TERIDIAN SEMICONDUCTOR CORP.

Simplifying System Integration™

73S1215F, 73S1217F
Device Firmware Upgrade
Host Driver/Application Development
User's Guide

April 27, 2009
Rev. 1.00
UG_12xxF_029

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

Figures

1 Firmware Upgrade Architecture

This document describes the Teridian USB (DFU class) Interface. Figure 1 shows the flow of this architecture.

The firmware architecture in this document is a single-threaded C program using the API services as detailed in the *73S12xxF Smart Card Terminal Controller Family Software User's Guide*. The software is compiled and linked using the Keil µVision3 development software. The functional areas of the firmware and driver are divided into modules as shown below.

The Windows XP Kernel mode DFU driver was developed using Microsoft Device Driver Development Kit (DDK) Version (3790.1830) Windows Server 2003.

The User Interface Application used for testing both the CCID and DFU driver was developed using Microsoft Visual Studio 2008 C# programming language. The USB-DFU driver and USB-CCID have the file names TSCDFU.sys and CCIDDFUTSC.sys respectively.

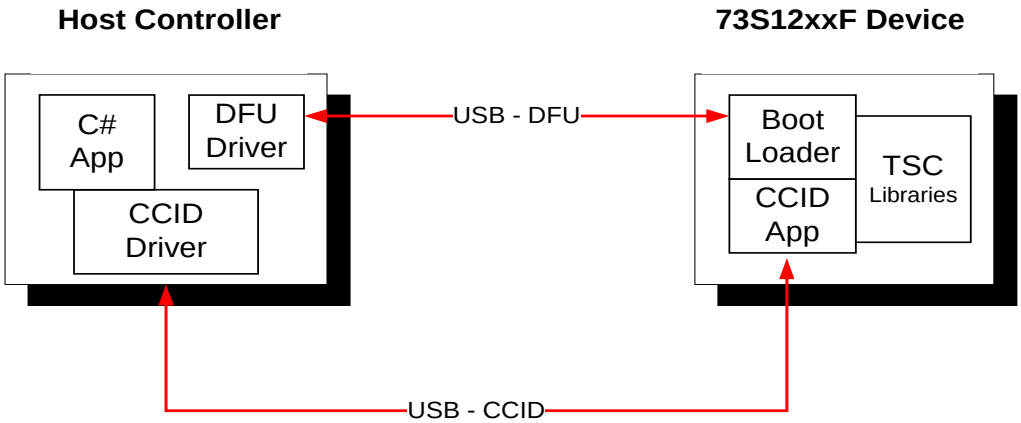

Figure 1: DFU and CCID Class Interface Architecture

The 73S12xxF Device side Boot Loader and CCID Application (or any application) are tightly coupled and co-exist together but only one module will be executed at any time. Which module is invoked is determined by a common 'Check Code'. A 'Check Code' is a two-byte signature with the predefined unique value of **0x875A** that resides at two places within Flash program space: the lowest two bytes of the Application Code section (0x1800 and 0x1801) and the highest two bytes of the Application Code section (0xFFFC and 0xFFFD). At Power On, the Boot code begins with a verification of this Check Codes. If the Check Code in the Boot Code section (0x1800) matches the Check Code in the Application Code section (0xFFFC), it will then begin to execute the Application Code. If the Check Codes do not have the same unique value, it will wait for a USB Reset to start the DFU Class Device enumeration and then begin the Firmware Upgrade programming.

2 DFU – Phases of Operation

The *USB Device Firmware Upgrade Specification Revision 1.1* identified four distinctive phases of operation:

1. Enumeration Phase
2. Reconfiguration Phase
3. Transfer Phase
4. Manifestation Phase

This user guide describes the IOCTLs as supported by Teridian's CCID-DFU driver and DFU driver.

2.1 Enumeration Phase

The specific driver being installed is dependent on the Firmware's Device Class type provided as part of the enumeration. If the firmware's DFU Boot Loader code is running, the DFU driver will be installed. If the firmware's CCID code is running, the CCID driver will be installed. Refer to the *73S1215F, 73S1217F Boot Loader – DFU Class Firmware Application Note* for details of this phase of operation. No DFU's IOCTL are exposed by the CCID driver during this phase.

2.2 Reconfiguration Phase – Detach CCID Class/Attach DFU Class

There are two distinctive steps to transition and complete a successful reconfiguration phase: Detaching the CCID Class Device and Attaching the DFU Class Device. Section 2.2.1 describes the methods of detaching the CCID Class Device. Review the *73S1215F, 73S1217F Boot Loader – DFU Class Firmware Application Note* for firmware's behavior when these commands are processed.

2.2.1 Detaching the CCID Class Device

Once the USB-CCID driver (ccidDFUtsc.sys) is loaded successfully, at anytime the host may start detaching the USB-CCID driver by one of two ways: USB-CCID's Escape Command or DFU_Detach Request. After successful execution of either of these two commands, the CCID driver will be detached and loading of the DFU driver will begin.

2.2.1.1 Escape Command

```
IOCTL : #define   IOCTL_DFU_ESCAPE                    SCARD_CTL_CODE(3500)
Supported driver : ccidDFUtsc.sys, tscdfu.sys
IN    : input (request) buffer length (size 5)
IN    : input (request) buffer ({0x41,0x00,0x00,0x00,0x00})
RETURN: NTSTATUS
Supported driver : ccidDFUtsc.sys
```

When the driver receives this IOCTL it will send an Escape command to the device. Review Section 5.2.4 of the *73S1215F, 73S1217 CCID Application Note* for more details of this function.

2.2.1.2 DFU Detach Request

```
IOCTL : #define   IOCTL_DFUTSC_DFU_DETACH             SCARD_CTL_CODE(2100)
IN    : not required
OUT   : not required
RETURN: NTSTATUS
Supported driver : ccidDFUtsc.sys, tscdfu.sys
```

When the driver receives this IOCTL, it sends a DFU_DETACH Class Specific Request to the device. Review the *73S1215F, 73S1217F Boot Loader – DFU Class Firmware Application Note* for firmware's behavior when, and after, this command is executed.

2.2.2 Attaching the DFU Class Device

Upon receiving escape command or dfu_detach, firmware boots up in DFU mode. This time USB-DFU driver (TSCDFU.sys) will be loaded by operating system (if driver is already installed, otherwise a new hardware found wizard will pop up by Windows). See the *73S1215F, 73S1217F Windows XP 32 USB CCID and DFU Drivers Installation Guide* on how to install drivers. USB-DFU driver (TSCDFU.sys) will enumerate the firmware for DFU class.

2.3 Transfer Phase – Flash Programming

During the transfer, the Boot Loader will process a tight loop to collect DFU_DNLOAD packets from the host. Upon receiving the first valid Intel Hex record, the entire Application Code section of Flash will be erased. Note: the firmware will need about 2.3 seconds to erase the whole 58K of Flash. It is recommended that the host/driver allow the firmware ample time to complete this task after the first Intel record. Thereafter, from the second record on, each record is written as soon as it is received.

After each downloaded block, the host solicits the device status with the DFU_GETSTATUS request. In order to speed up the firmware programming process, the Boot Loader Code will accept DFU_DNLOAD packets in sequential order without the need to get (and report to) a DFU_GETSTATUS request in between DFU_DNLOAD packets. When the last record of the file (record type = 0x01 indicating an End of File record) is received by the firmware, it will wait for a DFU_DNLOAD packet of zero length before it transitions to the Manifestation Phase.

2.3.1 DFU Download Request

```
IOCTL : #define IOCTL_DFUTSC_DFU_DNLOAD              SCARD_CTL_CODE(2101)
IN    : input (request) buffer length
IN    : input (request) buffer
OUT   : not required
RETURN: NTSTATUS
Supported driver : tscdfu.sys
```

Driver tscdfu.sys exposes this IOCTL_DFUTSC_DFU_DNLOAD to transfer (new) firmware file (Intel record/hex format only) to the device. The Intel hex record has to be parsed by the host application. Host appication should send each record in the "input buffer" with the "input buffer length". Once the driver/tscdfu.sys receives this IOCTL with data, it will transfer the data to the device. Upon completion, the driver returns NTSTATUS to the host application.

2.3.2 DFU Get Status Request

```
IOCTL : IOCTL_DFUTSC_DFU_GETSTATUS        SCARD_CTL_CODE(2103)
IN    : input (request) buffer length  (size 6)
IN    : input (request) buffer (of 6 bytes)
OUT   : output(response) buffer length
OUT   : output(response) buffer
RETURN: NTSTATUS
Supported driver : tscdfu.sys
```

Driver tscdfu.sys exposes this IOCTL_DFUTSC_DFU_GETSTATUS to facilitate synchronization with the device. The driver tscdfu.sys expects 6 bytes of input/output buffer. The firmware responds with status data in 6-byte buffer. The driver fills the information returned by the firmware in the output buffer and sends it back to the host application. Refer to Section 6.1.2 of the *USB Device Class Specification For Device Firmware Upgrade*, Revision 1.1 (DFU Class Specification) for interpretation of the 6-byte status returned by firmware/driver to the host application.

2.3.3 DFU Clear Status Request

```
IOCTL : IOCTL_DFUTSC_DFU_CLRSTATUS        SCARD_CTL_CODE(2104)
IN    : not required
OUT   : not required
RETURN: NTSTATUS
Supported driver : tscdfu.sys
```

Any time the firmware detects an error and reports an error status to the host in the response to a DFU_GETSTATUS request, it enters dfuERROR state. The firmware cannot by itself transition from the dfuERROR state. It is the host application/driver's responsibility to clear an error state by sending a DFU_CLRSTATUS request.

The host application shall send IOCTL_DFUTSC_DFU_CLRSTATUS to the driver tscdfu.sys. The driver, in turn, sends the DFU_CLRSTATUS command to the firmware. Upon receipt of DFU_CLRSTATUS, the firmware sets its status to OK and transitions to the dfuIDLE state.

2.3.4 DFU Get State Request

```
IOCTL : IOCTL_DFUTSC_DFU_GETSTATE         SCARD_CTL_CODE(2105)
IN    : input (request) buffer length  (size 1)
IN    : input (request) buffer (of 1 byte)
OUT   : output(response) buffer length
OUT   : output(response) buffer
RETURN: NTSTATUS
Supported driver : tscdfu.sys
```

This request solicits a report about current state of the firmware. The state reported is the current state of the firmware with no change in state upon transmission of the response. The values specified in the output buffer field are identical to those reported in DFU_GETSTATUS

2.3.5 DFU Abort Request

```
IOCTL : IOCTL_DFUTSC_DFU_ABORT            SCARD_CTL_CODE(2106)
IN    : not required
OUT   : not required
RETURN: NTSTATUS
Supported driver : tscdfu.sys
```

The DFU_ABORT request enables the host to exit from certain states and return to the DFU_IDLE state. The firmware sets the OK status on receipt of this request. See the *73S1215F, 73S1217F Boot Loader – DFU Class Firmware Application Note* for more information on the firmware's behavior.

2.4 Manifestation Phase – Device Restart

Read the *73S1215F, 73S1217F Boot Loader – DFU Class Firmware Application Note* for details of this phase's operations. There is no IOCTL for this phase.

3 Miscellaneous Features

3.1 Boot Loader and CCID Application Firmware Version Number

```
#define IOCTL_DFU_GET_BLFW_VER                  SCARD_CTL_CODE(3501)
IN    : not required
OUT   : output buffer length (size 4 bytes)
OUT   : output buffer (of 4 bytes)
RETURN: NTSTATUS
Supported driver : ccidDFUtsc.sys, tscdfu.sys
```

The USB-CCID driver ccidDFUtsc.sys provides one IO control call IOCTL_DFU_GET_BLFW_VER to the upper layer application program. When the driver receives this IOCTL, it sends a request to the boot loader firmware via Standard Device Request to Get String Descriptor from the firmware.

The firmware responds with 4-byte string: [Boot Loader Major Number], [Boot Loader Minor Number], [The last CCID Application Major Number], [The last CCID Application Minor Number].

4 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

5 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.

Revision History

Revision	Date	Description
1.00	4/27/2009	First publication.