User Manual

IDENTIVE Technologies AG

Multi-ISO HF Reader – USB

Contact:
www.identive-technologies.com
www.identive-group.com
support@identive-technologies.com

Document History

Version	Date	Description	Compatibility
1.0	Apr 1, 2009	First Draft	Firmware version above 1.04 Driver version 1.04 and above
1.1	Apr 13, 2009	First Release	Firmware version above 1.04 Driver version 1.04 and above
1.2	Apr 15, 2009	Sample APDU section removed	Firmware version above 1.04 Driver version 1.04 and above
1.3	Apr 17, 2009	Sample APDU section added	Firmware version above 1.04 Driver version 1.04 and above
1.4	May 26, 2009	Driver installation procedure is modified to reflect for signed driver	Firmware version above 1.04 Driver version 1.04 and above
1.5	June 8, 2009	DESFire EV1 commands are explained, modified ChangeKey command of DESFire as per spec.	Firmware version 1.06 and above Driver version 1.04 and above
1.6	June 20, 2009	OS support added in Section 1	Firmware version 1.06 and above Driver version 1.04 and above
1.7	Aug 04, 2009	- MIFARE Restore command description added - Change Key command description for DESFire cards added	Firmware version 1.06 and above Driver version 1.04 and above
1.8	Aug 12, 2009	- LoadKey command described for AES Keys - Reader LoadKeys added in the APDU examples - FCC Notice added	Firmware version 1.06 and above Driver version 1.04 and above
1.9	Sep 01, 2009	- “Firmware Upgrade” term is followed - Table numbers corrected - Default Reader PIN is specified - DESFire command examples corrected - Document Header modified - Value in the command description of MIFARE value blocks have been modified under section 7	Firmware version 1.06 and above Driver version 1.04 and above
1.10	19 Dec, 2009	- Added MIFARE ULC commands description and APDU samples - Added My-d Move commands description and APDU samples - Modified MIFARE UL read response length from 4Bytes to 16Bytes - MIFARE Plus commands added	Firmware version 1.11 and above Driver version 1.04 and above
1.11	Feb 23, 2010	Added DESFire EV1 3KTDES samples.	Firmware version 1.11 and above Driver version 1.04 and above
1.12	Jul 22, 2010	Terms and Abbreviations has been updated	Firmware version 1.11 and above Driver version 1.04 and above
1.13	Feb 02, 2011	- Firmware Configuration details have been added - EasyDESFire and Transparent mode details have been added	Firmware version 1.15 and above Driver version 1.04 and above
1.14	Jun 13, 2011	- Control TOM mode command has been added	Firmware version 1.17 and above Driver version 1.04 and above
1.15	Sep 28, 2011	- LED behaviors added	Firmware version 1.18 and above Driver version 1.04 and above
1.16	Oct 26, 2011	- Chapter 9 describing the details on CCID compliant firmware has been added - Details on accessing Topaz tags has been added (Sec: 7.6 & 8.8)	Firmware version 1.19 and above Driver version 2.06 and above or CCID Driver provided by OS
1.17	May 10, 2012	- Insignificant (Logo & headers modified)	Firmware version 1.19 and above Driver version 2.06 and above or CCID Driver provided by OS
1.18	Jun 26, 2012	- Toggle card function command added	Firmware version 1.20 and above Driver version 2.06 and above or CCID Driver provided by OS
1.19	Dec 11, 2012	- Added connector pin description	Firmware version 1.20 and above Driver version 2.06 and above or CCID Driver provided by OS

1.20	Mar 06, 2013	- Updated onboard connector part numbers - Added part numbers for cable assembly - Added part number for pre-assembled cable - Added section “HID & CCID compliant firmware” - Added “HID” to abbreviations - Updated Sec 1 regarding supported OS - Added Table and Figure numbers - Removed update description in sec. 9 - Added firmware flavor information - Reorganized sec. 9 & 10 - Updated Firmware Configuration Data Structure - Updated “Getting Started” (section 1)	Firmware version 1.20 and above Driver version 2.06 and above or CCID Driver provided by OS
1.21	Aug 14, 2013	- Added info about embedding update function into customer applications (section 9.2)	Firmware version 1.20 and above Driver version 2.06 and above or CCID Driver provided by OS

FCC NOTICE

Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communication. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:
--Reorient or relocate the receiving antenna
--Increase the separation between the equipment and receiver
--Connect the equipment into an outlet on a circuit different from that to which the receiver is connected
--Consult the dealer or an experienced radio/TV technician for help

THIS DEVICE COMPLIES WITH PART 15 OF FCC RULES. OPERATION IS SUBJECT TO THE FOLLOWING TWO CONDITIONS:

(1) THIS DEVICE MAY NOT CAUSE HARMFUL INTERFERENCE AND
(2) THIS DEVICE MUST ACCEPT ANY INTERFERENCE RECEIVED, INCLUDING INTERFERENCE THAT MAY CAUSE UNDESIRED OPERATION

WARNING: *CHANGES OR MODIFICATIONS NOT EXPRESSLY APPROVED BY THE PARTY RESPONSIBLE FOR COMPLIANCE COULD VOID THE USER'S AUTHORITY TO OPERATE THE EQUIPMENT.*

CONTENTS

1 Getting Started

The Multi-ISO HF USB Reader/Writer is a contactless smart card/tag reader and writer for accessing ISO14443-4 Type A, ISO14443-4 Type B, MIFARE (Classic, Ultralight C, DESFire, DESFire EV1, Plus), my-d move, NFC (Type 1, 2, 4) tags, ISO15693, and ICODE SLI tags. This document is intended for application developers who want to access contactless cards using the Multi-ISO HF USB Reader/Writer.

Following sections explain how to install the proprietary drivers for the Multi-ISO HF USB Reader/Writer in Windows operating system (illustrations are taken from Windows XP, same being applicable for other Windows versions). The proprietary Multi-ISO HF USB Reader/Writer drivers are applicable for 32/64-bit Windows XP/2003/Vista/2008/7.

1.1 Connector

The onboard connector for host communication is a male 8-pin connector (Molex 53261-0871). For cable assembly you need the connector housing (Molex 51021-0800) and the crimp terminals (Molex 50079-8000). You can also purchase black pre-assembled USB 2.0 cables from Identive (PN: KAB_USB1) with Molex connector.

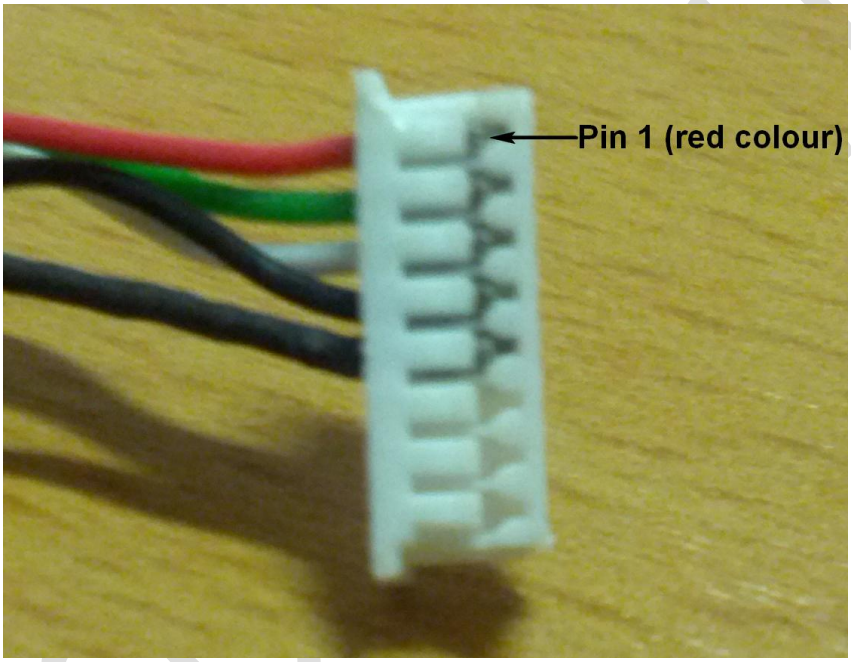

Figure 1 – Molex 8-pin connector of a pre-assembled cable

Connector Pin	*Cable Wire*	*Signal Description*
1	1	VBUS
2	2	D+
3	3	D+
4	4	Ground
5	5	Shield (Grounded)
6	-	Not connected
7	-	Not connected
8	-	Not connected

Table 1 – Pin Layout

Note: For users who make their own cable, it is recommended that the shield (Pin-5) of the cable shall also be connected to GROUND for the device to operate normally.

1.2 Driver Installation

Drivers are necessary to access Multi-ISO HF USB Reader/Writer. The following steps illustrate the installation procedure.

1. Plug in the reader in USB port

2. "**Found New Hardware Wizard**" will appear. Select "**No, not this time**" and click "**Next**" to continue driver installation.

Figure 2 – "Found New Hardware Wizard" showing welcome window

3. In the next appearing dialog box, select "**Install from a list or specific location (Advanced)**" and click on "**Next**"

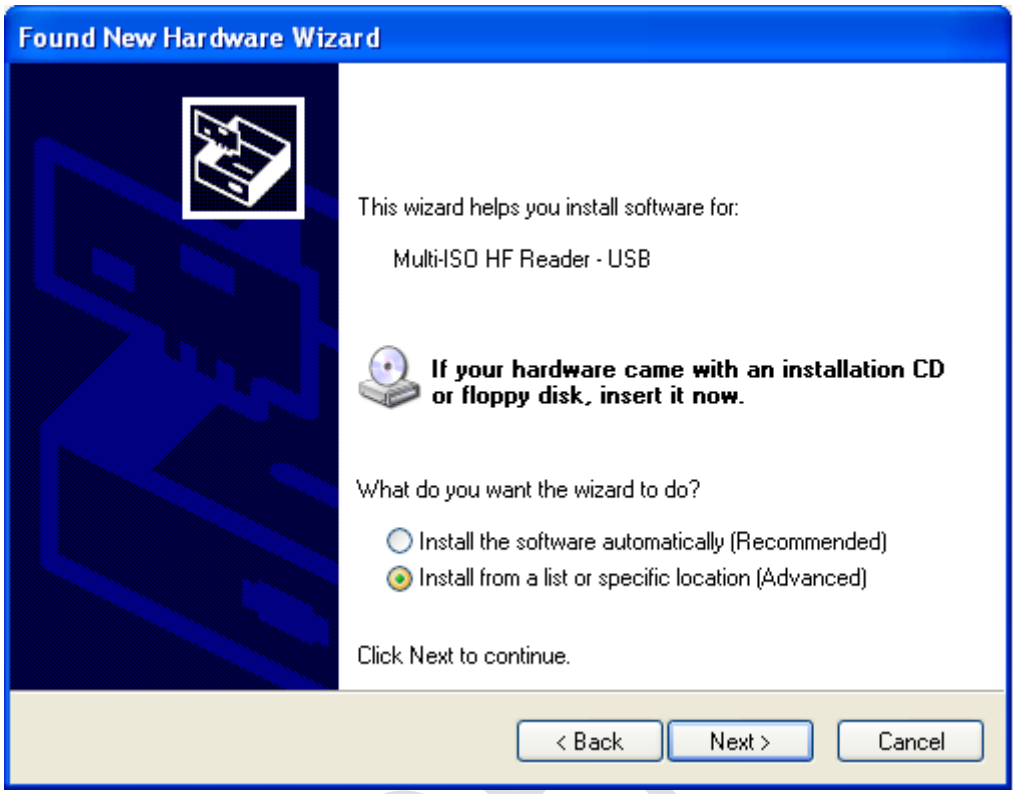

Figure 3 – "Found New Hardware Wizard" showing install options

4. In next dialog box select "**Search for the best driver in these locations**" and "**Search removable media (floppy, CD-ROM…)**" sub option. Insert the installation CD that is given along with the reader kit and click "**Next**".

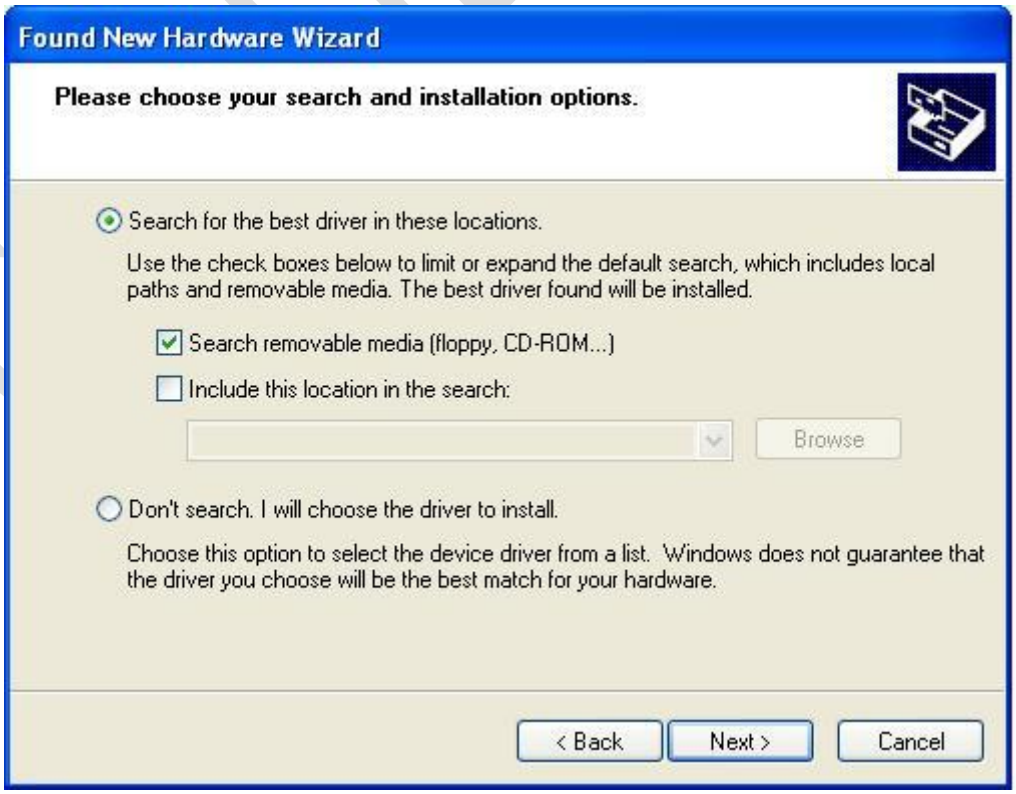

Figure 4 – "Found New Hardware Wizard" showing search options

5. Wait until the driver installation is completed by the operating system and the installation success dialog is displayed as shown below. Click "**Finish**". Now the drivers are installed successfully.

Figure 5 – "Found New Hardware Wizard" showing the final window

1.3 Trouble Shooting Driver Installation

Device will not function properly if wrong driver is installed or if there is a version incompatibility between the firmware and driver. In these cases, the installed driver must be uninstalled and proper driver must be installed as explained below

Un-installation procedure is as follows.

1. Open the device manager -> Smart card readers and select "**Multi-ISO HF Reader – USB"**
2. Right click on "**Multi-ISO HF Reader – USB**" and select "**Uninstall**"

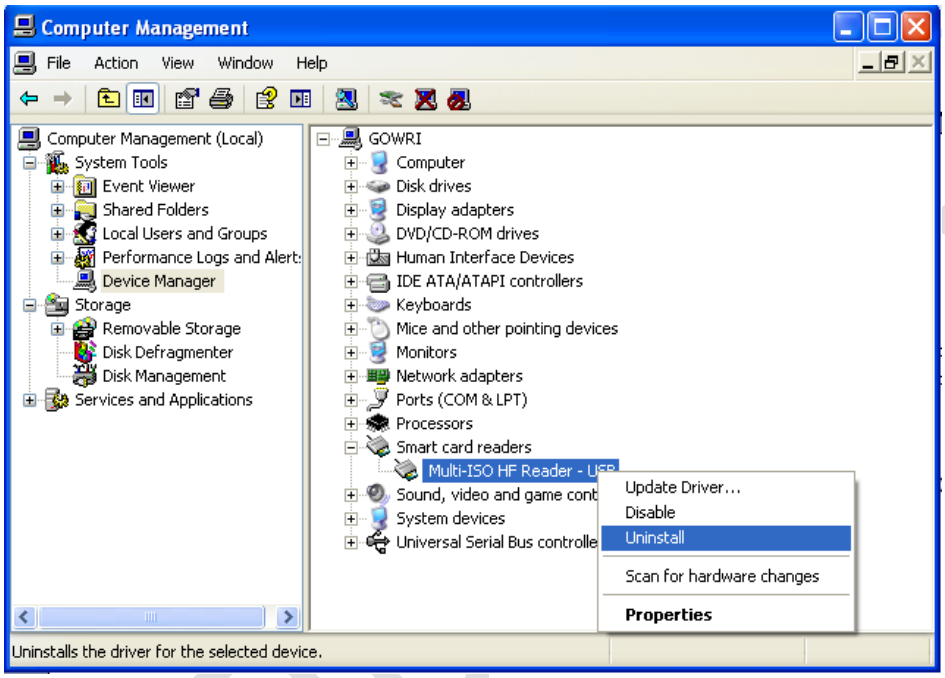

Figure 6 – Device Manager showing "Multi-ISO HF Reader – USB"

3. In the following dialog click "OK".

Figure 7 – Device Manager showing "Confirm Device Removal" dialogue

4. The device will now disappear from the device manager
5. Un-plug and re-plug the device
6. System will prompt for drivers. Install the drivers as described in the driver installation procedure section Driver Installation.

2 Diagnostics

Version of the driver and firmware plays an important role in the proper working of the device. Device may malfunction if incompatible drivers are installed for particular version of firmware. The compatibility of the driver and firmware can be found in the "Document history" section of this document.

2.1 Driver Version Detection

Driver version detection is described in the following procedure.

1. Right click on the "**My Computer**" icon and click on "**Manage**"

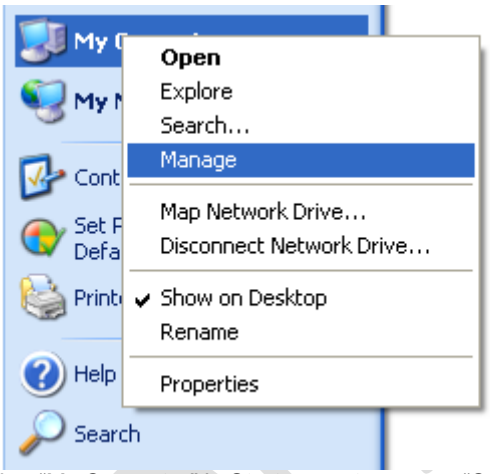

Figure 8 – Right clicking "My Computer" in Start menu to access "Computer Management"

2. In following dialog, select "**Device Manager**" under "**System Tools**" menu in the left pane.

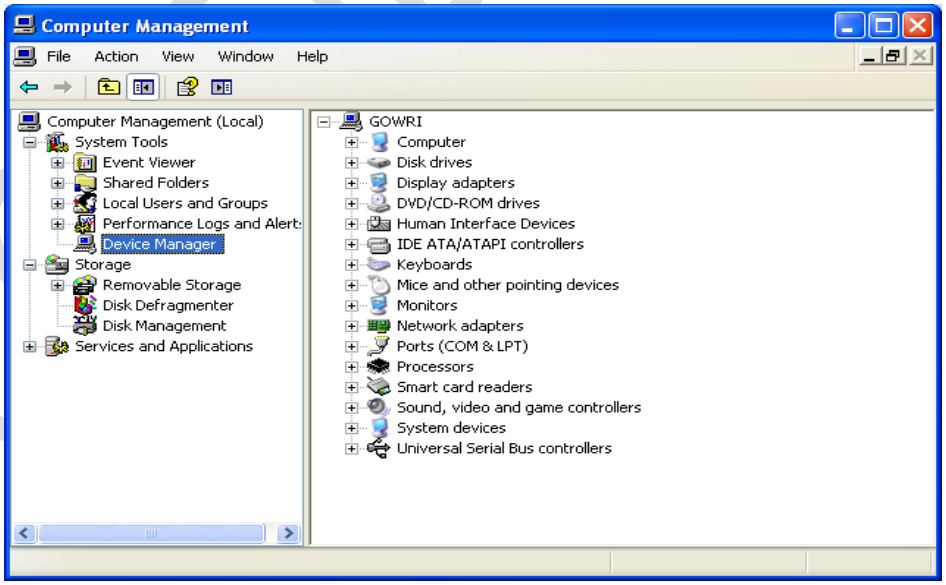

Figure 9 – Selecting the Device Manger in the Computer Management window

3. Double click on "**Smart card readers**" in the right pane. Right click on "**Multi-ISO HF Reader - USB**", and select "**Properties**".

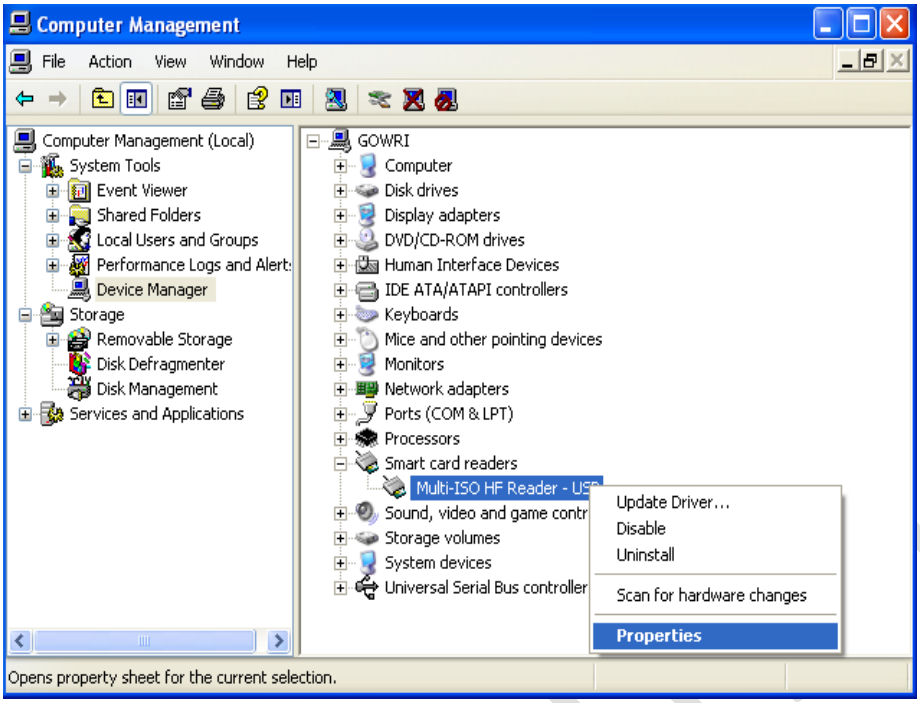

Figure 10 – Right clicking the "Multi-ISO HF Reader – USB" to access its properties

4. In the dialog box which appears, select the "Driver" tab, driver version can be found in the tab window. For example, the driver version will appear as

   "**Driver Version: 1.0.2.0**" for driver version 1.02

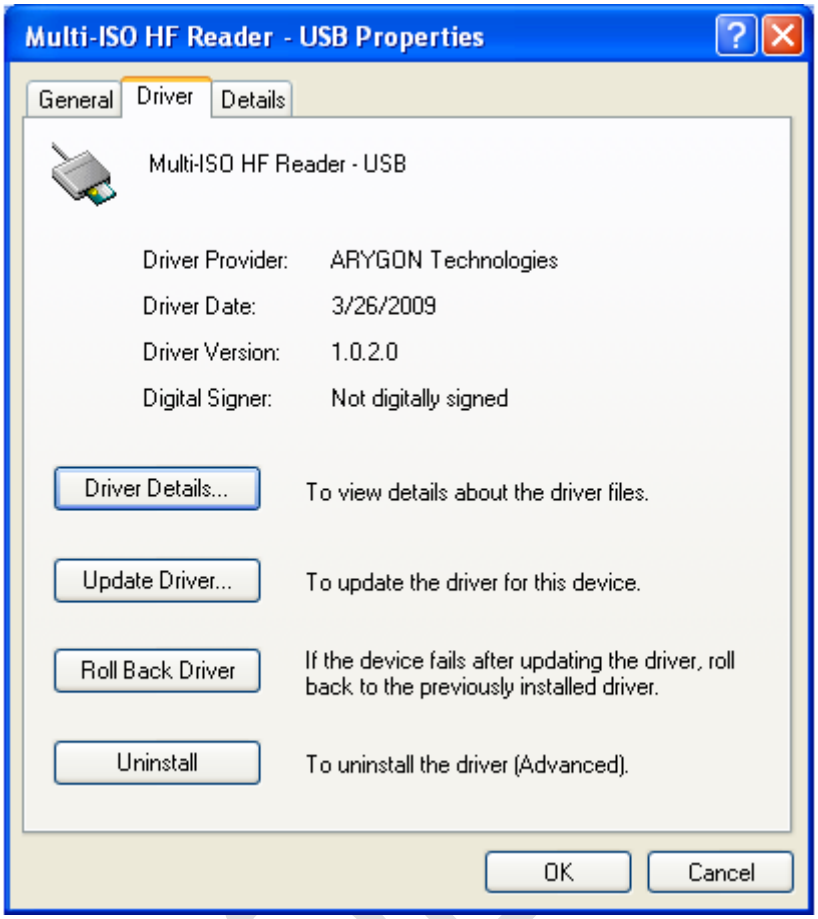

Figure 11- Driver tab showing the driver version

2.2 Firmware Version Detection

To detect the current version of the firmware in the device, follow the same procedures as in the Driver version detection up to step 3.

1. In the dialog box which appears, select "**Details**" tab. And in the drop down list box select "**Firmware Revision**" as shown below.

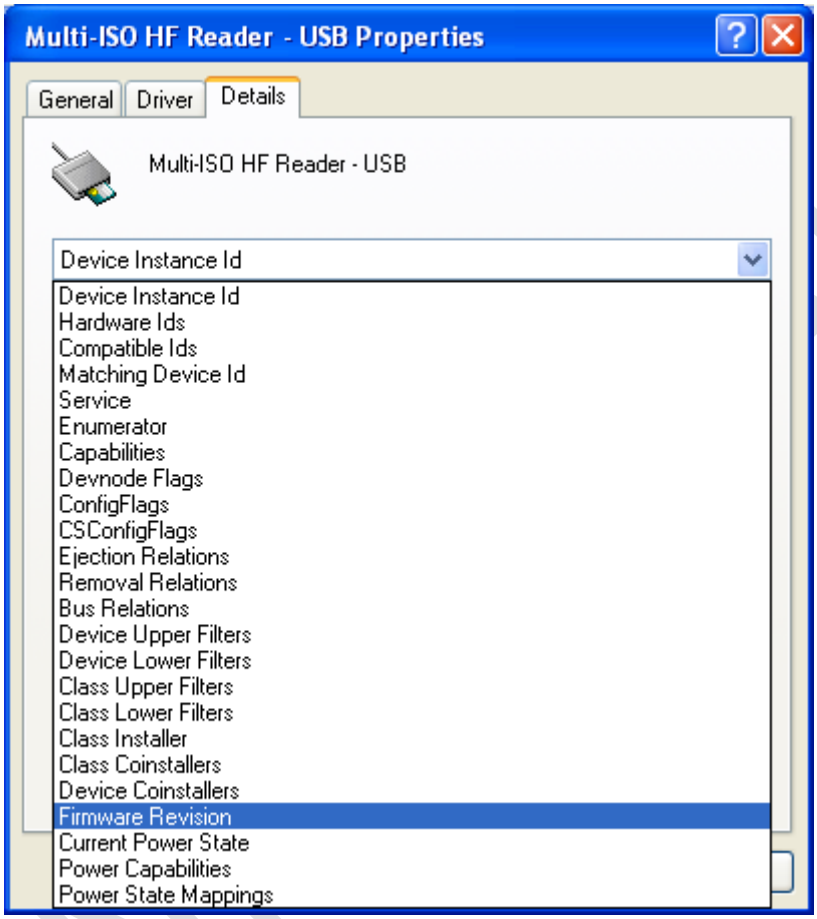

Figure 12 – Selecting the "Firmware Revision" under Details tab

2. In the dialog box, the version of the firmware currently in the device will be displayed as shown below

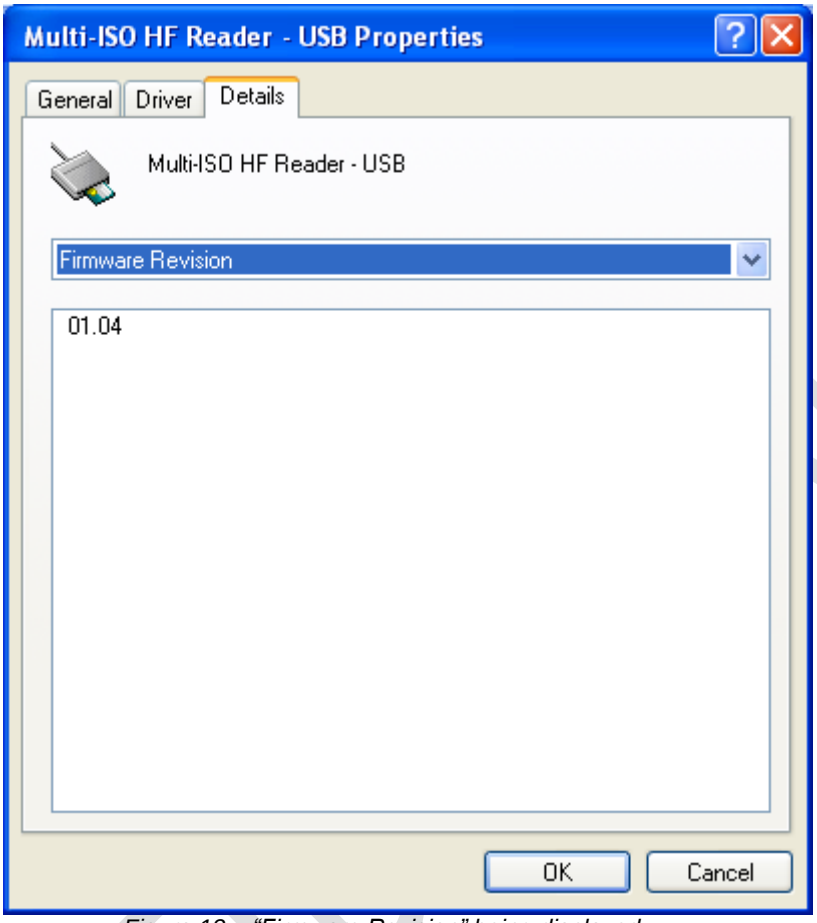

Figure 13 – "Firmware Revision" being displayed

3 LED blinking behavior

The following tables describe the LED behavior of the reader during possible reader states.

Firmware version ≤ 1.17

Reader States	*Green LED*	*Red LED*	*Blink-rate (s)*
DFU F/W is active	OFF	ON	-
Functional F/W active & Unconfigured	ON	OFF	-
Functional F/W active & Configured			
Waiting for card to be detected			
Card detected			
Host communicating with the card	ON	ON	Red LED will be ON until the reader responds to the host command
Error in card detection*	ON	OFF	-
Functional F/W enters suspend mode	OFF	OFF	-

Table 2 – LED behavior for firmware version ≤1.17

Firmware version ≥ 1.18

Reader States	*Green LED*	*Red LED*	*Blink-rate (s)*
DFU F/W is active	OFF	ON	-
Functional F/W active & Unconfigured	ON	OFF	-
Functional F/W active & Configured	OFF	ON	-
Waiting for card to be detected	OFF	ON	-
Card detected	Blinking	OFF	0.25
Host communicating with the card	ON	Blinking	0.05
Error in card detection*	ON	ON	-
Functional F/W enters suspend mode	OFF	OFF	-

Table 3 – LED behavior for firmware version ≥1.18

Note:
* For contactless cards, the card has not fully entered the reader's RF field or the card is not responding.

4 Card Reader Suite – Test Application

CardRdrSuite is a test application that is provided along with Multi-ISO HF USB Reader/Writer for customer use. CardRdrSuite consists of several sub-applications covering the following functions

4.1 Firmware Upgrade

Reader's firmware can be upgraded using this sub-application. Salient features are

- Auto reader detection
- User friendly GUI
- Provision to view device's current firmware version and the version of the firmware to be downloaded

The application operates in two modes

- DFU class mode

- Disaster recovery mode to recover reader if the reader is plugged out accidentally during firmware upgrade
- Individual DFU driver to handle this mode

- Proprietary mode
  - Firmware upgrade support without using the DFU driver

Refer to section “Firmware Upgrade” of Card Reader Suite help file for detailed description. Help file can be launched using the “Help” button in the main window of the Card Reader Suite application.

4.2 PC/SC Diagnostics

Any Card APDU or Pseudo APDU can be issued and tested using this sub-application.

Salient features are

- APDU execution from script file (.APDU)
- Easy to edit script files
- Error status logging in a file
- Working sample scripts for each card type supported

Refer to section “PC/SC Diagnostics” of the Card Reader Suite help file for detailed description

4.3 Binary configuration

All configurable parameters of the reader can be modified using these sub-applications. These applications will configure the firmware binary file only.

Configure Binary

- Configures
  - USB Chapter 09 parameters
  - Firmware Version parameters
  - Hardware Version parameters
  - Other firmware configurations
- Provides option to make the parameters non-modifiable after configuring

Refer to section “Configure binary” of Card Reader Suite help file for detailed description

Edit Binary

- Configures
  - RF parameters
  - MIFARE keys stored in the non-volatile memory of the reader
  - DESFire keys stored in the non-volatile memory of the reader
- Provides option to make the parameters non-modifiable after configuring

Refer to section “Edit Binary” of Card Reader Suite help file for detailed description

5 PC/SC 2.0

"Multi-ISO HF USB Reader/Writer" can be accessed using the standard PC/SC architecture for communicating with cards. This makes card access easy, as it uses the same communication procedure for all the cards.

The Microsoft Developer Network (MSDN) library gives valuable information and a detailed description of all SCard APIs for communicating with the reader, through the Windows PC/SC framework (WINSCARD APIs – Refer [R4]).

5.1 How to Access Contactless Cards via PC/SC?

Contactless cards can be easily accessed through PC/SC using standard WINSCARD smart card API's for card access. The following steps provide guidelines for using the PC/SC compliant WINSCARD API's to access cards using the Multi-ISO HF USB Reader/Writer.

5.1.1 Establish Context

This is the first step. This API will initialize all other PC/SC APIs and allocate all resources necessary for a smart card session. The SCardEstablishContext function establishes the resource manager context (the scope) within which database operations is performed.

```
LONG SCardEstablishContext ( IN DWORD dwScope,
                             IN LPCVOID pvReserved1,
                             IN LPCVOID pvReserved2,
                             OUT LPSCARDCONTEXT phContext);
```

5.1.2 List Readers

The next step is to get a list of all PC/SC readers connected to the system using the SCardListReaders function. Look for "Multi-ISO HF Reader – USB - 0000000000000001 0", in the list returned. If multiple Multi-ISO HF USB Reader/Writers are connected to your system, they will be enumerated with different serial numbers.

Example: "Arygon Multi-ISO 0000000000000001 0", "Arygon Multi-ISO 0000000000000002 0", could be the list returned when the following function is executed.

```
LONG SCardListReaders (IN SCARDCONTEXT *phContext,
                       IN LPCTSTR mszGroups,
                       OUT LPTSTR mszReaders,
                       IN OUT LPDWORD pcchReaders);
```

Note: The OUT parameter phContext of SCardEstablishContext is used as in parameter for this API.

5.1.3 Connect

Next step is to connect to the card via the reader/writer. The SCardConnect API establishes a connection (using a specific resource manager context) between the calling application and a smart card contained by the specific reader. If no card exists in the specified reader, an error is returned.

```
LONG SCardConnect (  IN SCARDCONTEXT *phContext,
                     IN LPCTSTR szReader,
                     IN DWORD dwShareMode,
                     IN DWORD dwPreferredProtocols,
                     OUT LPSCARDHANDLE phCard,
                     OUT LPDWORD pdwActiveProtocol);
```

Note: The OUT parameter phContext of SCardEstablishContext is used as in parameter for this API.

5.1.4 Data and Command transfer with Card

Command and data that are transferred to the card are called as APDUs (application protocol data unit) in smart card terminology. The SCardTransmit function sends a service request to the smart card, and expects data back from the card.

```
LONG SCardTransmit ( IN SCARDHANDLE *phCard,
                     IN LPC/SCARD_I0_REQUEST pioSendPci,
                     IN LPCBYTE pbSendBuffer,
                     IN DWORD cbSendLength,
                     IN OUT LPSCARD_IO_REQUEST pioRecvPci,
                     OUT LPBYTE pbRecvBuffer,
                     IN OUT LPDWORD pcbRecvLength);
```

Note: The OUT parameter phCard of SCardConnect is used as in parameter for this API.

5.1.5 Disconnect

It is not mandatory to disconnect the card after the completion of all transactions, but it is recommended. The SCardDisconnect function terminates a connection previously opened between the calling application and a smart card in the target reader.

```
LONG SCardDisconnect (IN SCARDHANDLE *phCard,
                      IN DWORD dwDisposition);
```

Note: The OUT parameter phCard of SCardConnect is used as in parameter for this API.

5.1.6 Release

This step ensures that all system resources are being released. The SCardReleaseContext function closes an established resource manager context, freeing any resources allocated under that context.

```
LONG SCardReleaseContext (IN SCARDCONTEXT *phContext);
```

Note: The OUT parameter phContext of SCardEstablishContext is used as in parameter for this API.

5.2 ATR Generation

To make contactless cards available within the PC/SC framework, the Multi-ISO HF USB Reader/Writer generates a PC/SC compliant ATR according to PC/SC v2.01.09 “Interoperability Specification for ICCs and Personal Computer Systems” (reference [R2])

5.2.1 CPU Cards

The ATR of Contactless processor cards are composed as described in PC/SC v2.01.09, Part3: Requirements for PC connected Interface Devices, 3.1.3.2.3.1, Table 3.5

5.2.2 Storage Cards

The ATR of storage cards (i.e. cards without CPU) are composed as described in PC/SC v2.01.09, Part3: Requirements for PC connected Interface Devices, 3.1.3.2.3.2, Table 3.6. In order to allow the HOST application to identify a storage card type properly, its standard and card name is mapped according to the Part3: Supplement Document of PC/SC v2.01.04

Note: The registered Application Provider Identifier (RID) returned by the Multi-ISO HF USB Reader/Writer for storage cards (cards without CPU) is “A0 00 00 03 06”, which is the RID of the PC/SC workgroup

6 Accessing Reader through PC/SC

In some specific situations, PC/SC specifications are not enough to cover the whole functional field. This happens typically

- When working with memory cards or even microprocessor based cards not following the ISO 7816-4 standard (APDU formalism)
- When needing to perform actions on to the reader/writer itself, and not onto the card (Like modifying the RF parameters of the reader/writer)

In order to cover the above two cases, proprietary 7816 wrappers are supported. These are called Pseudo APDUs. Refer to [R7] in order to understand the basic structure of APDU

Reader commands are covered under this section. While Pseudo APDU for accessing ISO 7816-4 non-compliant cards are covered in section Accessing Cards through PC/SC.

MIFARE cards and ISO 15693 cards use proprietary 7816 APDU structures. DESFire cards use the 7816 wrapper as described in the DESFire specifications [R3] & [R11]

All command and response bytes are sent and received as hexadecimal values respectively

Note: In the pseudo APDUs described in this section, specifying a value of 00 in the **Le** field indicates maximum no of available response bytes from the card, as described in reference [R7]

6.1 Reader Control Commands

The Reader/Writer control commands are used to modify the reader/writer parameters in order to fine tune the reader/writer performance or to suit the end applications requirements. It follows standard ISO 7816-4 (as per reference [R7]) command/ response format

The Reader/Writer control commands have the following general APDU format

Command Format:

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*		*Le*
FF	00	00	00	No of bytes in Data field	Reader/Writer Control Command	Command Parameters	00

Response Format:

Data	*Status Word*	
Reader/Writer Control Response	SW1	SW2

For possible values and description of status word, refer Table 13. The following are the Reader/Writer control commands.

6.1.1 Get Static RF Parameters

Get Static RF Parameters command is used to get the RF parameters from the non-volatile area of the reader/writer

Command Data:

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*			*Le*
FF	00	00	00	03	01	00	03	00

Response Data:

Data	*Status*
RF Parameters as in the Structure shown in Table 4 (128 bytes)	9000

Note:

- The command fails if the reader/writer configuration is invalid
- The command fails if any of the command byte is invalid

6.1.2 Set Static RF Parameters

Set Static RF Parameters command is used to modify the RF parameters in the non-volatile area of the reader/writer. The reader/writer uses these parameters from the next power ON

Command Data:

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*				*Le*
FF	00	00	00	83	02	00	03	RF Parameters as in the Structure shown in Table 4 (128 bytes)	-

Response Data:

Data	*Status Word*
-	9000

Note:

- In the above command data, modifying the parameter value 00,03 might cause the reader/writer to malfunction
- The command fails if the reader/writer configuration is invalid
- The command fails if any of the command byte is invalid
- The command fails if the length mentioned in the RF parameters structure is not 0x0080
- Modifying the Flag value to anything other than 0x01, might make the reader/writer un-usable

6.1.3 Get Dynamic RF Parameters

Get Dynamic RF Parameters command is used to get the RF parameters from the volatile area of the reader/writer (current in use parameters)

Command Data:

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*		*Le*
FF	00	00	00	01	03	-	00

Response Data:

Data	Status Word
RF Parameters as in the Structure shown in Table 4 (128 bytes)	9000

6.1.4 Set Dynamic RF Parameters

Set Dynamic RF Parameters command is used to modify the RF parameters in the volatile area of the reader/writer (temporarily for this session). Immediately following this control command, the reader/writer restarts its entire activity on the RF interface with the new parameters

Command Data:

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*		*Le*
FF	00	00	00	81	04	RF Parameters as in the Structure Shown in Table 4 (128 bytes)	-

Response Data:

Data	*Status Word*
-	9000

Note:

- The command fails if the length mentioned in the RF parameters structure is not 0x0080
- The command fails if any of the command byte is invalid

6.1.5 Get Firmware Configuration

Get Firmware Configuration command is used to the retrieve the Firmware Configuration Record from the reader.

Command Data:

CLA	INS	P1	P2	Lc	Data			Le
FF	00	00	00	03	01	1C	00	00

Response Data:

Data	Status
Firmware Configuration Record as shown in Table 7 (32 bytes)	9000

Note:

- The command fails if the reader/writer configuration is invalid
- The command fails if any of the command byte is invalid

6.1.6 Set Firmware Configuration

Set Firmware Configuration command is used to modify/update the Firmware Configuration Record into the reader.

Command Data:

CLA	INS	P1	P2	Lc	Data				Le
FF	00	00	00	23	02	1C	00	Firmware Configuration Record as shown in Table 7 (32 bytes)	-

Response Data:

Data	Status Word
-	9000

Note:

- In the above command data, modifying the parameter value 1C,00 might cause the reader/writer to malfunction
- The command fails if the reader/writer configuration is invalid
- The command fails if any of the command byte is invalid
- Modifying the Flag value to anything other than 0x01, might make the reader/writer un-usable

6.1.7 Control TOM Mode

Control TOM Mode command is used to enable/disable the TOM mode of the reader. By enabling the TOM mode, the reader's RF parameters would be optimally configured to improvise the tag readability for cases where in the reader is placed nearby to a metallic environment. It has to be noted that, this is a dynamic control, and hence would become ineffective, after a power-on cycle.

Command Data:

CLA	INS	P1	P2	Lc	Data		Le
FF	00	00	00	02	14	01 - Enable TOM Mode (or) 00 – Disable TOM Mode	-

Response Data:

Data	*Status Word*
-	9000

Note:

- The command fails if the reader/writer configuration is invalid
- The command fails if any of the command byte is invalid

6.1.8 Toggle card function

This command is used to toggle the functionality of a dual technology card.

C-APDU:

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	00	00	00	01	11	-

R-APDU:

Data	*Status Word*
-	9000

Command Usage:

- Dual Technology cards can function either as a MIFARE Classic card or as a Generic smart card. By default the reader will access the MIFARE function of the card. If at any time, if the host wishes to access the Generic smart card functionality, it has to send this command and perform a card-connect cycle.
- This command can be used any number of times.
- This command does not require the presence of the card in field.

Note:

- The command fails if any of the command byte is invalid.

6.1.9 RF Parameters Data Structure

Parameter	*No of bytes*	*Offset*
Length in bytes	2 [Value = 0x0080]	0
Flag	1 [Value = 0x01]	2
A106 - CWCONDUCTANCE	1	3
A106 – RXTHRESHOLD	1	4
A106 - RXCONTROL1	1	5
A212 - CWCONDUCTANCE	1	6
A212 - RXTHRESHOLD	1	7
A212 - RXCONTROL1	1	8
A424 - CWCONDUCTANCE	1	9
A424 - RXTHRESHOLD	1	10
A424 - RXCONTROL1	1	11
A828 - CWCONDUCTANCE	1	12
A828 - RXTHRESHOLD	1	13
A828 - RXCONTROL1	1	14
B106 - CWCONDUCTANCE	1	15
B106 - RXTHRESHOLD	1	16
B106 - RXCONTROL1	1	17
B106 - MODCONDUCTANCE	1	18
B106 - TYPEBFRAMING	1	19
B212 - CWCONDUCTANCE	1	20
B212 - RXTHRESHOLD	1	21
B212 - RXCONTROL1	1	22
B212 - MODCONDUCTANCE	1	23
B212 - TYPEBFRAMING	1	24
B424 - CWCONDUCTANCE	1	25
B424 - RXTHRESHOLD	1	26
B424 - RXCONTROL1	1	27
B424 - MODCONDUCTANCE	1	28
B424 - TYPEBFRAMING	1	29
B848 - CWCONDUCTANCE	1	30
B848 - RXTHRESHOLD	1	31
B848 - RXCONTROL1	1	32
B848 - MODCONDUCTANCE	1	33
B848 - TYPEBFRAMING	1	34
TESTANASELECT	1	35
TESTDIGISELECT	1	36
Reserved	11	37
RF Reset Width in milliseconds	2	48
Card de-bounce delay in milliseconds	2	50
All Timeout Values Multiplication scale	1	52
All Timeout Values Division scale	1	53
All constant Delay Multiplication scale	1	54
All constant Delay Division scale	1	55
All Loop counts Multiplication scale	1	56
All Loop counts Division scale	1	57
Type-A Max baud limit	1	58
Type-B Max baud limit	1	59
Card Polling scheme	1	60
Reserved	67	61

Table 4 – RF parameters data structure

Naming Convention

In the above table,

- The Parameters starting with 'A' or 'B' refer to the respective ISO 14443 card types
- The number following the alphabet indicates the baud rate at which the card should be operating
- The actual RF parameter follows the '-'. This parameter will take effect for that type of card operating at that baud rate

Example: **-** *A106 – CWCONDUCTANCE indicates the* CWCONDUCTANCE parameter of ISO 14443 TypeA cards operating at 106 Kbps

Parameter Description

RF Control Parameter
"CWCONDUCTANCE" parameter controls the strength of RF field when there is no modulation. Its value can vary from 0x00 to 0x3F. The chosen value would get directly programmed into the (Address 0x12) RFID reader silicon. For more details refer to the respective datasheet of RFID reader silicon (reference [R12]).

"RXTHRESHOLD" parameter controls the receiver input threshold levels. The specified value would get directly programmed into the (Address 0x1C) RFID reader silicon. For more details refer to the respective datasheet of RFID reader silicon (reference [R12]).

"RXCONTROL1" parameter controls the receiver input stage gain levels and the low pass filters. The specified value would get directly programmed into the (Address 0x19) RFID reader silicon. For more details refer to the respective datasheet of RFID reader silicon (reference [R12]).

"MODCONDUCTANCE" parameter controls the strength of RF field when there is 10% modulation for Type-B data transmission. Its value can vary from 0x00 to 0x3F. The chosen value would get directly programmed into the (Address 0x13) RFID reader silicon. For more details refer to the respective datasheet of RFID reader silicon (reference [R12]).

"TYPEBFRAMING" parameter controls the framing headers SOF & EOF of type B transmission frames. The specified value would get directly programmed into the (Address 0x17) RFID reader silicon. For more details refer to the respective datasheet of RFID reader silicon (reference [R12]).

"TESTANASELECT" parameter controls the analog debug output pin AUX. The specified value would get directly programmed into the (Address 0x3A) RFID reader silicon. For more details refer to the respective datasheet of RFID reader silicon (reference [R12]).

"TESTDIGISELECT" parameter controls the digital debug output pin MFOUT. The specified value would get directly programmed into the (Address 0x3D) RFID reader silicon. For more details refer to the respective datasheet of RFID reader silicon (reference [R12]).

RF Reset Width in milliseconds
This parameter defines the width of RF Reset (no RF Power) during the polling sequence. The value entered is in decimal, from 0 to 65535. The RF reset would have a width of this much amount of time in milliseconds.

Card de-bounce delay in milliseconds
This parameter defines the time in milliseconds for which the card arrival is polled and reconfirmed by repeated RNAK polling, before notifying the arrival of a new card into the RF field, to the host. The value entered is in decimal, from 0 to 65535.

All Timeout values Multiplication scale & All Timeout values Division scale
These two parameters are used to scale the Timeout values used by the Wait functions. The scaling is done using the following formula:

$$\text{TIMEOUT} = \frac{\textit{Timeout} * \textit{Timeout Multiplication Scale}}{\textit{Timeout Division Scale}}$$

All constant Delay Multiplication scale & All constant Delay Division scale

These two parameters are used to scale the constant Delays used in the Wait functions. The scaling is done using the following formula:

$$\text{CONSTDELAY} = \frac{\textit{DefaultRetryCount} * \textit{Loop Count Multiplication Scale}}{\textit{Loop Count Division Scale}}$$

All Loop counts Multiplication scale & All Loop counts Division scale

Under ISO 14443 Part 3 & Part 4, on occurrence of any communication errors like CRC, Framing, Parity and Timeout, the command will be re-tried by the reader/writer. The Retry count value is scaled using these two parameters, as per the following formula:

$$\text{RETRYCOUNT} = \frac{\textit{DefaultRetryCount} * \textit{Loop Count Multiplication Scale}}{\textit{Loop Count Division Scale}}$$

Type-A Max baud limit & Type-B Max baud limit

This parameter is used to limit the maximum baud rate at which the reader/writer can operate with the respective ISO 14443 card types

Value	*Maximum Baud Rate Supported*
0x00	106 Kbps
0x01	212 Kbps
0x02	424 Kbps
0x03	848 Kbps

Table 5 – Max Baud Limit

Card Polling Scheme

This parameter enables the user to select the card types he wants the reader/writer to detect. The Card types disabled here will not be detected

b7	*b6*	*b5*	*b4*	*b3*	*b2*	*b1*	*b0*	*Polling Scheme*
0	0	0	0	0	0	0	0	No Polling
X	X	X	X	X	X	X	1	Poll for 14443 TypeA cards
X	X	X	X	X	X	1	X	Poll for 14443 TypeB cards
X	X	X	X	1	X	X	X	Poll for SRIX cards
X	X	X	1	X	X	X	X	Poll for ISO 15693 cards
1	X	X	X	X	X	X	X	Reserved

Table 6 – Card Polling Scheme

Firmware Configuration Data Structure

Parameter	*No of bytes*	*Offset*
Length in bytes	2 [Value = 0x0020]	0
Flag	1 [Value = 0x01]	2
Reserved – 1	1 [Value = 0x01]	3
Reserved – 2	2 [Value = 0x03E8]	4
HID configuration data	22 [Value = 0xFF …]	6
Firmware Configuration Option	4 [Value = 0x00000000]	28

Table 7 – Data Structure of the Firmware Configuration Record

Firmware Configuration Option:
Firmware Configuration is a 32 bit value. Each of its bits enables or disables a feature in the firmware. The following is the bit definition for the same.

Bit	*Feature Description*
0	*Reserved for internal use*
1	*Reserved for internal use*
2	0 – Enables EasyDESFire Mode 1 – Disables EasyDESFire Mode i.e., Enables Transparent Mode
3	0 – Enables Memory Cards 1 – Disables Memory Cards Applicable to ISO14443-3 & ISO15693 cards
4	*Reserved for internal use*
5	*Reserved for internal use*
6	*Reserved for internal use*
7	*Reserved for internal use*
8	*Reserved for internal use*
9	*Reserved for internal use*
10-31	*Reserved for future use*

Table 8 – Bit definition of the Firmware Configuration Option

The default firmware contains a value of 0x00000000.

6.2 Key Management

The Multi-ISO HF USB Reader/Writer provides provision to store card keys in its non-volatile memory. The reader can be customized to store card keys. An authenticated user can later refer to them during card communication using key numbers. This section describes the commands used to achieve this functionality in detail

6.2.1 Reader Authentication

The Reader Authenticate command is used to authenticate with the reader/writer. The PIN specified in the command is verified with the PIN stored in the reader/writer

Only after a successful Reader Authenticate, the user can use the Load Keys command to store card specific keys or modify the Reader PIN in the non-volatile area of the reader/writer. The default Reader PIN is “00 00 00 00 00 00 00 00”.

This command is used to ensure that a malicious user does not gain access to modify the Card keys or Reader PIN stored in the reader/writer.

Command Format:

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*		*Le*
FF	00	00	00	09	09	Reader PIN (8 bytes)	-

Response Format:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

Note: The Authentication state will be cleared immediately after the first Load Keys command following the Reader Authenticate command, irrespective of whether the Load Keys command was successful or not. The user will have to authenticate with the reader/writer before issuing each Load Keys command

6.2.2 Load Keys

The Load Keys command is used to store Card authentication keys and Reader PIN in the non-volatile area of the reader/writer.

The user must use Reader Authenticate command to authenticate with the reader/writer before using this command

The reader/writer has provision to store

- 1 Reader PIN, 80 MIFARE Keys along with Key type
- 8 DESFire keys (1 PICC Master key and 7 Application keys) along with AID, PCD Key number and PICC Key number
- 14 MIFARE Plus AES sector keys and 10 special keys with key block number
- 1 authentication key for MIFARE Ultralight C cards

When a card specific Authenticate APDU is received from the host, the appropriate keys are fetched from the non-volatile memory of the reader and used for authentication

This command does not require the presence of a card over the reader; however it may also do so.

Command Format:

CLA	INS	P1	P2	Lc	Data		Le
FF	00	00	00	No of bytes in Data field	07	Key Data	-

Response Format:

Data	Status Word	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

Note: The user must make sure that he uses the Load Keys command to store the appropriate keys in the reader/writer before trying to issue an Authenticate APDU to the respective card

6.2.2.1 Load Reader Authentication PIN in to Reader

The Following is the Load Keys command format to change the reader PIN. The Reader PIN can be any 8 byte numeric value

Command Data:

CLA	INS	P1	P2	Lc	Data			Le
FF	00	00	00	0A	07	FF	Reader PIN (8 bytes)	-

Example: Change Reader PIN in the reader

Command: FF 00 00 00 0A 07 FF 01 02 03 04 05 06 07 08
Response: 90 00

6.2.2.2 Load MIFARE Authentication Keys in to Reader

The following is the Load Keys command format to load the MIFARE authentication keys into the reader

Command Data:

CLA	INS	P1	P2	Lc	Data			Le
FF	00	00	00	0A	07	00	MIFARE Key data (as shown below)	-

MIFARE key Data		
Key Number (1 Byte)	Key Type (1 Byte)	Key (6 Bytes)

Where,

Key Number - any value from 00 to 4F
Key Type - 60 (Key Type A) or 61 (Key Type B)

Example: Command for loading MIFARE Keys with PCD Key Number = 00, Key Type = Key A

Command: FF 00 00 00 0A 07 00 00 60 FF FF FF FF FF FF
Response: 90 00

6.2.2.3 Load DESFire Authentication Keys in to Reader

The following is the Load Keys command format to load DESFire authentication keys into the reader

Command Data:

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*			*Le*
FF	00	00	00	1F	07	01	DESFire Key data (as shown below)	-

DESFire key data					
PCD Key No (1 byte)	AID (3 bytes)	PICC Key No (1 byte)	Key1 (8 bytes)	Key2 (8 bytes)	Key3 (8 bytes)

Where,

PCD Key No - any value from 00 to 07
- Key no 00 refers to PICC Master Key
- Key no's 01 to 07 refer to Application Keys

AID - Application identifier in the card to which the Key belongs
- Must be 000000 for PICC Master Key

PICC Key No - Key No to be used in the DESFire Authenticate command
Key1-2-3 - It can be a DES/TDES or an AES Key
- DES Key : (Key1 = Key2)
- TDES Key : (Key1 ≠ Key2); (Key1 = Key3)
- AES Key : (Key3 = All zeros)
- 3KTDES : (Key1, Key 2, Key3 can be of any values)

Example 1:
Loading TDES PICC Master Key with, AID = '000000', PCD Key No = 00, PICC Key No = 00

Command: FF 00 00 00 1F 07 01 00 00 00 00 00 11 22 33 44 55 66 77 88 12 34 56 78 12 34 56 78 11 22 33 44 55 66 77 88
Response: 90 00

Example 2:
Loading TDES PICC Application Key with, AID = 'C1B1A1', PCD Key No = 01, PICC Key No = 00

Command: FF 00 00 00 1F 07 01 01 A1 B1 C1 00 11 22 33 44 55 66 77 88 12 34 56 78 12 34 56 78 11 22 33 44 55 66 77 88
Response: 90 00

Example 3:
Loading AES PICC Application Key with, AID = 'C3C2C1', PCD Key No = 02, PICC Key No = 01

Command: FF 00 00 00 1F 07 01 02 C1 C2 C3 01 AA AA AA AA BB BB BB BB CC CC CC CC EE EE EE EE 00 00 00 00 00 00 00 00
Response: 90 00

Example 4:
Loading 3KTDES Application Key with, AID = 'C3C2C1', PCD Key No = 03, PICC Key No = 01

Command: FF 00 00 00 1F 07 01 03 C1 C2 C3 01 AA AA AA AA BB BB BB BB CC CC CC CC EE EE EE EE 11 22 33 44 55 66 77 88
Response: 90 00

Note: RdrLoadKeys will fail if any of the command parameters is invalid.

6.2.2.4 Load MIFARE Plus Authentication Keys in to Reader

The following is the Load Keys command format to load MIFARE Plus AES authentication keys into the reader

Command Data:

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*			*Le*
FF	00	00	00	12	07	03	MIFARE Plus Key data (as shown below)	-

MIFARE Plus Key data:

PCD Key No (1 byte)	*PICC Key Block No (LSB)*	*PICC Key Block No (MSB)*	*KEY Data (16 bytes)*
See PCD Key No. table below	Refer to [R13] fro definition	Refer to [R13] for definition	AES Key

PCD Key Number:

PCD Key number determines the location where the key is being stored in the non-volatile memory of the reader

PCD Key Number	*Key description*
0x00 – 0x0D	SL3 AES Sector Keys
0x00 – 0x09	Special Keys (Refer to [R13] and [R14]for details)

PICC Key block number will be used to identify the type of key being loaded

Up to 14 AES sector keys and all special keys can be stored in the non-volatile memory of the reader using this command. Care should be taken to load keys at different locations by changing the PCD Key number or otherwise the keys will be overwritten and only the last loaded key will be available. AES sector keys and special keys do not share same memory space.

Example 1:

Loading SL3 AES sector key for → Sector 1; Key A; PCD Key No. = 1;
Key = 0x11223344556677881234567812345678

Command: FF 00 00 00 15 07 03 01 02 40 11 22 33 44 55 66 77 88 12 34 56 78 12 34 56 78 00
Response: 90 00

Example 2:

Loading Card Master Key (a special key) → PCD Key No. = 1;
Key = 0x11223344556677888877665544332211

Command: FF 00 00 00 15 07 03 01 00 90 11 22 33 44 55 66 77 88 88 77 66 55 44 33 22 11 00
Response: 90 00

Note: RdrLoadKeys will fail if any of the command parameters is invalid

6.2.2.5 Load MIFARE ULC Authentication Keys in to Reader

The following is the Load Keys command format to load MIFARE ULC authentication keys into the reader

Command Data:

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*			*Le*
FF	00	00	00	12	07	04	MIFARE ULC Key data (16bytes as shown below)	-

Example:
Loading MIFARE ULC keys in to the reader, Key1 = 49454D4B41455242, Key2 = 214E4143554F5946

Command: FF 00 00 00 12 07 04 49 45 4D 4B 41 45 52 42 21 4E 41 43 55 4F 59 46
Response: 90 00

Note: RdrLoadKeys will fail if any of the command parameters is invalid

7 Accessing Cards through PC/SC

MIFARE cards and ISO 15693 cards use proprietary 7816 APDU structures. DESFire cards use the 7816 wrapper as described in the DESFire specifications [R3] & [R11]

All command and response bytes are sent and received as hexadecimal values respectively

Note: In the pseudo APDUs described in this section, specifying a value of 00 in the **Le** field indicates maximum no of available response bytes from the card, as described in reference [R7]

7.1 MIFARE Cards

Pseudo APDUs supported for MIFARE cards are explained in this section

7.1.1 Authenticate

This APDU performs three pass authentication with the card for the Block No. specified in the data field. It uses the Key of the Key no specified

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	86	00	00	05	See Table below (5 bytes)	-

Data:

Byte 1	*Byte 2*	*Byte 3*	*Byte 4*	*Byte 5*
01	Block No (MSB)	Block No (LSB)	00	Key No.

Where,

Block No - Block number of the MIFARE card which is to be authenticated
Key No - Key number specified in the RDRLoadKeys command to store the corresponding Authentication key in the non-volatile area of the reader/writer

Response:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

7.1.2 Write Binary

This APDU writes data to the MIFARE Block No. specified

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	D6	00	MIFARE Block No	10	Data to Card	-

Response:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

7.1.3 Read Binary

This APDU reads data from the MIFARE Block No. specified

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	B0	00	MIFARE Block No	-	-	0x00 or 0x10

Response:

Data	*Status Word*	
16 Bytes of data	SW1	SW2

For possible values and description of status word, refer Table 13.

7.1.4 Value Increment

This APDU increments the data in a Value block, using the 4 byte value specified

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*			*Le*
FF	FC	00	00	06	C1	MIFARE Block No	4 bytes of value to be added to the block value (LSB first)	-

Response:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

7.1.5 Value Decrement

This APDU decrements the data in a Value block, using the 4 byte value specified.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*			*Le*
FF	FC	00	00	06	C0	MIFARE Block No	4 bytes of value to be subtracted from the block value (LSB first)	-

Response:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

7.1.6 Value Restore

This APDU copies the data present in the data register of the MIFARE card into the Value block. Technically, this command has no significance.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*			*Le*
FF	FC	00	00	06	C2	MIFARE Block No	4 bytes of value (no significance)	-

Response:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

Note:

1. The Pseudo APDUs for **Authenticate**, **Write Binary** and **Read Binary** described in this section are as defined in the **PC/SC v2.01.09**, Part3: Requirements for PC connected Interface Devices, under section 3.2.2.1

For all the value operations in the MIFARE card, Transfer command need not be sent from the application, as the reader implicitly performs it.

7.2 MIFARE ULC Cards

Pseudo APDUs supported for MIFARE ULC cards are explained in this section

7.2.1 Authenticate

This APDU performs three pass authentication with the card. It uses the Key, which is loaded into the reader using RdrLoadKeys command.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	86	00	00	05	0x0000000000	-

Response:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

7.2.2 Write Binary (16 Bytes)

This APDU writes data to the MIFARE ULC block no. specified

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	D6	00	MIFARE ULC Block No	10	Data to Card	-

Response:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

7.2.3 Write Binary (4 Bytes)

This APDU writes data to the MIFARE ULC block no. specified

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	D6	00	MIFARE ULC Block No	04	Data to Card	-

Response:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

7.2.4 Read Binary

This APDU reads data from the MIFARE ULC block no. specified

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	B0	00	MIFARE Block No	-	-	00 (or) 10

Response:

Data	*Status Word*	
16 Bytes of data	SW1	SW2

Note:

2. The Pseudo APDUs for **Authenticate**, **Write Binary** and **Read Binary** described in this section are as defined in the **PC/SC v2.01.09**, Part3: Requirements for PC connected Interface Devices, under section 3.2.2.1
3. For all the value operations in the MIFARE card, Transfer command need not be sent from the application, as the reader implicitly performs it.

7.3 my-d Move Cards

Pseudo APDUs supported for My-d Move cards are explained in this section

7.3.1 Access

This APDU performs password verification with the My-d move card,

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Command*	*Password*	*Le*
FF	FD	04	01	05	B2	4 Bytes	00

Response:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

7.3.2 Set Password

This APDU changes password in My-d move card with the new password value specified in command

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Command*	*Password*	*Le*
FF	FD	04	01	05	B1	4 Bytes	00

Response:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

7.3.3 Compatibility Write

This APDU writes 4 bytes of data to the My-d Move block no. specified

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Command*	*Block No.*	*Data*	*Le*
FF	FD	06	01	12	A0	1 Byte	16 Bytes	00

Response:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

7.3.4 Write 2 Blocks (8 Bytes)

This APDU writes 8 bytes of data to the My-d Move block no. specified

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Command*	*Block No.*	*Data*	*Le*
FF	FD	06	01	0A	A1	1 Byte	8 Bytes	00

Response:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

7.3.5 Write 1 Block (4 Bytes)

This APDU writes 4 bytes of data to the My-d Move block no. specified

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Command*	*Block No.*	*Data*	*Le*
FF	FD	04	01	06	A2	1 Byte	4 Bytes	00

Response:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

7.3.6 Read 4 Blocks (16 Bytes)

This APDU reads 16 bytes of data from My-d Move block no. specified

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Command*	*Block No.*	*Le*
FF	FD	01	01	02	30	1 Byte	00

Response:

Data	*Status Word*	
16 Bytes of data	SW1	SW2

For possible values and description of status word, refer Table 13.

7.3.7 Read 2 Blocks (8 Bytes)

This APDU reads 8 bytes of data from My-d Move block no. specified

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Command*	*Block No.*	*Le*
FF	FD	01	01	02	31	1 Byte	00

Response:

Data	*Status Word*	
8 Bytes of data	SW1	SW2

7.3.8 Decrement

This APDU decrements counter value of My-d move by the value specified in the command

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Command*	*Decrement value*	*Le*
FF	FD	06	01	03	D0	2 Bytes	00

Response:

Data	*Status Word*	
-	SW1	SW2

For possible values and description of status word, refer Table 13.

7.4 ISO 15693 Cards

Pseudo APDUs supported for ISO 15693 cards are explained in this section.

7.4.1 Read Single Block

This APDU reads 4 bytes of data from the block no specified.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	FC	00	00	No of bytes in Data field	Read single block command as described in reference [R1]	Expected no of bytes from card

Response:

Data	*Status Word*	
Read single block response as described in reference [R1]	SW1	SW2

For possible values and description of status word, refer Table 13.

7.4.2 Write Single Block

This APDU writes 4 bytes of data to the block no specified.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	FC	00	00	No of bytes in Data field	Write single block command as described in reference [R1]	-

Response:

Data	*Status Word*	
Write single block response as described in reference [R1]	SW1	SW2

For possible values and description of status word, refer Table 13.

7.4.3 Lock Block

This APDU Locks the specified Block no. Once successfully locked, the block will become read only.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	FC	00	00	No of bytes in Data field	Lock block command as described in reference [R1]	-

Response:

Data	*Status Word*	
Lock block response as described in reference [R1]	SW1	SW2

For possible values and description of status word, refer Table 13.

7.4.4 Read Multiple Blocks

This APDU reads 4 bytes of data from each of the requested no of blocks, starting from the block no specified.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	FC	00	00	No of bytes in Data field	Read multiple block command as described in reference [R1]	Expected no of bytes from card

Response:

Data	*Status Word*	
Read multiple block response as described in reference [R1]	SW1	SW2

For possible values and description of status word, refer Table 13.

7.4.5 Write AFI

This APDU writes the AFI value specified into the card's memory.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	FC	00	00	No of bytes in Data field	Write AFI command as described in reference [R1]	-

Response:

Data	*Status Word*	
Write AFI response as described in reference [R1]	SW1	SW2

For possible values and description of status word, refer Table 13.

7.4.6 Write DSFID

This APDU writes the DSFID value specified into the card's memory.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	FC	00	00	No of bytes in Data field	Write DSFID command as described in reference [R1]	-

Response:

Data	*Status Word*	
Write DSFID response as described in reference [R1]	SW1	SW2

For possible values and description of status word, refer Table 13.

7.4.7 Get System Information

This APDU retrieves system information, like UID, DSFID, AFI, Memory information, IC Manufacturer code etc from the card.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	FC	00	00	No of bytes in Data field	Get system information command as described in reference [R1]	Expected no of bytes from card

Response:

Data	*Status Word*	
Get system information response as described in reference [R1]	SW1	SW2

For possible values and description of status word, refer Table 13.

7.4.8 Get Multiple Block Security Status

This APDU retrieves the block security status of each of the requested no of blocks, starting from the block no specified.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	FC	00	00	No of bytes in Data field	Get multiple block security status command as described in reference [R1]	00

Response:

Data	*Status Word*	
Get multiple block security status response as described in reference [R1]	SW1	SW2

For possible values and description of status word, refer Table 13.

Note: In all the above 15693 card commands, the optional **Flags** byte and optional **UID** field must be omitted. In all the above 15693 card responses, **Flags** byte will be omitted and **Error Code** (if any) will be sent as SW2

7.5 Crypto RF Cards

Pseudo APDUs supported for Atmel CryptoRF cards are explained in this section.

7.5.1 Set User Zone

This APDU selects the specified user Zone. All further user zone operations will be done in the selected user zone. The command is also used to enable anti-tearing mode, following which all writes to this user zone will use anti-tearing.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	FC	00	00	No of bytes in Data field	Set User Zone Command as per reference [R10]	-

Response:

Data	*Status Word*	
Set User Zone response as per reference [R10]	SW1	SW2

For possible values and description of status word, refer Table 13.

7.5.2 Read User Zone

This APDU reads data from the currently selected user zone.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	FC	00	00	No of bytes in Data field	Read User Zone Command as per reference [R10]	Expected no of bytes from card

Response:

Data	*Status Word*	
Read User Zone response as per reference [R10]	SW1	SW2

For possible values and description of status word, refer Table 13.

7.5.3 Write User Zone

This APDU writes data to the currently selected user zone. In anti-tearing mode the maximum no of bytes that can be written is 8 bytes.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	FC	00	00	No of bytes in Data field	Write User Zone Command as per reference [R10]	-

Response:

Data	*Status Word*	
Write User Zone response as per reference [R10]	SW1	SW2

For possible values and description of status word, refer Table 13.

7.5.4 Read System Zone

This APDU reads system data from the configuration memory of the card. Depending on the value of the PARAM byte (part of the command), this command may read data from the configuration zone, the fuses or a checksum.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	FC	00	00	No of bytes in Data field	Read System Zone Command as per reference [R10]	Expected no of bytes from card

Response:

Data	*Status Word*	
Read System Zone response as per reference [R10]	SW1	SW2

For possible values and description of status word, refer Table 13.

7.5.5 Write System Zone

This APDU writes data to the configuration memory. Depending on the value of the PARAM byte (part of the command), this command may write data to the configuration zone or program fuses. The anti-tearing mode can also be enabled using the PARAM byte. The maximum number of bytes that can be written in anti-tearing mode is 8 bytes.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	FC	00	00	No of bytes in Data field	Write System Zone Command as per reference [R10]	-

Response:

Data	*Status Word*

Write System Zone response as per reference [R10]	SW1	SW2

For possible values and description of status word, refer Table 13.

7.5.6 Check Password

This APDU is used to send the password for validation against the password selected with the password index byte (part of the command). This command is used to gain access, to read or write in user zones that require password validation.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Data*	*Le*
FF	FC	00	00	No of bytes in Data field	Check Password Command as per reference [R10]	-

Response:

Data	*Status Word*	
Check Password response as per reference [R10]	SW1	SW2

For possible values and description of status word, refer Table 13.

Note:
In all the above CryptoRF card commands, the **CID** field (higher nibble of the command byte), must be set to 0. In all the above CryptoRF card responses, the **Command** byte and **ACK/NACK** byte will be omitted. **Status** byte will be sent as SW2.

7.6 Topaz Cards

APDUs supported for Topaz cards are explained in this section

7.6.1 RID

This APDU sends the RID command to the card. This reads the Header ROM value (HRx) and the least significant 4 UID bytes (UID0-4)) of the Topaz card.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Command*	*Le*
FF	FC	00	00	01	78	00

Response:

Data						*Status Word*	
HR0	HR1	UID0	UID1	UID2	UID3	SW1	SW2

For possible values and description of status word, refer Table 13.

7.6.2 RALL

This APDU sends the RALL command to the card. This inturn reads all the 122 data bytes that are present in the Topaz card's memory.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Command*	*Le*
FF	FC	00	00	01	00	00

Response:

Data	*Status Word*	
122 Bytes	SW1	SW2

For possible values and description of status word, refer Table 13.

7.6.3 READ

This APDU sends the READ command to the card. This inturn reads the data byte that is present in the [block number & byte offset] mentioned in the APDU. The 2 byte of the response data contains the data byte.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Command*	*Block No.*	*Byte Offset*	*Le*
FF	FC	00	00	03	01	1 Byte	1 Byte	00

Response:

Data	*Status Word*	
2 Bytes	SW1	SW2

For possible values and description of status word, refer Table 13.

7.6.4 WRITE_E

This APDU sends the WRITE_E command to the card. This inturn erases the data byte that is present in the [block number & byte offset] and writes the data mentioned in the APDU. The second byte of the response data contains the data present in that memory after the execution of the command.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Command*	*Block No.*	*Byte Offset*	*Le*
FF	FC	00	00	03	53	1 Byte	1 Byte	00

Response:

Data	*Status Word*	
2 Bytes	SW1	SW2

For possible values and description of status word, refer Table 13.

7.6.5 WRITE_NE

This APDU sends the WRITE_NE command to the card. This inturn writes the data byte into the [block number & byte offset] mentioned in the APDU, without erasing the already existing data byte. The second byte of the response data contains the data present in that memory after the execution of the command.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Command*	*Block No.*	*Byte Offset*	*Le*
FF	FC	00	00	03	1A	1 Byte	1 Byte	00

Response:

Data	*Status Word*	
2 Bytes	SW1	SW2

For possible values and description of status word, refer Table 13.

7.7 DESFire Cards

In the Multi-ISO reader there are two modes in which one could communication with a DESFire card. They are:

a) EasyDESFire Mode
b) Transparent Mode

In both of the above modes, knowledge on DESFire specification (as per reference [R3]) is a mandate.

7.7.1 EasyDESFire Mode

In the EasyDESFire mode, the application has to just follow the 7816-4 APDU wrapper command set as described in the DESFire specification. The reader would internally take care of all the necessary cryptographic operations and key management.

In order to activate this mode, bit-2 of Firmware Configuration Option (described in section Firmware Configuration Data Structure) has to be cleared using the Set Firmware Configuration command.

Note:

1. "Change Key" commands need special data formatting for DES/TDES, 3KTDES, AES keys.

   If you need to modify a DES/TDES or 3KTDES key in the card, the following is the command format.

CLA	INS	P1	P2	Lc	Data	Le
FF	C4	00	00	19	24 Byte Key Data	00

   If you need to modify an AES key in the card, the following is the command format.

CLA	INS	P1	P2	Lc	Data	Le
FF	C4	00	00	12	16 Byte Key Data + 1 Byte Key Version	00

2. In order to perform an authentication with the changed key, one has to load the changed Key in the reader's non volatile memory using the Load Key command.

3. The following are the reader specific DESFire error codes,

Status Word (HEX)		Description
SW1	SW2	
91	7C	Specified Key does not exist in the PCD
91	7D	Buffer size exceeds PCD limit

Table 9 – DESFire custom error codes

7.7.2 Transparant Mode

In this mode, the application has to perform all the necessary cryptographic operations as required by the DESFire card.

In order to activate this mode, bit-2 of Firmware Configuration Option (described in described in section Firmware Configuration Data Structure) has to be set using the Set Firmware Configuration command.

7.8 MIFARE Plus Cards

At security level 1, MIFARE plus cards behave just like MIFARE cards and hence all the commands supported for MIFARE cards will work as is when card is in this mode.

Commands listed here are supported for both MIFARE Plus S and X cards based on the applicability.

7.8.1 AES Authenticate

This APDU is used to authenticate with the card in order to obtain access to secured data sectors. Based on the key block number, authentication effect would be different. For details refer to [R13]. This is common to all security levels

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*Key block number (LSB)*	*Key block number (MSB)*	*Le*
FF	FB	00	00	03	0x76	Refer to [R13] and [R14] for definition	Refer to [R13] and [R14] for definition	0

Response:

Status Word	
SW1	SW2

This command can be issued when the card is in any security level. Based on the security level and the key block number, the authentication will be done. Find in the below table possible authentications that can be done using this command

Security Level	*Key Block No.*	*Authentication purpose*
SL0/SL1/SL2/SL3	Originality Key Block No.	To check the originality of the card chip
SL1	SL1 AES Authentication Key	To perform additional AES authentication when the card is in SL1 mode
SL2/SL3	AES Sector Key	To perform AES authentication when the card is in SL2/SL3 mode
SL1	Security Level 2 switch Key	To switch to security level 2
SL1/SL2	Security Level 3 switch Key	To switch to security level 3
SL2/SL3	Card Master Key	To perform authentication in order to modify the card master key, card configuration key, Installation Identifier or ATS
SL2/SL3	Card Configuration Key	To perform authentication in order to modify the Field configuration block, card configuration key and Virtual Card keys
SL2/SL3	Virtual Card Polling Enc Key	Refer to [R13] & [R14] for details
SL2/SL3	Virtual Card Polling MAC Key	
SL2/SL3	Select Virtual Card Key	
SL2SL3	Proximity Check Key	

Table 10 – Possible Authentications with a MIFARE Plus card

For possible values and description of status word, refer Table 13.

7.8.2 SL1 Commands

SL1 communication is same as that of standard MIFARE classic, as described in section MIFARE Cards.

Optionally SL1 AES authentication can also be done as described in section AES Authenticate of MIFARE Plus Cards.

7.8.3 SL2 Commands

Below are the APDUs that PCD supports for MIFARE plus SL2 cards.

7.8.3.1 Authenticate

SL2 authentication is same as that of standard MIFARE classic authentication procedure as described in section Authenticate of MIFARE Cards. Before doing this authentication both AES sector key and the MIFARE classic sector key should be loaded into PCD. PCD will internally do both the authentications.

Optionally AES authentication can also be done separately as described in section AES Authenticate of MIFARE Plus Cards.

7.8.3.2 Read

SL2 read is same as that of standard MIFARE classic read procedure as described in section Read Binary of MIFARE Cards. For Multiblock read, following command/response pair is used.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*Block number*	*No. of Blks to read*	*Le*
FF	FD	01	02	03	38	00 to FF	1 to FF (sector trailers not counted)	0

Response:

Data read from the card	*Status Word*	
(No. of blks * 16) bytes	SW1	SW2

For possible values and description of status word, refer Table 13.

7.8.3.3 Write

SL2 write is same as that of standard MIFARE classic read procedure as described in section Write Binary of MIFARE Cards. For Multiblock write, following command/response pair is used.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*Block number*	*Number of blocks*	*Data to write*	*Le*
FF	FB	00	00	3 + (Length of data bytes to be written)	A9	00 to FF	01 to 03	16 * number of blocks	0

Response:

Status Word	
SW1	SW2

7.8.4 SL3 Generic Commands

Below are the APDUs that PCD supports for MIFARE plus SL3 cards.

7.8.4.1 Authentication

SL3 AES authentication is done as described in section AES Authenticate of MIFARE Plus Cards.

7.8.4.2 Reset Authentication

This APDU is used to reset the authenticated status obtained by a successful authenticate command. For details refer to [R13]

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*Le*
FF	FB	00	00	03	0x78	0

Response:

Status Word	
SW1	SW2

For possible values and description of status word, refer Table 13.

7.8.4.3 Read

This APDU is used to read data from the card in SL3.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*Block number (LSB)*	*Block number (MSB)*	*No. of Blks to read*	*Le*
FF	FB	00	00	04	Refer to [R13] and [R14] for definition	Refer to [R13] and [R14] for definition	Refer to [R13] and [R14] for definition	1 to 216 (sector trailers not counter)	0

Response:

Data read from the card	*Status Word*	
(No. of blks * 16) bytes	SW1	SW2

For possible values and description of status word, refer Table 13.

7.8.4.4 Write

This APDU is used to write data to the card in SL3.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*Block number (LSB)*	*Block number (MSB)*	*Data to write*	*Le*
FF	FB	00	00	3 + (Length of data bytes to be written)	Refer to [R13] and [R14] for definition	Refer to [R13] and [R14] for definition	Refer to [R13] and [R14] for definition	16/32/48 bytes	0

Response:

Status Word	
SW1	SW2

For possible values and description of status word, refer Table 13.

7.8.5 SL3 Value Operation commands

These APDUs are used to do value operations on MIFARE plus SL3 card.

7.8.5.1 Increment and decrement operations

This APDU is used to increment or decrement the value in the value block. Transfer command must be sent for the increment/decrement operation to be permanent in card. If restore is used after increment/decrement operation will be temporary.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*Block number (LSB)*	*Block number (MSB)*	*Increment or decrement Data*	*Le*
FF	FB	00	00	07	Refer to [R13] and [R14] for definition	Refer to [R13] and [R14] for definition	Refer to [R13] and [R14] for definition	4 Bytes (LSB first)	0

Response:

Status Word	
SW1	SW2

For possible values and description of status word, refer Table 13.

7.8.5.2 Increment transfer and decrement transfer operations

This APDU is used to increment/decrement value in value block. As the command name indicates no external transfer command is needed for the transaction to be permanent.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*Source Block number (LSB)*	*Source Block number (MSB)*	*Destination Block number (LSB)*	*Destination Block number (MSB)*	*Increment or decrement Data*	*Le*
FF	FB	00	00	09	Refer to [R13] and [R14] for definition	Refer to [R13] and [R14] for definition	Refer to [R13] and [R14] for definition	Refer to [R13] and [R14] for definition	Refer to [R13] and [R14] for definition	4 Bytes (LSB first)	0

Response:

Status Word	
SW1	SW2

For possible values and description of status word, refer Table 13.

7.8.5.3 Transfer and restore operations

This APDU is used to make increment/decrement operations to be permanent in the card.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*Block number (LSB)*	*Block number (MSB)*	*Le*
FF	FB	00	00	03	Refer to [R13] and [R14] for definition	Refer to [R13] and [R14] for definition	Refer to [R13] and [R14] for definition	0

Response:

Status Word	
SW1	SW2

For possible values and description of status word, refer Table 13.

7.8.6 SL3 Virtual Card commands

These APDUs are used to do virtual card related operations on MIFARE plus SL3 card.

7.8.6.1 Write IID

This APDU is used to write card specific IID to the card.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*IID Block number (LSB)*	*IID Block number (MSB)*	*Card IID*	*Le*
FF	FB	00	00	13	A1	01	B0	16 Bytes	0

Response:

Status Word	
SW1	SW2

For possible values and description of status word, refer Table 13.

7.8.6.2 Virtual card support

This APDU is used to check if the IID is registered in the card or not.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*Card IID*	*Le*
FF	FB	00	00	11	42	16 Bytes	0

Response:

Status Word	
SW1	SW2

For possible values and description of status word, refer Table 13.

7.8.6.3 Virtual card support last

This APDU is used to get UID and PICC capabilities from the card.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*Card IID*	*Le*
FF	FB	00	00	11	4B	16 Bytes	0

Response:

Status Word	
SW1	SW2

For possible values and description of status word, refer Table 13.

7.8.6.4 Select virtual card

This APDU is used to select the card with the UID retrieved in the previous command.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*PICC capabilities*	*Card UID*	*Le*
FF	FB	00	00	07 (or) 0A depends on UID size	40	2 Bytes	4 (or) 7 Bytes	0

Response:

Status Word	
SW1	SW2

For possible values and description of status word, refer Table 13.

7.8.6.5 Deselect virtual card

This APDU is used to deselect the card which is selected in the previous command.

***Command* APDU:**

CLA	***INS***	***P1***	***P2***	***Lc***	***Opcode***	***Le***
FF	FB	00	00	01	48	0

Response:

Status Word	
SW1	SW2

For possible values and description of status word, refer Table 13.

7.8.7 SL3 Proximity check commands

These APDUs are used to do proximity checks with the card.

7.8.7.1 Proximity check enable

This APDU is used to make proximity check mandatory for all the transactions in the card.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*Proximity check config Block Number(LSB)*	*Proximity check config Block Number(MSB)*	*Data To enable Proximity Check*	*Le*
FF	FB	00	00	13	A1	03	B0	0055AA00000000 0000000000000000 000	0

Response:

Status Word	
SW1	SW2

For possible values and description of status word, refer Table 13.

7.8.7.2 Proximity check

This APDU is used to do proximity chek in the proximity check enabled cards.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*Le*
FF	FB	00	00	01	F0	0

Response:

Status Word	
SW1	SW2

For possible values and description of status word, refer Table 13.

7.8.7.3 Proximity check disable

This APDU is used to disable mandatory proximity check for card communication.

***Command* APDU:**

CLA	*INS*	*P1*	*P2*	*Lc*	*Opcode*	*Proximity check config Block Number(LSB)*	*Proximity check config Block Number(MSB)*	*Data To disable Proximity Check*	*Le*
FF	FB	00	00	13	A1	03	B0	005555000000 0000000000000 00000000	0

Response:

Status Word	
SW1	SW2

For possible values and description of status word, refer Table 13.

7.9 Generic APDUs

This section describes the generic Pseudo APDUs used with all supported cards.

7.9.1 Get UID

This APDU retrieves the card Unique ID (UID). Length of the UID varies depending on the card.

***Command* APDU:**

CLA	INS	P1	P2	Lc	Data	Le
FF	CA	00	00	-	-	00

Response:

Data	Status Word	
UID of the Card	SW1	SW2

For possible values and description of status word, refer Table 13.

7.9.2 Traverse

Traverse APDU is used to send the "Raw Card Command" in the data field to the card without any command specific processing by the reader/writer and returns the response data from the Card. The reader/writer only takes care of the protocol specific processing (CRC, Prologue field etc ...). This command is mainly used for sending ISO15693 custom commands. For more details, refer section APDU Samples to Access Cards.

The reader/writer uses the Frame type specified in the P2 parameter field and the Frame waiting time specified in the P1 parameter field, while sending the command and receiving the response, respectively

***Command* APDU:**

CLA	INS	P1	P2	Lc	Data	Le
FF	FD	FWT Code (as defined in the table 11)	Frame Type (as defined in the table 12)	No of data bytes sent to the card	Raw Card Command	00

FWT Code	FWT (in microseconds)
00	500
01	1000
02	2000
03	5000
04	10000
05	25000
06	50000
07	75000
08	100000
09	250000
0A	500000
0B	750000
0C	1000000
0D	1250000
0E	1500000
0F	1750000
10	2000000
11	2500000
12	3000000
13	4000000
14	5000000

Table 11 – List of FWT codes

Frame Type	Description
00	FRAMETYPE_SHORT
01	FRAMETYPE_STD
01	FRAMETYPE_ACBITORIENTED

Table 12 – Frame Type

Response:
The response from the card is returned as such without any processing. Reception of any response from the card is considered as success irrespective of the content of the response

Data	***Status Word***	
Response from Card	SW1	SW2

For possible values and description of status word, refer Table 13.

7.10 Status Word

Status Word (HEX)		***Description***
SW1	***SW2***	
90	00	Command Successful
63	00	Reason for error unknown
69	83	Authentication is required to access the block in the card
69	82	Block's security status prevents access
69	88	Wrong key no. was specified to authenticate with the block
67	00	Length parameter in the APDU is wrong
68	00	Class byte in the APDU is wrong
6B	00	Invalid parameter in the APDU
6A	81	Command is not supported
6C	xx	Wrong Le field. Actual Le is mentioned in place of xx
6F	00	No Precise diagnosis
6D	00	Instruction code not supported or invalid

Table 13 – Status Word Description

8 APDU Samples to Access Cards

The basic card access sequence using would be:

- Connect to the card using **SCardConnect** API
- Send commands to the card using **SCardTransmit** API
- Use **SCardDisconnect** API to disconnect from the card

8.1 How to access MIFARE classic cards?

Get Uid
Command: FF CA 00 00 00
Response: XX XX XX XX 90 00

Reader Authenticate (PIN : '0000000000000000')
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

MIFARE Load Keys (PCD Key No. : 00, Key Type : Key A)
Command: FF 00 00 00 0A 07 00 00 60 FF FF FF FF FF FF
Response: 90 00

Authenticate (Block No. : 001E, PCD Key No. : 00)
Command: FF 86 00 00 05 01 00 1E 00 00
Response: 90 00

Read Binary (Block No. : 1E)
Command: FF B0 00 1E 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Read Binary (Block No. : 1E)
Command: FF B0 00 1E 10
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Write Binary (Block No. : 1E)
Command: FF D6 00 1E 10 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16
Response: 90 00

Read Binary (Block No. : 1E)
Command: FF B0 00 1E 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Write Binary (Block No. : 1E)
Command: FF D6 00 1E 10 A1 B2 C3 D4 E5 F6 1F 2E 3D 4C 5B 6A BB CC DD EE
Response: 90 00

Read Binary (Block No. : 1E)
Command: FF B0 00 1E 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Prepare Block as value block (Block No. : 1E, Value : 00000064)
Command: FF D6 00 1E 10 64 00 00 00 9B FF FF FF 64 00 00 00 00 FF 00 FF
Response: 90 00

Value Increment (Block No. : 1E, Increment by 0x000001)
Command: FF FC 00 00 06 C1 1E 01 00 00 00
Response: 90 00

Value Decrement (Block No. : 1E, Decrement by 0x000001)

Command: FF FC 00 00 06 C0 1E 01 00 00 00
Response: 90 00

Value Restore (Block No. : 1E)
Command: FF FC 00 00 06 C2 1E 00 00 00 00
Response: 90 00

8.2 How to access MIFARE UL cards?

Get Uid
Command: FF CA 00 00 00
Response: XX XX XX XX XX XX XX 90 00

Read Binary (Block No. : 09)
Command: FF B0 00 09 00
Response: XX XX XX XX 90 00

Write Binary (Block No. : 09)
Command: FF D6 00 09 04 01 02 03 04
Response: 90 00

Read Binary (Block No. : 09)
Command: FF B0 00 09 00
Response: XX XX XX XX 90 00

Write Binary (Block No. : 09)
Command: FF D6 00 09 10 A1 B2 C3 D4 01 01 01 01 01 01 01 01 01 01 01 01
Response: 90 00

Read Binary (Block No. : 09)
Command: FF B0 00 09 00
Response: XX XX XX XX 90 00

8.3 How to access MIFARE ULC cards?

Get Uid
Command: FF CA 00 00 00
Response: XX XX XX XX XX XX XX 90 00

Following four write commands to blocks 2C, 2D, 2E, 2F, changes the key value in card to Key1 = 49454D4B41455242, Key2 = 214E4143554F5946.

Write Binary (Block No. : 2C)
Command: FF D6 00 2C 04 42 52 45 41
Response: 9000

Write Binary (Block No. : 2D)
Command: FFD6002D044B4D4549
Response: 9000

Write Binary (Block No. : 2E)
Command: FF D6 00 2E 04 46 59 4F 55
Response: 90 00

Write Binary (Block No. : 2F)
Command: FF D6 00 2F 04 43 41 4E 21
Response: 90 00

Reader Authenticate (PIN : '0000000000000000')

Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

MIFARE ULC Load Keys
Command: FF 00 00 00 12 07 04 49 45 4D 4B 41 45 52 42 21 4E 41 43 55 4F 59 46
Response: 90 00

MFUL-C Auth
Command: FF 86 00 00 05 00 00 00 00 00
Response: 90 00

Write Binary (Block No. : 15)
Command: FF D6 00 15 04 AA BB CC DD
Response: 90 00

Write Binary (Block No. : 16)
Command: FF D6 00 16 10 EE FF AB CD 00 00 00 00 00 00 00 00 00 00 00 00
Response: 90 00

Read Binary (Block No. : 15)
Command: FF B0 00 15 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

8.4 How to access My-d Move cards?

Get Uid
Command: FF CA 00 00 00
Response: XX XX XX XX XX XX XX 90 00

Access (Pass. : 00000000)
Command: FF FD 04 01 05 B2 00 00 00 00 00
Response: 90 00

Set Password (Pass. : 12345678)
Command: FF FD 04 01 05 B1 12 34 56 78 00
Response: XX XX XX XX 90 00

Access (Pass. : 12345678)
Command: FF FD 04 01 05 B2 12 34 56 78 00
Response: 90 00

Write 1 Block (Block No. : 11)
Command: FF FD 04 01 06 A2 11 11 22 33 44 00
Response: 90 00

Write 2 Blocks (Block No. : 12, 13)
Command: FF FD 06 01 0A A1 12 55 66 77 88 99 AA BB CC 00
Response: 90 00

Write 4 Block (Block No. : 14)
Command: FF FD 06 01 12 A0 14 DD EE FF 00 10 20 30 40 50 60 70 80 90 A0 B0 C0 00
Response: 90 00

Read 2 Blocks (Block No. : 11, 12)
Command: FF FD 01 01 02 31 11 00
Response: XX XX XX XX XX XX XX XX 90 00

Read 4 Blocks (Block No. : 11, 12, 13, 14)
Command: FF FD 01 01 02 30 11 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Access (Pass. : 12345678)
Command: FF FD 04 01 05 B2 12 34 56 78 00
Response: 90 00

Set Password (Pass. : 00000000)
Command: FF FD 04 01 05 B1 00 00 00 00 00
Response: XX XX XX XX 90 00

Access (Pass. : 00000000)
Command: FF FD 04 01 05 B2 00 00 00 00 00
Response: 90 00

Decrement counter (by value 2)
Command: FF FD 06 01 03 D0 02 00 00
Response: XX XX 90 00

8.5 How to access ISO15693 cards?

Get UID of card
Command: FF CA 00 00 00
Response: XX XX XX XX XX XX XX XX 90 00

Write Single Block
Command: FF FC 00 00 06 21 0F 01 02 03 04
Response: 90 00

Lock Block (Block no: 13)
Command: FF FC 00 00 02 22 13
Response: 90 00

Read Single Block (Block no: 0F)
Command: FF FC 00 00 03 20 0F 00
Response: XX XX XX XX 90 00

Write Multiple Block (Start Block no: 10, No of Blocks: 02)
Command: FF FC 00 00 0B 24 10 01 01 01 01 01 01 01 01
Response: 90 00

Read Multiple Block (Start Block no: 10, No of Blocks: 04)
Command: FF FC 00 00 04 23 10 03 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Write AFI
Command: FF FC 00 00 02 27 F0
Response: 90 00

Write DSFID
Command: FF FC 00 00 02 29 F0
Response: 90 00

Lock AFI
Command: FF FC 00 00 01 28
Response: 90 00

Lock DSFID

Command: FF FC 00 00 01 2A
Response: 90 00

Get Multiple Block Security Status (Start Block no: 00, No of Blocks: 02)
Command: FF FC 00 00 04 2C 00 01 00
Response: XX XX 90 00

Get System Info
Command: FF FC 00 00 02 2B 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

8.6 How to access ICODE-SLI cards?

Get UID of card
Command: FF CA 00 00 00
Response: 20 4D CC 08 00 01 04 E0 90 00

Traverse - Inventory Read (Start Block no: 10, No of Blocks: 05)
Command: FF FD 03 01 06 66 A0 04 00 10 04 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Traverse - Set EAS
Command: FF FD 04 01 0B 22 A2 04 BA 4D CC 08 00 01 04 E0 00
Response: 00 90 00

Traverse - EAS Alarm
Command: FF FD 04 01 0B 22 A5 04 BA 4D CC 08 00 01 04 E0 00
Response: 00 XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Traverse - Lock EAS
Command: FF FD 04 01 0B 22 A4 04 BA 4D CC 08 00 01 04 E0 00
Response: 00 00 90 00

Traverse - Reset EAS
Command: FF FD 04 01 0B 22 A3 04 BA 4D CC 08 00 01 04 E0 00
Response: 009000

Traverse - EAS Alarm
Command: FF FD 04 01 0B 22 A5 04 BA 4D CC 08 00 01 04 E0 00
Response: 00 XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

8.7 How to access Crypto RF cards?

Get UID of card
Command: FF CA 00 00 00
Response: 50 FF FF FF 90 00

Set User Zone (User Zone: 00)
Command: FF FC 00 00 02 01 00
Response: 90 00

Write User Zone (Start Address: 0000, Length: 10)
Command: FF FC 00 00 14 03 00 00 0F 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0E 0E 0F
Response: 90 00

Write User Zone - Traverse (Start Address: 0000, Length: 01)

Command: FF FD 04 01 05 03 00 00 00 0B 00
Response: XX 00 00 90 00

Read User Zone (Start Address: 0000, Length: 20)
Command: FF FC 00 00 04 02 00 00 1F 20
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Set User Zone with Anti Tearing (User Zone: 00)
Command: FF FC 00 00 02 01 80
Response: 90 00

Write User Zone with Anti Tearing (Start Address: 0000, Length: 08)
Command: FF FC 00 00 0C 03 00 00 07 08 09 0A 0B 0C 0D 0E 0F
Response: 90 00

Read User Zone (Start Address: 0000, Length: 20)
Command: FF FC 00 00 04 02 00 00 1F 20
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Read System Zone Fuse (Length: 01)
Command: FF FC 00 00 04 06 01 FF 00 00
Response: XX 90 00

Read System Zone Check Sum (Length: 02)
Command: FF FC 00 00 04 06 02 FF 01 00
Response: XX XX 90 00

Check Password
Command: FF FC 00 00 05 0C 07 10 14 7C
Response: 90 00

Write System Zone (Start Address: 0008, Length: 10)
Command: FF FC 00 00 14 04 00 08 0F 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
Response: 90 00

Read System Zone (Start Address: 0008, Length: 41)
Command: FF FC 00 00 04 06 00 08 40 41
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX xx

Write System Zone with Anti tearing (Start Address: 0008, Length: 08)
Command: FF FC 00 00 0C 04 80 08 07 08 09 0A 0B 0C 0D 0E 0F
Response: 90 00

Read System Zone (Start Address: 0008, Length: 41)
Command: FF FC 00 00 04 06 00 08 40 41
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX xx

Write System Zone Fuse (Fuse: SEC, Fuse Address: 07)
Command: FF FC 00 00 05 04 01 07 00 01
Response: 90 00

8.8 How to access Topaz cards?

Get UID
Command: FF CA 00 00 00
Response: XX XX XX XX XX XX XX 90 00

RID (Cmd: 0x78)
Command: FF FC 00 00 01 78 00
Response: XX XX XX XX XX XX 90 00

RALL (Cmd: 0x00)
Command: FF FC 00 00 01 00 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XXXX 90 00

READ (Cmd: 0x01, Block: 0x02, Offset: 0x03)
Command: FF FC 00 00 03 01 02 03 00
Response: XX XX 90 00

WRITE_E (Cmd: 0x53, Block: 0x02, Offset: 0x03, Data: 0xA3)
Command: FF FC 00 00 04 53 02 03 A3 00
Response: XX A3 90 00

WRITE_NE (Cmd: 0x1A, Block: 0x02, Offset: 0x04, Data: 0xA4)
Command: FF FC 00 00 04 1A 02 04 A4 00
Response: XX A4 90 00

8.9 How to access DESFIRE cards?

Select Application (AID : '000000')
Command: 90 5A 00 00 03 00 00 00 00
Response: 91 00

Authenticate (PICC Key No. : 00)
Command: 90 0A 00 00 01 00 00
Response: 91 00

Format PICC
Command: 90 FC 00 00 00
Response: 91 00

Create DES/TDES Application (AID : 'C3C2C1')
Command: 90 CA 00 00 05 C1 C2 C3 0F 07 00
Response: 91 00

ReaderAuthenticate (PIN : '0000000000000000')
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Desfire Load Keys (PCD Key No. : 01, AID : 'C3C2C1', PICC Key No. : 00)
Command:F0 00 00 00 1F 07 01 01 C1 C2 C3 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
Response: 90 00

ReaderAuthenticate (PIN : '0000000000000000')
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00

Response: 90 00

Desfire Load Keys (PCD Key No. : 02, AID : 'C3C2C1', PICC Key No. : 01)
Command:F0 00 00 00 1F 07 01 02 C1 C2 C3 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Response: 90 00

ReaderAuthenticate (PIN : '0000000000000000')
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Desfire Load Keys (PCD Key No. : 03, AID : 'C3C2C1', PICC Key No. : 02)
Command: F0 00 00 00 1F 07 01 03 C1 C2 C3 02 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Response: 90 00

Select Application (AID : 'C3C2C1')
Command: 90 5A 00 00 03 C1 C2 C3 00
Response: 91 00

Authenticate (PICC Key No. : 00)
Command: 90 0A 00 00 01 00 00
Response: 91 00

ChangeKeySettings (Free Access)
Command: 90 54 00 00 01 0F 00
Response: 9100

Change Key (PICC Key No. : 02)
Command: 90 C4 00 00 19 02 FF FF FF FF FF FF FF FF AA AA AA AA AA AA AA AA FF FF FF FF FF FF FF FF 00
Response: 91 00

ReaderAuthenticate (PIN : '0000000000000000')
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

DESFire Load keys (PCD Key No. : 03, AID : 'C3C2C1', PICC Key No. : 02)
Command: FF 00 00 00 1F 07 01 03 C1 C2 C3 02 FF FF FF FF FF FF FF FF AA AA AA AA AA AA AA AA FF FF FF FF FF FF FF FF
Response: 90 00

Change Key (PICC Key No. : 00)
Command: 90 C4 00 00 19 00 AA AA BB BB CC CC DD DD EE EE FF FF 11 11 22 22 AA AA BB BB CC CC DD DD 00
Response: 91 00

ReaderAuthenticate (PIN : '0000000000000000')
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

DESFire Load keys (PCD Key No. : 01, AID : 'C3C2C1', PICC Key No. : 00)
Command: FF 00 00 00 1F 07 01 01 C1 C2 C3 00 AA AA BB BB CC CC DD DD EE EE FF FF 11 11 22 22 AA AA BB BB CC CC DD DD
Response: 90 00

Authenticate (PICC Key No. : 02)
Command: 90 0A 00 00 01 02 00
Response: 91 00

Authenticate (PICC Key No. : 00)
Command: 90 0A 00 00 01 00 00
Response: 91 00

Create StdData File (File No. : 01)
Command: 90 CD 00 00 07 01 00 EE EE FF 02 00 00
Response: 91 00

Create Value File (File No. : 02)
Command: 90 CC 00 00 11 02 00 EE EE 00 00 00 00 FF 00 00 00 00 00 00 00 01 00
Response: 91 00

Create Linear Record File (File No. : 03)
Command: 90 C1 00 00 0A 03 00 EE EE 20 00 00 08 00 00 00
Response: 91 00

Change File Settings (File No. : 01, Comm Mode : Encrypted)
Command: 90 5F 00 00 04 01 03 0E 00 00
Response: 91 00

Write Data (File No. : 01)
Command: 90 3D 00 00 87 01 00 01 00 80 00 00 AA AA BB BB CC CC DD DD AA AA BB BB CC CC DD DD AA AA BB BB CC CC DD DD AA AA B BB BC CC CD DD DA AA AB BB B CC CC DD DD AA AA BB BB CC CC DD DD AA AA BB BB CC CC DD DD AA AA BB BB CC CC DD DD AA AA BB BB CC CC DD DD AA AA BB BB CC CC DD DD AA AA BB BB CC CC DD DD AA AA BB BB CC CC DD DD AA AA BB BB CC CC DD DD AA AA BB BB CC CC DD DD AA AA BB BB CC CC DD DD 00
Response: 91 00

Read Data (File No. : 01)
Command: 90 BD 00 00 07 01 00 00 00 F4 01 00 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 91 00

Get Value (File No. : 02)
Command: 90 6C 00 00 01 02 00
Response: XX XX XX XX 91 00

Credit (File No. : 02)
Command: 90 0C 00 00 05 02 10 00 00 00 00
Response: 91 00

Commit Transaction

Command: 90 C7 00 00 00
Response: 91 00

Get Value (File No. : 02)
Command: 90 6C 00 00 01 02 00
Response: XX XX XX XX 91 00

Debit (File No. : 02)
Command: 90 DC 00 00 05 02 08 00 00 00 00
Response: 91 00

Commit Transaction
Command: 90 C7 00 00 00
Response: 91 00

Get Value (File No. : 02)
Command: 90 6C 00 00 01 02 00
Response: XX XX XX XX 91 00

Debit (File No. : 02)
Command: 90 DC 00 00 05 02 04 00 00 00 00
Response: 91 00

Abort Transaction
Command: 90 A7 00 00 00
Response: 91 00

Get Value (File No. : 02)
Command: 90 6C 00 00 01 02 00
Response: XX XX XX XX 91 00

Write Records (File No. : 03)
Command: 90 3B 00 00 27 03 00 00 00 20 00 00 AA AA BB BB CC CC DD DD AA AA BB BB CC CC DD DD AA AA BB BB CC CC DD DD AA AA BB BB CC CC DD DD 00
Response: 91 00

Commit Transaction
Command: 90 C7 00 00 00
Response: 91 00

Read Records (File No. : 03)
Command: 90 BB 00 00 07 03 00 00 00 00 00 00 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 9100

Authenticate (PICC Key No. : 00)
Command: 90 0A 00 00 01 00 00
Response: 91 00

Format PICC
Command: 90 FC 00 00 00
Response: 91 00

8.9.1 DESFIRE EV1 Specific commands

Select Application (AID : '000000')
Command: 90 5A 00 00 03 00 00 00 00
Response: 91 00

Authenticate (PICC key No : 00)

Command: 90 0A 00 00 01 00 00
Response: 91 00

Change Key (Master Level from DES/TDES to AES)
Command: 90 C4 00 00 12 80 12 34 56 78 9A BC DE F1 23 45 67 89 AB CD EF EE AE 00
Response: 91 00

ReaderAuthenticate (PIN : '0000000000000000')
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

DESFire Load keys (PCD Key No. : 00 AID: '000000', PICC Key No. : 00)
Command: FF 00 00 00 1F 07 01 00 00 00 00 00 12 34 56 78 9A BC DE F1 23 45 67 89 AB CD EF EE 00 00 00 00 00 00 00 00
Response: 90 00

Authenticate (Master level AES Authentication)
Command: 90 AA 00 00 01 00 00
Response: 91 00

Change Key (Change Master Key from AES to 3KTDES)
Command: 90 C4 00 00 19 40 11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00 12 34 56 78 90 AB CD EF 00
Response: 91 00

ReaderAuthenticate (PIN : '0000000000000000')
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Load DESFire key with PCD Key No: 0x00, AID: '000000' and PICC Key No: 0x00
Command: FF 00 00 00 1F 07 01 00 00 00 00 00 11 22 33 44 55 66 77 88 99 AA BB CC DD EE FF 00 12 34 56 78 90 AB CD EF
Response: 90 00

Authenticate (Master level 3K3DES Authentication)
Command: 90 1A 00 00 01 00 00
Response: 91 00

GetVersion
Command: 90 60 00 00 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 91 00

Get Card UID
Command: 90 51 00 00 00
Response: XX XX XX XX XX XX XX 91 00

Get FreeMemory
Command: 90 6E 00 00 00
Response: XX XX XX 91 00

Set Configuration
Command: 90 5C 00 00 02 00 00 00
Response: 91 00

Set Configuration
Command: 90 5C 00 00 1A 01 11 11 22 22 33 33 44 44 55 55 66 66 77 77 88 88 11 11 22 22 33 33 44 44 FF 00
Response: 91 00

Set Configuration
Command: 90 5C 00 00 0D 02 0C 75 77 81 02 80 00 00 00 00 00 00 00 00
Response: 91 00

Change Key (Master level from 3K3DES to DES/TDES)
Command: 90 C4 00 00 19 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Response: 91 00

ReaderAuthenticate (PIN : '0000000000000000')
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

DESFire Load keys (PCD Key No. : 00 AID : '000000', PICC Key No. : 00)
Command: FF 00 00 00 1F 07 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Response: 90 00

Authenticate (PICC Key No. : 00)
Command: 90 0A 00 00 01 00 00
Response: 91 00

Format PICC
Command: 90 FC 00 00 00
Response: 91 00

8.10 How to access MIFARE Plus cards?

Accessing MIFARE Plus SL2 Cards:

Reader Authenticate (PIN: '0000000000000000')
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

MIFARE Load Keys (PCD Key No. : 00, Key Type : Key A, Key: FFFFFFFFFFFF)
Command: FF 00 00 00 0A 07 00 00 60 FF FF FF FF FF FF
Response: 90 00

Authenticate Reader with PIN '0000000000000000'
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Load MIFARE Plus AES Sector Key (Sector: 0, PCD Key No: 0x00, Key type: Key A, Key: 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF)
Command: FF 00 00 00 15 07 03 00 00 40 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

MFClassic Authenticate & SL2 AES Authenticate (Sector : 0, Block No. : 0000, PCD Key No. : 00, Key type: Key A)
Command: FF 86 00 00 05 01 00 00 00 00
Response: 90 00

Write Binary (Block No. : 01)
Command: FF D6 00 01 10 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16
Response: 90 00

ReadBinary (Block No. : 01)
Command: FF B0 00 01 00

Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Multi Block Write (Block No. : 01 and 02)
Command: FF FB 00 00 23 A9 01 02 12 34 56 78 9A BC DE F0 0F ED CB A9 87 65 43 21 12 34 56 78 9A BC DE F0 0F ED CB A9 87 65 43 21 00
Response: 90 00

Multi Block Read (Block No. : 00, 01 and 02)
Command: FF FD 01 02 03 38 00 03 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Accessing MIFARE Plus S SL3 Cards:

Authenticate Reader with PIN '0000000000000000'
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Load MIFARE Plus AES Sector Key for sector 1 with PCD Key No: 0x00, Key type: Key A and Key: 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
Command: FF 00 00 00 15 07 03 00 02 40 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

Authenticate with card to access sector 1 using the key loaded in to the reader memory
Command: FF FB 00 00 03 76 02 40 00
Response: 90 00

Write data to 3 consecutive blocks - block0 to block2 in sector 1
Command: FF FB 00 00 33 A3 04 00 0F 0E 0D 0C 0B 0A 09 08 07 06 05 04 03 02 01 00 55 AA 44 BB 33 CC 22 DD 11 EE 00 FF AB CD EF 55 AA 55 BB 44 CC 33 DD 22 EE 11 FF 00 99 66 88 77 00
Response: 90 00

Read data from 3 consecutive blocks - block0 to block2 in sector 1
Command: FF FB 00 00 04 33 04 00 03 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Change Sector 1 AES key (Key A) to 0x0F0E0D0C0B0A09080706050403020100
Command: FF FB 00 00 13 A1 02 40 0F 0E 0D 0C 0B 0A 09 08 07 06 05 04 03 02 01 00 00
Response: 90 00

Authenticate Reader with PIN '0000000000000000' to load the new AES Sector 1 key
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

MIFARE Plus Load Key - AES sector key - Sector 1 - Type A (Key - 0x0F0E0D0C0B0A09080706050403020100)
Command: FF 00 00 00 15 07 03 00 02 40 0F 0E 0D 0C 0B 0A 09 08 07 06 05 04 03 02 01 00 00
Response: 90 00

Authenticate with card to access sector 1 using the key loaded in to the reader memory
Command: FF FB 00 00 03 76 02 40 00
Response: 90 00

Change Card Master Key to 0x00000000000000000000000000000000

Authenticate Reader with PIN '0000000000000000'
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Load MIFARE Plus Card Master Key with PCD Key No: 0x00 and Key: 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
Command: FF 00 00 00 15 07 03 00 00 90 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

Authenticate with card using the Card Master Key loaded in to the reader memory
Command: FF FB 00 00 03 76 00 90 00
Response: 90 00

Change Card Master Key to 0x00000000000000000000000000000000
Command: FF FB 00 00 13 A1 00 90 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Response: 90 00

Load new Card Master Key to reader non-volatile memory

Authenticate Reader with PIN '0000000000000000' to load the new Card Master Key to the reader memory
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Load new Card Master Key with PCD Key No: 0x00 and Key: 0x00000000000000000000000000000000
Command: FF 00 00 00 15 07 03 00 00 90 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Response: 90 00

Authenticate with the card using the new Card Master Key loaded in to the reader memory
Command: FF FB 00 00 03 76 00 90 00
Response: 90 00

Accessing MIFARE Plus X SL3 Cards:

Authenticate Reader with PIN '0000000000000000'
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Load MIFARE Plus AES Card Master Key (PCD Key No: 0x00, Key: 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF)
Command: FF 00 00 00 15 07 03 00 00 90 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

MIFARE Plus Aes Authenticate (Card Master key Authentication)
Command: FF FB 00 00 03 76 00 90 00
Response: 90 00

Write to Block 0xB000 (MFP Configuration block) Change number of unmaced commands in one session to 0x20 and make plain access to all blocks
Command: FF FB 00 00 13 A1 00 B0 20 0F 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Response: 90 00

Authenticate Reader with PIN '0000000000000000'
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Load MIFARE Plus AES Sector Key (Sector No: 0x03, PCD Key No: 0x01, Key type: Key A and Key: 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF)
Command: FF 00 00 00 15 07 03 01 06 40 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

MIFARE Plus Aes Authenticate (Sector : 0x03, Key type: Key A)
Command: FF FB 00 00 03 76 06 40 00

Response: 90 00

Write data (Sector No: 0x03, Block No: 0x0C, No of Blocks: 0x01, Write Type: Encrypted, MAC on Command, MAC on Response)
Command: FF FB 00 00 13 A1 0C 00 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

Read data (Sector No: 0x03, Block No: 0x0C, No of Blocks: 0x01, Read Type : Encrypted, MAC on Command, MAC on Response)
Command: FF FB 00 00 04 31 0C 00 01 00
XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 9000

Write data (Sector No: 0x03, Block No: 0x0C, 0x0D, No of Blocks: 0x02, Write Type: Plain, MAC on Command, MAC on Response)
Command: FF FB 00 00 23 A3 0C 00 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE 00
Response: 90 00

Read Data (Sector No: 0x03, Block No: 0x0C, 0x0D, No of Blocks: 0x02, Read Type: Plain, MAC on Command, MAC on Response)
Command: FF FB 00 00 04 33 0C 00 02 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Write data (Sector No: 0x03, Block No: 0x0C, 0x0D, 0x0E, No of Blocks: 0x03, Write Type: Encrypted, MAC on Command, MAC on Response)
Command: FF FB 00 00 33 A1 0C 00 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE EE DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD 00
Response: 90 00

Read Data (Sector No: 0x03, Block No: 0x0C, 0x0D, No of Blocks: 0x02, Read Type: Encrypted, No MAC on Command, MAC on Response)
Command: FF FB 00 00 04 35 0C 00 03 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Authenticate Reader with PIN '0000000000000000'
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Load MIFARE Plus AES Sector Key (Sector No: 0x05, PCD Key No: 0x01, Key type: Key A and Key: 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF)
Command: FF 00 00 00 15 07 03 01 0A 40 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

MIFARE Plus Aes Authenticate (Sector: 0x05, Key type: Key A)
Command: FF FB 00 00 03 76 0A 40 00
Response: 90 00

Prepare Value Block (Sector No: 0x05, Block No: 0x14, Write Type: Encrypted, MAC on Command, MAC on Response, Value: 000000FF)
Command: FF FB 00 00 13 A1 14 00 FF 00 00 00 00 FF FF FF FF 00 00 00 0F F0 0F F0 00
Response: 90 00

Read data (Sector No: 0x05, Block No: 0x14 No of Blocks: 0x01, Read Type: Encrypted, MAC on Command, MAC on Response)
Command: FF FB 00 00 04 31 14 00 01 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Value Decrement (Sector No: 0x05, Block No: 0x14, Decrement Type : Encrypted, MAC on Command, MAC on Response, Value : 00000008)
Command: FF FB 00 00 07 B3 14 00 08 00 00 00 00
Response: 90 00

Transfer Decrementd value to the block (Sector No: 0x05, Block No: 0x14, Transfer Type: MAC on Command, MAC on Response)
Command: FF FB 00 00 03 B5 14 00 00
Response: 90 00

Value Increment (Sector No: 0x05, Block No: 0x14, Increment Type: Encrypted, MAC on Command, MAC on Response, Value: 00000008)
Command: FF FB 00 00 07 B1 14 00 02 00 00 00 00
Response: 90 00

Transfer Incremented value to the block (Sector No: 0x05, Block No: 0x14, Transfer Type: MAC on Command, MAC on Response)
Command: FF FB 00 00 03 B5 14 00 00
Response: 90 00

Value Decrement and Transfer (Sector No: 0x05, Block No: 0x14, Decrement Type: Encrypted, MAC on Command, MAC on Response, Value: 00000008)
Command: FF FB 00 00 09 B9 14 00 14 00 08 00 00 00 00
Response: 90 00

Value Increment and Transfer Value of Block 0x14 by 2
Command: FF FB 00 00 09 B7 14 00 14 00 02 00 00 00 00
Response: 90 00

Read data (Sector No: 0x05, Block No: 0x14 No of Blocks: 0x01, Read Type: Plain, MAC on Command, MAC on Response)
Command: FF FB 00 00 04 33 14 00 01 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Value Decrement (Sector No: 0x05, Block No: 0x14, Decrement Type: Encrypted, MAC on Command, MAC on Response, Value: 00000008)
Command: FF FB 00 00 07 B3 14 00 08 00 00 00 00
Response: 90 00

Restore previous contents of the block (Sector No: 0x05, Block No: 0x14, Transfer Type: MAC on Command, MAC on Response)
Command: FF FB 00 00 03 C3 14 00 00
Response: 90 00

Read data (Sector No: 0x05, Block No: 0x14 No of Blocks: 0x01, Read Type: Plain, No MAC on Command, MAC on Response)
Command: FF FB 00 00 04 37 14 00 01 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

MIFARE plus Aes Authenticate (Card Master key Authentication)
Command: FF FB 00 00 03 76 00 90 00
Response: 90 00

Write to Block 0xB000 (MFP Configuration block) Change number of unmaced commands in one session to 0x00 and make plain access to all blocks
Command: FF FB 00 00 13 A1 00 B0 00 0F 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Response: 90 00

SL1 Switching from SL0

WritePerso (Card Master Card Key)
Command: FF FB 00 00 13 A8 00 90 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

WritePerso (Card Configuration Key)
Command: FF FB 00 00 13 A8 01 90 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

WritePerso (Security Level2 Switch Key)
Command: FF FB 00 00 13 A8 02 90 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

WritePerso (Security Level3 Switch Key)
Command: FF FB 00 00 13 A8 03 90 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

CommitPerso (Make all the above changes permanent in card and switch to SL1)
Command: FF FB 00 00 01 AA 00
Response: 90 00

SL2 Switching from SL1

Authenticate Reader with PIN '0000000000000000'
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

MIFARE Plus Load SL2 Switch Key (Pcd Key No. : 0x03, Key Value: 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF)
Command: FF 00 00 00 15 07 03 03 02 90 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

MIFARE plus Aes Authenticate (SL1 to SL2 Switching)
Command: FF FB 00 00 03 76 02 90 00
Response: 90 00

SL3 Switching from SL1 or from SL2

Authenticate Reader with PIN '0000000000000000'
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

MIFARE Plus Load SL3 Switch Key (Pcd Key No. : 0x03, Key Value: 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF)
Command: FF 00 00 00 15 07 03 04 03 90 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

MIFARE Plus Aes Authenticate (SL1 to SL3 Switching or SL2 to SL3 Switching)
Command: FF FB 00 00 03 76 03 90 00
Response: 90 00

SL3 Virtual Card commands

Authenticate Reader with PIN '0000000000000000' to load the card master key to the reader memory
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Load card master key in to reader non-volatile memort with PCD Key No: 0x00 and Key: 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
Command: FF 00 00 00 15 07 03 00 00 90 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00

Response: 90 00

Authenticate with the card using the card master key loaded in to the reader memory
Command: FF FB 00 00 03 76 00 90 00
Response: 90 00

Write to IID block (IID: 0x00000000000000000000000000000000)
Command: FF FB 00 00 13 A1 01 B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Response: 90 00

Authenticate Reader with PIN '0000000000000000' to load VC Polling Mac key to reader non-volatile memory
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Load MIFARE plus VC Polling Mac Key to reader non volatile memory (PCD Key No: 0x00 and Key: 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF)
Command: FF 00 00 00 15 07 03 00 81 A0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

Authenticate Reader with PIN '0000000000000000' to load VC Polling Enc key to reader non-volatile memory
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Load MIFARE plus VC Polling Enc Key to reader non volatile memory (PCD Key No: 0x01 and Key: 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF)
Command: FF 00 00 00 15 07 03 01 80 A0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

Authenticate Reader with PIN '0000000000000000' to load VC Select Mac key to reader non-volatile memory
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Load MIFARE plus VC Select Mac Key to reader non volatile memory (PCD Key No: 0x02 and Key: 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF)
Command: FF0000001507030200A0FFFFFFFFFFFFFFFFFFFFFFFFFFFFFF00
Response: 90 00

Issue Virtual Card Support command (This is an optional command - refer spec for further details)
Command: FF FB 00 00 11 42 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Response: 90 00

Issue Virtual Card Support Last command
Command: FF FB 00 00 11 4B 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Response: XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX XX 90 00

Issue Select Virtual Card command
Command: FF FB 00 00 11 40 00 0B 04 81 53 A9 61 28 80 00 00 00 00 00 00 00 00
Response: 90 00

Issue Deselect Virtual Card command
Command: FF FB 00 00 01 48 00
Response: 90 00

SL3 proximity check commands

Authenticate Reader with PIN '0000000000000000' to load Card Configuration key to the reader non-volatile memory
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Load MIFARE plus Card Configuration Key to reader non-volatile memory (PCD Key No: 0x01 and Key: 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF)
Command: FF 00 00 00 15 07 03 01 01 90 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF 00
Response: 90 00

Authenticate with card using the card configuration key loaded in to the reader memory
Command: FF FB 00 00 03 76 01 90 00
Response: 90 00

Set the PC mandatory byte in the Field Configuration Block
Command: FF FB 00 00 13 A1 03 B0 00 55 AA 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Response: 90 00

Authenticate Reader with PIN '0000000000000000' to load Proximity Check key to reader non-volatile memory
Command: FF 00 00 00 09 09 00 00 00 00 00 00 00 00
Response: 90 00

Load the Proximity Check key to reader non-volatile memory (PCD Key No: 0x08 and Key: 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF)
Command: FF0000001507030801A0FFFFFFFFFFFFFFFFFFFFFFFFFFFFFF00
Response: 90 00

Issue Proximity check command to the card
Command: FF FB 00 00 01 F0 00
Response: 90 00

Authenticate with card using the card configuration key loaded in to the reader memory
Command: FF FB 00 00 03 76 01 90 00
Response: 90 00

Clear the Proximity Check mandatory byte in the Field Configuration Block
Command: FF FB 00 00 13 A1 03 B0 00 55 55 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Response: 90 00

9 Reader firmware

9.1 Flavors

The below provided information shows the available firmware flavors for Multi-ISO USB readers:

PC/SC	PC/SC compliant firmware including feature to use escape commands. Proprietary Identive PC/SC driver is required.
CCID	PC/SC compliant firmware with CCID interface that uses the standard CCID drivers provided by the OS.
CCID+HID	Same functionality as CCID firmware but with additional keyboard emulation feature which uses the HID device class drivers provided by the OS. HID feature can be turned on/off via command.

Default firmware flavors for the Multi-ISO reader products:

PN:	Firmware version:
AMID2US00	PC/SC
AMIM2US00	PC/SC
AMID2US00-KBD	CCID+HID

Table 14 - Default firmware flavor

9.2 Upgrade

The versatile Multi-ISO reader series is capable to run every above mentioned firmware version. The different upgrade processes using the Identive Card Reader Suite application are shown in the following document:

Identive_Multi-X_Firmware_Update_Manual_xx.pdf

If you want to embed the reader firmware update functionality into your own application you can use the “DFU Console” application. See the following document for details:

Identive_DFU_Console_UM_ xx.pdf.

9.3 VID & PID

The vendor ID (VID) and product ID (PID) of the Multi-ISO firmwares are:

Firmware	VID	PID
PC/SC	1FFAh	0001h
CCID	1FFAh	000Bh
CCID+HID	1FFAh	000Ch

Table 15 - VID & PID

10 CCID & CCID/HID specifics

10.1 Applications communicating with Multi-ISO reader

There is a different function call required for sending escape commands to CCID and non CCID readers. For the CCID readers the application has to send escape commands in SCARD_SHARE_DIRECT mode using SCardControl () API with the escape IOCTL as mentioned in this Microsoft page: http://msdn.microsoft.com/en-us/windows/hardware/gg487509

Whereas for the non CCID readers the applications have to use the SCardTransmit () API.

10.2 Usage of escape commands

For sending the escape-commands/pseudo-apdus to a CCID reader in Windows, refer to the Microsoft link above.

Note: For enabling escape commands in Windows 7, one has to remember that a non-zero EscapeCommandEnable DWORD value has to be created under the .\DeviceParameters\WUDFUsbccidDriver\ key in the registry. The following is the screenshot of the same:

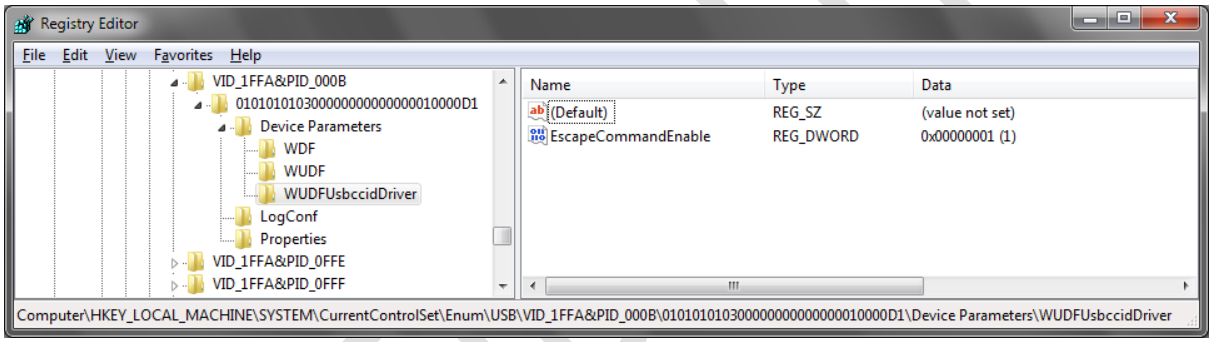

Figure 14 –Showing the registry key for creating "EscapeCommandEnable" value

The following is the registry path of the same:
HKLM\SYSTEM\CurrentControlSet\Enum\USB\VID_1FFA&PID_000B*x*\DeviceParameters\WUDFUsbCcidDriver\

Where *x* is the serial number of the Multi-ISO device

10.3 Enable / Disable HID interface

The recommended way to enable/disable the HID interface of the Multi-ISO CCID+HID reader is to use the keyboard configuration tool. If the tool is not available, one can perform the same using the following APDUs.

Get Keyboard Configuration Data:
Command: FF 00 00 00 01 1A
Resonse: *<22 bytes of Keyboard Configuration Data>* 90 00

Set Keyboard Configuration Data:
Command: FF 00 00 00 17 19 *<22 bytes of Keyboard Configuration Data>*
Resonse: 90 00

For disabling the HID interface, one has to perform **Get Keyboard Configuration** and patch the first byte of the *Keyboard Configuration Data* as 0xFF and send the modified configuration data (without disturbing the remaining bytes) using **Set Keyboard Configuration**.

To re-enable the HID interface, one has to again perform **Get Key Configuration** and patch the first byte of the *Keyboard Configuration Data* as per the following and send the modified configuration (without disturbing the remaining bytes) using **Set Keyboard Configuration**.

0x00 Tag UID reading
0x01 Tag Memory reading

Appendix A Terms and Abbreviations

Terms / Abbreviations	*Description*
3KTDES	3-Key Triple Data Encryption Standard
ACK	Acknowledgement
AES	Advanced Encryption Standard
AFI	Application Family Identifier
AID	Application Identifier
APDU	Application Protocol Data Unit
API	Application Programming Interface
ATR	Answer to Reset
CCID	Chip Card Interface Device
CID	Card ID Number
CRC	Cyclic Redundancy Check
DES	Data Encryption Standard
DFU	Device Firmware Upgrade
DSFID	Data Storage Format Identifier
EAS	Electronic Article surveillance
FCC	Federal Communications Commissions
FWT	Frame Waiting Time
GUI	Graphical User Interface
HF	High Frequency
HID	Human Interface Device
ICC	Integrated Circuit Card
ISO	International Standard Organization
LSB	Least Significant Byte
MIFARE UL	MIFARE Ultralight
MIFARE ULC	MIFARE Ultralight with cryptographic engine
MSB	Most Significant Byte
NACK	Negative Acknowledgement
NFC	Near Field Communication
PC	Personal Computer
PC/SC	Personal Computer/Smart Card
PCD	Proximity Coupling Device
PICC	Proximity integrated circuit card
PIN	Personal Identification Number
RF	Radio Frequency
RFID	Radio Frequency Identification
SLx	Security Level – x
TDES	Triple Data Encryption Standard
TOM	Tag On Metal
UID	Unique Identifier
USB	Universal Serial Bus

Table 16 – Terms and Abbreviations

Appendix B References

[R1] ISO/IEC 15693 Part 3, Identification cards – Contactless integrated circuit(s) cards – Vicinity card(s)
[R2] Interoperability Specification for ICCs and Personal Computer Systems Part 3
[R3] NXP MIFARE® DESFire Datasheet (M075031.pdf)
[R4] Microsoft's PC/SC reference documentation is included in most Visual Studio help system and available online at http://msdn.microsoft.com. Enter "WinSCard" or "SCardTransmit" keywords in the search box.
[R5] PC/SC workgroup: http://www.pcscworkgroup.com/
[R6] ISO/IEC 7816-3 Third Edition 2006-11-01
[R7] ISO/IEC 7816-4 Second Edition 2005-01-15
[R8] ISO/IEC 14443-4 First Edition 2001-02-01
[R9] ISO/IEC 14443-4 Amendment-1 2006-03-15
[R10] Atmel CryptoRF Specification (AT88SCxxxxCRF) Rev 2.0 2007-04-13
[R11] NXP MIFARE® DESFire EV1 Functional Specification (MF3ICD81)
[R12] Philips CL RC632 Multiple Protocol Contactless Reader IC Datasheet Rev 3.0 May 2003
[R13] MF 1PLUSx0y1 Mainstream contactless smart card IC for fast and easy solution development Rev 3.0
[R14] MF 1SPlusx0y1 Mainstream contactless smart card IC for fast and easy solution development Rev 3.1