



R/W Protocol Specification

Added batch function and DESFire Card

DUALi Inc.

Document Version: 1.02

Last Revised Date: 11 Sep 2017

Copyright © 2009 DUALi Inc. All rights reserved. You are strictly prohibited to copy, disclose, distribute, or use this document in part or as a whole for any purposes other than those for which this document is disclosed. This document is copyrighted and contains confidential information and other intellectual property rights of DUALi Inc. Any unauthorized use, copy, disclosure or distribution constitutes infringement of DUALi's intellectual property rights.

DUALi Inc. reserves the right to make changes to its applications or services or to discontinue any application or service at any time without notice. DUALi provides customer assistance in various technical areas, but does not have full access to data concerning the use and applications of customer's products.

Therefore, DUALi assumes no liability and is not responsible for customer applications or software design or performance relating to systems or applications incorporating DUALi products. In addition, DUALi assumes no liability and is not responsible for infringement of patents and/or any other intellectual or industrial property rights of third parties, which may result from assistance provided by DUALi.

Composition of the information in this manual has been done to the best of our knowledge. DUALi does not guarantee the correctness and completeness of the details given in this manual and may not be held liable for damages ensuing from incorrect or incomplete information. Since, despite all our efforts, errors may not be completely avoided, we are always grateful for your useful tips.

We have our development center in South Korea to provide technical support. For any technical assistance can contact our technical support team as below;

Tel: +82 31 213 0074

e-mail : duali@duali.com

Revision History

- 2014.06.05(Ver. 1.00) : First Release
- 2016.02.01(Ver. 1.01) : Release with AES
- 2017.07.26(Ver. 1.02) : Add function for DesFire EV2
-

CONTENTS

1.	INTRODUCTION	5
2.	INTERFACE SPECIFICATION	6
2.1	COMMUNICATION PROTOCOL FRAME FORMAT	6
2.1.1	Command frame format	6
2.1.2	Response frame format	6
2.1.3	RS-232 Protocol (option)	6
3.	COMMAND DEFINITION	7
3.1.1	RF Find Card (Command = 0x4C, _DE_FIND_CARD)	7
3.1.2	DESFire Authentication	8
3.1.2.1	(Command = 0x3A 0x01, _DE_DESFIRE_AUTH)	8
3.1.2.2	(Command = 0x3A 0x05, _DE_DESFIRE_AUTH_AES)	8
3.1.2.3	(Command = 0x3A 0x07, _DE_DESFIRE_AUTH_ASE_FIRST_AUTH)	8
3.1.3	DESFire Transparent (Command = 0x3A 0x02, _DE_DESFIRE_ TRANSPARENT)	9
4.	EXAMPLE	11
4.1.1	Card Authentication	11
4.1.2	Create Application	11
4.1.3	Delete Application.....	13
4.1.4	Create Std Data File (Plain)	14
4.1.5	Create Std Data File (Encrypt)	16
4.1.6	Create Value File	20
4.1.7	Change Key (PICC Master Key)	25
4.1.8	Change Key (ChangeKey)	27
4.1.9	Change Key (Other Key)	28
5.	BATCH COMMAND	30
5.1	DESFire SAVE VALUE COMMAND	30
5.1.1	DESFire Save Value Command with DES (Command = 0x3A 0x03)	30
5.1.2	DESFire Save Value Command with AES (Command = 0x3A 0x06)	30
5.2	DESFire BATCH COMMAND (COMMAND = 0x3A 0x04)	31
5.2.1	2key	31
5.2.2	3key	34

5.2.3	AES	37
6.	RESPONSE CODE DEFINITION.....	41
6.1	RESPONSE FROM READER	41
6.2	RESPONSE FROM DESFIRE CARD	43

1. Introduction

This document defines the USB and serial communication protocol between DUALi's readers and a host computer (Some readers support USB only, but some other readers and modules support RS-232 with same protocol. So this document includes the RS-232 also).

This document defines batch function for DESFire card only.

Duali' readers support TDES, 3KTDES and AES algorithm for DESFire card.

The protocols in this document are all for developers using DESFire Card.

All Duali's reader doesn't have this protocol by default.

So you should check that your device has this protocol before you develop your project.

This document is dedicated for all readers and modules. So, when you use some reader product of DUALi, those readers have a possibility to return code UNKNOWN COMMAND ERROR(23, 0x17), it means your reader or module don't support this command.

DesFire® are registered trademarks of NXP Semiconductors

2. Interface Specification

2.1 Communication Protocol Frame Format

2.1.1 Command frame format

(To show a Hexadecimal number, "0x" is appended to the first of the number)

Name	STX	LEN-H	LEN-L	CMD	Data[n]	LRC
Values	0x02	0xHH	0xHH	0xHH	Data[n]	0xHH
Length.	1-byte	1-byte	1-byte	1-byte	n-byte	1-byte

(STX and LRC are for RS-232. In case of USB, these elements should be removed)

- STX : 0x02
- Data length = command (It must be 1) + Length of data.
 - LEN-H : Higher byte of data length
 - LEN-L : Lower byte of data length
- CMD : Command byte
- Data[n] : Data bytes
- LRC : XORing from LEN-H to Data[n]

2.1.2 Response frame format

Name	STX	LEN-H	LEN-L	Resp	Data[n]	LRC
Values	0x02	0xHH	0xHH	0xHH	Data[n]	0xHH
Length.	1-byte	1-byte	1-byte	1-byte	n-byte	1-byte

(STX and LRC are for RS-232. In case of USB, these elements should be removed)

- STX : 0x02
- Data length = response (it must be 1) + Length of data.
 - LEN-H : Higher byte of data length
 - LEN-L : Lower byte of data length
- Resp : Response code from reader
- Data[n] : Data Bytes
- LRC : XORing from LEN-H to Data[n]

2.1.3 RS-232 Protocol (option)

- Speed : 115200bps (except DE-ABM4 : 57600bps, DE-930/DE-950 :9600bps)
- Data: 8 bit
- Parity : No
- Stop : 1 bit

3. Command Definition

3.1.1 RF Find Card (Command = 0x4C, _DE_FIND_CARD)

This command detects card in the RF field.

■ Command frame

LEN-H	LEN-L	CMD	Data[0]	Data[1]	Data[2]	Data[3]
0x0004		0x4C	BRate	CID	NAD	Option

- Data[0]

Data[0]	Max Card Baud Rate Use same speed for TX and RX.
0x00	0x00 = 106 Kbps
0x01	0x01 = 212 Kbps
0x02	0x02 = 424 Kbps
0x03	0x03 = 848 Kbps
0x04	VHBR, 1.6 ~ 6.8Mbps

- Data[1] : CID

- Data[2] : NAD

- Data[3] : Option ('A': only A-type, 'B': only B-type, 'K': NFC Barcode, other: All)

■ Response frame

LEN-H	LEN-L	Resp	Data[0]	Data[1]	Data[2...N-1]
N+1		OK	Type	CSPD	CD[..],UID[0]...UID[3]

- Data[0]

Data[0]	RF Type
'M', 0x4D, 77	Mifare
'A', 0x41, 65	A type
'B', 0x42, 66	B type

- Data[1]: Card communication speed

- Data[2]: Card Data and Card UID. It is different each card.

- Response: OK(0x00), Err(pls refer to the response code).

■ Response frame for NFC Barcode

LEN-H	LEN-L	Resp	Data[0...N-1]
N+1		OK	16bytes or 32bytes UID

3.1.2 DESFire Authentication

3.1.2.1 (Command = 0x3A 0x01, _DE_DESFIRE_AUTH)

This command performs DESFire authentication with TDES or 3KTDES.

■ Command frame

LEN-H	LEN-L	CMD	Data[0]	Data[1]	Data[2~17] or Data[2~25]
0x0011 or 0x0019		0x3A	0x01	KeyNo	Key

- Data[1] : Key number to perform authentication.

- Data[2~n] : Key to perform authentication.

If your card uses TDES, length of key is 16.

If your card uses 3KTDES, length of key is 24

■ Response frame

LEN-H	LEN-L	Resp	Data[0~15] or Data[0~23]
N+1		OK	Session key

- Data[0 ~ n] : Generated session key from authentication.

- Response: OK(0x00), Err(pls refer to the response code).

3.1.2.2 (Command = 0x3A 0x05, _DE_DESFIRE_AUTH_AES)

This command performs DESFire authentication with AES.

■ Command frame

LEN-H	LEN-L	CMD	Data[0]	Data[1]	Data[2~17]
0x0011 or 0x0019		0x3A	0x05	KeyNo	Key

- Data[1] : Key number to perform authentication.

- Data[2~17] : Key to perform authentication.

If your card uses AES, length of key is 16.

■ Response frame

LEN-H	LEN-L	Resp	Data[0~15]
N+1		OK	Session key

- Data[0 ~ n] : Generated session key from authentication.

- Response: OK(0x00), Err(pls refer to the response code).

3.1.2.3 (Command = 0x3A 0x07, _DE_DESFIRE_AUTH_ASE_FIRST_AUTH)

This command performs first authentication of DesFire EV2 card.

■ Command frame

LEN-H	LEN-L	CMD	Data[0]	Data[1]	Data[2~17]
0x0011 or 0x0019		0x3A	0x07	KeyNo	Key

- Data[1] : Key number to perform authentication.

- Data[2~17] : Key to perform authentication.

If your card uses AES, length of key is 16.

■ Response frame

LEN-H	LEN-L	Resp	Data[0~15]	Data[16~31]
N+1		OK	Session key for MAC	Session key for encryption

- Data[0 ~ n] : Generated session key from authentication.

- Response: OK(0x00), Err(pls refer to the response code).

3.1.3 DESFire Transparent (Command = 0x3A 0x02, _DE_DESFIRE_TRANSPARENT)

This command is used to send data with computed CRC from a host to the card.

Host doesn't need to compute 2-byte CRC. Reader extracts 2-byte CRC data from card response data.

This command makes frame for sending to DESFire card.

You can read/write data MAX 250byte at one time using this command.

■ Command frame

LEN-H	LEN-L	CMD	Data[0]	Data[1]	Data[2]	Data[3~n]
0XXXXX		0x3A	0x02	Flag	CMD len	Data

- Data[1] : Flag for ciphering data and adding CRC.

■ Bit Definitions

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	Add cmac	Check AF	Add CRC32	Add CRC16	Cipher mode

- Bit4 : Set to 0, the device doesn't add cmac and padding

Set to 1, the device adds cmac and padding

⇒ for secure message2 of Desfire EV2

- Bit3 : Set to 0, the device send response data when the card return "AF"

Set to 1, the device doesn't send response data when the card return "AF"

- Bit2 : Set to 0, the device doesn't add crc32 and padding

Set to 1, the device adds crc32 and padding

- Bit1 : Set to 0, the device doesn't add crc16 and padding

Set to 1, the device adds crc16 and padding

- Bit0 : Set to 0, the device doesn't encipher data to transmit

Set to 1, the device encipher data to transmit

- Data[2] : Length of command part.
- Data[3~n] : Data to transmit (from command to data)

EX] protocol to write data (0x1122334455) with encryption. (key length is 24)

Int len = 2;

```
Sbuf[len++] = 0x3A;
Sbuf[len++] = 0x02;
Sbuf[len++] = 0x05; //set bit2 and bit0
Sbuf[len++] = 0x07; command length
// start command part
Sbuf[len++] = 0x3D; //write command of DESFire card
Sbuf[len++] = 0x00; //number of file
Sbuf[len++] = 0x00; //offset high
Sbuf[len++] = 0x00; //offset low
Sbuf[len++] = 0x00; //length to write
Sbuf[len++] = 0x00; //length to write
Sbuf[len++] = 0x05; //length to write (write 3A04yte data)
Sbuf[len++] = 0x11; //data to write
Sbuf[len++] = 0x22; // data to write
Sbuf[len++] = 0x33; // data to write
Sbuf[len++] = 0x44; // data to write
Sbuf[len++] = 0x55; // data to write
Sbuf[0] = len/256;
Sbuf[1] = len%256;
```

■ Response frame

LEN-H	LEN-L	Resp code	Resp data
N+1		OK	

- Resp code: OK(0x00), Err(pls refer to the response code).
- Resp data : Response data of a DESFire card.

4. Example

4.1.1 Card Authentication

■ Condition

- AID : 000000
- keyno : 00,
- key : 00 (3keyDES/24bytes)
key : 00 (AES/16bytes)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000000 : Select Application
 - DESFire Transparent CMD : 3A02
 - Flag : 00
 - CMD len : 04
 - Select Application CMD : 5A
 - AID : 000000
3. 3A0100 : Authenticate
 - DESFire Authentication CMD : 3A01 (3keyDES) / 3A05 (AES) /3A07(AES-for First Authentication of DesFire EV2)
 - Keyno : 00
 - Key : 00 (3keyDES/24bytes)
key : 00 (AES/16bytes)

4.1.2 Create Application

■ Condition

- AID : 000001
- keyno : 00
- key : 00 (3keyDES)
key : 00 (AES/16bytes)
- Key Sett.1 : 0F (default)
- Key Sett.2 : 41 (3K3DES, No 2 byte file identifier, 1 key)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001: Select Application
3. 3A0100 (3keyDES)

3A050000000000000000000000000000 (AES) : Authenticate

4. 3A020006CA0000010F41 : Create Application

- DESFire Transparent CMD : 3A02
- Flag : 00
- CMD len 06
- CreateApplicaication CMD : CA
- AID : 000001
- Key Sett.1 : 0F

Bit 7...4	Change Key Access Hold the Access Rights for changing application keys(ChangeKey command) -0x0: Application master key authentication is necessary to change any key(default) -0x1 .. 0xD: Authentication with the specified key is necessary to change any key. -0xE: Authentication with the key to be changed(same KeyNo) is necessary to change a key. -0xF: All Keys (except application master key, see Bit0) within this application are frozen
Bit 3	Configuration Changeable Codes whether a change of the application master key settings is allowed -0: configuration not changeable anymore (frozen). -1: this configuration is changeable if authenticated with the application master key (default)
Bit 2	Free create / delete without master key Codes whether application master key authentication is needed before CreateFile/DeleteFile -1: CreateFile/DeleteFile is permitted only with application master key authentication. -0: CreateFile/DeleteFile is permitted also without application master key authentication (default)
Bit 1	Free directory list access without master key Codes whether application master key authentication is needed for file directory access -0: Successful application master key authentication is required for executing the GetFileIDs, GetFileSettings and GetKeySettings commands. -1: GetFileIDs, GetFileSettings and GetKeySettings commands succeed independently of a preceding application master key authentication (default)
Bit 0	Allow change master key Codes whether the application master key is changeable -0: Application master key is not changeable anymore (frozen)

- File Size : 0A0000

4.1.4.2 Read

- Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A0100 (3keyDES)
 3A0500 (AES) : Authenticate
4. 3A020008BD000000000A0000 : Read Data
 - DESFire Transparent CMD : 3A02
 - Flag : 00
 - CMD len 08
 - Read CMD : BD
 - FileNo : 00
 - Offset : 000000
 - Length : 0A0000

4.1.4.3 Write

- Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A0100 (3keyDES)
3A0500 (AES) : Authenticate
4. 3A0200083D000000000A000022222222222222222222 : Write Data
 - DESFire Transparent CMD : 3A02
 - Flag : 00
 - CMD len 08
 - Write CMD :3D
 - FileNo : 00
 - Offset : 000000
 - Length : 0A0000
 - Data : 22222222222222222222222222222222

4.1.4.4 Delete

- Command

- Access Right : 0000

- File Size : 0A0000

- Command

- Command

- DUALi Inc. (<http://www.duali.com>)

- Flag : 05
- CMD len 08
- Write CMD :3D
- FileNo : 00
- Offset : 000000
- Length : 0A0000
- Data : 22222222222222222222

4.1.5.4 Read (2Key)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A01000000000000000000000000000000 : Authenticate
4. 3A020308BD000000000A0000 : Read File
 - DESFire Transparent CMD : 3A02
 - Flag : 03
 - CMD len 08
 - Read CMD : BD
 - FileNo : 00
 - Offset : 000000
 - Length : 0A0000

4.1.5.5 Write (2Key)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A01000000000000000000000000000000 : Authenticate
4. 3A0203083D000000000A00002222222222222222 : Write Data
 - DESFire Transparent CMD : 3A02
 - Flag : 03
 - CMD len 08
 - Write CMD :3D
 - FileNo : 00
 - Offset : 000000
 - Length : 0A0000

- Data : 22222222222222222222

4.1.5.6 Read (AES)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A050000000000000000000000000000 : Authenticate
4. 3A020508BD000000000A0000 : Read File
 - DESFire Transparent CMD : 3A02
 - Flag : 05
 - CMD len 08
 - Write CMD :BD
 - FileNo : 00
 - Offset : 000000
 - Length : 0A0000

4.1.5.7 Write (AES)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A050000000000000000000000000000 : Authenticate
4. 3A0205083D000000000A00002222222222222222 : Write Data
 - DESFire Transparent CMD : 3A02
 - Flag : 05
 - CMD len 08
 - Write CMD :3D
 - FileNo : 00
 - Offset : 000000
 - Length : 0A0000
 - Data : 22222222222222222222

4.1.5.8 Read (AES-after First Authentication)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A070000000000000000000000000000 : Authenticate
4. 3A021108BD000000000A0000 : Read File

- DESFire Transparent CMD : 3A02
- Flag : 11
- CMD len 08
- Write CMD :BD
- FileNo : 00
- Offset : 000000
- Length : 0A0000

4.1.5.9 Write (AES-after First Authentication)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A070000000000000000000000000000 : Authenticate
4. 3A0211083D000000000A00002222222222222222 : Write Data
 - DESFire Transparent CMD : 3A02
 - Flag : 11
 - CMD len 08
 - Write CMD :3D
 - FileNo : 00
 - Offset : 000000
 - Length : 0A0000
 - Data : 22222222222222222222

4.1.6 Create Value File

4.1.6.1 Create

■ Condition

- AID : 000001
- keyno : 00,
- key : 00 (3keyDES/24bytes)
- 00 (AES/16bytes)
- FileNo : 01
- Com Set : 03 (Fully DES/3DES enciphered communication)
- Access Rights : 0000 (Access key No 00)
- File Size : 0A0000 (10)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application

3. 3A0100 (3keyDES)

3A0500 (AES) : Authenticate

4. 3A020012CC010300000000000000FFFFF7F2710000001 : Create Value File

- DESFire Transparent CMD : 3A02
- Flag : 00
- CMD len 12
- CreateValueFile CMD : CC
- FileNo : 01
- Com. Setting : 03

00	Plain communication
01	Plain communication secured by DES/3DES MACing
03	Fully DES/3DES enciphered communication

- Access Right : 0000

Byte 3	Change AccessRights Access Key No: 00~0D, 0E(Free Access)
Byte 2	Read&Write Access (GetValue, Debit, LimitedCredit, Credit for value files) Access Key No: 00~0D, 0E(Free Access)
Byte 1	Write Access (GetValue, Debit, LimitedCredit for value files) Access Key No: 00~0D, 0E(Free Access)
Byte 0	Read Access (GetValue, Debit, Debit for value files) Access Key No: 00~0D, 0E(Free Access)

- Lower Limit : 00000000
- Upper Limit : FFFFFFF7F
- Value : 27100000
- Limited Credit Enabled : 01

Bit 0	Enable "LimitedCredit" feature. Here '0' means that "LimitedCredit" is disabled and '1' enables this feature.
Bit 1	Enable "Free GetValue" feature, which allows free read access to the value file. Here '0' means that "Free GetValue" is disabled and '1' enables this feature. If the access rights are set to disable any reading, this feature cannot used.

4.1.6.2 Get Value (3Key)

■ Command

1. 4C00000100 : DetectCard

2. 3A0200045A000001 : Select Application
3. 3A01000 : Authenticate
4. 3A0205026C01 : Get Value
 - DESFire Transparent CMD : 3A02
 - Flag : 05
 - CMD len 02
 - GetValue CMD :6C
 - FileNo : 01

4.1.6.3 Credit (3Key)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A01000 : Authenticate
4. 3A0205020C0110000000 : Credit
 - DESFire Transparent CMD : 3A02
 - Flag : 05
 - CMD len 02
 - Credit CMD :0C
 - FileNo : 01
 - Value : 10000000

4.1.6.4 Debit (3Key)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A01000 : Authenticate
4. 3A020502DC0110000000 : Debit
 - DESFire Transparent CMD : 3A02
 - Flag : 05
 - CMD len 02
 - Debit CMD :DC
 - FileNo : 01
 - Value : 10000000

4.1.6.5 Get Value (2Key)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A01000000000000000000000000000000 : Authenticate
4. 3A0203026C01 : Get Value
 - DESFire Transparent CMD : 3A02
 - Flag : 03
 - CMD len 02
 - GetValue CMD :6C
 - FileNo : 01

4.1.6.6 Credit (2Key)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A01000000000000000000000000000000 : Authenticate
4. 3A0203020C0110000000 : Credit
 - DESFire Transparent CMD : 3A02
 - Flag : 03
 - CMD len 02
 - Credit CMD :0C
 - FileNo : 01
 - Value : 10000000

4.1.6.7 Debit (2Key)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A01000000000000000000000000000000 : Authenticate
4. 3A020302DC0110000000 : Debit
 - DESFire Transparent CMD : 3A02
 - Flag : 03
 - CMD len 02
 - Debit CMD :DC

- FileNo : 01
- Value : 10000000

4.1.6.8 Get Value (AES)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A050000000000000000000000000000 : Authenticate
4. 3A0205026C01 : Get Value
 - DESFire Transparent CMD : 3A02
 - Flag : 05
 - CMD len 02
 - GetValue CMD :6C
 - FileNo : 01

4.1.6.9 Credit (AES)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A050000000000000000000000000000 : Authenticate
4. 3A0205020C0110000000 : Credit
 - DESFire Transparent CMD : 3A02
 - Flag : 05
 - CMD len 02
 - Credit CMD :0C
 - FileNo : 01
 - Value : 10000000

4.1.6.10 Debit (AES)

■ Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A050000000000000000000000000000 : Authenticate
4. 3A020502DC0110000000 : Debit

- DESFire Transparent CMD : 3A02
- Flag : 05
- CMD len 02
- Debit CMD :DC
- FileNo : 01
- Value : 10000000

4.1.6.11 Transaction

- Command

1. 4C00000100 : DetectCard
2. 3A0200045A000001 : Select Application
3. 3A0100 (3keyDES)
3A0500 (AES) : Authenticate
4. 3A020501C7 : Transaction
 - DESFire Transparent CMD : 3A02
 - Flag : 05(3Key) / 01(AES)
 - CMD len 01
 - Transaction CMD : C7

4.1.7 Change Key (PICC Master Key)

If the AID '0x00' is selected, the change applies to the PICC master key. The PICC master key is only able to change the key type. This part handles about changing the key type. If you want to change the key value, please refer to [Change Key \(Change Key\)](#) and [Change Key \(Other Key\)](#). After a successful change of the key used to reach the current authentication status, this authentication is invalidated.

4.1.7.1 DES/3DES Key Type

- Command

- [illegible]

- #### 4.1.8 Change Key (ChangeKey)

[illegible]

- DESFire Transparent CMD: 0x3A02
- Flag: 0x00
- CMD len: 0x02
- Transaction CMD: 0xC4
- KeyNo: 0x01
- KeyVersion: 0x00 (only valid for AES)
- OldKeyLen: 0x18
- NewKeyLen: 0x18
- OldKey: 0x00
- NewKey: 0x11

4.1.9 Change Key (Other Key)

The change key command depends on current authentication and key setting. The key setting allows you to set access rights to change the keys. If the authenticated current key is a ChangeKey, you can change the key of the other KeyNo as well as your own key. Else, you can only change the key of the same Key. The following example is valid if the authenticated current key is a ChangeKey or the KeySetting should be 0xE.

4.1.9.1 DES/TDES, AES Key Type

- Command (authentication key number : #1 , key number to change : #1)

1. 4C00000100: Detect Card
2. 3A0200045A000001: Select Application
3. 3A0100 (DES/3DES)
Or 3A0500 (AES): Authenticate
4. 3A020002C40100001011: Change Key
 - DESFire Transparent CMD: 0x3A02
 - Flag: 0x00
 - CMD len: 0x02
 - Transaction CMD: 0xC4
 - KeyNo: 0x01
 - KeyVersion: 0x00 (only valid for AES)
 - OldKeyLen: 0x00
 - NewKeyLen: 0x10
 - NewKey: 0x11

4.1.9.2 3K3DES Key Type

- Command (authentication key number : #1 , key number to change : #1)

1. 4C00000100: DetectCard

- [illegible]

5. Batch Command

5.1 DESFire Save Value Command

5.1.1 DESFire Save Value Command with DES (Command = 0x3A 0x03)

This command save AID, key(2key or 3key), key number data in the reader before you call read or write command.

When you call this command the reader don't send any command to the card.

■ Command frame

CMD	Data[0]	Data[1~3]	Data[4~19] or Data[4~27]	Data[20] or Data[28]
0x3A	0x03	AID	Key	keyno

- Data[1~3] : AID (application number)
- Data[4~19] or Data[4~27]: key(2key or 3key) Key to perform authentication.
- Data[20] or Data[28]: Key number to perform authentication.

If your card uses TDES, length of key is 16.

If your card uses 3KTDES, length of key is 24

■ Response frame

Resp
0x00

- RESP : 0x00 is OK. Err(Please refer to 6. Response Code Definition.)

■ Example Command

3A 03 000001 00000000000000000000000000000000 00 (2key 21byte)

3A 03 000001 00000000000000000000000000000000 00 (3key 29byte)

5.1.2 DESFire Save Value Command with AES (Command = 0x3A 0x06)

This command save AID, key(AES), key number data.

■ Command frame

CMD	Data[0]	Data[1~3]	Data[4~19]	Data[20]
0x3A	0x06	AID	Key	keyno

- Data[1~3] : AID (application number)
- Data[4~19] : 16bytes AES Key to perform authentication.
- Data[18] or Data[26]: Key number to perform authentication.

If your card uses AES, length of key is 16.

■ Response frame

Resp
0x00

- RESP : 0x00 is OK. Err(Please refer to 6. Response Code Definition.)

■ Example Command

3A 06 000001 00000000000000000000000000000000 00 (AES 21bytes)

5.2 DESFire Batch Command (Command = 0x3A 0x04)

You can read/write/credit/debit/read value without you call find the card, select applications and authentication.

If you call this command, the reader run from find the card to send read/write/credit/debit/read value command at one time.

You should call this command after you call 3A03 command or 3A06 command.

This batch command can read/write data MAX 250byte at one time.

Please refer to the part [3.1.3](#) for check Flag

5.2.1 2key

5.2.1.1 Standard file read

■ Command frame

CMD	Data [0]	Data [1]	Data [2]	Data [3]	Data [4~6]	Data [7~9]
0x3A	0x04	Flag	0xBD	fileno	offset	filesize

- Data [1] : flag (1byte)

- Data [2] : std file read command (1byte)

- Data [3] : fileno(1byte)

- Data [4~6] : offset(3byte)

- Data [7~9] : filesize(3byte)

■ Response frame

Resp1	Resp2	Data
0x00	0x00	Read Data[filesize]

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code Definition.](#)))

- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#)))

- Data : Read Data[filesize].

■ Example Command

3A 04 00 3D 00 000000 0A0000 (10byte)

5.2.1.2 Standard file write

■ Command frame

CMD	Data [0]	Data [1]	Data [2]	Data [3]	Data [4~6]	Data [7~9]	Data [10~n]
0x3A	0x04	flag	0x3D	fileno	offset	filesize	data

- Data [1] : flag(1byte)
- Data [2] : std file read command(1byte)
- Data [3] : fileno(1byte)
- Data [4~6] : offset(3byte)
- Data [7~9] : filesize(3byte)
- Data [10~n] : data(filesize의 크기)

- ■ Response frame

Resp1	Resp2
0x00	0x00

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code Definition.](#)))
- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#)))

■ Example Command

3A 04 00 3D 00 000000 0A0000 33333333332222224fff (20byte)

5.2.1.3 Value file credit

Automatic transaction after the credit operation.

■ Command frame

CMD	Data [0]	Data[1]	Data[2]	Data[3]	Data[4~7]
0x3A	0x04	flag	0x0C	fileno	Value

- Data[1] : flag (1byte)
- Data[2] : value file credit command (1byte)

- Data[3] : fileno(1byte)
- Data[4~7] : value(4byte)

■ Response frame

Resp1	Resp2
0x00	0x00

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code Definition.](#)))
- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#)))

■ Example Command

3A 04 00 0C 02 10000000 (8byte)

5.2.1.4 Value file debit

Automatic transaction after the debit operation.

■ Command frame

CMD	Data [0]	Data[1]	Data[2]	Data [3]	Data [4~7]
0x3A	0x04	flag	0xDC	fileno	value

- Data [1] : flag (1byte)
- Data [2] : value file credit command (1byte)
- Data [3] : fileno(1byte)
- Data [4] : value(4byte)

■ Response frame

Resp1	Resp2
0x00	0x00

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code Definition.](#)))
- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#)))

■ Example Command

3A04 00 DC 02 10000000 (8byte)

5.2.1.5 Value file read

■ Command frame

CMD	Data [0]	Data [1]	Data [2]	Data [3]
0x3A	0x04	flag	0x6C	fileno

- Data [1] : flag (1byte)
- Data [2] : value file read command (1byte)
- Data [3] : fileno(1byte)

■ Response frame

Resp1	Resp2	Data
0x00	0x00	Value Data

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code Definition.](#)))
- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#)))
- Data – Value Data

■ Example Command

3A 04 00 6C 02(4byte)

5.2.2 3key

Fully DES/3DES enciphered Communication mode

Depending on the communication mode should replace the flag.

5.2.2.1 Standard file read

■ Command frame

CMD	Data [0]	Data [1]	Data [2]	Data [3]	Data [4~6]	Data [7~9]
0x3A	0x04	flag	0xBD	fileno	offset	Filesize

- Data [1] : flag (1byte)
- Data [2] : std file read command (1byte)
- Data [3] : fileno(1byte)
- Data [4~6] : offset(3byte)
- Data [7~9] : filesize(3byte)

■ Response frame

Resp1	Resp2	Data
0x00	0x00	Read Data[filesize]

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code Definition.](#)))
- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#)))
- Data : Read Data[filesize].

■ Example Command

3A 04 05 3D 00 000000 0A0000 (10byte)

5.2.2.2 Standard file write

■ Command frame

CMD	Data [0]	Data [1]	Data [2]	Data [3]	Data [4~6]	Data [7~9]	Data [10~n]
0x3A	0x04	flag(1)	0x3D	fileno	offset	filesize	data

- Data [1] : flag (1byte)
- Data [2] : std file read command (1byte)
- Data [3] : fileno(1byte)
- Data [4~6] : offset(3byte)
- Data [7~9] : filesize(3byte)
- Data [10~n] : data(filesize의 크기)

■ Response frame

Resp1	Resp2
0x00	0x00

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code Definition.](#)))
- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#)))

■ Command

3A04 05 3D 00 000000 0A0000 33333333332222224fff (20byte)

5.2.2.3 value file credit

Automatic transaction after the credit operation.

■ Command frame

CMD	Data [0]	Data[1]	Data[2]	Data[3]	Data[4~7]
-----	----------	---------	---------	---------	-----------

0x3A	0x04	flag	0x0C	fileno	value
------	------	------	------	--------	-------

- Data[1] : flag (1byte)
- Data[2] : value file credit command (1byte)
- Data[3] : fileno(1byte)
- Data[4~7] : value(4byte)

■ Response frame

Resp1	Resp2
0x00	0x00

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code Definition.](#)))
- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#)))

■ Example Command

3A04 05 0C 02 10000000 (8byte)

5.2.2.4 value file debit

Automatic transaction after the debit operation.

■ Command frame

CMD	Data [0]	Data[1]	Data[2]	Data [3]	Data [4~7]
0x3A	0x04	flag	0xDC	fileno	value

- Data [1] : flag (1byte)
- Data [2] : value file credit command (1byte)
- Data [3] : fileno(1byte)
- Data [4~7] : value(4byte)

■ Response frame

Resp1	Resp2
0x00	0x00

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code Definition.](#)))
- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#)))

■ Example Command

3A04 05 DC 02 10000000 (8byte)

5.2.2.5 value file read

■ Command frame

CMD	Data [0]	Data [1]	Data [2]	Data [3]
0x3A04	0x04	flag	0x6C	fileno

- Data [1] : flag (1byte)
- Data [2] : value file read command (1byte)
- Data [3] : fileno(1byte)

■ Response frame

Resp1	Resp2	Data
0x00	0x00	Value Data

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code Definition.](#)))
- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#)))
- Data – Value Data

■ Example Command

3A04 05 6C 02(4byte)

5.2.3 AES

Fully AES enciphered Communication mode

Depending on the communication mode should replace the flag.

5.2.3.1 Standard file read

■ Command frame

CMD	Data [0]	Data [1]	Data [2]	Data [3]	Data [4~6]	Data [7~9]
0x3A	0x04	flag	0xBD	fileno	offset	Filesize

- Data [1] : flag (1byte)
- Data [2] : std file read command (1byte)
- Data [3] : fileno(1byte)
- Data [4~6] : offset(3byte)
- Data [7~9] : filesize(3byte)

■ Response frame

Resp1	Resp2	Data
0x00	0x00	Read Data[filesize]

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code Definition.](#)))
- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#)))
- Data : Read Data[filesize].

■ Example Command

3A 04 05 3D 00 000000 0A0000 (10byte)

5.2.3.2 Standard file write

■ Command frame

CMD	Data [0]	Data [1]	Data [2]	Data [3]	Data [4~6]	Data [7~9]	Data [10~n]
0x3A	0x04	flag(1)	0x3D	fileno	offset	filesize	data

- Data [1] : flag (1byte)
- Data [2] : std file read command (1byte)
- Data [3] : fileno(1byte)
- Data [4~6] : offset(3byte)
- Data [7~9] : filesize(3byte)
- Data [10~n] : data(filesize의 크기)

■ Response frame

Resp1	Resp2
0x00	0x00

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code Definition.](#)))
- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#)))

■ Example Command

3A04 05 3D 00 000000 0A0000 33333333332222224fff (20byte)

5.2.3.3 value file credit

Automatic transaction after the credit operation.

■ Command frame

CMD	Data [0]	Data[1]	Data[2]	Data[3]	Data[4~7]
0x3A	0x04	flag	0x0C	fileno	value

- Data[1] : flag (1byte)
- Data[2] : value file credit command (1byte)
- Data[3] : fileno(1byte)
- Data[4~7] : value(4byte)

■ Response frame

Resp1	Resp2
0x00	0x00

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code Definition.](#)))
- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#)))

■ Example Command

3A04 05 0C 02 10000000 (8byte)

5.2.3.4 value file debit

Automatic transaction after the debit operation.

■ Command frame

CMD	Data [0]	Data[1]	Data[2]	Data [3]	Data [4~7]
0x3A	0x04	flag	0xDC	fileno	value

- Data [1] : flag (1byte)
- Data [2] : value file credit command (1byte)
- Data [3] : fileno(1byte)
- Data [4~7] : value(4byte)

■ Response frame

Resp1	Resp2
0x00	0x00

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code](#)

Definition.)

- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#))

■ Example Command

3A04 05 DC 02 10000000 (8byte)

5.2.3.5 value file read

■ Command frame

CMD	Data [0]	Data [1]	Data [2]	Data [3]
0x3A04	0x04	flag	0x6C	fileno

- Data [1] : flag (1byte)
- Data [2] : value file read command (1byte)
- Data [3] : fileno(1byte)

■ Response frame

Resp1	Resp2	Data
0x00	0x00	Value Data

- Resp1 : Response from Reader. (0x00 is OK. Err([Please refer to 6.1 Response Code Definition.](#))
- Resp2 : Response from Card. (0x00 is OK. Err([Please refer to 6.2 Response Code Definition.](#))
- Data – Value Data

■ Example Command

3A04 05 6C 02(4byte)

6. Response Code Definition

6.1 Response from Reader

Received correct response from card	: 0 (0x00)
No response from card	: 2 (0x02)
Wrong CRC was transmitted from card	: 3 (0x03)
Smart card is not inserted	: 4 (0x04)
MIFARE card key authentication error	: 5 (0x05)
Smart card is in turned off state	: 5 (0x05)
Wrong parity bit was transmitted from type-A card	: 6 (0x06)
Wrong command code was transmitted from host	: 7 (0x07)
Check byte of UID is wrong	: 8 (0x08)
This command is not authenticated	: 10 (0x0A)
Bit count of received data is incorrect	: 11 (0x0B)
More or less data from protocol was sent from card	: 12 (0x0C)
Error occurred when it write data to MIFARE card	: 15 (0x0F)
Error occurred when it increment data to MIFARE card	: 16 (0x10)
Error occurred when it decrement data to MIFARE card	: 17 (0x11)
Error occurred when it read FeliCa card	: 18 (0x12)
FIFO was overflowed when receive data form card	: 19 (0x13)
Received data is out of correct frame (protocol)	: 21 (0x15)
Unsupported command was sent from host	: 23 (0x17)
Collision was detected when receive data from card	: 24 (0x18)
Communication with RF chip is unable, RF chip error	: 26 (0x19)
Chaining retry overflowed limited count	: 33 (0x21)
ACK was received for deselect command	: 34 (0x22)
Retry count overflowed maximum limit	: 35 (0x23)
Receive buffer is smaller than expected data length from card	: 49 (0x31)
More data than receive buffer was transmitted from card	: 50 (0x32)
Wrong data from protocol was received or transmitted	: 52 (0x34)
RF command is not supported when contact card exists	: 64 (0x40)
Other response was received when ACK is supposed from card	: 65 (0x41)
NAK was received when waiting for other response from card	: 66 (0x42)
Temperature error	: 78 (0x4C)
Action for this command is not implemented yet	: 100(0x64)
Error occurred when write data to FIFO	: 109(0x6D)

Data beyond limit was sent from host	: 123(0x7B)
Read value from card is different from written value	: 123(0x7B)
Card responded value error for the command	: 124(0x7C)

(NFC Related Codes)

RF is not in ready state to communicate with card (NFC)	: 51 (0x33)
Invalid data from protocol format was sent from host (NFC)	: 55 (0x37)
Wrong parameter from protocol format was sent from host (NFC)	: 60 (0x3C)
Invalid parameter from protocol format was sent from host (NFC)	: 61 (0x3D)
Unsupported NFC command was sent from host (NFC)	: 63 (0x3F)
User mode values and register mode values are different	: 0x90
User speed and register speed are different	: 0x91
Protocols between two nfc devices are abnormal	: 0x92
There is no response until time-out value	: 0x93
Transmission failed until retry number is over 3 times	: 0x94
Scope data value is out of ranges defined in the specifications	: 0x95
Trying to connect each other during disconnected mode	: 0x96
Two NFC devices are using different DID settings	: 0x97
Target didn't get Information PDU during Get_Target Command	: 0x98
MAC activation error when connect LLC communication	: 0xA0
There is no response to LLC connect command	: 0xA1
There is no response to LLC symmetry command	: 0xA2
There is no response to LLC information command	: 0xA3
There is no response to LLC receive ready command	: 0xA4
There is no response to LLC disconnect command	: 0xA5
MAC deactivation error when close LLC	: 0xA6
Access Conditions, Data Size and Version are not different from defined value	: 0xA7
Checksum values in Attribute Block of Type 3 tag are wrong	: 0xA8
There is no data from remote LLC in a target mode	: 0xA9
There is error during sending PDU to remote LLC	: 0xAA
Magic Number Error	: 0xAB
Length error in LLC PDU format	: 0xAC

6.2 Response from DESFire Card

Table 11. Coding of status- and error codes

Hex code	Status	Description
0x00	OPERATION_OK	Successful operation
0x0C	NO_CHANGES	No changes done to backup files, CommitTransaction / AbortTransaction not necessary
0x0E	OUT_OF_EEPROM_ERROR	Insufficient NV-Memory to complete command
0x1C	ILLEGAL_COMMAND_CODE	Command code not supported
0x1E	INTEGRITY_ERROR	CRC or MAC does not match data Padding bytes not valid
0x40	NO_SUCH_KEY	Invalid key number specified
0x7E	LENGTH_ERROR	Length of command string invalid
0x9D	PERMISSION_DENIED	Current configuration / status does not allow the requested command
0x9E	PARAMETER_ERROR	Value of the parameter(s) invalid
0xA0	APPLICATION_NOT_FOUND	Requested AID not present on PICC
0xA1	APPL_INTEGRITY_ERROR	Unrecoverable error within application, application will be disabled [1]
0xAE	AUTHENTICATION_ERROR	Current authentication status does not allow the requested command
0xAF	ADDITIONAL_FRAME	Additional data frame is expected to be sent
0xBE	BOUNDARY_ERROR	Attempt to read/write data from/to beyond the file's/record's limits. Attempt to exceed the limits of a value file.
0xC1	PICC_INTEGRITY_ERROR	Unrecoverable error within PICC, PICC will be disabled [1]
0xCA	COMMAND_ABORTED	Previous Command was not fully completed Not all Frames were requested or provided by the PCD
0xCD	PICC_DISABLED_ERROR	PICC was disabled by an unrecoverable error [1]
0xCE	COUNT_ERROR	Number of Applications limited to 28, no additional CreateApplication possible
0xDE	DUPLICATE_ERROR	Creation of file/application failed because file/application with same number already exists
0xEE	EEPROM_ERROR	Could not complete NV-write operation due to loss of power, internal backup/rollback mechanism activated [1]
0xF0	FILE_NOT_FOUND	Specified file number does not exist
0xF1	FILE_INTEGRITY_ERROR	Unrecoverable error within file, file will be disabled [1]

[1] These errors are not expected to appear during normal operation.