



# mPay for MR5 Reader Android SDK User Manual

Ver: 1.0

Singular Technology

2016/11/28



## 1 API Function List

S/N	Function Name	Function Description
<b>Bluetooth use</b>		
1.	BTLib(Context context_lib, <b>boolean</b> listenBluetooth)	Builder class instance
2.	setBTListener(BluetoothListener listener)	Set bluetooth listener
3.	btStart()	Start bluetooth
4.	btStop()	Stop bluetooth
5.	connectBTDeviceByAddress(String address)	Bluetooth address connect
6.	connectBTDeviceByName(String name)	Bluetooth name connect
7.	scanBTDevice()	Scan Bluetooth devices
8.	disconnectBTDevice()	Bluetooth disconnect
<b>General use</b>		
9.	setCommandListener(CommandListener commandListener)	Set command listener
10.	GetSDKVersion()	Get sdk version string
11.	cmdSetUseVersion(String ver, <b>long</b> cmdTimeOut)	Get version string
12.	cmdGetVersion( <b>long</b> cmdTimeOut)	Get version string
13.	cmdSetBluetoothDeviceName(String name, <b>long</b> cmdTimeOut)	Set bluetooth device name
14.	cmdGetReaderSN( <b>long</b> cmdTimeOut)	Get reader serial number
15.	cmdSetReaderSN(String sn, <b>long</b> cmdTimeOut)	Set reader serial number
16.	cmdDetectBattery( <b>long</b> cmdTimeOut)	Detect battery level
17.	cmdSetSleepTimer( <b>int</b> sec, <b>long</b> cmdTimeOut)	Set reader sleep timer
18.	cmdICCStatus( <b>long</b> cmdTimeOut)	Get IC card slot status
19.	cmdSetICCPort( <b>int</b> index, <b>long</b> cmdTimeOut)	Set IC card port
20.	cmdICCPowerOn( <b>long</b> cmdTimeOut)	Power on IC card
21.	cmdICCPowerOff( <b>long</b> cmdTimeOut)	Power off IC card
22.	cmdICCAccess(String cAPDU, <b>long</b> cmdTimeOut)	Access IC card
23.	cmdGetCardInfo( <b>long</b> cmdTimeOut)	Get credit card information
24.	cmdPICCActivate( <b>long</b> cmdTimeOut)	Activate RF card
25.	cmdPICCDeactivate( <b>long</b> cmdTimeOut)	Deactivate RF card
26.	cmdPICCRate( <b>long</b> cmdTimeOut)	Rate RF card
27.	cmdPICCAccess(String cAPDU, <b>long</b> cmdTimeOut)	Access RF card
28.	cmdMifareAuth(String kType, <b>int</b> sector, String Key, <b>long</b> cmdTimeOut)	Auth mifare card
29.	cmdMifareReadBlock( <b>int</b> block, <b>long</b> cmdTimeOut)	Read mifare card block
30.	cmdMifareWriteBlock( <b>int</b> Block, String data, <b>long</b> cmdTimeOut)	Write mifare card block
31.	cmdMifareIncrement( <b>int</b> Block, String data, <b>long</b> cmdTimeOut)	Incremen mifare card
32.	cmdMifareDecrement( <b>int</b> Block, String data, <b>long</b> cmdTimeOut)	Decrement mifare card
33.	cmdSelectMemoryCardType(String type, <b>long</b> cmdTimeOut)	Select memory card Type
34.	cmdMemoryCardPowerOn( <b>long</b> cmdTimeOut)	Power on memory card
35.	cmdMemoryCardGetType( <b>long</b> cmdTimeOut)	Get memory card type
36.	cmdMemoryCardReadData(String adrH, String adrL, <b>int</b> len, <b>long</b> cmdTimeOut)	Read memory card data



37.	cmdMemoryCardWriteData(String adrH, String adrL, <b>int</b> len, String cAPDU, <b>long</b> cmdTimeOut)	Write memory card data
38.	cmdMemoryCardPowerOff( <b>long</b> cmdTimeOut)	Power off memory card
39.	cmdMemoryCardReadErrorCounter( <b>long</b> cmdTimeOut)	Read memory card error counter
40.	cmdMemoryCardVerifyPSC( <b>int</b> len, String psc, <b>long</b> cmdTimeOut)	Verify memory card PSC
41.	cmdMemoryCardGetPSC( <b>int</b> len, <b>long</b> cmdTimeOut)	Get memory card PSC
42.	cmdMemoryCardModifyPSC( <b>int</b> len, String psc, <b>long</b> cmdTimeOut)	Modify memory card PSC
43.	cmdMemoryCardReadDataWithProtectBit(String adrH, String adrL, <b>int</b> len, <b>long</b> cmdTimeOut)	Read memory card data eith protectbit
44.	cmdMemoryCardWriteDataWithProtectBit(String adrH, String adrL, <b>int</b> len, String data, <b>long</b> cmdTimeOut)	Write memory card data with protectbit
45.	cmdMemoryCardReadProtectionData( <b>long</b> cmdTimeOut)	Read memory card protection data
46.	cmdMemoryCardWriteProtectionData(String adrH, String adrL, <b>int</b> len, String data, <b>long</b> cmdTimeOut)	Write memory card protection data
47.	cmdGiveUpAction( <b>long</b> cmdTimeOut)	Give up action

## 2 Callback Function List

S/N	Function Name	Function Description
<b>Bluetooth use</b>		
1.	onBluetoothState( <b>boolean</b> enable)	Respond bluetooth enable state
2.	onBluetoothDeviceScanning()	Respond bluetooth device scanning
3.	onBluetoothDeviceFound(DevItem item)	Respond bluetooth device found
4.	onBluetoothDeviceScanOver()	Respond bluetooth device scan over
5.	onBluetoothDeviceBounding()	Respond bluetooth device bounding
6.	onBluetoothDeviceBoundSuccess()	Respond bluetooth device bound success
7.	onBluetoothDeviceBoundFailed()	Respond bluetooth device bound fail
8.	onBluetoothDeviceConnecting()	Respond bluetooth device connecting
9.	onBluetoothDeviceConnected()	Respond bluetooth device connected
10.	onBluetoothDeviceConnectFailed()	Respond bluetooth device connect failed
11.	onBluetoothDeviceDisconnected()	Respond bluetooth device disconnect
<b>General use</b>		
12.	onReaderResponse( <b>int</b> returnCode, String returnMessage, String functionName)	Respond reader status
13.	onSDKResponse( <b>int</b> returnCode, String returnMessage, String functionName)	Respond sdk status
14.	onSetUseVersion( <b>boolean</b> status)	Respond set version status
15.	onGetVersion( <b>boolean</b> status, String version)	Respond version string
16.	onSetBluetoothDeviceName( <b>boolean</b> status)	Respond set status
17.	onGetReaderSN( <b>boolean</b> status, String sn)	Respond serial number



18.	onSetReaderSN( <b>boolean</b> status)	Respond set serial number status
19.	onDetectBattery( <b>boolean</b> status, String energy)	Respond battery level
20.	onSetSleepTimer( <b>boolean</b> status)	Respond set timer status
21.	onSetICCPort( <b>boolean</b> status)	Respond set IC card port status
22.	onICCStatus( <b>boolean</b> status, String iccStatus)	Respond IC card inserted or not
23.	onICCPowerOn( <b>boolean</b> status, String atr)	Respond IC card ATR
24.	onICCPowerOff( <b>boolean</b> status)	Respond status
25.	onICCAccess( <b>boolean</b> status, String rAPUD)	Respond receive apdu
26.	onGetCardInfo( <b>boolean</b> status, String info, String pan, String cardholderName, String expDate)	Respond credit card information
27.	onPICCActivate( <b>boolean</b> status, String cardSN)	Respond RF card SN
28.	onPICCDeactivate( <b>boolean</b> status)	Respond deactivate status
29.	onPICCRate( <b>boolean</b> status, String ats)	Respond RF card ats
30.	onPICCAccess( <b>boolean</b> status, String rAPUD)	Respond receive apdu
31.	onMifareAuth( <b>boolean</b> status)	Respond auth status
32.	onMifareReadBlock( <b>boolean</b> status, String data)	Respond read block data
33.	onMifareWriteBlock( <b>boolean</b> status)	Respond write block status
34.	onMifareIncrement( <b>boolean</b> status)	Respond increment status
35.	onMifareDecrement( <b>boolean</b> status)	Respond decrement status
36.	onSelectMemoryCardType( <b>boolean</b> status)	Respond select status
37.	onMemoryCardPowerOn( <b>boolean</b> status, String atr)	Respond memory card ATR
38.	onMemoryCardGetType( <b>boolean</b> status, String type_Code, String type)	Respond type
39.	onMemoryCardReadData( <b>boolean</b> status, String rAPDU)	Respond data
40.	onMemoryCardWriteData( <b>boolean</b> status)	Respond write data status
41.	onMemoryCardPowerOff( <b>boolean</b> status)	Respond power off status
42.	onMemoryCardReadErrorCounter( <b>boolean</b> status, int errorCounter)	Respond read error counter
43.	onMemoryCardVerifyPSC( <b>boolean</b> status)	Respond verify status
44.	onMemoryCardGetPSC( <b>boolean</b> status, String psc)	Respond PSC
45.	onMemoryCardModifyPSC( <b>boolean</b> status)	Respond modify PSC status
46.	onMemoryCardReadDataWithProtectBit( <b>boolean</b> status, String data)	Respond data with protectbit
47.	onMemoryCardWriteDataWithProtectBit( <b>boolean</b> status)	Respond write data with protectbit status
48.	onMemoryCardReadProtectionData( <b>boolean</b> status, String data)	Respond protection data
49.	onMemoryCardWriteProtectionData( <b>boolean</b> status)	Respond write protection data status
50.	onGiveUpAction( <b>boolean</b> status)	Respond give up action status

### 3 API Function Description

#### 3.1 **public** BTLib(Context context\_lib, **boolean** listenBluetooth) **throws** Exception



*Initialize a Builder class instance .*

**Parameter:**

`context_lib`— Specifies the context of the application.

`listenBluetooth`— Listen bluetooth enable state

*Example:* `public BTLib btLib = new BTLib(mPosActivity, true);`

### 3.2 `public void` setBTListener(BluetoothListener listener)

*Set bluetooth events callback function .*

**Parameter:**

`listener`— Bluetooth events callback function.

**Return Refer:**

`onBluetoothState(boolean enable)`

`onBluetoothDeviceScanning()`

`onBluetoothDeviceFound(DevItem item)`

`onBluetoothDeviceScanOver()`

`onBluetoothDeviceBounding()`

`onBluetoothDeviceBoundSuccess()`

`onBluetoothDeviceBoundFailed()`

`onBluetoothDeviceConnecting()`

`onBluetoothDeviceConnected()`

`onBluetoothDeviceConnectFailed()`

`onBluetoothDeviceDisconnected()`

*Example:* `btLib.setBTListener(this);`

### 3.3 `public boolean` btStart()

*Start bluetooth .*

*Example:* `btLib.btStart();`

### 3.4 `public void` btStop()

*Stop bluetooth . You must leave the program in use, to avoid missing a call to unregisterReceiver*

*Example:* `public void onBackPressed()  
{  
 btLib.btStop();  
 finish();  
}`

### 3.5 `public boolean` connectBTDeviceByAddress(String address)

*Bluetooth address connect .*



**Parameter:**

**address** — Bluetooth device address

**Return Refer:**

onBluetoothDeviceBounding()  
onBluetoothDeviceBoundSuccess()  
onBluetoothDeviceBoundFailed()  
onBluetoothDeviceConnecting()  
onBluetoothDeviceConnected()  
onBluetoothDeviceConnectFailed()

*Example:* String **mBt\_address** = **btLib**.connectBTDeviceByAddress(**mBt\_address**) ;//XX:XX:XX:XX

### 3.6 **public boolean** connectBTDeviceByName(String **name**)

*Bluetooth name connect.*

**Parameter:**

**name** — Bluetooth device name

**Return Refer:**

onBluetoothDeviceBounding()  
onBluetoothDeviceBoundSuccess()  
onBluetoothDeviceBoundFailed()  
onBluetoothDeviceConnecting()  
onBluetoothDeviceConnected()  
onBluetoothDeviceConnectFailed()

*Example:* String **mBt\_name** = **btLib**.connectBTDeviceByName(**mBt\_name**) ;

### 3.7 **public boolean** scanBTDevice()

*Scanning for devices.*

**Return Refer:**

onBluetoothDeviceScanning()  
onBluetoothDeviceFound(DeviceItem **item**)  
onBluetoothDeviceScanOver()

*Example:* **btLib**.scanBTDevice();

### 3.8 **public void** disconnectBTDevice()

*Bluetooth device disonnect.*

**Return Refer:**

onBluetoothDeviceDisconnected ()

*Example:* **btLib**.disconnectBTDevice ();

### 3.9 **public void** setCommandListener(CommandListener **commandListener**)



*Set command events callback function .*

**Parameter:**

**commandListener**— Command events callback function.

**Return Refer:**

onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
onSetUseVersion(**boolean** status)  
onGetVersion(**boolean** status, String version)  
onSetBluetoothDeviceName(**boolean** status)  
onGetReaderSN(**boolean** status, String sn)  
onSetReaderSN(**boolean** status)  
onDetectBattery(**boolean** status, String energy)  
onSetSleepTimer(**boolean** status)  
onSetICCPort(**boolean** status)  
onICCStatus(**boolean** status, String iccStatus)  
onICCPowerOn(**boolean** status, String atr)  
onICCPowerOff(**boolean** status)  
onICCAccess(**boolean** status, String rAPUD)  
onGetCardInfo(**boolean** status, String info, String pan, String cardholderName, String expDate)  
onPICCActivate(**boolean** status, String cardSN)  
onPICCDeactivate(**boolean** status)  
onPICCRate(**boolean** status, String ats)  
onPICCAccess(**boolean** status, String rAPUD)  
onMifareAuth(**boolean** status)  
onMifareReadBlock(**boolean** status, String data)  
onMifareWriteBlock(**boolean** status)  
onMifareIncrement(**boolean** status)  
onMifareDecrement(**boolean** status)  
onSelectMemoryCardType(**boolean** status)  
onMemoryCardPowerOn(**boolean** status, String atr)  
onMemoryCardGetType(**boolean** status, String type\_Code,String type)  
onMemoryCardReadData(**boolean** status, String rAPDU)  
onMemoryCardWriteData(**boolean** status)  
onMemoryCardPowerOff(**boolean** status)  
onMemoryCardReadErrorCounter(**boolean** status, **int** errorCounter)  
onMemoryCardVerifyPSC(**boolean** status)  
onMemoryCardGetPSC(**boolean** status, String psc)  
onMemoryCardModifyPSC(**boolean** status)  
onMemoryCardReadDataWithProtectBit(**boolean** status, String data)  
onMemoryCardWriteDataWithProtectBit(**boolean** status)  
onMemoryCardReadProtectionData(**boolean** status, String data)  
onMemoryCardWriteProtectionData(**boolean** status)  
onGiveUpAction(**boolean** status)

<i>Example:</i>	<b>btLib.setCommandListener(this) ;</b>
-----------------	---



### 3.10 **public static** String getSDKVersion()

*Get SDK version string .*

**Return:** SDK version string

*Example:* String `str_SDKVer` = GetSDKVersion() ;

### 3.11 **public abstract void** cmdSetUseVersion(String ver, **long** cmdTimeOut)

*Set terminal version string .*

**Parameter:**

`ver` — Terminal version string (ASCII code)

`cmdTimeOut` — Function wait respond timer.

**Return Refer:**

`success` — onGetVersion(**boolean** status, String version)

`fail` — onReaderResponse(**int** returnCode, String returnMessage, String functionName)

onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.12 **public abstract void** cmdGetVersion(**long** cmdTimeOut)

*Get terminal version string .*

**Parameter:**

`cmdTimeOut` — Function wait respond timer.

**Return Refer:**

`success` — onGetVersion(**boolean** status, String version)

`fail` — onReaderResponse(**int** returnCode, String returnMessage, String functionName)

onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.13 **public abstract void** cmdSetBluetoothDeviceName(String name, **long** cmdTimeOut)

*Set terminal bluetooth name .*

**Parameter:**

`name` — Bluetooth device name (ASCII code)

`cmdTimeOut` — Function wait respond timer.

**Return Refer:**

`success` — onSetBluetoothDeviceName(**boolean** status)

`fail` — onReaderResponse(**int** returnCode, String returnMessage, String functionName)

onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.14 **public abstract void** cmdGetReaderSN(**long** cmdTimeOut)

*Get Reader serial number .*

**Parameter:**

`cmdTimeOut` — Function wait respond timer.



**Return Refer:**

success — onGetReaderSN(**boolean** status, String sn)  
fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.15 public abstract void** cmdSetReaderSN(String sn, **long** cmdTimeOut)

*Set Reader serial number .*

**Parameter:**

sn — Reader serial number  
cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onSetReaderSN(**boolean** status)  
fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.16 public abstract void** cmdDetectBattery(**long** cmdTimeOut)

*Detect battery level .*

**Parameter:**

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onDetectBattery(**boolean** status, String energy)  
fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.17 public abstract void** cmdSetSleepTimer(**int** sec, **long** cmdTimeOut)

*Set sleep timer .*

**Parameter:**

sec — 1~30 (min) , 0 : not sleep  
cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onSetSleepTimer(**boolean** status)  
fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.18 public abstract void** cmdSetICCPort(**int** index, **long** cmdTimeOut)

*Set IC card port .*

**Parameter:**

index — Smart card:0 (0x00) or Sam card :1(0x01)  
cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onSetICCPort(**boolean** status)



fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.19 **public abstract void** cmdICCStatus(**long** cmdTimeOut)

*Get IC card slot status.*

**Parameter:**

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onICCStatus(**boolean** status, String iccStatus)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.20 **public abstract void** cmdICCPowerOn(**long** cmdTimeOut)

*Power on IC card and get ATR.*

**Parameter:**

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onICCPowerOn(**boolean** status, String atr)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.21 **public abstract void** cmdICCPowerOff(**long** cmdTimeOut)

*Power off IC card.*

**Parameter:**

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onICCPowerOff(**boolean** status)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.22 **public abstract void** cmdICCAccess(String cAPDU, **long** cmdTimeOut)

*Access IC card.*

**Parameter:**

cAPDU — Send command to IC card

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onICCAccess(**boolean** status, String rAPUD)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.23 **public abstract void** cmdGetCardInfo(**long** cmdTimeOut)



#### *Get card information .*

**Parameter:**

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onGetCardInfo(**boolean** status, String info, String pan, String cardholderName, String expDate)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.24 **public abstract void** cmdPICCActivate(**long** cmdTimeOut)

#### *Activation PICC .*

**Parameter:**

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onPICCActivate(**boolean** status, String cardSN)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.25 **public abstract void** cmdPICCDeactivate(**long** cmdTimeOut)

#### *Deactivation PICC .*

**Parameter:**

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onPICCDeactivate(**boolean** status)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.26 **public abstract void** cmdPICCRate(**long** cmdTimeOut)

#### *Rate PICC.*

**Parameter:**

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onPICCRate(**boolean** status, String ats)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.27 **public abstract void** cmdPICCAccess(String cAPDU, **long** cmdTimeOut)

#### *Access PICC.*

**Parameter:**

cAPDU — Send command to RF card



cmdTimeOut— Function wait respond timer.

**Return Refer:**

success— onPICCAccess(**boolean** status, String rAPUD)

fail— onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.28 public abstract void** cmdMifareAuth(String kType, **int** sector, String key, **long** cmdTimeOut)

*Authentication Mifare card .*

**Parameter:**

kType— Type A(0x41) or Type B(0x42)

sector— Sector number

Key— Auth key

cmdTimeOut— Function wait respond timer.

**Return Refer:**

success— onMifareAuth(**boolean** status)

fail— onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.29 public abstract void** cmdMifareReadBlock(**int** block, **long** cmdTimeOut)

*Read Mifare block.*

**Parameter:**

type— Type A(0x41) or Type B(0x42)

cmdTimeOut— Function wait respond timer.

**Return Refer:**

success— onMifareReadBlock(**boolean** status, String data)

fail— onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.30 public abstract void** cmdMifareWriteBlock(**int** block, String data, **long** cmdTimeOut)

*Write Mifare block.*

**Parameter:**

block— Block number

data—

cmdTimeOut— Function wait respond timer.

**Return Refer:**

success— onMifareWriteBlock(**boolean** status)

fail— onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.31 public abstract void** cmdMifareIncrement(**int** Block, String data, **long** cmdTimeOut)

*Increments Mifare Card. .*



**Parameter:**

block — Block number  
data —  
cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onMifareIncrement(**boolean** status)  
fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.32 public abstract void cmdMifareDecrement(int block, String data, long cmdTimeOut)**

*Decrements Mifare card.*

**Parameter:**

block — Block number  
data —  
cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onMifareDecrement(**boolean** status)  
fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.33 public abstract void cmdSelectMemoryCardType(String type, long cmdTimeOut)**

*Operate the specified IC Card in the Socket1.*

**Parameter:**

type — SLE4442(0x02) 、 SLE4428(0x04) 、 AT24C01(0x0A) 、 AT24C02(0x0B) 、  
AT24C04(0x0C) 、 AT24C08(0x0D) 、 AT24C16(0x0E) 、 AT24C32(0x13) 、  
AT24C64(0x14)  
cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onSelectMemoryCardType(**boolean** status)  
fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.34 public abstract void cmdMemoryCardPowerOn(long cmdTimeOut)**

*Power on (reset) the memory card selected.*

**Parameter:**

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onMemoryCardPowerOn(**boolean** status, String atr)  
fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)



### 3.35 **public abstract void** cmdMemoryCardGetType(**long** cmdTimeOut)

*Get the memory Card type in the Socket1.*

**Parameter:**

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onMemoryCardGetType(**boolean** status, String type\_Code,String type)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.36 **public abstract void** cmdMemoryCardReadData(String adrH, String adrL, **int** len, **long** cmdTimeOut)

*Read the memory card selected and then send back the response from the card (read APDU response) .*

**Parameter:**

adrH — High byte of the start address of the card.

adrL — Low byte of the start address of the card.

len — Read data length.

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onMemoryCardReadData(**boolean** status, String rAPDU)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.37 **public abstract void** cmdMemoryCardWriteData(String adrH, String adrL, **int** len, String cAPDU, **long** cmdTimeOut)

*Write the memory card selected (write APDU command).*

**Parameter:**

adrH — High byte of the start address of the card.

adrL — Low byte of the start address of the card.

len — Write data length.

cAPDU —

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onMemoryCardWriteData(**boolean** status)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

### 3.38 **public abstract void** cmdMemoryCardPowerOff(**long** cmdTimeOut)



*Power off the memory card selected.*

**Parameter:**

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onMemoryCardPowerOff(**boolean** status)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.39 public abstract void cmdMemoryCardReadErrorCounter(long cmdTimeOut)**

*Read the Error Counter of memory card selected. It's for SLE4442 /SLE4428 card only.*

**Parameter:**

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onMemoryCardReadErrorCounter(**boolean** status, **int** errorCounter)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.40 public abstract void cmdMemoryCardVerifyPSC(int len, String psc, long cmdTimeOut)**

*Verify PSC of memory card selected. It's for SLE4442 /SLE4428 card only.*

**Parameter:**

len — PSC length.

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onMemoryCardVerifyPSC(**boolean** status)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.41 public abstract void cmdMemoryCardGetPSC(int len, long cmdTimeOut)**

*Get PSC of memory card selected. It's for SLE4442 /SLE4428 card only.*

**Parameter:**

len — PSC length.

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onMemoryCardGetPSC(**boolean** status, String psc)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.42 public abstract void cmdMemoryCardModifyPSC(int len, String psc, long cmdTimeOut)**



*Modify PSC of memory card selected. It's for SLE4428 card only.*

**Parameter:**

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onMemoryCardModifyPSC(**boolean** status)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.43 public abstract void** cmdMemoryCardReadDataWithProtectBit(String adrH, String adrL, **int** len, **long** cmdTimeOut)

*Read data with protect bit of memory card selected. It's for SLE4428 card only.*

**Parameter:**

adrH — High byte of the start address of the card.

adrL — Low byte of the start address of the card.

len — Read data length.

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onMemoryCardReadDataWithProtectBit(**boolean** status, String data)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.44 public abstract void** cmdMemoryCardWriteDataWithProtectBit(String adrH, String adrL, **int** len, String data, **long** cmdTimeOut)

*Write data with protect bit of memory card selected. It's for SLE4428 card only.*

**Parameter:**

adrH — High byte of the start address of the card.

adrL — Low byte of the start address of the card.

len — Write data length.

data —

cmdTimeOut — Function wait respond timer.

**Return Refer:**

success — onMemoryCardWriteDataWithProtectBit(**boolean** status)

fail — onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**3.45 public abstract void** cmdMemoryCardReadProtectionData(**long** cmdTimeOut)

*Read data with protect bit of memory card selected. It's for SLE4442 card only.*



**Parameter:**

`cmdTimeOut` — Function wait respond timer.

**Return Refer:**

`success` — `onMemoryCardReadProtectionData(boolean status, String data)`

`fail` — `onReaderResponse(int returnCode, String returnMessage, String functionName)`  
`onSDKResponse(int returnCode, String returnMessage, String functionName)`

**3.46 public abstract void** `cmdMemoryCardWriteProtectionData(String adrH, String adrL, int len, String data, long cmdTimeOut)`

*Request the reader to write data with protect bit of memory card selected. It's for SLE4442 card only.*

**Parameter:**

`adrH` — High byte of the start address of the card.

`adrL` — Low byte of the start address of the card. (SLE4442 Protection Memory Address:  
0~31(decimal))

`len` — Write data length.

`data` —

`cmdTimeOut` — Function wait respond timer.

**Return Refer:**

`success` — `onMemoryCardWriteProtectionData(boolean status)`

`fail` — `onReaderResponse(int returnCode, String returnMessage, String functionName)`  
`onSDKResponse(int returnCode, String returnMessage, String functionName)`

**3.47 public abstract void** `cmdGiveUpAction(long cmdTimeOut)`

*Give Up Action .*

**Parameter:**

`cmdTimeOut` — Function wait respond timer

**Return Refer:**

`success` — `onGiveUpAction(boolean status)`

`fail` — `onReaderResponse(int returnCode, String returnMessage, String functionName)`  
`onSDKResponse(int returnCode, String returnMessage, String functionName)`

## 4 Callback Function Description

**4.1** `onBluetoothState(boolean enable)`

*Callback bluetooth enable state .*

**Return :** `enable` — `false`: not enable  
`true`: enable



#### 4.2 onBluetoothDeviceScanning()

*Callback device scanning .*

#### 4.3 onBluetoothDeviceFound(DevItem item)

*Callback bluetooth found device .*

**Return:**

item — <b>public</b> String dev_name	// device name
<b>public</b> String dev_address	// device address
<b>public boolean</b> dev_isBond	// device is bond state
<b>public int</b> dev_rssi	// device rssi

#### 4.4 onBluetoothDeviceScanOver()

*Callback bluetooth device scan over .*

#### 4.5 onBluetoothDeviceBounding()

*Callback bluetooth device bounding .*

#### 4.6 onBluetoothDeviceBoundSuccess()

*Callback bluetooth device bound success .*

#### 4.7 onBluetoothDeviceBoundFailed()

*Callback bluetooth device bound fail .*

#### 4.8 onBluetoothDeviceConnecting()

*Callback bluetooth device connecting .*

#### 4.9 onBluetoothDeviceConnected()

*Callback bluetooth device connected .*

#### 4.10 onBluetoothDeviceConnectFailed()

*Callback bluetooth device connect fail .*

#### 4.11 onBluetoothDeviceDisconnected()

*Callback bluetooth device disonnect .*

#### 4.12 **public abstract void** onReaderResponse(**int** returnCode, String returnMessage, String functionName)



*Callback device respond status.*

**Return:**

returnCode & returnMessage — 0(0x00):PASS  
1(0x01):Command error  
2(0x02):Parameter error  
3(0x03):CRC error  
4(0x04): Time out  
36(0x24): Data error  
functionName — Response function name

**4.13 public abstract void** onSDKResponse(**int** returnCode, String returnMessage, String functionName)

*Callback device respond status.*

**Return:**

returnCode & returnMessage — 0(0x00):PASS  
3(0x03):CRC error  
4(0x04): Time out  
35(0x23): Data length error  
36(0x24): Data error  
functionName — Response function name

**4.14 public abstract void** onSetUseVersion(**boolean** status)

*Callback set version status .*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**4.15 public abstract void** onGetVersion(**boolean** status, String version)

*Callback get terminal version result .*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
version — terminal version string

**4.16 public abstract void** onSetBluetoothDeviceName(**boolean** status)

*Callback set bluetooth device name status .*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)



#### 4.17 **public abstract void** onGetReaderSN(**boolean** status, String sn)

*Callback reader SN.*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
sn — Reader serial number

#### 4.18 **public abstract void** onSetReaderSN(**boolean** status)

*Callback set reader SN status.*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

#### 4.19 **public abstract void** onDetectBattery(**boolean** status, String energy)

*Callback get device battery.*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
energy — Empty , 1/3, 2/3, Full, Full & Connect USB, Charging

#### 4.20 **public abstract void** onSetSleepTimer(**boolean** status)

*Callback set sleep timer status.*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

#### 4.21 **public abstract void** onSetICCPort (**boolean** status)

*Callback set IC card port staust.*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

#### 4.22 **public abstract void** onICCStatus(**boolean** status, String iccStatus)

*Callback get IC card status result.*



**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
r\_code & iccStatus — 0(0x00): Card not insert  
1(0x01): Card inserted

**4.23 public abstract void onICCPowerOn(boolean status, String atr)**

*Callback get IC card power on result.*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
atr — Answer-to-reset.

**4.24 public abstract void onICCPowerOff(boolean status)**

*Callback get IC card power off staust.*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**4.25 public abstract void onICCAccess(boolean status, String rAPUD)**

*Callback get IC card access result.*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
rAPUD — get response from IC card.

**4.26 public abstract void onGetCardInfo(boolean status, String info, String pan, String cardholderName, String expDate)**

*Callback credit card information*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
info — A1 " pan(hex)" B1 " cardholder name(hex)" C1 " expire date(hex)"  
pan — respond pan (ASCII code)  
cardholder\_name — respond cardholder name (ASCII code)  
exp\_date — respond expire date (ASCII code)



#### 4.27 **public abstract void** onPICCActivate(**boolean** status, String cardSN)

*Callback activate RF card and get card SN .*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
cardSN — RF card SN

#### 4.28 **public abstract void** onPICCDeactivate(**boolean** status)

*Callback deactivate RF card status .*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

#### 4.29 **public abstract void** onPICCRate(**boolean** status, String ats)

*Callback RF card ats .*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
ats — RF card ats

#### 4.30 **public abstract void** onPICCAccess(**boolean** status, String rAPUD)

*Callback get RF card access result*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
rAPUD — get response from RF card

#### 4.31 **public abstract void** onMifareAuth(**boolean** status)

*Callback authentication mifare card status .*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

#### 4.32 **public abstract void** onMifareReadBlock(**boolean** status, String data)



*Callback read mifare card block data .*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
data — Mifare card Block data

**4.33 public abstract void** onMifareWriteBlock(**boolean** status)

*Callback write mifare card block status .*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**4.34 public abstract void** onMifareIncrement(**boolean** status)

*Callback increments mifare card status .*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**4.35 public abstract void** onMifareDecrement(**boolean** status)

*Callback decrements mifare card status .*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**4.36 public abstract void** onSelectMemoryCardType(**boolean** status)

*Callback select memory card type status .*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**4.37 public abstract void** onMemoryCardPowerOn(**boolean** status, String atr)

*Callback power on memory card result .*

**Return:**

status — true: success



false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
atr — Answer-to-reset.

#### 4.38 **public abstract void** onMemoryCardGetType(**boolean** status, String type\_Code,String type)

*Callback memory card type.*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
type & typeCode — SLE4442(0x02) 、 SLE4428(0x04) 、 AT24C01(0x0A) 、 AT24C02(0x0B) 、  
AT24C04(0x0C) 、 AT24C08(0x0D) 、 AT24C16(0x0E) 、 AT24C32(0x13) 、  
AT24C64(0x14)

#### 4.39 **public abstract void** onMemoryCardReadData(**boolean** status, String rAPDU)

*Callback read memory card data result.*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
rAPDU —

#### 4.40 **public abstract void** onMemoryCardWriteData(**boolean** status)

*Callback write memory card data status.*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

#### 4.41 **public abstract void** onMemoryCardPowerOff(**boolean** status)

*Callback power off memory card status.*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

#### 4.42 **public abstract void** onMemoryCardReadErrorCounter(**boolean** status, **int** errorCounter)

*Callback read memory card rrror counter result.*

**Return:**

status — true: success  
false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)



onSDKResponse(**int** returnCode, String returnMessage, String functionName)  
errorCounter—SLE4442:00~07(0~7)、SLE4428:00~FF(0~255)

#### 4.43 **public abstract void** onMemoryCardVerifyPSC(**boolean** status)

*Callback verify memory card psc status.*

**Return:**

status—true: success

false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

#### 4.44 **public abstract void** onMemoryCardGetPSC(**boolean** status, String psc)

*Callback get memory card psc result.*

**Return:**

status—true: success

false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

psc—

#### 4.45 **public abstract void** onMemoryCardModifyPSC(**boolean** status)

*Callback modify memory card psc status.*

**Return:**

status—true: success

false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

#### 4.46 **public abstract void** onMemoryCardReadDataWithProtectBit(**boolean** status, String data)

*Callback read memory card data with protectbit result.*

**Return:**

status—true: success

false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

sn—Reader serial number

#### 4.47 **public abstract void** onMemoryCardWriteDataWithProtectBit(**boolean** status)

*Callback write memory card data with protectbit status.*

**Return:**

status—true: success

false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

#### 4.48 **public abstract void** onMemoryCardReadProtectionData(**boolean** status, String data)



*Callback read memory card protection data result.*

**Return:**

**status** — true: success

false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**sn** — Reader serial number

#### 4.49 **public abstract void** onMemoryCardWriteProtectionData(**boolean** status)

*Callback write memory card protection data status.*

**Return:**

**status** — true: success

false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

#### 4.50 **public abstract void** onGiveUpAction(**boolean** status)

*Callback give up action status.*

**Return:**

**status** — true: success

false: onReaderResponse(**int** returnCode, String returnMessage, String functionName)  
onSDKResponse(**int** returnCode, String returnMessage, String functionName)

**sn** — Reader serial number

#### Revision History:

Revision	Description	Data
V1.0	Modify function name and description	2016/11/28