

7003 Mifare Manual

1. DOCUMENT VERSION	2
2. BUILD DEVELOPMENT ENVIRONMENT	2
2.1 ECLIPSE	2
2.1.1 Load dynamic library	2
2.1.2 Load jar file	3
2.2 ANDROID_STUDIO	3
2.2.1 Load dynamic library	3
2.2.2 Load jar file	4
3. USE MIFARE	5
3.1 INTRODUCE MIFARE	5
3.2 CREATE MIFARE	6
3.3 OPEN MIFARE	6
3.4 CLOSE MIFARE	6
4. MIFARE INTERFACE FUNCTION	6
4.1 FUNCTION LIST	7
4.2 FUNCTION DESCRIPTION	7
4.2.1 open	7
4.2.2 clsoe	8
4.2.3 getVersion	9
4.2.4 seek	9
4.2.5 verifyMifareCard	10
4.2.6 writeMifareCard	11
4.2.7 readMifareCard	12

1. Document version

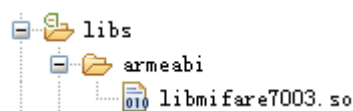
Data	version	author	audit	note
2016-6-1	1.0	chen	edasion	First version
2016-6-27	1.1	chen	edasion	add build development environment of android studio

2. Build development environment

2.1 Eclipse

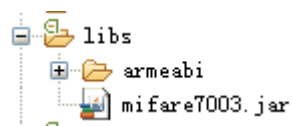
2.1.1 Load dynamic library

In your Android Project, In the libs folder to create a armeabi folder,put libmifare7003.so in,as shown in the figure.



2.1.2 Load jar file

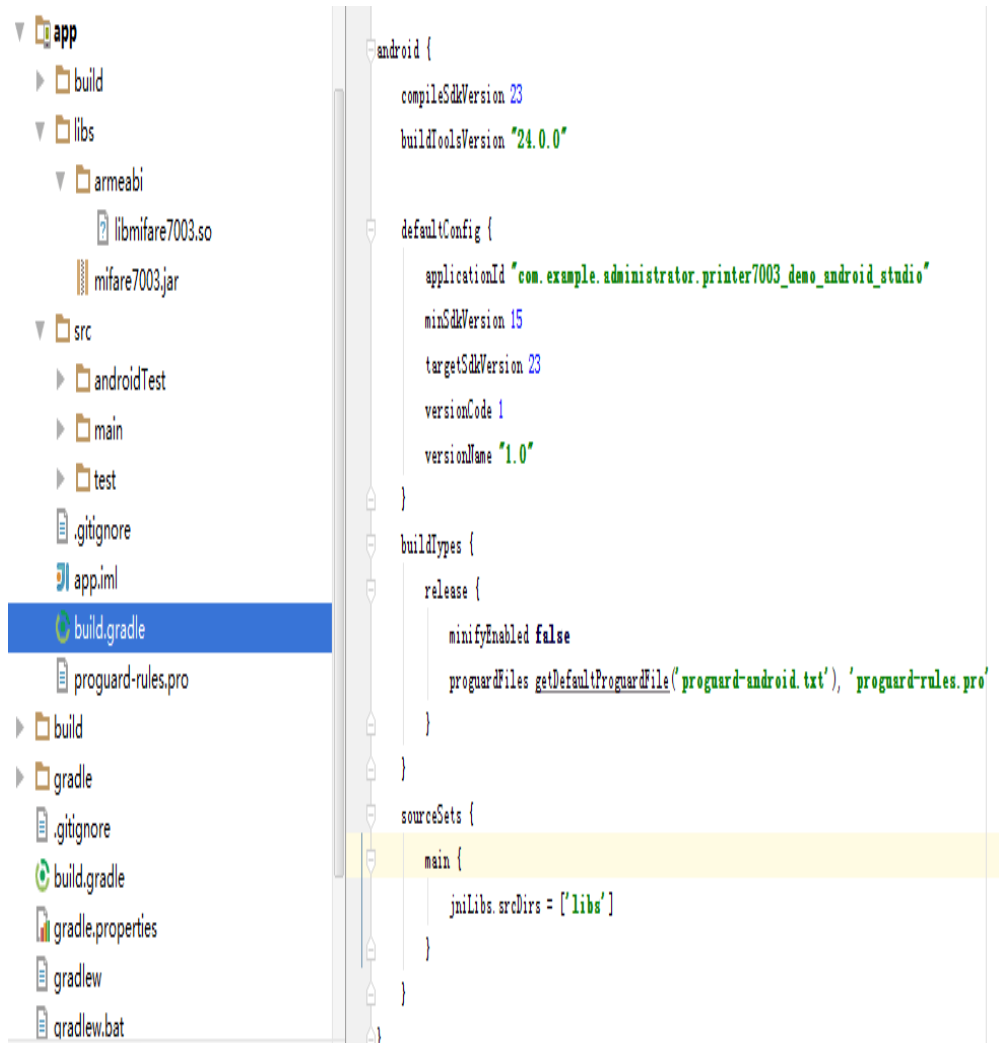
Put mifare7003.jar into the libs, as shown in the figure.



2.2 Android_studio

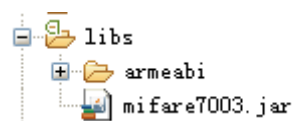
2.2.1 Load dynamic library

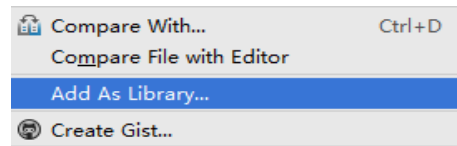
In your Android Project, In the libs folder to create a armeabi folder,put libmifare7003.so in, and in build.gradle file add sourceSets { main {jniLibs.srcDirs=['libs'] } }, as shown in the figure.



2.2.2 Load jar file

Put mifare7003.jar into the libs, in mifare7003.jar
right click the mouse and Select menu -> add as library jar
file,as shown in the figure.





3. Use Mifare

3.1 Introduce Mifare

Before use a Mifare card, You must have certain knowledge of Mifare card. Mifare card is generally divided into several sectors, each sector is divided into several blocks. We can read or write block only. To S50 mifare card as example. It has 16 sectors, each sector has 4 blocks, first 3 area is data area (0, 1, 2 block is data area), after 1 area is password area (3 block is password area), each block has 16 byte. So, its capacity is 1 k, 64 block, 0~63 (3, 7, 11, 15, 19, 23, 27, 31... 63 block is password area). But we only can read or write data area. Before write or read data area, you must verify password area, if verify fail, you can not do anything. One thing is very important, You only have three times the opportunity to verify password, if all verify failed, the sector will be never locked. So, you want read or write data to the specified block of the mifare card, verify first. Then, you can read and write data to the specified block.

3.2 Create Mifare

```
Mifare mifare = new Mifare ();
```

3.3 open Mifare

mifare.open(); if this operation is successful,you can read data from mifare by mifare API.

Step1: seek mifare,if seek successful,you can continue next setp.

Step2: verify mifare,if verify password successful,you can read or write.

Step3: write data to mifare data area.

Step4: read data from mifare data area.

3.4 Close Mifare

One thing is very important,if you no longer use the mifare,you must call mifare.close() to close mifare,or it will consume electricity and reduced life.

4. Mifare interface function

4.1 Function list

In mifare7003.jar,it provides some function interface to you
use mifare,refer to following table:

no	name	description
1	open	Open iccard
2	close	Close iccard
3	getVersion	Get version
4	seek	Seek mifare card
5	verifyMifareCard	Verify password to mifare
6	writeMifareCard	Write data to mifare card
7	readMifareCard	Read data from mifare card

4.2 function description

4.2.1 open

function:

```
public int open();
```

description:

Open mifare card

In parameter:

no

out parameter:

no

return value:

-1: fail

0: success

4.2.2 clsoe

function:

```
public int close();
```

description:

Close mifare card.

in parameter:

no

out parameter:

no

return value:

-1: fail

0: success

4.2.3 getVersion

function:

```
static public String getVersion()
```

description:

Get version of this SDK,note: this
function is static

in parameter:

no

out parameter:

no

return value:

String: SDK version

4.2.4 seek

function:

```
public int seek ();
```

description:

seek mifare card.

in parameter:

no

out parameter:

no

return value:

0: success

-1: fail

-2: time out

4.2.5 verifyMifareCard

function:

```
public int verifyMifareCard (int verify_type  
                             int block_index  
                             byte[] in_dat  
                             int in_dat_len  
                             );
```

description:

verify the specified block of mifare card by password.

in parameter:

verify_type: this value is 0x60 or 0x61,
Usually, two values can be
ok, but some mifare card must
fill the specified value, so you

had better consult mifare
vendor.

block_index: block index of mifare.

in_dat : the content of the password.

in_dat_len : the length of the password.

out parameter:

no

return value:

0: success

-1: fail

-2: time out

4.2.6 writeMifareCard

function:

```
public int writeMifareCard ( int block_index  
                             byte[] in_dat  
                             int in_dat_len  
                             );
```

description:

write data to the specified block of mifare
card.

in parameter:

block_index : block index of mifare.

in_dat : the content of the write data.

in_dat_len : the length of the write data.

out parameter:

no

return value:

0: success

-1: fail

-2: time out

4.2.7 readMifareCard

function:

```
public int writeMifareCard ( int block_index  
                             byte[] out_dat  
                             );
```

description:

read data from the specified block of mifare card.

in parameter:

block_index : block index of mifare.

out parameter:

out_dat : data which be read from a
the specified block of
mifare card.

return value:

>0: data length

-1: fail

-2: time out