HPSDR PROJECT

July 19, 2006

1.0 Initial OZY EEPROM C0 Load Programming

Initially the OZY board will have a blank EEPROM installed. When the FX2 on the OZY enumerates on USB, it has a default VID of 04b4 and PID of 8613 when the EEPROM is missing or not programmed properly. We need to program the on board EEPROM with a "C0" load which basically means that the FX2 will get its VID and PID from the EEPROM instead of using the defaults of 04b4 and 8613. For the host assisted firmware load (meaning we are loading firmware from the computer) we use a VID of FFFE and a PID of 0007. To get your system ready to program the OZY EEPROM we have to accomplish two tasks:

1. Download the libusb-win32 driver files and load the driver.
2. Install the OZY USB utilities and libraries.

1.1 Installing libusb-win32 driver for OZY V1

1. Obtain a copy of the libUSB-Win32 device driver from:
   http://www.philcovington.com/HPSDR/libUSB
2. Place the files in a folder on your system named libUSB (or something you can remember). As a minimum you want to download the libusb0.sys, libusb0.dll, OZY_HOST_LOAD.cat, and OZY_HOST_LOAD.inf files.
3. Connect the OZY to the USB. If you do not already have the Cypress cyusb.sys driver or the Xylo xylousb.sys driver installed on your system Windows will start the New Hardware Wizard. If either of these drivers exist on your system you will probably hear the OZY enumerate but the New Hardware Wizard may not appear. If the New Hardware Wizard appears go to step 4. below. If you just hear the enumeration with no New Hardware Wizard, go to step 5. below.
4. When the New Hardware Wizard comes up you need to point it to the OZY_HOST_LOAD.inf file. Select not to search for the driver. Select to choose to install the driver. Select "Have Disk". Navigate to the folder where you placed the libusb0.sys and OZY_HOST_LOAD.inf files and select the OZY_HOST_LOAD.inf file. Click "Open". Click "OK". Click "Next" and then click "Finish". If a window pops up warning that the driver is unsigned click install anyhow. Open device manager and verify that there is a LibUSB-Win32 Devices entry and under that entry you see "OZY FX2LP - EEPROM missing" device. Proceed to step 6 below.
5. Open Device Manager and find the driver that was associated with the OZY. This may be the Cypress USB driver, the Xylo USB driver, or some other driver on your system that is associated with the VID of 04b4 and PID of 8613. Right click on the driver and select "Update Driver". Select not to search for the driver. Select to choose to install the driver. Select "Have Disk". Navigate to the folder where you placed the libusb0.sys and OZY_HOST_LOAD.inf files and select the OZY_HOST_LOAD.inf file. Click "Open". Click "OK". Click "Next" and then click "Finish". If a window pops up warning that the driver is unsigned click install anyhow. Open

device manager and verify that there is a LibUSB-Win32 Devices entry and under that entry you see "OZY FX2LP - EEPROM missing" device. Proceed to step 6 below.

6. The VID of 0x04b4 and PID of 8613 are now associated with the libusb0.sys driver. The next step is to program the EEPROM on OZY.

1.2 Programming the OZY EEPROM with a "C0" load

1. Download a copy of the OZY utilities from:

   http://www.philcovington.com/HPSDR/OZY/UTILS/

2. Place the files in a folder on your system named OZY_Utils. If you do not have the .NET 2.0 framework installed you will prompted to install it when you try to run the OZY utilities.
3. To be able to download the EEPROM on the OZY the FX2 will have to be running some firmware. We will load the OZYV1FW.hex file into the FX2. Open a Command Prompt and navigate to the folder where you saved the OZY utility files. At the command prompt type:

   load_firmware 0x04b4 0x8613 OZYV1FW.hex

4. Upon successful download of the firmware to the OZY, LED D10 will blink about once per second. You will also hear the OZY's FX2 enumerate again.
5. Now that OZY is running the firmware we can program the EEPROM. At the command prompt type:

   program_OZYEEPROM 0xfffe 0x0007

6. The OZY EEPROM should now be programmed with the VID of 0xFFFE and PID of 0x0007. Either unplug OZY from USB or turn off power to the OZY. Plug OZY back in or reapply power. The OZY should now enumerate with the new VID/PID. If Windows asks to install a driver follow step 4 in section "Installing libusb-win32 driver for OZY V1" above.
7. Repeat step 3 above to load the firmware, but this time use the command:

   load_firmware 0xfffe 0x0007 OZYV1FW.hex

1.3 Uploading the Cyclone II FPGA

1. Make sure that jumpers J23 and J24 on the OZY board are set for ***passive serial*** mode. J23 should be set in the "0" position and J24 should be set in the "1" position.
2. Open a command prompt and type:

   upload_FPGA 0xfffe 0x0007 blinkity.rbf

3. Upon successful upload of the .rbf file LEDs D1, D3-D9 should start blinking in sequence.

2.0 Using the OZY Utilities

load_firmware <VID> <PID> <filename>

description: loads the FX2 firmware intel hex format file

example: load_firmware 0xfffe 0x0007 OZYV1FW.hex

program_OZYEEPROM <VID> <PID>

description: programs the OZY EEPROM with a "C0" load with VID:0xfffe PID:0007

example: program_OZYEEPROM 0xfffe 0x0007

read_EEPROM <VID> <PID> <i2c_address> <start_pos> <length>

description: reads a serial EEPROM on the I2C bus, all command line parameters are entered in hex

example: read_EEPROM 0xfffe 0x0007 0x51 0x00 0x08 ← will read 8 bytes of the OZY EEPROM starting at address 0

read_I2C <VID> <PID> <i2c_address> <length>

description: reads <length> bytes from the I2C bus, all command line arguments are entered in hex

example: read_I2C 0xfffe 0x0007 0x10 0x08 ← will read 8 bytes from i2c address 0x10

set_LED <VID> <PID> <which_led = 0|1> <state = ON|OFF)

description: turns on or off LED D10 or D11 on OZY. LED D10 is 0, LED D11 is 1

example: set_LED 0xfffe 0x0007 1 ON ← will turn LED D11 on

upload_FPGA <VID> <PID> <filename>

description: uploads a .rbf file to the Cyclone II FPGA

example: upload_FPGA 0xfffe 0x0007 blinkity.rbf

write_EEPROM <VID> <PID> <i2c_address> <start_pos> <value>

description: writes one byte to the EEPROM on OZY at <start_pos>, all command line parameters are entered in hex

example: write_EEPROM 0xfffe 0x0007 0x51 0x09 0xff ← will write 0xff to address 0x09 of OZY's EEPROM

write_I2C <VID> <PID> <i2c_address> <value1> <value2> <value3> <count>

description: writes up to three bytes at a time to the I2C bus

example: write_I2C 0xfffe 0x0007 0x10 0x01 0x02 0x03 3 ← will write 3 bytes to I2C address 0x10

Appendix A - Location of Driver Files on Windows XP

There are two folders where Windows stores information about the drivers that it uses:

1. Windows\System32\Drivers\
2. Windows\Inf

- ✓ Windows alsos store information in the registry about the current driver set.
- ✓ For USB drivers it uses the class ID (CLSID): {36FC9E60-C465-11CF-8056-444553540000} which is associated with the "Universal Serial Bus controllers" branch in Device Manager. Both the Cypress cyusb.sys and xylousb.sys drivers use this class ID.

- ✓ The libUSB-Win32 driver libusb0.sys uses it's own CLSID: {EB781AAF-9C70-4523-A5DF-642A87ECA567} which is associated with the " LibUSB-Win32 Devices" branch in the Device Manager.

You can find the above CLSIDs in the HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\ of the Windows Registry

- ✓ HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{36FC9E60-C465-11CF-8056-444553540000} : Universal Serial Bus controllers
- ✓ HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{EB781AAF-9C70-4523-A5DF-642A87ECA567} : LibUSB-Win32 Devices

When Windows installs a driver, it moves the .sys file into the Windows\System32\Drivers\ folder and it moves the .inf file into the Window\Inf folder. Windows renames the inf file to something like oemXX.inf, where XX is a sequentially generated number. To find out what name that Windows has given the inf file you need to look in the above CLSID entries in the registry. For example under the HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{36FC9E60-C465-11CF-8056-444553540000} key, there will be multiple sub-keys numbered 0000, 0001, 0002, and so on. You can open these keys are look at the DriverDesc to see which driver it is associated with. On my system the Xylo FPGA board is associated with key 0006 with DriverDesc: "Xylo FPGA board - www.fpga4fun.com", InfPath: " oem21.inf", MatchingDeviceId: " usb\vid_04b4&pid_8613". If I were to navigate to the Windows\Inf folder I would find an oem21.inf file which is the renamed xylousb.inf file.

To remove the Cypress cyusb.sys and Xylo xylousb.sys drivers I can delete the .sys file from the Windows\System32\Drivers\ folder and the .inf file (whose name I determined above by looking in the registry key) from the Windows\Inf folder. Note that if the driver is currently loaded in the system you may not be permitted to delete its .sys file. You will have to go the Device Manager, select the device from the list that is using that driver, and right click->Uninstall Driver.

The Xylo board presents a bit of a problem - it does not have an EEPROM connected to the FX2 so it always enumerates as VID 04b4 and PID 8613. Since we want to associate VID 04b4 and PID 8613 with the libUSB driver (so we can use the OZY program_EEPROM utility to program a "C0" load), when Xylo enumerates it will probably be associated with the libUSB driver instead of the xylousb.sys driver. There is a place on the bottom of the Xylo board to solder a serial EEPROM. It would probably be best if a 24LC128 is installed on the Xylo and the EEPROM programmed with a different VID/PID. The Xylo inf file can then be edited so that the new VID/PID combination loads the xylousb.sys driver. A look at the

Cypress inf file shows that there are some other VID/PID combinations that could be used such as the combination for one of the Cypress development boards.