JMY600 Series IC Card Module

MIFARE DESfire Card Operation Guide

（Revision 1.00）

Jinmuyu Electronics Co., LTD

April 7, 2015

Please read this manual carefully before using. If any problem, please feel free to contact us, we will offer a satisfied answer ASAP.

Contents

1 Overview

This file describes how to operate MIFARE DESfire card and the sequence via using JMY600 Series RFID module. It is suitable for the programmers who are using it to do the development.

Any questions during the programming, please feel free to contact our technical support via jinmuyu@vip.sina.com.

2 Features

- Mutual three pass authentication
- Hardware DES/3DES Data encryption on RF-channel with replay attack protection using 56/112 bit Keys featuring key versioning
- 2K/4K/8K byte NV-Memory
- Unique 7 Byte serial number for each device
- True deterministic anticollision
- Flexible file system
- Up to 28 applications simultaneously on one PICC
- Data retention of 10 years
- Write endurance 100 000 cycles
- Operating temperature: -20 ~ 50 ℃
- Operating frequency: 13.56 MHz
- Fast data transfer: 106 kbit/s
- Operating distance: Up to 100 mm (depending on antenna geometry)

3 General Description

NXP has developed the MIFARE DESFire to be used with Proximity Coupling Devices (PCDs) according to ISO14443 Type A. The communication protocol complies to part ISO 14443-4. The MF3 IC D40 is primarily designed for secure contactless transport applications and related loyalty programs.

In addition to ISO 14443 DESFire also support the use of ISO 7816-3 compliant APDU message structure.

4 Memory Organisation

The 2/4/8 KB NV memory is organized using a flexible file system. This file system allows a maximum of 28 different applications on one MIFARE DESFire EV1. Each application provides up to 32 files. Every application is represented by its 3 bytes Application IDentifier (AID).

Before using the new blank DESFire card, the first thing is to create new applications on the PICC. An application is identified by an 'Application Identifier', AID, which is implemented as a 24 bit number. Application Identifier 0x00 00 00 is reserved as a reference to the PICC itself.

In order to explain how to create new applications on a new PICC, we give the following example for reference.

Master key: 16bytes (0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00).

AID = 0x4A4D59.

GetKeySettings:0F(0x0F), the GetKeySettings command allows to get configuration information on the PICC and application master key configuration settings as described in Section 8.3.2(MF3 IC D40 Contactless). In addition it returns the maximum number of keys which can be stored within the selected application.

Number of Keys: 0C (0x0C), 12.

Create a new application in this AID:

Value File ID = 01; Each File ID is coded in one byte and is in the range from 0x00 to 0x0F.

Communication mode: 00 (0x00), Plain communication.
File Access rights: 11 11 (0x1111), Key1 with full Read&Write Access.
Min. Value: 00 00 00 00 (0x00000000), Signed int.
Max. Value: FF FF FF 7F (0x7FFFFFFF), Signed int.
Initial Value: 78 56 34 12（0x12345678).
Limited Credit: 00 (0x00); Allows a limited increase of a value stored in a Value File without having full Credit permissions to the file.

STD DATA FILE ID = 02; It is used to create files for the storage of plain unformatted user data within an existing application on the PICC.

Communication mode: 00 (0x00), Plain communication.
File Access rights: 11 11 (0x1111), Key1 with full Read&Write Access.
File Size: 00 01 00 (0x000100), 256bytes.

BACKUP DATA FILE ID = 03; It is used to create files for the storage of plain unformatted user data within an existing application on the PICC, additionally supporting the feature of an integrated backup mechanism.

Communication mode: 00 (0x00), Plain communication.
File Access rights: 11 11 (0x1111), Key1 with full Read&Write Access.
File Size: 00 01 00 (0x000100), 256bytes.

LINER RECORD FILE ID = 04; It is used to create files for multiple storage of structural data, for example for loyalty programs, within an existing application on the PICC. Once the file is filled completely with data records, further writing to the file is not possible unless it is cleared, see command ClearRecordFile, Section 8.6.9(MF3 IC D40 Contactless).

Communication mode: 00 (0x00), Plain communication.

File Access rights: 11 11 (0x1111), Key1 with full Read&Write Access.
Single Record Length: 10 00 00 (0x000010), 16bytes.
Total Record Numbers: 10 00 00 (0x000010), 16 records.

LINER RECORD FILE ID = 04; It is used to create files for multiple storage of structural data, for example for logging transactions, within an existing application on the PICC. Once the file is filled completely with data records, the PICC automatically overwrites the oldest record with the latest written one. This wrap is fully transparent for the PCD.

Communication mode: 00 (0x00), Plain communication.
File Access rights: 11 11 (0x1111), Key1 with full Read&Write Access.
Single Record Length: 10 00 00 (0x000010), 16bytes.
Total Record Numbers: 10 00 00 (0x000010), 16 records.

5 Card Operation

5.1 Active Mode

Under this working mode, the reader module just output card Serial Number. For DESFire Card, we don't recommend to use.

5.2 Passive Mode

During operating the DESFire card, the module auto-detecting card function must be shut down. For the multi-card operation function, the user may choose according to the needs.

Put a new DESFire card into antenna RF fields, sending the commands step by step like the following:

- Request Card according to EMV and PBOC:
  TransPort input: 32
  Host sends: 00 04 00 32 36
  Success: 00 0E 00 20 BD 32 30 63 35 D8 9F 04 00 08 8C
- PICC MasterKeyAuthentication:
  TransPort input: 90 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  Host sends: 00 15 00 90 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 85
  Success: 00 15 01 90 00 CC 6C E1 74 46 42 09 8D 1B 78 17 03 49 4C 67 A1 85
  Info: "CC 6C E1 74 46 42 09 8D 1B 78 17 03 49 4C 67 A1" is the SenssionKey (16 bytes). The senssion key will be sent back only after a successful authentication. The senssion key will be used in the following card operations. It is the key to decrypt the encrypted data in encrypted communication process.
- DeleteApplication:
  TransPort input: 99
  Host sends: 00 04 00 99 9D
  Success: 00 05 01 99 00 9D

Permanently deactivates applications on the PICC.

- CreateApplication:
  TransPort input: 95 4A 4D 59 0F 0C
  Host sends: 00 09 00 95 4A 4D 59 0F 0C C1
  Success: 00 05 01 95 00 91
- SelectApplication:
  TransPort input: 98 4A 4D 59
  Host sends: 00 07 00 98 4A 4D 59 C1
  Success: 00 05 01 98 00 9C
- ApplicationKeyAuthentication:
  TransPort input: 90 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  Host sends: 00 15 00 90 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 85
  Success: 00 15 01 90 00 E1 74 46 42 09 8D 1B CC 6C 78 17 03 49 4C 67 A1 85
- CreateValueFile:
  TransPort input: A0 01 00 11 11 00 00 00 00 FF FF FF 7F 78 56 34 12 00
  Host sends: 00 15 00 A0 01 00 11 11 00 00 00 00 FF FF FF 7F 78 56 34 12 00 3C
  Success: 00 05 01 A0 00 A4
- CreateStdDataFile:
  TransPort input: 9E 02 00 11 11 00 01 00
  Host sends: 00 0B 00 9E 02 00 11 11 00 01 00 96
  Success: 00 05 01 9E 00 9A
- CreateBackupDataFile:
  TransPort input: 9F 03 00 11 11 00 01 00
  Host sends: 00 0B 00 9F 03 00 11 11 00 01 00 96
  Success: 00 05 01 9F 00 9B
- CreateLinearRecordFile:
  TransPort input: A1 04 00 11 11 10 00 00 10 00 00
  Host sends: 00 0E 00 A1 04 00 11 11 10 00 00 10 00 00 AB
  Success: 00 05 01 A1 00 A5
- CreateCyclicRecordFile:
  TransPort input: A2 05 00 11 11 10 00 00 10 00 00
  Host sends: 00 0E 00 A2 05 00 11 11 10 00 00 10 00 00 A9
  Success: 00 05 01 A2 00 A6

Hereto, the CreateApplication in the PICC is finished. Then we will do some application testings:

- Request Card according to EMV and PBOC:
  Please reference the above
- SelectApplication:
  TransPort input: 98 4A 4D 59
  Host sends: 00 07 00 98 4A 4D 59 C1
  Success: 00 05 01 98 00 9C
- ApplicationKey1Authentication (Note: Key1 is the MasterKey for the file that we built.)
  TransPort input: 90 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  Host sends: 00 15 00 90 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 84
  Success: 00 15 01 90 00 E1 74 46 42 6C 78 17 03 49 4C 09 8D 1B CC 67 A1 85

- GetValue:

  TransPort input: A6 01

  Host sends: 00 05 00 A6 01 A2

  Success: 00 0D 01 A6 00 E9 87 7E 14 CC 04 BE 51 89

  During the Value operation, if the Value File need be authenticated, then the communication process need be used ciphertext. In here, we need use the SenssionKey (E1 74 46 42 6C 78 17 03 49 4C 09 8D 1B CC 67 A1) to decipher the receiving ciphertext data (E9 87 7E 14 CC 04 BE 51) via DESfireTool that we offered.The Value is 78 56 34 12 F6 5E 00 00. The previous 4byes (78 56 34 12(0x12345678)) is the Value. The following 2bytes (F6 5E) is CRC checksum

- Credit:

  To increase a value (0x00000001) stored in a Value File, then we need input 01000000(LSB first) in DataIn of DESfire Tool. Meantime don't choose 3DES, but choose CRC, and then hitting encryption button. At last we get 8DB1D85CA753132B.

  TransPort input: A7 01 8D B1 D8 5C A7 53 13 2B

  Host sends: 00 0D 00 A7 01 8D B1 D8 5C A7 53 13 2B DF

  Success: 00 05 01 A7 00 A3

- GetValue:

  TransPort input: A6 01

  Host sends: 00 05 00 A6 01 A2

  Success: 00 0D 01 A6 00 F2 51 6A 78 63 10 E8 84 04

  Using the SenssionKey (E1 74 46 42 6C 78 17 03 49 4C 09 8D 1B CC 67 A1) to decipher the receiving ciphertext data (F2 51 6A 78 63 10 E8 84) via DESfireTool that we offered.The Value is 79 56 34 12 4D 42 00 00. The previous 4byes (79 56 34 12(0x12345678)) is the Value. The following 2bytes (4D 42) is CRC checksum

- Write data into BackupDataFiles03:

  TransPort input: A5 03 00 00 00 08 00 00 01 02 03 04 05 06 07 08

  Host sends: 00 13 00 A5 03 00 00 00 08 00 00 01 02 03 04 05 06 07 08 B5

  Success: 00 05 01 A5 00 A1

- Read data out from BackupDataFiles03:

  TransPort input: A4 03 00 00 00 08 00 00

  Host sends: 00 0B 00 A4 03 00 00 00 08 00 00 A4

  Success: 00 0D 01 A4 00 00 00 00 00 00 00 00 00 A8

  Note: The Readout data is not the data to be written, so the changing is invalid.

- Validate the Backup Data Files within one application:

  TransPort input: AD

  Host sends: 00 04 00 AD A9

  Success: 00 05 01 AD 00 A9

- Read data out from BackupDataFiles03:

  TransPort input: A4 03 00 00 00 08 00 00

  Host sends: 00 0B 00 A4 03 00 00 00 08 00 00 A4

  Success: 00 0D 01 A4 00 01 02 03 04 05 06 07 08 A8

  Note: The Readout data is the data to be written, so the changing is valid.

