CYUSB3KIT-001

EZ-USB® FX3™ Development Kit Guide

Doc. #: 001-70237 Rev. *B

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
Phone (USA): 800.858.1810
Phone (Intnl): 408.943.2600
http://www.cypress.com

Copyrights

© Cypress Semiconductor Corporation, 2011-2012. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Any Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.

EZ-USB® is a registered trademark of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are property of the respective corporations.

Flash Code Protection

Cypress products meet the specifications contained in their particular Cypress PSoC Data Sheets. Cypress believes that its family of PSoC products is one of the most secure families of its kind on the market today, regardless of how they are used. There may be methods, unknown to Cypress, that can breach the code protection features. Any of these methods, to our knowledge, would be dishonest and possibly illegal. Neither Cypress 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'.

Cypress is willing to work with the customer who is concerned about the integrity of their code. Code protection is constantly evolving. We at Cypress are committed to continuously improving the code protection features of our products.

Contents

1. Introduction **5**

1.1 Kit Contents 5
1.1.1 Software on CD-ROM 5
1.1.2 Tools not Included 5
1.1.3 Other Suggested Tools....6
1.2 Additional Learning Resources....6
1.3 Document History 6
1.4 Documentation Conventions 6

2. Getting Started **7**

2.1 FX3 DVK Software CD Installation 7
2.2 Upgrade to Latest FX3 SDK Using Cypress Update Manager 11
2.3 Upgrade to Latest FX3 SDK from SDK Webpage 12
2.4 FX3 SDK Installation on Linux OS....14
2.5 Eclipse IDE 15
2.6 JTAG Debugger....15

3. Kit Operation **17**

3.1 Default Jumper Settings on DVK Board 17
3.2 Bus-Power Mode 18
3.3 Self-Power Mode 18
3.4 First Time USB Enumeration 19
3.4.1 Manual Installation of Cypress Driver 20
3.5 Boot Options 22
3.6 USB Boot 24
3.6.1 Download Firmware Image to FX3 RAM....25
3.7 I2C Boot....27
3.7.1 Download Firmware Image to I2C EEPROM 27
3.7.2 Booting from I2C EEPROM....30
3.7.3 Choosing Correct I2C EEPROM Configuration....30
3.8 SPI Boot....31
3.8.1 Download Firmware Image to SPI Flash....32
3.8.2 Booting from SPI Flash 35
3.9 ADMUX Boot 36
3.10 USB 3.0 Enumeration 36
3.11 Testing Bulkloop Firmware Examples 37
3.12 Modifying Bulkloop Firmware Example....37
3.12.1 Debugging Firmware Example....42

4. Hardware **43**

4.1 Power Supply....43
4.2 USB Receptacle 44

4.3 Clocking Mechanism for FX3 45
4.4 GPIF II Connector 47
4.4.1 Interconnecting to the GPIF II Connector 51
4.5 Reset Circuit 54
4.6 Serial Interfaces 54
4.6.1 I2C Interface 54
4.6.2 I2S 55
4.6.3 SPI and UART 55
4.6.4 JTAG 57

A. Appendix - Troubleshooting 59

A.1 Self-Power Mode 59
A.2 Bus-Power Mode 59
A.3 USB 2.0 Enumeration 59
A.4 USB Driver Installation 60
A.5 USB Boot 60
A.6 I2C Boot 61
A.7 SPI Boot 61
A.8 USB3.0 Enumeration 62

B. Appendix - PCB Layout 63

B.1 Assembly Top 63
B.2 Assembly Bottom 64

1. Introduction

The Cypress EZ-USB® FX3™ Development Kit (DVK) is a combination of hardware, software, and documentation that enables customers to evaluate the FX3 device. This document describes how to install the software related to the FX3 DVK and operate the board. It also briefly explains different hardware interfaces available on the DVK board. The appendix at the end of this document provides a troubleshooting guide, which helps to isolate the root cause of errors while operating the DVK board along with a corresponding solution.

1.1 Kit Contents

The Cypress EZ-USB FX3 DVK includes the following:

- Development kit PCB
- USB3.0 A to Micro B cable
- Quick Start Guide
- Kit CD
- 5-V DC adapter

Visit http://www.cypress.com/shop for more information. Inspect the contents of the kit. If any parts are missing, contact your nearest Cypress sales office for further assistance.

1.1.1 Software on CD-ROM

- FX3 Development Kit Installer: It includes documentation, such as user guide and release notes, and DVK hardware files, such as schematic, PCB Layout, and Gerber.
- FX3 Software Development Kit (SDK) Installer: It includes a list of firmware examples, which were designed and tested with the FX3 DVK hardware.
- Super-speed USB Suite: It includes a sample collection of USB PC applications in C++ and C# .NET framework designed to interact with the kit firmware examples.
- Eclipse IDE: It is an open source IDE that provides combined framework to edit and compile firmware examples using ARM GCC toolchain. Also, you can perform single-step debugging through IDE using the J-link debugger hardware.

1.1.2 Tools not Included

- Microsoft Visual C++ and C# software required for editing and building USB PC application source code.
- USB 3.0 capable PC Host: The FX3 DVK firmware examples can work in either USB 2.0 or USB 3.0 speeds.To achieve maximum performance out of FX3 hardware, a PC with USB 3.0 Host controller ports should be used.

1.1.3 Other Suggested Tools

USB Protocol analyzers: The following analyzers can be used to analyze the traffic between the PC host and the FX3 device:

- Hardware analyzers
  - Ellisys USB Explorer 280
  - Lecroy USB Voyager M3i
  - Beagle USB 5000 SuperSpeed protocol analyzer
- Software protocol analyzers
  - SourceQuest SourceUSB
  - SysNucleus USBTrace

1.2 Additional Learning Resources

Visit http://www.cypress.com for additional learning resources in the form of datasheets, technical reference manual, and application notes.

1.3 Document History

Revision	PDF Creation Date	Origin of Change	Description of Change
**	06/23/2011	MRKA	Initial version of kit guide
*A	08/11/2011	MRKA	Updated figures and table in section 2.1.4; added section 2.1.4.1. Updated Table 2-3 and Figure 2-11. Added Figure 2-12.
*B	11/29/2012	NMMA	Added the Appendix section and the Hardware chapter. Updated Getting Started chapter.

1.4 Documentation Conventions

Table 1-1. Document Conventions for Guides

Convention	Usage
`Courier New`	Displays file locations, user entered text, and source code: `C:\ ...cd\icc\`
Italics	Displays file names and reference documentation: Read about the *sourcefile.hex* file in the *PSoC Designer User Guide*.
[**Bracketed, Bold**]	Displays keyboard commands in procedures: [**Enter**] or [**Ctrl**] [**C**]
File > Open	Represents menu paths: File > Open > New Project
Bold	Displays commands, menu paths, and icon names in procedures: Click the **File** icon and then click **Open**.
Times New Roman	Displays an equation: 2 + 2 = 4
Text in gray boxes	Describes cautions or unique functionality of the product.

2. Getting Started

This chapter guides you through the installation of the FX3 DVK and FX3 Software Development Kit (SDK).

2.1 FX3 DVK Software CD Installation

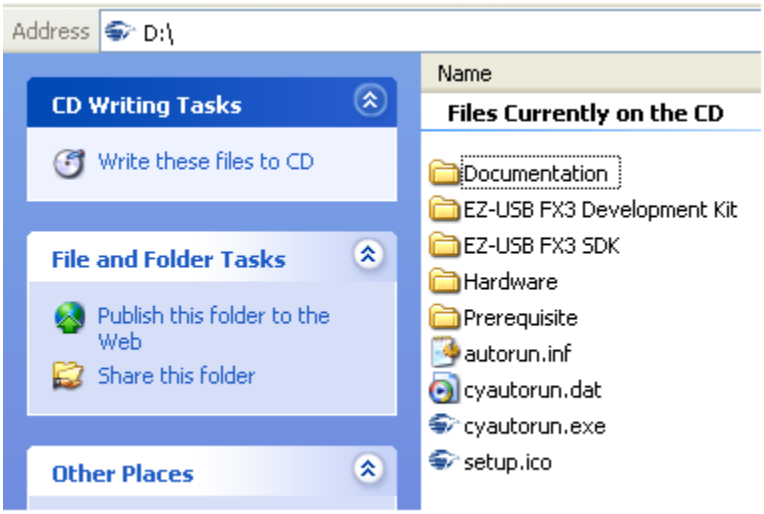
To install the kit software, follow these steps:

1. Insert the kit CD into the CD drive of your PC. The CD is designed to auto-run and the kit installer startup screen appears. You can also download the latest kit installer ISO file from http://www.cypress.com/go/CYUSB3KIT-001. Create an installer CD or extract the ISO using WinRar and install the executables.
2. Click **Install EZ-USB FX3 Development Kit** to start the installation, as shown in Figure 2-1.

Figure 2-1. Kit Installer Startup Screen

Note If auto-run does not execute, double-click on the *cyautorun.exe* file in the root directory of the CD, as shown in Figure 2-2.

Figure 2-2. Kit Installer Folder

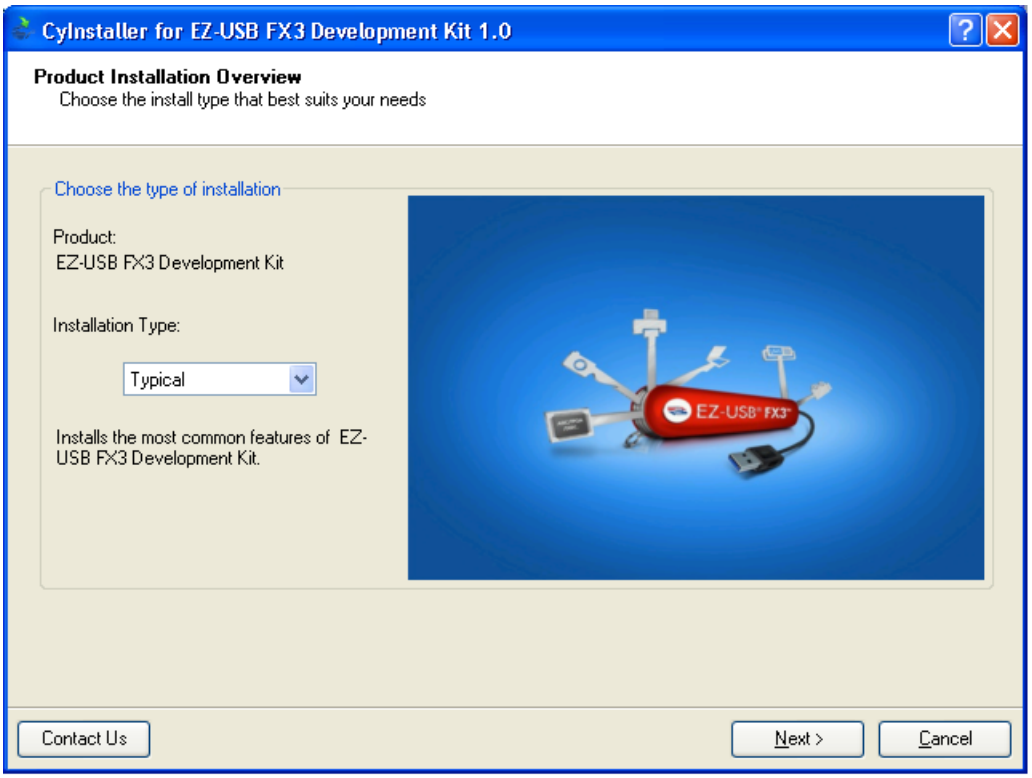


3. The **InstallShield Wizard** screen appears. The default location for setup is shown on the InstallShield Wizard screen. You can change the location for setup using **Change**, as shown in Figure 2-3. Choose the default settings for the installer.
4. Click **Next** to launch the kit installer.

Figure 2-3. InstallShield Wizard

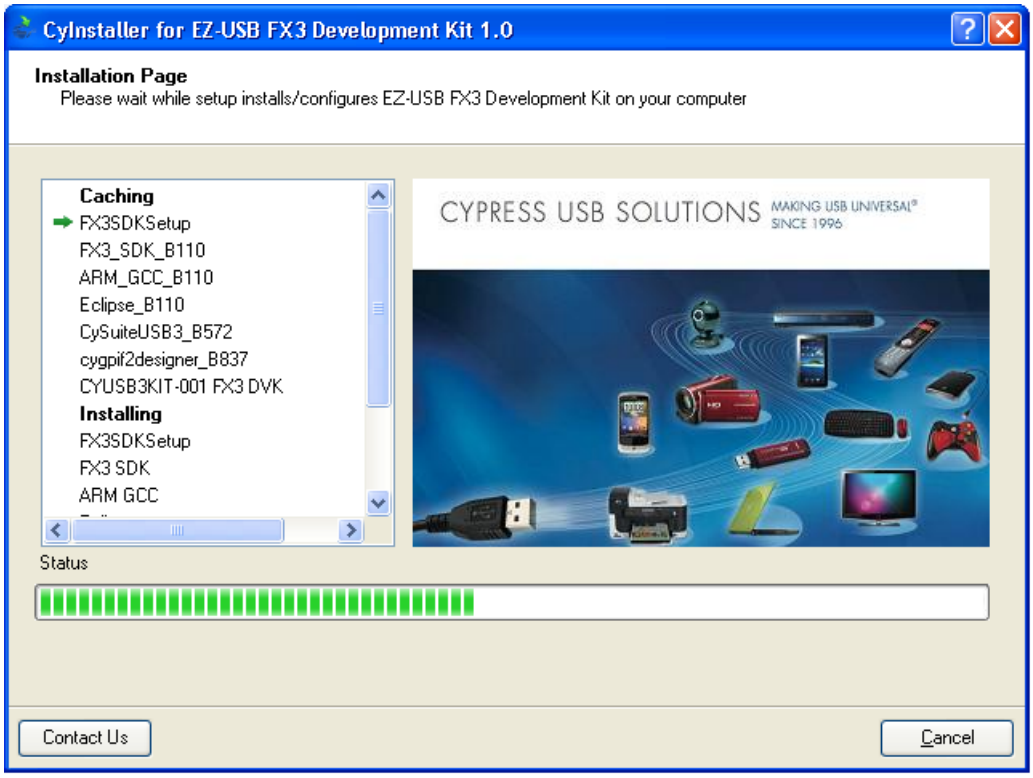
5. On the **Product Installation Overview** screen, the drop-down menu has **Typical, Custom,** and **Complete** options. Choose the **Typical** option to install common list of software, as shown in Figure 2-4. Click **Next** to start the installation.

Figure 2-4. Installation Type Options



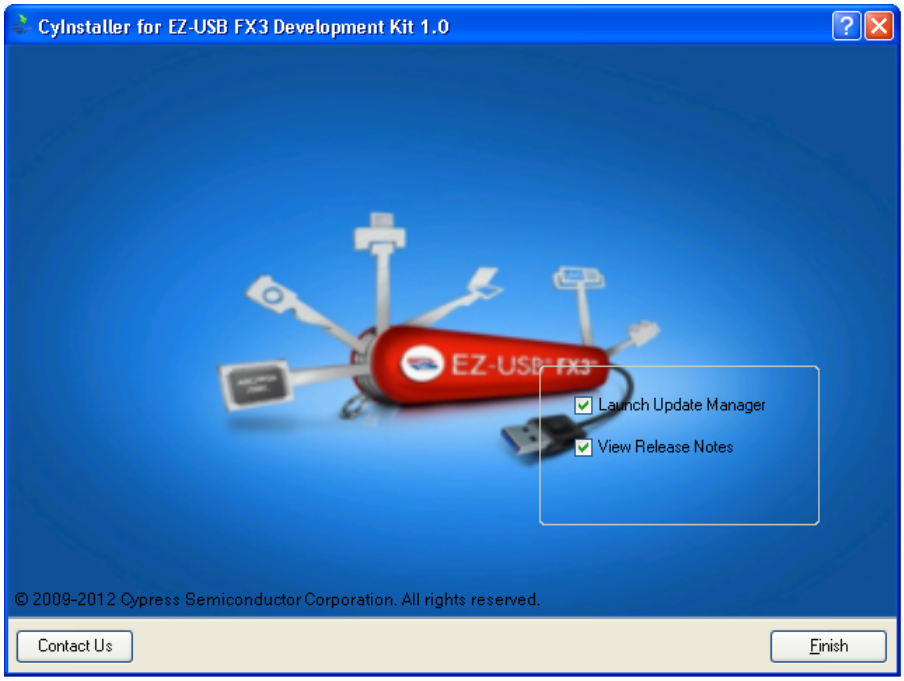
6. Review and accept the **License Agreement** and click **Next**. The licenses are for the Cypress tools, CodeSourcery compiler, and Eclipse IDE. Wait until all the packages are downloaded and installed successfully.
7. When the installation begins, all packages are listed on the Installation. In addition to DVK set up software (*FX3DVKSetup.exe*), the CD provides the FX3 Software Development Kit (SDK) Installer (*FX3SDKSetup.exe)*, which is an assembly of MSI packages. The MSI packages include
   a. **Eclipse_<Build_no>.msi:** he Eclipse IDE for C/C++ Developer is provided as part of the FX3 SDK. This IDE comprises the base Eclipse platform and the CPP feature plug-ins required for editing sample firmware examples provided in the FX3 SDK.
   b. **ARM_GCC_<Build_no>.msi:** The ARM GCC tool chain from CodeSourcery is an open source toolchain which is used to compile sample firmware examples provided in the FX3 SDK.
   c. **FX3_SDK_B110.msi:** The SDK MSI package contains SDK documentation, sample firmware examples to perform hands-on with DVK hardware.
   d. **CySuiteUSB3_<Build_no>.msi:** The SuiteUSB installer contains Windows host driver (cyusb3.sys), C and C# API libraries, and sample PC applications such as Control Center, Bulkloop, and Streamer.
   e. **cygpif2designer_<Build_no>.msi:** This software tool is used to define state machine wave-forms. The waveforms are required for GPIF II interface in FX3 to communicate with external devices. The tool generates equivalent source, which can be integrated into SDK firmware example to test the communication and timing.

Figure 2-5. Installation Page



8. Click on **Finish** to complete the installation.

Figure 2-6. Installation Complete



Note After software installation, verify the installation and setup. Following are the default directory paths for the installed software:

- FX3 DVK content at `C:\Program Files\Cypress\EZ-USB FX3 Development Kit\<version>\`
- FX3 SDK content at `C:\Program Files\Cypress\EZ-USB FX3 SDK\<SDK_version>\`
- In Windows 64-bit platform `C:\Program Files\.. directory equivalent is C:\Program Files(x86)\`

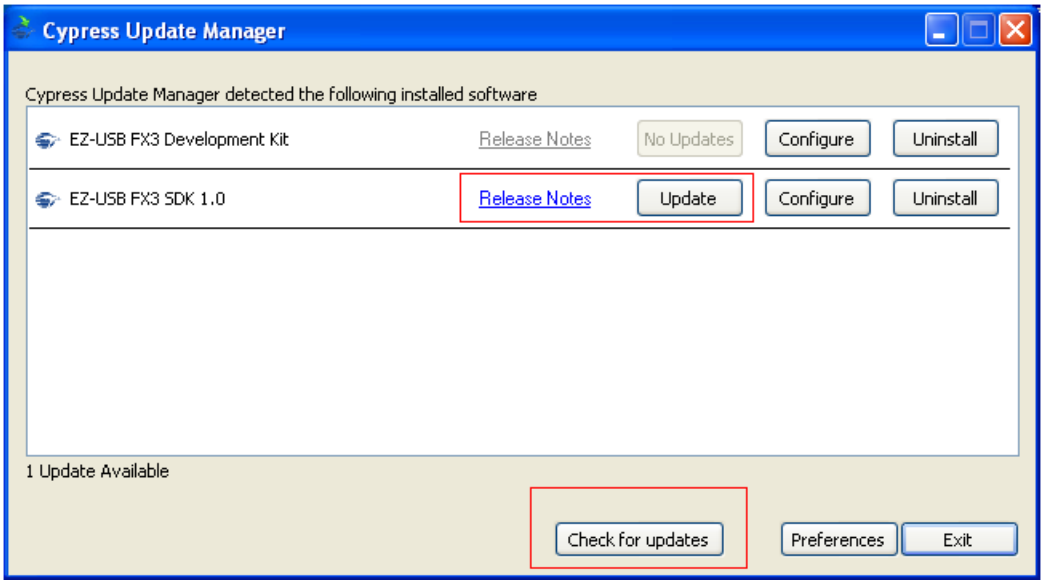
2.2 Upgrade to Latest FX3 SDK Using Cypress Update Manager

Follow these steps to upgrade the existing FX3 SDK in PC/Laptop to latest release.

Note Take a backup of existing SDK content (if modified) before proceeding further. By default, the new SDK installer removes the older SDK before installation.

1. Click on the Cypress Update Manager.exe window located at **Start > All Programs > Cypress > Cypress Update Manager**.
2. Click on the **Check for Updates** button at the bottom of the window as shown in Figure 2-7 on page 11. The greyed out **No Updates** button adjacent to the EZ-USB FX3 SDK entry transforms into an **Update** button if there is a new release of SDK. The **Release notes** hyperlink for the latest SDK also is activated as shown in Figure 2-7 on page 11.

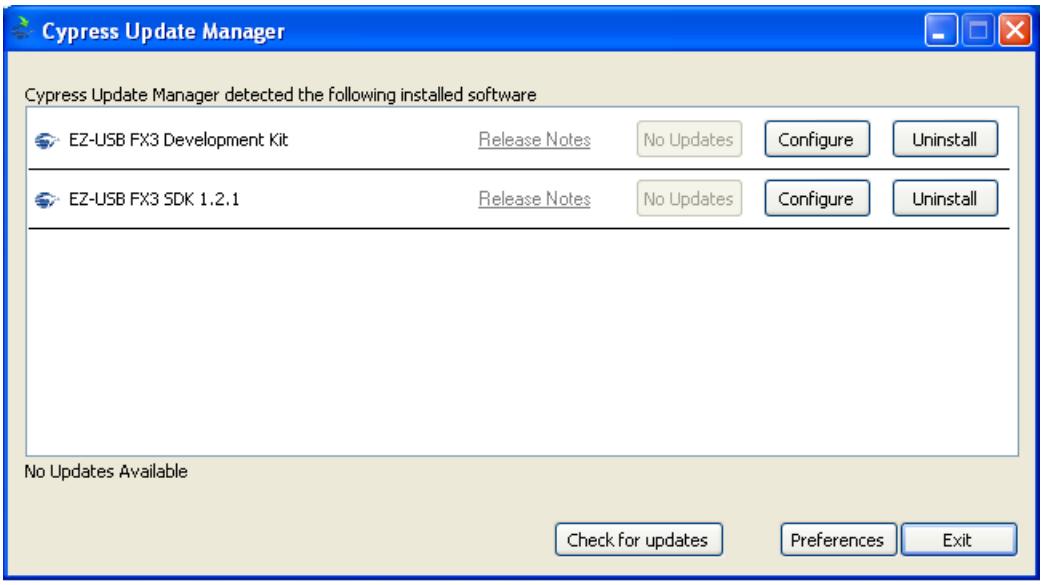
Figure 2-7. FX3 DVK and SDK Installer entries in Update Manager



3. Review the release notes to know the latest SDK revision, new features, and patches implemented compared to the current SDK. Click on **Update** button.
4. The SDK installer downloads several sub-packages (ARM_GCC, SuiteUSB, Eclipse, and so on) to local drive of the PC/Laptop. Then it un-installs the current SDK components and installs the latest SDK component sequentially. The list of packages downloaded, un-installed and installed at each step are visible on the installer screen.
5. After complete installation the final installer window with Finish button pops up. Click on **Finish** button to complete the entire installation process. Close the update manager and re-open the window located at **Start > All Programs > Cypress > Cypress Update Manager**. The latest SDK version is visible in the update manager window as shown in Figure 2-8.

Note In Windows Vista/7 OS platforms verify **Start > Programs > Cypress > Cypress Update Manager**

Figure 2-8. Cypress Update Manager



Note The updated SDK is available at `C:\Program Files\Cypress\EZ-USB FX3 SDK\<SDK_Version>`. The FX3 DVK software can also be upgraded in a similar way using **Update Manager**.

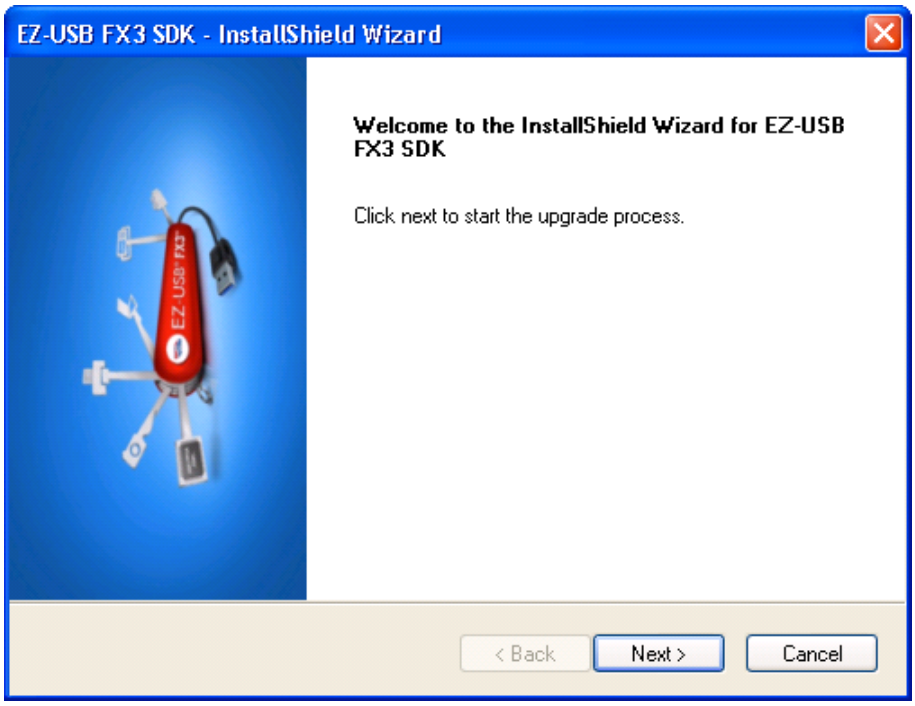
2.3 Upgrade to Latest FX3 SDK from SDK Webpage

The EZ-USB FX3 SDK is updated on regularly with additional features and patches. Starting with the SDK version 1.2, support for the Linux operating system is also added. To upgrade the existing SDK installation, follow these steps:

1. Download the latest SDK Installer (*FX3SDKSetup.exe*) from http://www.cypress.com/?rID=57990.
2. The FX3 SDK Installer will give an option to upgrade or remove the existing SDK. Upgrade to new version of the SDK.

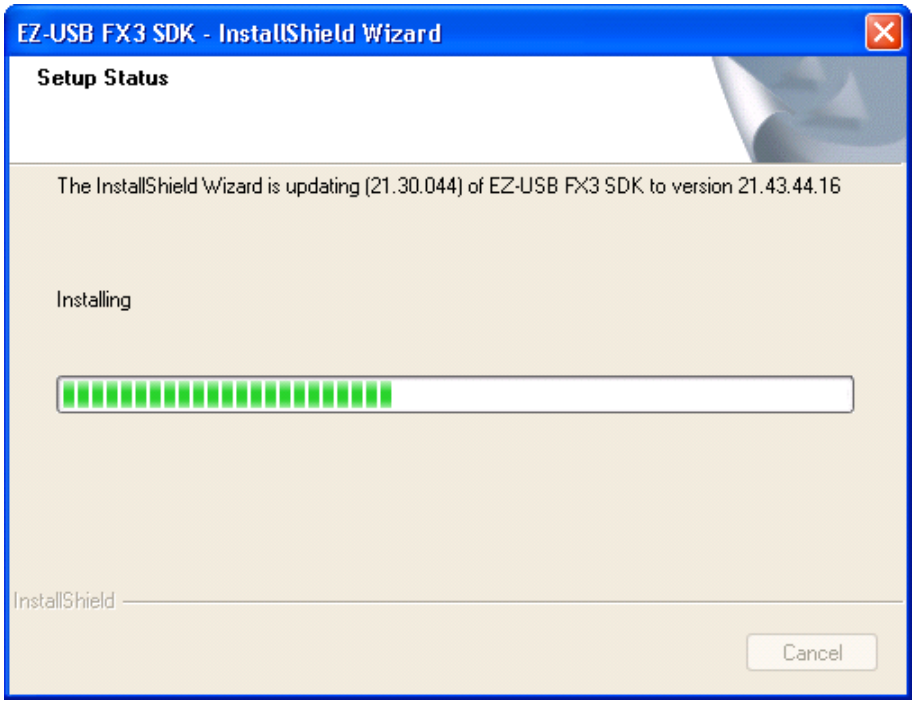
   Note If the current SDK examples are edited, take a backup of the existing firmware examples before uninstalling the older version of the SDK.
3. Click on the downloaded **FX3SDKSetup.exe**. This displays a window prompting for an upgrade of existing SDK, as shown in Figure 2-9. Click **Next**.

Figure 2-9. FX3 SDK Installsheild Wizard



4. Observe the SDK update in progress with a progressive green bar in the middle.

Figure 2-10. SDK update in Progress



5. The major components of the FX3 SDK are as follows:
   a. **Documentation:** The SDK documentation can be located at `C:\Program Files\Cypress\EZ-USB FX3 SDK\<SDK_version>\doc` after installation. The documentation illustrates how to get started with firmware examples and overview of firmware API.
   b. **Firmware Examples:** After installing SDK, the firmware examples can be located at `C:\Program Files\Cypress\FX3 SDK\<SDK_version>\firmware`.
   c. **Utilities:** The SDK provides utilities such as *elf2img.exe*, which converts raw firmware binary into a bootable image format.
   d. The SuiteUSB provides sample USB 3.0 drivers and PC application for both 32-bit and 64-bit Windows OS platforms.These applications can be used to communicate with the DVK Board. After installation the contents can be located at `C:\Program Files\Cypress\EZ-USB FX3 SDK\<SDK_Version>\bin`

   Note In Windows 64-bit platform `C:\Program Files\..` directory equivalent is `C:\Program Files(x86)\..` .

2.4 FX3 SDK Installation on Linux OS

The FX3 SDK version 1.2 and later revisions support firmware development with Eclipse IDE and debugging using the J-Link JTAG debugger probe on a Linux platform. The EZ-USB FX3 SDK for Linux is released in the form of a gzipped tar archive called "FX3_SDK.tar.gz". On extraction, this tar archive contains the following gzipped tar archives:

- FX3_Firmware.tar.gz: The FX3 firmware library and examples.
- ARM_GCC.tgz: Sourcery ARM GNU toolchain to compile firmware examples.
- eclipse_x86.tgz: Eclipse IDE for 32-bit Linux OS.
- eclipse_x64.tgz: Eclipse IDE for 64-bit Linux OS.
- cyusb_linux_<Build_no>.tar.gz: The CyUSB Suite provides QT based USB applications to communicate with FX3 device.

The installation procedure involves extraction of these archives and the setting some environment variables. The installation procedure is as follows:

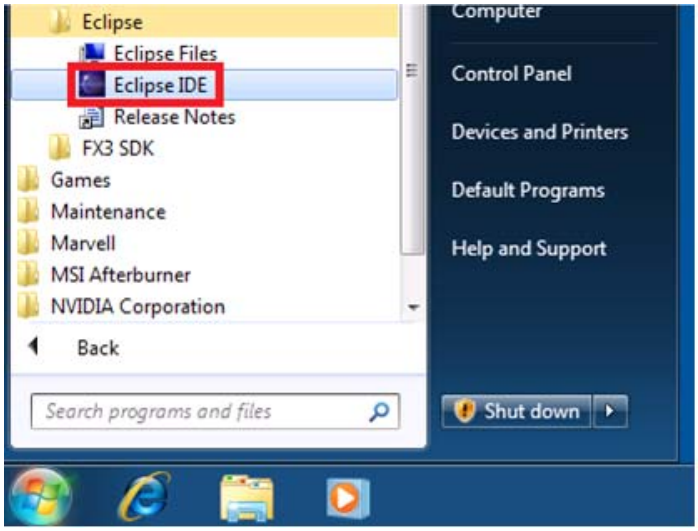
- Extract the contents of the FX3_SDK.tar.gz archive at a preferred location, say, $HOME/Cypress.
- Change to the install location ($HOME/Cypress) and extract the contents of the FX3_Firmware.tar.gz, ARM_GCC.tgz, and eclipse_x86.tgz (or eclipse_x64.tgz) files.
- This creates a directory structure of firmware examples, software tools, Eclipse IDE, and so on.

Note For more details on Linux SDK installation and configuration, see "FX3 SDK and Software for Linux" in the document *Getting Started with FX3 SDK.pdf* at `C:\Program Files\Cypress\FX3 SDK\1.0\doc`.

2.5 Eclipse IDE

Launch Eclipse IDE from the **Start > Cypress > Eclipse > Eclipse IDE**.

Figure 2-11. Eclipse IDE Entry in Windows Programs Directory



To learn more on how to compile SDK firmware examples using Eclipse IDE, see section “Setting Up and Using Eclipse IDE" from document titled *Getting Started with FX3 SDK.pdf* at `C:\Program Files\Cypress\EZ-USB FX3 SDK\<SDK_version>\doc`.

2.6 JTAG Debugger

The Segger J-Link probe is the preferred JTAG probe for the FX3 SDK. This probe, along with the Segger J-Link ARM GDB server, is used for debug. The Eclipse IDE will connect to the J-link GDB server when debugging the firmware. To get Eclipse working with the GDB server, create a debug configuration for the J-link. Details on how to perform this configuration and debugging is available in the FX3 Programmers Manual at `C:\Program Files\Cypress\EZ-USB FX3 SDK\<SDK_version>\doc` (section 12.2.2.3 Executing and Debugging).

3. Kit Operation

The FX3 DVK board provides interfaces such as I2C, SPI, and I2S to configure and evaluate its functionality. The DVK board also provides the Samtec connector to connect different types of external devices through the GPIF II interface. Sample examples are provided in the SDK to evaluate each of these interfaces.

3.1 Default Jumper Settings on DVK Board

The jumpers and dip switches on FX3 DVK board are set by default in self-power mode. The default jumper and switch settings are shown in Table 3-1.

Table 3-1. Default Jumper Settings

S.No	Jumper/Switch	Pins to be shorted using jumpers	Function
1	J101	1 and 2	GPIO_46=UART_RTS
2	J102	1 and 2	GPIO_47=UART_CTS
3	J103	1 and 2	GPIO_48=UART_TX
4	J104	1 and 2	GPIO_49=UART_RX
5	J136	3 and 4	VIO1(3.3V)
6	J144	3 and 4	VIO2(3.3V)
7	J145	3 and 4	VIO3(3.3V)
8	J146	3 and 4	VIO4(3.3V)
9	J134	4 and 5	VIO5(3.3V)
10	J135	2 and 3	CVDDQ(3.3V)
11	J143	1 and 6	VBATT(2.5V)
12	J96 & SW25	2 and 3	a. PMODE0 Pin state (ON/OFF) selection using SW25. b. SW25.1 should be OFF
13	J97 & SW25	2 and 3	a. PMODE0 Pin state (ON/OFF) selection using SW25. b. SW25.2 should be OFF
14	J98	1 and 2	PMODE2 Pin Floating
13	J72	1 and 2	RESET
14	J42	Not Installed	GPIO_58=I2C_SCL
15	J45	Not Installed	GPIO_59=I2C_SDA
16	J100	1 and 2	GPIO_21=CTL4

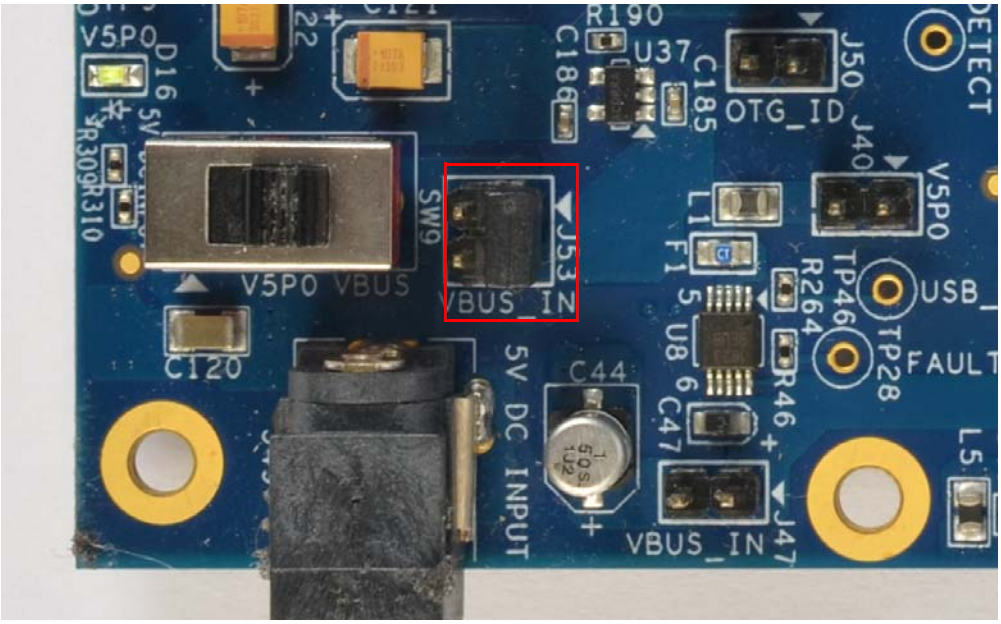
Note For more details on each I/O pin refer to Chapter Hardware on page 43.

3.2 Bus-Power Mode

The FX3 DVK board can also operate in bus-power mode. Follow these steps:

1. Verify if pins 1-2 of Jumper J53 are shorted.
2. Use the toggle switch SW9 to power ON the board. The switch should point to the direction labeled VBUS_IN instead of V5P0.

Figure 3-1. Bus-power Mode Setting on FX3

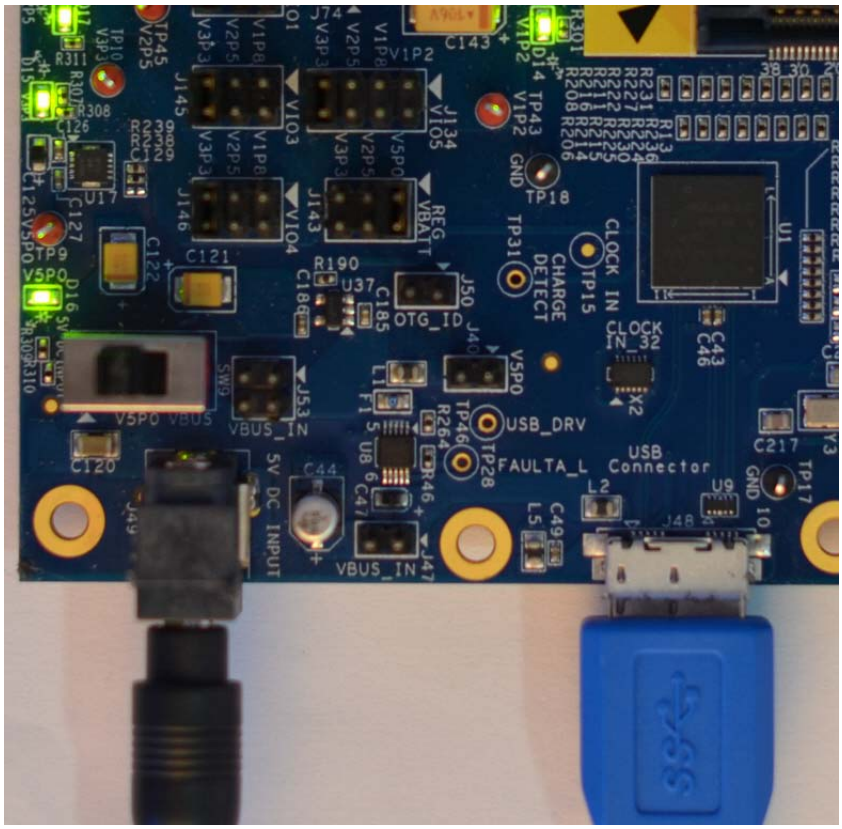


3.3 Self-Power Mode

The FX3 DVK board default settings allow the board to operate in self-power mode. Follow these steps to connect the board in self-power mode.

1. Remove jumper on pins 1-2 of Jumper J53.
2. Plug the 5-V power supply adapter supplied with the kit to the J49 power jack. Use the toggle switch SW9 to power on the board. The switch should point to the direction labeled V5P0 instead of VBUS_IN.
3. LEDs D14, D15, D16, D17, and D18 glow green, as shown in Figure 3-2.

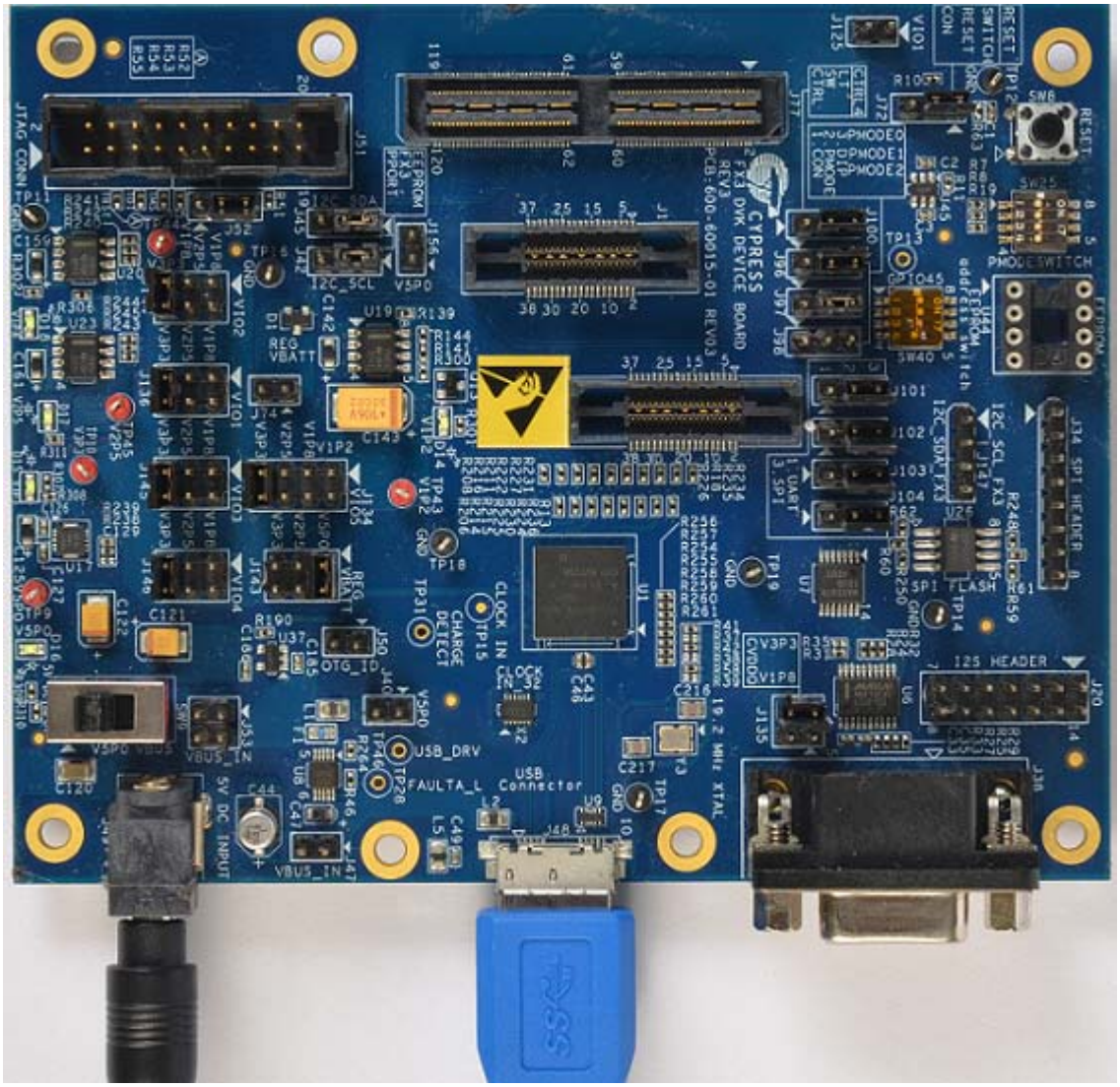
Figure 3-2. LED Indicator in Self-power Mode



3.4 First Time USB Enumeration

When the FX3 board is powered for the first time, connect a USB 3.0 A-to-Micro B cable supplied with the kit to the J48 connector, as shown in the Figure 3-3.

Figure 3-3. USB 3.0 Cable to J48 Connector

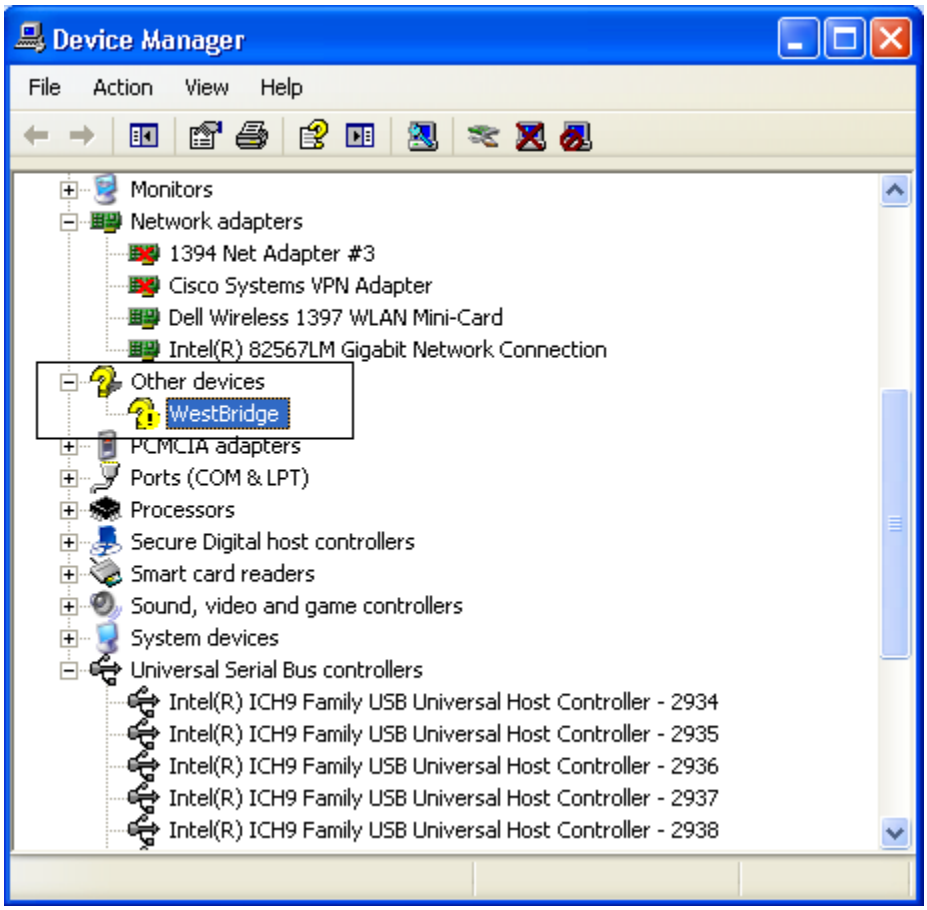


Note The bootloader in FX3 ROM by default enumerates as USB 2.0 device. After downloading, the firmware to the FX3 DVK board will enumerate as either USB 3.0 or USB 2.0 peripheral depending on the configuration defined in firmware and the host PC capability.

3.4.1 Manual Installation of Cypress Driver

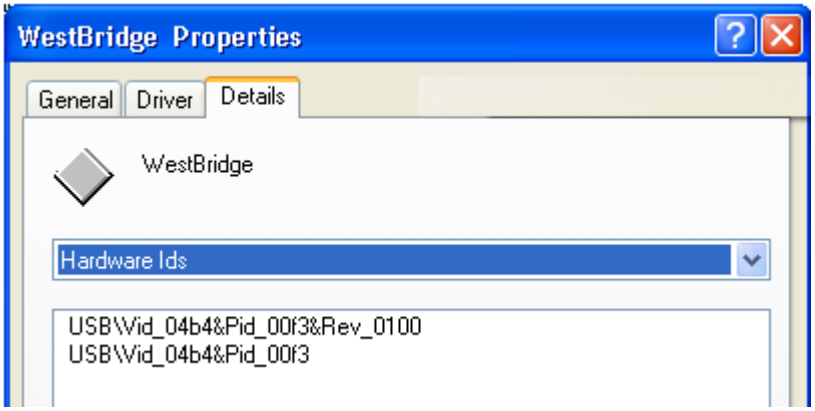
1. In **Windows Start > My Computer** or go to **Computer** (for Windows 7) > right-click **Properties > Hardware Tab > Select Device Manager**. Locate the FX3 device entry with a yellow symbol in the Other Devices list. Sometimes, the FX3 DVK enumerates as **Westbridge**, as shown in Figure 3-4.

Figure 3-4. FX3 Device Entry in Windows Device Manager



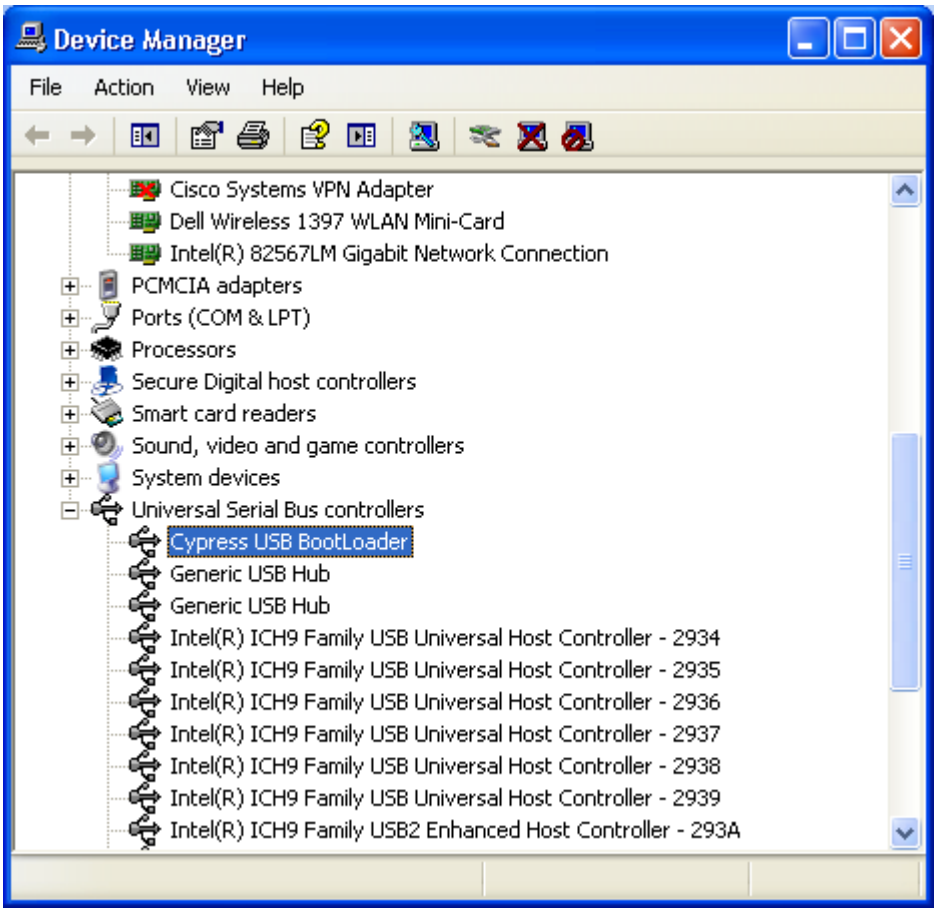
2. Right-click on the yellow device entry to verify the VID/PID of the device. Select **Properties > Details**. Select **Hardware IDs** and observe if the default VID/PID is 0x04B4/0x00F3, as shown in Figure 3-5.

Figure 3-5. Default FX3 Bootloader VID/PID



3. The Windows OS hardware wizard prompts for a USB driver to the connected FX3 device. To manually bind the USB driver, follow the procedure outlined in section **Installing Drivers** of *Getting Started with FX3*. After successfully binding the USB drivers, the FX3 DVK board is visible as "Cypress Bootloader" in Windows Device Manager, as shown in Figure 3-6.

Figure 3-6. FX3 Entry in Device Manager after Driver Binding



3.5 Boot Options

The EZ-USB FX3 provides several booting options. The boot option is determined by the PMODE[2:0] input pins. The PMODE[2:0] pins can be configured using combination of Jumpers(J98, J97, and J96) and dip switch SW25. For I2C and SPI boot options additional jumpers (J42, J45, J101, J102, J103, and J104) and dip switch (SW40) needs to be configured.

Figure 3-7 summarizes the major components available on FX3 DVK board to configure different boot options.

Figure 3-7. Boot Dip Switch and Jumper Settings

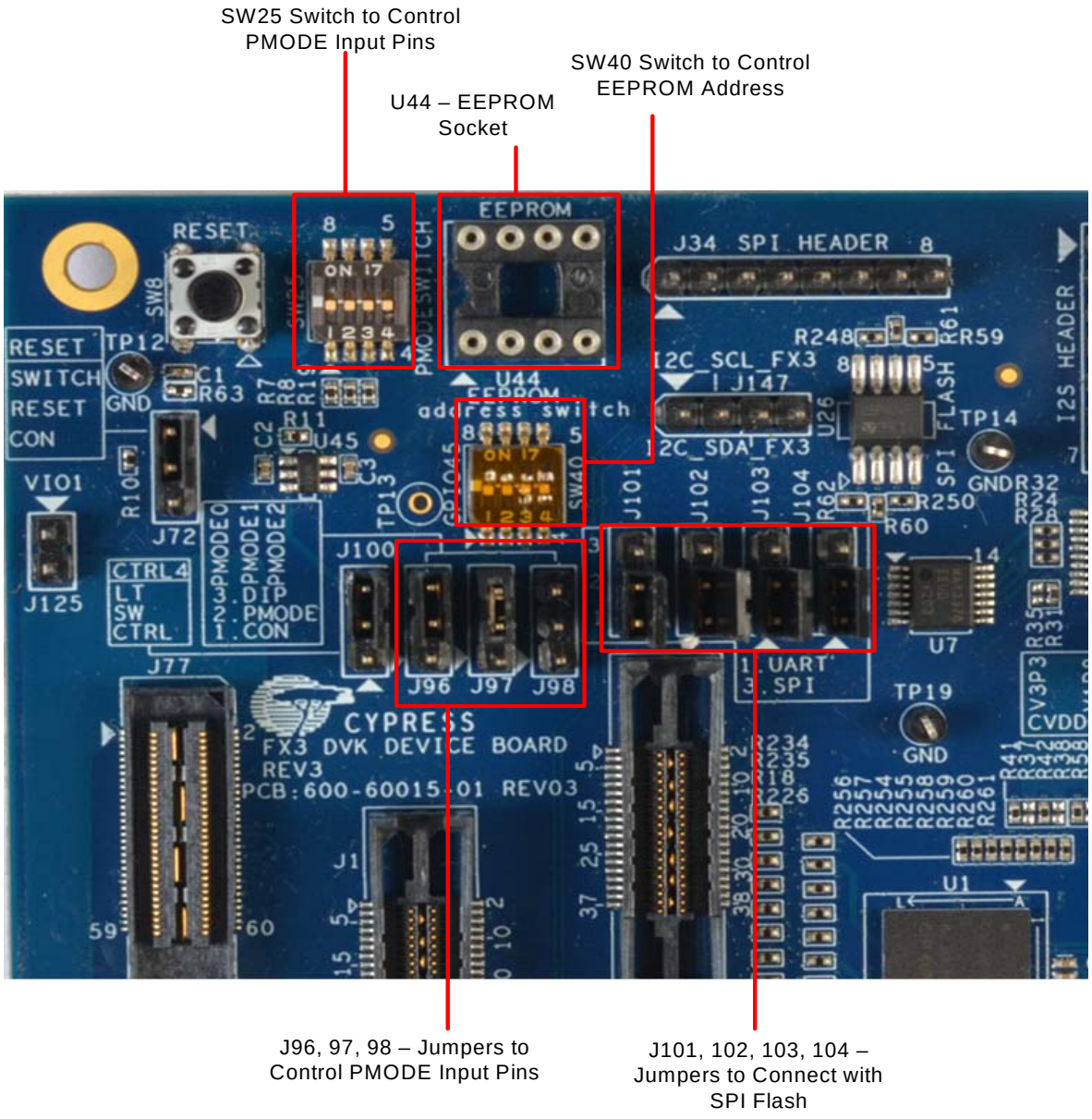


Table 3-2 provides the list of boot interfaces supported on the FX3 device for corresponding PMODE [2:0] pins.

Table 3-2. Boot Options with PMODE Pins

PMODE2 pin state	PMODE1 pin state	PMODE0 pin state	Boot option	USB Fallback
Z	0	0	Sync ADMUX(16-bit)	No
Z	0	1	Async ADMUX(16-bit)	No
Z	1	1	USB Boot	Yes

Table 3-2. Boot Options with PMODE Pins

PMODE2 pin state	PMODE1 pin state	PMODE0 pin state	Boot option	USB Fallback
Z	0	Z	SRAM16-bit)	No
1	Z	Z	I2C	No
Z	1	Z	I2C=> USB	Yes
0	Z	1	SPI=> USB	Yes

Note "Z" = floating I/O pin state.

On the DVK board, the PMODE [2:0] pins are available through jumpers J98, J97, J96, and switch SW25. Table 3-3 shows the details of jumper and switch combination to select a specific boot option.

Table 3-3. PMODE Pin Options on DVK Board

PMODE pin	PMODE Pin State	Jumper and Dip Switch Combination for PMODE Pin State
PMODE2	Z	1. No jumper installed on J98 2. SW25.3 - Don't care (the switch can be either turned ON or OFF)
	1	1. Short pins 2-3 of jumper J98 2. SW25.3 set to OFF
	0	1. Short pins 2-3 of jumper J98 2. SW25.3 set to ON
PMODE1	Z	1. No jumper installed on J97 2. SW25.2 - Don't care (the switch can be either turned ON or OFF)
	1	1. Short pins 2-3 of jumper J97 2. SW25.2 set to OFF
	0	1. Short pins 2-3 of jumper J97 2. SW25.2 set to ON
PMODE0	Z	1. No jumper installed on J96 2. SW25.1 - Don't care (the switch can be either turned ON or OFF)
	1	1. Short pins 2-3 of jumper J96 2. SW25.1 set to OFF
	0	1. Short pins 2-3 of jumper J96 2. SW25.1 set to ON

Note The "OFF" Setting on dipswitch SW25 means Logical "1" on the corresponding PMODE pin.

See AN76405 - EZ-USB FX3 Boot Options for an explanation of each of these boot options.

3.6 USB Boot

The FX3 DVK board boots in USB mode if PMODE [2:0] pins are set to Z11. The board is by default set to PMODE [2:0] =Z1Z. This indicates an I2C boot interface with a USB fallback option if there is no EEPROM or bad EEPROM image. By default, there is no on-board EEPROM and hence the bootloader will automatically fallback to USB mode. The board enumerates with bootloader Vendor ID/Product ID (VID/PID-0x04B4/0x00F3) when the USB cable is connected to USB Host PC.

3.6.1 Download Firmware Image to FX3 RAM

Follow the procedure outlined here to download the firmware image to FX3 RAM.

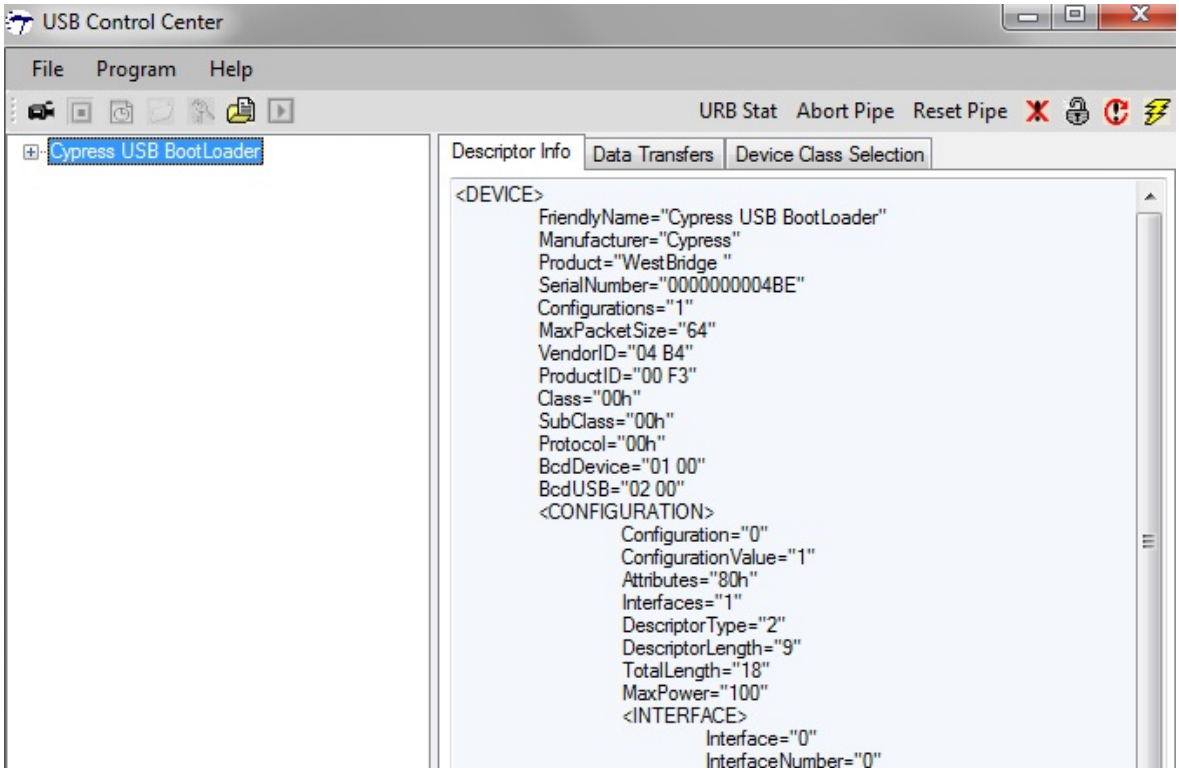
1. Enable USB boot, by setting the PMODE[2:0] pins to Z11. On the DVK board, this is done by configuring the jumpers and switches, as shown in Table 3-4.

Table 3-4. Jumper Settings for USB Boot

PMODE pin	Required PMODE Pin State	Jumper and Dip Switch Combination
PMODE2	Z	1. No jumper installed on J98 2. SW25.3 - Don't care (the switch can be either turned ON or OFF)
PMODE1	1	1. Short pins 2-3 of jumper J97 2. SW25.2 set to OFF
PMODE0	1	1. Short pins 2-3 of jumper J96 2. SW25.1 set to OFF

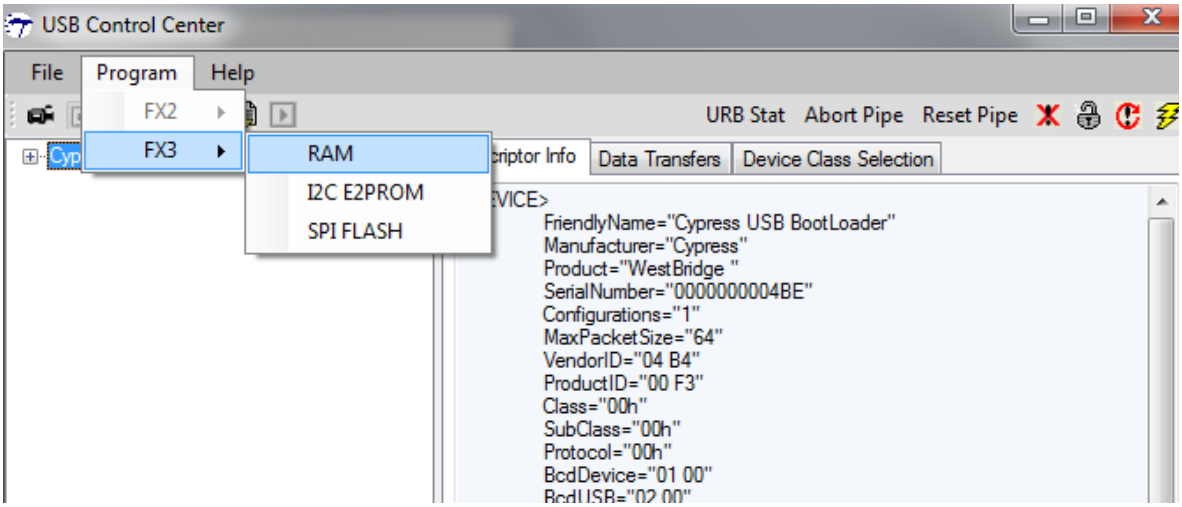
2. When connected to a USB host, the FX3 device enumerates in Control Center as "Cypress USB Bootloader", as shown in Figure 3-8.

Figure 3-8. FX3 Default Bootloader Entry in Control Center



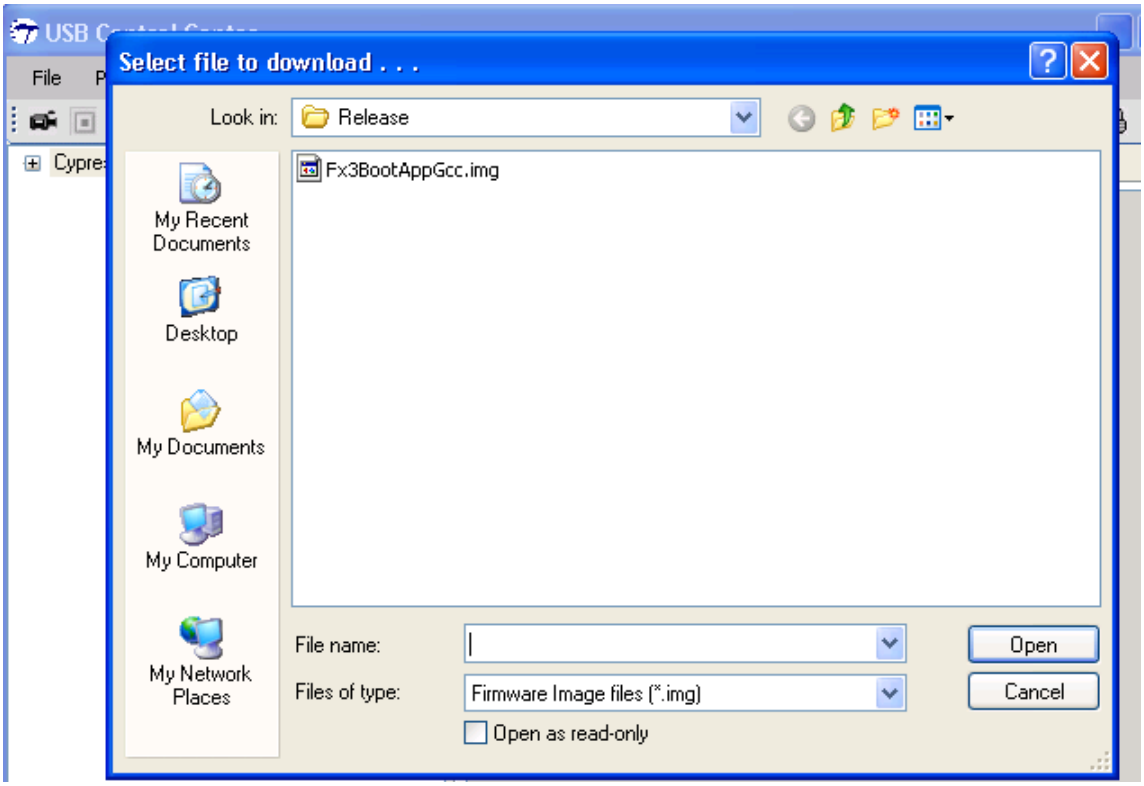
3. In Control Center, select **Program > FX3 > RAM**, as shown in Figure 3-9.

Figure 3-9. Select FX3 RAM from Control Center



4. Browse to the firmware image (.img) path to be programmed into the FX3 RAM. Double-click on the .img file, as shown in Figure 3-10.

Figure 3-10. Firmware Image Selection before Download



5. A **Programming Succeeded** message is displayed on the bottom left pane of the Control Center and the FX3 device re-enumerates with the programmed firmware.

3.7 I2C Boot

The FX3 DVK board provides I2C interface as one of the boot options. Mount an I2C EEPROM into U44 socket and verify this feature by programming the firmware image.

3.7.1 Download Firmware Image to I2C EEPROM

Follow the procedure outlined here to download the firmware image to I2C EEPROM:

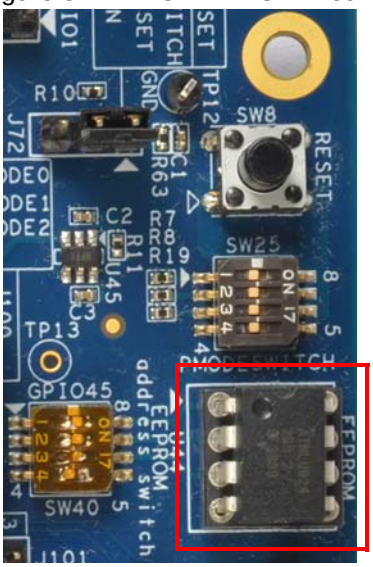
1. Disconnect the USB cable between the USB host PC and FX3 DVK board.
2. First enable USB boot, by setting the PMODE[2:0] pins to Z11. On the DVK board, this is done by configuring the jumpers and switches, as shown in Table 3-5.

Table 3-5. USB Boot Jumper and Dipswitch Settings

PMODE pin	Required PMODE Pin State	Jumper and Dip Switch Combination
PMODE2	Z	1. No jumper installed on J98 2. SW25.3 - Don't care (the switch can be either turned ON or OFF)
PMODE1	1	1. Short pins 2-3 of jumper J97 2. SW25.2 set to OFF
PMODE0	1	1. Short pins 2-3 of jumper J96 2. SW25.1 set to OFF

3. Verify the firmware image (.img) size before downloading to EEPROM. To select EEPROM of correct configuration, see Choosing Correct I2C EEPROM Configuration on page 30.
4. Mount the selected I2C EEPROM onto the FX3 board. Ensure the I2C EEPROM I/O pins are oriented and firmly fixed in U44 socket. Figure 3-11 shows a mounted I2C EEPROM into U44 socket.

Figure 3-11. I2C EEPROM Mounted into U44 Socket



5. Before attempting to program the EEPROM, ensure that the address signals of the EEPROM are configured correctly using the dip switch SW40 (for example, Microchip part 24AA1025, 1-8 ON, 2-7 ON, 3-6 OFF). Also, the I2C clock (SCL) and data line (SDA) jumpers J42 and J45 pins 1-2 should be shorted on the DVK board.

6. When connected to a USB host, the FX3 device enumerates in Control Center as "Cypress USB Bootloader".
7. In Control Center, select the FX3 device and then select **Program > FX3 > I2C EEPROM**. This causes a special I2C boot firmware to be programmed into the FX3 device, which then enables programming of the I2C device connected to FX3. Hence, now the FX3 device re-enumerates as "Cypress USB BootProgrammer".

Figure 3-12. I2C EEPROM Download Option in Control Center

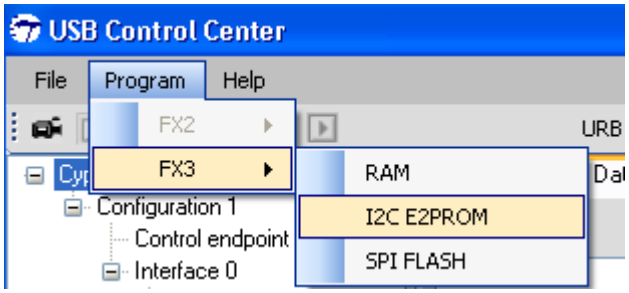


Figure 3-13. FX3 Re-enumerates as Cypress USB Boot Programmer

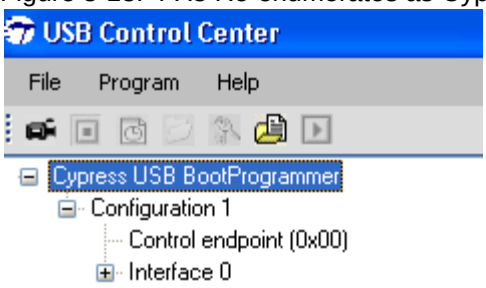
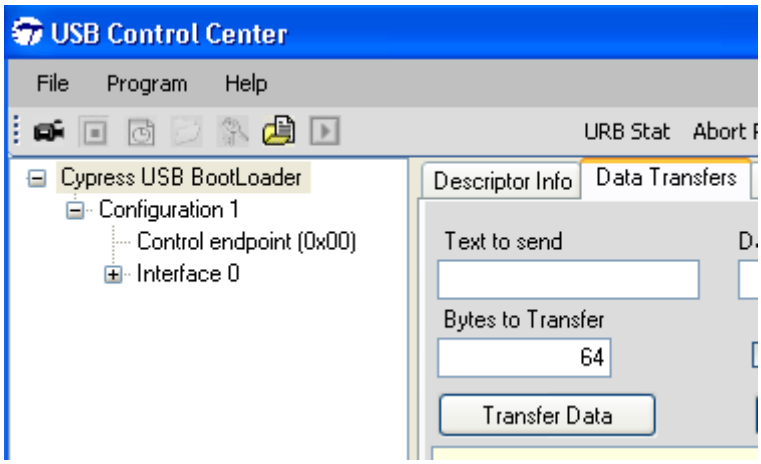


Figure 3-14. FX3 USB Bootloader Entry in Control Center



8. After the FX3 DVK board enumerates as "Cypress USB BootProgrammer", the Control Center application prompts the user to select the firmware binary to download. Browse to the relevant release mode firmware binary, as shown in Figure 3-15.

Figure 3-15. Select Firmware Image to Download

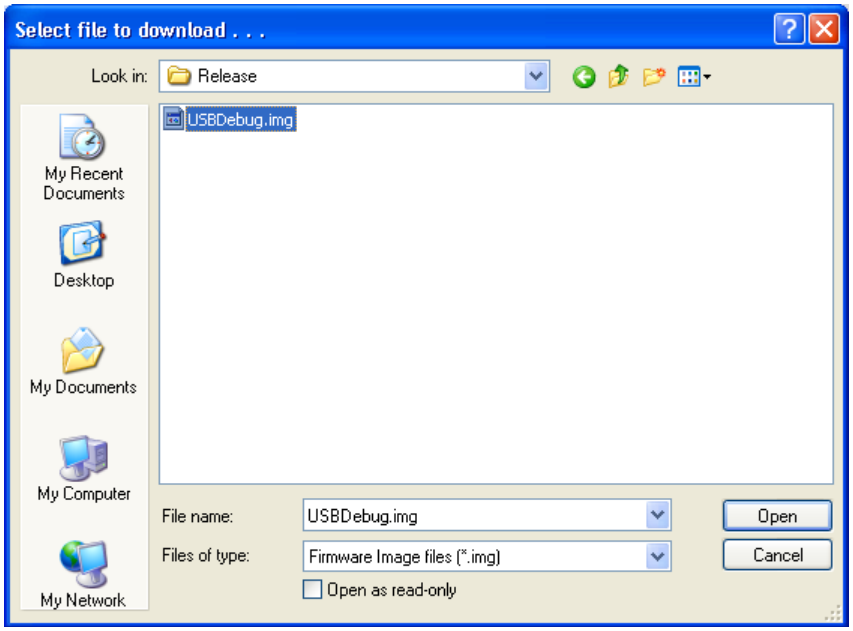
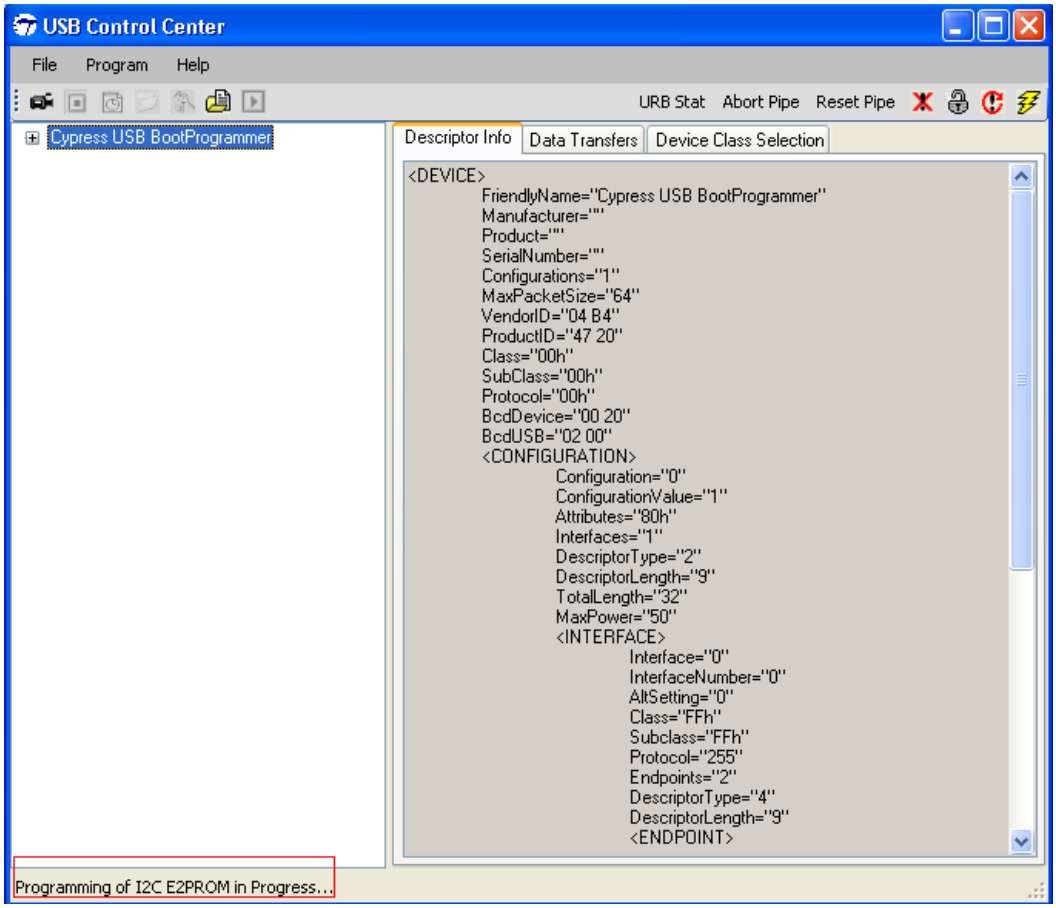


Figure 3-16. I2C EEPROM Programming Update in Control Center



9. The bottom left corner of the window displays "Programming of I2C E2PROM Succeeded".

3.7.2 Booting from I2C EEPROM

Change the PMODE pins on the DVK board to Z1Z to enable I2C boot. On the DVK board, this is done by configuring the jumpers and switches, as shown in Table 3-6. Press RESET button SW8. The FX3 DVK board re-enumerates with the boot image in I2C EEPROM.

Table 3-6. I2C Boot Jumper and Dip Switch Settings

PMODE)in	Required PMODE Pin State	Jumper and Dip Switch Combination
PMODE2	Z	1. No jumper installed on J98 2. SW25.3 - Don't care (the switch can be either turned ON or OFF
PMODE1	1	1. Short pins 2-3 of jumper J97 2. SW25.2 set to OFF
PMODE0	Z	1. No jumper installed on J96 2. SW25.1 - Don't care (the switch can be either turned ON or OFF)

3.7.3 Choosing Correct I2C EEPROM Configuration

Before mounting I2C EEPROM into U44 socket, the following factors needs to be considered to select EEPROM of correct configuration for the respective size of the firmware image.

- Verify if the I2C EEPROM can accommodate the firmware image. Check the addressing (A0, A1, and A2) mechanism for the EEPROM part and select these pins on board using SW40. Follow the steps to decide on the addressing mechanism for different configurations of EEPROM.
  - The typical EERPOM pins consists of A0, A1, and A2 address lines. EZ-USB FX3 can address eight EEPROM(000-111). If the firmware image size is less than 32 KB, then the corresponding address line can be pulled low/high. If multiple firmware image need to be booted, then connect multiple EEPROMs (for example, 24LC256) can be connected and incremental addresses can be assigned to each EEPROM. The address selection can be done using dip switch SW40.

Figure 3-17. EEPROM Block Diagram

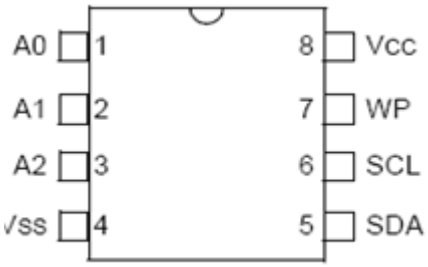
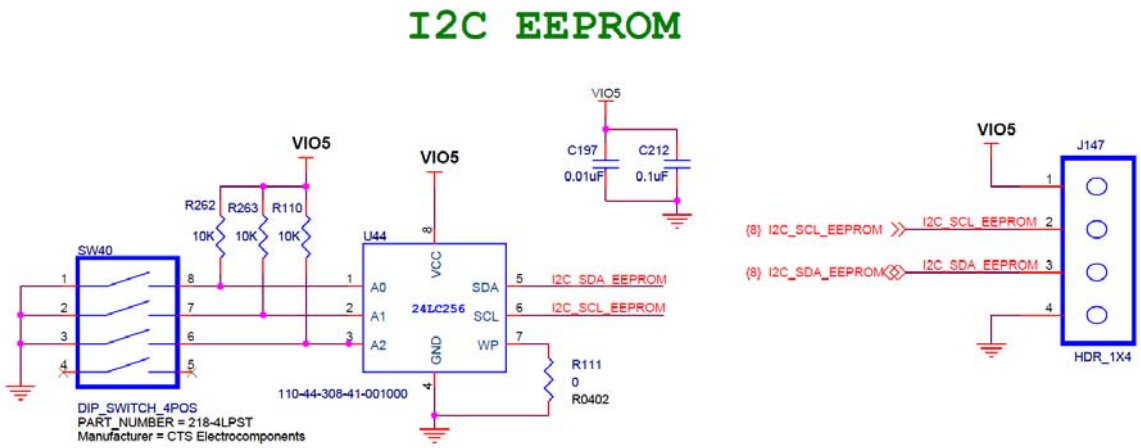


Figure 3-18. SW40 and EEPROM Interconnect



- If firmware output image size is 60 KB, then use two EEPROMS (Microchip: 24LC1024 or Atmel 24C1024). For larger images, equivalent large EEPROMS can be used. Microchip uses A0 and A1 for chip select and A2 is unused for 24LC1024 (128KB) EEPROMS. Atmel EEPROMS use A2 and A1 for chip select and A0 is unused. Table 3-7 shows addressing scheme for four 128-KB EEPROMS from Microchip or Atmel.

Table 3-7. Address Configuration for 128KB EEPROM

Device No	Address Range	Microchip (24LC1024)			Atmel (24C1024)		
		A2	A1	A0	A2	A1	A0
1	0x00000-0x1FFFF	Vcc	0	0	0	0	NC
2	0x20000-0x3FFFF	Vcc	0	1	0	1	NC
3	0x40000-0x5FFFF	Vcc	1	0	1	0	NC
4	0x60000-0x7FFFF	Vcc	1	1	1	1	NC

Note NC indicates No Connection

- The firmware image has a fixed header format to boot from I2C EEPROM. After 2-byte "CY" signature, the third byte bits[3:1] can be used to specify the EEPROM size format. The parameter "bImageCTL" should be modified to reflect the correct EEPROM size. For 128 KB/64 KB Atmel EEPROMs, bit[3:1] should be 6; for 128 KB Microchip EEPROM (24LC1024), it should be 7.
- As shown in the above dip switch Figure 3-18, if the A[2:0] pins are set to 'ON' (closed position), the corresponding address line is connected to ground. If the switch position is set to OFF (open position), the corresponding address line is pulled high to VIO5 voltage level.

Note For 24LC1026 Microchip EEPROM, follow same addressing mechanism as Atmel EEPROM See AN68914 - EZ-USB FX3 I2C Boot Option for more details on I2C booting mechanism.

3.8 SPI Boot

The FX3 DVK board can boot through SPI interface if PMODE [2:0] pins are set to 0Z1.To verify the SPI boot functionality, initially the firmware needs to be programmed to onboard flash memory (M25P40) using the Control Center. Then, the DVK board needs to be reset to boot the image from the flash memory.

3.8.1 Download Firmware Image to SPI Flash

Follow this procedure to download the firmware image to SPI flash:

1. Disconnect the USB cable between the USB host PC and FX3 DVK board.
2. Enable USB boot, by setting the PMODE[2:0] pins to Z11. On the DVK board, this is done by configuring the jumpers and switches as shown in Table 3-8.

Table 3-8. USB Boot Jumper and Dip Switch Settings

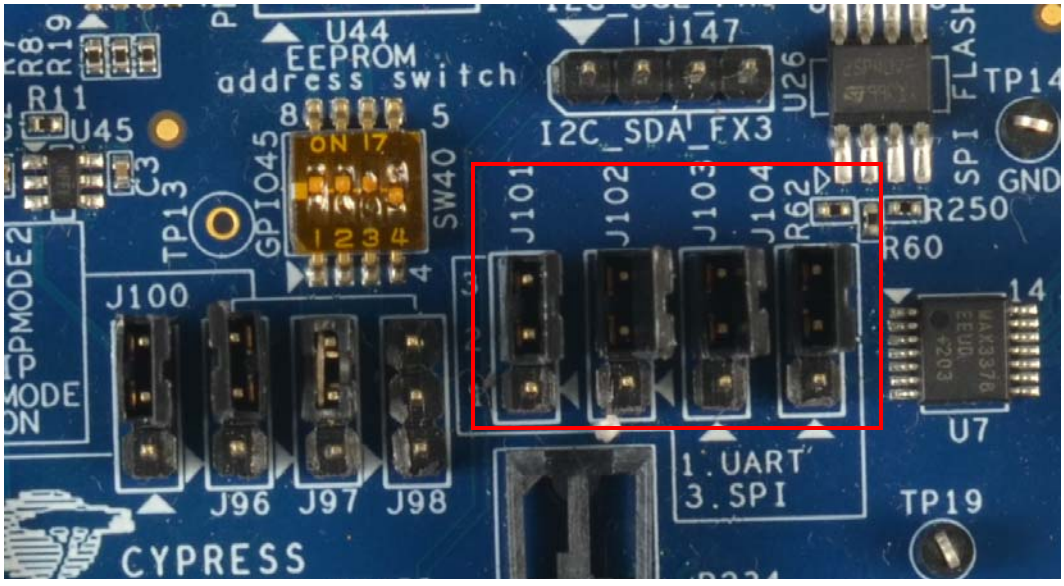
PMODE Pin	Required PMODE Pin State	Jumper and Dip Switch Combination
PMODE2	Z	1. No jumper installed on J98 2. SW25.3 - Don't care (the switch can be either turned ON or OFF)
PMODE1	1	1. Short pins 2-3 of jumper J97 2. SW25.2 set to OFF
PMODE0	1	1. Short pins 2-3 of jumper J96 2. SW25.1 set to OFF

3. When the FX3 DVK board is connected to a USB PC host using the USB 3.0 cable, the FX3 device enumerates in Control Center as "Cypress USB Bootloader".
4. Verify the SPI mode related jumper settings, as shown in Table 3-9. These jumpers are used to select either UART or SPI I/O pins.

Table 3-9. SPI Mode Jumper Settings

Sl. No.	Pin Description	Jumper Position
1	J101-SPI_CLK	2-3
2	J102-SPI_SSN	2-3
3	J103-SPI_MISO	2-3
4	J104-SPI_MOSI	2-3

Figure 3-19. Select SPI Interface using Jumpers (J101-J104)



Note The white arrows on the DVK board indicate pin1 of the corresponding jumper.

5. In Control Center, select **Program > FX3 > SPI FLASH**. The FX3 bootloader does not support firmware download to SPI flash programming by default. The Control Center application downloads the relevant code, which supports firmware download to the flash immediately when user selects the SPI FLASH option in Control Center. The FX3 DVK board re-enumerates again as "Cypress USB BootProgrammer", as shown in Figure 3-20.

Figure 3-20. Select SPI Flash Download in Control Center

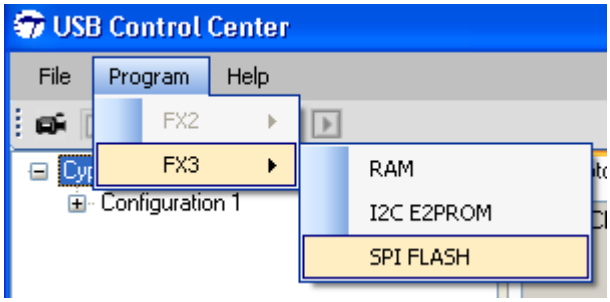
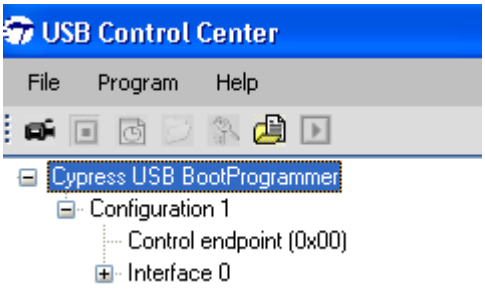
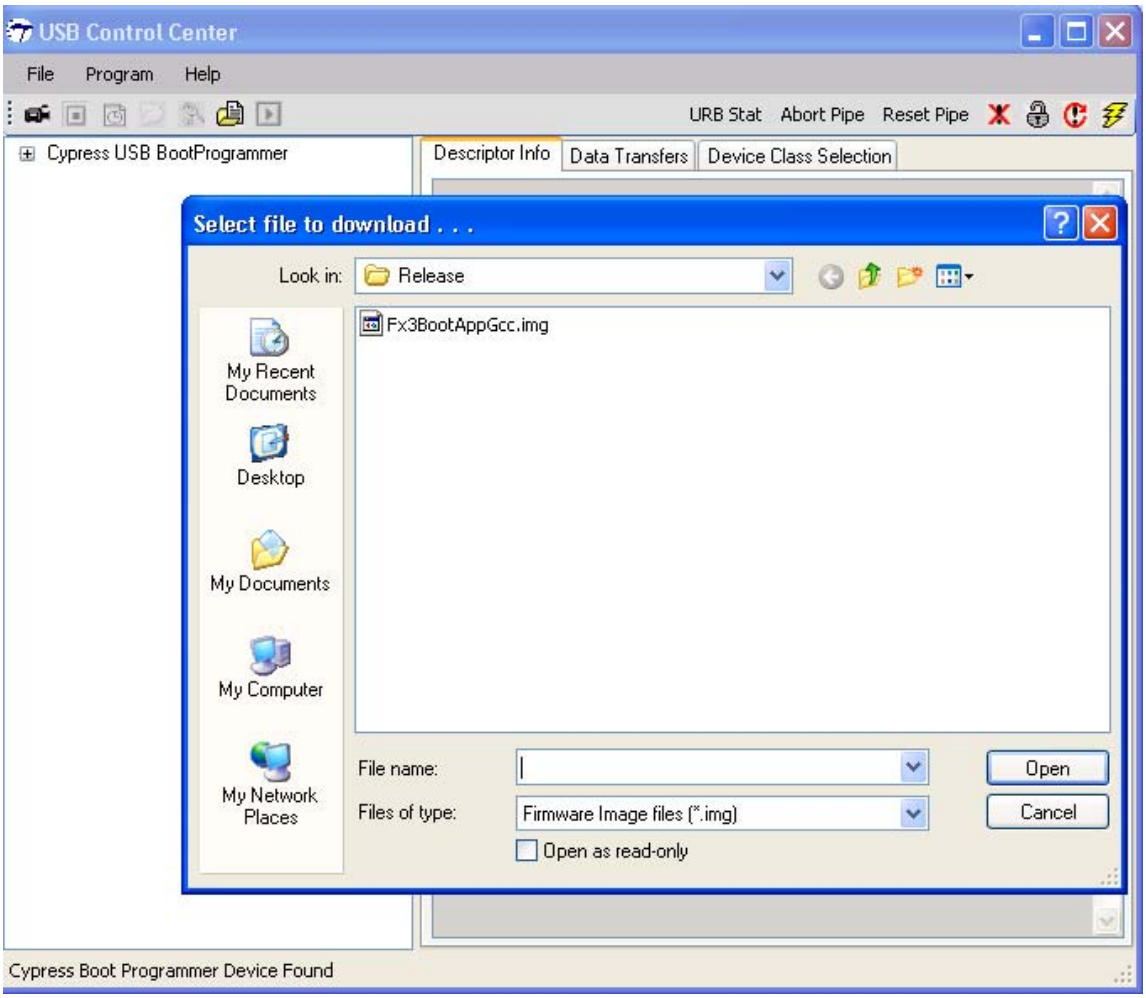


Figure 3-21. Device after Programmer Firmware Update



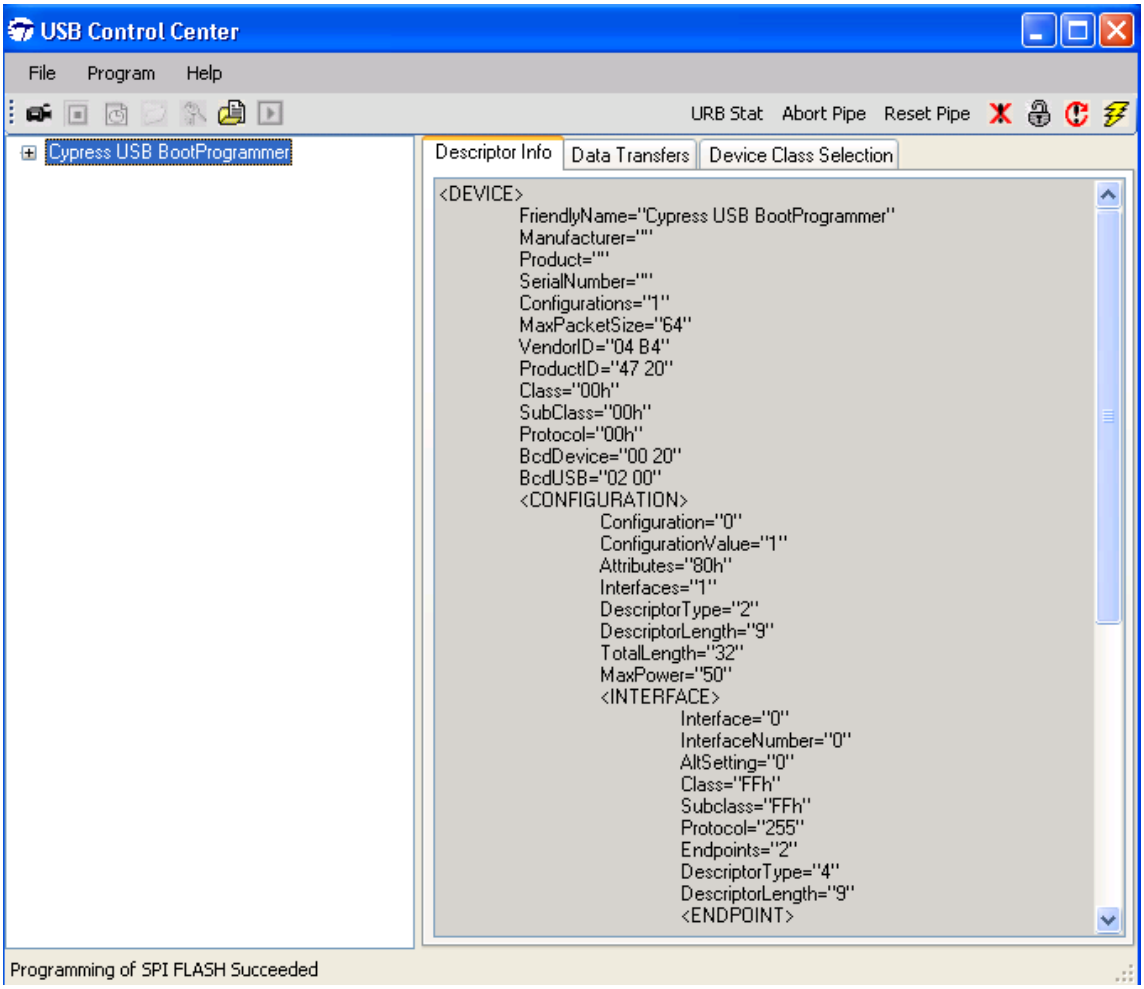
6. After the FX3 DVK board enumerates as "Cypress USB BootProgrammer", the Control Center application prompts the user to select the firmware binary to download. Browse to the relevant release mode firmware binary (.img) and select the image.

Figure 3-22. Select Firmware Image to Download to SPI Flash



7. The application then starts downloading the firmware to the flash memory. The downloading process status can be observed at the bottom left corner of the Control Center application, as shown in Figure 3-23.

Figure 3-23. SPI Firmware Download Update in Control Center



3.8.2 Booting from SPI Flash

After programming firmware to SPI Flash, set PMODE [2:0] pins to 0Z1 to enable SPI boot. On the DVK board, this is done by configuring the jumpers and switches, as shown in Table 3-10.

Table 3-10. SPI Boot Jumper and Dip Switch Settings on FX3 DVK Board

PMODE Pin	Required PMODE Pin State	Jumper and Dip Switch Combination
PMODE2	0	1. Short pins 2-3 of jumper J98 2. SW25.3 set to ON
PMODE1	Z	1. No jumper installed on J97 2. SW25.2 - Don't care (the switch can be either turned ON or OFF)
PMODE0	1	1. Short pins 2-3 of jumper J96 2. SW25.1 set to OFF

Press RESET button SW8. This prompts the FX3 device to monitor the PMODE I/O pins again and boot from the SPI interface.

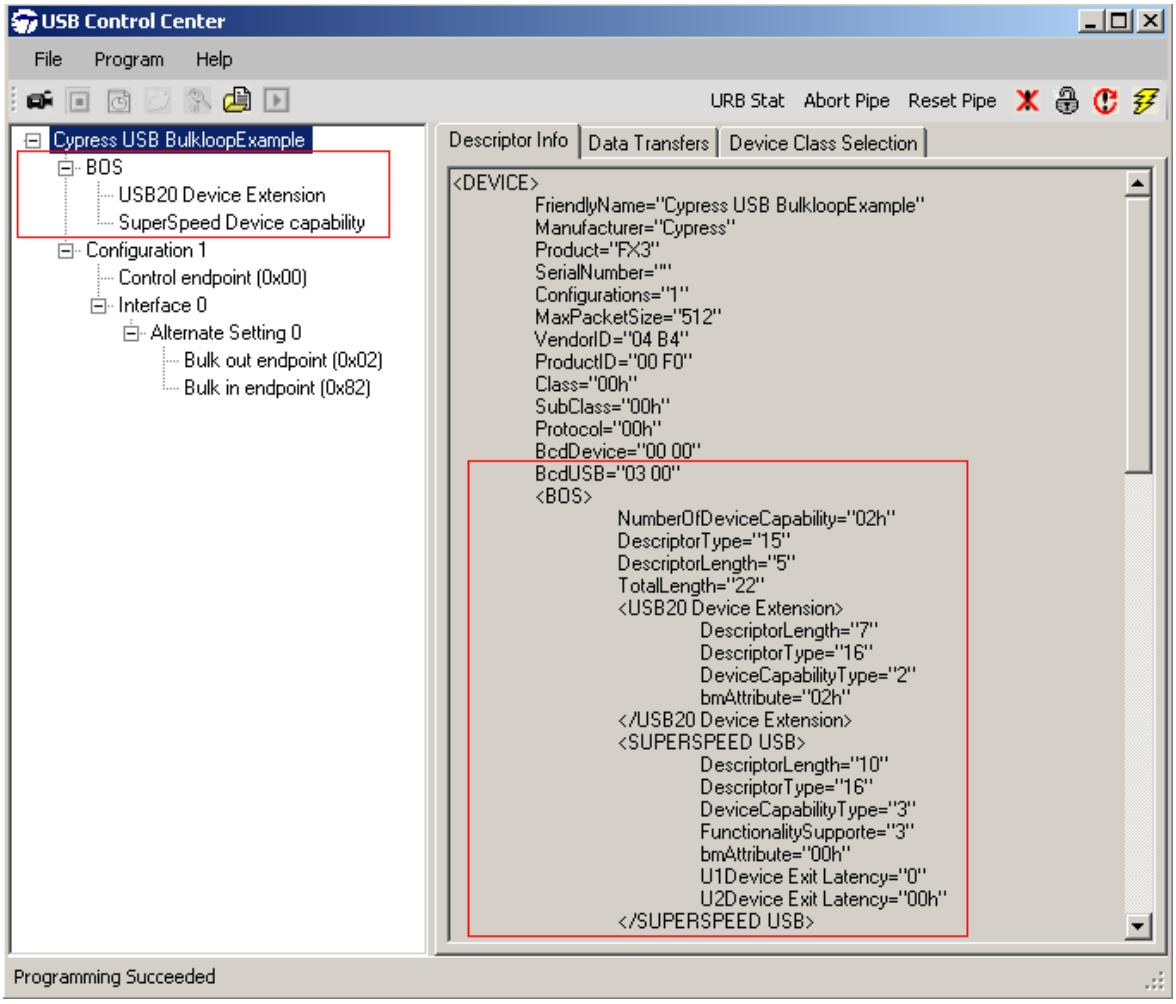
3.9 ADMUX Boot

The FX3 device can boot from synchronous and asynchronous ADMux interface if PMODE [2:0] pins are set to Z00 and Z01, respectively. For more details, see application note AN76405 - EZ-USB FX3 Boot Options.

3.10 USB 3.0 Enumeration

The FX3 DVK board enumerates as a USB 2.0 device using the default bootloader in ROM. To enumerate as USB 3.0 device, the firmware image needs to be downloaded to the FX3 board. The PC host should contain a USB 3.0 capable host controller. The firmware examples can work in USB 2.0 device mode if the PC host contains USB 2.0 capable ports. To get better data transfer rate for the end applications, a USB 3.0 capable host PC should be used. After firmware download, the Control Center shows the list of descriptors (Figure 3-24) for a USB 3.0 device.

Figure 3-24. FX3 USB 3.0 Device Descriptor Sample in Control Center



3.11 Testing Bulkloop Firmware Examples

The SuiteUSB applications are available at `C:\Program Files\Cypress\Cypress USBSuite \bin`. The Bulkloop application is designed in both C# and C++ framework. Follow these steps to test the Bulkloop firmware(CyBulkloopAuto) example:

1. Download the firmware binary image (.img) by following the steps outlined in section "Loading USBBulkLoopAuto Firmware " from the document *Getting Started with FX3.pdf*.
2. If the Windows OS prompts for USB driver, follow steps 1 to 13 outlined in section "Installing the Driver for Bulk Loop Firmware" of *Getting Started with FX3.pdf*.
3. Click on the Bulkloop C# application and verify the loopback functionality .The procedure is outlined in section "Verifying with Bulk Loop Tool" of the same document.

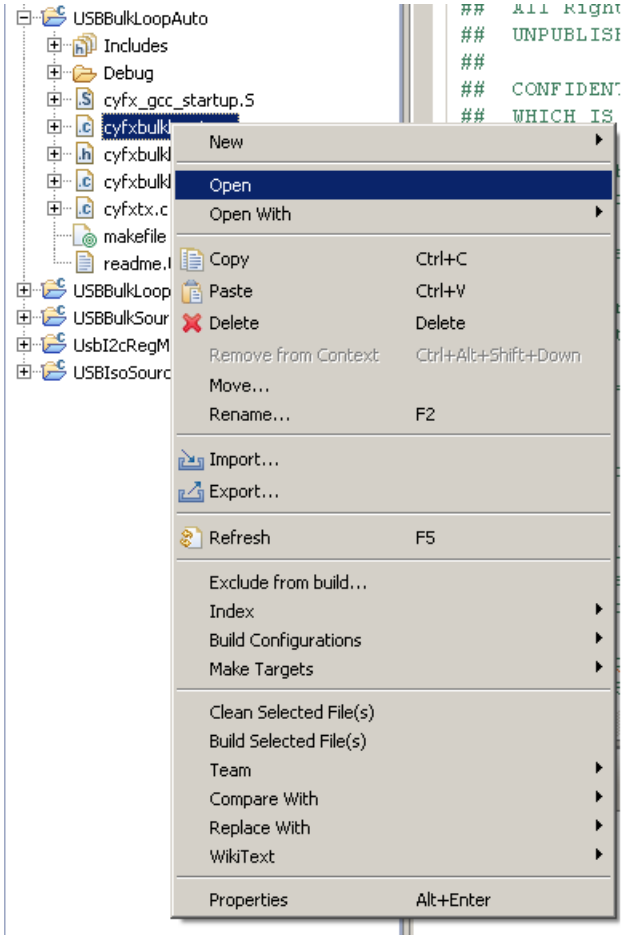
3.12 Modifying Bulkloop Firmware Example

The firmware examples can be modified and recompiled using Eclipse IDE. The modified firmware binary can be later downloaded to FX3 RAM and tested using a SuiteUSB application. To demonstrate the modification, CyBulkloopAuto firmware example of FX3 SDK is used. The bulk endpoint number and its associated socket is modified here to simply demonstrate how to modify, recompile the firmware example, and test the functionality using a PC application.

Follow the procedure outlined in section 11.2.2.1 Importing Eclipse Projects of *FX3 Programmers Manual.pdf* and import all firmware projects into Eclipse IDE.

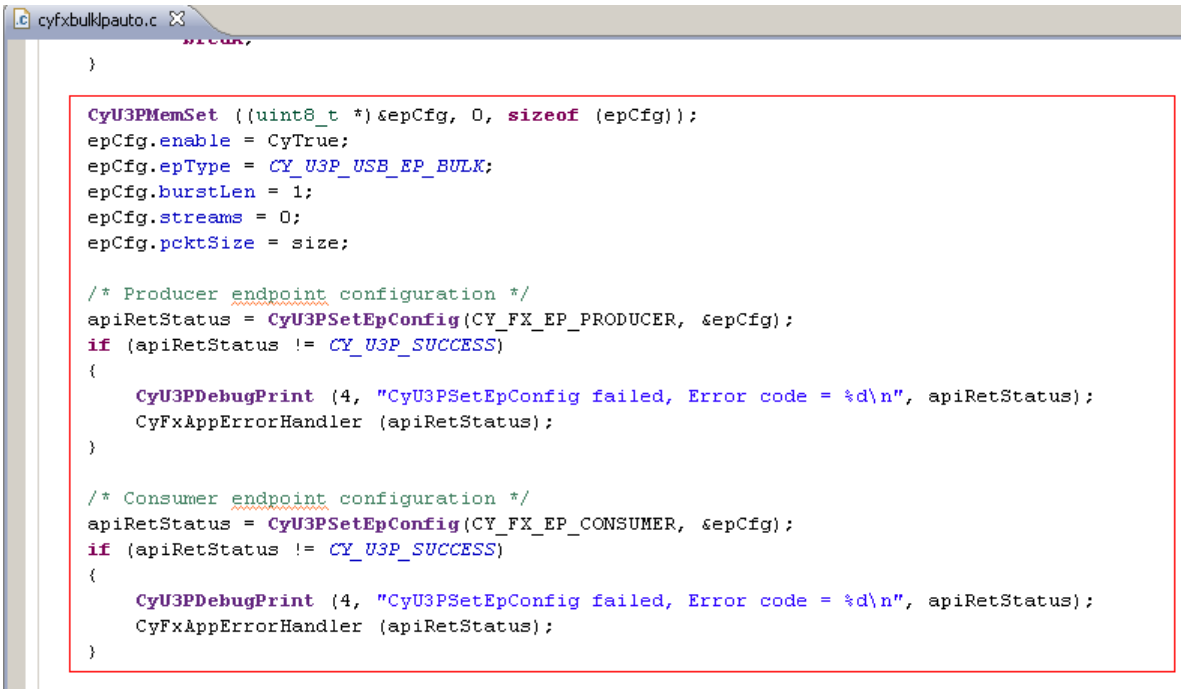
Select the **CyBulkloopAuto** example in the left window of the Eclipse IDE, as shown in Figure 3-25. In the list of .c files, select cyBulklpauto.c.

Figure 3-25. Selecting cyfxbulklpauto.c in Eclipse IDE



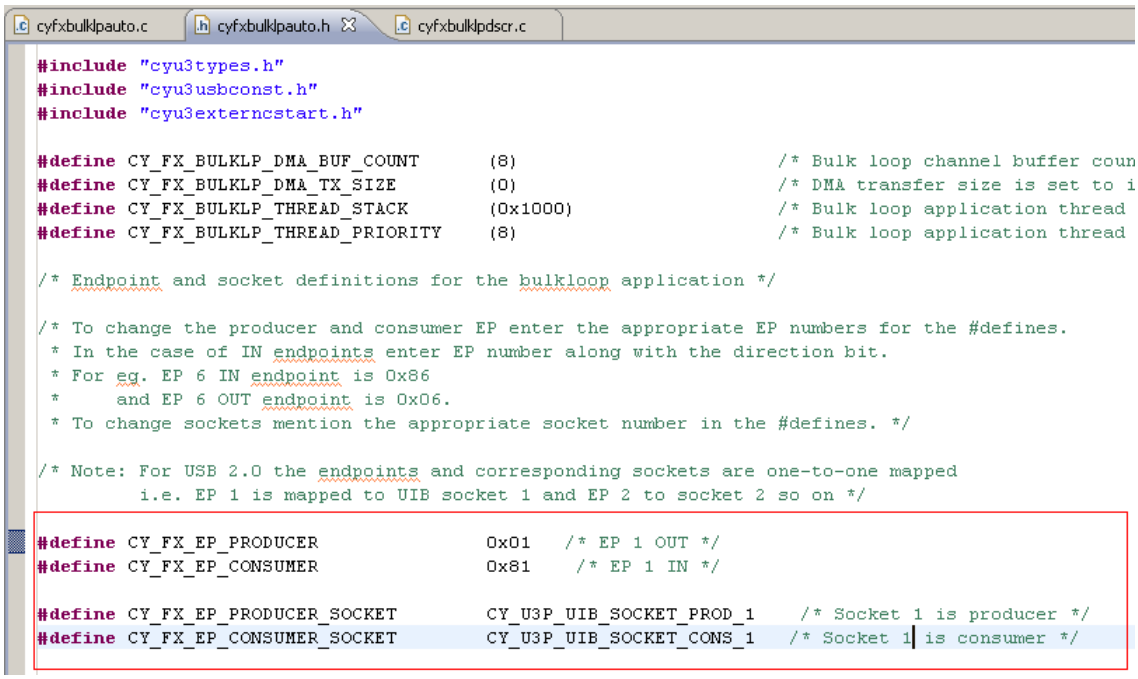
In this example, the bulk endpoint EP1 handles both IN and OUT transfers. The endpoint configuration is performed in CyFxBulkAppInStart() function. The code snippet from Cyfxbulklpauto.c, shown in Figure 3-26, displays how EP1 is configured as bulk endpoint type using API CyU3PSetEpConfig().

Figure 3-26. Selecting cyfxbulklpauto.c in Eclipse IDE



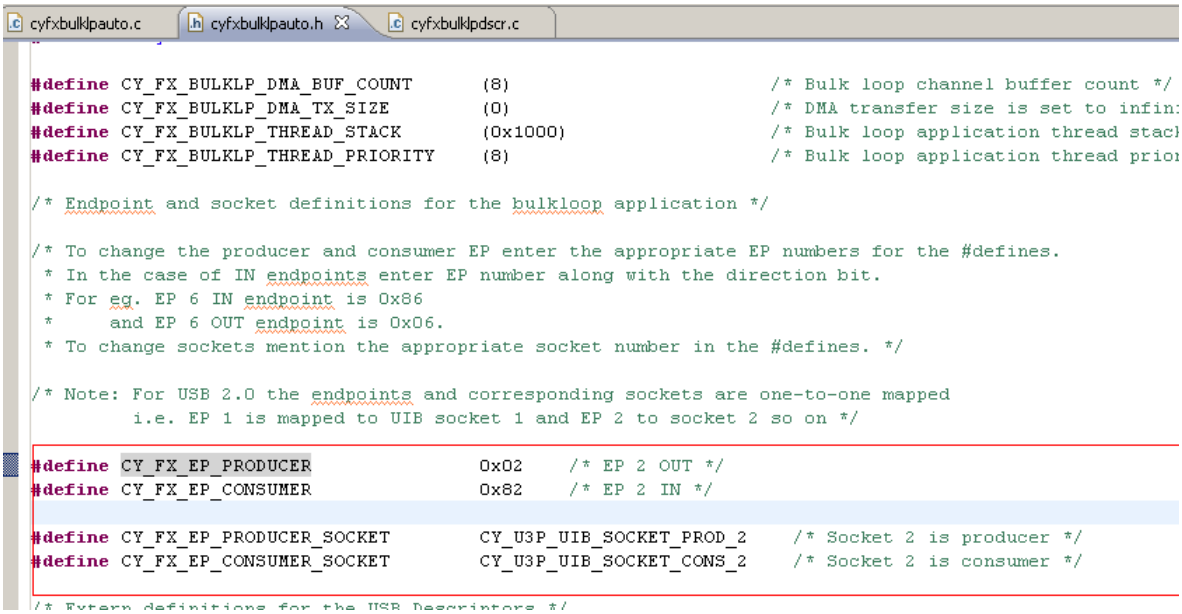
The endpoint number and their associated socket are defined in .h using macros CY_FX_EP_PRODUCER, CY_FX_EP_CONSUMER, CY_FX_EP_PRODUCER_SOCKET, and CY_FX_EP_CONSUMER_SOCKET. Figure 3-27 shows the code snippet defined in this file.

Figure 3-27. Endpoint and Socket Definitions in cyfxbulklpauto.h



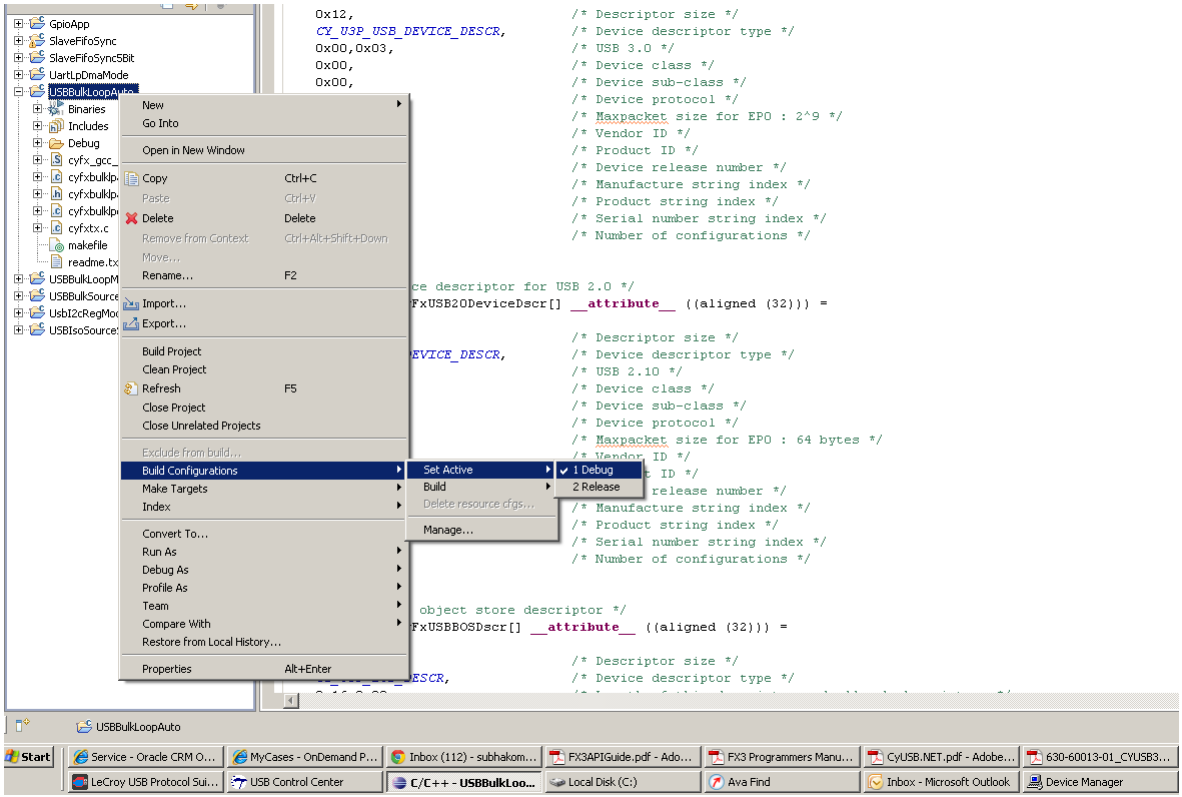
Using these macros, the endpoint number and associated socket are changed from EP1 to EP2 (Figure 3-28). Now EP2 acts as Bulk IN and OUT endpoint in the modified firmware.

Figure 3-28. Endpoint Number and Associated Socket Modified from EP1 to EP2



Compile the modified firmware example, as shown in Figure 3-29. Build the firmware example in both Debug and Release modes by selecting **Build Configuration > Set Active > Debug/Release**.

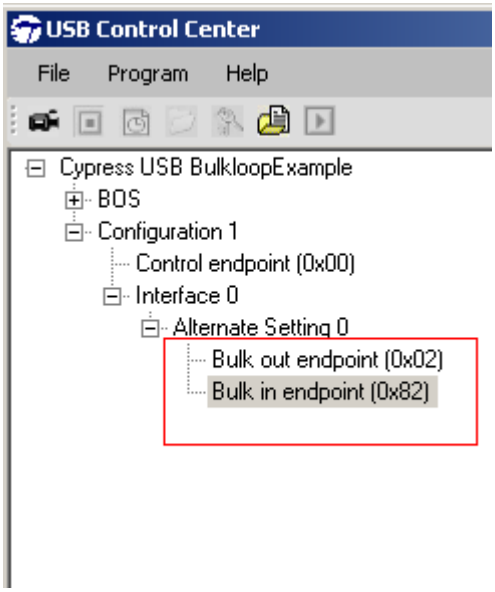
Figure 3-29. Selecting Build Configuration in Eclipse IDE



Note The firmware image size in Release mode is of lesser size compared to Debug mode image. The Debug mode image contains additional debug symbols to allow the user to perform step-by-step debugging using JTAG hardware.

Follow the procedure outlined in USB Boot on page 24 to download firmware image to to FX3 RAM using Control Center. The FX3 DVK board enumerates as "Cypress USB BulkloopExample" in Control Center. Observe the modified firmware example shown EP2 as both Bulk IN and OUT in Control Center.

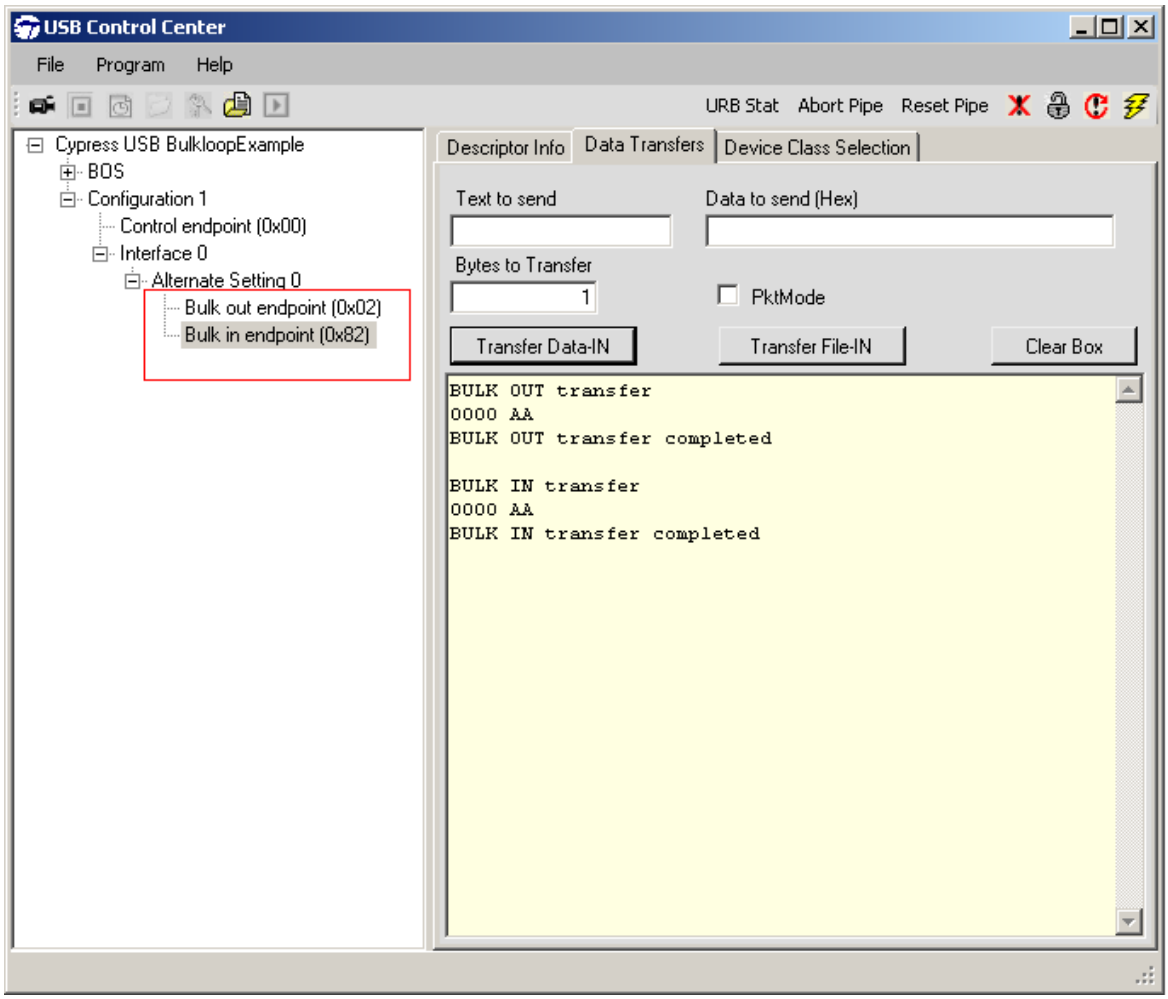
Figure 3-30. Bulkloop Device with EP2 in Control Center



Note If the FX3 device is not displayed as mentioned above, then manually bind the FX3 device to cyusb3.sys driver at `C:\Program Files\Cypress\EZ-USB FX3 SDK \<SDK_version>\driver\bin`. Choose the driver for the corresponding Windows platform in 32-bit (x86) or 64-bit (x64) OS configuration. See First Time USB Enumeration on page 19 to know how to bind the USB driver manually.

Select "Bulk out endpoint (0x2)" in the left pane of Control Center. Enter **Data to send = "AA"**, **Bytes to Transfer Length = 1**, and click **Transfer Data-OUT** button. To receive the same data on EP2 IN, select **Bulk in endpoint (0x82)**, **Bytes to Transfer Length = 1**, and click on **Transfer Data-IN** button to receive the same data. Figure 3-31 summarizes the whole sequence.

Figure 3-31. Bulkloop OUT and IN transfer on EP2



3.12.1 Debugging Firmware Example

The Segger J-Link probe is the preferred JTAG probe for the FX3 SDK. This probe, along with the Segger J-Link ARM GDB server, is used for debug. The Eclipse IDE will connect to the J-link GDB server when debugging your firmware. To get Eclipse working with the GDB server, you need to create a debug configuration for the J-link. Details on how to do this is available in FX3 programmers Manual, section 12.2.2.3 Executing and Debugging.

4. Hardware

This chapter describes the different hardware interfaces available on FX3 DVK. It explains the following hardware components of the DVK board:

- Power supply
- USB 3.0 Micro-B receptacle connector
- Clocking mechanism using on-board crystal
- GPIFII connector
- Reset circuit
- Serial interfaces – I2S, SPI, UART, and JTAG.

4.1 Power Supply

The FX3 DVK board can be powered in two ways:

- Self power: Use the external 5-V power adapter supplied with the kit to power the DVK board.
- Bus power: The board can be powered using the USB 3.0 cable. Connect the USB cable supplied with the kit between the PC host and J48 USB connector on board. The PC host supplies power to the DVK board through the USB cable. Short pins 1-2 of jumper J53 in bus-powered mode.

FX3 supports several I/O interfaces including I2C, SPI, GPIF II, and I2S. These interfaces are capable of operating at different voltage levels. The voltage levels for FX3 I/O power domains can be selected using on-board jumpers. Table 4-1 shows the power domain selection option through jumpers.

Table 4-1. Power Supply Domain

Power Domain	Jumper	Jumper positions	Voltage levels
VIO1	J136	1-6(labeled V1P8 on board)	1.8 V
		2-5(labeled V2P5 on board)	2.5 V
		3-4(labeled V3P3 on board)	3.3 V
VIO2	J144	1-6(labeled V1P8 on board)	1.8 V
		2-5(labeled V2P5 on board)	2.5 V
		3-4(labeled V3P3 on board)	3.3 V
VIO3	J145	1-6(labeled V1P8 on board)	1.8 V
		2-5(labeled V2P5 on board)	2.5 V
		3-4(labeled V3P3 on board)	3.3 V
VIO4	J146	1-6(labeled V1P8 on board)	1.8 V
		2-5(labeled V2P5 on board)	2.5 V
		3-4(labeled V3P3 on board)	3.3 V

Table 4-1. Power Supply Domain

Power Domain	Jumper	Jumper positions	Voltage levels
VIO5	J134	1-8(labeled V1P2 on board)	1.2 V
		2-7(labeled V1P8 on board)	1.8 V
		3-6(labeled V2P5 on board)	2.5 V
		4-5(labeled V3P3 on board)	3.3 V
REG_VBATT	J143	1-6(labeled V2P5 on board)	2.5 V
		2-5(labeled V3P3 on board)	3.3 V
		3-4(labeled V5P0 on board)	5 V
CVDDQ	J135	1-2(labeled V1P8 on board	1.8 V
		3-4 (labeled V3P3 on board)	3.3 V

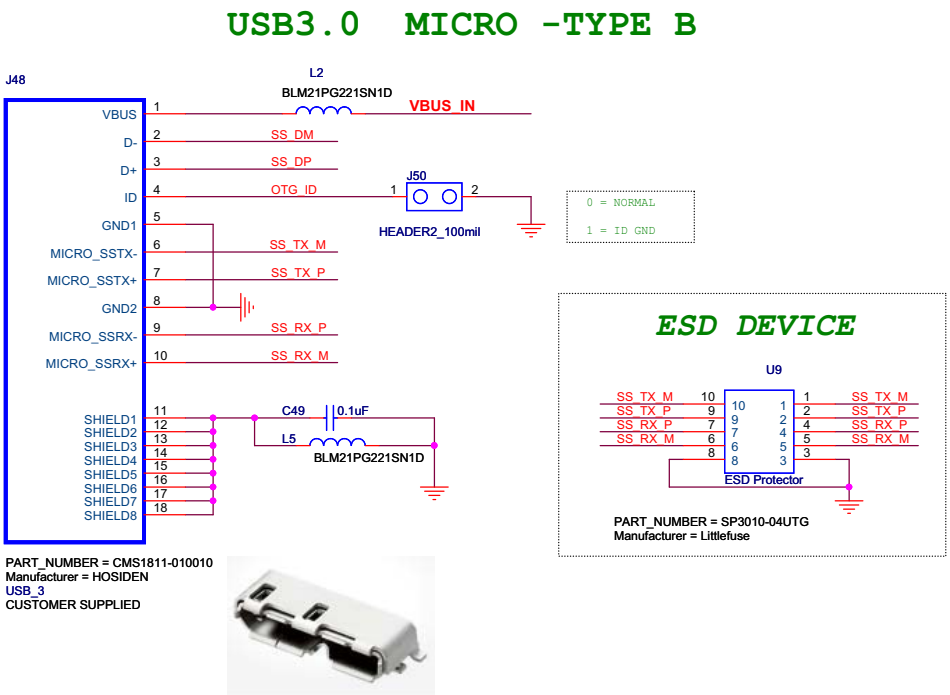
Notes

- The FX3 device also has additional power domains (U3TXVDDQ, U3RXVDDQ, AVDD, and VDD). These are fixed at 1.2 V.
- Refer to Pin Description section of the EZ-USB FX3 SuperSpeed USB Controller datasheet for details on how the I/O in each power domain can be configured.

4.2 USB Receptacle

A standard Micro B receptacle is used on the FX3 DVK board. Figure 4-1 shows the USB 3.0 pins (SS_TX_M, SS_TX_P, SS_RX_P, and SS_RX_M) and USB 2.0 pins (OTG_ID, D+, and D–) available on the J48 USB connector. The FX3 DVK board can be bus-powered using VBUS pin on the connector. The USB3.0 and USB2.0 lines go through an ESD protection device for additional ESD protection.

Figure 4-1. USB3.0 Micro B Connector and ESD Device



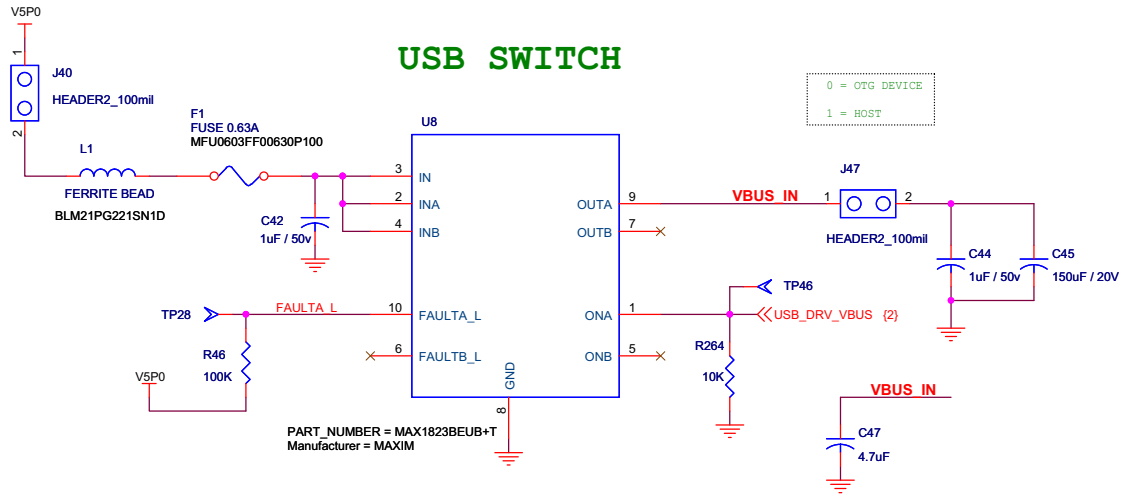
The DVK board supports USB 3.0 peripheral and Hi-Speed OTG functionality using the J48 connector. Table 4-2 shows the jumper settings required to operate in USB 2.0 OTG device or Host mode.

Table 4-2. OTG/Host Mode Jumper Settings

Jumper	Pin Position	Mode: Description
J40	Short	OTG Host: To supply VBUS to connected peripheral as OTG Host from the power adapter(V5P0)
J50	Short	OTG Host: When connecting a USB device to FX3 OTG Host with a USB 2.0 Micro-B-to-A receptacle cable the OTG_ID pin can be grounded using this jumper. FX3 then acts as OTG Host by default.
	Open	OTG Device: The OTG_ID pin state depends upon the pin state at J48. The role of FX3 (as OTG A or B device) depends on the OTG cable (Micro-A to Micro-B type) orientation.
J47	Short	OTG Host: To select proper capacitance on VBUS between OTG host manually short this VBUS_IN jumper.
	Open	OTG device: To select proper capacitance on VBUS as OTG B-device.
J100	1-2	Routes the CTL_4 signal to Samtec connector(GPIF II interface)
	2-3	OTG Host/OTG Device: USB_DRV_VBUS line connected to CTL_4 signal. The I/O line can turn the power (VBUS_IN) ON or OFF using firmware.

To operate the DVK in USB 3.0 peripheral mode, refer to default jumper settings mentioned in Table 3-1 on page 17 and follow the procedure outlined in section USB 3.0 Enumeration on page 36

Figure 4-2. OTG and Host Mode Power Configuration Jumpers



4.3 Clocking Mechanism for FX3

FX3 allows either a crystal to be connected between the XTALIN and XTALOUT pins or an external clock to be connected at the CLKIN pin. The XTALIN, XTALOUT, and CLKIN I/O pins can be left unconnected if they are not used.

Crystal frequency supported is 19.2 MHz, while the external clock frequencies supported are 19.2, 26, 38.4, and 52 MHz.

The FSLC[2:0] pins must be configured appropriately to select the crystal- or clock-frequency option. The configuration options are shown in Table 4-3.

Table 4-3. Crystal/Clock Frequency Selection

FSLC[2]	FSLC[1]	FSLC[0]	Crystal/Clock Frequency
0	0	0	19.2-MHz crystal
1	0	0	19.2-MHz input CLK
1	0	1	26-MHz input CLK
1	1	0	38.4-MHz input CLK
1	1	1	52-MHz input CLK

On the DVK board, the clock for the FX3 device is provided through an on-board 19.2-MHz crystal that is connected to the XTALIN and XTALOUT pins of FX3. The FSLC[2:0] lines of FX3 are tied to ground; which means that the device is configured to only use the 19.2 MHz crystal for clocking.

Figure 4-3. Crystal Circuit

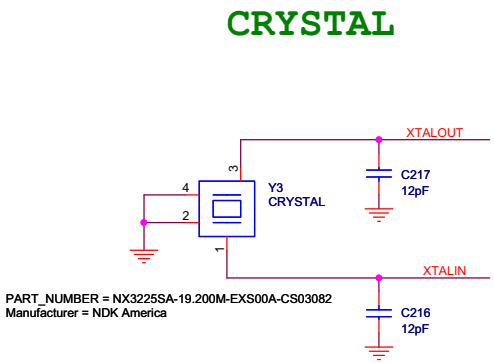
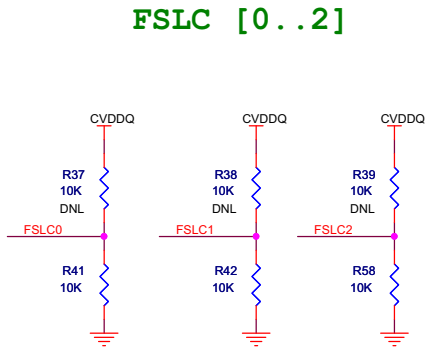


Figure 4-4. FSCL[0:2] Lines Pulled to Ground



Note For the 19.2-MHz crystal option, FSLC [0:2] pins are tied to ground and hence R37, R38, and R39 are not populated on the board.

4.4 GPIF II Connector

FX3 offers a high-performance General Programmable Interface, GPIF II. This interface enables functionality similar to but more advanced than FX2LP's GPIF. The GPIF II is a programmable state machine that enables a flexible interface, which functions either as a master or slave in industry standard or proprietary interfaces. Both parallel and serial interfaces are implemented with GPIF II.The DVK board consists of a Samtec expansion connector to interface with external processors, ASICs, DSPs, or FPGAs. The GPIF II lines going to the Samtec connector also come out on debug mictors, J1 and J2. Logic analyzer pods can be connected to J1 and J2 for debugging.

Figure 4-5 shows the Samtec expansion connector on the FX3 DVK board.

Figure 4-5. Samtec and Mictor Debug Connectors on DVK

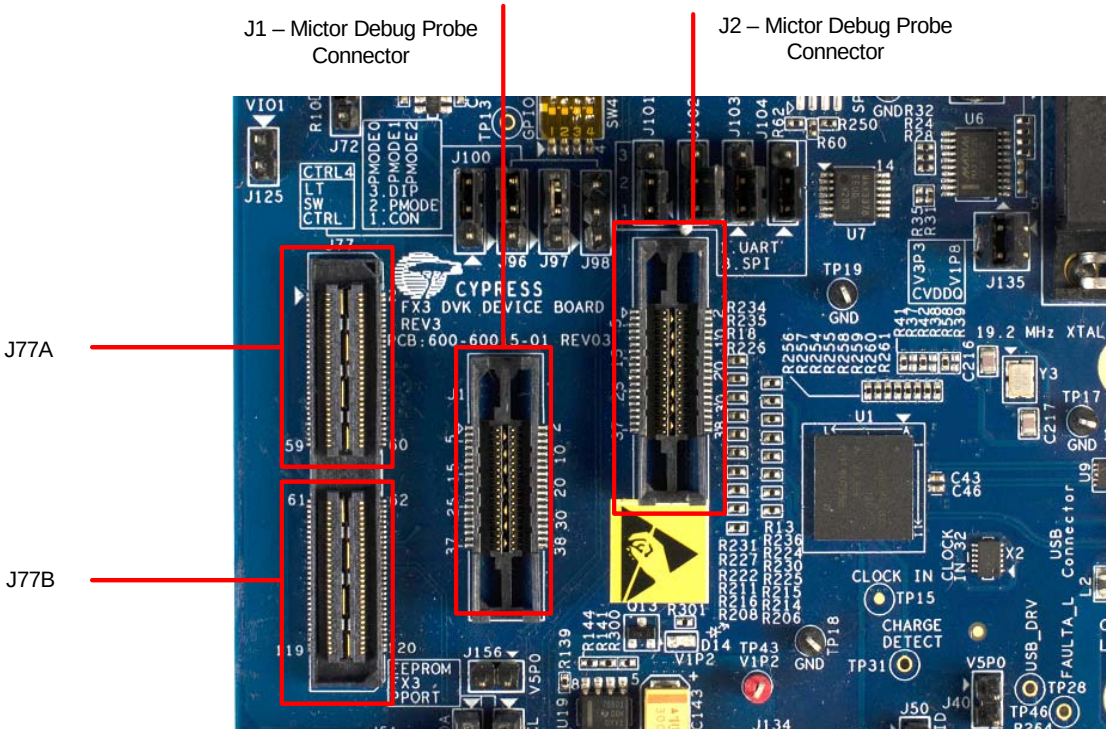


Figure 4-6 and Figure 4-7 show the schematic view of Samtec expansion connector with GPIF II signals.

Figure 4-6. Samtec Expansion Connector Circuit1

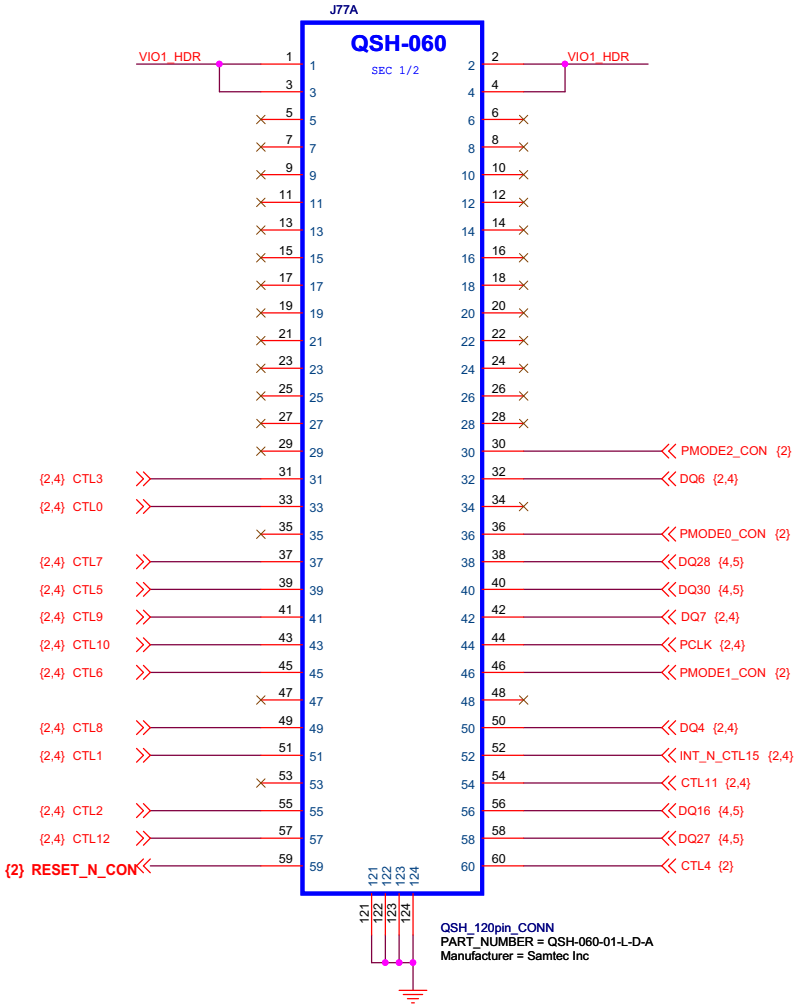
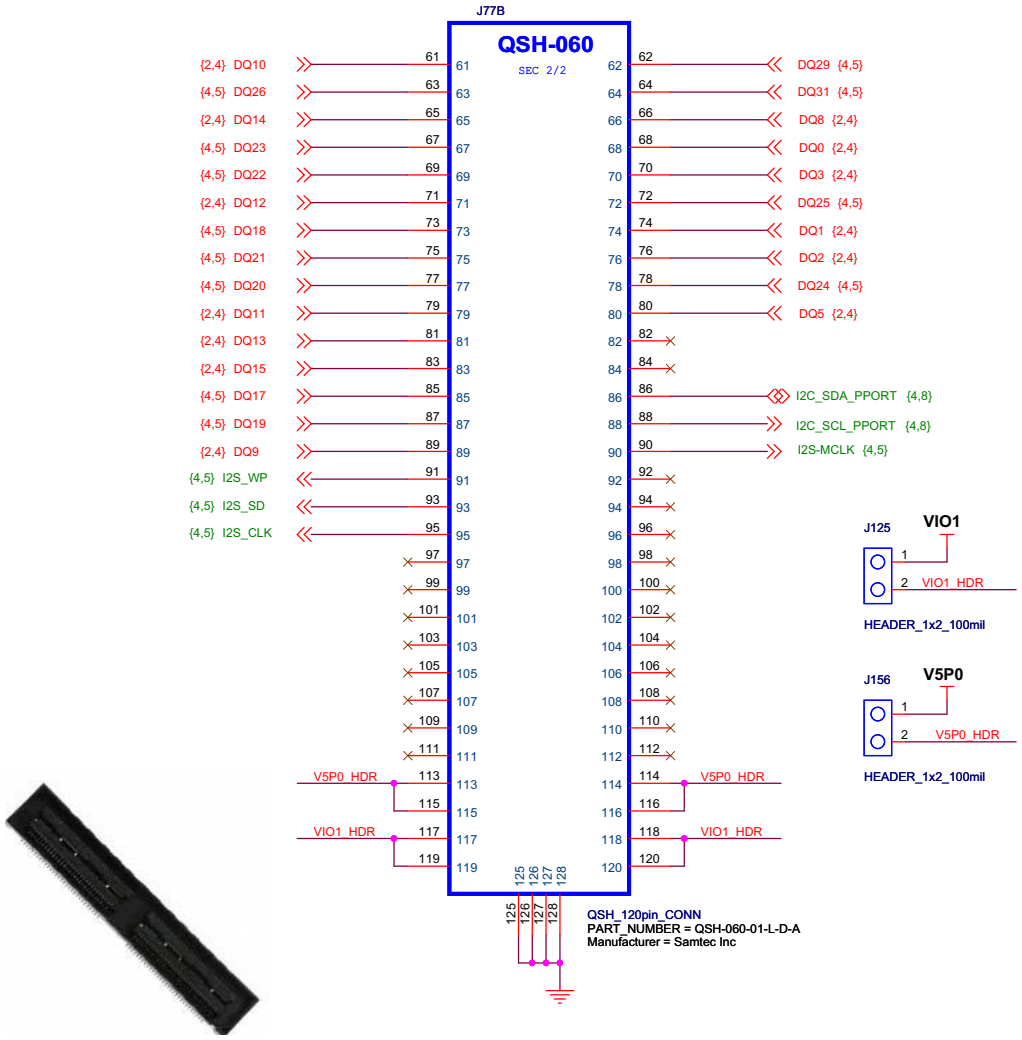


Figure 4-7. Samtec Expansion Connector Circuit 2



Note GPIF II I/O pins are shared between J77A and J77B connectors. In addition I2S and I2C interface I/O lines are also brought out onto J77B connector.

Table 4-4 shows the detailed pinout of the GPIF II interface on the Samtec expansion connector.

Table 4-4. FX3 GPIF II Signal Pinout on Samtec Connector

Connector J77 Pin Nos.	FX3 Signals	GPI/O
68	D0	GPIO[0]
74	D1	GPIO[1]
76	D2	GPIO[2]
70	D3	GPIO[3]
50	D4	GPIO[4]
80	D5	GPIO[5]
32	D6	GPIO[6]
42	D7	GPIO[7]

Table 4-4. FX3 GPIF II Signal Pinout on Samtec Connector

Connector J77 Pin Nos.	FX3 Signals	GPI/O
66	D8	GPIO[8]
89	D9	GPIO[9]
61	D10	GPIO[10]
79	D11	GPIO[11]
71	D12	GPIO[12]
81	D13	GPIO[13]
65	D14	GPIO[14]
83	D15	GPIO[15]
44	CLK	GPIO[16]
33	CTL0	GPIO[17]
51	CTL1	GPIO[18]
55	CTL2	GPIO[19]
31	CTL3	GPIO[20]
60	CTL4	GPIO[21]
39	CTL5	GPIO[22]
45	CTL6	GPIO[23]
37	CTL7	GPIO[24]
49	CTL8	GPIO[25]
41	CTL9	GPIO[26]
43	CTL10	GPIO[27]
54	CTL11	GPIO[28]
57	CTL12	GPIO[29]
36	PMODE0	GPIO[30]
46	PMODE1	GPIO[31]
30	PMODE2	GPIO[32]
52	INT	int#
59	RESET	reset#
56	D16	GPIO[33]
85	D17	GPIO[34]
73	D18	GPIO[35]
87	D19	GPIO[36]
77	D20	GPIO[37]
75	D21	GPIO[38]
69	D22	GPIO[39]
67	D23	GPIO[40]
78	D24	GPIO[41]
72	D25	GPIO[42]
63	D26	GPIO[43]
58	D27	GPIO[44]
38	D28_UART-RTS	GPIO[46]

Table 4-4. FX3 GPIF II Signal Pinout on Samtec Connector

Connector J77 Pin Nos.	FX3 Signals	GPI/O
62	D29_UART-CTS	GPIO[47]
40	D30_UART-TX	GPIO[48]
64	D31_UART-RX	GPIO[49]
95	I2S-CLK	GPIO[50]
93	I2S-SD	GPIO[51]
91	I2S-WS	GPIO[52]
90	I2S-MCLK	GPIO[57]
88	I2C-SCL	GPIO[58]
86	I2C-SDA	GPIO[59]

As shown in Figure 4-7, J125 if populated can be used to enable the VIO1 (1.8 V, 2.5 V, and 3.3 V – see Table 4-1) power going out on the Samtec connector (GPIF II interface). Similarly, J156 can also be used to enable the 5-V power going out on the Samtec connector.

The CTRL_4 line can be routed either to J77A Samtec connector (as GPIF II interface I/O line) or can be used to enable/disable the on-board USB switch. As explained in USB Receptacle on page 44, the J100 jumper can be used to control the OTG power. Figure 4-8 shows how the selection for pin2-CTL4_SW is made.

Figure 4-8. CTRL_4 Selection Jumper

As shown in Figure 4-8, if 1 and 2 are connected on J100 through a jumper, the CTRL_4 line goes to the GPIF II interface on the Samtec connector (J77A). If 2 and 3 are connected, the CTRL_4 line routes to the USB switch(U8) for OTG power control(VBUS.

4.4.1 Interconnecting to the GPIF II Connector

The external processor board can be connected to the FX3 device using an interconnection board that hooks up to the Samtec expansion connector J77. The mating connector to the J77 connector is a QTH-060 series Samtec connector. The following figures show the clearances and location of the Samtec connector on the FX3 DVK board. Note that these dimensions apply only to Rev 02 and Rev 03 releases of the DVK board.

Figure 4-9. Clearance to Samtec Connector (all dimensions in mils)

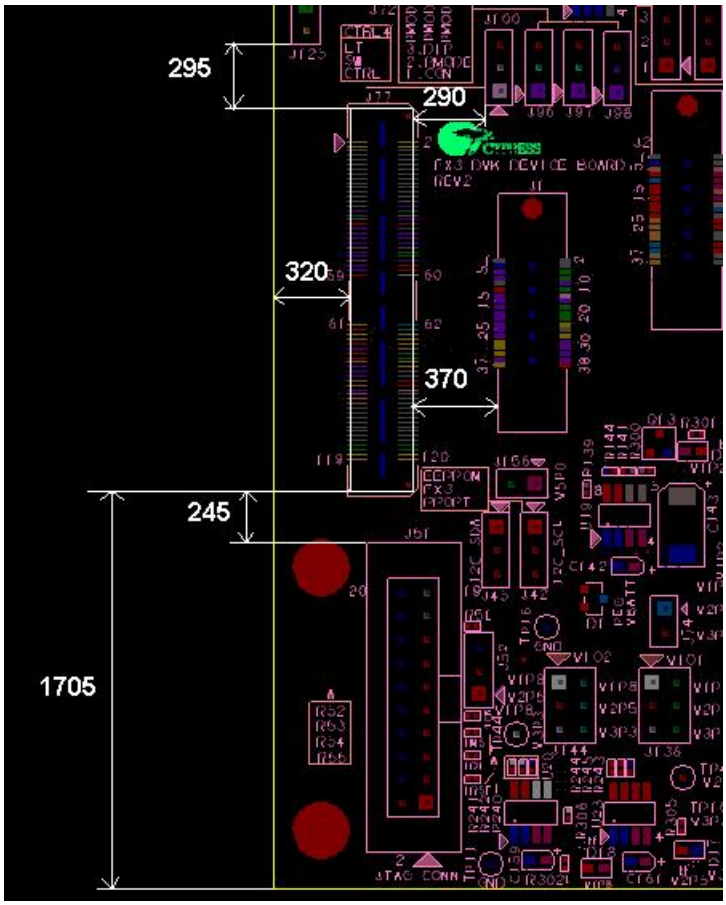


Figure 4-9 shows the clearances to the Samtec connector from the near by components. The white colored outline on the J77 connector is the outline of the actual physical connector.

Figure 4-10. Distance from Center of Pin 1 to Top Edge of Board (all dimensions in mils)

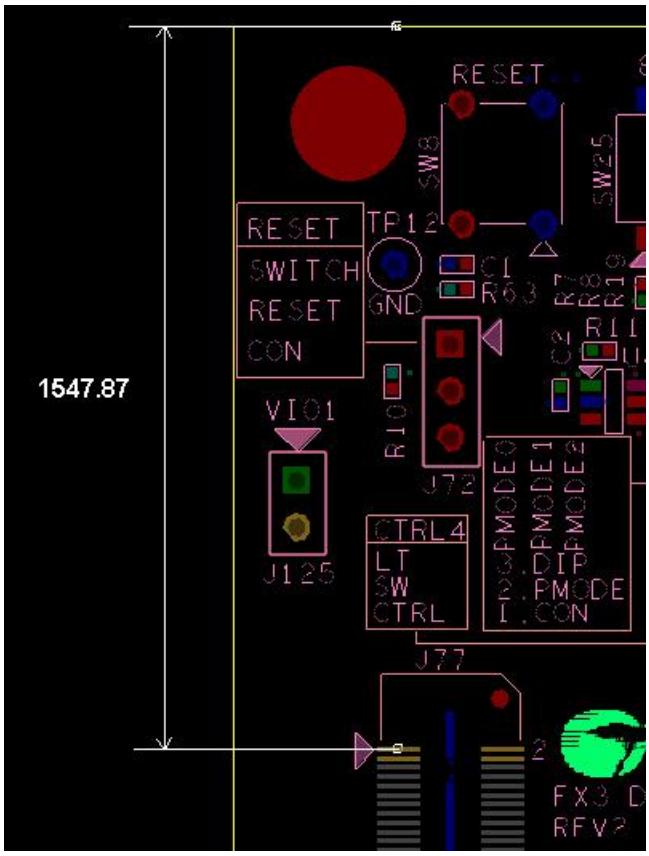
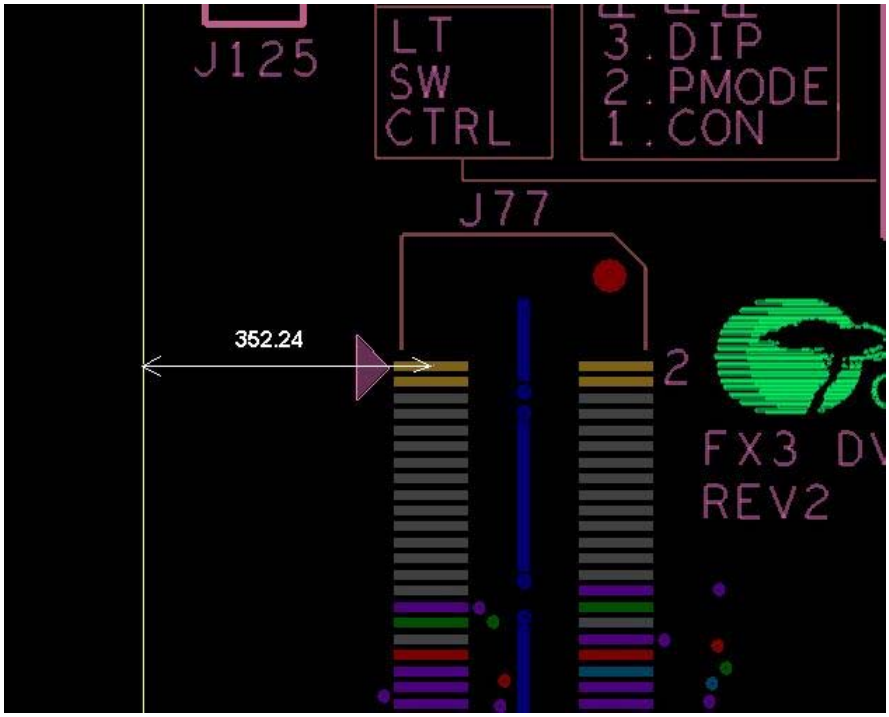


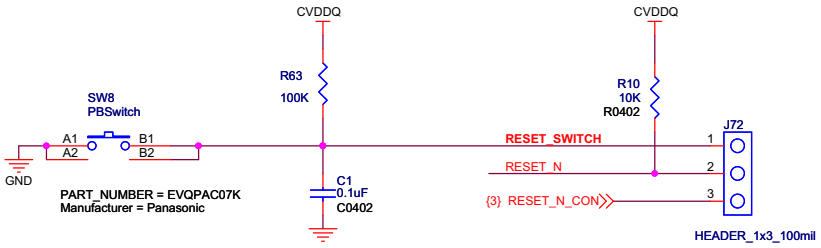
Figure 4-11. Distance from Center of Pin 1 to Left Edge of Board (all dimensions in mils)



4.5 Reset Circuit

The FX3 device can either be reset from the external processor hooked up to the Samtec connector or from an on-board push button. This selection can be made on J72; see Figure 4-12.

Figure 4-12. Reset Circuit and Selection Headers



Based on the selection made on J72, either this RESET_N signal goes to the FX3 device (pins 1 and 2 connected on J72) or a signal from the external processor resets the FX3 device (pins 2 and 3 connected on J72).

4.6 Serial Interfaces

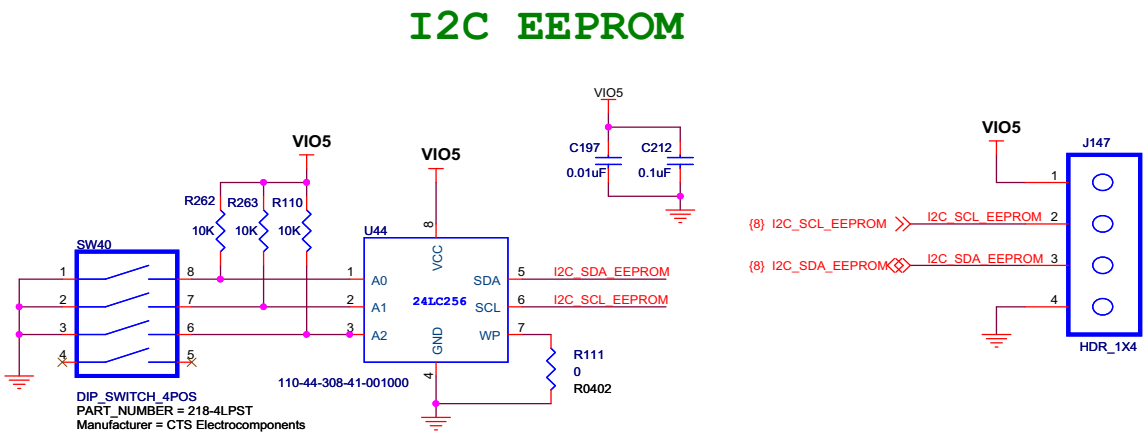
EZ-USB FX3 supports the following serial interfaces:

- I2C
- I2S
- SPI and UART
- JTAG

4.6.1 I2C Interface

The I2C interface lines on the FX3 device come out to headers for probing and expansion, and also connect to an on-board EEPROM device. The EEPROM address bits A2, A1, and A0 can be set using the on-board dip switch SW40.

Figure 4-13. EEPROM and Address Selection Switches



As shown in the above dip switch Figure 4-13, if the A[2:0] pins are set towards the 'ON' (closed position), the corresponding address line is connected to ground. If the switch position is set to OFF (open position), the corresponding address line is pulled high to VIO5 voltage level. See section "I2C EEPROM Boot" of the application note AN76405 - EZ-USB FX3 Boot Options for details.

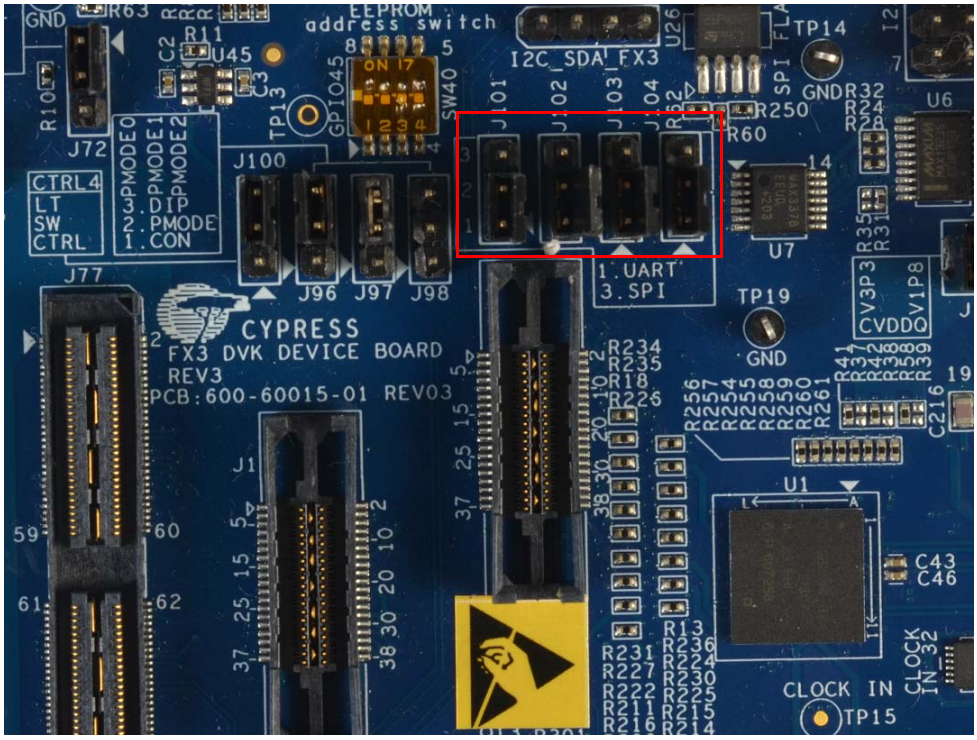
4.6.2 I2S

EZ-USB FX3 has an I2S port to support external audio codec devices. FX3 functions as an I2S master (transmitter only). The I2S interface consists of four signals: clock line (I2S_CLK), serial data line (I2S_SD), word select line (I2S_WS), and master system clock (I2S_MCLK). FX3 can generate the system clock as an output on the I2S_MCLK line or accept an external system clock input on the same line. All four I2S lines come out on header J20.

4.6.3 SPI and UART

EZ-USB FX3 supports an SPI master interface on the serial peripheral port. The SPI I/O lines are shared with the UART I/O lines on the FX3 device. The FX3 DVK jumper settings do not allow operation of both SPI and UART interfaces simultaneously.

Figure 4-14. SPI/UART Selection Headers on DVK



Note The arrow marked near the jumpers indicate pin1 of the jumper.

Figure 4-15 shows the jumper J101-J104 schematic symbols and the different pin names (UART/SPI) on each of these jumpers.

Figure 4-15. UART/SPI Selection Headers

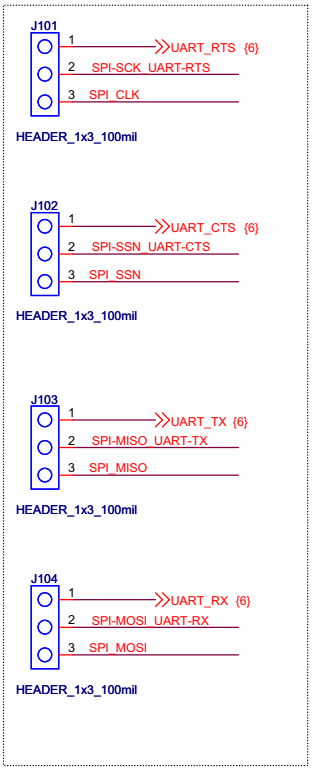


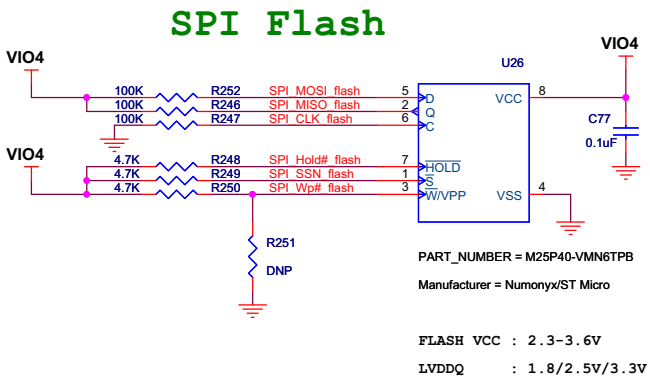
Table 4-5 shows the jumper J101-J104 pin combination to select either UART or SPI Interface.

Table 4-5. Jumper Pin Selection for UART and SPI I/O Lines

Jumper	Jumper Pins Shorted (UART I/O Lines)	Jumper Pins Shorted (SPI I/O Lines)
J101	1-2 (UART_RTS)	2-3(SPI_CLK)
J102	1-2(UART_CTS)	2-3(SPI_SS)
J103	1-2 (UART_TX)	2-3(SPI_MISO)
J104	1-2(UART_RX)	2-3(SPI_MOSI)

The SPI lines also connect to an onboard SPI flash, as shown in Figure 4-16.

Figure 4-16. Flash Circuit of DVK Board



Note Pull-ups/Pull-down resistors are not recommended on SPI_MISO/SPI_MOSI signals. These are not populated on REV-03 FX3 DVK board.

4.6.4 JTAG

FX3's JTAG interface provides a standard five-pin interface for connecting to a JTAG debugger. The JTAG circuit on DVK board provides an option to debug the firmware through the CPU core's on-chip debug circuitry. Industry standard debugging tools for the ARM926E-J-S core can be used for FX3 application development. The JTAG pins of FX3 come out on J51. Details on how to perform JTAG configuration and debugging is available in *FX3 programmers Manual.pdf,* section 12.2.2.3 Executing and Debugging. The document can be downloaded from the SDK webpage.

A. Appendix - Troubleshooting

A.1 Self-Power Mode

Problem	Possible Cause	Possible Solution
FX3 DVK board does not power on when power supply is connected	Incorrect switch setting	Set switch SW9 to V5P0 instead of VBUS_IN
	DVK board or power adapter failure	Insert the power plug firmly into the J49 power jack
		If problem persists, then the power supply unit may be faulty. Test the DVK board with another power supply unit with the same current/voltage ratings
		If the FX3 DVK board does not power up even after replacing the power adapter the board may be faulty. Contact http://www.cypress.com/go/support for technical assistance

A.2 Bus-Power Mode

Problem	Possible Cause	Possible Solution
FX3 DVK board does not power on when USB cable is connected	Incorrect switch setting	Set switch SW9 to V5P0 instead of VBUS_IN
	Incorrect setting of jumper J53	Verify that pins 1-2 of J53 are shorted

A.3 USB 2.0 Enumeration

Problem	Possible Cause	Possible Solution
FX3 DVK board does not enumerate when connected to PC host	Incorrect power mode jumper settings	If the DVK board is self-powered, ensure that the jumper settings are similar as specified in "Self-Power Mode" on page 18
	Incorrect setting of jumper J53	Verify that pins 1-2 of J53 are shorted in Bus-Power mode
	USB 3.0 cable failure	Verify that the USB cable is properly connected. Use the cable supplied with the kit or a USB-IF certified cable
	Windows OS fails to indicate enumeration to user	The FX3 device enumerates correctly but sometimes Windows OS does not display any window to indicate that the device has enumerated. See Figure 3-4 and Figure 3-5 and locate the FX3 device entry in Windows OS Device Manager

A.4 USB Driver Installation

Problem	Possible Cause	Possible Solution
Cannot Install cyusb3.sys driver	Unsigned cyusb3.sys driver	FX3 SDK before version 1.1.1 contains unsigned USB drivers. To use these drivers, reboot the PC host and during boot press **[F8]** and select **Disable Driver Signature enforcement** in the list of PC BIOS options
		Use drivers from SDK version 1.1.1 and later, which contain Microsoft certified *cyusb3.sys* driver along with a secure catalog file (*cyusb3.cat*). The latest SuiteUSB can also be downloaded separately from FX3 SDK webpage.
	Incompatible driver for OS platform	Install correct SuiteUSB installer. For 32-bit Windows XP/Vista/7 install CyUSB3_x86_<Build_no>.msi For 64-bit OS platform install CyUSB3_x64__<Build_no>..msi Perform manual driver re-binding as described in section XX using Windows Hardware Wizard.

A.5 USB Boot

Problem	Possible Cause	Possible Solution
Control Center displays **"Programming Failed"** when FX3 RAM is programmed with firmware	Incorrect PMODE settings	Verify that PMODE [2:0] pin settings for USB boot are as specified in Table 3-5
	Limitation of Etron host controller. (The Control Center application in SDK builds prior to 1.2.1 downloads firmware to FX3 in pieces of 4096 bytes. The Etron host controller further splits this data into separate chunks of 4058 bytes and 38 bytes)	Migrate to FX3 SDK version 1.2.1 or higher. (In FX3 SDK 1.2.1, the control transfer size for firmware download is reduced to 2048 bytes in Control Center application)
	USB 3.0 cable failure	Verify if the USB cable is connected firmly. Use the cable supplied with the kit or a USB-IF certified cable.

A.6 I2C Boot

Problem	Possible Cause	Possible Solution
"**Programming of I2C EEPROM failed"** message appears in Control Center	Incorrect EEPROM mounting	Verify that EEPROM I/O pins are mounted into correct slots of the DIP socket U44
	Incorrect combination of PMODE jumpers and dip switch settings	Before programming the EEPROM, set the board in USB boot mode. After DVK board enumerates using default bootloader, ensure that the address pins are configured correctly using SW40 before programming the EEPROM. Refer "Download Firmware Image to FX3 RAM" on page 25 for details
	Incorrect settings on jumpers J42 and J45	Verify that pins 1-2 are shorted
	Incorrect settings on SW40 for EEPROM address pins (A[2:0])	Ensure that the EEPROM address pins A[2:0] are configured according to the requirement specified for the EEPROM being used (refer to the EEPROM datasheet)
	Faulty EEPROM	Replace EEPROM if programming still fails and redo the steps above
FX3 DVK does not boot from I2C EEPROM after programming	Incorrect PMODE settings	Verify Table 3-6 for correct PMODE settings for I2C boot
	Incorrect boot firmware header format	Refer AN76405 - EZ-USB® FX3 Boot Options and verify that elf2img command line utility generates correct header format, as explained in the application note

A.7 SPI Boot

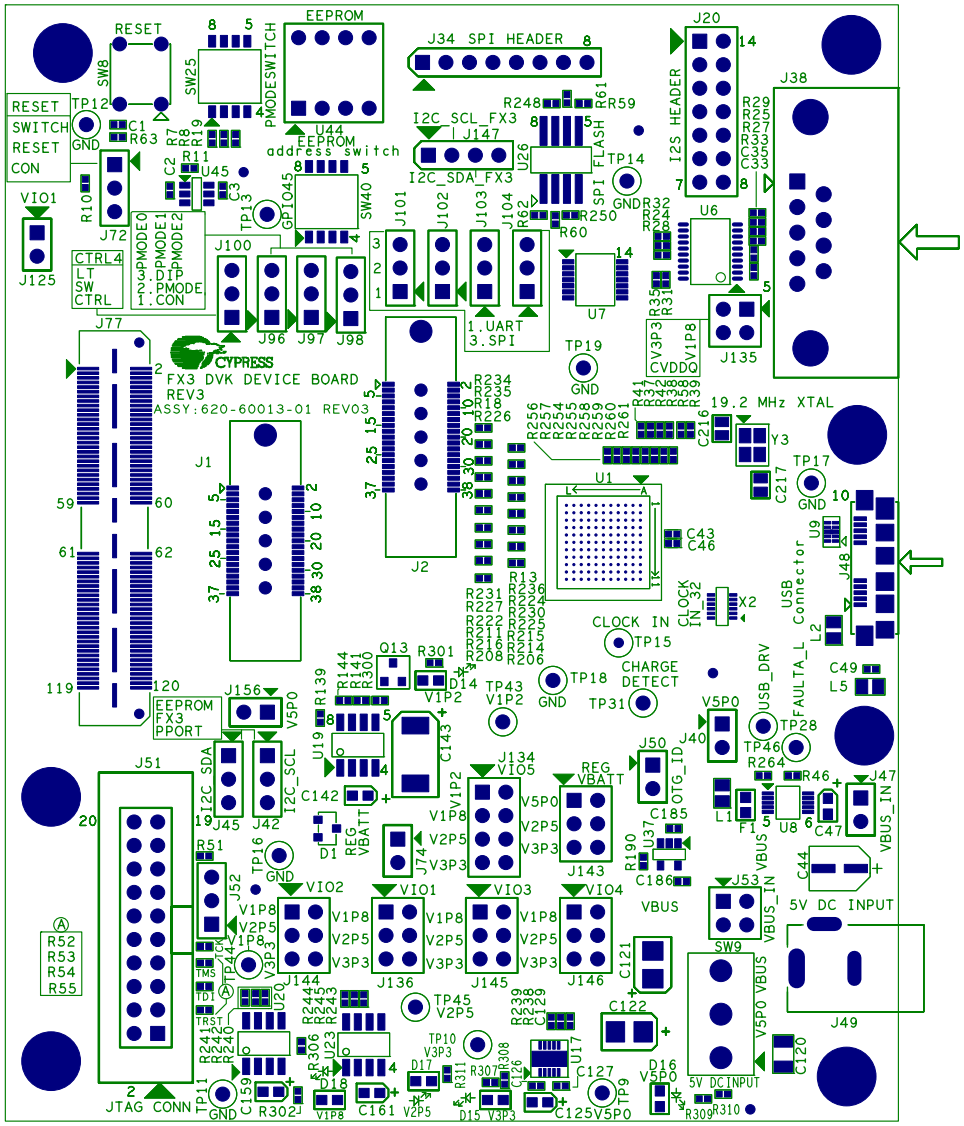
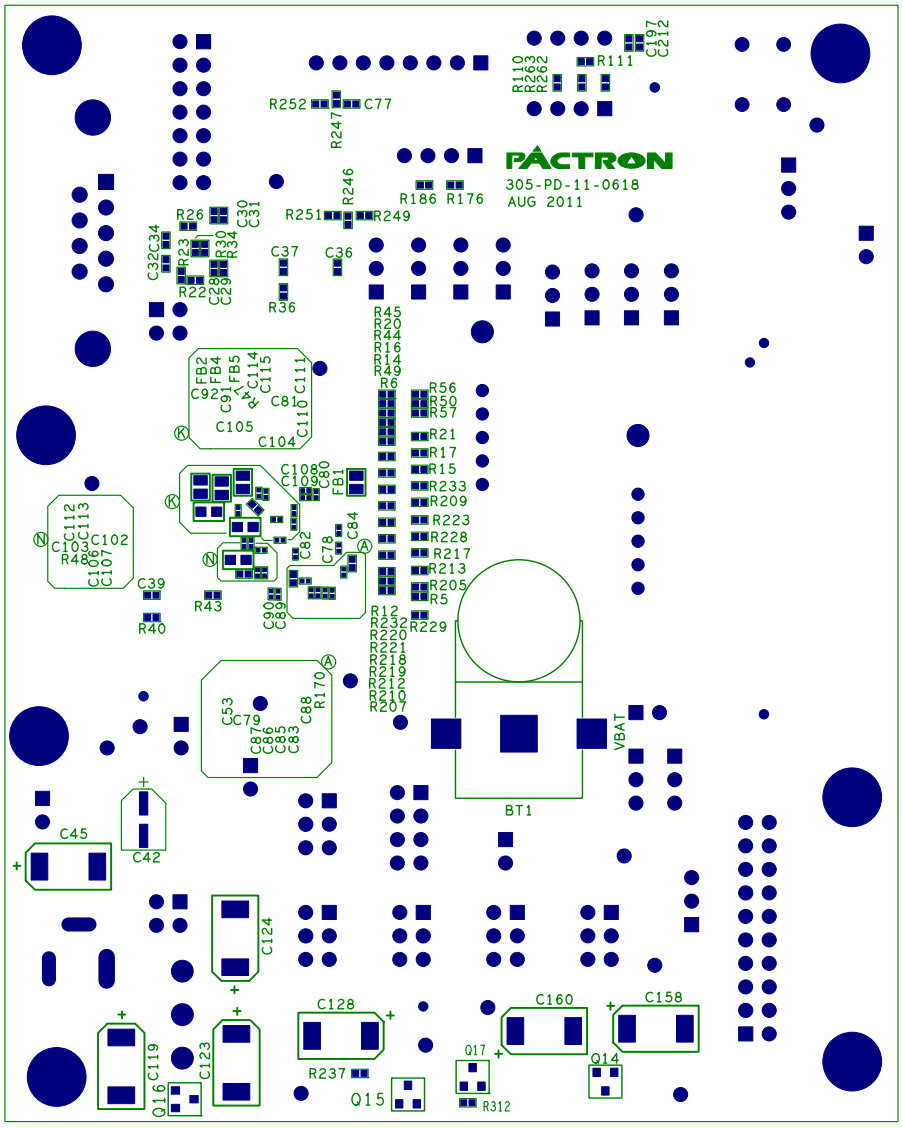
Problem	Possible Cause	Possible Solution
"Programming of SPI Flash failed" message appears in the Control Center	Incorrect PMODE settings	See Table 3-10 to know jumper and dip switch (SW25) settings. The settings vary before and after programming the firmware
	Incorrect Jumper positions	J101, J102, J103, and J104 pins 2-3 should be shorted before programming. This setting should remain for entire SPI boot sequence
FX3 DVK does not boot from SPI Flash after programming	Incorrect PMODE settings	See Table 3-8 to know jumper and dip switch (SW25) settings. The settings vary before and after programming the firmware
	Pull-up resistors on SPI I/O lines	Some FX3 Rev-03 DVK boards contain pull-up resistors R246 and R252 on MISO and MOSI lines of SPI. Remove these pull-up resistors and retry the booting sequence
		FX3 DVK boards prior to REV-03 does not support SPI boot.

A.8 USB3.0 Enumeration

Problem	Possible Cause	Possible Solution
FX3 enumerates as USB 2.0 device even though it boots firmware that enables USB 3.0	Applicable only to NEC host controllers - Excessive spread spectrum reference clock from motherboard chipset. (The NEC host controller does not have its own reference clock for USB 3.0. If the motherboard has an out-of-spec PCIe reference clock (either incorrect frequency or excessive spread spectrum), the device attached to the USB 3.0 controller will not be able to track the signal and will fail to enumerate on the system.)	Check the Spread Spectrum setting in the PC BIOS. If enabled, disable Spread Spectrum manually
	Incorrect firmware configuration	Verify that the FX3 firmware has **CyU3PConnectState** set to CyTrue. This enables USB PHY on FX3 device and connects to the USB host. The CyU3PGetConnectState API returns CyTrue if the FX3 device is connected and VBUS is detected by FX3. Also, the CyU3PUsbGetSpeed API provides information on the connection speed. See the FX3 API Guide for details

B. Appendix - PCB Layout

B.1 Assembly Top



B.2 Assembly Bottom