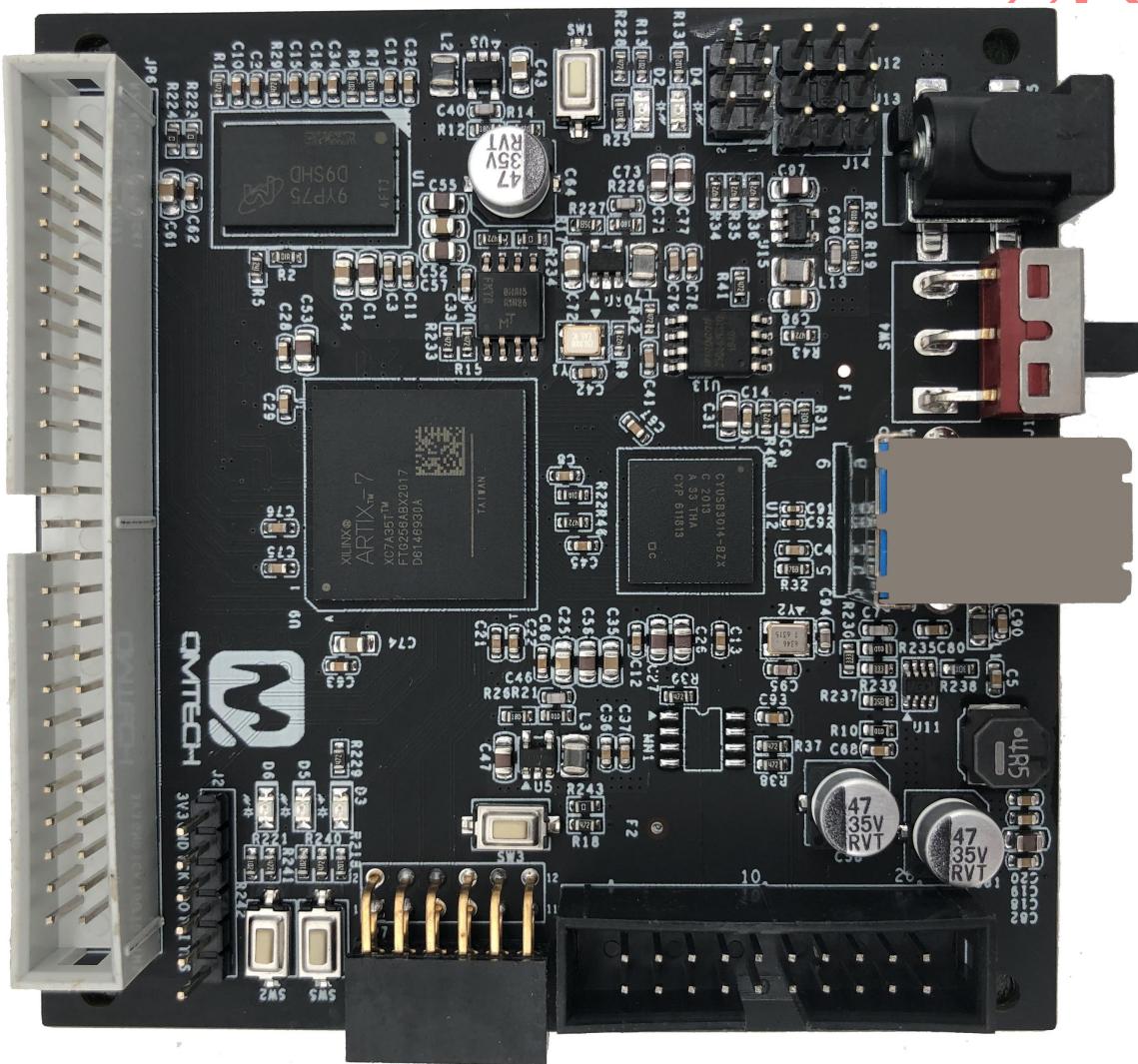


ARTIX-7 USB3.0 DEVELOPMENT BOARD

USER MANUAL(SOFTWARE)



Preface

The QMTECH® Artix-7 USB 3.0 development board uses Cypress's EZ-USB® FX3™ high-bandwidth USB 3.0 peripheral controller and Xilinx Artix®-7 device to demonstrate USB 3.0 SuperSpeed technology's high usable bandwidth and great power delivery. This user manual introduces the software design of USB3.0 communication with FPGA by using SlaveFifo, StreamIN, etc.

Table of Contents

1. CYUSB3014 SDK INSTALLATION.....	3
1.1 INTRODUCTION.....	3
1.2 SDK CONTENT.....	3
1.3 SDK INSTALLATION.....	3
2. USB3.0 DRIVER INSTALLATION.....	6
3. USB 3.0 LOOPBACK TEST.....	8
3.1 HARDWARE AND SOFTWARE PREPARATION.....	8
3.2 COMPILE LOOPBACK FIRMWARE.....	9
3.3 LOOPBACK TEST PROCEDURE.....	13
4. SPI FLASH BOOT UP TEST.....	18
4.1 HARDWARE AND SOFTWARE PREPARATION.....	18
4.2 SPI FLASH BOOT UP TEST.....	19
5. USB 3.0 PERFORMANCE TEST.....	21
5.1 HARDWARE AND SOFTWARE PREPARATION.....	21
5.2 COMPILE BULKSEND FIRMWARE.....	21
5.3 BULKSEND TEST.....	22
6. USB 3.0 GPIF TESTS WITH ARTIX-7 FPGA.....	25
6.1 SLAVE FIFO TEST PREPARATION.....	25
6.2 SLAVE FIFO TEST PROCEDURE.....	26
6.3 STREAMIN TEST PREPARATION.....	32
6.4 STREAMIN TEST PROCEDURE.....	33
7. REFERENCE.....	35
8. VERSION.....	36

1. CYUSB3014 SDK Installation

1.1 Introduction

CYUSB3014 SDK is provided and maintained by Cypress, named as FX3 SDK. Below image shows the architecture of the FX3 SDK, it contains whole set of software develop environment, e.g. Eclipse based EZ USB Suite IDE, GNU compiler/assembler/linker, FX3 API and Drivers.

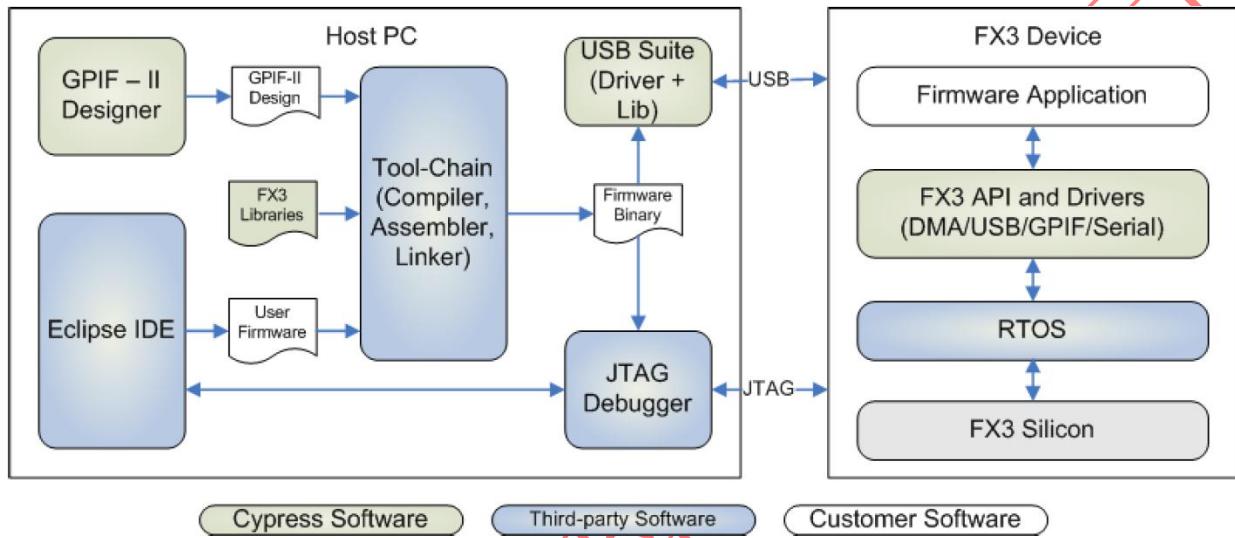


Figure 1-1. FX3 SDK Architecture

1.2 SDK Content

Cypress FX3 SDK's latest version is v1.3.3 which could be downloaded from Cypress Officiate site directly. The name of the FX3 SDK installer is FX3_SDK_Windows_v1.3.3.exe. Below lists the contents of the FX3 SDK:

- EZ-USB FX3 SDK Installer
- Firmware Library Zip
- USB Suite Zip
- FX3 SDK for Linux platforms
- FX3 SDK for MacOS platforms
- User Manuals
- USB Suite Source Code Zip
- Others

1.3 SDK Installation

Below section describes the procedure of the FX3 SDK installation. This example installation procedure is performed with computer DELL Precision Tower 3620: Win10 64bit Home Edition, 8GB DDR3 SDRAM, Intel i5-6500 CPU@3.2GHz, 3.19GHz. This procedure is just for customer reference, users might meet different situations due to e.g. different Windows system versions: 32Bit and 64Bit.



1. After FX3 SDK completely downloaded, double click the installer: FX3_SDK_Windows_v1.3.3.exe. Change the installation directory if needed, then click the NEXT button:

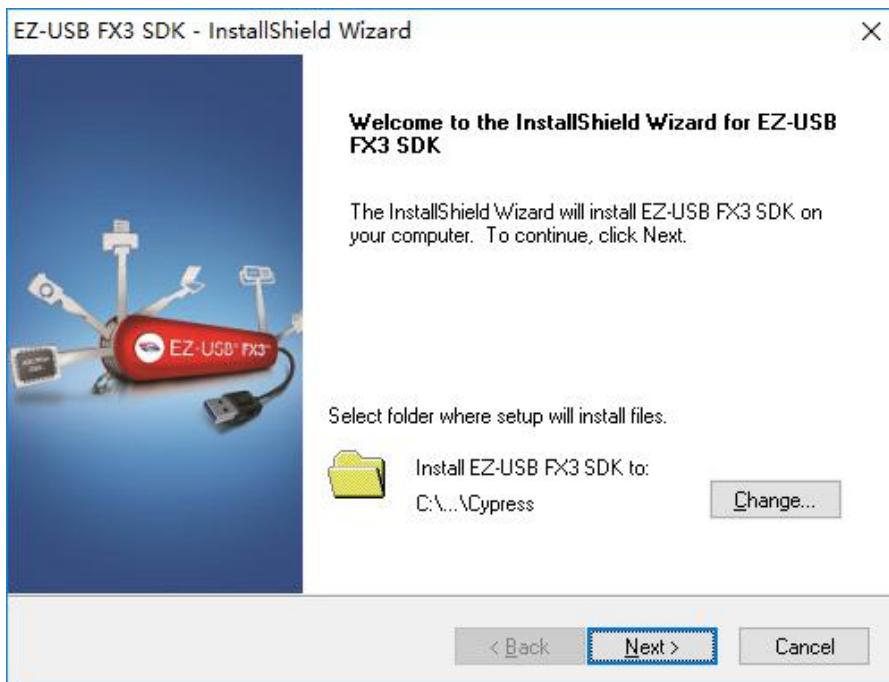


Figure 1-2. Installation Directory

2. Choose the installation type: Typical, and then click NEXT button:

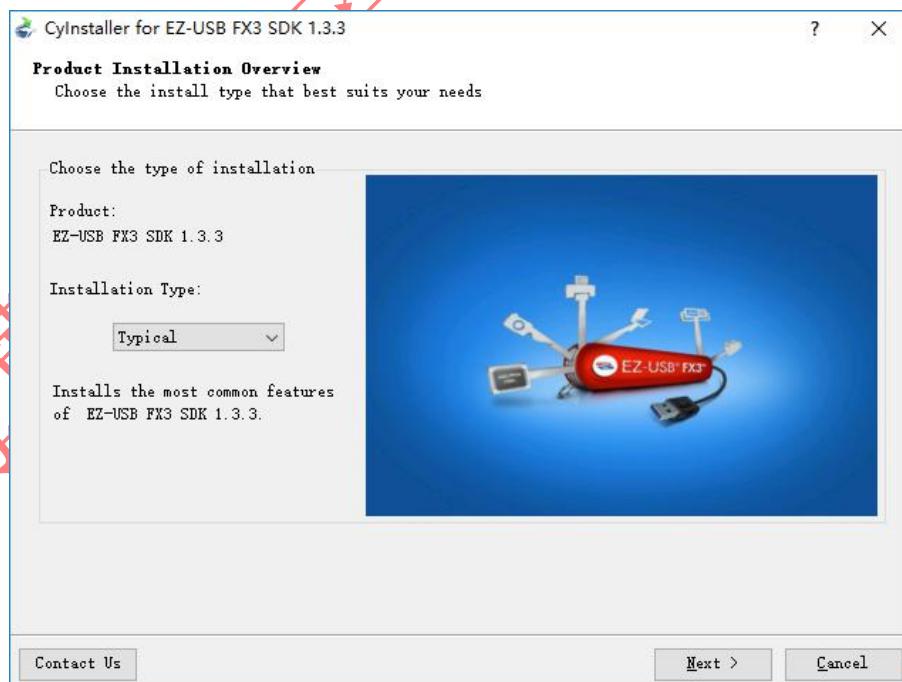


Figure 1-3. Installation Type

3. Accept License Agreement, and click Next button:

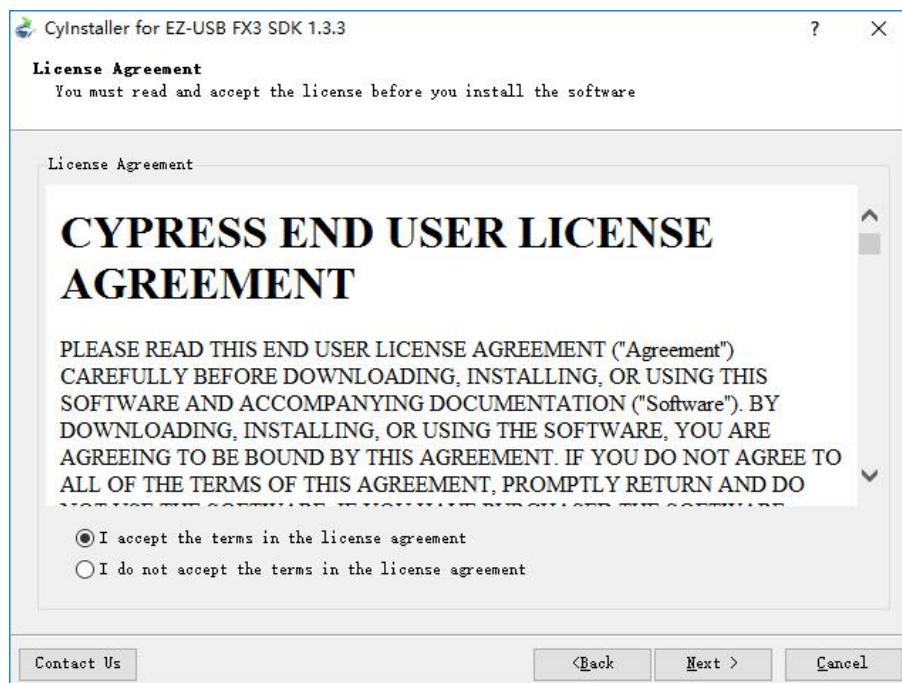


Figure 1-4. License Agreement

4. Below image shows the components will be installed:

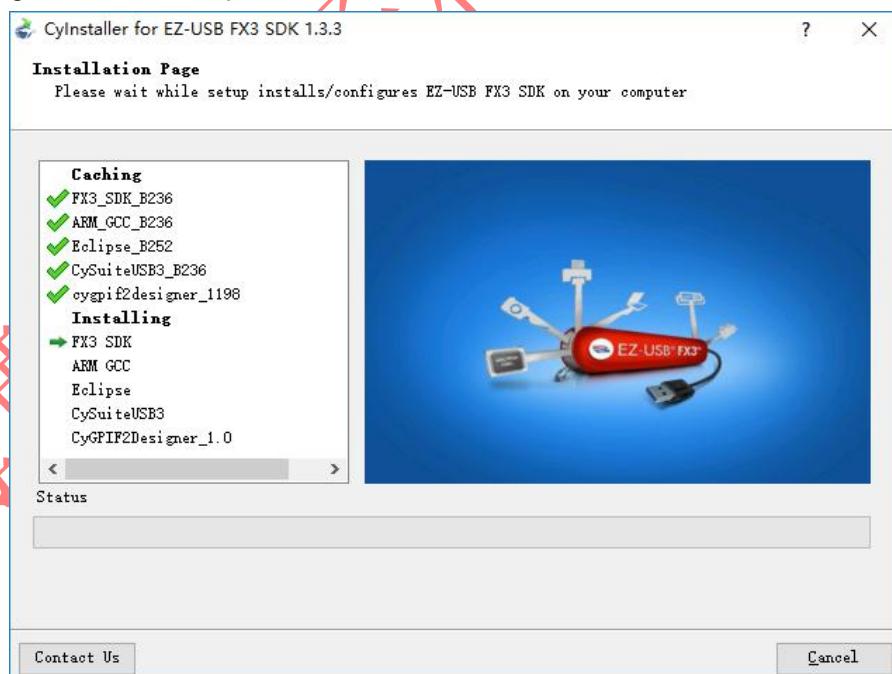


Figure 1-5. Installation Procedure

2. USB3.0 Driver Installation

Firstly, plug the USB 3.0 Type-B to Type-A cable into the USB 3.0 development board's Type-B female header. The on board Type-B female header and Type-B male header are highlighted with red rectangle in below images:

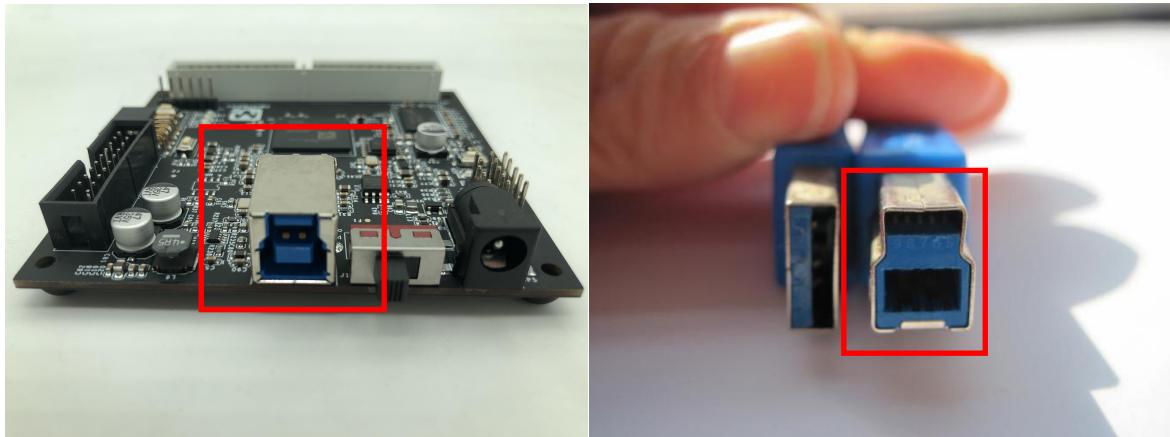


Figure 2-1. USB 3.0 Type-B Connector

Then, plug the USB 3.0 cable's Type-A header into the computer's USB connector with SS descriptor. Here, the SS descriptor represents Super Speed, which means this USB port supports USB 3.0 features.

After the computer detects USB 3.0 peripheral is connected, the driver info will be displayed in Windows Device Manager: WestBridge.

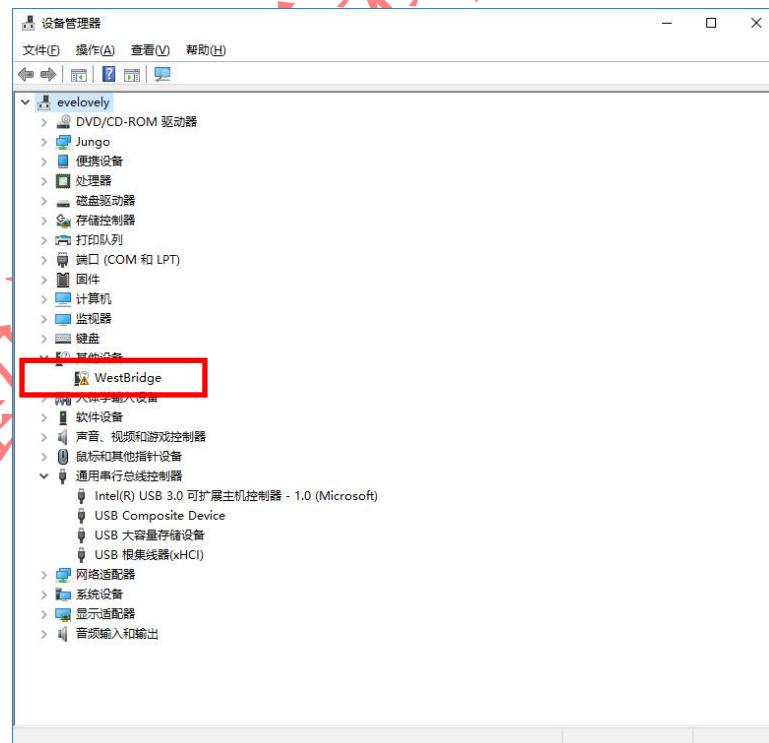


Figure 2-2. USB 3.0 Driver Info

Users need to manually install the USB 3.0 driver provided by Cypress. The USB 3.0 driver is located in FX3 SDK's folder shown as below image:

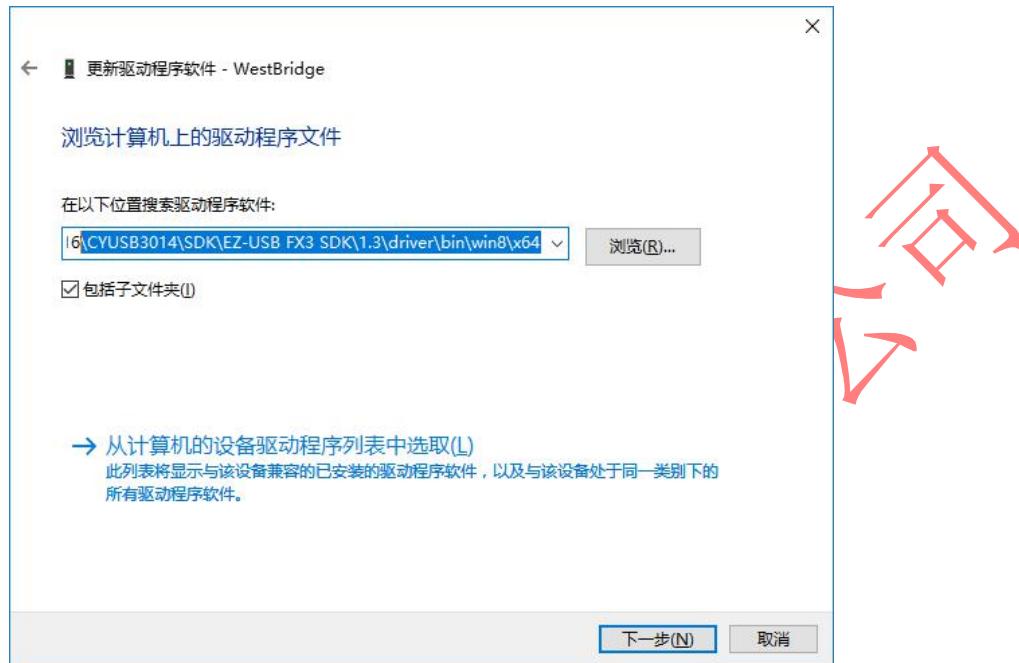


Figure 2-3. USB 3.0 Driver Installation

After the USB 3.0 device driver is correctly installed, the USB 3.0 development board will be enumerated as below device: Cypress FX3 USB Bootloader Device:

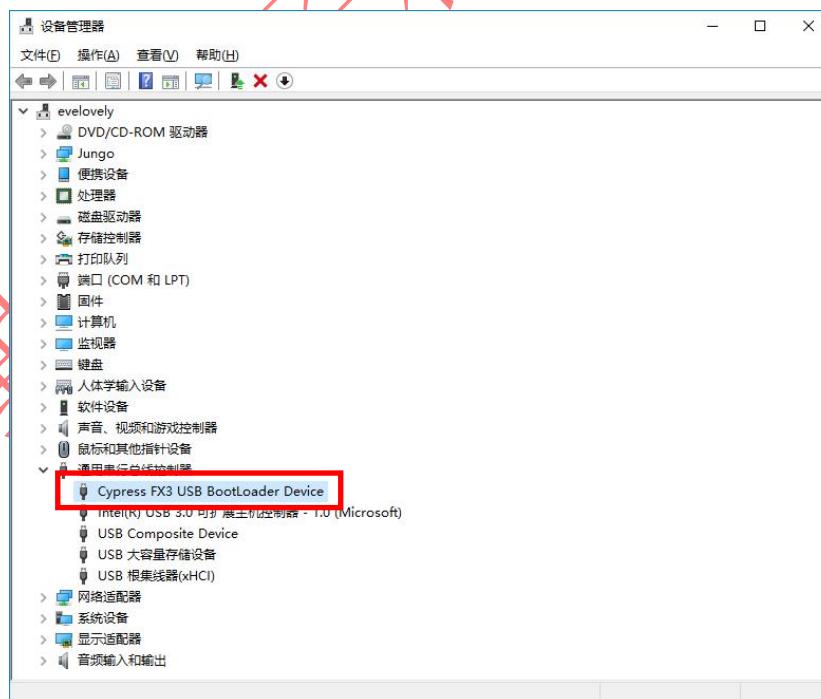


Figure 2-4. Cypress FX3 USB Bootloader Device

3. USB 3.0 Loopback Test

3.1 Hardware and Software Preparation

Users need to configure on board jumpers J12/J13/J14 in status shown as below image. With this boot mode setting, the USB 3.0 device will be configured as USB Boot Mode after the USB 3.0 board powered on.

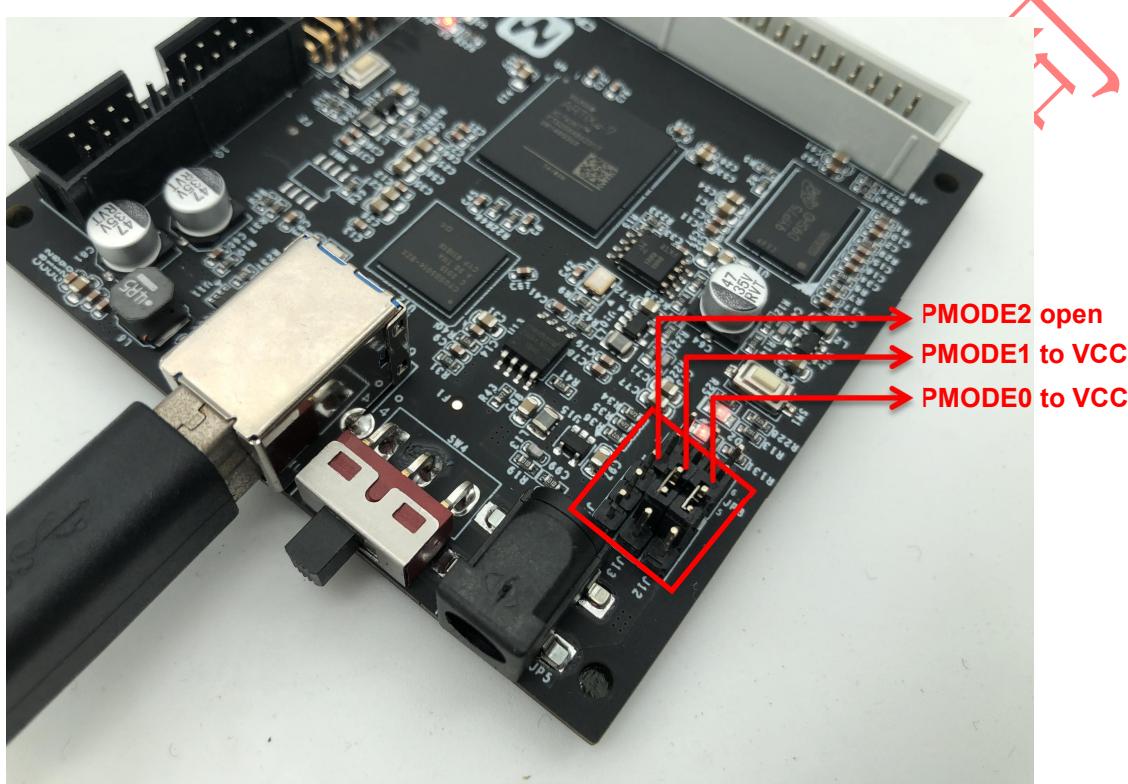


Figure 3-1. Boot Jumpers Setting for USB Boot Mode

In later sections, the User Manual will also introduce other boot modes such as SPI Flash Boot. For other Boot Modes and related PMODE Pin settings, users may refer to below Boot Option table:

PMODE[2:0] Pins			Boot Option
PMODE[2]	PMODE[1]	PMODE[0]	
Z	0	0	Sync ADMUX (16-bit)
Z	0	1	Async ADMUX (16-bit)
Z	0	1	Async SRAM (16-bit)
Z	1	1	USB Boot
1	Z	Z	I2C
Z	1	Z	I2C; on failure, USB Boot is enabled
0	Z	1	SPI; on failure, USB Boot is enabled
FX3S Specific Boot Options			
Z	1	0	PMMC Legacy
0	0	0	S0-port (eMMC); On Failure, USB Boot is enabled
1	0	0	S0-port (eMMC)

Z = Pin is floating; left unconnected.

Figure 3-2. Boot Option

In USB boot mode, the USB device will be enumerated as below device before loading firmware into the CYUSB3014 internal RAM:

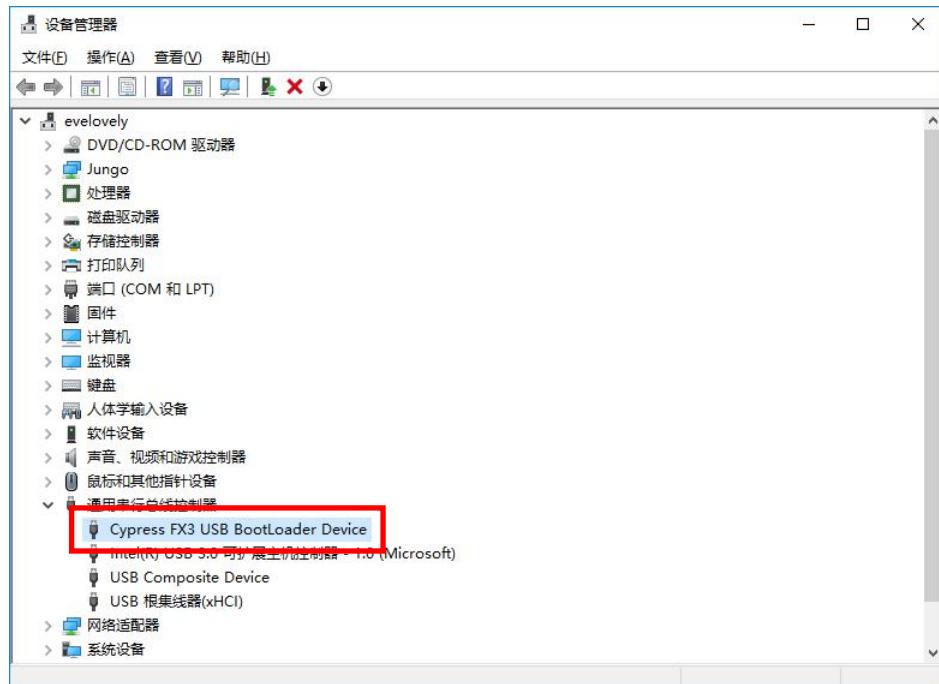


Figure 3-3. Bootloader Device

3.2 Compile Loopback Firmware

The loopback firmware is developed under EZ USB Suite, click [Start] → [Cypress] → [EZ USB Suite]:

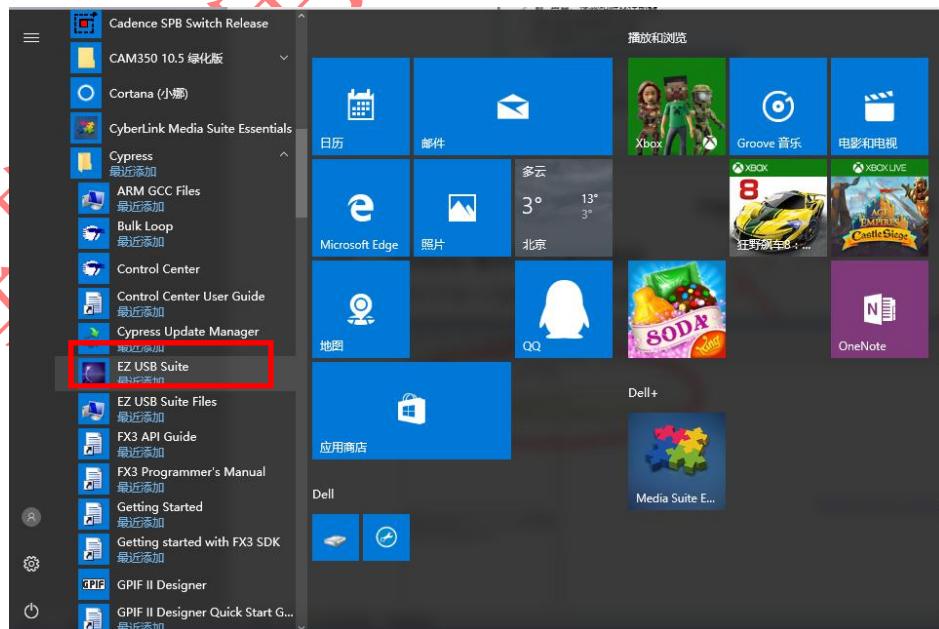


Figure 3-4. EZ USB Suite

After EZ USB Suite lunched, set the default working directory and click OK button:

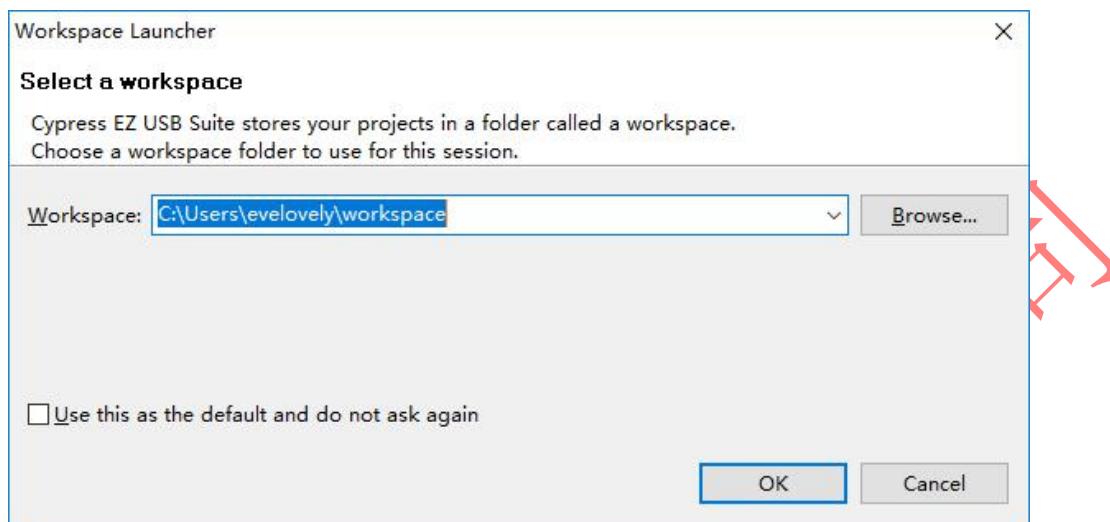


Figure 3-5. Set Default Working Directory

Click EZ USB Suite toolbar [File] → [Import], below Import menu will be displayed. Choose [General] → [Existing Projects into Workspace] and then click Next button.

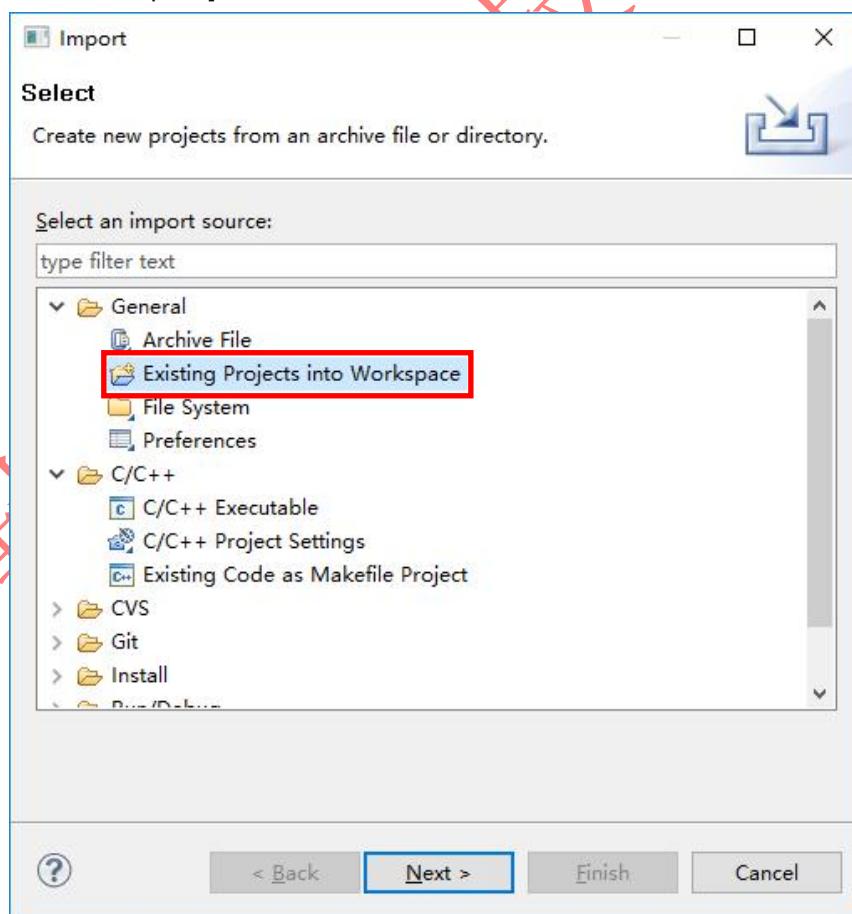


Figure 3-6. Import Existing Projects

Click [Browse] button and select the Loopback firmware from SDK. Below image shows the example directory: \EZ-USB FX3 SDK\1.3\firmware\basic_examples\cyfxbulklpautoenum.

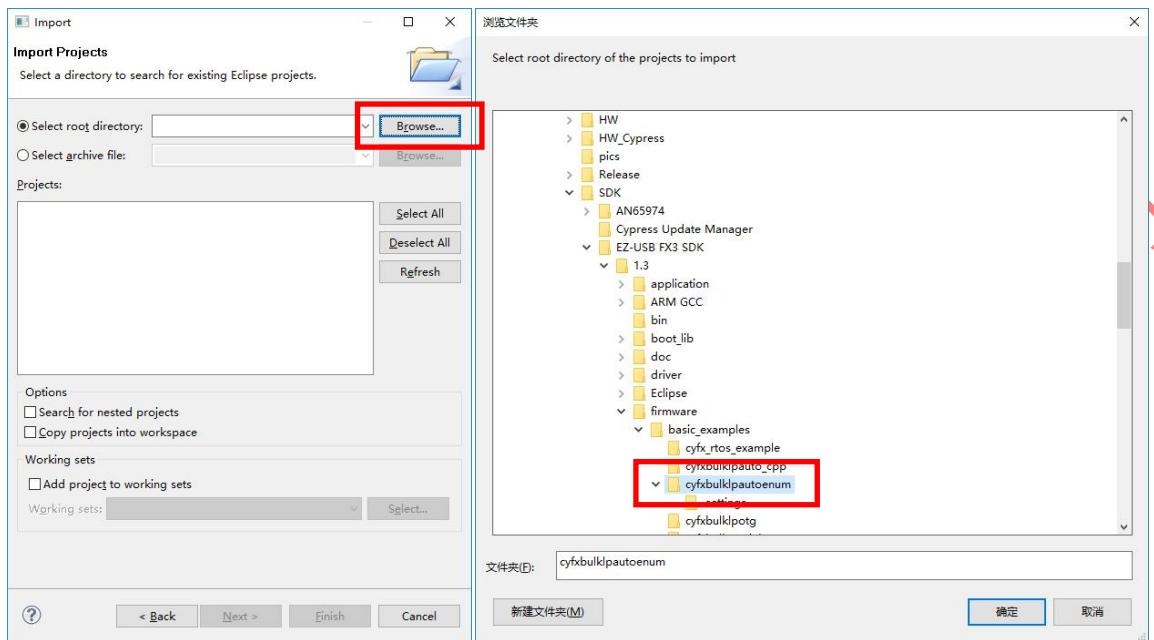


Figure 3-7. Import cyfxbulklpautoenum Project

After the project correctly imported, the project info will be displayed as shown in below image. Then click [Finish] button.

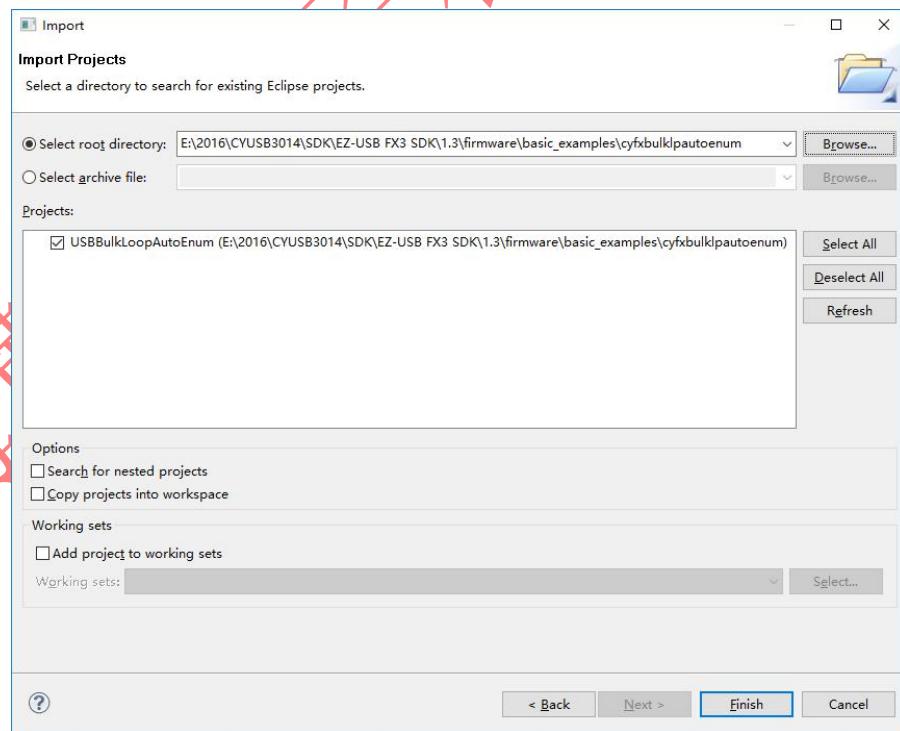


Figure 3-8. Import cyfxbulklpautoenum Project

Below image shows the Cyfxbulklpautoenum Project launched by EZ USB Suite:

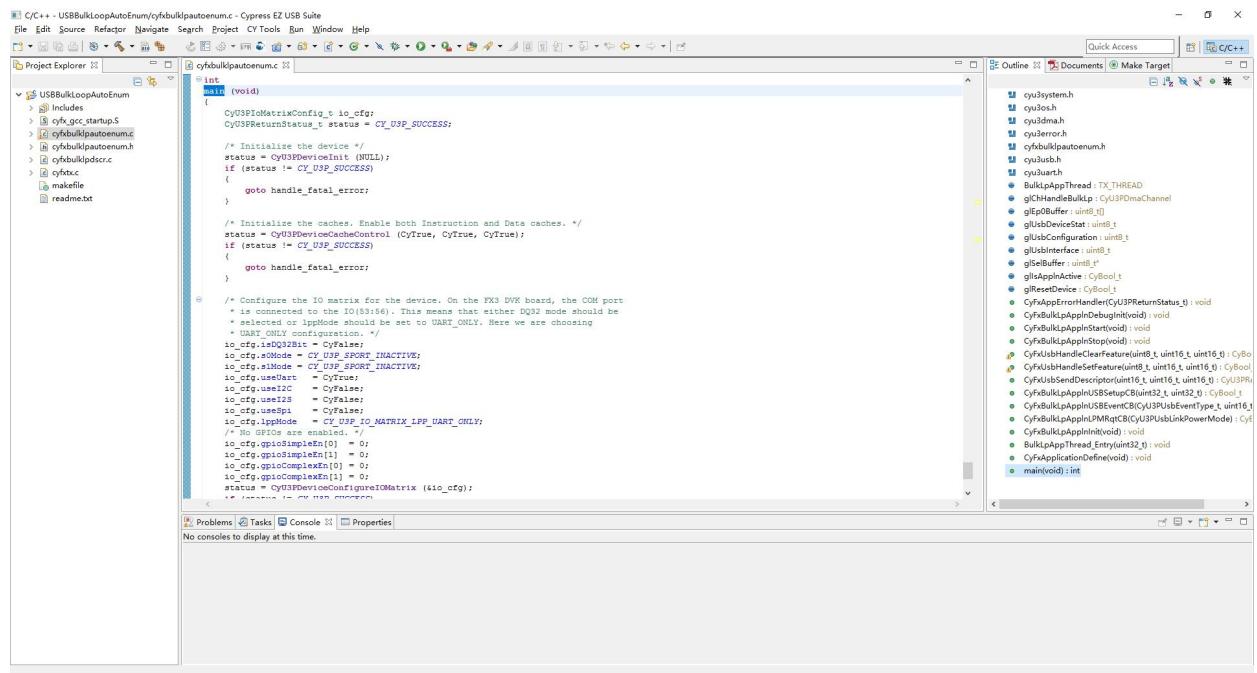


Figure 3-9. Cyfxbulklpautoenum Project

Right click the project and select Build Project option to compile the whole project:

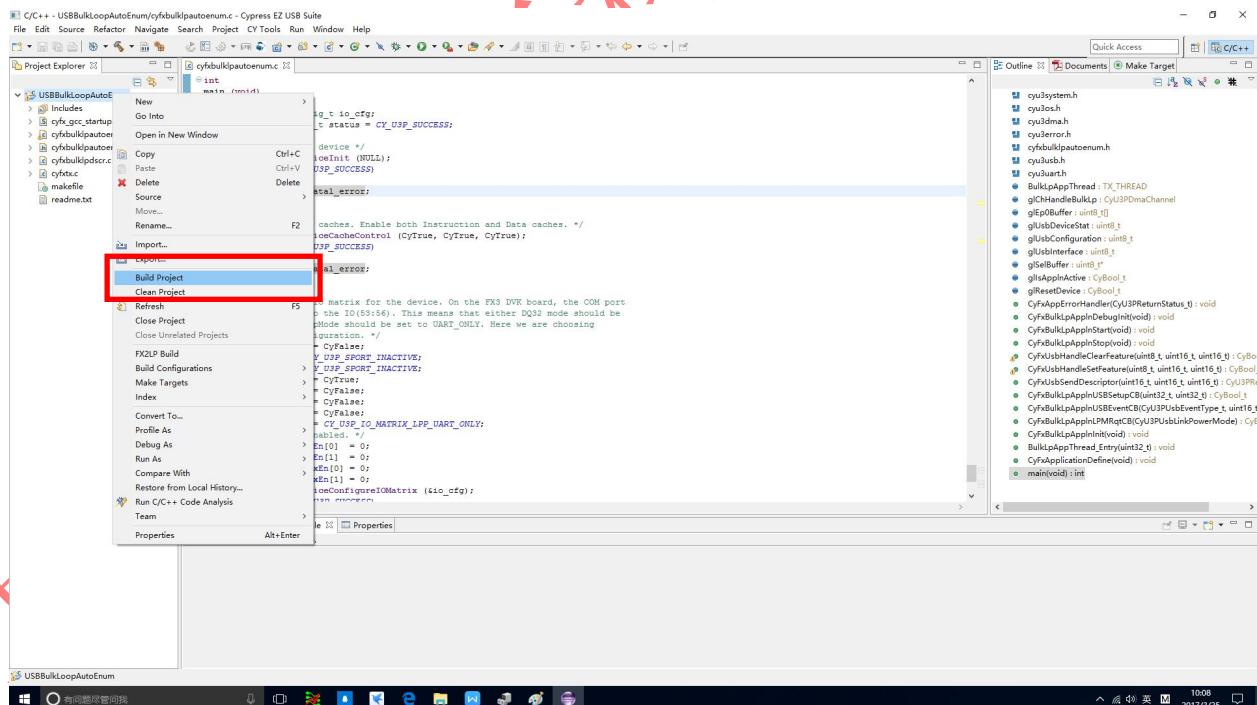
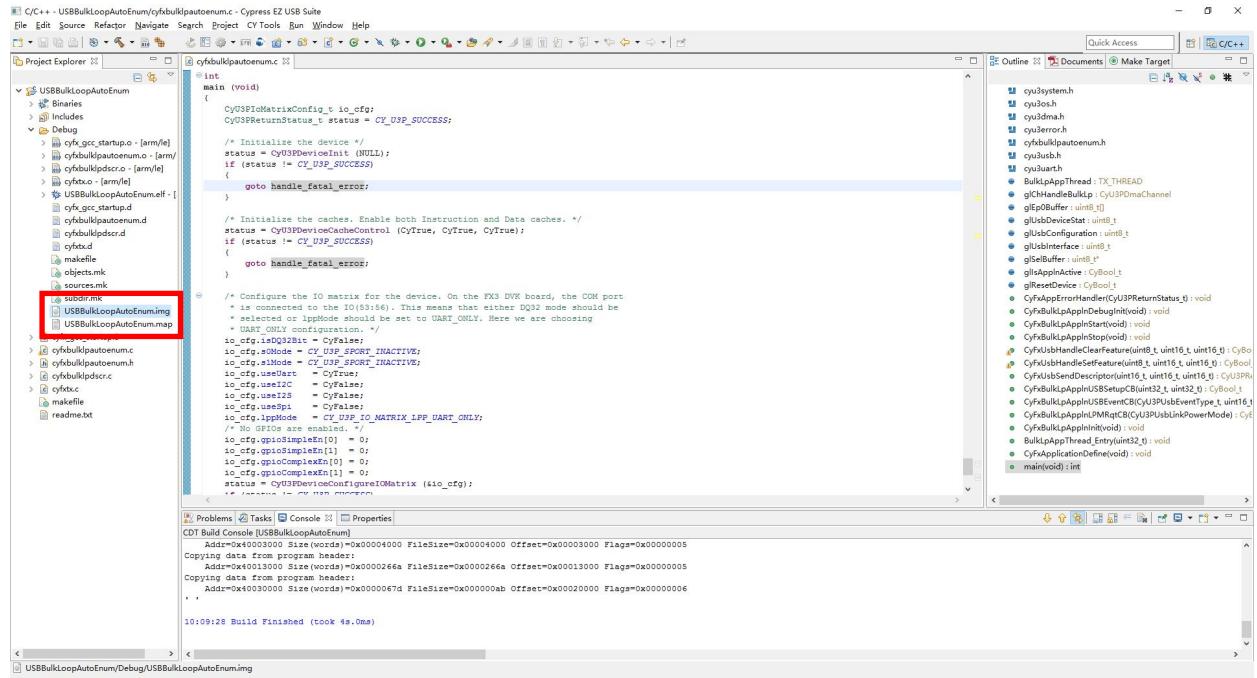


Figure 3-10. Compile cyfxbulklpautoenum Project

After the cyfbulkpautoenum project correctly compiled, it will generate an executable binary image named as USBBulkLoopAutoEnum.img:



```
C/C++ - USBBulkLoopAutoEnum/cyfbulkpautoenum.c - Cypress EZ USB Suite
File Edit Source Refactor Navigate Search Project CY Tools Run Window Help
Project Explorer C/C++ Outline Quick Access Documents Make Target
USBBulkLoopAutoEnum Binaries Includes Debug USBBulkLoopAutoEnum.map
cyfbulkpautoenum.o [arm/e] cyfbulkpautoenum.a [arm] cyfbulkpautoenum.o [arm/e] cyfbulkpautoenum.map
cyfbulkpautoenum.d cyfbulkpdsr.d cyfbulkpdsr.c cyfbulkpautoenum.e [arm/e] cyfbulkpautoenum.map
makefile objects.mk sources.mk
subdir.mk USBBulkLoopAutoEnum.map
USBBulkLoopAutoEnum.map
main (void)
{
    CyGPIOMatrixConfig *io_cfg;
    CyGPIEReturnStatus_t status = CY_USP_SUCCESS;

    /* Initialize the device */
    status = CyU3PDeviceInit((NULL));
    if (status != CY_U3P_SUCCESS)
    {
        goto handle_fatal_error;
    }

    /* Initialize the caches. Enable both Instruction and Data caches. */
    status = CyU3PDeviceCacheControl (CyTrue, CyTrue, CyTrue);
    if (status != CY_U3P_SUCCESS)
    {
        goto handle_fatal_error;
    }

    /* Configure the IO matrix for the device. On the FX3 DVK board, the COM port
     * is connected to the IO[53:56]. This means that either DQ32 mode should be
     * selected or the IO matrix should be set to UART_ONLY. Here we are choosing
     * UART ONLY configuration. */
    io_cfg.ioQ32B1c = CyFalse;
    io_cfg.xMode = CY_USP_SPORT_INACTIVE;
    io_cfg.yMode = CY_USP_SPORT_INACTIVE;
    io_cfg.uselane = CyTrue;
    io_cfg.usel2C = CyFalse;
    io_cfg.usel2S = CyFalse;
    io_cfg.usespI = CyFalse;
    io_cfg.usespO = CyFalse;
    /* No GPIOs are enabled. */
    io_cfg.gpioSimpleIn[0] = 0;
    io_cfg.gpioSimpleIn[1] = 0;
    io_cfg.gpioSimpleIn[2] = 0;
    io_cfg.gpioSimpleIn[3] = 0;
    status = CyU3PDeviceConfigureIOMatrix (&io_cfg);
    /* ... */

    /* Copying data from program header:
       Addr=0x0003000 Size=words=0x00004000 FileSize=0x00004000 Offset=0x00003000 Flags=0x00000005
       Addr=0x0013000 Size=words=0x0000266a Offset=0x00013000 Flags=0x00000005
       ...
       Addr=0x0050000 Size=words=0x0000067d FileSize=0x000000ab Offset=0x00020000 Flags=0x00000006
    */

    10:09:18 Build Finished (took 49.0ms)

USBBulkLoopAutoEnum/Debug/USBBulkLoopAutoEnum.img
```

Figure 3-11. Generate USBBulkLoopAutoEnum.img

3.3 Loopback Test Procedure

Open [Start] → [Cypress] → [Control Center]

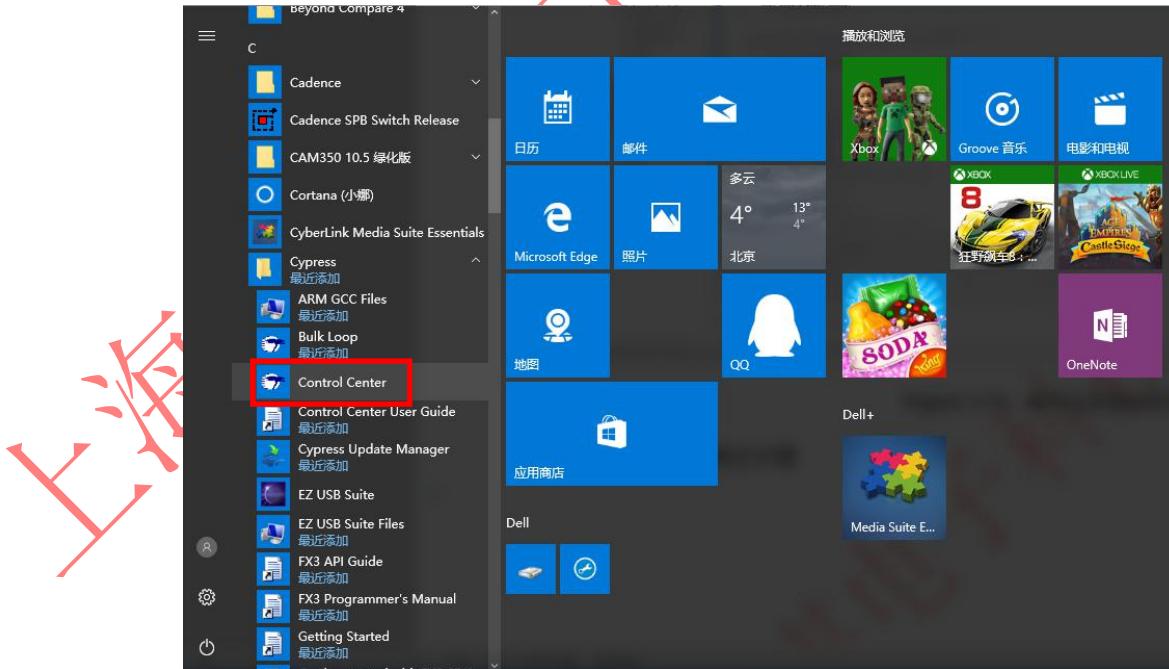


Figure 3-12. Open Control Center

USB Control Center GUI is shown in below image, the USB 3.0 development board is enumerated as Cypress FX3 USB Bootloader Device:

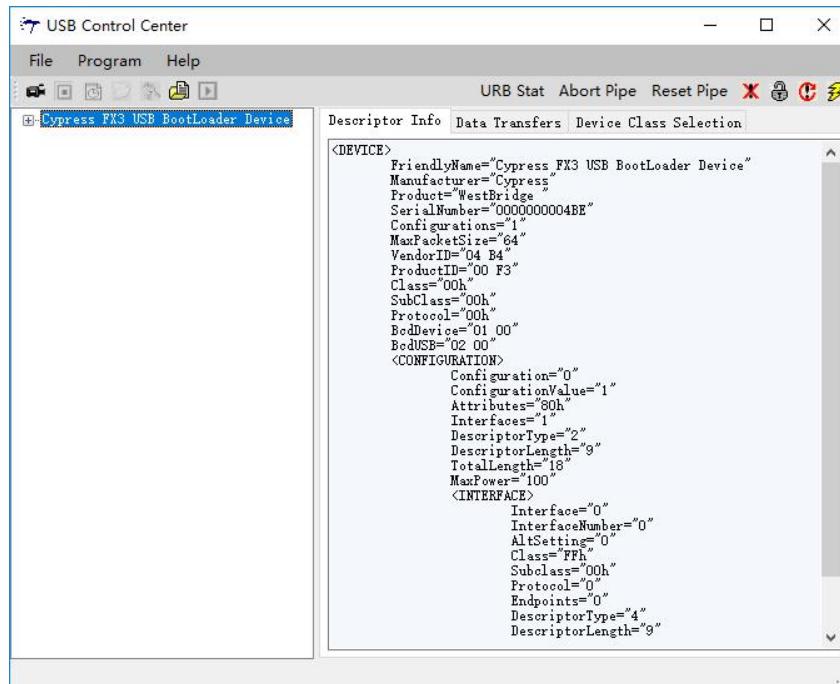


Figure 3-13. Enumerated Device

Load the generated USBBulkLoopAutoEnum.img into CYUSB3014 internal RAM, Click [Program] → [FX3] → [RAM]:

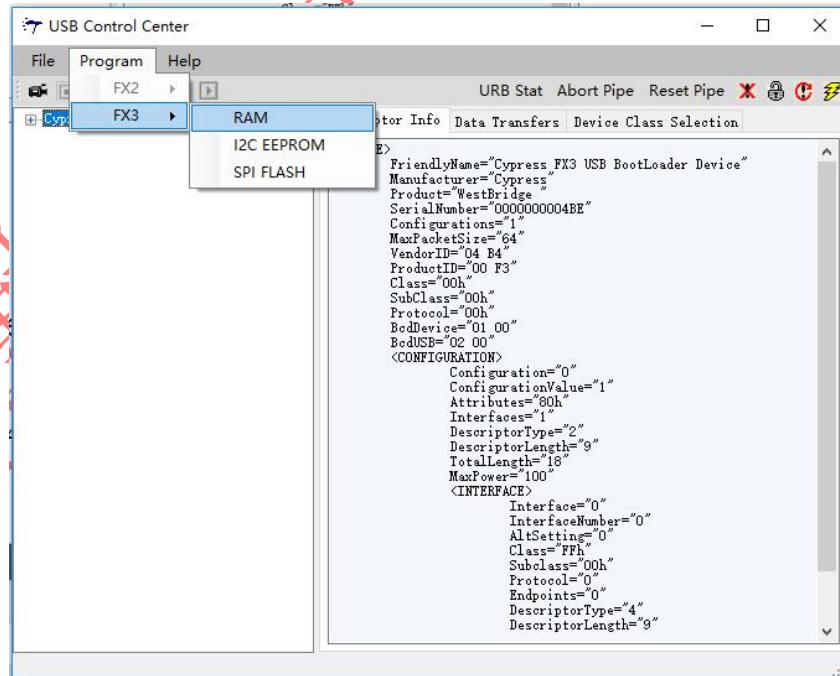


Figure 3-14. Download USBBulkLoopAutoEnum.img into RAM

Choose the USBBulkLoopAutoEnum.img in the Debug folder, and click [Open] button:

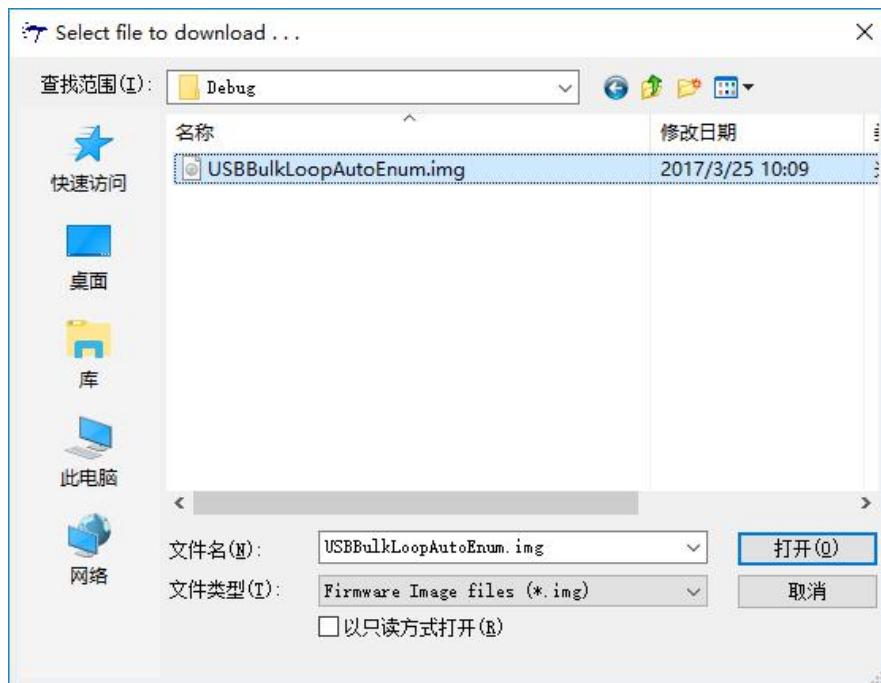


Figure 3-15. Choose USBBulkLoopAutoEnum.img

After the USBBulkLoopAutoEnum.img successfully downloaded into CYUSB3014 internal RAM, the USB 3.0 development board will be enumerated as Cypress FX3 USB BulkloopExample Device. More device info could be found in the Descriptor Info Tab.

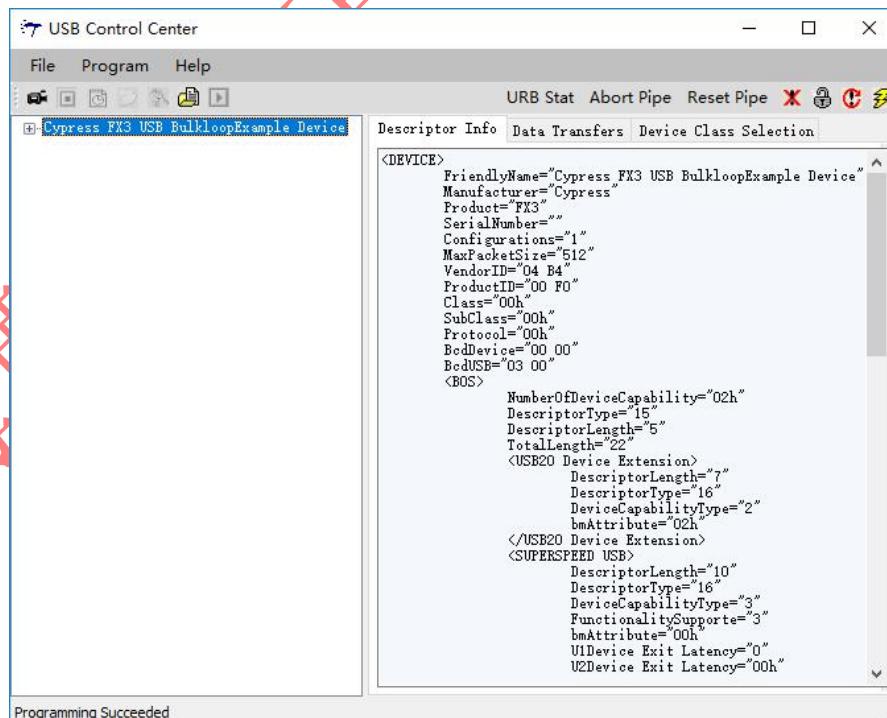


Figure 3-16. Cypress FX3 USB BulkloopExample Device

Fill some test data into the Data to Send (Hex) edit box, e.g. "12 34 56 78 90 AB CD EF". And perform below three steps in sequence to transfer the above test data:

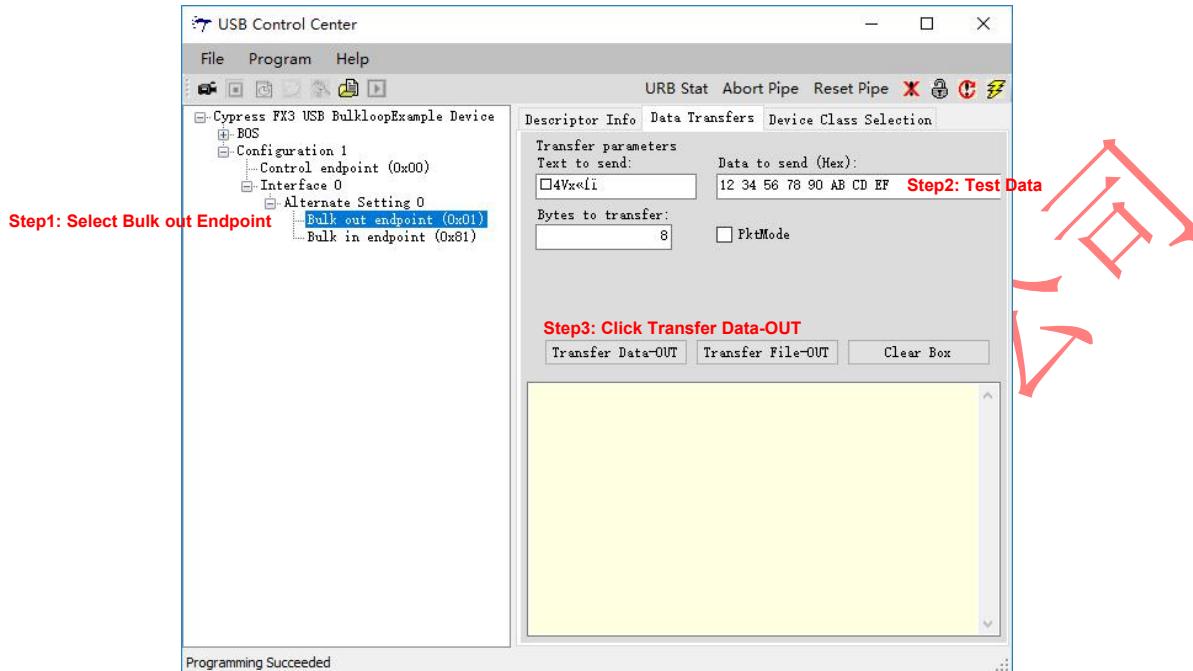


Figure 3-17. Send Test Data

After the test data correctly transferred into CYUSB3014, the USB Control Center GUI will display info: BULK OUT Transfer Completed. And then perform below two steps to receive the test data from CYUSB3014. Below image shows the USB Control Center correctly receives the test data: 12 34 56 78 90 AB CD EF.

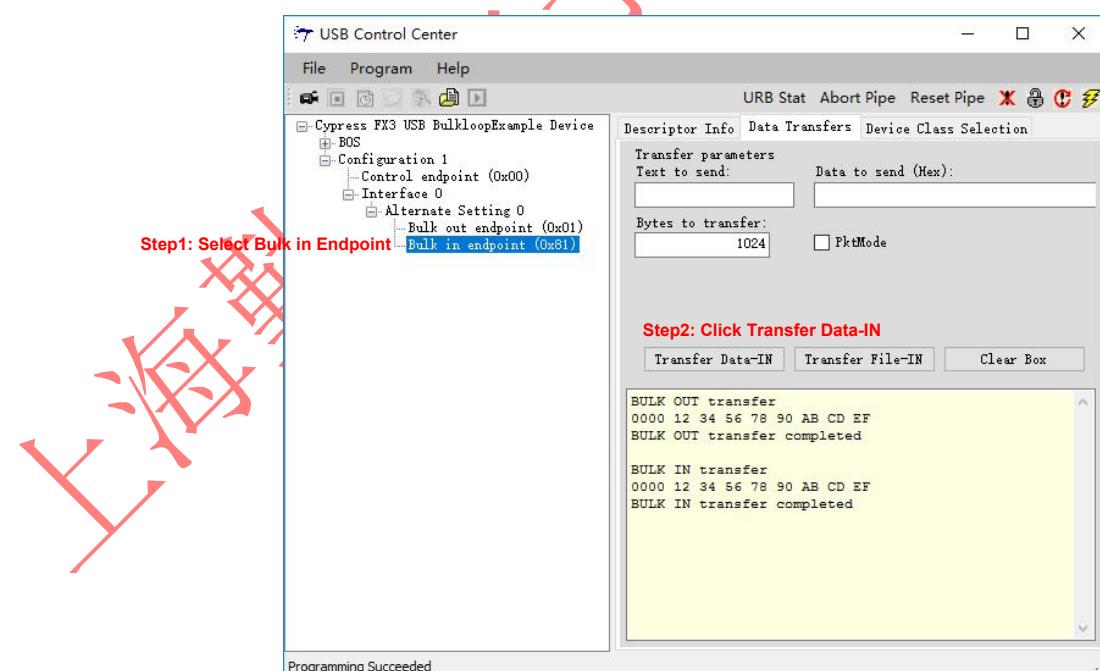


Figure 3-18. Receive Test Data

With this loopback test firmware, users could perform USB 3.0 bulk transfer test with Bulk Loop software, Click [Start] → [Cypress] → [Bulk Loop]:

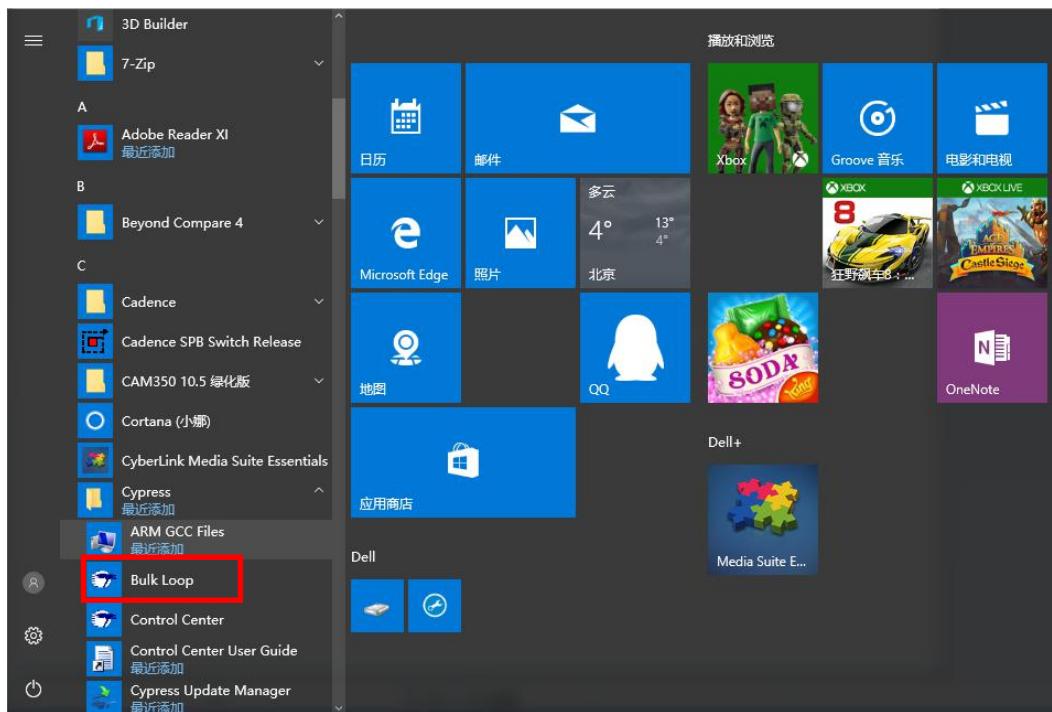


Figure 3-19. Open Bulk Loop Test Software

With default settings, click Start button in below GUI. Users could check whether there will some error info appear during the bulk loop test:

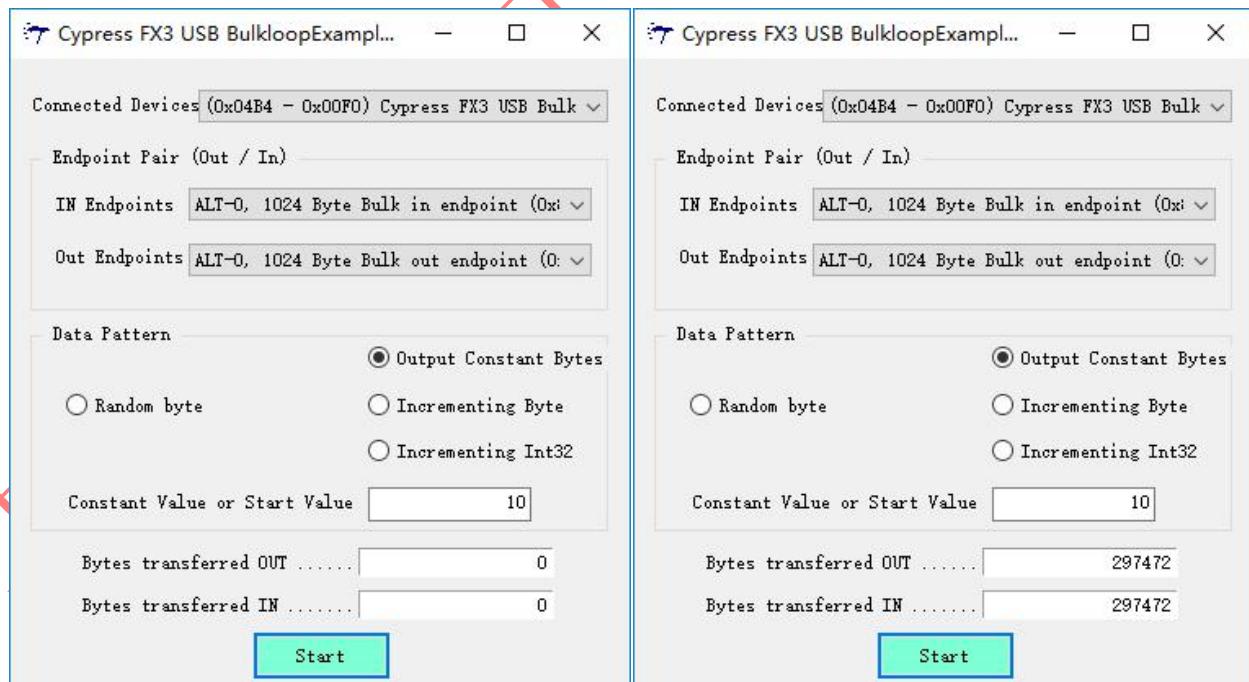
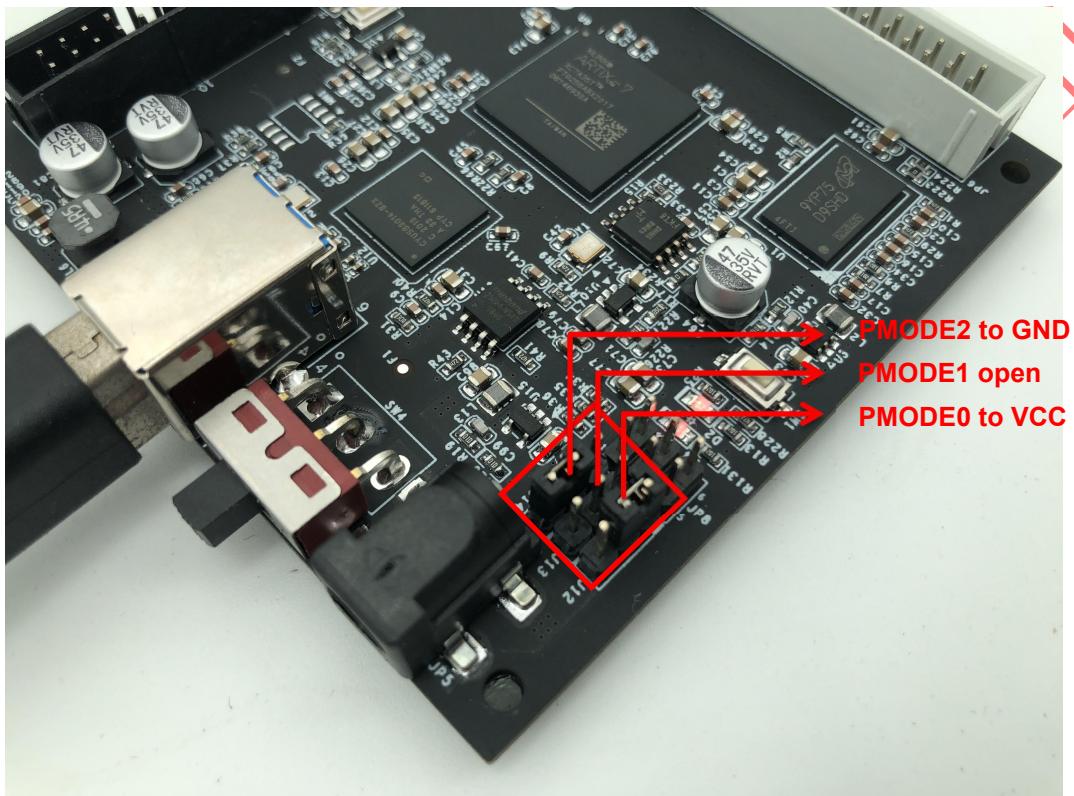


Figure 3-20. Bulk Loop Test

4. SPI Flash Boot Up Test

4.1 Hardware and Software Preparation

The below image shows the PMODE Pins' configuration which could make the CYUSB3014 boot up from external SPI Flash:



~~Figure 4-1. PMode Pins Configuration for SPI Flash Boot~~

Below table shows the detailed SPI Flash boot settings:

PMODE[2:0] Pins			Boot Option
PMODE[2]	PMODE[1]	PMODE[0]	
Z	0	0	Sync ADMUX (16-bit)
Z	0	1	Async ADMUX (16-bit)
Z	0	Z	Async SRAM (16-bit)
Z	1	1	USB Boot
1	Z	Z	I2C
Z	1	Z	I2C; on failure, USB Boot is enabled
0	Z	1	SPI; on failure, USB Boot is enabled
FX3S Specific Boot Options			
Z	1	0	PMMC Legacy
0	0	0	S0-port (eMMC); On Failure, USB Boot is enabled
1	0	0	S0-port (eMMC)

Z = Pin is floating; left unconnected.

~~Figure 4-2. SPI Flash Boot Up Truth Table~~

4.2 SPI Flash Boot Up Test

Before perform the SPI Flash boot test, users need to set the CYUSB3014 in default **USB Boot** mode. And then plug in the USB 3.0 cable into the development board. Click [Start] → [Cypress] → [Control Center]. Select Cypress FX3 USB Bootloader Device in the Control Center GUI:

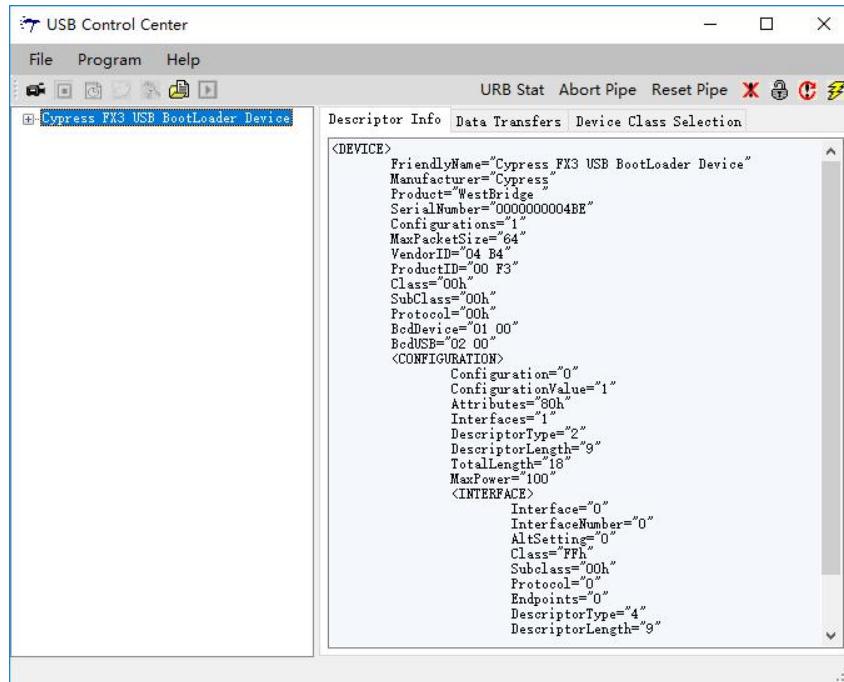


Figure 4-3. Open Control Center

Click [Program] → [FX3] → [SPI Flash]. Choose the generated USBBulkLoopAutoEnum.img compiled by EZ USB Suite in Debug folder and then click [Open] button:

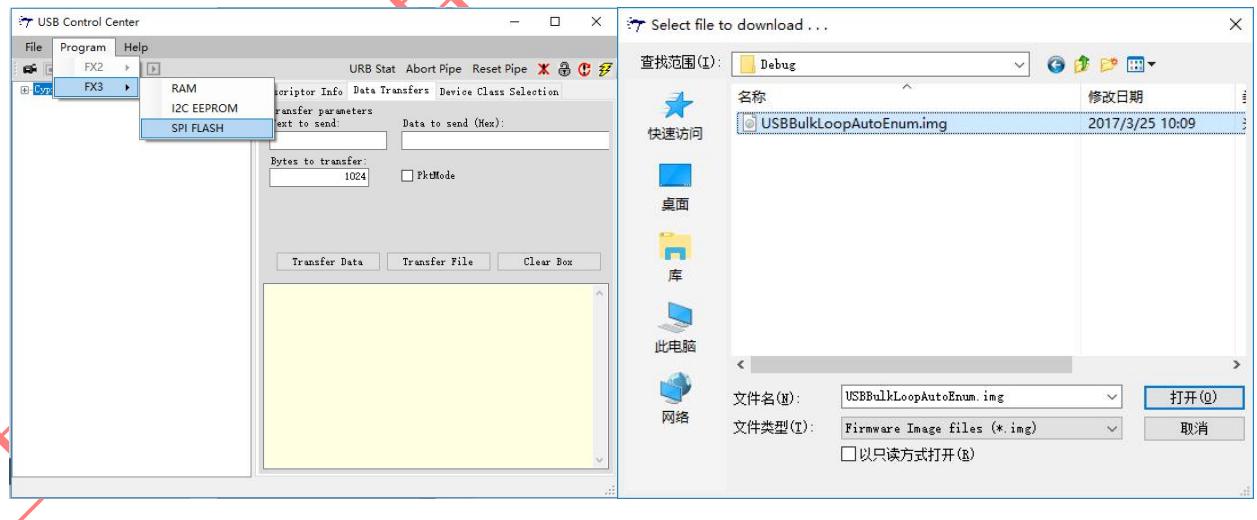


Figure 4-4. Download USBBulkLoopAutoEnum.img

After the USBBulkLoopAutoEnum.img correctly programed into SPI Flash, the USB Control Center will display some info: Programming of SPI Flash Succeeded:

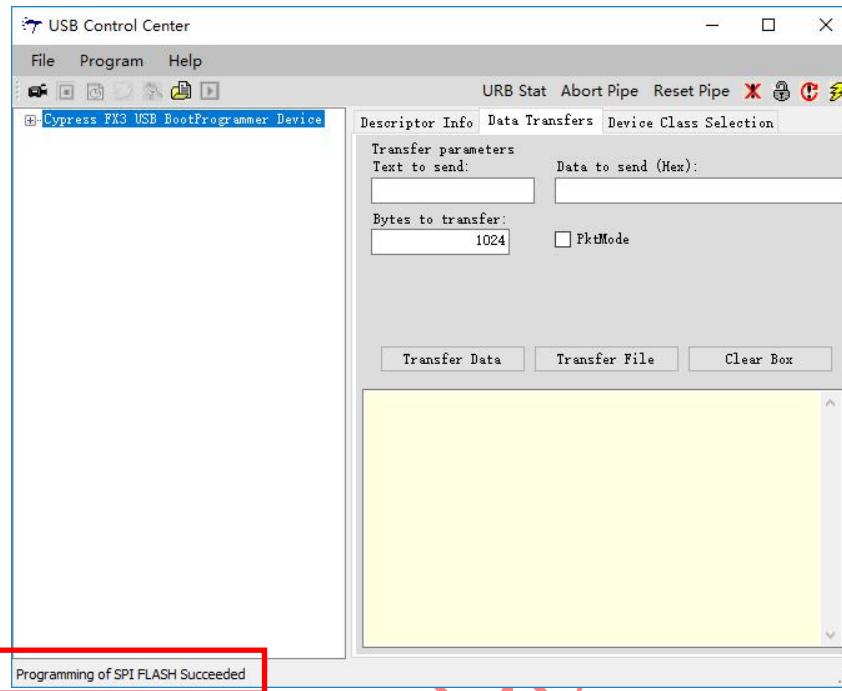


Figure 4-5. Program SPI Flash

Now, users could set the CYUSB3014 into SPI Flash boot mode as described at the beginning of this chapter and re-power on the board. If nothing goes wrong within the previous steps, the CYUSB3014 will be enumerated as Cypress FX3 USB BulkLoopExample Device which means the chip successfully boots up from SPI Flash. Then, users could re-perform all the loopback related tests described in chapter 3.

5. USB 3.0 Performance Test

5.1 Hardware and Software Preparation

Set the CYUSB3014 into **USB Boot mode**, plug in the USB 3.0 cable. Click [Start] → [Cypress] → [Control Center], select Cypress FX3 USB Bootloader Device in the USB Control Center GUI.

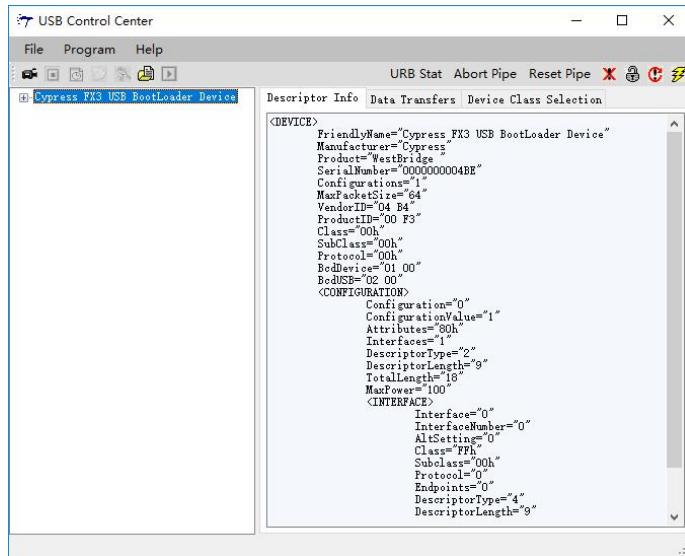


Figure 5-1. Open USB Control Center

5.2 Compile BulkSink Firmware

Click [Start] → [Cypress] → [EZ USB suite], then click [File] → [Import] and select Existing Projects into Workspace. Choose the cyfbulksrcsink project from the FX3 SDK, below image shows the example project directory:

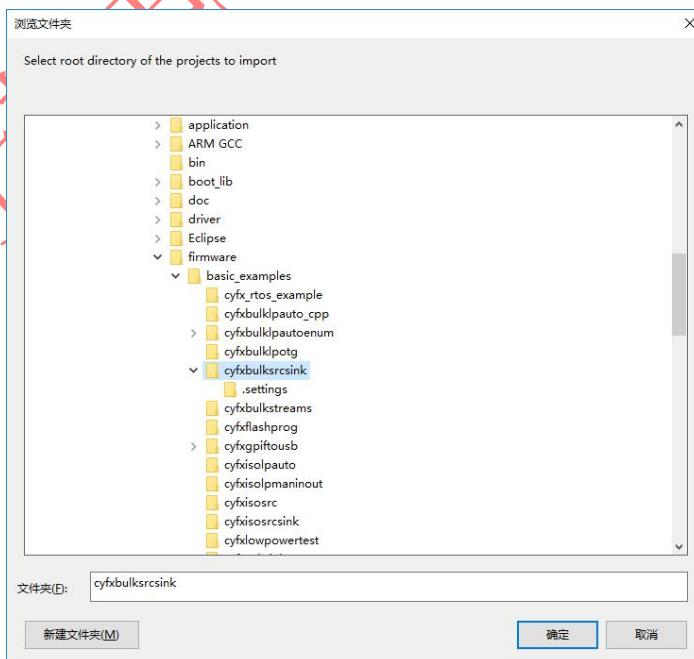


Figure 5-2. Open Project

After the cyfxbulksrcsink successfully lunched with the EZ USB Suite, right click the project and select the Build Project option. After the project correctly compiled, the USBBulkSourceSink.img will be generated in Debug folder.

```

C/C++ - USBBulkSourceSink\cyfxbulksrcsink - Cypress EZ USB Suite
File Edit Source Refactor Navigate Project CY Tools Run Window Help
Project Explorer
cyfxbulksrcsink
  +-- USBBulkLoopAutoEnum
  +-- USBBulksrcsink
    +-- Includes
    +-- Debug
      > cyfx_gcc_startup.o - [armv6]
      > cyfxbulksrcsink.o - [armv6]
      > cyfxbulksrcsink.o - [armv6]
      > cyfx.o - [armv6]
      > USBBulkSourceSinkelf - [arm]
        cyfx_gcc_startup.d
        cyfxbulksrcsink.d
        cyfxbulksrcsink.d
        cyfx.h
        Makefile
        objects.mk
        sources.mk
        subdir.mk
        subdr.mk
        USBBulkSourceSink.map
        USBBulkSourceSink.map
      > cyfx_gcc_startup.S
      > cyfxbulksrcsink.c
      > cyfxbulksrcsink.h
      > cyfx.h
      > Makefile
      > readme.txt
    +-- Problems Tasks Console Properties
    CDT Build Console [USBBulkSourceSink]
      Add=0x40003000 Size=(words)=0x00004000 FileSize=0x00004000 Offset=0x00003000 Flags=0x00000005
      Copying data from program header:
        Addr=0x40013000 Size=(words)=0x00002ef4 FileSize=0x00002ef4 Offset=0x00013000 Flags=0x00000005
        Copying data from program header:
          Addr=0x40030000 Size=(words)=0x000000c3 FileSize=0x000000ad Offset=0x00020000 Flags=0x00000006
      11:59:43 Build Finished (took 3s.423ms)
  
```

Figure 5-3. Compile cyfxbulksrcsink Project

5.3 BulkSink Test

Download the USBBulkSourceSink.img into CYUSB3014 internal RAM, click [Program] → [FX3] → [RAM]. Select the target image and click [Open] button:

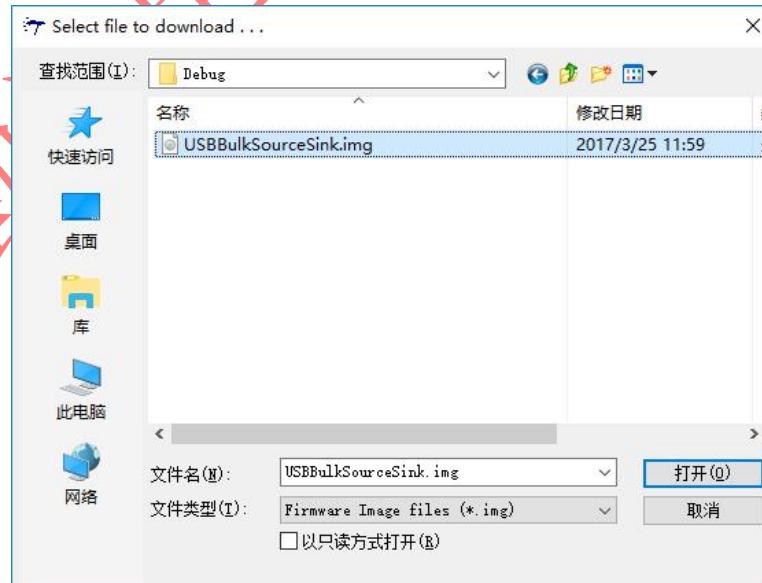


Figure 5-4. Download USBBulkSourceSink.img



After the firmware correctly downloaded into internal RAM, the USB 3.0 device will be enumerated as "Cypress FX3 USB StreamerExample Device" shown in below figure:

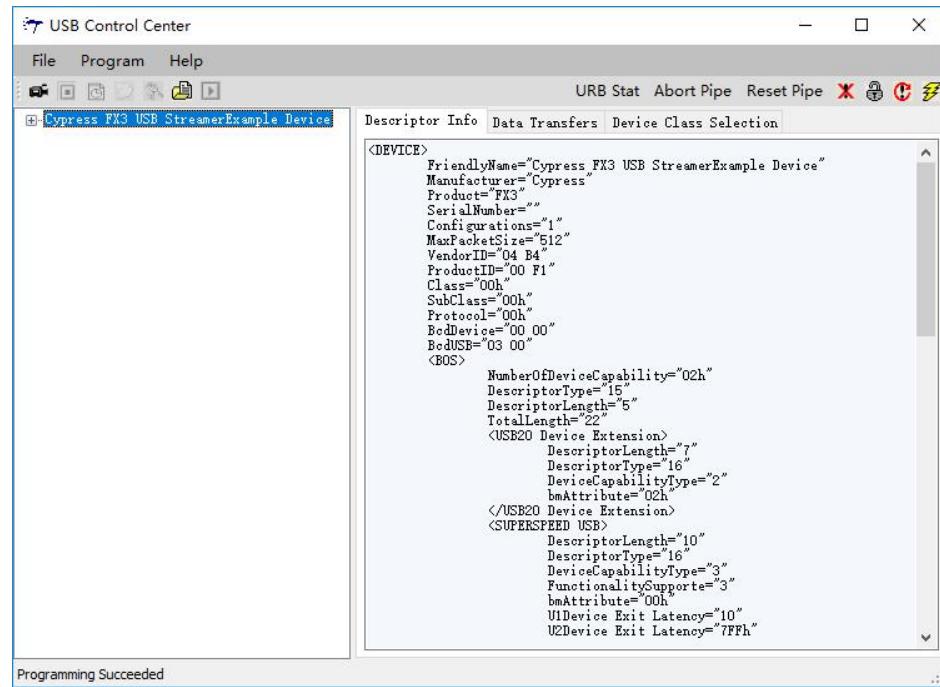


Figure 5-5. Cypress FX3 USB StreamerExample Device

Open BulkSink test software Streamer, click [Start] → [Cypress] → [Streamer] shown in below figure:

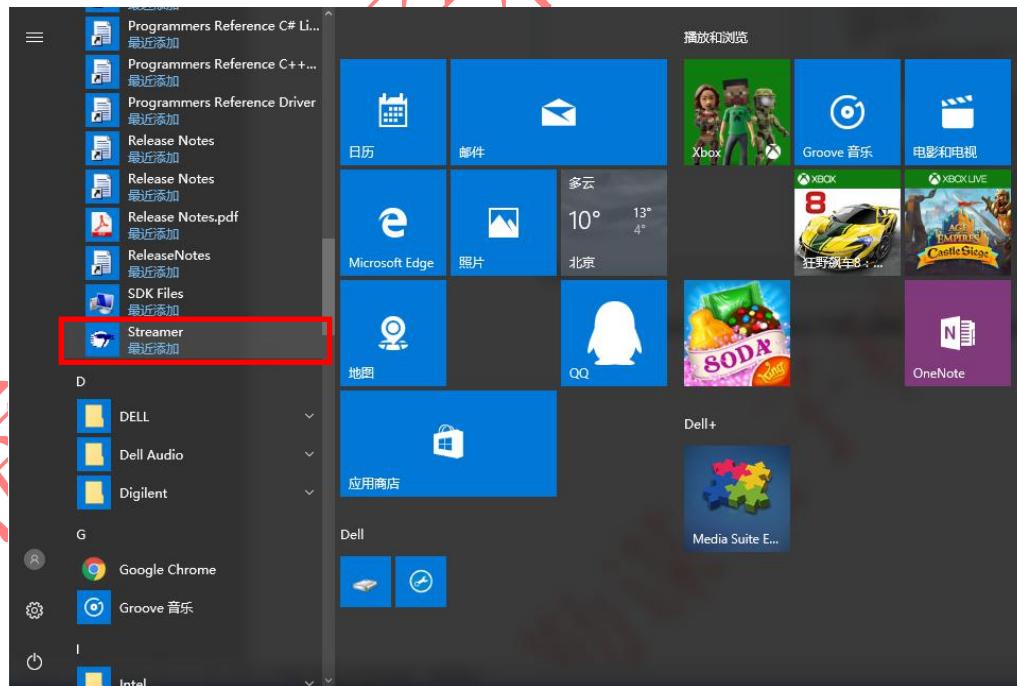


Figure 5-6. Open Streamer

After C++ Streamer GUI lunched, the Connected Devices and related configuration info will be automatically detected. Set all these parameters in default status and click [Start] button:

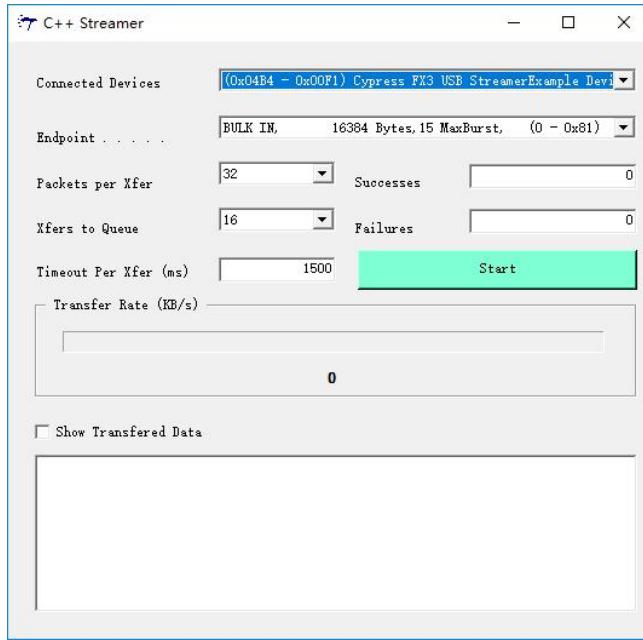


Figure 5-7. C++ Streamer GUI

With the USB 3.0 development board and the test computer mentioned in chapter 1.3, the USB 3.0 Transfer Rate is up to 436.8MB/s. Users could re-perform this kind of test with their own USB 3.0 development board and own test computers.

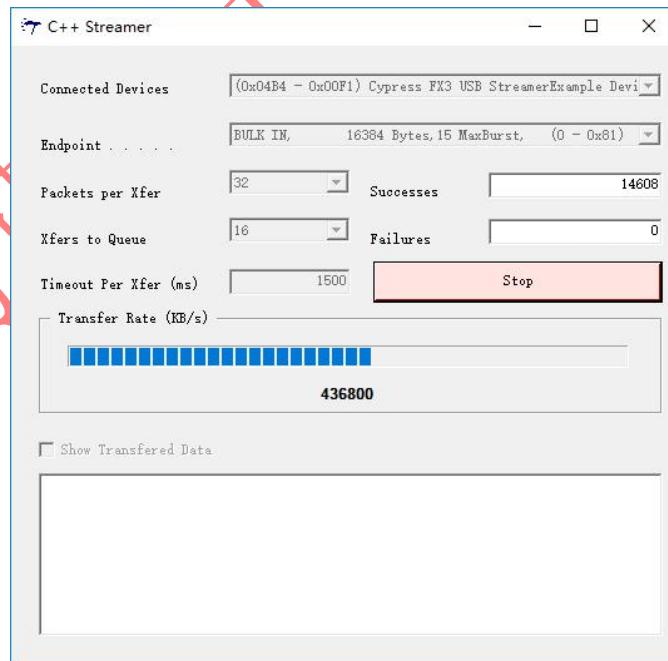


Figure 5-8. Streamer Test Result

6. USB 3.0 GPIF Tests With Artix-7 FPGA

6.1 SLAVE FIFO Test Preparation

Unzip the file \Release\Software\FX3 Firmware_slavefifo2b_ok.zip, below image shows the folder content:

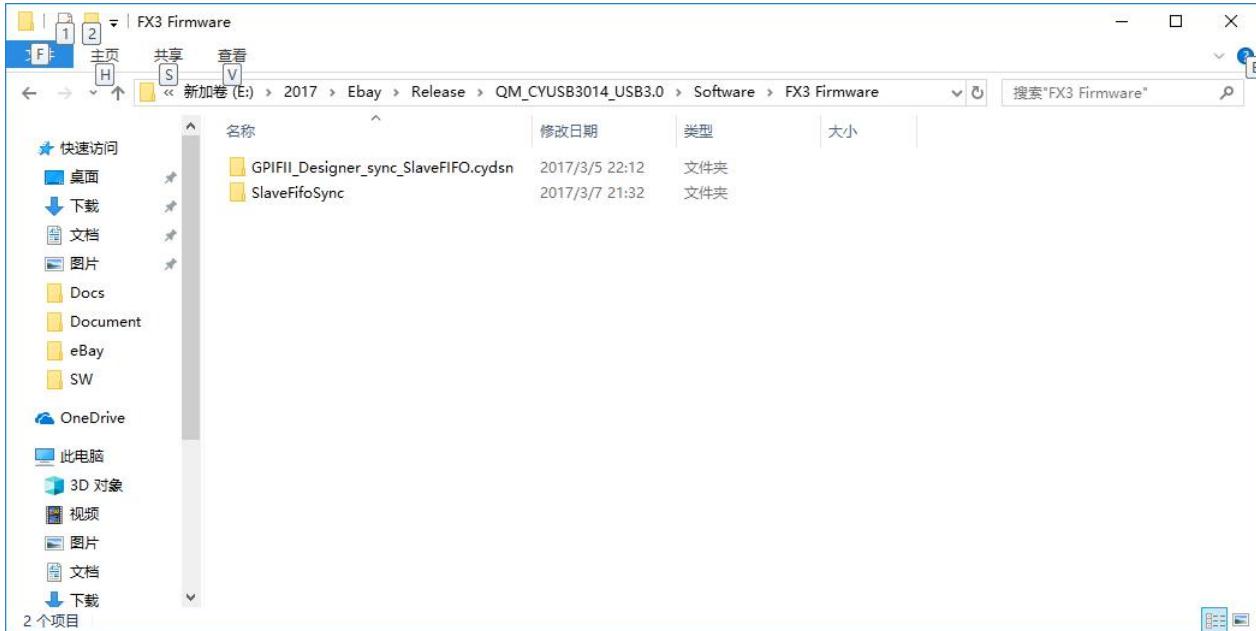


Figure 6-1. Upzip SlaveFifoSync Project

Right click the project and choose Build Project option, then the project will generate the executable slave FIFO test image: SlaveFifoSync.img.

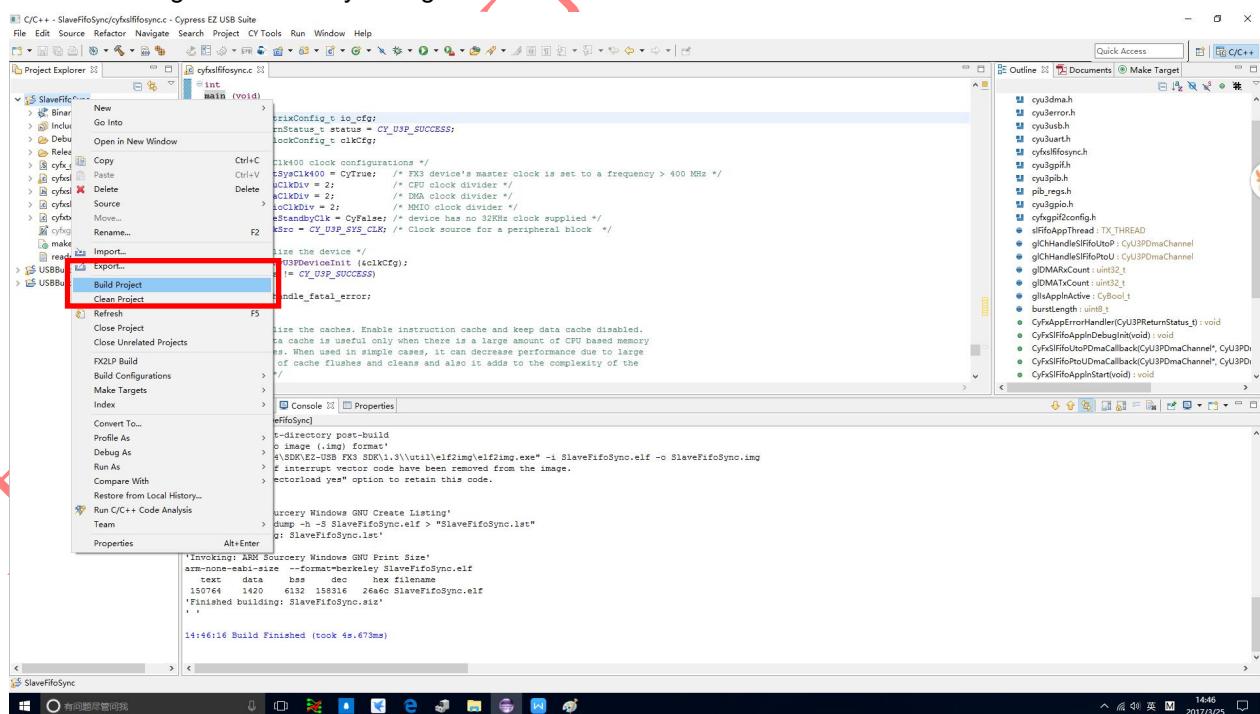


Figure 6-2. Build SlaveFifoSync Project

6.2 Slave FIFO Test Procedure

Unzip the file Release\Software\cyusb3014_slavefifo_Artix7.zip, use Vivado2016.4 to open the project project_led.xpr.

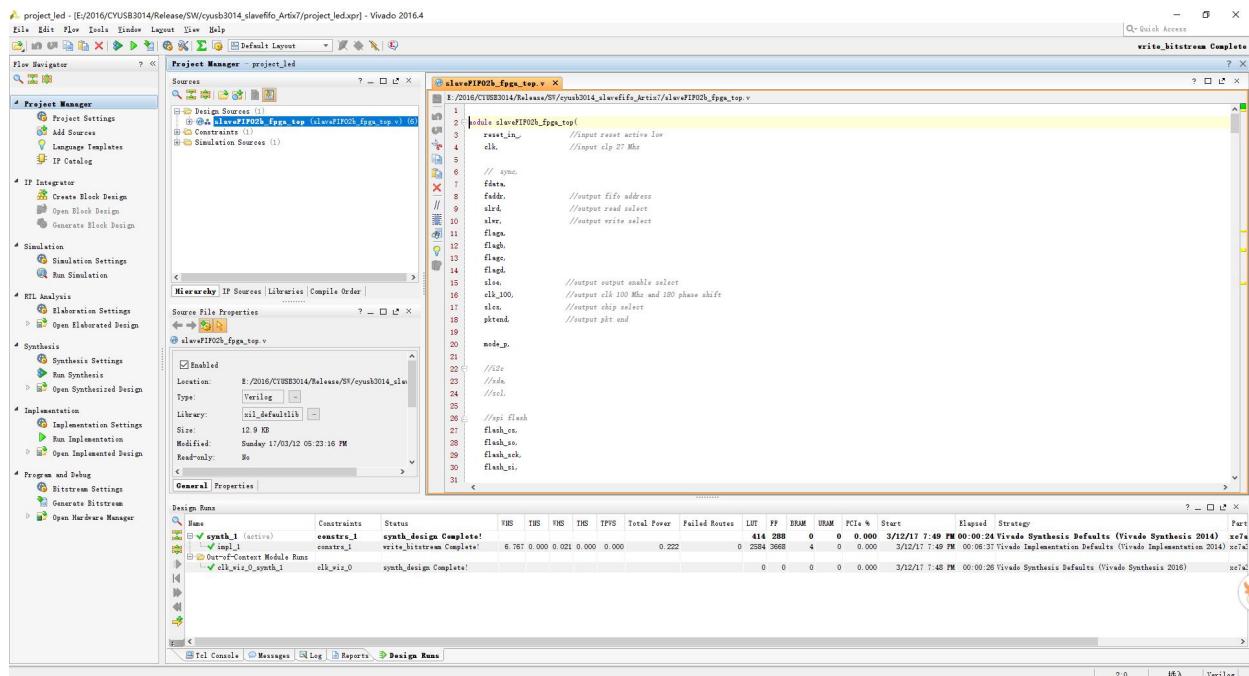


Figure 6-3. Open Slave FIFO Test Project

Then perform below three steps to generate executable bit file:

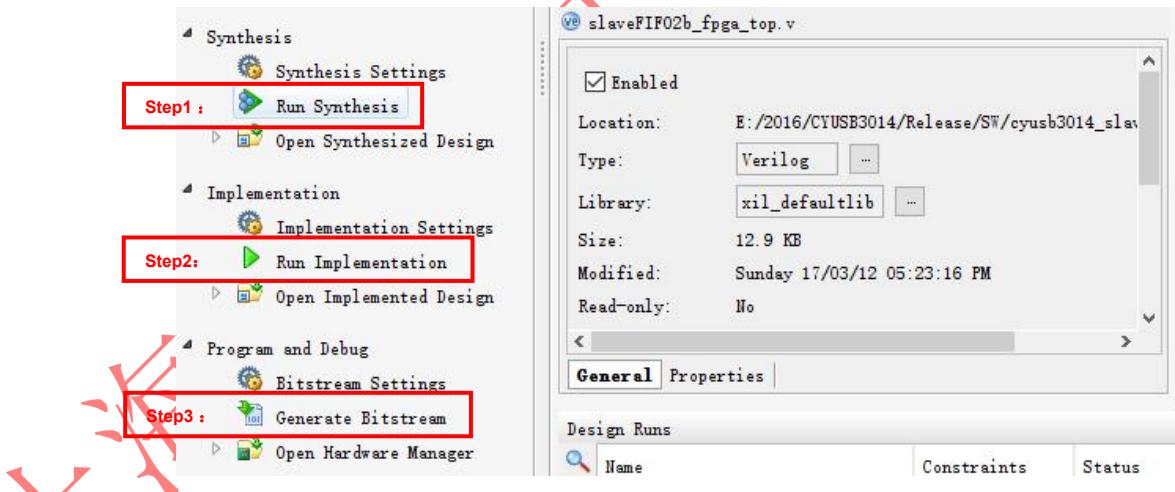


Figure 6-4. Generate Bit File

Correctly set up the hardware connection of Artix-7 board and CYUSB3014 board. Then in VIVADO 2016.4 environment, click [Open Hardware Manager] → [Open Target] → [Auto Connect]:



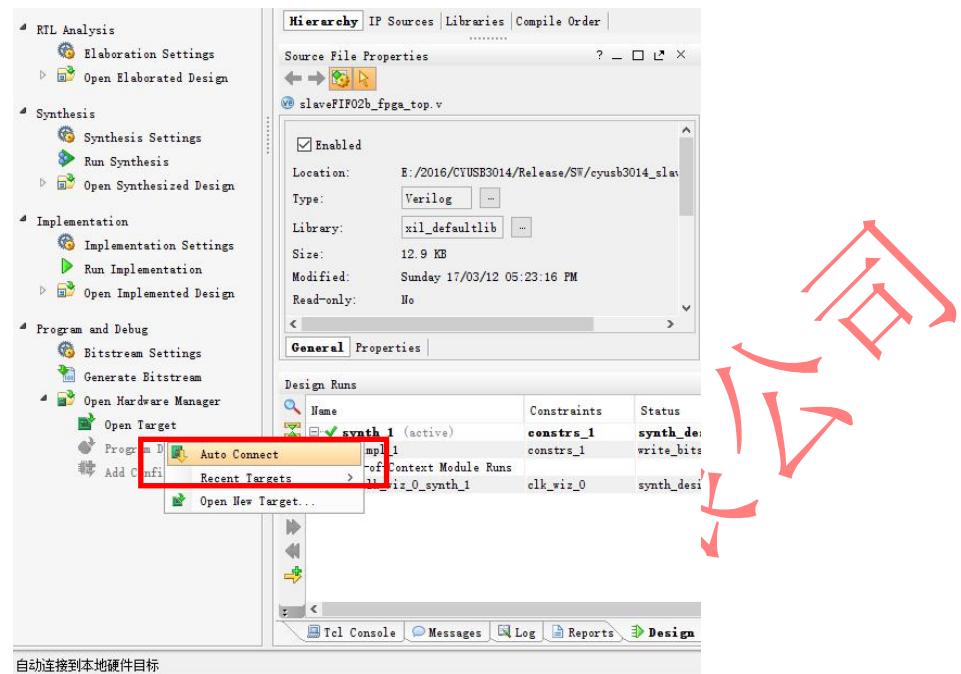


Figure 6-5. Connect Hardware

Right click the detected chip xc7a35t_0, and choose option Program Device. Then program the generated bit file slaveFIFO2b_fpga_top.bit into Artix-7 FPGA:

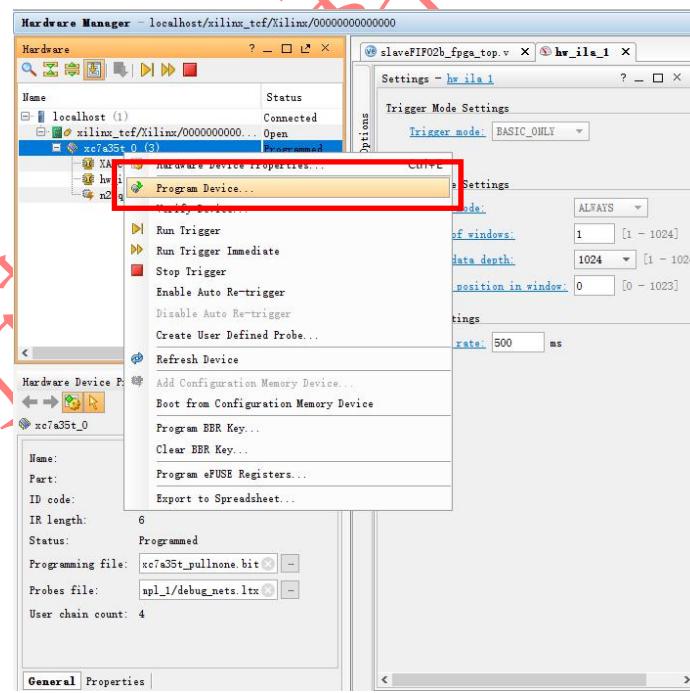


Figure 6-6. Program FPGA

After the bit file successfully downloaded into FPGA, users could monitor the slave FIFO bus signals between Artix-7 FPGA and CYUSB3014 chip by using ILA tool:

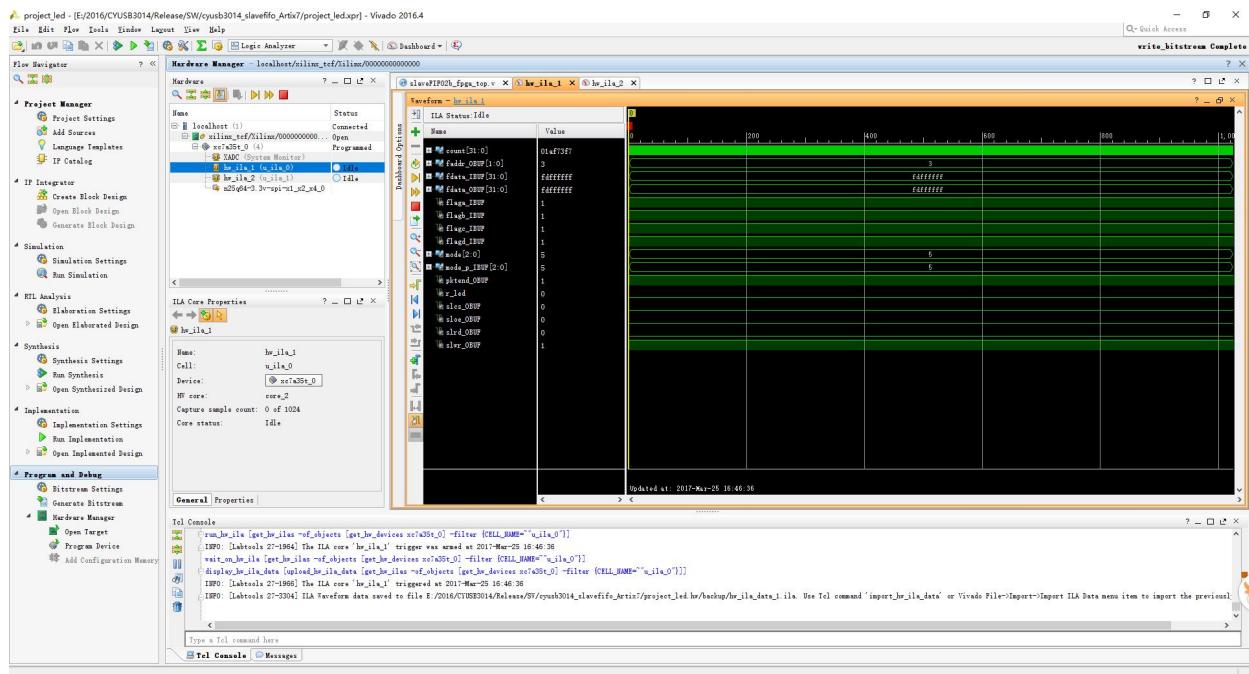


Figure 6-7. ILA Tool

上海勤謀電子



Set CYUSB3014 into USB Boot mode and use Control Center tool to download the SlaveFifoSync.img generated by EZ USB Suite environment. If this slave FIFO test firmware successfully downloaded, the USB device will be enumerated Cypress FX3 USB StreamerExample Device.

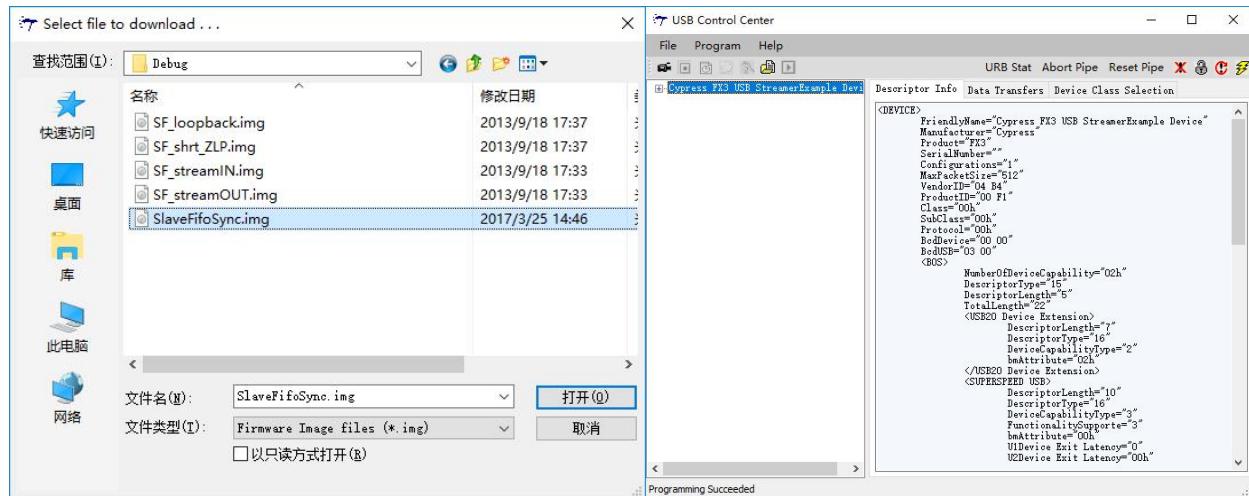


Figure 6-8. Download SlaveFifoSync.img

Follow below three steps to perform sending data to slave FIFO:

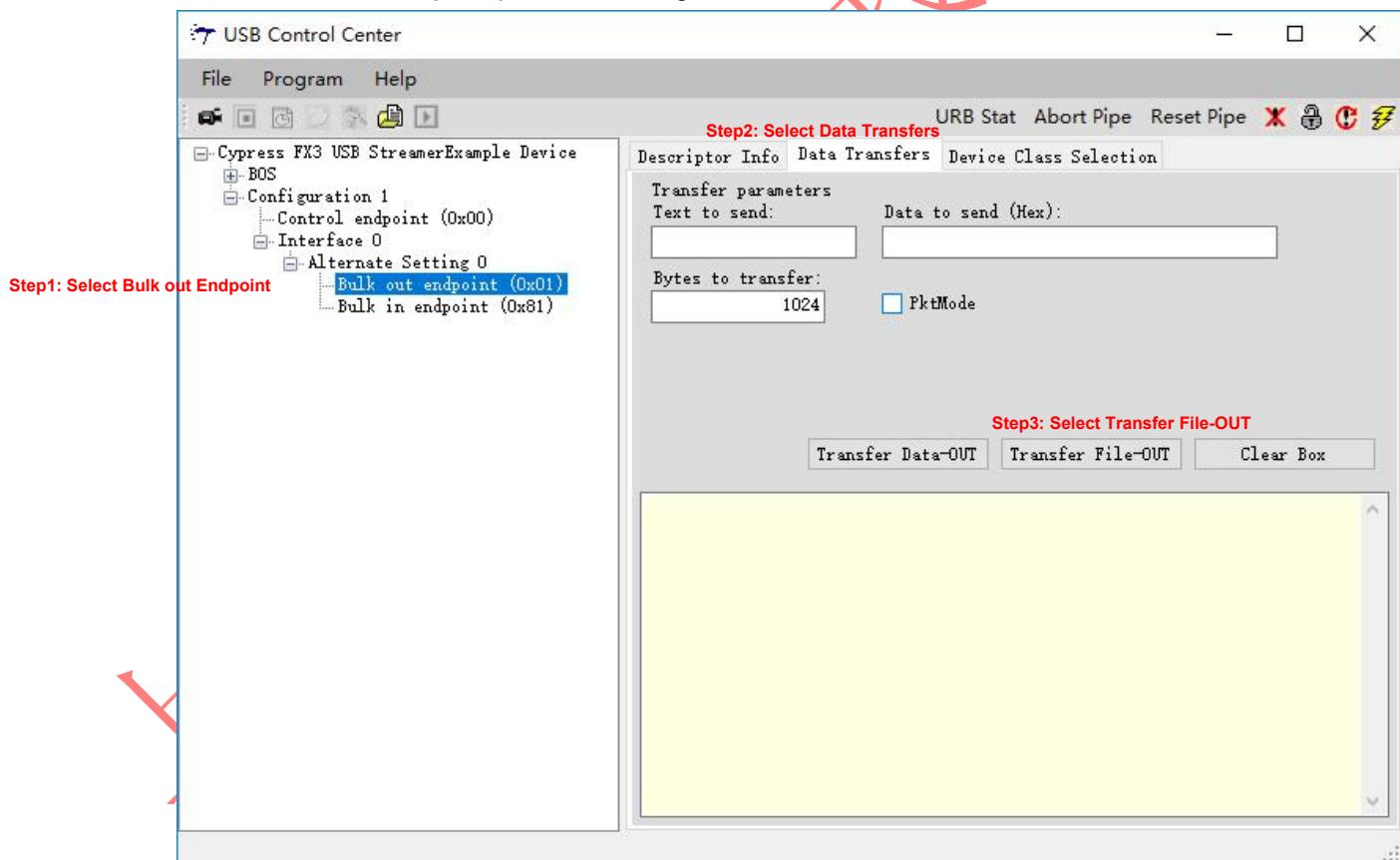


Figure 6-9. Send Data



In below GUI, choose the test data file \Release\Software\TEST.txt and then click [open] button:

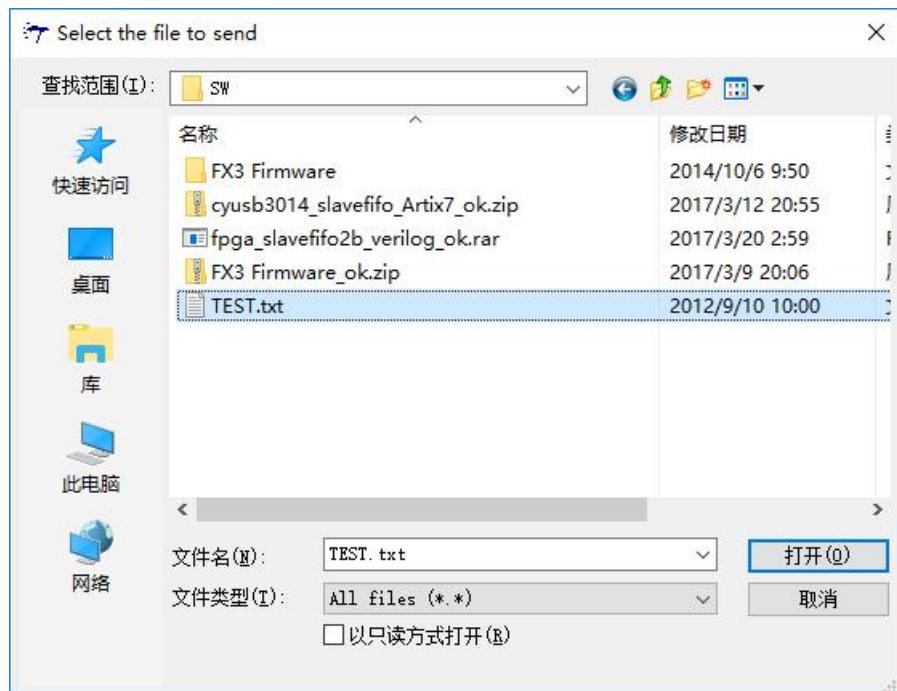


Figure 6-10. Choose Test Data

Click Transfer Data-OUT button to transfer the test data, the example test data are shown in below figure:

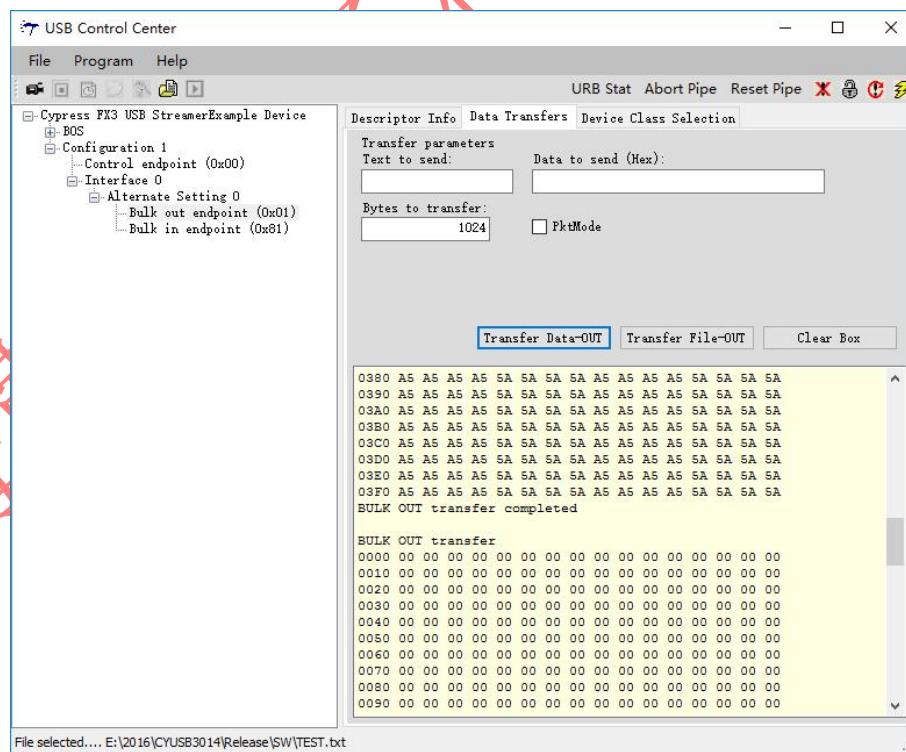


Figure 6-11. Send Test Data

Perform below two steps to receive the transferred test data. Select Bulk in Endpoint and then click the Transfer Data-IN button to check whether the received data is same as the original test data or not.

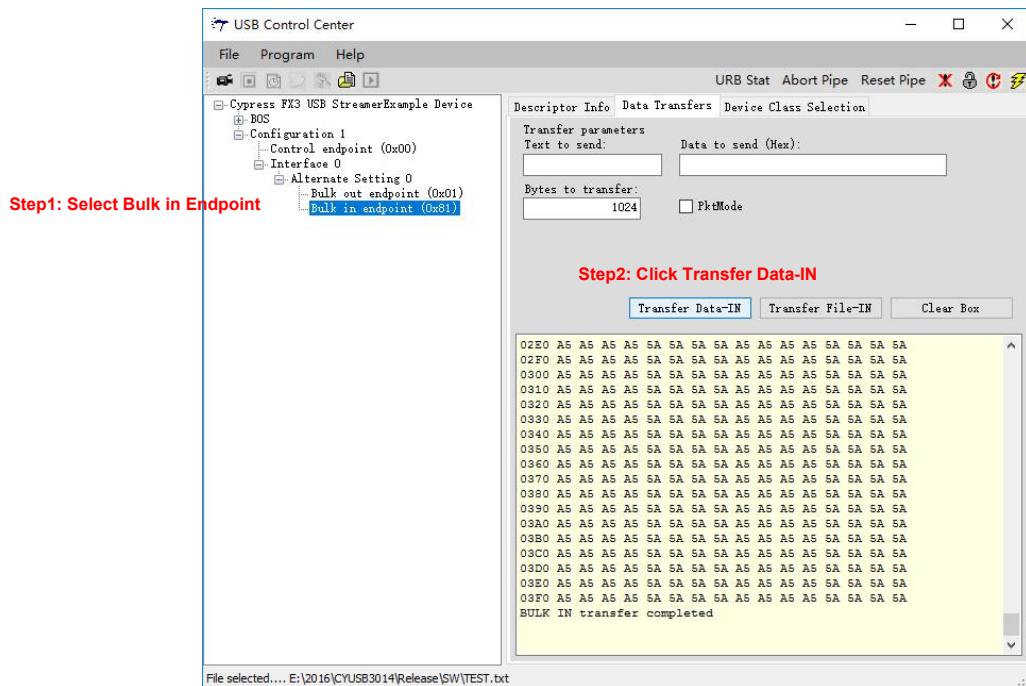


Figure 6-12. Check Received Data

6.3 streamIN Test Preparation

Unzip the file \Release\Software\FX3 Firmware_streamIN_ok.zip, below image shows the folder content:

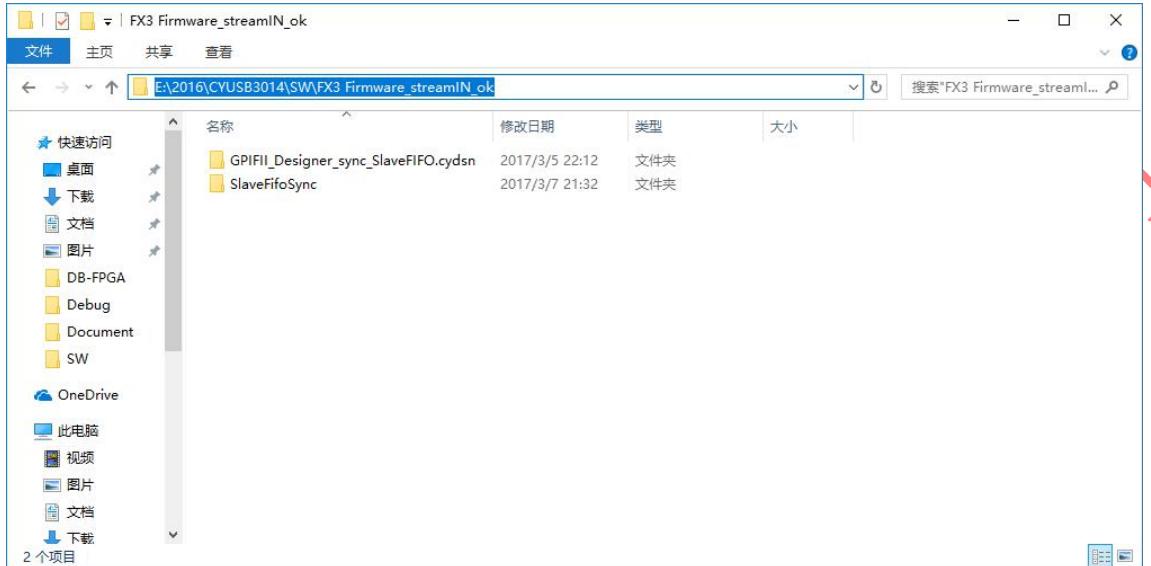


Figure 6-13. Upzip StreamIN Project

Use the Cypress UZ USB Suite import the StreamIN test project, make sure the macro definition STREAM_IN_OUT in cyfxslififosync.h is enabled. Then right click the project and choose Build Project option, the project will generate the executable slave FIFO test image: SF_streamIN.img.

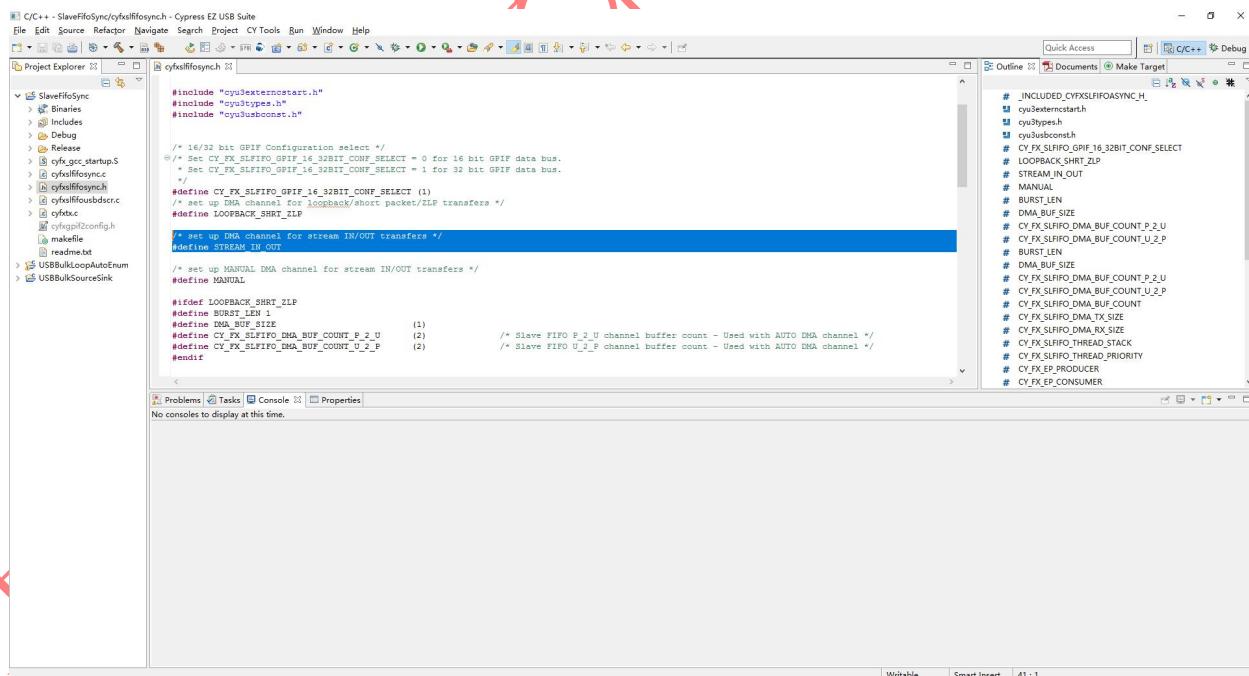


Figure 6-14. Compile streamIN project

6.4 streamIN Test Procedure

Unzip the file \Release\Software\cyusb3014_streamIN_Artix7.zip and use VIVADO 2018.3 to open the streamIN test project. After correctly synthesis and implement project, the slaveFIFO2b_fpga_top.bit will be generated. And then use VIVADO program tool to load the bit file into FPGA.

Then set the CYUSB3014 into USB Bootloader mode and download the SF_streamIN.img located in folder SW\FX3Firmware_streamIN_ok\SlaveFifoSync\Debug:

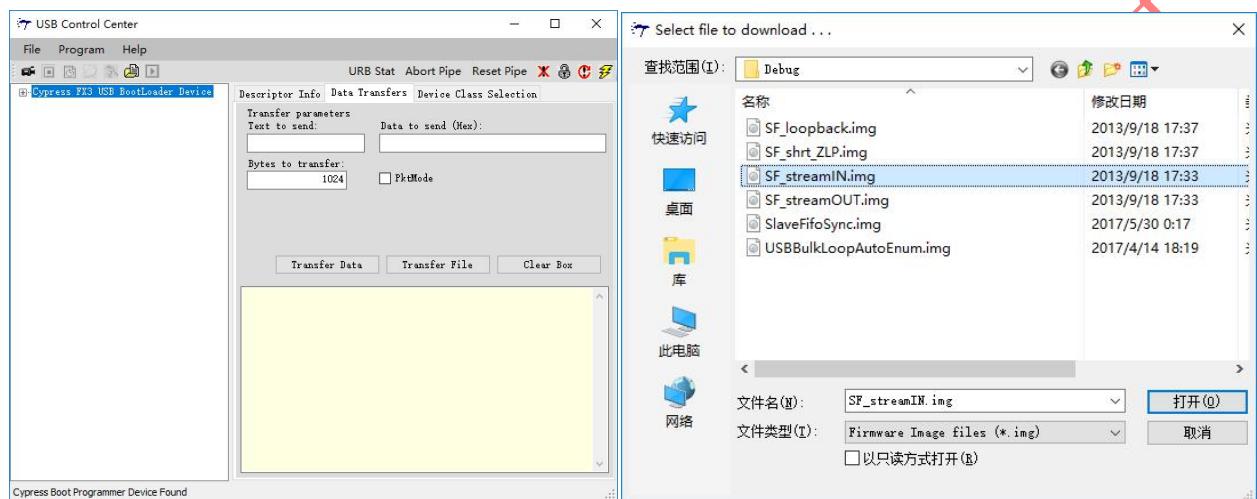


Figure 6-15. Download streamIN Test Firmware

If the streamIN test firmware successfully downloaded, the USB device will be enumerated as Cypress FX3 USB StreamerExample Device:

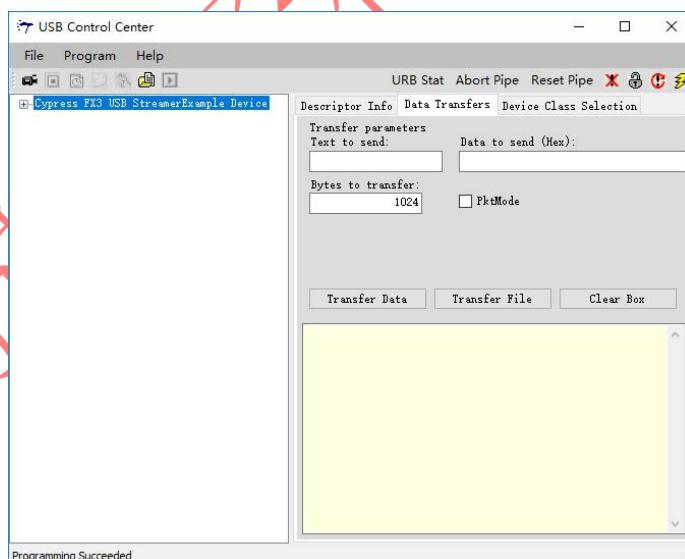


Figure 6-16. Download streamIN Test Firmware

Perform below two steps to test the streamIN functionality. Step1: Select Bulk in endpoint (0x81) and then click Transfer Data-IN button. Users could find the test pattern generated at FPGA side is continuously updated in below window.



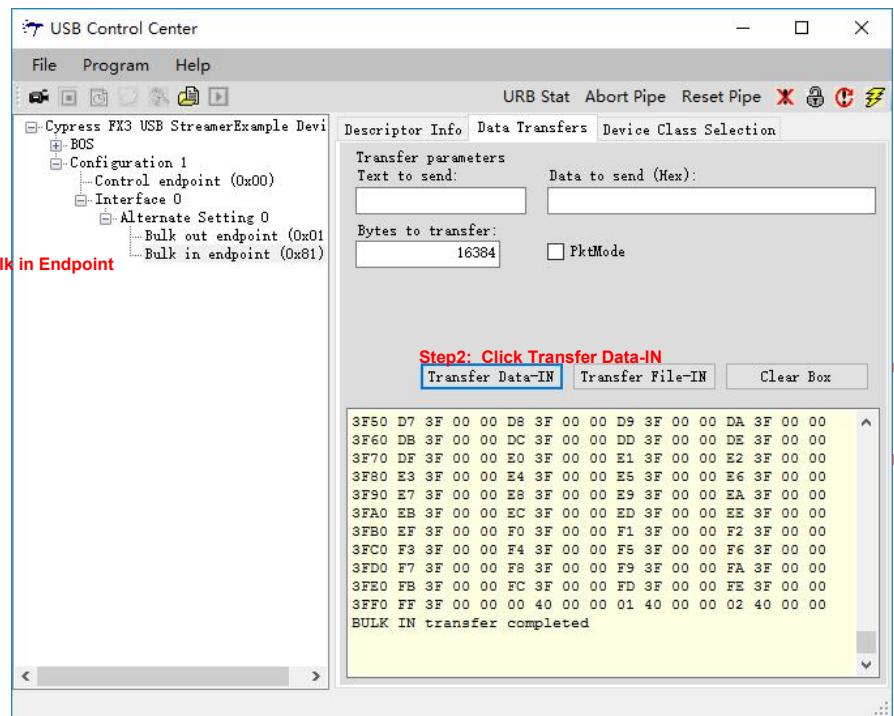


Figure 6-17. Received Test Data

In the other way, users could use the Cypress Streamer software to check the StreamIN Transfer Rate:

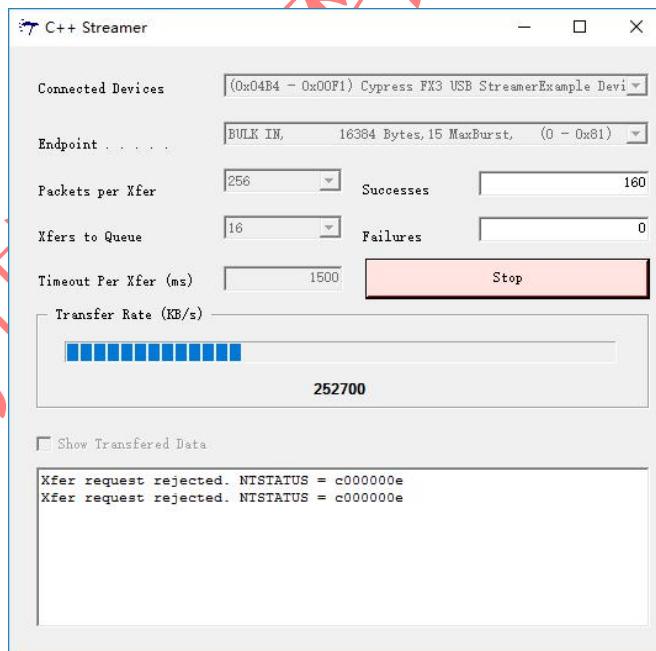


Figure 6-18. Transfer Rate

7. Reference

- [1] EZ-USB FX3 SuperSpeed USB Controller.pdf
- [2] MP2359.pdf
- [3] NCP1529-D.PDF
- [4] M25P80.pdf
- [5] Atmel-8734-SEEPROM-AT24C128C-Datasheet.pdf
- [6] 使用 EZ-USB® FX3™ 从设备 FIFO 接口进行设计.pdf

上海勤谋电子科技有限公司



8. Version

Doc. Rev.	Date	Comments
0.1	07/02/2020	Internal Release.
1.0	07/11/2020	Formal Release.

上海勤谋电子科技有限公司

