TERIDIAN SEMICONDUCTOR CORP.

Simplifying System Integration™

73S12xxF
USB-CCID Linux DFU Host Application
Users Guide

April 27, 2009
Rev. 1.00
UG_12xxF_038

Teridian Semiconductor Corp., 6440 Oak Canyon, Suite 100, Irvine, CA 92618
TEL (714) 508-8800, FAX (714) 508-8877, http://www.teridian.com

Table of Contents

Figure

1. Introduction

The Linux DFU host application program, `dfu-util`, is an open source user space program that provides the host-side ability to upgrade firmware in a compatible device over a standard USB connection. To communicate with a device, this program uses the user space USB API as supplied by the library module `libusb` in the Linux operating system. A specialized driver is not needed.

For more information about and how to download the source code of `dfu-util` program, visit `http://wiki.openmoko.org/wiki/Dfu-util`.

The version of `dfu-util` host program described in this document is based on the original code obtained from openmoko.org website. The code has been modified specifically to communicate with a smart card reader device that has been programmed with the CCID-DFU version of the TSC smart card reader firmware, for the purpose of upgrading the firmware in that device. This program is also used as a tool to demonstrate and test the implementation of the device-side of the USB DFU protocol in TSC smart card reader firmware. From this point on in this user guide, the name `dfu-util` refers to the TSC-modified version of the program.

The physical setup of the USB CCID-DFU system is shown in Figure 1. This document applies to the gray box. The Reader Device (herein referred to as Device) is connected to the Linux host PC via a standard USB cable.

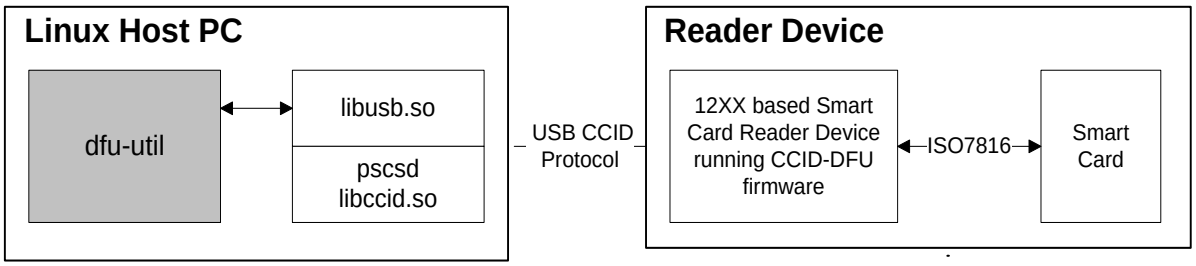

Figure 1: USB CCID-DFU System

2. Building the `dfu-util` Program

Follow these steps to build the `dfu-util` program:

1. Go to `http://wiki.openmoko.org/wiki/Dfu-util`. In the Source Code section of the website, it states that for certain versions of Linux, library modules such as `libusb-dev` need to be installed before building the original `dfu-util` program. Follow these instructions if this prerequisite applies to you.
2. Download the appropriate source package for the version of the Linux operating system that you use. As an alternative, if a svn client has been installed on your Linux PC, you can retrieve the uncompressed source files directly from the archives at openmoko.org by executing the following commands from command line: (Suppose there is a directory named `/home/user/` in your Linux PC.)

   ```
   cd /home/user
   svn co https://svn.openmoko.org/trunk/src/host/dfu-util/
   ```

   After running this command, go to step 4.

3. Depending on the type of source package that you downloaded in previous step, use the appropriate method to have the source files decompressed and installed. For example, if your operating system is Fedora 10, you should download an `rpm` package. In this case run the `rpm --install` command. For Ubuntu users, the package downloaded is in a `.deb` file. Double-click on the file

name to install the package. In the case of other versions of Linux, such as Slackware, the downloaded package may be a compressed file with a `.tgz` extension. To install it, run command `tar -xvf` on the file.

4. After the previous step completed, the original `dfu-util` package has been installed in the directory `/home/user/dfu-util`. (Your actual path name could be different.) Copy the file `dfu-util-src-tsc.tar` from the TSC release CD into this directory.

5. Create a directory named `/home/user/dfu-util/TSC`, and decompress the file `dfu-util-src-tsc.tar` into that directory as follows:

```
cd /home/user/dfu-util
mkdir TSC
tar –xvf dfu-util-src-tsc.tar –C ./TSC/
```

6. To build a `dfu-util` program that works with the TSC smart card reader device, replace the original source files with TSC-modified ones. To do so, change to the directory `/home/user/dfu-util`, save the original files, and then copy the files provided by TSC into directory `/home/user/dfu-util/src`. The command sequence below serves as an example. The actual directory names are probably different in your PC. Note that the TSC-modified source files have been placed in directory `/home/user/dfu-util/TSC` in previous step.

```
cd /home/user/dfu-util/
mkdir original-src-files
mv ./src/*.c original-src-files/
mv ./src/*.h original-src-files/
cp ./TSC/* ./src/
```

7. Run the following commands to compile and build `dfu-util` program.

```
cd /home/user/dfu-util
./autogen.sh
./configure
make
```

   When all commands complete successfully, the resulting executable binary file is `/home/user/dfu-util/src/dfu-util`.

3. Running the `dfu-util` Program

Super user privileges are required to run the `dfu-util` program. Temporarily log in as a super user before executing the program.

1. Make sure your smart card reader device has been programmed with a version of the TSC smart card reader firmware that supports the USB DFU protocol. An example of such version is `ccidusb-MSDFU.hex`.
2. Create a directory named `/home/user/dfu-util/images`, and place a firmware image file into this directory. For the purpose of illustration, we use the file name `ccidusb-MSDFU.hex`.
3. Connect your smart card reader device with the Linux host PC through a standard USB cable.
4. From command line, run the `dfu-util` program with the `-l` (list) option, as follows:

```
cd /home/user/dfu-util/src
./dfu-util –l
```

   This is to make sure the smart card reader device has initialized, and has been enumerated by the Linux host PC properly. You should see the following output from the `dfu-util` program.

```
 [root@lx95t src]# ./dfu-util -l
dfu-util - (C) 2007-2008 by OpenMoko Inc.
This program is Free Software and has ABSOLUTELY NO WARRANTY

Found Runtime: [0x1862:0x0001] devnum=2, cfg=0, intf=1, alt=0,
name="Teridian Semiconductors", str_idx=1
[root@lx95t src]#
```

Among the information displayed in the output above, take note of the vendor:product ID pair, and alt (alternate interface), whose value is 0x1862:0x0001, and 0, respectively. This information is needed for the next command.

5. To download (upgrade) the firmware in the device, you need to run `dfu-util` program with the following options:

```
-d 0x1862:0x0001 This option identifies the device through its vender ID and product ID.
-i 0             This option specifies the DFU interface by alternate interface number.
-D <file name>   This option specifies the image file to be downloaded into the device.
```

(For an explanation of all available command options for `dfu-util` program, run the program with the –h option.)

Now from the same directory as in step 4, execute `dfu-util` program again, this time with the options described above, as follows:

```
./dfu-util -d 0x1862:0x0001 -i 0 -D ../images/ccidusb-MSDFU.hex
```

While the program is being run, the following messages are displayed:

```
root@lx95t src]#
[root@lx95t src]# ./dfu-util -d 0x1862:0x0001 -i 0 -D /root/dfu-hex-
files/ccidusb-MSDFU.hex
dfu-util - (C) 2007-2008 by OpenMoko Inc.
This program is Free Software and has ABSOLUTELY NO WARRANTY

Opening USB Device 0x1862:0x0001...
Claiming USB DFU Runtime Interface...
Device really in Runtime Mode, send DFU detach request...
Resetting USB...
Opening USB Device...
Found Runtime: [0x1862:0x1100] devnum=3, cfg=0, intf=0, alt=0,
name="Teridian Semiconductor  ", str_idx=1
Claiming USB DFU Interface...
dfuStatus = 00 (OK)
dfuState  = 02 (dfuIDLE)

dfuIDLE, continuing
Transfer Size = 0x0010
Starting download:
[################################################################] finished!
dfuStatus = 00 (OK)
dfuState  = 08 (dfuMANIFEST-WAIT-RESET)

state(8) = dfuMANIFEST-WAIT-RESET, status(0) = No error condition is present
Done!
[root@lx95t src]#
```

4. Known Issues

During the testing at Teridian, a problem was encountered. The scenario is described below.

A string descriptor that a version of TSC CCID-DFU firmware supplies to identify the manufacturer of the device happens to be 16, 32, 48, or 64 characters long. That is, the length of the string descriptor is a multiple of 16. If the `dfu-util` program requests this string when the device is in DFU mode, the firmware may respond with a STALL. Since the provision of string descriptors is optional, however, this problem does not cause `dfu-util` program to fail. In this case, when the program is run with the `-l` option, the output displays the word `UNDEFINED` for the parameter "name", as shown below.

```
[root@lx95t src]# ./dfu-util -l
dfu-util - (C) 2007-2008 by OpenMoko Inc.
This program is Free Software and has ABSOLUTELY NO WARRANTY

Found Runtime: [0x1862:0x0001] devnum=2, cfg=0, intf=1, alt=0,
name="UNDEFINED", str_idx=1
[root@lx95t src]#
```

5. Related Documentation

The following 73S12xxF documents are available from Teridian Semiconductor Corporation:

71S1215F Data Sheet
71S1217F Data Sheet
73S12xxF Smart Card Terminal Controller Family Software User's Guide
73S12xxF Evaluation Board User's Guide
Teridian Flash Programming Tool
73S1215F, 73S1217F Boot Loader – DFU Class Firmware Application Note
73S1215F, 73S1217F Windows XP 32 USB CCID and DFU Drivers Installation Guide
73S1215, 73S1217F CCID Application Note

6. Contact Information

For more information about Teridian Semiconductor products or to check the availability of the 73S12xxF, contact us at:

6440 Oak Canyon Road
Suite 100
Irvine, CA 92618-5201

Telephone: (714) 508-8800
FAX: (714) 508-8878
Email: scr.support@teridian.com

For a complete list of worldwide sales offices, go to http://www.teridian.com.

7. Revision History

Revision	Date	Description
1.0	4/27/2009	First publication.