

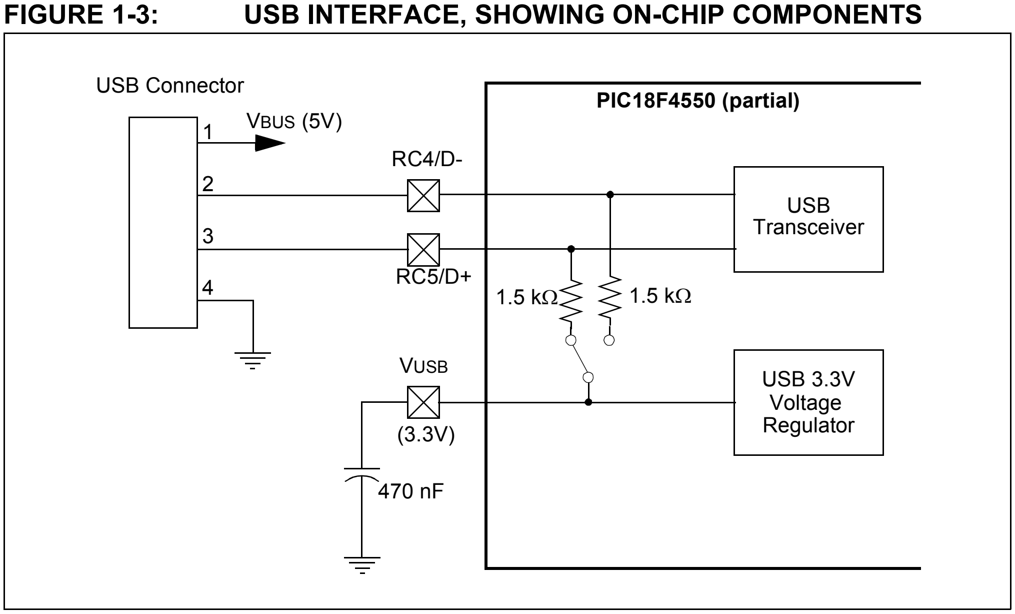
PORTA<1>	PORTA<2>	Status
1	1	USB cable and power supply are both connected; board is self-powered, D8 is lit
1	0	USB cable is attached; board is bus-powered, D7 is lit
0	1	Power supply only is attached; board is self-powered, D8 is lit
0	0	N/A

1.4.3 USB Interface

The PICDEM FS USB board utilizes the on-chip USB voltage regulator, transceivers and pull-up resistors of the PIC18F4550. This helps reduce the number of external components. The USB connection can be electrically detached by disabling the USB module in the firmware. By disabling the USB module in firmware (setting the USBEN bit in the UCON register to '0'), the on-chip USB voltage regulator will also be disabled. This simulates the physical detachment of the USB cable.

Figure 1-3 shows the electrical connection for the USB interface on the board.

FIGURE 1-3: USB INTERFACE, SHOWING ON-CHIP COMPONENTS



1.4.4 RS-232 Interface

The PICDEM FS USB board fully supports RS-232 serial communications, including hardware flow control (RTS/CTS signalling) generated by pins RA2 and RA3 of the microcontroller. One RS-232 female connector and all supporting circuitry are included.

1.4.5 Status LEDs

There are four firmware-controllable LEDs, D1 through D4. An LED is turned on when the corresponding port bit has the value of '1', and off when the port bit is '0'.

The PICDEM FS USB demo board comes preprogrammed with firmware which uses LEDs, D1 and D2, to indicate the current USB device state. Table 1-2 summarizes the LED values in relation to different USB device states. The meaning of each of the USB device states can be found in Chapter 9 of the USB Specification.

> **Note:** In bus-powered applications, flashing D1 and D2 in Suspended mode is not recommended; the current draw is likely to exceed the limits specified in the USB Specification. It is implemented in the original firmware for the demonstration board for demonstration purposes only.

TABLE 1-2: USB DEVICE STATE LED STATUS

USB Device State	LED D1	LED D2
Detached	Off	Off
Attached	On	On
Powered	On	Off
Default	Off	On
Addressed	Blink	Off
Configured	Alternate Flashing	
Suspended	Fast Synchronous Flashing	

1.5 DEMO TOOL APPLICATION SOFTWARE

Included with the demonstration board is the USB Demo Tool software. This simple graphic interface allows users to monitor and control simple board features, and provides the ability to reprogram the PIC18F4550 controller via a bootloader demonstration.

The overall operation of the host software is discussed in **Chapter 3. "Using the Demo Tool Application"**.

Chapter 2. Getting Started with the PICDEM™ FS USB Board

2.1 HIGHLIGHTS

This chapter will cover the following topics:

- Host Computer Requirements
- Installing the Demonstration Board

2.2 HOST COMPUTER REQUIREMENTS

The USB module included on the PIC18F4550 family of microcontrollers is not specific to Windows operating system-based USB platforms. The module may be used to develop USB-based applications intended to interface with Windows operating system, Linux, Macintosh® computers or other types of USB capable host systems. However, the demonstration firmware preprogrammed on the PICDEM FS USB Demo Board is intended to interface with the PICDEM FS USB Demo Tool, which is a Windows operating system-based application. In order to use the preprogrammed demonstration firmware and software, the following hardware and software requirements must be met:

- PC-compatible system
- An available USB port
- CD-ROM drive (for use with the accompanying CD)
- Windows operating system, see supported Windows version in the MCHPFSUSB Release Notes.

2.3 INSTALLING THE DEMONSTRATION BOARD

As a USB device, the demonstration board requires very little effort to install; most of the work is done by the operating system. The three steps required are:

1. Installing the MCHPFSUSB software package
2. Connecting the PICDEM FS USB board
3. Installing the USB driver

2.3.1 Installing the MCHPFSUSB Software Package

The software package that accompanies the PICDEM FS USB board contains all the software required to start using the board immediately. It includes the Demo Tool application, the Microchip general purpose (custom class) USB driver, the reference code projects and related documentation. To begin the installation process, log onto the host PC system with elevated user privileges, insert the CD and run the `MCHPFSUSB_Setup_vX.X.exe` executable (where `vX.X` is the version number of the package). This setup file can also be downloaded from www.microchip.com/usb. This installer will extract files (including the custom class USB driver, the Windows operating system-based demo tool application and a variety of MPLAB IDE based USB projects). The installer will also add start menu shortcuts to the demo tool and related documents. Make sure to read the release notes after installing the package.

> **Note:** Using the default installation directory (`C:\MCHPFSUSB\`) allows the reference projects to retain their original project paths, and requires no additional configuration during setup. If a different installation directory is used, make sure to update the MPLAB IDE project include path directory or the projects may not build correctly. Refer to the *“MCHPFSUSB Firmware User’s Guide”* (DS51679) for more details.

If problems are encountered while trying to install the package, make sure to close background applications before proceeding. If possible, temporarily disable any anti-virus software that the host system is running before installing the package, and re-enable it after the installation is complete. This is also not absolutely necessary, but may be helpful in some system configurations.

> **Note:** It is possible that some organizations may implement a desktop computer policy sufficiently restrictive to prevent the user from loading any software at all. In theory, this can be done with **any** Windows operating system on a network. If this describes your situation, contact your local Information Services provider for assistance in installing this software.

2.3.2 Connecting the PICDEM FS USB Board

To connect the demonstration board :

1. Unbox and unwrap the board, and set it on a non-conductive surface near the host system.
2. Connect the USB cable (supplied in the kit) to an open USB port on the host system or a USB hub connected to the host system, and to the USB connector on the board.
3. Check the board. The green Power LED D7 should light up (if self-powered D8 should light up instead). Windows should automatically detect the new device and launch the Found New Hardware Wizard. For additional assistance, refer to **Chapter 6. “Troubleshooting”**.

2.3.3 Installing the USB Device Driver

Not all USB devices require user-provided device drivers. Some types of USB-related drivers are provided by and distributed along with the operating system. For example, modern versions of Windows, Linux, and Macintosh operating systems are distributed with Human Interface Device (HID) class USB drivers. Upon detecting the connection of a HID class USB device, such as a mouse, the operating system automatically uses the built-in drivers, and no user-provided driver installation is required.

The PICDEM FS USB Demo Board comes preprogrammed with custom device class firmware, which relies on the Microchip general purpose USB driver. Since this driver is not provided with the operating system, it is necessary to manually install the driver before the PICDEM FS USB Demo Tool can be used.

When the Found New Hardware Wizard appears after plugging in the device, Windows will be looking for a `*.inf` file with a USB Vendor ID (VID) and Product ID (PID) matching the VID and PID programmed into the USB firmware. When the wizard appears, use the “Browse” feature to manually point it to the `.inf` file located in the MCHPUSB Driver Release directory, as indicated in the MCHPFSUSB release notes. In MCHPFSUSB release version 1.2, the `.inf` file will be found in the `C:\MCHPFSUSB\Pc\MCHPUSB Driver\Release` directory.

The `.inf` file is a short, plain text installation instructions file. This file may be edited in text editors, such as Notepad. Among other things, the `.inf` file lets the Windows operating system know what driver binary files (ex: `mchpusb.sys`) it should use with the new hardware, what text strings should appear in the device manager describing the device and what (if any) special registry entries should be made for the new hardware.

After pointing the Found New Hardware Wizard to the `.inf` file in the MCHPUSB Driver Release directory, the rest of the driver installation process should be automatic. After the wizard completes, the new device should be visible in the device manager, and the hardware should be ready to be used with the PICDEM FS USB Demo Tool application. Figure 2-1 shows an example of how the device may appear in the device manager after driver installation. The exact appearance, text descriptions and icon can be modified by editing the `.inf` file. The exact appearance may differ between different versions of the `.inf` file included in different versions of the MCHPFSUSB framework and with different operating systems. Technical details describing how `.inf` files work, and how to edit them, can be found in the Microsoft Developer Network (MSDN) documentation.

> **Note 1:** The PICDEM FS USB demonstration board is configured in firmware as two devices: a Demo application and a Bootload application. The USB driver will install when each application is launched the first time.
>
> **2:** After the drivers are installed, if the device is plugged into a different USB port on the same system, the drivers may have to be reinstalled again. This excess driver (re)installation can potentially be avoided by using a USB "serial number" string descriptor in the device firmware.
>
> **3:** On most systems, elevated user privileges are required to install hardware device drivers. If problems are encountered during driver installation, make sure to use an account with high enough privileges.

FIGURE 2-1: DEVICE MANAGER AFTER DRIVER INSTALLATION

2.3.4 Confirming Operation

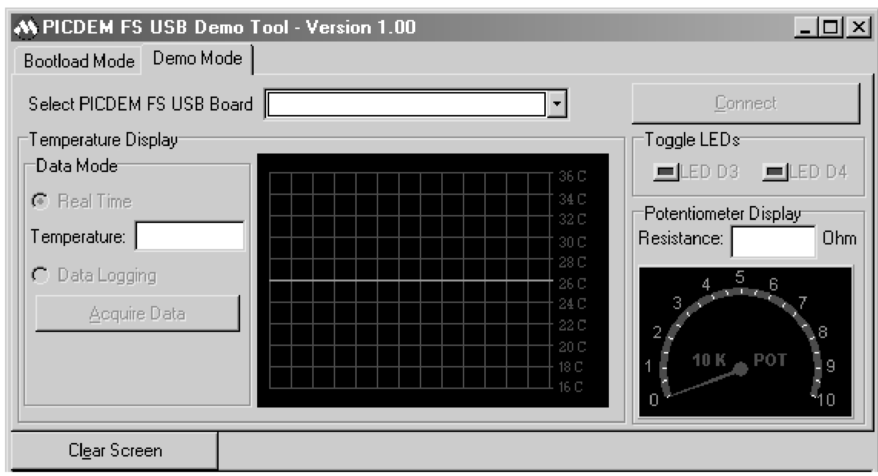
If Power LED D7 is lit (or D8, if an external power supply is being used) and the driver installation completed successfully, it can generally be assumed that the demonstration board's hardware is working correctly. However, it may be useful to verify that the board can actually communicate with the host system using the Demo Tool application.

To do this:

1. Launch the Demo Tool application by clicking on the "PICDEM FS USB Demo Tool" icon from the Start menu.
2. Select the **Demo Mode** tab to access the Demo Mode controls (Figure 2-2).
3. In the "Select PICDEM FS USB Board" dropdown list, select "PICDEM FS USB 0 (Demo)".
4. Click the **Connect** button. A temperature number should appear in the "Real Time Temperature" window almost immediately.

If an appropriate temperature appears, the demonstration board is ready for your use.

FIGURE 2-2: THE DEMO MODE WINDOW (PARTIAL VIEW)



NOTES:

Chapter 3. Using the Demo Tool Application

3.1 HIGHLIGHTS

The items discussed in this chapter are:

- Software Overview
- Starting the Program
- Demo mode
- Bootload mode

3.2 SOFTWARE OVERVIEW

The PICDEM FS USB Demo Tool (more simply, PDFSUSB or the "Demo Tool") is a Windows operating system-based software application designed to be used with the PICDEM FS USB board for evaluating some of Microchip's full-speed USB solutions. Using this software, you can evaluate USB features and performance offered by the PIC18F4550 microcontroller.

There are two modes available: Demo mode and Bootload mode. Demo mode demonstrates communication to and from the host PC through USB in a typical embedded system setup. The application shown includes data logging, real-time sensing, and how to use the PC to control peripheral components. Bootload mode allows designers to download and evaluate a different firmware program without using an additional external programmer.

The Demo and Bootloader modes are two separate firmware applications. Both programs are totally separate and do not share any USB functions or descriptors. The bootloader code is write-protected and cannot be overwritten by firmware. An external programmer, such as MPLAB ICD 2, is required to erase the bootloader. The two programs also have different USB product IDs (but the same Vendor ID). For reference, the product IDs are:

- Bootload: VID 0x04D8, PID 0x000B
- Demo: VID 0x04D8, PID 0x000C

Note:	Because the demo and bootloader are two different applications, they are treated as two devices. The Microchip USB driver will attempt to install when each of the applications is launched for the first time. In the procedure described for the previous chapter, the USB driver for the demo application was installed. The driver for the bootload application will begin installation on its first launch.

3.3 STARTING THE PROGRAM

The demo board is shipped from the factory with the default demonstration firmware program and the bootloader. In the case that the demonstration program has been overwritten, you can restore the firmware by using the bootloader, or an external programmer, such as MPLAB ICD 2.

To run the Demo Tool application, select Programs > Microchip > MCHPFSUSB vX.X > PICDEM FS USB Demo Tool from the Start menu. Alternatively, double-click on the PDFSUSB icon or shortcut. You will see the Demo Tool window (Figure 3-1). By default, the application launches in Bootload mode. To switch to Demo mode, click on the **Demo Mode** tab (Figure 3-2).

FIGURE 3-1: THE DEMO TOOL WINDOW, BOOTLOAD MODE

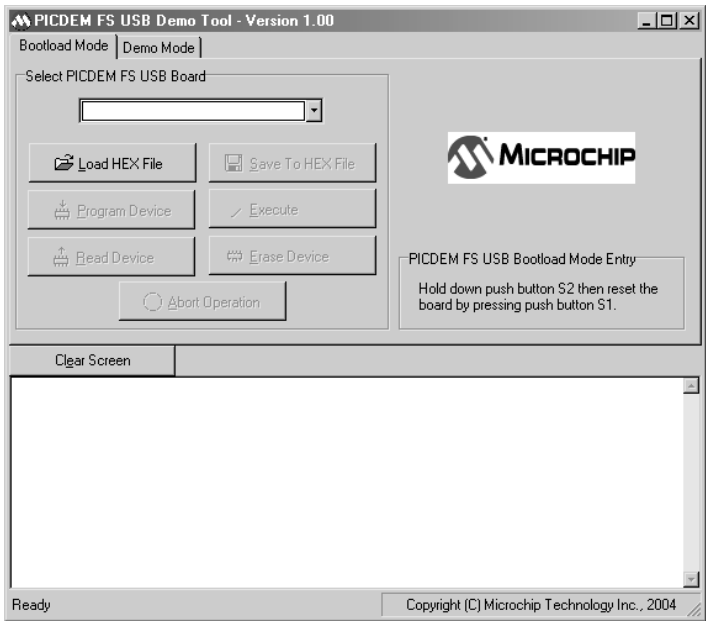
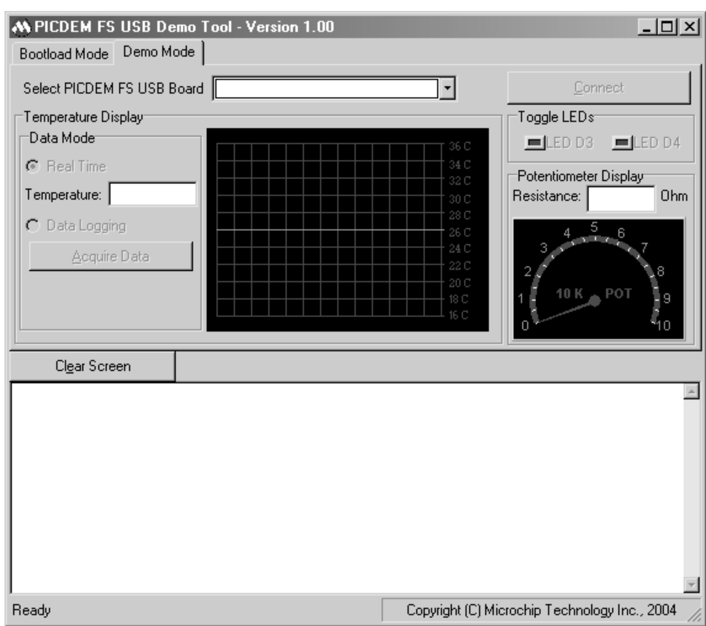


FIGURE 3-2: THE DEMO TOOL WINDOW, DEMO MODE



3.4 DEMO MODE

The Demo mode provides a simple interface between the board and the host system to demonstrate USB connectivity. To start using the application, select “PICDEM FS USB 0 (Demo)” from the “Select PICDEM FS USB Board” dropdown box. Click the “Connect” button in the upper right hand corner. When the Demo Tool application successfully connects to the board, the message “USB Demo Firmware Version x.x” will appear on the left side of the status bar at the bottom of the window. There are three interactive features in this mode.

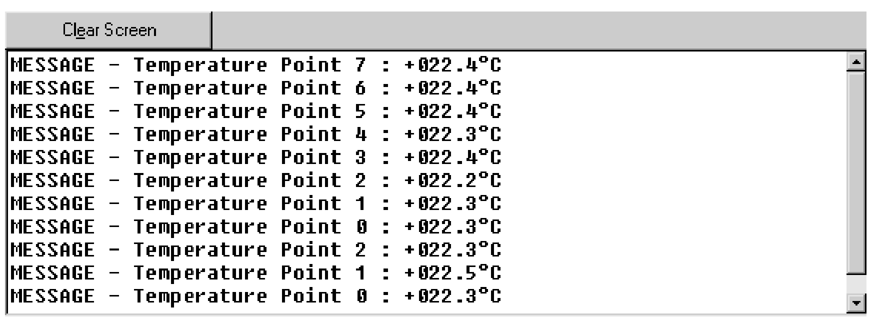
Temperature Display

There are two temperature modes: Real Time and Data Logging. In Real-Time mode, temperature data is streamed from the board to the PC continuously. Both the temperature graph and the temperature value are updated periodically. All data is displayed in centigrade unit with the graph display range of 16°C to 36°C. The easiest way to change the temperature is to lightly blow air (warm or cold) on the sensor (U4).

In the Data Logging mode, temperature data is not sent to the host continuously. Instead, temperature data is sampled every second and stored in the data memory buffer until the “Acquire Data” button is clicked. At that point, the firmware sends everything in the buffer to the PC, then empties the buffer. Data can be acquired at any time; if the time elapsed since the last acquisition is less than 30 seconds, then a smaller set of data points will be acquired and displayed in the message window. Clicking on “Acquire Data” too rapidly may return a “WARNING-No Data Acquired” message; this only means that the controller has not had an opportunity to collect a new temperature sample. All data points are displayed at one time; the temperature graph is also updated at that time. Clicking on “Clear Screen” clears the message window.

The firmware can store up to 30 data points; therefore, it takes 30 seconds to fill the data memory buffer. When the buffer becomes full, the firmware replaces the oldest data point with a newer data value.

FIGURE 3-3: MESSAGE WINDOW IN DATA LOGGING MODE



Toggle LEDs

The Toggle LEDs box contains two buttons used to control the status of LED D3 and D4 on the demonstration board. The corresponding LED is turned on when a button is pressed and off when the same button is pressed again.

Potentiometer Display

The Potentiometer Display reflects the current value of the potentiometer on the demonstration board. The value ranges from 0 to 10 kΩ. Turning the potentiometer on the board causes the analog dial gauge and digital display to change accordingly in real time.

Ending Demo Mode

To stop the demonstration program, click “Disconnect” or exit the application. Clicking on the **Bootload Mode** tab also terminates Demo mode.

3.5 BOOTLOAD MODE

The Bootload mode provides a simple means of reprogramming the on-board microcontroller without using an external programming device. The PICDEM FS USB board is preprogrammed with a bootload program that allows designers to download and test new code through the Demo Tool application.

This section describes the Bootload mode, as well as a brief overview of the architecture of the bootload program. An understanding on how the bootload firmware coexists with the user firmware is important in developing a successful end application.

3.5.1 Bootload Mode Entry

To start using the application, press and hold S2 while resetting the board (pressing and releasing S1). The entry condition on the PICDEM FS USB board and the provided firmware is determined by the status of the switch button, S2, which is checked once after each Reset. If the button is held down during a Reset, the microcontroller enters the Bootload mode; otherwise, it starts executing user code from address 0x0800.

> **Note:** Even if the Demo Tool application is running in Bootload mode, a simple reset of the board (pressing S1) will not cause the board itself to enter Bootload mode.

After the Reset, select "PICDEM FS USB 0 (Boot)" from the "Select PICDEM FS USB Board" dropdown box. When the Demo Tool application successfully connects to the board, the message "USB Bootload Firmware Version x.x" will appear on the left side of the status bar at the bottom of the window.

3.5.2 Memory Organization

Program Memory Usage

Figure 3-4 shows the memory map of the PIC18F4550. The first 2,048 bytes of program memory are reserved as the boot block, which is utilized almost entirely for the bootloader firmware. The bootloader is a self-contained program with its own USB driver firmware, USB descriptor set, bootload command interpreter and bootload function handlers. The USB driver firmware used with the bootloader is a modified version of the Microchip USB firmware framework library, and incorporates a smaller set of features and more static design.

The boot block can be write-protected to prevent accidental overwriting of the boot program. The default setting from factory has the boot block write-protect option enabled.

Remapped Vectors

Since the hardware Reset and interrupt vectors lie within the boot area and cannot be edited if the block is write-protected, they are remapped through software to the nearest parallel location outside the boot block: 0x0800 for Reset, 0x0808 for the high-priority interrupt vector and 0x0818 for the low-priority interrupt vector. Remapping is simply a `GOTO` instruction for interrupts. Users should note that an additional latency of two instruction cycles is required to handle interrupts.

Memory Spaces

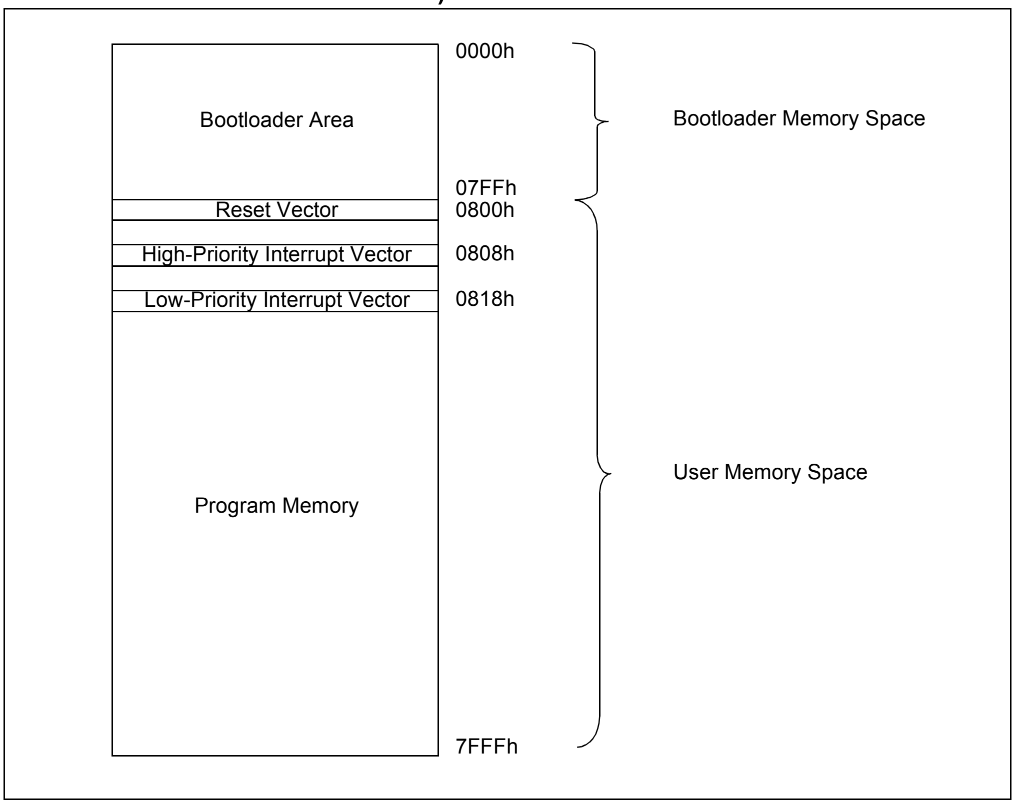
There are four memory spaces that the bootloader can access and program:

- Program Memory (0x0800-0x7FFF)
- User ID Memory (0x200000-0x200007)
- EEPROM Memory (0x0F0000-0x0F00FF)
- Configuration Memory (0x300000-0x30000D)

The bootload also reads and displays the read-only Device ID Words at 0x3FFFFE and 0x3FFFFF.

Note:	The data EEPROM is actually located in a different memory space than shown here. The address range given is remapped by the bootloader to the proper memory when data EEPROM access is required.

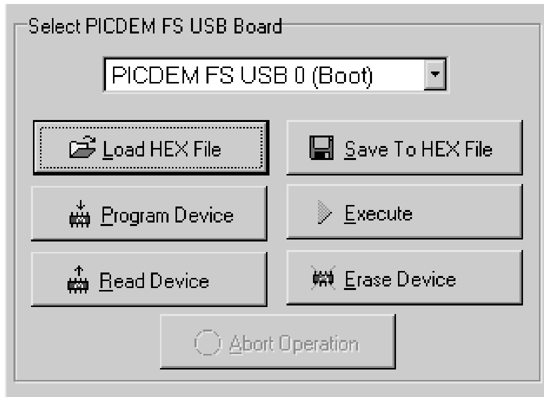
FIGURE 3-4: PIC18F4550 PROGRAM MEMORY MAP (BOOTLOADER IMPLEMENTED)



3.5.3 Using the Bootloader

Once the board has been selected in the dropdown list, the bootloader control buttons become active (Figure 3-5).

FIGURE 3-5: BOOTLOAD MODE CONTROLS



- **Load HEX File:** This loads a hex file into the memory buffer and displays the content in the message window. The data loaded can be used to program a target device. A valid hex file must conform to the Intel® HEX 32 format. If an invalid file is selected, a warning message, "WARNING - No HEX file data.", is displayed. If device configuration data is present, the bootloader firmware will check it for configuration conflicts that might disable the board. These are discussed in more detail in **Section 3.5.4 "Consideration When Using the Bootloader"**. Each project's hex file, MCHPUSB.hex, can be found in the '_output' directory (i.e., `C:\MCHPFSUSB\fw\Cdc\_output` and `C:\MCHPFSUSB\fw\Hid\Mouse\_output`).
- **Read Device:** The read device function reads the entire memory range into the memory buffer and displays the content in the message window. Program memory, EEPROM, the User ID and configuration data are all read. A successful read operation is indicated by the message "MESSAGE - Read Completed" in the Message window. If the operation failed, a warning, "WARNING - Failed to read", is shown.
- **Erase Device:** The erase device function erases the user program memory space only (0x0800 to 0x7FFF). It does not erase the EEPROM, User ID or configuration data.
- **Execute:** The execute function sends a Reset command to the bootloader firmware, causing a microcontroller Reset. If S2 is not pressed during this time, the user code in program memory will start executing; otherwise, the firmware will re-enter the Bootload mode. If the operation failed, perform a hard Reset on the demo board by pressing S1.
- **Save To HEX File:** The save to hex file function saves the data contained in the memory buffer to a file, whether it was loaded from the read device or load hex file functions. It only saves the data type present in the memory buffer. If a hex file is loaded and does not have a particular type of memory (i.e., EEPROM), that data section will not be saved in the output file. All memory sections are always loaded after a read device function.

- **Program Device:** The program device function programs the target device with the data loaded in the memory buffer. Only the type of memory present in the memory buffer will be programmed. The function automatically erases the program memory (0x0800 to 0x7FFF) and user ID memory (0x200000 to 0x200007) before writing new data to these locations. Memory contents are verified after the write operation.
- **Abort Operation:** The abort operation function is enabled when programming, reading or erasing the device. This function causes the current operation to terminate.

3.5.4 Consideration When Using the Bootloader

The PIC18F4550 microcontroller for the PICDEM FS USB board has a specific configuration setting that is necessary for the bootload program to function. The USB voltage regulator and device oscillator settings are both critical and cannot be changed. Other settings, such as code protection, WDT and LVP, are less critical, but may cause irreversible side effects. The default configuration values are shown in Table 3-1.

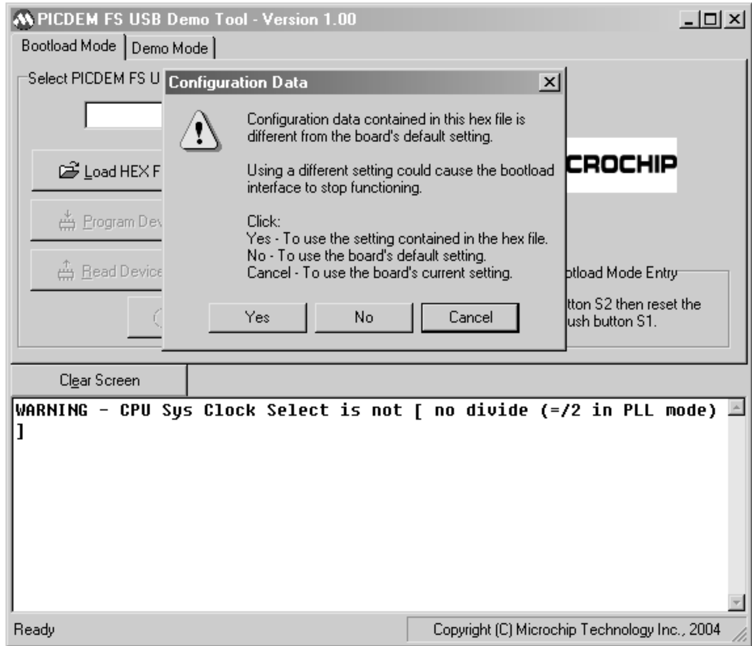
The configuration data contained in a hex file may violate the restrictions on the critical configuration settings described above. Should this happen, the Demo Tool will display a warning dialog box (Figure 3-6). This will also be accompanied by a brief text description of the configuration conflict. Users have the option to accept the new configuration settings, use the board's factory configuration settings or maintain the current configuration setting.

TABLE 3-1: DEFAULT CONFIGURATION WORD VALUES FOR THE PICDEM™ FS USB BOARD

Address	Register	Value	Comment
0x300000	CONFIG1L	0x24	Clock configuration
0x300001	CONFIG1H	0x0E	Clock configuration
0x300002	CONFIG2L	0x3F	BOR, PWRT, USB voltage regulator
0x300003	CONFIG2H	0x1E	WDT configuration
0x300005	CONFIG3H	0x81	$\overline{\text{MCLR}}$, CCP2, A/D configuration
0x300006	CONFIG4L	0x81	Core microcontroller configuration
0x300008	CONFIG5L	0x0F	Code-protect (program memory)
0x300009	CONFIG5H	0xC0	Code-protect (boot block/EEPROM)
0x30000A	CONFIG6L	0x0F	Write-protect (program memory)
0x30000B	CONFIG6H	0xA0	Write-protect (boot block/EEPROM)
0x30000C	CONFIG7L	0x0F	Table read-protect (program memory)
0x30000D	CONFIG7H	0x40	Table read-protect (boot block/EEPROM)

Note: Only implemented Configuration Words are listed.

FIGURE 3-6: CONFIGURATION CONFLICT WARNING DIALOG WITH TYPICAL DIAGNOSTIC MESSAGE



When a file is loaded, either from a device or a file, the bootloader demo software also parses the file to determine if code is available for the different memory areas. If a device is programmed with that file, only those memory areas with code present are programmed; the other memory areas are left unchanged. For example, a hex file that is missing data for the data EEPROM or Configuration Words will only program the target device's program memory and user ID spaces; the existing configuration and stored EEPROM information will be preserved.

3.5.5 Writing Application Code with the Bootloader

The bootloader operates as a separate entity, which means that an application can be developed with very little concern about what the bootloader is doing. This is as it should be; the bootloader should be dormant code until an event initiates its operation. Ideally, bootloader code should never be running during an application's intended normal operation.

When developing an application with a resident bootloader, some basic principles must be kept in mind:

3.5.5.1 WRITING IN ASSEMBLY

When writing in assembly, the boot block and new vectors must be considered. For modular code, this is usually just a matter of changing the linker script file for the project. An example is shown in Example 3-1. If an absolute address is assigned to a code section, the address must point somewhere above the boot block.

For those who write absolute assembly, all that is necessary is to remember that the new Reset vector is at 800h, and the interrupt vectors are at 808h and 818h. Except for the bootloader, no code should reside in the boot block area.

EXAMPLE 3-1: ASSEMBLY LINKER SCRIPT FOR USE WITH BOOTLOADER

```
// Sample linker command file for 18F4550 with Bootloader

LIBPATH .

CODEPAGE   NAME=boot        START=0x0        END=0x7FF     PROTECTED
CODEPAGE   NAME=page        START=0x800      END=0x7FFF
CODEPAGE   NAME=idlocs      START=0x200000   END=0x200007  PROTECTED
CODEPAGE   NAME=config      START=0x300000   END=0x30000D  PROTECTED
CODEPAGE   NAME=devid       START=0x3FFFFE   END=0x3FFFFF  PROTECTED
CODEPAGE   NAME=eedata      START=0xF00000   END=0xF000FF  PROTECTED

ACCESSBANK NAME=accessram   START=0x0        END=0x5F
DATABANK   NAME=gpr0        START=0x60       END=0xFF
DATABANK   NAME=gpr1        START=0x100      END=0x1FF
DATABANK   NAME=gpr2        START=0x200      END=0x2FF
DATABANK   NAME=gpr3        START=0x300      END=0x3FF
DATABANK   NAME=usb4        START=0x400      END=0x4FF     PROTECTED
DATABANK   NAME=usb5        START=0x500      END=0x5FF     PROTECTED
DATABANK   NAME=usb6        START=0x600      END=0x6FF     PROTECTED
DATABANK   NAME=usb7        START=0x700      END=0x7FF     PROTECTED
ACCESSBANK NAME=accesssfr   START=0xF60      END=0xFFF     PROTECTED

SECTION    NAME=CONFIG      ROM=config
```

3.5.5.2 WRITING IN C

When using the MPLAB C18 C compiler to develop firmware for an application, the user has the choice to either rebuild the standard start-up object (`c018.o` or `c018i.o`) with the new Reset vector, or to insert the extra code shown in Example 3-2. The latter method is recommended.

Like modular assembly, the linker file must be changed to incorporate the protected boot block and new vectors. An example is shown in Example 3-3.

For users of other compilers, check with the compiler's software user guide to determine how to change the start-up code and vectors.

EXAMPLE 3-2: RESET VECTOR INSERT FOR APPLICATIONS IN C

```
extern void _startup (void);
// See c018i.c in your C18 compiler directory
#pragma code _RESET_INTERRUPT_VECTOR = 0x000800
void _reset (void)
{
      _asm goto _startup _endasm
}
#pragma code
```

EXAMPLE 3-3: C18 LINKER SCRIPT FOR USE WITH BOOTLOADER

```
// Sample linker command file for 18F4550 with Bootloader

LIBPATH .

FILES c018i.o
FILES clib.lib
FILES p18f4550.lib

CODEPAGE   NAME=boot         START=0x0         END=0x7FF     PROTECTED
CODEPAGE   NAME=vectors      START=0x800       END=0x829     PROTECTED
CODEPAGE   NAME=page         START=0x82A       END=0x7FFF
CODEPAGE   NAME=idlocs       START=0x200000    END=0x200007  PROTECTED
CODEPAGE   NAME=config       START=0x300000    END=0x30000D  PROTECTED
CODEPAGE   NAME=devid        START=0x3FFFFE    END=0x3FFFFF  PROTECTED
CODEPAGE   NAME=eedata       START=0xF00000    END=0xF000FF  PROTECTED

ACCESSBANK NAME=accessram    START=0x0         END=0x5F
DATABANK   NAME=gpr0         START=0x60        END=0xFF
DATABANK   NAME=gpr1         START=0x100       END=0x1FF
DATABANK   NAME=gpr2         START=0x200       END=0x2FF
DATABANK   NAME=gpr3         START=0x300       END=0x3FF
DATABANK   NAME=usb4         START=0x400       END=0x4FF     PROTECTED
DATABANK   NAME=usb5         START=0x500       END=0x5FF     PROTECTED
DATABANK   NAME=usb6         START=0x600       END=0x6FF     PROTECTED
DATABANK   NAME=usb7         START=0x700       END=0x7FF     PROTECTED
ACCESSBANK NAME=accesssfr    START=0xF60       END=0xFFF     PROTECTED

SECTION    NAME=CONFIG       ROM=config

STACK SIZE=0x100             RAM=gpr3
```

Chapter 4. Using the Microchip USB Firmware Framework

4.1 HIGHLIGHTS

The items discussed in this chapter are:

- Overview of the Framework

4.2 OVERVIEW OF THE FRAMEWORK

The example code preprogrammed on the PICDEM FS USB Demo Board was created using the Microchip USB framework known as MCHPFSUSB. Refer to the *"MCHPFSUSB Firmware User's Guide"* (DS51679) and the accompanying release notes for a description of the framework. This framework can be found on the Microchip USB design center: http://www.microchip.com/usb. The framework includes a number of MPLAB IDE based USB firmware projects which can be used as a starting point for developing new USB applications.

NOTES:

Chapter 5. Reconfiguring the PICDEM™ FS USB Hardware

5.1 HIGHLIGHTS

This chapter covers the following:

- Configuring the Demonstration Board Options
- Restoring the PICDEM™ FS USB Firmware

5.2 CONFIGURING THE DEMONSTRATION BOARD OPTIONS

The PICDEM FS USB board can be configured to enable or disable its various hardware features. A total of 16 jumper locations are controlled in various places around the board. As shipped from the factory, all of the locations are bridged by circuit traces and all of the features are enabled. To change this, the user will need to cut the traces, and install pins and a block jumper. Afterwards, the features can be enabled or disabled easily by installing or removing the jumper.

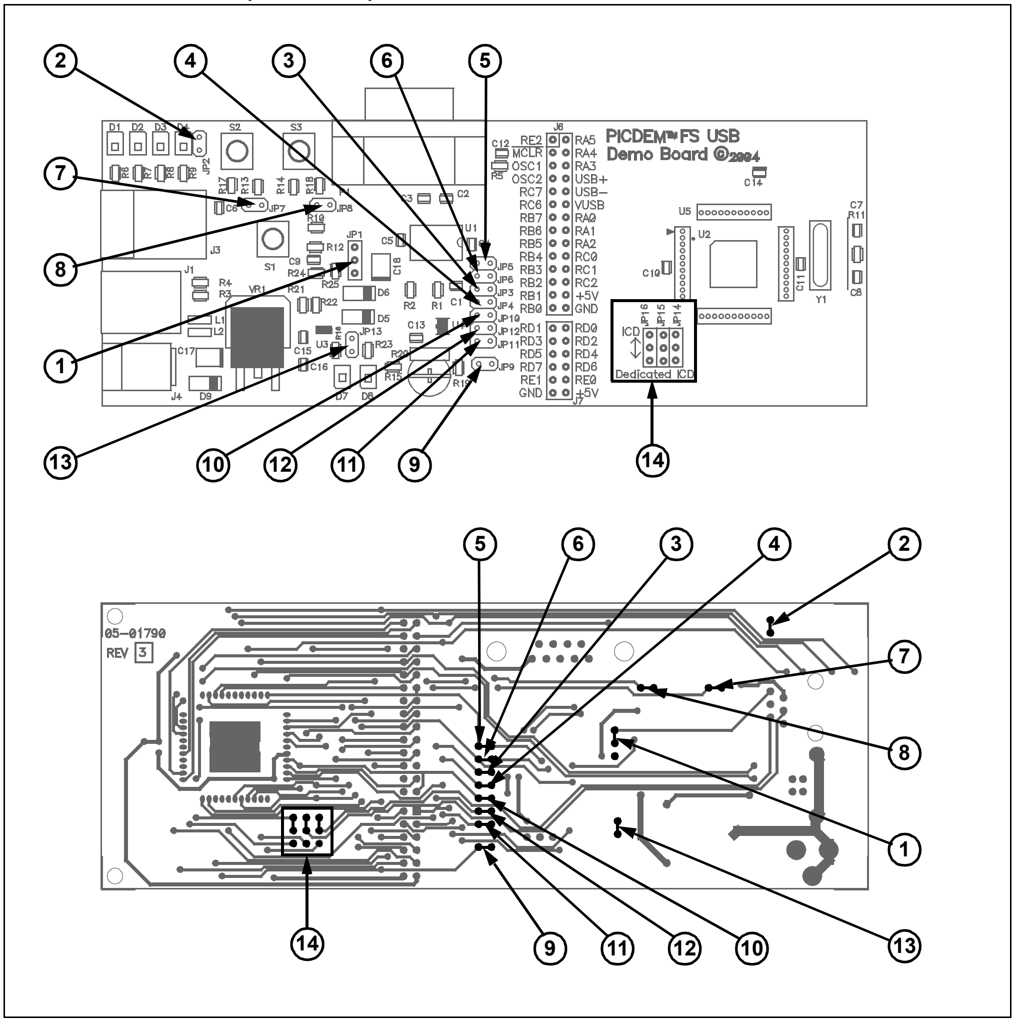
In some instances, a single function (such as the digital temperature sensor) is connected to the rest of the board through more than one jumper. This allows selective tailoring of the controller's I/O ports to any application that the user may develop. Specific cases are discussed in the following sections.

The functions of the jumpers are listed in Table 5-1; their locations are shown in Figure 5-1.

TABLE 5-1: JUMPER DESCRIPTION

Number	Board ID(s)	Type	Function
1	JP1	2-way	Selects user-controlled Reset (S1) or microcontroller disable for external emulation
2	JP2	Bridge	Status LED bank (D1 through D4)
3	JP3	Bridge	USART Receive (microcontroller's perspective)
4	JP4	Bridge	USART Flow Control (RTS)
5	JP5	Bridge	USART Transmit (microcontroller's perspective)
6	JP6	Bridge	USART Flow Control (CTS)
7	JP7	Bridge	S2 (user-defined switch 1)
8	JP8	Bridge	S3 (user-defined switch 2)
9	JP9	Bridge	R20 (potentiometer)
10	JP10	Bridge	U4 (temperature sensor)
11	JP11	Bridge	
12	JP12	Bridge	
13	JP13	Bridge	D7 (USB bus power LED)
14	JP14, JP15, JP16	2-way	ICD Port Configuration

FIGURE 5-1: JUMPER LOCATIONS (TOP) AND TRACE LOCATIONS (BOTTOM) ON THE DEMONSTRATION BOARD



5.2.1 Serial Port Configuration

The RS-232 serial port on the demonstration board incorporates hardware flow control with CTS and RTS for applications that require these control signalizes makes the port useful for developing a wide range of serial to USB translators. For those applications that do not require hardware flow control, or use software flow control instead, the hardware flow control feature can be selectively disabled by cutting the traces at JP4 and JP6.

Note that the CTS signal and the self-power sense signal share a single controller port (RA2). These two functions cannot be used concurrently. The CTS is an output signal from the microcontroller, while the self-power sense signal is an input signal. The port must be configured either as an input or an output to use each mode. JP6 is provided to explicitly disable the CTS signal.

5.2.2 Disabling the Temperature Sensor

For the TC77 temperature sensor, each of the three lines that it uses to communicate with the controller are bridged with a separate jumper. Removing all three jumpers (JP10, JP11 and JP12) disables the sensor's function, and makes all three controller ports available to the user.

5.2.3 ICSP/ICD Configuration

Using the MPLAB IDE and MPLAB ICD 2, users can reprogram the board's microcontroller using In-Circuit Serial Programming™ (ICSP™), and debug firmware code using In-Circuit Debugger. The RJ-11 receptacle (J3), also known as the ICSP/ICD connector, is the standard MPLAB ICD 2 interface found on most Microchip development boards.

Most PIC18 microcontrollers only have one legacy ICSP/ICD port, which shares I/O pins RB6 and RB7. When used, the legacy port prevents applications from using these pins as normal I/O ports. The 44-pin TQFP version of the PIC18F4550 is the only variant of the full-speed USB family of devices to have a second ICSP/ICD port on pins not used for I/O. This dedicated port allows RB6 and RB7 to be utilized by the user's application.

The PICDEM FS USB board can be configured to work with either the legacy or dedicated port by using jumpers, JP14 through JP16. As shipped, the demonstration board is hardware-configured for the legacy port. To change this, the user must cut the traces indicated in Figure 5-1 (item 14), and install pins and box jumpers. When installed, the ICD connector is configured by the jumpers as shown in Table 5-2.

In addition to setting the jumpers, the ICPRT Configuration bit in CONFIG4L must also be properly set. To enable the dedicated ICSP/ICD port, ICPRT must be set (= `1`).

When the board is configured to use the dedicated ICSP/ICD port, the Reset switch, S1, will no longer function. The ICRST pin acts only as an active-high Reset port, which works when the MPLAB ICD 2 sends a high-voltage programming signal, (V_{IHH}). The $\overline{\text{MCLR}}$ pin acts as both active-low and active-high Reset ports. Thus, the Reset switch, S1, only works when connected to the $\overline{\text{MCLR}}$ pin.

Additional information regarding the dedicated ICSP/ICD port can be found in the "Special Features" section of the device data sheet (DS39632).

TABLE 5-2: JUMPER CONFIGURATION FOR THE ICSP™/ICD CONNECTOR

ICD Connector Configuration	JP14	JP15	J16
Legacy ICD (CHP_$\overline{\text{MCLR}}$/RB7/RB6)	Pins 1-2	Pins 1-2	Pins 1-2
Dedicated Port (ICRST/ICPGD/ICPGC)	Pins 2-3	Pins 2-3	Pins 2-3

Note 1: For JP14 through JP16, pin 1 is the location closest to the microcontroller.

2: Before using the ICD connector for programming or debugging, verify that all three jumpers are set correctly. Failure to do so may result in programming or debugger failure.

5.2.4 Adding In-Circuit Emulation Capability

In addition to the on-board PIC18F4550 microcontroller, the PICDEM FS USB demonstration board can also be configured to work with hardware emulators, such as the MPLAB ICE 2000 and MPLAB ICE 4000. To do this, the user must install a 44-pin riser in the area designated U5 on the board. This is comprised of four 11-pin pads arranged in a square around the microcontroller. A device adapter (Microchip part number DVA18PQ440) will also be required for either of the MPLAB ICE devices.

Before using an external emulator, users must also configure the board to disable the existing controller. This is done by installing and configuring jumper, JP1. In its default configuration, with the upper pins connected, JP1 assigns device Reset control to S1. With the lower pins connected, JP1 grounds the $\overline{MCLR}$ pin of the PIC18F4550, holding it in permanent Reset. In this state, any external controller or emulator connected to the riser will control the demonstration board.

5.3 RESTORING THE PICDEM FS USB FIRMWARE

As shipped from the factory, the microcontroller on the PICDEM FS USB board is preprogrammed with the firmware required to interact with the Demo Tool application. This provides the code that makes the interactive Demo mode possible, and enables communication to establish the Bootload mode.

As users develop their own USB applications, it is likely that the controller will be reprogrammed with new firmware. If the original firmware is replaced, the status LEDs and other interactive features of the board may no longer work as previously described. It is even possible that USB connectivity may become disabled. For those users developing USB bootloader applications, it is possible that a change of device configuration with firmware loaded through a bootloader may render the demonstration board inoperable. In either case, it will be necessary to reprogram the board using an in-circuit programmer, such as MPLAB ICD 2.

Should it ever become necessary to return the board to its original state, it will be necessary to restore the original firmware. The necessary files have been installed on the host PC with the rest of PICDEM FS USB software. In MCHPFSUSB v1.2, the original hex file can be found in the directory, `MCHPFSUSB\fw\_factory_hex\`. The exact location of this file may be different in other versions of the framework, so refer to the MCHPFSUSB release notes if the file cannot be found. To reprogram the microcontroller directly via the ICD connector with the original firmware, use the hex file, `picdemfsusb.hex`.

Users should follow the procedure appropriate for their device programmer and development environment.

Chapter 6. Troubleshooting

6.1 HIGHLIGHTS

This chapter discusses the following:

- Common issues with the PICDEM FS USB demonstration board and how to solve them

6.2 COMMON PROBLEMS

1. The Power LED is not lit.

Normally, the PICDEM FS USB board behaves by default as a bus-powered device. If it is connected to a functioning USB port, the board will power-up and D7 will light. If this does not happen:

- Verify that the USB port on the host system is actually working by plugging a USB device that is known to be good into the port
- Verify that the cable is working by substituting a known functional cable

If a power supply is connected to the board, it will switch to Self-Power mode automatically; D7 will not be lit and D8 will light instead. If this does not happen:

- Verify that the power supply is plugged in and the wall outlet has power. If battery connection is used, verify that the correct polarity is used.
- Check that voltage is available (9 VDC) at the power supply's plug. If an OEM power supply is not being used, check for appropriate voltage (9 VDC).
- Check that the regulated voltage (5 VDC) is available.

2. The appropriate Power LED is lit, but the system does not recognize the board.

Check the USB cable for proper connections to the board and the computer. If necessary, verify the cable by swapping in another cable that is known to be good.

If the Demo Tool software is running, press and release S1 to reset the board.

In Windows Device Manager (accessed through the System applet in the Control Panel), use the "Scan for Hardware Changes" function to attempt to force the system to detect the board.

It is possible that the only firmware programmed into the microcontroller is the bootload firmware. In this case, enter Bootload mode by pressing and holding S2 while pressing and releasing S1. If the Bootload mode does not work, it is possible that the firmware has become corrupted. In this case, it will be necessary to reprogram the microcontroller with the original firmware through an external in-circuit programmer.

3. The board is functioning, but has stopped communicating with the Demo Tool.

In Demo mode, press and release S1, then reselect the board from the dropdown menu and click on "Connect".

In Bootload mode, press and hold S2 while resetting the board (pressing and releasing S1), then reselect the board from the dropdown menu.

4. After switching from Demo mode to Bootload mode in the Demo Tool, the board is not available for selection in the dropdown list.

The demonstration board's default firmware behaves as two separate USB devices. Invoking the bootloader firmware requires a special hardware signal, generated by holding S2 while resetting the board (pressing and releasing S1).

Similarly, the board will not be available automatically when switching from Bootload to Demo mode. In this case, it should only be necessary to reset the board (press and release S1).

5. On running Bootload mode for the first time, the host system tries to repeat the driver installation process.

Because the demo and bootloader firmware are essentially two separate pieces of code with individual product IDs, the host system will see each as a separate device. Each device will require the USB driver to be installed separately for it. Allow the Windows Install Wizard to use the same `.inf` file used for the original driver installation.

6. I'm using Windows Vista and can't install the custom USB driver.

Make sure you are attempting to install the latest version of the driver. Older versions of the custom driver did not support Windows Vista. The latest version of the custom driver, as well as the latest versions of the USB reference projects, can be obtained from: www.microchip.com/usb.

7. I'm using Windows Vista, and even though I can install the custom USB driver correctly, the PICDEM FS USB Demo Tool does not appear to be working correctly. The demo tool is generating pop-up error messages, or my device does not appear in the drop-down device selection list, even though it is definitely plugged in correctly.

Make sure you are using the latest version of the custom class USB driver and the PDFSUSB demo tool application. Older versions did not support Windows Vista. The latest versions can be obtained from: www.microchip.com/usb. After downloading and installing the package, the driver which Windows is actively using may still need to be manually updated. Open the device manager, right click on the device and use the "Update Driver" option to install the new version of the driver.

Appendix A. PICDEM™ FS USB Board Technical Information

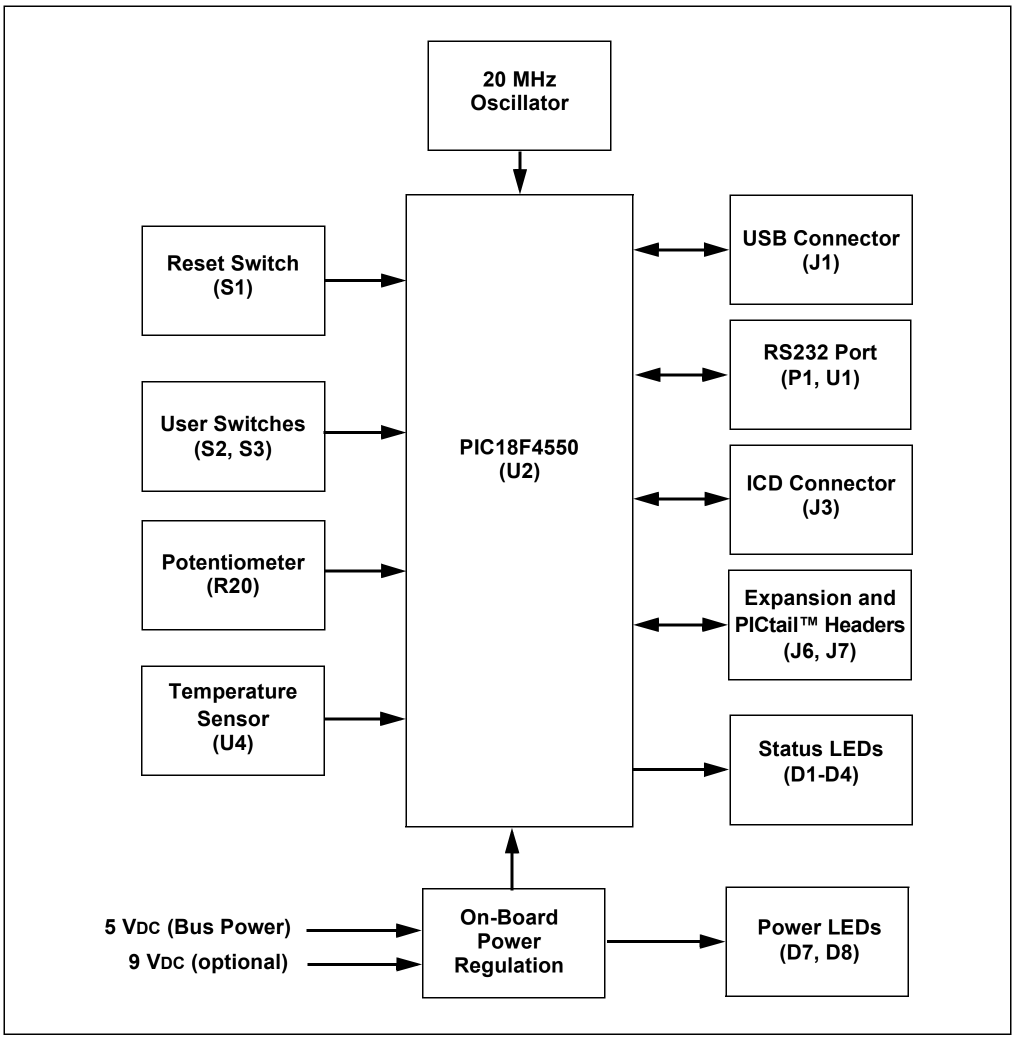
A.1 HIGHLIGHTS

This chapter will cover the following topics:

- PICDEM FS USB Block Diagram
- PICDEM FS USB Board Schematics

A.2 BLOCK DIAGRAM

FIGURE A-1: PICDEM™ FS USB DEMONSTRATION BOARD FUNCTIONAL BLOCK DIAGRAM



A.3 PICDEM FS USB BOARD SCHEMATICS

FIGURE A-2: BOARD SCHEMATIC, PART 1 (MICROCONTROLLER, VOLTAGE REGULATION AND ASSOCIATED PARTS)

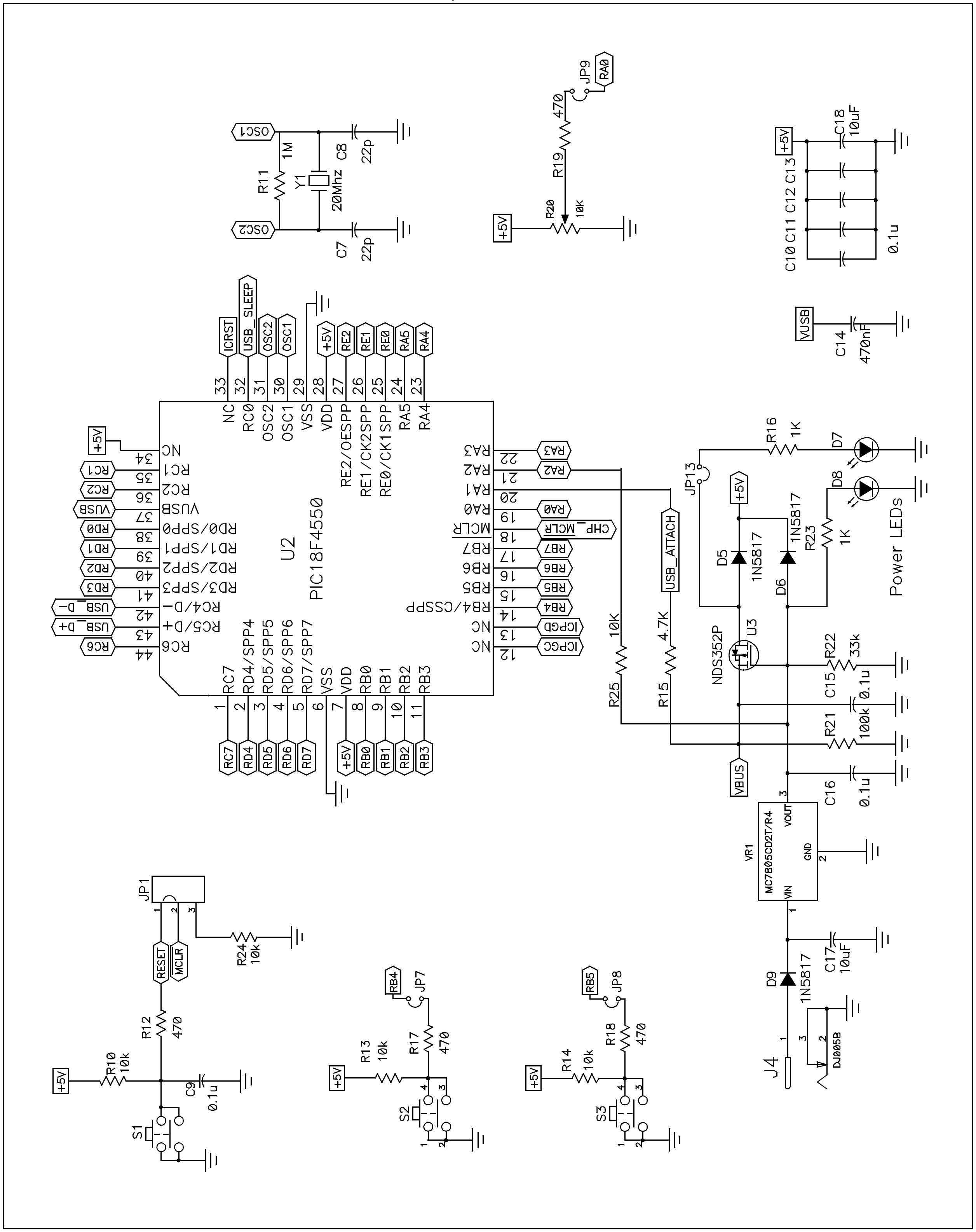


FIGURE A-3: BOARD SCHEMATIC, PART 2 (USART, USB INTERFACE, THERMAL SENSOR AND STATUS LEDS)

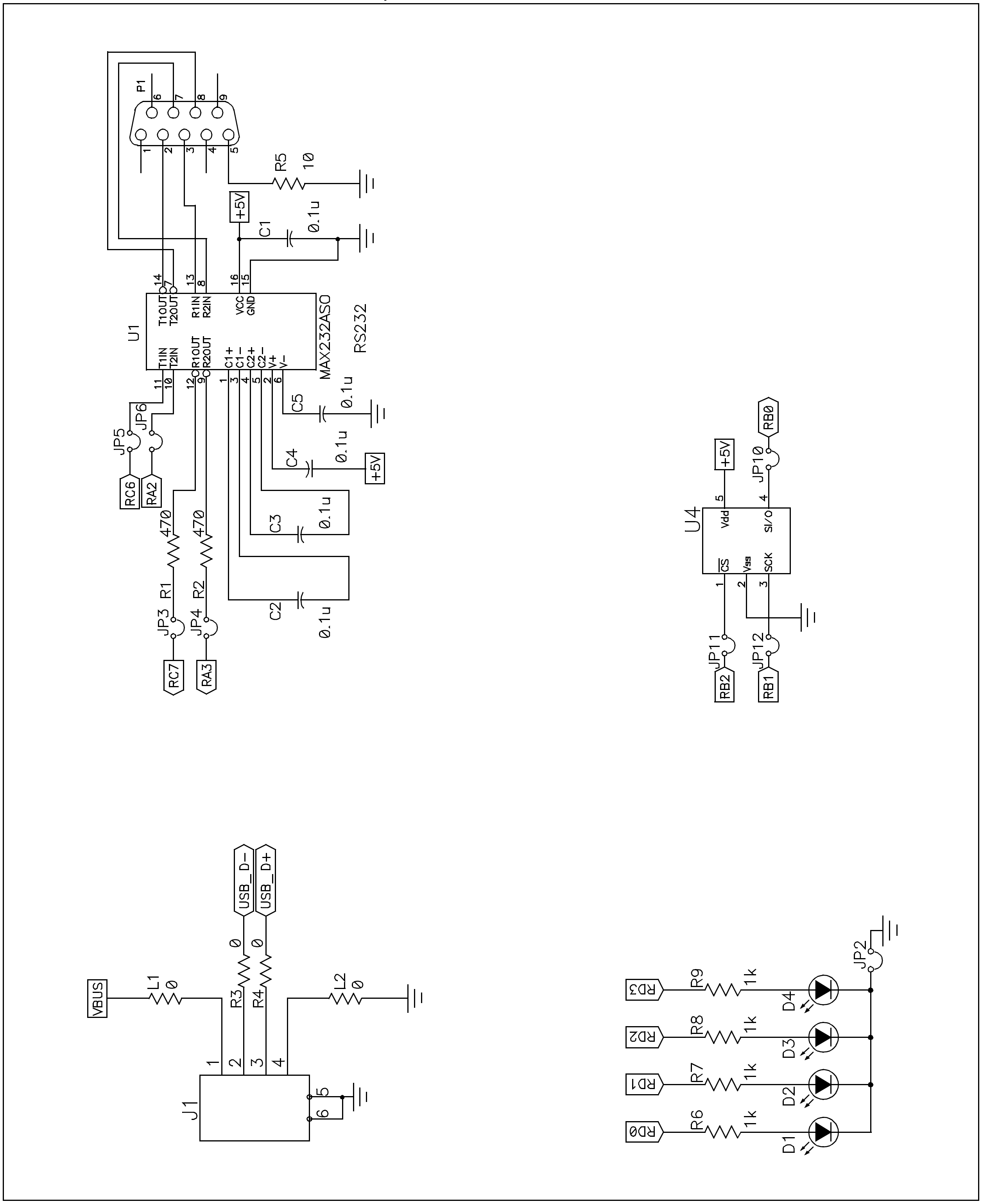


FIGURE A-4: BOARD SCHEMATIC, PART 3 (CONFIGURATION JUMPERS, PICkit™ HEADERS AND OPTIONAL MPLAB® ICE HEADER)

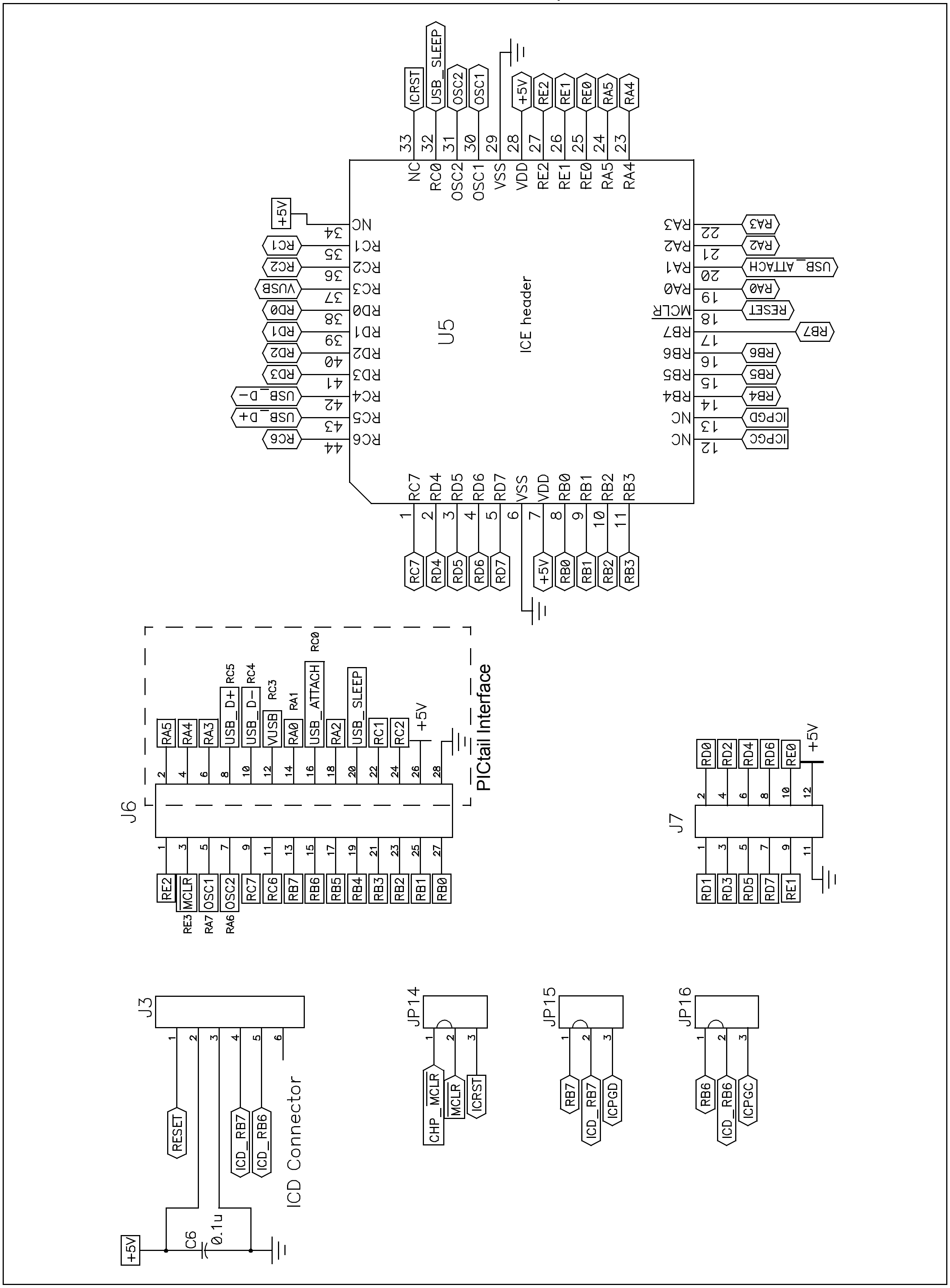


TABLE A-1: SIGNALS USED IN THE PICDEM™ FS USB SCHEMATIC

Signal Name	Function
CHP_$\overline{\text{MCLR}}$	Hardware Master Clear signal (direct connection to $\overline{\text{MCLR}}$ pin of microcontroller)
ICPGC, ICPGD, ICRST	Dedicated ICD port clock, data and chip Reset signals
ICD_RB6, ICD_RB7	Legacy ICD port signals (PGC and PGD)
$\overline{\text{MCLR}}$	Hardware Master Clear signal
OSC1, OSC2	External oscillator input for the microcontroller
RESET	Generic chip Reset signal (routable to $\overline{\text{MCLR}}$ or ICRST)
RXn	Bit n of microcontroller PORTx
USB_D+, USB_D-	Differential USB signals
USB_ATTACH	Bus attachment signal generated by power supply
USB_SLEEP	Optional USB suspend signal from an external transceiver
VUSB	Regulated 3.3V VUSB supplied by microcontroller's on-chip regulator
VBUS	Bus power voltage supplied from USB cable

TABLE A-2: RS-232 INTERFACE PINOUT

RS-232 Pin	PIC18F4550 Pin	Pin Function
1	—	CD
2	RC6/TX	RX (Received Data, PC Perspective)
3	RC7/RX	TX (Transmit Data, PC Perspective)
4	—	DTR
5	Vss	GND
6	—	DSR
7	RA3	RTS (Request To Send)
8	RA2	CTS (Clear To Send)
9	—	RI

TABLE A-3: PERIPHERAL INPUT AND OUTPUT PIN ASSIGNMENTS

PIC18F4550 Pin	Function
RD0	LED D1 Output
RD1	LED D2 Output
RD2	LED D3 Output
RD3	LED D4 Output
RA0	R20 (potentiometer) Input
SDI/RB0	TC77 (SPI Data Input)
SCK/RB1	TC77 (SPI Clock Output)
RB2	TC77 (Chip Select Output)

NOTES:

Index

WORLDWIDE SALES AND SERVICE

AMERICAS

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

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

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

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

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

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

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

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

ASIA/PACIFIC

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

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

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

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

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

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

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

ASIA/PACIFIC

India - Bangalore
Tel: 91-80-4182-8400
Fax: 91-80-4182-8422

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

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

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

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

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

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

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

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

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

Taiwan - Hsin Chu
Tel: 886-3-572-9526
Fax: 886-3-572-6459

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

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

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

EUROPE

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

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

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

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

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

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

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

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

01/02/08

