API Interface Specification

Programmer's Reference Manual (Android)

V- 3.08.014

2026-05-07

Shenzhen Xinguodu Technology Co., Ltd. Published

www.xinguodu.com

Version	Date	Editor	Description
V0.1	2016-08-12	Liu Ting	
V0.2	2016-9-12	Liu Ting	Startprint () function to modify the number of Parameters
V0.3	2016-10-12	Zhou Xiaoxin	Add the appendix Return Value that is refined into the class Description, as well as some function changes
V 0.4	2016-10-31	Zhou Xiaoxin	Modify the buzzer interface Modify EMV card module power up and down from the application layer control Modify PIN pad work Cadogan a key type
V 0.5	2016-11-11	Zhou Xiao Xin	Increase silent uninstall Interface
V 0.6	2016-11-25	Zhou Xin	EMV interface modification, the amount of correction, whether the use of electronic cash callback
V 0.7	2016-12-05	Zhou Xin	Print Increment Print One-dimensional code Two-dimensional code interface
V0.8	2016-12-08	Zhou Xiaoxin	Increase the camera scan interface
V1.0. 4	2017-01-13	WangHongyang	According to the reference document, organize the structure of the document
V2.0.1	2017-05-04	QiangFang	Increase DUKTP
V2.0.6	2018-01-04	Hassan	Add onSetAfterFinalSelectedAppResponse, and onAfterFinalSelectedApp method.
V2.0.7	2018-05-28	Hassan	Sync new methods
V2.0.9	2018-10-29	Hassan	1. OnEmvProcessListener add callback: onPrompt, onRemoveCard 2. EmvHandler add method: onSetRemoveCardResponse, onSetPromptResponse 3. Remove armeabi(include libnexgo_emvjni.so, libnexgo_gencode.so, libnexgo_sdkemvjni.so). 4. Change SDK package format: jar-→aar 5. CardReader add Felica method: setSupportFelica: if support Felica card setFelicaRequestCode, setFelicaSystemCode 6. CardReader add Read Mag Stripe original Track data method: setMagReaderRawData 7. Delete method loadKeyByCom and cancelLoadKey 8. Change others
V2.1.1	2019-01-24	Hassan	1. Fixed mag-stripe reader issue

			2. Fixed emv onRequestAmount method 3. Other fix
V2.1.1	2019-03-18	ShaoPu	Add Desfire card API
V2.2.1	2019-06-20	Hassan	1. EMV add new funcion for Pure kernel 2. add new API for set pure kernel Aid and CapkEntity 3. add new API for set pure enable aid select first 4. fix custom layout pinpad issue 5. paypass add RRP function 6. add new API for get sigature statue . 7. add new API for security
V2.2.1	2019-09-20	Randall	1. Add scanner 2 for customized UI scan
V2.3.1	2019-12-20	Hassan	1. Add Mifare Ultralight API 2. Add DRL for paywave and amex
V2.3.3	2020-05-11	Hassan	1. Optimize MAC algorithm 2. Fix paypass RRP issue 3. Optimize emv flow 4. other issue fixed
V3.0.1	2020-06-11	Hassan	1. add emvHandler2 API 2. Fix Pure card terminal capability issue. 3. Remove the install and uninstall API
V3.0.2	2020-10-21	Hassan	1. Emv add new MB kernel 2. Compatible with N86 3. Remove the HSM APIs 4. other issue fixed
V3.0.3	2021-04-12	Hassan	1. add arm64 library. 2. Add PB contactless for Italy 3. Add platform object, includes Install, Uninstall, Reboot, UpdateFirmware ..etc 4. Emvhandler2 ,EMV contactless add read application data mode 5. Emvhandler1 add onAfterFinalSelectedApp callback for MIR,RUPAY 6. Add API dukptCipherKeyInject, inject the cipher BDK/IPEK key 7. Other issue fixed
V3.0.4	2021-08-09	Hassan	1. Add new API UsbSerial 2. Fix the Paypass 9F7B tag process issue for emvHandler 3. Optimize the process with card power on and power off 4. Platform adds shutDownDevice , enableUsbCdc, disableUsbCdc, getUsbCdcStatus,

			executeGeneralMethod 5. Update the EMV L1, L2 version code in emvKernelVersionInfo 6. Fix get device model for N5 with getDeviceInfo method 7. Add get firmWareFullVersion 8. Integrate NexgoSystemSdk 9. Other issue fixed
V3.0.5	2022-07-22	Hassan	1. Add new API NTAGCardHandler to process the NTAG card 2. Support extended APDU(Expand to 2K) for CPU card 3. Fix type 9C issue of RUPAY card 4. Add EMV error code: Emv_CTLS_EndApplication Emv_CTLS_Torn 5. Compatible with UN20 6. Pinpad add inputPinExternal to support External Pinpad to enter the PIN 7. Add MDB MdbSerialPortDriver and MdbLEDDriver 8. Fix TVR(online pin set) for DPAS 9. Add support type B cards :CTS512 and SRT512 10. Add PSAM4 for UN20 11. Compatible with External Pinpad K110 12. Add setRupayForceOnline for Rupay 13. BeepVolumeModeEnum add BEEP_MODE_SYSTEM_VOLUME. Beep sound following system volume 14. Optimize the printing of Qr-code and Bar-code 15. Platform add showNavigationBar and hideNavigationBar API 16. Other issue fixed
V3.0.6	2022.10.24	Hassan	1. Add switchMobileDataNetwork setNetworkStatusListener for mobile data in Platform 2. Fix MIR card issue 3. Add Capce Kernel 4. Paypass support 9F0A 5. Paywave and Pure contctless refund kernel return Online Approve instead of Offline Approve if terminal request AAC 6. Optimize print function 7. JCB contactless support PIN bypass

			8. Fix EMV cancel issue 9. Aid Maximum number change from 50 to 100 compatible
V3.07.001	2023.06.12	Hassan	1. Fix compatible with MasterCard tag BF0C during final application selection 2. Add setNetworkStatusListener in Platform 3. Compatible with N96, N82, N6 android 10 4. Fix the issue of Null applabel in the multi application selection list(onSelApp) 5. Fix the issue of EMV country code and currency code 6. Add Paypass new aid in old API EmvHandler 7. UN20 Add GPIO API 8. Add rfu Member variable in CandidateAppInfoEntity, it is used for returning extro EMV tags such as 42, 5F55 during the multi application selection list(onSelApp) for EMV contact. Application can select specify application by these tags. 9. Fix Scanner1 init failed issue 10. Scanner1 add “hideFrame”(set in Bundle) to hide the Preview box during scan 11. Scanner1 add List<SymbolEnum> in ScannerCfgEntity to support the specify Codes 12. N96 add DECORATIVE_LIGHT 13. Optimize EMV contactless refund flow 14. Emvhandler2 add initReader to support exteral reader to read contactless card 15. Add ExtPinpad class to support use external pinpad inject key, encryption, pin enter or calculate PinBlock 16. Optimize CBC mode iv issue with new model 17. Optimize pure kernel CDCVM 18. Fix CDA issue of DPAS 19. Paywave add API skip Process Restrict 20. UN20 MdbSerialPortDriver support Master mode and Slave mode read and write at same time 21. DeviceInfo add secure chip Core version and Boot version 10 and 11 points need latest firmware to support.
V3.07.002	2023.11.12	Hassan	1. DPAS support read 9F6E tag

			2. MADA support read 9F19,9F24,9F25,9F52 and DF4B tags
V3.08.001	2024.03.25	Hassan	1. DUKPT support AES algorithm 2. Optimize EMV random number processing 3. EMV second GAC add paramater "EmvTerminalDecisionForSecondGAC",to force TC regardless of host response code 4. Add API contactlessConfigKernelId to assign EMV contactless flow if card return kernel ID 5. Optimize K110 display text and image API 6. DPAS add setDpasVersion to support DPAS 2.0 if needed 7. Pinpad add injectKBPK, injectTr31Key to support TR31 key injection
V3.08.002	2024.04.07	Hassan	1. Optimize DPAS MSD
V3.08.003	2024.06.13	Hassan	1. EMV add Emv_CTLS_TransTryAgain，Emv_CTLS_CardInException error code
V3.08.004	2024.10.30	Hassan	1. EMV contactless PPSE return Cardblock error incase card return 6A81 2. Support Extended Selection 3. EMV support read TAG 87， 5F2D， 9F29, BF0C 4. Fix EMV auth code 5. EMV Add API to check card type is Physical card or mobile phone 6. Optimize expresspay response code process 7. PinPad add “dukptDeleteKey”, only apply to N96 8. Add CT20 and N80 emv kernel version 9. Platfrom add forceStopApp stop application 10. Platfrom add grantLogPermission get log permission 11. Pinpad add injectTr31KeyWithTMK
V3.08.006	2024.12.06	Hassan	1. Pinpad injectTr31 support DUKPT AES 2. Add N92 EMV kernel version 3. Update EMV L1/L2 kernel version 4. Optimize EMV response code process 5. Optimize EMV onFinish error code 6. Fix CPACE reselection application initialization issue 7. Fix CPACE issue when GPO not return AFL
V3.08.007	2025.02.17	Hassan	1. Compatible with N86 Pro, N6 Pro, N6 Lite 2. Support N92 ExtendLcd 3. Add API: CardEmulationHandler, support models: N86 Pro, N6 Pro, N6 Lite, N92
V3.08.008	2025.03.14	Lee	1. Add initPlatform interface in Platform, use for

			getDeviceEngine and getPlatform at the same time, then call interfaces in Platform immediately 2. Change the setSystemClock interface into Platform 3. Updated CardEmulationHanlder interfaces
v3.08.009	2025.03.21	lee	1. Update extend lcd interface
v3.08.010	2025.05.28	Lee	1. Added beep, setBeepVolume, inputText interfaces in ExtPinPad 2. Added dukpt decrypted interface 3. Added mifare sam class 4. Pinpad add getBlindPinPadCapability to check device support Blind pinpad or not 5. Blind Pinpad support customized layout for separating “cancel” button and “clear” button 6. Pinpad encryptByMKey support AES encryption 7. Optimize TR31 API to support inject work key 8. Update EMV L1/L2 kernel version 9. EMV add setPayPassForceEmvMode to for paypass run EMV mode. 10. EMV add setPureImplOptions, setPureMTOL, contactlessConfigKernelId APIs 11. Cardreader add setSearchReader, getRfDriverVersion, getIccDriverVersion APIs 12. Platform Add enableDataRoaming, enableMobileData, executeGeneralMethod APIs 13. emvHandler2 add API onSetPinInputResponseV2 identify pinpad status error. 14. Add P2PE API; currently only works for N96 Pinpad add below APIs: isP2PEMode, isP2PEAppInWhitelist, inputOnlinePinP2PE, manualInput, setManualEntryTypeMode, setManualInputCardLayout, P2PEGetSustomizedCipherText Cardreader add below APIs: SetP2PEConfiguration Emvhandler2 add below APIs: P2PEGetSustomizedCipherText
v3.08.011	2025.10.11	Hassan	1. Compatible with P300, P300MINI, EF20, N96P 2. SerialPortDriver add detail description for portNum

			with different models and base. 3. Compatible for N92(with second LCD) print 4. Printer add API setInvert(boolean invert); it works for Vector Print API and text print only Printer LineOptionEntity add setInvert. 5. UsbSerial add API write(byte[] data, int dataLen, int timeout) which is integrated with KD69 6. Pinpad add DUKPT AES APIs: setDukptAlgorithmMode setDukptAESGenerateKeyType 7. Pinpad add API dukptKsnIncreaseV2 can return the KSN increase status 8. Pinpad support AES-MAC for N96 9. Blind PIN support play .mp3 audio file 10. ExtendLcd add big FONT 11. ExtendLcd add API isExtendLcdExist to check if second LCD exist 12. ExtendLcd add APIs showBitmap(Bitmap bitmap) and getExtendLcdSize() 13. MdbSerialPortDriver add API for Capturing MDB logs 14. MdbSerialPortDriver add API mdbSetACKMode, application can set the poll ACK is reponsed by application or by MDB kernel 15. EmvHandler2 add API configPrivateTag to get private tags with getTlv method 16. EmvHandler2 add API getLastApduCommand to get the last EMV APDU command 17. EmvTransConfigurationEntity add setTerminalDecisionForFirstGAC to config first GAC terminal decision. See EmvContactTerminalDecisionForFirstGAC definition. 18. Pure kernel transfer errorcode ONDEVICECVMREQUESTED to Emv_Plz_See_Phone 19. Platform add APIs isSupportBlindPinPad and isSupportRkl 20. Update N86 EMV L2 kernel version to XGD-EMVCT Version V4.4

			21. Optimize EMV contact script processing
3.08.012	2026.01.10	Hassan	1. Fix CPACE Kernel TAG 9F37 RRP issue 2. EMV add API getPureKernelCapab 3. Optimize the ExpressPay kernel process. When CDA fails and 9F4B verification fails, the transaction is rejected 4. JCB support onRemoveCard callback 5. Poweroff RF contactless read during onRemoveCard callback 6. DPAS support config contactless limit in onTransInitBeforeGPO 7. SDK native so library meet the requirement for uploading to google play store 8. GPIO API support N6202 new base 9. N86 DPAS kernel version update to XGD-DPAS V2.1 10. Optimize the initialization of the work key index to prevent false detection of work keys
3.08.013	2026.04.15	Hassan	1. Platform add API to control enable or disable PPS enableIccPsamPPSControl 2. Platform add API to control enable or disable beep during pin enter enablePinpadBeep 3. GPIO add openGpio and closeGpio for UN10 openGpio closeGpio 4. Cpace kernel add API to deside the pin enter is processed by app or by kernel: EmvTransConfigurationEntity add CpaceTransDataEntity.isKernelTriggerPinEnter 5. Cpace contactless refund return AAC with online process instead of decline 6. Optimize EMV contactless card process with K110 7. Optimize platform useage for PSS to avoid the restriction of system broadcast
3.08.014	2026.05.07	Hassan	1. EMV EmvHandler2 support new card brand Interac 2. Fix JCB Contactless API EmvHandler2.getEmvContactlessCardType failed issue

			3. EMV add DF8129 tag, app can read DF8129 with getTlv API 4. Fix the issue where Scanner2 cannot automatically focus when scanning codes for N96 5. Optimize the configuration and processing of the front and rear cameras of Scanner 2

SOFTWARE LICENSE AGREEMENT

IMPORTANT: YOU SHOULD CAREFULLY READ ALL THE TERMS, CONDITIONS AND RESTRICTIONS OF THIS LICENSE AGREEMENT BEFORE INSTALLING THE SOFTWARE PACKAGE. YOUR INSTALLATION OF THE SOFTWARE PACKAGE PRESUMES YOUR ACCEPTANCE OF THE TERMS, CONDITIONS, AND RESTRICTIONS CONTAINED IN THIS AGREEMENT. IF YOU DO NOT AGREE WITH THESE TERMS, CONDITIONS, AND RESTRICTIONS, PROMPTLY RETURN THE SOFTWARE PACKAGE AND ASSOCIATED DOCUMENTATION TO THE ADDRESS ON THE FRONT PAGE OF THIS DOCUMENT, ATTENTION: CUSTOMER SUPPORT.

TERMS, CONDITIONS, AND RESTRICTIONS

Shenzhen Xinguodu Technology Co., Ltd (the "Licensor") owns and has the right to distribute the described software and documentation, collectively referred to as the "Software".

LICENSE: Licensor grants you (the "Licensee") the right to use the Software in conjunction with Xinguodu products. LICENSEE MAY NOT COPY, MODIFY, OR TRANSFER THE SOFTWARE IN WHOLE OR IN PART EXCEPT AS EXPRESSLY PROVIDED IN THIS AGREEMENT. Licensee may not decompile, disassemble, or in any other manner attempt reverse reverse engineer the Software.

Licensee shall not tamper with, bypass, or alter any security features of the software or attempt to do so.

TRANSFER: Licensee may not transfer the Software or license to the Software to another party without the prior written authorization of the Licensor. If Licensee transfers the Software without authorization, all rights granted under this Agreement are automatically terminated.

COPYRIGHT: The Software is copyrighted. Licensee may not copy the Software except for archival purposes or to load for business operations. All other copies of the Software are in violation of this Agreement.

TERM: This Agreement is in effect as long as Licensee continues the use of the Software. The Licensor also reserves the right to terminate this Agreement if Licensee fails to comply with any of the terms, conditions, or restrictions contained herein. Should Licensor terminate this Agreement due to Licensee's failure to comply, Licensee agrees to return the Software to Licensor. Receipt of returned Software by the Licensor shall mark the termination.

LIMITED WARRANTY: Licensor warrants to the Licensee that the disk(s) or other media on which the Software is recorded are free from defects in material or workmanship under normal use.

THE SOFTWARE IS PROVIDED AS IS. LICENSOR MAKES NO OTHER WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Because of the diversity of conditions and PC hardware under which the Software may be used, Licensor does not warrant that the software will meet Licensee specifications or that the operation of the Software will be uninterrupted or free of errors.

IN NO EVENT WILL LICENSOR BE LIABLE FOR ANY DAMAGES, INCLUDING ANY LOST PROFITS, LOST SAVINGS, OR OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE, OR INABILITY TO USE, THE SOFTWARE. Licensee's sole remedy in the event of a defect in material or workmanship is expressly limited to replacement of the Software disk(s), if applicable.

GOVERNING LAW: If any provision of this Agreement is found to be unlawful, void, or unenforceable, that provision shall be removed from consideration under this Agreement and will not affect the enforceability of any of the remaining provisions. This Agreement shall be governed by the Laws of the State of California and shall inure to the benefit of Shenzhen Xinguodu Technology Co., Ltd, its successors or assigns.

ACKNOWLEDGMENT: LICENSEE ACKNOWLEDGES THAT HE HAS READ THIS AGREEMENT, UNDERSTANDS ALL OF ITS TERMS, CONDITIONS, AND RESTRICTIONS, AND AGREES TO BE BOUND BY THEM. LICENSEE ALSO AGREES THAT THIS AGREEMENT SUPERSEDES ANY AND ALL VERBAL AND WRITTEN COMMUNICATIONS BETWEEN LICENSOR AND LICENSEE OR THEIR ASSIGNS RELATING TO THE SUBJECT MATTER OF THIS AGREEMENT.
QUESTIONS REGARDING THIS AGREEMENT SHOULD BE ADDRESSED IN WRITING TO Shenzhen Xinguodu Technology Co., Ltd, ATTENTION: CUSTOMER SUPPORT, AT THE ADDRESS LISTED IN THIS DOCUMENT OR E-MAILED TO SUPPORT@XINGUODU.COM.

Table of Content

1 Introduction

This document helps developers develop third-party applications on N5 devices. The company provides API interface in the form of jar package, based on the API interface, the developer can conveniently and efficiently develop third-party applications to meet the personalized needs of developers.

1.1 Demo Description

The demo program demonstrates how to use the aar package to call API's various interfaces to meet the needs of the developer.

1.2 The Term

SdkResult returns the value class. The fields in the class define all the Return Values in the document. All fields are described in the Appendix.

1.3 SDK Content

File	Description
nexgo-smart-sdk-vx.x.x.aar	aar package API interface

1.4 System Requirements

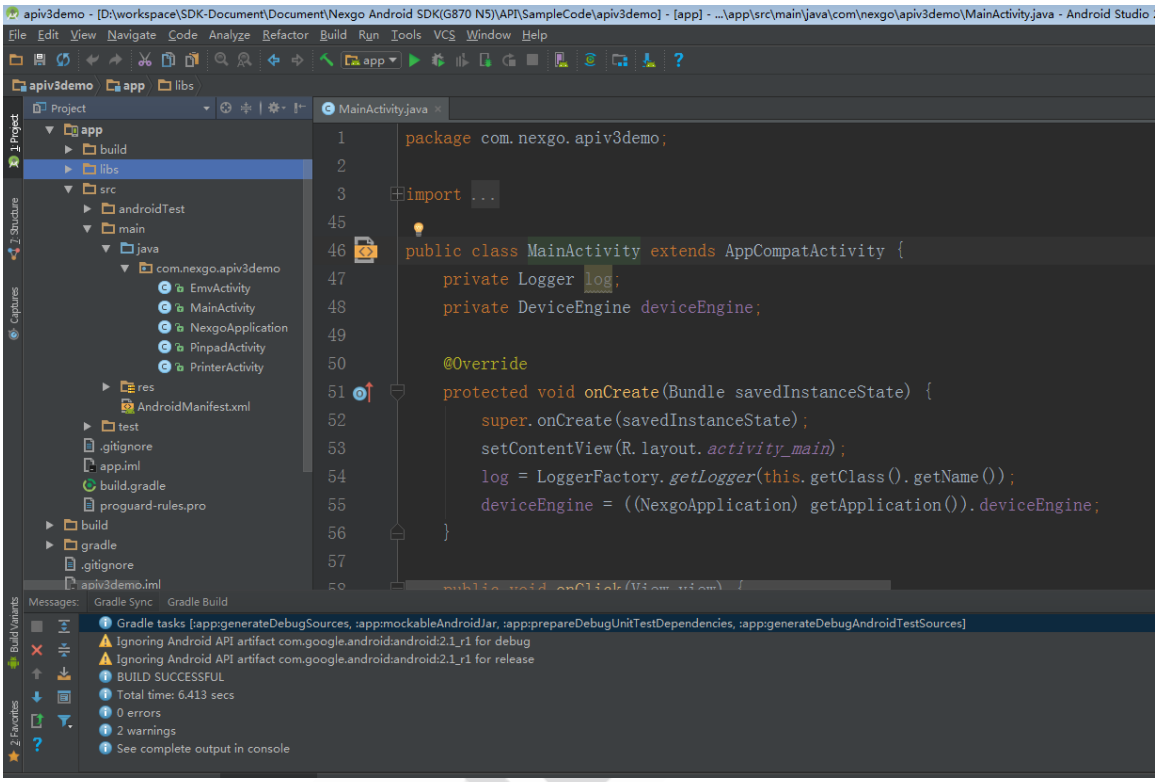
Development environment: Android Studio 2.0 or later

N5 Operating System Version: 5.1.1

2 How to Create a Project

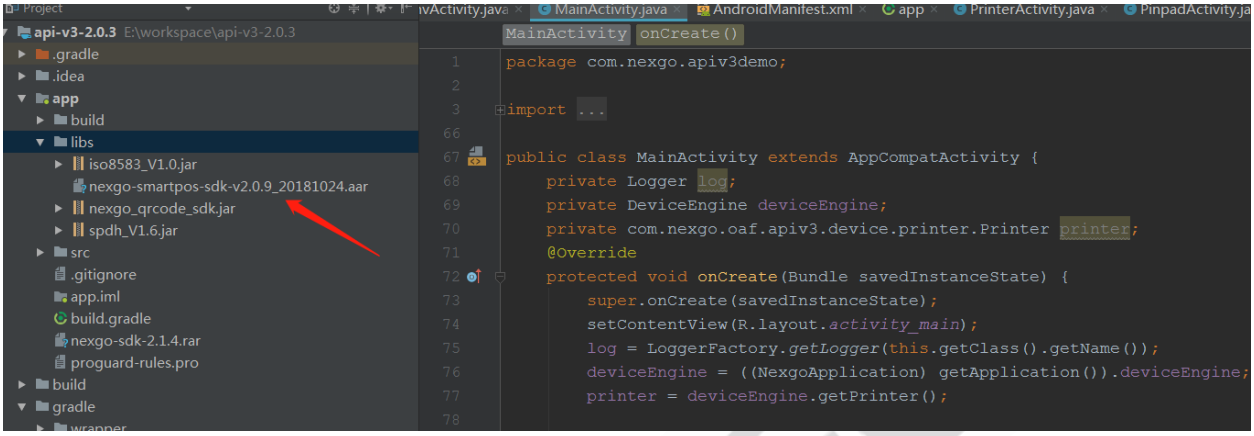
To add the jar package to the developer project, follow these steps:

1) With Android studio to create or open the customer's project.



2) Copy nexgo-smart-sdk-vx.x.x.aar to libs / directory, configure project build.gradle to load aar package.

```
repositories{
    flatDir{
        dirs 'libs'
    }
}
 dependencies{
complie(name: 'nexgo-smartpos-sdk-vx.x.x', ext: 'aar')
}
```



3) Get the global object of the device operation.

   DeviceEngine deviceEngine = APIProxy getDeviceEngine (this) ;

4) Get the object of the device sub-module, and operate the interface

   For example: get buzzer operation object, ring 500 milliseconds.

   final Beeper beeper = deviceEngine getBeeper () .;

   beeper.beep (500);

3 Class methods

The following is divided into 10 categories, 5 global methods. First, get the object of each class, then call the member method in the class.

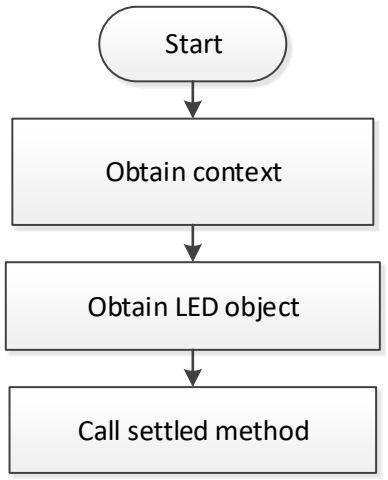
3.1 LED class

LED class is responsible for managing POS LED lights.

Get the object of the LED class:

LEDDriver ledDriver = deviceEngine.getLEDDriver ();

This module operates using the basic flow chart:



3.1.1 SetLed

Drive POS red, green, yellow, blue light switch.

Public void setLed (LightModeEnum light, boolean isOn);

Parameters:

Parameter	Description
light	Enumerated type red, green, yellow, blue LED lights
isOn	True: on, false: off

LightModeEnum

Enumeration Name	Description
RED	Red light
GREEN	Green light
YELLOW	Yellow light
BLUE	Blue light
DECORATIVE_LIGHT	Decorative light for N96 only

Return Value: None

3.2 Printer class

The printer class is responsible for managing POS printers.

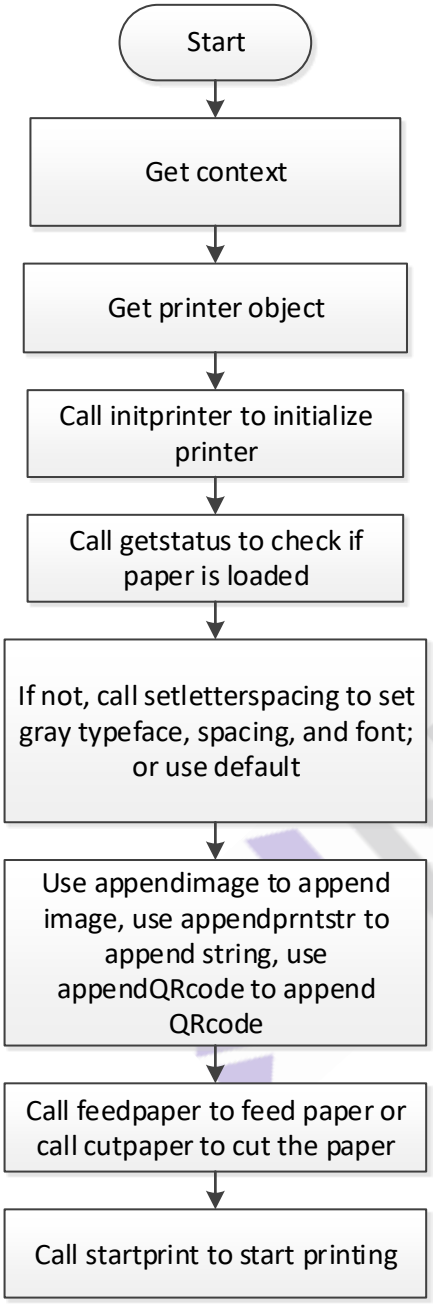
Get the object of the printer class:

Printer printer = deviceEngine getPrinter (). ;

The following table shows the Return Values supported by the method of the printer class:

Constant Name	Constant Value	Description
Printer_Base_Error	-1000	
Printer_Print_Fail	Printer_Base_Error – 1	Print failed
Printer_AddPrnStr_Fail	Printer_Base_Error – 2	Setting string buffer failed
Printer_AddImg_Fail	Printer_Base_Error – 3	Setting image buffer failed
Printer_Busy	Printer_Base_Error – 4	The printer is busy
Printer_PaperLack	Printer_Base_Error – 5	The printer is out of paper
Printer_Wrong_Package	Printer_Base_Error – 6	Print packet is wrong
Printer_Fault	Printer_Base_Error – 7	Printer failure
Printer_TooHot	Printer_Base_Error – 8	The printer is overheating
Printer_UnFinished	Printer_Base_Error – 9	The print is not complete
Printer_NoFontLib	Printer_Base_Error – 10	The printer does not have a font
Printer_OutOfMemory	Printer_Base_Error – 11	The packet is too long
Printer_Other_Error	Printer_Base_Error-999	Other exception error

This module operates using the basic flow chart:



3.2.1 initPrinter

Initialize the printer.

public int initPrinter ();

Parameters: None

Return Value:

SdkResult.Success success

3.2.2 getStatus

Get the printer status.

Public int getStatus ();

Parameters: None

Return Value:

SdkResult.Success	print successful
SdkResult. Printer_UnFinished	print is not complete
SdkResult. Printer_PaperLack	printer is out of paper
SdkResult. Printer_Too_Hot	printer is overheating
SdkResult. Printer_Fail	print failed
SdkResult.Fail	other errors

3.2.3 appendImage

Append bitmap.

Public int appendImage (Bitmap bitmap, AlignEnum align);

Parameters:

Parameter	Description
bitmap	Bitmap data
align	Enumerated type of alignment

AlignEnum

Enumeration Name	Description
LEFT	Left alignment
RIGHT	Right alignment
CENTER	Centered

Return Value:

SdkResult.Success success

SdkResult.Printer_AddImg_Fail additional picture failure

3.2.4 appendPrnStr

Add text to print buffer.

Public int appendPrnStr (String text, int fontsize, AlignEnum align,
Boolean isBoldFont);

Parameters:

Parameter	Description
text	The string data to be added
fontSize	Font Size small: 16; normal: 20; large: 24; x-large: 32
align	Enumerated type of alignment
isBoldFont	Whether bold, true: yes, false: no

AlignEnum

Enumeration Name	Description
LEFT	Left alignment
RIGHT	Right alignment
CENTER	Centered

Return Value:

SdkResult.Success success

SdkResult.Printer_Wrong_Package print packet format error

SdkResult.Printer_AddPrnStr_Fail string buffer is set to fail

3.2.5 appendPrnStr

Add text to print buffer.

public int appendPrnStr(String text, int fontsize, AlignEnum align, LineOptionEntity ops);
Parameters:

Parameter	Description
text	The string data to be added
fontSize	Font Size small: 16; normal: 20; large: 24; x-large:

	32
align	Enumerated type of alignment
ops	LineOptionEntity: additional option

AlignEnum

Enumeration Name	Description
LEFT	Left alignment
RIGHT	Right alignment
CENTER	Centered

LineOptionEntity

attribute	Description
boolean isUnderline	Print underline: true: yes; false:no
int marginLeft	Left margin size
Boolean isZoomX	lateral magnification
Boolean isZoomY	longitudinal magnification
Boolean isBold	Whether bold, true: yes, false: no
boolean isInvert;	Print invert; false- not inverted; true-invert; default is false It works with Vector Print API only.

Return Value:

SdkResult.Success success

SdkResult.Printer_Wrong_Package print packet format error

SdkResult.Printer_AddPrnStr_Fail string buffer is set to fail

3.2.6 appendPrnStr

Append string both sides at the same time to print buffer.

Public int appendPrnStr (String leftText, String rightText, int fontsize, Boolean isBoldFont);

Parameters:

Parameter	Description
leftText	Left alignment data
rightText	Right alignment data
fontsize	Font Size small: 16; normal: 20; large: 24; x-large: 32
isBoldFont	Whether bold, true: yes, false: no

AlignEnum

Enumeration Name	Description

LEFT	Left alignment
RIGHT	Right alignment
CENTER	Centered

Return Value:

SdkResult.Success success

SdkResult.Printer_Wrong_Package print packet format error

SdkResult.Printer_AddPrnStr_Fail string buffer is set to fail

3.2.7 appendPrnStr

Append string both sides at the same time to print buffer.

public int appendPrnStr(String leftText, String rightText, int fontsize, LineOptionEntity ops);
Parameters:

Parameter	Description
leftText	Left alignment data
rightText	Right alignment data
fontsize	Font Size small: 16; normal: 20; large: 24; x-large: 32
ops	LineOptionEntity :additional option

LineOptionEntity

attribute	**Description**
boolean isUnderline	Print underline: true: yes; false:no
int marginLeft	Left margin size
Boolean isZoomX	lateral magnification
Boolean isZoomY	longitudinal magnification
Boolean isBold	Whether bold, true: yes, false: no
boolean isInvert;	Print invert; false- not inverted; true-invert; default is false It works with Vector Print API only.

Return Value:

SdkResult.Success success

SdkResult.Printer_Wrong_Package print packet format error

SdkResult.Printer_AddPrnStr_Fail string buffer is set to fail

3.2.8 appendBarcode

Append barcode to print buffer.

public int appendBarcode(String content, int height, int margin, int scale, BarcodeFormatEnum barcodeFormat, AlignEnum align);

Parameters:

Parameter	Description
content	Generates Barcode data
height	Generates Barcode height for printing; ranges greater than 0. Suggest 50
margin	Margin, suggest 0
scale	Scale, >= 1, suggest 2
barcodeFormat	Refer to BarcodeFormatEnum
align	Alignment

BarcodeFormatEnum

Format Name	Description
AZTEC	
CODABAR	
CODE_39	
CODE_93	
CODE_128	
DATA_MATRIX	
EAN_8	
EAN_13	
ITF	
MAXICODE	
PDF_417	
QR_CODE	
RSS_14	
RSS_EXPANDED	
UPC_A	
UPC_E	
UPC_EAN_EXTENSION	

AlignEnum

Enumeration Name	Description
LEFT	Left alignment
RIGHT	Right alignment
CENTER	Centered

Return Value:

SdkResult.Success success

SdkResult.Printer_AddImg_Fail failure

3.2.9 appendQRCode

Append QR code to print buffer.

public int appendQRcode (String content, int height, AlignEnum align);

Parameters:

Parameter	Description
content	Generates QR code data
height	Generates QR code height for printing; ranges greater than 0
align	Alignment

AlignEnum

Enumeration Name	Description
LEFT	Left alignment
RIGHT	Right alignment
CENTER	Centered

Return Value:

SdkResult.Success success

SdkResult.Printer_AddImg_Fail failure

3.2.10 appendQRcode

Append QR code to print buffer.

public int appendQRcode(String content, int height, int version, int level, AlignEnum align);

Parameters:

Parameter	Description
content	Generates QR code data
height	Generates QR code height for printing; ranges greater than 0
version	QR code version is 1-40
level	Error correction level, from low to high, 0-3
align	Alignment

AlignEnum

Enumeration Name	Description
LEFT	Left alignment
RIGHT	Right alignment
CENTER	Centered

Return Value:

SdkResult.Success success

SdkResult.Printer_AddImg_Fail failure

3.2.11 feedPaper

Feed paper.

Public void feedPaper (int value);

Parameters:

Parameter	Description
value	Paper length is in pixels; range of greater than or equal to 0; if the user has not set, the default value equals 0

Return Value: None

3.2.12 startPrint

Start printing with print buffer

public int startPrint (boolean rollPaperEnd, OnPrintListener listener);

Parameters:

Parameter	Description
rollPaperEnd	Advance to the end of the paper automatically; true: yes, false: no

listener	The callback interface after printing is complete

Return Value:

SdkResult.Success operation is successful; listener can successfully callback

SdkResult.Printer_Busy printer is busy

SdkResult.Printer_Print_Fail print data is empty

SdkResult.Param_In_Invalid illegal Parameter

3.2.13 setLetterSpacing

Set the spacing between the print order lines.

public void setLetterSpacing (int value);

Parameters:

Parameter	Description
value	Line spacing is in pixels; the default value equals 4

Return Value: None

3.2.14 setGray

Set the grayscale.

public void setGray (GrayLevelEnum level);

Parameters:

Parameter	Description
level	Establish gray value; if the user has not set, the default value is LEVEL_0. The higher the grayscale, the darker the print font, the slower the print speed.

GrayLevelEnum

Enumeration Name	Description
LEVEL_0	Printer gray level 0
LEVEL_1	Printer gray level 1
LEVEL_2	Printer gray level 2
LEVEL_3	Printer gray level 3
LEVEL_4	Printer gray level 4

Return Value: None

3.2.15 setTypeface

Set the font type. Recommend using Typeface.DEFAULT

public void setTypeface (Typeface typeface);

Parameters:

Parameter	Description
typeface	Android SDK Typeface font type; user can use the default value: Typeface.DEFAULT

Return Value: None

3.2.16 setInvert

Set print invert, it works with Vector Print API and text print only. Does not support Dot matrix API.

It is a global configuration, all print lines(text) will be invert.

public void setInvert(boolean invert);

Parameters:

Parameter	Description
invert	False: not inverted; default is false. True: invert

Return Value: None

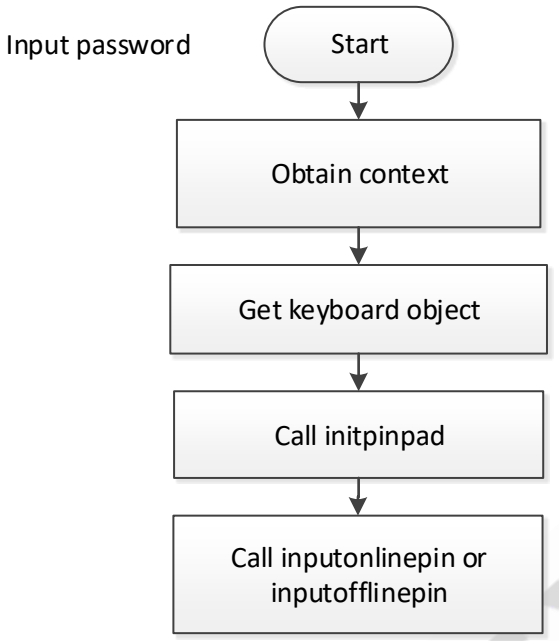
3.3 Pinpad Class

The password keyboard class is responsible for managing the POS password keyboard.

Get the object of the password keyboard class:

. PinPad pinpad = deviceEngine getPinPad () ;

This module operates using the basic flow chart:



3.3.1 initPinPad

Initialize the password keyboard.

Public int initPinPad (PinPadTypeEnum ppType);

Parameters:

Parameter	Description
ppType	PinPadTypeEnum Enum type Password keyboard type; currently only supports built-in password keyboard

PinPadTypeEnum

Enumeration Name	Description
INTERNAL	Built-in password keyboard, default value is INTERNAL
EXTERNAL	External password keyboard， such as G101, K110 Deprecated, Not Recommended. Please use ExtPinPad

Return Value:

SdkResult.Success success

SdkResult.Fail fail

3.3.2 setAlgorithmMode

Set PinPad work in DUKPT model or classical model. Default is DES mode.

Public void setAlgorithmMode(AlgorithmModeEnum algMode);

AlgorithmModeEnum

Enumeration Name	Description
DES	DES mode(MK/SK mode, Includes DES/TDES)
SM4	PBOC use it(China market)
DUKPT	DUKPT mode

Note: this configuration is global configuration; if you want to switch to other mode, you need to call this API to config again.

For example, if you want to use MK/SK mode, you need to config:

setAlgorithmMode(AlgorithmModeEnum.DES)

later, if you want to switch to DUKPT mode, you need to config:

setAlgorithmMode(AlgorithmModeEnum.DUKPT)

3.3.3 setDukptAlgorithmMode

Set DUKPT Algorithm DES or AES. Default is DES mode. Only apply to DUKPT mode.

If set AES, then need to call setDukptAESGenerateKeyType to config AES key type with AES 128, AES 192, AES256

Public void setDukptAlgorithmMode (DukptAlgorithmModeEnum algMode);

DukptAlgorithmModeEnum

Enumeration Name	Description
DES	DES Algorithm, include DES and TDES
AES	AES 128, AES 192, AES 256

Note: this configuration is global configuration; if you want to switch to other mode, you need to call this API to config again.

For example, if you want to set DUKPT AES Algorithm, you need to config:

setDukptAlgorithmMode(DukptAlgorithmModeEnum.AES)

later, if you want to switch to DUKPT DES Algorithm, you need to config:

setDukptAlgorithmMode(DukptAlgorithmModeEnum.DES)

3.3.4 setDukptAESGenerateKeyType

Set DUKPT AES key type, AES128, AES 192, AES256

Public void setDukptAESGenerateKeyType(DukptAESGenerateKeyTypeEnum generateKeyType);

DukptAESGenerateKeyTypeEnum

Enumeration Name	Description
DUKPT_MODE_KEY_TYPE_2TDEA	2TDEA
DUKPT_MODE_KEY_TYPE_3TDEA	3TDEA
DUKPT_MODE_KEY_TYPE_AES128	AES128
DUKPT_MODE_KEY_TYPE_AES192	AES192
DUKPT_MODE_KEY_TYPE_AES256	AES256

Note: this configuration is global configuration; if you want to switch to other mode, you need to call this API to config again.

For example, if you want to set DUKPT AES128 type, you need to config:

setDukptAESGenerateKeyType(DukptAESGenerateKeyTypeEnum.AES128)

later, if you want to switch to DUKPT AES256 type, you need to config:

setDukptAESGenerateKeyType(DukptAESGenerateKeyTypeEnum.AES256)

Below is an example of how to use DUKPT DES:

setAlgorithmMode(AlgorithmModeEnum.DUKPT)

setDukptAlgorithmMode(DukptAlgorithmModeEnum.DES)

Below is an example of how to use DUKPT AES256:

setAlgorithmMode(AlgorithmModeEnum.DUKPT)

setDukptAlgorithmMode(DukptAlgorithmModeEnum.AES)

setDukptAESGenerateKeyType(DukptAESGenerateKeyTypeEnum.AES256)

3.3.5 setCipherMode

Set PinPad Cipher mode, default is ECB mode.

Public void setCipherMode (CipherModeEnum cipherMode);

CipherModeEnum

Enumeration Name	Description
ECB	
CBC	

3.3.6 setCipherInitializationVector

Set PinPad Cipher iv, it is used for CBC Cipher mode.

Public void setCipherInitializationVector (byte[] iv);

3.3.7 setPinKeyboardMode

Set password keyboard mode, default mode is random password keyboard.

public void setPinKeyboardMode(PinKeyboardModeEnum keyboardMode);
Parameters:

Parameter	Description
keyboardMode	Mode, default value RANDOM

PinKeyboardModeEnum

Enumeration Name	Description
RANDOM	Random key board
FIXED	Fixed key board
ACCESSIBILITY	Accessiblity board (Blind PinPad)

Return Value:

None

3.3.8 setPinKeyboardViewMode

Set pin keyboard view mode, default is DEFAULT pin keyboard view.

public void setPinKeyboardViewMode(PinKeyboardViewModeEnum viewMode);
Parameters:

Parameter	Description
viewMode	PinKeyboardViewModeEnum, default value is DEFAULT view

PinKeyboardViewModeEnum

Enumeration Name	Description
DEFAULT	Default view
VIEW_1	View1
VIEW_2	View2
VIEW_3	View3
VIEW_4	Blind PinPad fixed layout view
VIEW_5_ACCESSIBILITY_MODE	Blind PinPad custom layout view

Return Value:

None

3.3.9 writeMKey

Inject the master key(plaintext key)

Public int writeMKey (int mKeyIdx, byte [] keyData, int keyDataLen);

Parameters:

Parameter	Description
mKeyIdx	Master Key Index 0-199
keyData	Plaintext master key data
keyDataLen	The length of the plain key range: 8,16,24

Return Value:

SdkResult.Success success

SdkResult.Param_In_InValid Parameter is not legitimate

SdkResult.PinPad_Dstkey_Idx_Error wrong key index

SdkResult.PinPad_Key_Len_Error wrong key length

SdkResult.Fail other errors

3.3.10 writeMKey

Inject the master key(cipher key)

Public int writeMKey (int mKeyIdx, byte [] keyData, int keyDataLen, int decMKeyIdx);

Parameters:

Parameter	Description
mKeyId	Master Key Index 0-199
keyData	Ciphertext master key data
keyDataLen	Ciphertext master key length should be not less than 8, and must be a multiple of 8 bytes
decMKeyIdx	Decrypt the master key index 0-199

Return Value:

SdkResult.Success success

SdkResult.Param_In_InValid Parameter is not legitimate

SdkResult.PinPad_Dstkey_Idx_Error wrong key index

SdkResult.PinPad_Key_Len_Error wrong key length

SdkResult.PinPad_No_Key_Error key does not exist

SdkResult.Fail other errors

3.3.11 isKeyExist

Whether the master key exists. It is only suitable for TMK/session key, do not suitable for DUKPT.

Public boolean isKeyExist (int mKeyIdx);

Parameters:

Parameter	Description
mKeyId	Master Key Index 0-199

Return Value:

True success, key exist

False failure, key not exist or error.

3.3.12 calcWKeyKCV

Calculate the work key KCV (check value).

Public byte [] calcWKeyKCV (int mKeyIdx, WorkKeyTypeEnum wKeyType);

Parameters:

Parameter	Description
mKeyIdx	Master Key Index 0-199
wKeyType	Working key type

WorkKeyTypeEnum

Enumeration Name	Description
PINKEY	PIN key
MACKEY	MAC key
TDKEY	Track key
ENCRYPTIONKEY	Data encryption key, providing encryption and decryption

Return Value:

Success, return an array of check values

Failure, returning null

3.3.13 writeWKey

Inject work key.

Public int writeWKey (int mKeyIdx, WorkKeyTypeEnum wKeyType, byte [] keyData, int keyDataLen);

Parameters:

Parameter	Description
mKeyIdx	Master key index number 0-199
wKeyType	Working key type
keyData	Working key cipher text data
keyDataLen	Working key cipher text length

WorkKeyTypeEnum

Enumeration Name	Description
PINKEY	PIN key
MACKEY	MAC key

TDKEY	Track key
ENCRYPTIONKEY	Data encryption key, providing encryption and decryption

Return Value:

SdkResult.Success success

SdkResult.Param_In_InValid Parameter is invalid

SdkResult.PinPad_Dstkey_Idx_Error wrong key index object; not within the scope

SdkResult.PinPad_Key_Len_Error wrong key length

SdkResult.Fail other errors

3.3.14 isKeyExist

Whether the work key exists. It is only suitable for TMK/session key, do not suitable for DUKPT.

Public boolean isKeyExist (int mKeyIdx, WorkKeyTypeEnum wKeyType);
Parameters:

Parameter	Description
mKeyId	Master Key Index 0-199
wKeyType	Working key type

WorkKeyTypeEnum

Enumeration Name	Description
PINKEY	PIN key
MACKEY	MAC key
TDKEY	Track key
ENCRYPTIONKEY	Data encryption key, providing encryption and decryption

Return Value:

True success ,work key exist

False failure, work key not exist, or error

3.3.15 calcByWkey

public byte[] calcByWKey(int mKeyIdx, WorkKeyTypeEnum wKeyType, byte[] data, int dataLen, CalcModeEnum calcMode);
Parameters:

Parameter	Description
mKeyIdx	Master Key Index 0-199
wKeyType	Working key type
data	Input data
dataLen	Data length
calcMode	encryption or decryption

CalcModeEnum

Enumeration Name	Description
ENCRYPT	ENCRYPT mode
DECRYPT	DECRYPT mode

3.3.16 desByWKey

Work key encryption and decryption.

Public byte [] desByWKey (int mKeyIdx, WorkKeyTypeEnum wKeyType, byte [] data, int dataLen, DesKeyModeEnum keyMode, CalcModeEnum calcMode);

Parameters:

Parameter	Description
mKeyIdx	Master Key Index 0-199
wKeyType	Working key type
data	Input data
dataLen	Data length
keyMode	Key Mode
calcMode	Calc Mode

WorkKeyTypeEnum

Enumeration Name	Description
PINKEY	PIN key
MACKEY	MAC key
TDKEY	Track key
ENCRYPTIONKEY	Data encryption key, providing encryption and decryption

DesKeyModeEnum

Enumeration Name	Description
KEY_ALL	
KEY_FIRST	Specify the type of algorithm, key double length is used to

	do the first 8 bytes DES operation
KEY_LAST	Specify the type of algorithm, key double length is used to do the last 8 bytes DES operation

CalcModeEnum

Enumeration Name	Description
ENCRYPT	Encrypt
DECRYPT	Decrypt

Return Value:

Success returns the computed array

Failure, returns null

3.3.17 encryptTrackData

Use TDK work key to Encrypt track data.

public byte[] encryptTrackData(int mKeyId, byte[] trackData, int trackDataLen);
Parameters:

Parameter	Description
mKeyId	Master key index 0-199
trackData	Track data
trackDataLen	Len of track data

Return Value:

Success returns the computed array

Failure, returns null

3.3.18 calcMac

Use TAK work key to Calculate MAC.

Public byte [] calcMac (int mKeyIdx, MacAlgorithmModeEnum macAlgMode, byte [] data);

Parameters:

Parameter	Description
mKeyIdx	Master Key Index 0-199
macAlgMode	MAC algorithm approach

data	Input data

MacAlgorithmModeEnum

Enumeration Name	Description
ECB	ECB Algorithm
X99	ANSI X9.9 Encryption Algorithm
X919	ANSI X9.19 Encryption Algorithm
MAC9606	MAC 9606
AES_CMAC	AES CMAC

Return Value:

Success returns the computed array

Failure, returning null

3.3.19 calcMac

Calculate MAC.

Public byte [] calcMac (int mKeyIdx, MacAlgorithmModeEnum macAlgMode, DesKeyModeEnum keyMode , byte [] data);

Parameters:

Parameter	Description
mKeyIdx	Master Key Index 0-199
macAlgMode	MAC algorithm approach
desAlgMode	Algorithm type
data	Input data

MacAlgorithmModeEnum

Enumeration Name	Description
ECB	ECB Algorithm
X99	ANSI X9.9 Encryption Algorithm
X919	ANSI X9.19 Encryption Algorithm
MAC9606	MAC 9606
AES_CMAC	AES CMAC

DesKeyModeEnum

Enumeration Name	Description
KEY_ALL	
KEY_FIRST	Do des with the first 8 bytes of the key
KEY_LAST	Do des with the last 8 bytes of the key

Return Value:

Success returns the computed array

Failure, returning null

3.3.20 calcMac(DUKPT)

byte[] calcMac(int mKeyIdx, MacAlgorithmModeEnum macAlgMode, DukptKeyModeEnum keyMode, byte[] data);

Parameters:

Parameter	Description
mKeyId	Key Index 0-19(DUKPT only support 0-19 key index)
macAlgMode	Mac Alg mode
keyMode	Key mode
data	Data, lack of an integer multiple of 8, after the meeting 0x00 or …the fill data is decided by application

MacAlgorithmModeEnum

Enumeration Name	Description
ECB	
CBC	
X919	
MAC9606	
AES_CMAC	Does not work for DUKPT

DukptKeyModeEnum

Enumeration Name	Description
REQUEST	Request mode
RESPONSE	Response mode

3.3.21 encryptByMKey

Master key encryption.

Public byte [] encryptByMKey (int mKeyId, byte [] data, int dataLen);
Parameters:

Parameter	Description
mKeyId	Master Key Index 0-199
data	Data, lack of an integer multiple of 8, after the meeting 0x00
dataLen	Length, maximum 1024 bytes
desAlgMode	DES algorithm type

DesAlgorithmModeEnum

Return Value:

Success returns the computed array

Failure, returning null

3.3.22 setPinpadLayout

Set the password keyboard layout. After this method is called, when inputOnlinePin or inputOfflinePin is called , the layout of the password keyboard will be drawned by the app layer itself, without using the system default password keyboard interface.

public byte[] setPinpadLayout(PinpadLayoutEntity pinpadLayout);
Parameters:

Parameter	Description
pinpadLayout	Coordinates of 10 digital keys and 3 function keys

PinpadLayoutEntity

attribute	Description
Rect key1	The coordinate of the number key "1"
Rect key2	The coordinate of the number key "2"
Rect key3	The coordinate of the number key "3"
Rect key4	The coordinate of the number key "4"
Rect key5	The coordinate of the number key "5"
Rect key6	The coordinate of the number key "6"
Rect key7	The coordinate of the number key "7"
Rect key8	The coordinate of the number key "8"
Rect key9	The coordinate of the number key "9"

Rect key10	The coordinate of the number key "0"
Rect keyCancel	The coordinate of the key "cancel"
Rect keyConfirm	The coordinate of the key "confirm"
Rect keyClear	The coordinate of the key "clear"

Return Value:

Byte[]	Returns 0-9 digits for 10 key successfully
null	Failed

3.3.23 setBlindPinpadLayout

Set the blind pin keyboard layout. After this method is called, when inputOnlinePin or inputOfflinePin is called, the layout of the blind pin keyboard will be drawed by the app layer itself, without using the system default blind pin keyboard interface.

public byte[] setBlindPinpadLayout(PinpadLayoutEntity pinpadLayout);

Parameters:

Parameter	Description
pinpadLayout	Coordinates of 10 digital keys and 3 function keys

PinpadLayoutEntity

attribute	Description
Rect key1	The coordinate of the number key "1"
Rect key2	The coordinate of the number key "2"
Rect key3	The coordinate of the number key "3"
Rect key4	The coordinate of the number key "4"
Rect key5	The coordinate of the number key "5"
Rect key6	The coordinate of the number key "6"
Rect key7	The coordinate of the number key "7"
Rect key8	The coordinate of the number key "8"
Rect key9	The coordinate of the number key "9"
Rect key10	The coordinate of the number key "0"
Rect keyCancel	Blind PinPad cancel function and clear function will use same one button, when the pin input, the button is clear function, when no pin input, the button is cancel function. The SDK will use coordinate of “clear”, don’t use this one.
Rect keyConfirm	The coordinate of the key "confirm"
Rect keyClear	The coordinate of the key "clear"

Return Value:

Byte[]	Returns 0-9 digits for 10 key successfully
null	Failed

3.3.24 setBlindPinpadOnHoverListener

Set the blind pinpad onhover listener, it will callback cancel button and ok button touch event, so you can play your sound when it callback.

void setBlindPinPadOnHoverListener(OnHoverListener listener);

Parameters:

Parameter	Description
listener	OnHoverListener

Return Value:

None

3.3.25 inputOnlinePin(Deprecated)

Enter the online PIN. Not recommended

Public int inputOnlinePin (int [] pinLen, int timeout, byte [] panBlock, int mKeyId, PinAlgorithmModeEnum pinAlgMode, OnPinPadInputListener listener);

Parameters:

Parameter	Description
pinLen	The length of the support, such as{0x00,0x04,0x05,0x06,0x07,0x08,0x09,0x0a, 0x0b, 0x0c}
timeout	Enter a timeout in seconds; recommended value 60
panBlock	Card number, asc coding
mKeyId	Master Key Index 0-199(if DUKPT, is 0-19)
pinAlgMode	PIN encryption algorithm mode
listener	Monitor callback interface

PinAlgorithmModeEnum

Enumeration Name	Description
ISO9564FMT0	Format 0, Currently only supports Format 0
ISO9564FMT1	Format 0, Currently only supports Format 0
ISO9564FMT2	Format 0, Currently only supports Format 0
ISO9564FMT3	Format 0, Currently only supports Format 0

Return Value:

SdkResult.Success successful execution listener callback interface

SdkResult.Param_In_InValid illegal Parameter

SdkResult.Fail other errors

3.3.26 inputOnlinePin

Enter the online PIN

Public int inputOnlinePin (int [] pinLen, int timeout, String cardNumber, int mKeyId,
PinAlgorithmModeEnum pinAlgMode, OnPinPadInputListener listener);

Parameters:

Parameter	Description
pinLen	The length of the support, such as{0x00,0x04,0x05,0x06,0x07,0x08,0x09,0x0a, 0x0b, 0x0c}
timeout	Enter a timeout in seconds; recommended value 60
cardNumber	Card number
mKeyId	Master Key Index 0-199(if DUKPT, is 0-19)
pinAlgMode	PIN encryption algorithm mode
listener	Monitor callback interface

PinAlgorithmModeEnum

Enumeration Name	Description
ISO9564FMT0	Format 0
ISO9564FMT1	Format 1
ISO9564FMT2	Format 2
ISO9564FMT3	Format 3
ISO9564FMT4	Format 4

Return Value:

SdkResult.Success successful execution listener callback interface

SdkResult.Param_In_InValid illegal Parameter

SdkResult.Fail other errors

3.3.27 inputPinExternal

Enter the PIN with the external Pinpad, such as G101,K110. Need to set Pinpad type to External.

pinpad.initPinPad (PinPadTypeEnum. EXTERNAL);

Public int inputPinExternal (final int mKeyIdx, final int minLen, final int maxLen, final String pan, final int timeout, final OnPinPadInputListener listener);

Parameters:

Parameter	Description
mKeyIdx	Master Key Index 0-19(does not support DUKPT)
minLen	Min length of the PIN
maxLen	Max length of the PIN
pan	Card number
timeout	Input PIN timeout, suggest use 60s
listener	Monitor callback interface

Return Value:

SdkResult.Success successful execution listener callback interface

SdkResult.Param_In_InValid illegal Parameter

SdkResult.Fail other errors

Note: inputPinExternal support online PIN and Offline PIN. But it does not support DUKPT mode.

3.3.28 inputOfflinePin

Enter the offline PIN(offline plaintext pin, or offline cipher pin).

Public int inputOfflinePin (int [] pinLen, int timeout, OnPinPadInputListener listener);

Parameters:

Parameter	Description
pinLen	The length of the support, such as{0x00,0x04,0x05,0x06,0x07,0x08,0x09,0x0a, 0x0b, 0x0c}
timeout	Enter a timeout in seconds; recommended value 60
listener	Monitor callback interface

Return Value:

SdkResult.Success successful execution listener callback interface

SdkResult.Param_In_InValid illegal Parameter

SdkResult.Fail other errors

3.3.29 isInputting

Whether the keyboard is typing.

Public boolean isInputting ();
Parameters: None

Return Value:

True success

False failure

3.3.30 cancelInput

Cancel the keyboard input.

Public void cancelInput ();
Parameters: None

Return Value: None

3.3.31 format

Format the key area.

Public boolean format ();
Parameters: None

Return Value:

True success

False failure

3.3.32 deleteMKey

Clear the master key.

Public boolean deleteMKey (int mKeyId);
Parameters:

Parameter	Description

mKeyId	Master Key Index 0-199

Return Value:

True success

False failure

3.3.33 dukptKeyInject

Inject BDK(or IPEK) and KSN for DUKPT.

Public int dukptKeyInject(int mKeyIdx, DukptKeyTypeEnum keyType, byte[] keyData, int keyDataLen, byte[] ksn);
Parameters:

Parameter	Description
mKeyIdx	Key Index 0-19
keyType	Key type
keyData	BDK
keyDataLen	BDK length
ksn	KSN

DukptKeyTypeEnum

Enumeration Name	Description
BDK	BDK
IPEK	IPEK

Return value:

SdkResult.Success,

SdkResult.Fail,

SdkResult.PinPad_KeyIdx_Error,

SdkResult.Param_In_Invalid

3.3.34 dukptCipherKeyInject

Inject cipher BDK(or IPEK) and KSN for DUKPT.

Public int dukptCipherKeyInject(int dukptKeyIdx, int decKeyIdx, WorkKeyTypeEnum workKeyTypeEnum, DukptKeyTypeEnum keyType, CalcModeEnum calcModeEnum, byte[] keyCipherData, byte[] ksn);
Parameters:

Parameter	Description
dukptKeyIdx	Key Index 0-19
decKeyIdx	Decrypt Key index 0-199(use MK/SK key to decrypt the cipher BDK/IPEK and inject)
workKeyTypeEnum	If workKeyTypeEnum != null, use work key to decrypt and inject cipher BDK/IPEK ; If workKeyTypeEnum == null, use master key to decrypt and inject cipher BDK/IPEK ;
keyType	BDK/IPEK
calcModeEnum	Encryption or decryption
keyCipherData	Cipher BDK/IPEK
ksn	KSN

WorkKeyTypeEnum

Enumeration Name	Description
PINKEY	PIN key
MACKEY	MAC key
TDKEY	Track key
ENCRYPTIONKEY	Data encryption key, providing encryption and decryption

DukptKeyTypeEnum

Enumeration Name	Description
BDK	BDK
IPEK	IPEK

CalcModeEnum

Enumeration Name	Description
ENCRYPT	ENCRYPT mode
DECRYPT	DECRYPT mode

Return value:

SdkResult.Success,

SdkResult.Fail,

SdkResult.PinPad_KeyIdx_Error,

SdkResult.Param_In_Invalid

3.3.35 dukptKsnIncrease

Use it to increase ksn, otherwise the ksn will not change.

Public void dukptKsnIncrease(int mKeyIdx);
Parameters:

Parameter	Description
mKeyIdx	Key Index 0-19

3.3.36 DukptKsnIncreaseV2

Use it to increase ksn, otherwise the ksn will not change. This API will return the KSN increase status, such as Key not exist, or KSN exceed limit.

Public int dukptKsnIncrease(int mKeyIdx);
Parameters:

Parameter	Description
mKeyIdx	Key Index 0-19

Return value:

SdkResult.Success,

SdkResult.Param_In_Invalid, param incorrect

SdkResult.PinPad_No_Key_Error, key not exist

SdkResult. PinPad_KSN_Exceed_Limit, KSN exceed limit. Please reload keys

SdkResult.Fail, other errors

3.3.37 dukptCurrentKsn

Get current Ksn value.

Public byte[] dukptCurrentKsn(int mKeyIdx);
Parameters:

Parameter	Description
mKeyIdx	Key Index 0-19

If key exist, it will return the KSN
Otherwise, it will return null.
This API can be used to check if the DUKPT key exists or not.

3.3.38 dukptEncrypt

Encrypt data in dukpt model.

Public byte[] dukptEncrypt(int mKeyIdx, DukptKeyModeEnum keyMode, byte[] data, int dataLen);
Parameters:

Parameter	Description
mKeyIdx	Key Index 0-19
keyMode	Encrypt model
data	Encrypt data
dataLen	Encrypt data's length

DukptKeyModeEnum

Enumeration Name	Description
REQUEST	
RESPONSE	

Return value:

Bytes Array,

Null

3.3.39 dukptEncrypt

Encrypt data in dukpt model.

Public byte[] dukptEncrypt(int mKeyIdx, DukptKeyModeEnum keyMode, byte[] data, int dataLen, DesAlgorithmModeEnum desMode, byte[] iv);
Parameters:

Parameter	Description
mKeyIdx	Key Index 0-19
keyMode	Encrypt model
data	Encrypt data

dataLen	Encrypt data's length
desMode	Use ECB or CBC
iv	Iv for CBC mode

DukptKeyModeEnum

Enumeration Name	Description
REQUEST	
RESPONSE	

DesAlgorithmModeEnum

Enumeration Name	Description
ECB	
CBC	

Return value:

Bytes Array,

Null

3.3.40 injectKBPK

Inject KBPK, this API should work together with injectTr31Key

int injectKBPK(int kbpkIndex, byte[] keyData, int keyDataLen);

Parameters:

Parameter	Description
kbpkIndex	KBPK Key Index 0-9
keyData	Plaintext KBPK
keyDataLen	The length of the KBPK

Return Value:

SdkResult.Success success

SdkResult.Param_In_InValid Parameter is not legitimate

PinPad_Open_Or_Close_Error Pinpad open or close failed

SdkResult.Fail other errors

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.3.41 injectTr31Key

Inject TR31 key block, must call API injectKBPK to inject KBPK first

int injectTr31Key(int kbpkIndex, int mKeyIdx, byte[] tr31Block, byte[] kcv);

Parameters:

Parameter	Description
kbpkIndex	KBPK Key Index 0-9
mKeyIdx	Key index(0-99 for MK/SK, 0-19 for IPEK)
tr31Block	Standard TR31 block data, can support MK/SK, DUKPT
byte[] kcv	Injected key check value

Return Value:

SdkResult.Success success

SdkResult.Param_In_InValid Parameter is not legitimate

PinPad_Open_Or_Close_Error Pinpad open or close failed

SdkResult.Fail other errors

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.3.42 injectTr31KeyWithTMK

Inject TR31 key block, the KBPK should be terminal master key(TMK). For example, writeMKey to inject the TMK, or the TMK is pre-injected in factory or by RKI.

int injectTr31KeyWithTMK(int tmkIndex, int mKeyIdx, byte[] tr31Block, byte[] kcv);

Parameters:

Parameter	Description
tmkIndex	TMK Key Index 0-9
mKeyIdx	Key index(0-99 for MK/SK, 0-19 for IPEK)
tr31Block	Standard TR31 block data, can support MK/SK, DUKPT
byte[] kcv	Injected key check value

Return Value:

SdkResult.Success success

SdkResult.Param_In_InValid Parameter is not legitimate

PinPad_Open_Or_Close_Error Pinpad open or close failed

SdkResult.Fail other errors

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.3.43 dukptDeleteKey

delete DUKPT key

int dukptDeleteKey(boolean isDeleteAllKey, int mKeyIdx);

Parameters:

Parameter	Description
isDeleteAllKey	If delelet all dukpt keys
mKeyIdx	Key index(0-19)

Return Value:

SdkResult.Success success

SdkResult.Param_In_InValid Parameter is not legitimate

PinPad_Open_Or_Close_Error Pinpad open or close failed

SdkResult.Fail other errors

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.3.44 dukptDecryptV2

DUKPT decrypt

Byte[] dukptDecryptV2 (int mKeyIdx, DesAlgorithmModeEnum desMode, DukptKeyModeEnum keyMode, byte[] data, int dataLen, byte[] iv);

Parameters:

Parameter	Description
mKeyIdx	DUKPT key index
desMode	ECB/CBC
keyMode	REQUEST key/RESPONSE Key
data	Wait for decrypt data
dataLen	Data length
iv	Iv for CBC mode

Return Value:

Decrypted data, when it excute successful

Null, when it excute failed

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.3.45 getBlindPinPadCapability

check terminal has blind pinpad capability or not

int getBlindPinPadCapability ();

Return Value:

1: support blind pinpad

0: does not support blind pinpad

Others: error

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.3.46 isP2PEMode

check device is P2PE(SRED) MODE

Boolean isP2PEMode();

Return Value:

true: P2PE mode. SDK will return cipher data(card number, track data,etc) for all the application. for some applications want to get plaintext data, should set the applications in whitelist

false: normal mode

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.3.47 isP2PEAppInWhitelist

check app is in P2PE(SRED) whitelist, if return ture, the app can get the plaintext data(card number, track data,etc)

boolean isP2PEAppInWhitelist();

Return Value:

true: in whitelist. SDK will return plaintext data (card number, track data, etc)

false: SDK will return ciphertext data (card number, track data, etc)

note: This API only works if the terminal is in P2PE mode

3.3.48 setManualEntryTypeMode

Set manual input type, card number, expired data, or CVV

public int setManualEntryTypeMode(P2PEManualEntryTypeEnum manualEntryTypeEnum);
Parameters:

Parameter	Description
manualEntryTypeEnum	Type, card number, expired date or CVV

P2PEManualEntryTypeEnum

Enumeration Name	Description
CARD_NUMBER	card number
EXPIRE_DATE	expired date
CVV	CVV

Return Value:

None

3.3.49 setManualInputCardNoLayout

Set the Manual input cardnumber layout. After this method is called, when manualInput is called, the layout of the keyboard will be drawned by the app layer itself.
This API can use for manual input card number, expired date, CVV.

public byte[] setManualInputCardNoLayout(PinpadLayoutEntity pinpadLayout);

Parameters:

Parameter	Description
pinpadLayout	Coordinates of 10 digital keys and 3 function keys

PinpadLayoutEntity

attribute	Description
Rect key1	The coordinate of the number key "1"
Rect key2	The coordinate of the number key "2"
Rect key3	The coordinate of the number key "3"
Rect key4	The coordinate of the number key "4"
Rect key5	The coordinate of the number key "5"
Rect key6	The coordinate of the number key "6"
Rect key7	The coordinate of the number key "7"
Rect key8	The coordinate of the number key "8"
Rect key9	The coordinate of the number key "9"
Rect key10	The coordinate of the number key "0"
Rect keyCancel	The coordinate of the key "cancel"
Rect keyConfirm	The coordinate of the key "confirm"
Rect keyClear	The coordinate of the key "clear"

Return Value:

Byte[]	Returns 0-9 digits for 10 key successfully
null	Failed

3.3.50 manualInput

Manual input card number, expired date, or CVV

Public int manualInput(int timeout, OnPinPadManualInputListener listener);
Parameters:

Parameter	Description
timeout	Enter a timeout in seconds; recommended value 60
listener	Monitor callback interface

Return Value:

SdkResult.Success successful execution listener callback interface

SdkResult.Param_In_InValid illegal Parameter

SdkResult.Fail other errors

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.3.51 inputOnlinePinP2PE

Enter the online PIN if device is P2PE mode

Public int inputOnlinePinP2PE (int [] pinLen, int timeout, int mKeyId, PinAlgorithmModeEnum pinAlgMode, OnPinPadInputListener listener);
Parameters:

Parameter	Description
pinLen	The length of the support, such as{0x00,0x04,0x05,0x06,0x07,0x08,0x09,0x0a, 0x0b, 0x0c}
timeout	Enter a timeout in seconds; recommended value 60
mKeyId	Master Key Index 0-199(if DUKPT, is 0-19)
pinAlgMode	PIN encryption algorithm mode
listener	Monitor callback interface

PinAlgorithmModeEnum

Enumeration Name	Description
ISO9564FMT0	Format 0
ISO9564FMT1	Format 1
ISO9564FMT2	Format 2
ISO9564FMT3	Format 3
ISO9564FMT4	Format 4

Return Value:

SdkResult.Success successful execution listener callback interface

SdkResult.Param_In_InValid illegal Parameter

SdkResult.Fail other errors

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.3.52 P2PEGetCustomizedCipherText

user can speicify the encryption rule. For example, combine the data of track 1 and track 2 according to the rules of manually inputting card numbers, and return the encrypted result

byte[] P2PEGetCustomizedCipherText(P2PECustomizedEncryptionEntity encryptionEntity);

Parameters:

Parameter	Description
encryptionEntity	Encryption rule configuration

P2PECustomizedEncryptionEntity

attribute	Description
byte[] dataEncryptionRules	Data encryption rule; use TLV format to combine the encrypt data. Tag 01 = user data Tag 02 = card number(pan) Tag 03 = track 1 Tag 04 = track 2 Tag 05 = track 3 Tag 06 = expired date Tag 07 = CVV
P2PEEncryptPaddingModeEnum encryptPaddingModeEnum	*PADDING_LEFT* *PADDING_RIGHT*
byte encryptPaddingCharacter = 0x00	Padding character, default is 0x00

Return Value:

Byte[]	Returns the encryption data
null	Failed

Example of dataEncryptionRules:

Card number = 42690100001234

Track 1 = 6225111111118665^HUA/CONG ^300920115941 999900317000000

Track 2 = 6225111111118665=300920115941317

Expired date = 3009

CVV = 123

MSR Both tracks(track1 & track 2)

dataEncryptionRules = 0101%03000101?0101;04000101?

The expected encryption data =

%B6225111111118665^HUA/CONG ^300920115941 999900317000000?;6225111111118665=300920115941317?

Secure firmware will combine and encrypt the expected data and return the cipher text data.

Manual input with CVV:

dataEncryptionRules=

0102%M02000112^MANUALLY/ENTERED^06000106000000070001070000000?0101;02000101=060001060000000700010300001 01?

The expected encryption data =

%M6225111111118665^MANUALLY/ENTERED^3009000000123000000?;6225111111118665=3009000000123000?

Secure firmware will combine and encrypt the expected data and return the cipher text data.

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.4 Scaner#1(default UI)

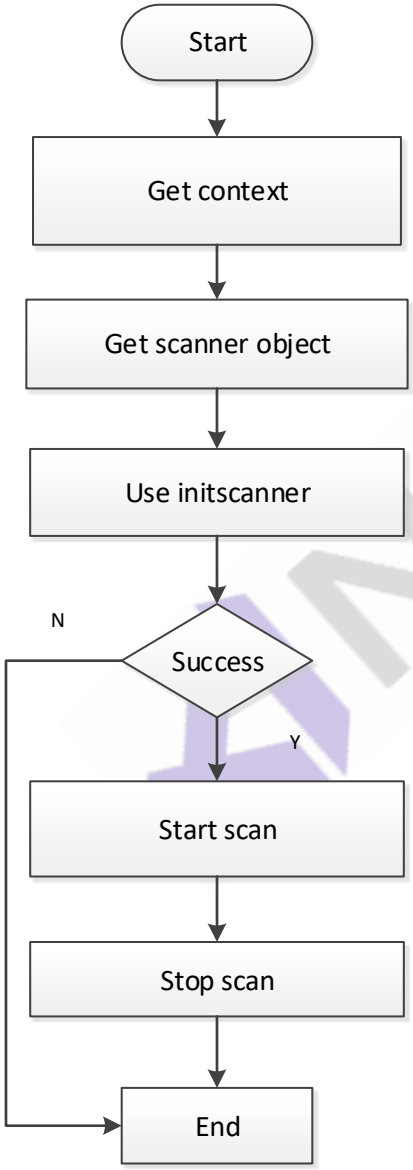
Camera scan code class is responsible for managing POS camera; must be initialized before use.

Get the object of the camera scan class:
Scanner scanner = deviceEngine getScanner();

The following table shows the Return Values supported by the method of the camera sweep class:

Constant Name	Constant Value	Description
Scanner_Base_Error	-2000	
Scanner_Customer_Exit	Scanner_Base_Error - 1	Active user exit
Scanner_Other_Error	Scanner_Base_Error - 2	Scan code fails

This module operates using the basic flow chart:



3.4.1 initScanner

Initialize the scan configuration.

Public int initScanner (ScannerCfgEntity cfgEntity, OnScannerListener listener);

Parameters:

Parameter	Description
cfgEntity	Initialize the configuration
listener	Callback interface

ScannerCfgEntity

Attributes	Description
boolean isUsedFrontCcd	Whether to use the front camera, if only back camera, then open the back camera by default
boolean isBulkMode	Whether continuous scan mode, open the scan after the success of the scan does not exit the interface
int interval	Continuous scan code interval, in milliseconds; default 1000
boolean isAutoFocus	Whether it is auto focus
boolean isNeedPreview	Whether it is need pre-view
List<SymbolEnum> symbolEnumList	List support symbol code, default will support all codes; Need latest firmware support
Bundle mBundle Key boolean showBar boolean showBack boolean showTitle boolean showSwitch boolean showMenu String Title	Use bundle to transfer parameter to customized the Scanner UI Description If show Bar Whether show the back button Whether show the Title text Whether show the button for switching front and back camera Whether show the Menu Customized the title text

int TitleSize	The size of the Title text
string ScanTip	Customized the Scan tip text
int TipSize	The size of the tip text
boolean hideFram	Hide the preview box, need latest firmware support

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.4.2 startScan

start scan

Public int startScan (int timeout, OnScannerListener listener);

Parameters:

Parameter	Description
timeout	Scan code timeout in seconds; recommended value 60
listener	Callback interface

Return Value:

SdkResult.Sucess success

SdkResult.Fail failure

SdkResult.Param_In_Invalid illegal Parameter

3.4.3 stopScan

Stop scanning.

Public void stopScan();

Parameters: None

Return Value: None

3.4.4 decode

decode the image

public String decode(byte[] imageData, int imageWidth, int imageHeight);
Parameters:

Parameter	Description
imageData	Image, date type is YUV420SP
imageWidth	Image width
imageHeight	Image height

Return Value:

Failed : None

Success: decode result

3.5 Scanner#2(customizable UI)

The scanning UI can be customized. For details, see demo

Get the object of camera code scanning class:

Scanner scanner = deviceEngine.getScanner2() ;

3.5.1 initScanner

Initialize scan configuration

public void initScanner(ScannerCfgEntity cfgEntity, Set<SymbolEnum> enableSymbols);

Parameters:

Parameters:	Description
cfgEntity	Initialize the configuration
enableSymbols	Set supported code type

ScannerCfgEntity

Attributes	Description
boolean isUsedFrontCcd	Whether to use the front camera or not. If only the back camera is used, the back camera will be turned on by default
boolean isBulkMode	Continuous code scanning mode. If it is enabled, the code scanning interface will not exit after the code scanning succeeds
int interval	Continuous code scanning interval, unit: Ms default value: 1000

boolean isAutoFocus	Auto focus or not
boolean isNeedPreview	Preview required, default required
Bundle mBundle	User defined interface display settings, you can set the following table key values through bundle. Key Type description showBar boolean Show title bar or not BarColor int Title bar background color showBack boolean Show back button or not showTitle boolean Display title text or not showSwitch boolean is front/back camera switch button displayed showMenu boolean Show menu or not Title String Custom title text TitleSize int Title Text Size TitleColor int Title Text Color MaskColor int Preview mask color AngleColor int Color of four corners of code box FrameColor int Frame color SlideColor int Scanline color ScanTip int Custom prompt text TipColor int Prompt text color TipSize int Prompt text size Pendant String Image mount path

return value : none

3.5.2 getBestPreviewSize

Get the best preview resolution

public Size getBestPreviewSize();

parameter： none

return value： Size

3.5.3 setSurface

Set the preview surface. If not, there will be no preview scanning. Generally, this method is used to call getbestpreviewsize() to return the resolution supported by the camera, and then set it

public void setSurface(Surface surface, int width, int height);
parameter：

parameter	Description
surface	
width	width
height	height

return value : none

3.5.4 start

Start camera scanning

public void start(OnScannerListener listener);
parameter：

parameter	Description
listener	Decode listener

return value : none

3.5.5 stop

Stop scanning code and call when user initiatively exits.

public void stop();
parameter： none

return value : none

3.5.6 switchCamera

Before and after the switch, the camera is called after start. If you want to set up which camera to use from the beginning, please send it in initScanner configuration.

public void switchCamera(boolean usedFrontCcd);
parameter：

parameter	Description
usedFrontCcd	Front camera or not

return value : none

3.5.7 flashTrigger

Turn on the flash and call after start.

public void flashTrigger(boolean on);
parameter：

parameter	Description
on	Turn on flash or not

return value : none

3.5.8 focusTrigger

Open autofocus and call after start.

public void focusTrigger(boolean auto);
parameter：

parameter	Description
auto	Turn on auto connect focus

return value : none

3.5.9 setZoom

Set up an enlarged preview and call it after start.

public void setZoom(float scale);
Parameter term：

Parameter	Description
scale	0f~1.0f, restore default when 0

return value : none

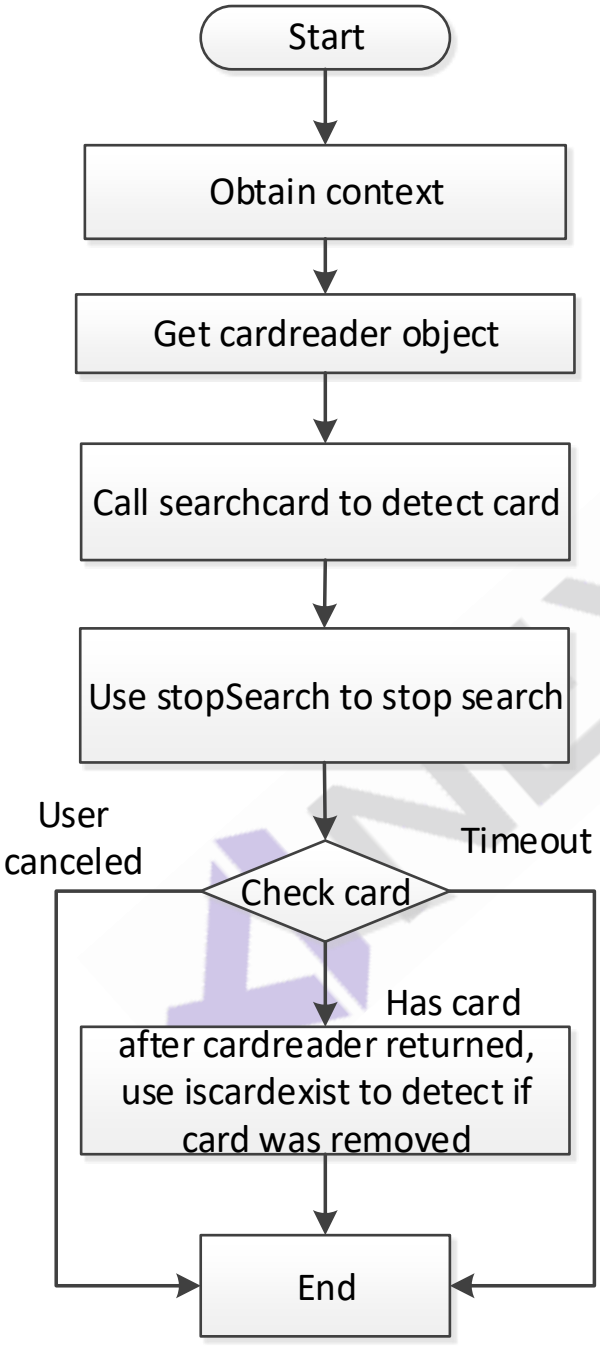
3.6 Card Reader Class

Card reader is responsible for managing the POS card reader (Note: 1, the user can take the initiative to stop the card operation; 2, find card operation automatically stop after the card is found).

Get the object of the reader class:

CardReader reader = deviceEngine getCardReader ();

This module operates using the basic flow chart:



3.6.1 searchCard

Open the corresponding card reader, check the corresponding card slot has a card.

public int searchCard(HashSet<CardSlotTypeEnum> slotTypes,int timeout, OnCardInfoListener listener);
Parameters:

Parameter	Description
slotTypes	Slot enumerated type CardSlotTypeEnum; supports a variety of combinations of slots
timeout	Timeout in seconds; recommended value 60
listener	Callback interface OnCardInfoListener

CardSlotTypeEnum

Enumeration Name	Description
ICC1	Default IC card slot (Contact card solt)
ICC2	Unavailable
ICC3	Unavailable
PSAM1	PSAM slot 1
PSAM2	PSAM slot 2
PSAM3	Unavailable (only available for UN20)
PSAM4	Unavailable (only available for UN20)
RF	Conactless card slot
SWIPE	Magnetic stripe card slot

Return Value:

SdkResult.Success successful execution listener callback interface

SdkResult.Fail other errors

3.6.2 stopSearch

Turn off the corresponding card reader and stop detecting if there is a card in the card slot.

Public void stopSearch();

Parameters: None

Return Value: None

3.6.3 isCardExist

When the card reader operation is finished, call the card to check whether the contact IC card is pulled out or if there is a contactless card taped on the reader.

Public boolean isCardExist (CardSlotTypeEnum slotTypes);

Parameters:

Parameter	Description
slotTypes	Slot enumerated type CardSlotTypeEnum

CardSlotTypeEnum

Enumeration Name	Description
ICC1	Default IC card slot (Contact card solt)
ICC2	Unavailable
ICC3	Unavailable
PSAM1	PSAM slot 1
PSAM2	PSAM slot 2
PSAM3	Unavailable (only available for UN20)
PSAM4	Unavailable (only available for UN20)
RF	Contactless card slot
SWIPE	Magnetic stripe card slot

Return Value:

True exists

False does not exist

3.6.4 isCardExist

When the card reader operation is finished, call the card to check whether the contact IC card is pulled out or if there is a contactless card taped on the reader.

User need to open the card reader first with “open” API.

boolean isCardExist(CardSlotTypeEnum cardSlotType, int checkTimes);

Parameters:

Parameter	Description
slotTypes	Slot enumerated type CardSlotTypeEnum

CardSlotTypeEnum

Enumeration Name	Description
ICC1	Default IC card slot (Contact card solt)
ICC2	Unavailable
ICC3	Unavailable

PSAM1	PSAM slot 1
PSAM2	PSAM slot 2
PSAM3	Unavailable (only available for UN20)
PSAM4	Unavailable (only available for UN20)
RF	Contactless card slot
SWIPE	Magnetic stripe card slot

Return Value:

True exists

False does not exist

3.6.5 open

Open the specified slot, and if you have already called searchCard to find the card, you do not need to call open again

public void open(CardSlotTypeEnum cardSlotType);

Parameters:

Parameter	**Description**
cardSlotType	CardSlotTypeEnum

CardSlotTypeEnum

Enumeration Name	**Description**
ICC1	Default IC card slot(Contact card solt)
ICC2	Unavailable
ICC3	Unavailable
PSAM1	PSAM slot 1
PSAM2	PSAM slot 2
PSAM3	Unavailable(only available for UN20)
PSAM4	Unavailable (only available for UN20)
RF	Conactless card slot
SWIPE	Magnetic stripe card slot

Return Value: None

3.6.6 close

Close the specified slot.

public void close(CardSlotTypeEnum cardSlotType);

Parameters:

Parameter	Description
cardSlotType	CardSlotTypeEnum

CardSlotTypeEnum

Enumeration Name	Description
ICC1	Default IC card slot(Contact card solt)
ICC2	Unavailable
ICC3	Unavailable
PSAM1	PSAM slot 1
PSAM2	PSAM slot 2
PSAM3	Unavailable(only available for UN20)
PSAM4	Unavailable (only available for UN20)
RF	Conactless card slot
SWIPE	Magnetic stripe card slot

Return Value: None

3.6.7 getRfCardType

Get contactless card type

public RfCardTypeEnum getRfCardType(CardSlotTypeEnum cardSlotType);

Parameters:

Parameter	Description
cardSlotType	CardSlotTypeEnum

CardSlotTypeEnum

Enumeration Name	Description
ICC1	Default IC card slot(Contact card solt)
ICC2	Unavailable
ICC3	Unavailable
PSAM1	PSAM slot 1
PSAM2	PSAM slot 2
PSAM3	Unavailable(only available for UN20)
PSAM4	Unavailable (only available for UN20)
RF	Conactless card slot
SWIPE	Magnetic stripe card slot

RfCardTypeEnum

Enumeration Name	Description
TYPE_A_CPU	
TYPE_B_CPU	
S50	
FELICA	
S70	
ULTRALIGHT	
MEMORY_OTHER	
S50_PRO	
S70_PRO	

Return Value:

Success return RfCardTypeEnum

Fail null

3.6.8 setETU

reset ETU

public void setETU(CardSlotTypeEnum cardSlotType, int val);

Parameters:

Parameter	Description
cardSlotType	CardSlotTypeEnum
val	Value 0:372(standard card, default support adaptive 4-fold)

CardSlotTypeEnum

Enumeration Name	Description
ICC1	Default IC card slot(Contact card solt)
ICC2	Unavailable
ICC3	Unavailable
PSAM1	PSAM slot 1
PSAM2	PSAM slot 2
PSAM3	Unavailable (only available for UN20)
PSAM4	Unavailable (only available for UN20)
RF	Conactless card slot
SWIPE	Magnetic stripe card slot

Return Value:

None

3.6.9 setSupportFelica

set if support Felica Card .

public void setSupportFelica(boolean var1);

Return Value:

None

3.6.10 setFelicaSystemCode

set Felica Card system code

void setFelicaSystemCode(byte[] code);

Return Value:

None

3.6.11 setFelicaRequestCode

set Felica Request Code.

void setFelicaRequestCode(byte code);

Return Value:

None

3.6.12 setReadOnlyOneCardType

Read specific card type; Can pass TYPE_A_CPU, TYPE_B_CPU, and FELICA

public void setReadOnlyOneCardType(RfCardTypeEnum rfCardTypeEnum);

Parameters:

RfCardTypeEnum

Enumeration Name	Description
TYPE_A_CPU	

TYPE_B_CPU	
S50	
FELICA	
S70	
ULTRALIGHT	
MEMORY_OTHER	
S50_PRO	
S70_PRO	

Return Value:

None

3.6.13 setSearchReader

config contactless reader internal or external.

public int setSearchReader(ReaderTypeEnum readerTypeEnum);

Parameters:

readerTypeEnum

Enumeration Name	Description
INNER	
OUTER	

Return Value:

SdkResult.Success

Others, failed

3.6.14 setSearchReader

config contactless reader internal or external.

public int setSearchReader(ReaderTypeEnum readerTypeEnum, boolean isBeep);

Parameters:

Parameter	Description
readerTypeEnum	Slot enumerated type CardSlotTypeEnum
isBeep	Set beep or not If set reader external.

readerTypeEnum

Enumeration Name	Description
INNER	
OUTER	

Return Value:

SdkResult.Success

Others, failed

3.6.15 getRfDriverVersion

Get contactless reader driver version

public String getRfDriverVersion();

3.6.16 getIccDriverVersion

Get contact ICC reader driver version

public String getIccDriverVersion();

3.6.17 setP2PEConfiguration

set P2PE configuration, such as KeySchemeMode(MK/SK, or DUKPT), algorithm type, key index, etc.

This function can be used for some clients required P2PE(point to point encryption). The card sensitive data will not appear in the SDK or application level. Application can only get the ciphertext of the card sensitive data, such as card number, expire date, track data.

int setP2PEConfiguration(P2PEConfiguration securityConfiguration);

Parameters:

Parameter	Description

securityConfiguration	P2PE configuration

P2PEConfiguration

Attributes	Description
P2PEKeySchemeModeEnum keySchemeModeEnum	KeySchemeMode, can be MK/SK or DUKPT
P2PEAlgTypeEnum algTypeEnum	DES, or AES
P2PECalcModeEnum calcModeEnum	ECB or CBC
int keyIndex	Key index
byte[] iv	Iv, CBC required
P2PEDukptModeEnum dukptKeyModeEnum	DES_REQUEST_MODE DES_RESPONSE_MODE AES_USAGE_DATA_ENC_MODE AES_USAGE_DATA_BOTH_MODE
P2PEExpiredDateMode expiredDateMode	Expire date is plaintext or cipher text
P2PECvvMode cvvMode	Cvv is plaintext or cipher text, only apply for Manual imput
int preLenPanMask	The first few digits of the card number are not masked
int sufLenPanMask	The last few digits of the card number are not masked For example: preLenPanMask = 6 sufLenPanMask = 4 card number= 4567890000001234 The masked card number=4567890****1234
P2PEPaddingConfiguration MSRTrack1PaddingConfiguration	MSR Track 1 padding configuration
P2PEPaddingConfiguration MSRTrack2PaddingConfiguration	MSR Track 2 padding configuration
P2PEPaddingConfiguration MSRTrack3PaddingConfiguration	MSR Track 3 padding configuration
P2PEPaddingConfiguration MSRPanPaddingConfiguration	MSR PAN padding configuration
P2PEPaddingConfiguration MSRExpiredDatePaddingConfiguration	MSR expired padding configuration
P2PEPaddingConfiguration EMVTrack1PaddingConfiguration	EMV contact and contactless Track 1 padding configuration

P2PEPaddingConfiguration EMVTrack2PaddingConfiguration	EMV contact and contactless Track 2 padding configuration
P2PEPaddingConfiguration EMVPanPaddingConfiguration	EMV contact and contactless card number padding configuration
P2PEPaddingConfiguration ManualPanPaddingConfiguration	Manual input card number padding configuration. It is also applied to manual expired date and cvv

P2PEKeySchemeModeEnum

Enumeration Name	Description
MK_SK	
DUKPT	

P2PEAlgTypeEnum

Enumeration Name	Description
TYPE_2DES	
TYPE_3DES	
TYPE_AES128	
TYPE_AES192	
TYPE_AES256	

P2PECalcModeEnum

Enumeration Name	Description
ECB_MODE	
CBC_MODE	

P2PEDukptModeEnum

Enumeration Name	Description
DES_REQUEST_MODE	
DES_RESPONSE_MODE	
AES_USAGE_DATA_ENC_MODE	
AES_USAGE_DATA_BOTH_MODE	

P2PEExpiredDateMode

Enumeration Name	Description
PLAINTEXT_MODE	
CIPHERTEXT_MODE	

P2PECvvMode

Enumeration Name	Description
PLAINTEXT_MODE	
CIPHERTEXT_MODE	

P2PEPaddingConfiguration

Attributes	Description
Int leftAppendDataLen	The length of left append data
byte[] leftAppendData	left append data
int rightAppendDataLen	The length of right append data
byte[] rightAppendData	right append data
P2PEEncryptPaddingModeEnum encryptPaddingModeEnum	PADDING_LEFT or PADDING_RIGHT Default is PADDING_RIGHT
byte encryptPaddingCharacter	Padding Character, default 0x00; After padding data(left or right padding), if the length is not a multiple of 8, then use Padding Character to padding

Return Value:

SdkResult.Success success

SdkResult.Fail failed

3.7 CPU Cards

The CPU card class is responsible for managing the CPU card.

Get the object of the CPU card class:

CPUCardHandler cpucard = deviceEngine getCPUCardHandler (CardSlotTypeEnum slotType).;

Parameters:

Parameter	Description
slotType	Card slot type

CardSlotTypeEnum

Enumeration Name	Description
ICC1	Default IC card slot(Contact card solt)
ICC2	Unavailable
ICC3	Unavailable
PSAM1	PSAM slot 1
PSAM2	PSAM slot 2
PSAM3	Unavailable (only available for UN20)
PSAM4	Unavailable (only available for UN20)
RF	Conactless card slot

This module operates using the basic flow chart:

3.7.1 readUid

Read Uid of the card

Public String readUid ();

Parameters: None

Return Value:

Success Uid

Note : if the card is Felica , the UID = IDm + PMm

Failure Null

3.7.2 powerOn

Power-on reset, only for ICC1, PSAM1, PSAM2.

Public boolean powerOn (byte [] atr);

Parameters:

Parameter	Description
atr	Power returns atr, the first length byte hexadecimal representation, followed by the standard atr data

Return Value:

True success

False failure

3.7.3 active

Activated, only for contactless card(RF).

Public boolean active ();

Parameters: None

Return Value:

True success

False failure

3.7.4 exchangeAPDUCmd

Interactive APDU command.

Public int exchangeAPDUCmd (APDUEntity cmd);

Parameters:

Parameter	Description
cmd	APDUEntity Command data

APDUEntity

Attributes	Description
byte p1	Instruction to attach a specific Parameter
byte p2	Instruction to attach a specific Parameter
int lc	The number of bytes to transfer data
int le	Expect the maximum number of bytes to return
byte ins	Instruction code
byte cla	Command category
byte swa	Back swa
byte swb	Back swb
int dataOutLen	Returns the length of the data

byte [] dataIn	Sent data
byte [] dataOut	Return data

Return Value:

SdkResult.Success success

SdkResult.Fail failure

3.7.5 exchangeAPDUCmd

Interactive APDU command.

public byte[] exchangeAPDUCmd(byte[] cmd);

Parameters:

Parameter	Description
cmd	Apdu command data

Return Value:

Success return response data

Fail null

3.7.6 powerOff

Power down.

Public void powerOff();

Parameters: None

Return Value: None

3.7.7 remove

remove contactless card

public boolean remove();

Return Value:

SdkResult.Success success

SdkResult.Fail failure

3.8 EMV class (Emvhandler2)

The EMV class is responsible for managing the EMV operation of the POS.

Get the object of the EMV class:

EmvHandler2 EmvHandler = deviceEngine getEmvHandler2 (String appId);

Parameters:

Parameter	Description
appId	Application ID is mainly used to distinguish between aid and capk storage paths

Please note: the maximum number for EMV AID and CAPK are 100.

3.8.1 delAllAid

Remove all AIDs.

Public void delAllAid ();

Parameters: None

Return Value: None

3.8.2 delOneAid

Delete an AID.

Public boolean delOneAid (byte [] aid);

Parameters:

Parameter	Description
aid	Enter aid

Return Value:

True success

False failure

3.8.3 delAllCapk

Remove all CAPK.

Public void delAllCapk ();

Parameters: None

Return Value: None

3.8.4 delOneCapk

Delete a CAPK.

Public boolean delOneCapk (byte [] rid, int capkIdx);

Parameters:

Parameter	Description
rid	Enter rid
capkIdx	capk Index

Return Value:

True success

False failure

3.8.5 setAidParaList

Set the AID.

public int setAidParaList(List<AidEntity> aidParaTlvList);

Parameters:

Parameter	Description
aidParaTlvList	Aid list

AidEntity

attribute	Description
String aid	Application ID
int asi	Application selection indicator 0- needn't match exactly(partial match up to the length); 1- match exactly
String tacDefault	Terminal Action Code – Default
String tacOnline	Terminal Action Code – Online
String tacDenial	Terminal Action Code – Denial
String appVerNum	Application Version Number

String ddol	DDOL
long threshold	Threshold value for biased random selection
int maxTargetPercent	The maximum target percentage to be used for biased random selection
int targetPercent	The target percentage to be used for random selection
int onlinePinCap	Terminal online Pin capability
long floorLimit	Contact floor limit
long transLimit	Electronic cash limit(union pay used in china market)
long contactlessCvmLimit	Contactless cvm limit
long contactlessTransLimit	Contactless transaction limit
long contactlessFloorLimit	Contactless floor limit
String transType	Transaction type, EMV tag 9c, “00”-sale, “20”-refund.. Default value is “FF”,it means adapt to all transaction type
AidEntryModeEnum aidEntryModeEnum	AID_ENTRY_CONTACT_CONTACTLESS: default value, means this aid can used for both contact and contactless AID_ENTRY_CONTACT: This aid is only used for contact AID_ENTRY_CONTACTLESS: This aid is only used for contactless So, the same aid can config 2 aid with aidEntryModeEnum different, one is only for contact, and one is only for contactless

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.6 setAidParaList

Set the AID.

Public int setAidParaList (List <byte []> aidParaTlvList);

Parameters:

Parameter	Description

aidParaTlvList	Enter the number of aid data list, such as: aidParaTlvList.add(ByteUtils.hexString2ByteArray("9F0607A0000000043060DF010 1009F08020002DF1105FC5058A000DF1205F85058F800DF130504000000009F1B0 400000000DF150400000000DF160199DF170199DF14039F3704DF180101DF20060 00999999999"));

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.7 setAidParaList

Set the AID.

public int setAidParaList(List<String> aidParaTlvList);

Parameters:

Parameter	Description
aidParaTlvList	he number of aid data list, such as: aidParaTlvList.add("9F0607A0000000043060DF01 01009F08020002DF1105FC5058A000DF1205F850 58F800DF130504000000009F1B0400000000DF15 0400000000DF160199DF170199DF14039F3704DF 180101DF2006000999999999");

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.8 setCAPKList

Set CAPK.

public int setCAPKList(List<CapkEntity> capkTlvList);
Pamameter：

Parameter	Description

capkTlvList	Capk list

CapkEntity

attribute	Description
String rid	Registered Application Identifier
int capkIdx	Unique CA public key index number
int hashInd	Cryptographic algorithm ID used to generate the CAPK
String modulus	CA Public Key modulus
String exponent	CA Public Key exponent
String checkSum	CA Public Key checkSum
String expireDate	CA Public Key expireDate(YYYYMMDD)

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.9 setCAPKList

Set CAPK.

Public int setCAPKList (List <byte []> capkTlvList);

Parameters:

Parameter	Description
capkTlvList	Enter multiple capk data list, such as: capkTlvList.add (ByteUtils.hexString2ByteArray ("9F0605A0000000659F220109DF05083230303931323331DF060101DF070101DF02 8180B72A8FEF5B27F2B550398FDCC256F714BAD497FF56094B7408328CB626AA6F0 E6A9DF8388EB9887BC930170BCC1213E90FC070D52C8DCD0FF9E10FAD36801FE93F C998A721705091F18BC7C98241CADC15A2B9DA7FB963142C0AB640D5D0135E77EB AE95AF1B4FEFADCF9C012366BDDA0455C1564A68810D7127676D493890BDDF0401 03DF03144410C6D51C2F83ADFD92528FA6E38A32DF048D0A"));

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.10 setCAPKList

Set CAPK.

public int setCAPKList(List<String> capkTlvList);

Parameters:

Parameter	Description
capkTlvList	Enter multiple capk data list, such as: capkTlvList.add("9F0605A0000000659F220109DF050 832303039313233 31DF060101DF070101DF028180B 72A8FEF5B27F2B550398FDCC256F714BAD497FF560 94B7408328CB626AA6F0E6A9DF8388EB9887BC9301 70BCC1213E90FC070D52C8DCD0FF9E10FAD36801FE 93FC998A721705091F18BC7C98241CADC15A2B9DA 7FB963142C0AB640D5D0135E77EBAE95AF1B4FEFA DCF9C012366BDDA0455C1564A68810D7127676D49 3890BDDF040103DF03144410C6D51C2F83ADFD925 28FA6E38A32DF048D0A");

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.11 getAidListNum

get aid list number

public int getAidListNum();

Return Value:

Number of aid list

3.8.12 getAidList

get aid list

public List<AidEntity> getAidList();

AidEntity

Attributes	Description
String aid	Application ID
int asi	Application selection indicator 0- needn't match exactly(partial match up to the length); 1- match exactly
String tacDefault	Terminal Action Code – Default
String tacOnline	Terminal Action Code – Online
String tacDenial	Terminal Action Code – Denial
String appVerNum	Application Version Number
String DDOL	DDOL
long threshold	Threshold value for biased random selection
int maxTargetPercent	The maximum target percentage to be used for biased random selection
int targetPercent	The target percentage to be used for random selection
int onlinePinCap	Terminal online Pin capability
long floorLimit	
long transLimit	
long contactlessCvmLimit	
long contactlessTransLimit	
long contactlessFloorLimit	

Return Value:

Success return aid list

Fail return null

3.8.13 getCapkListNum

get capk list number

public int getCapkListNum();

Return Value:

Number of capk list

3.8.14 getCapkList

get capk list

public List<CapkEntity> getCapkList();

CapkEntity

Attributes	Description
String rid	Registered Application Identifier
int capkIdx	Unique CA public key index number
int hashInd	Cryptographic algorithm ID used to generate the CAPK
String modulus	CA Public Key modulus
String exponent	CA Public Key exponent
String checkSum	CA Public Key checkSum
String expireDate	CA Public Key expireDate(MMYY)

Return Value:

Success return capk list

Fail return null

3.8.15 emvDebugLog

enable EMV log for checking emv issues, default false

public void emvDebugLog(boolean isEnable);

Parameters:

Parameter	Description
isEnable	True , false

Return Value: None

3.8.16 setDynamicReaderLimitListForPaywave

Set DRL for paywave

public int setDynamicReaderLimitListForPaywave(List<DynamicReaderLimitEntity> drlEntityList)

Parameters:

Parameter	Description
drlEntityList	DRL list

DynamicReaderLimitEntity

attribute	Description
byte[] appProgID	
boolean statusCheck	
boolean authOfZeroCheck	
byte authOfZeroCheckOption;	
boolean readerContactlessTransLimitCheck;	
boolean readerCVMReqLimitCheck;	
boolean readerContactlessFloorLimitCheck;	
private boolean drlSupport;	
byte[] readerContactlessTransLimit;	
byte[] readerCVMReqLimit;	
byte[] readerContactlessFloorLimit;	

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.17 setDynamicReaderLimitListForExpressPay

Set DRL for Amex Expresspay

public int setDynamicReaderLimitListForExpressPay (List<DynamicReaderLimitEntity> drlEntityList)

Parameters:

Parameter	Description
drlEntityList	DRL list

DynamicReaderLimitEntity

attribute	Description
byte[] appProgID	Application Prog ID
boolean statusCheck	statusCheck
boolean authOfZeroCheck	
byte authOfZeroCheckOption;	
boolean readerContactlessTransLimitCheck;	
boolean readerCVMReqLimitCheck;	
boolean readerContactlessFloorLimitCheck;	
private boolean drlSupport;	
byte[] readerContactlessTransLimit;	
byte[] readerCVMReqLimit;	
byte[] readerContactlessFloorLimit;	

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.18 getTlv

Get tag.

Public byte [] getTlv (byte [] tag, EmvDataSourceEnum pathId);

Parameters:

Parameter	Description
tag	tag value
pathId	tag source

EmvDataSourceEnum

Enumeration Name	Description
FROM_KERNEL	Data sources kernel
FORM_CARD	Data sources cards

Return Value:

Tlv successful Return Value

Else return null

3.8.19 getTlvByTags

public String getTlvByTags(String[] tags);

Parameters:

Parameter	Description
tags	Tag such as: String[] TAGS = {"9f26", "9f27", "9f10", "9f37", "9f36", "95", "9a", "9c", "9f02", "5f2a", "82", "9f1a", "9f03","9f33", "9f34", "9f35", "9f1e", "9f09", "84", "9f41"}

Return Value:

Tlv successful Return string Value

Else return null

3.8.20 setTlv

Settings tag for EMV processing

public int setTlv (byte [] tag, byte [] value);

Parameters:

Parameter	Description
tag	tag value
value	data

Return Value:

SdkResult.Success success

SdkResult.Fail failure

SdkResult.Param_In_Invalid Parameter error

3.8.21 initTermConfig

Allows the user to set the terminal personalization attribute, initialize the EMV kernel, and use the EMV kernel default attribute if the user does not call it. (Not recommended, please use method setTlv instead of this method)- Deprecated

Public int initTermConfig (byte [] cfgTlv);

Parameters:

Parameter	Description
cfgTlv	Standard tlv data stream

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.22 emvProcess

Start emv process

Public int emvProcess (EmvTransConfigurationEntity transData, OnEmvProcessListener2 listener);

Parameters:

Parameter	Description
transData	EMV transactions Entity Info
listener	EMV flow monitor interfaces

EmvTransDataEntity

Attributes	Description
String traceNo	trace number, length 8
String transAmt	Amount, length 12, for example “000000010000” = 100.00
String cashbackAmt	Cash back amount, length 12
String transDate	Transaction date yyMMdd, length 6
String transTime	Transaction date hhmmss, length 6
byte[] merName	merchant name
String merId	merchant ID, length 15
String termId	terminal ID, length 8
byte emvTransType	EMV Transaction Type, sale-0x00, refund-0x20…
String countryCode	Country code , emv tage 9f1a
String currencyCode	Currency code, emv tag 5f2a
EmvEntryModeEnum entryModeEnum	Entry mode: contact or contactless
EmvProcessFlowEnum processFlowEnum	Standard flow(full flow) Read app data flow
boolean isContactForceOnline	True: Contact transaction force online False: standard process
EmvContactTerminalDecisionForFirstGAC terminalDecisionForFirstGAC	DECISION_KERNEL, default value, decision by EMV kernel DECISION_TERMINAL_AAC, terminal request AAC DECISION_TERMINAL_ARQC, terminal request ARQC if kernel decision is not AAC
MasterCardTransDataEntity entity	Master card parameter
VisaTransDataEntity entity	Visa parameter
AmexTransDataEntity entity	Amex parameters

UnionPayTransDataEntity entity	UnionPay parameter
CpaceTransDataEntity	Cpace parameter

EmvProcessFlowEnum

Enumeration Name	Description
EMV_PROCESS_FLOW_STANDARD	Standard emv flow
EMV_PROCESS_FLOW_READ_APPDATA	Read application data, card number ..etc(it is suitable for contact and contactless)

EmvEntryModeEnum

Enumeration Name	Description
EMV_ENTRY_MODE_CONTACT	Contact
EMV_ENTRY_MODE_CONTACTLESS	Contactless

MasterCardTransDataEntity

Attributes	Description
Boolean isSupportContactQps	True: support conatact QPS False: do not support
String contactNoCvmLimit	Contact QPS limit, 12 bytes. If transaction amount < contactNoCvmLimit, for master credit card, No cvm replace signature.

VisaTransDataEntity

Attributes	Description
Boolean isSupportContactQps	True : support conatact QPS False: do not support
String contactNoCvmLimit	Contact QPS limit, 12 bytes. If transaction amount < contactNoCvmLimit, for master credit card, No cvm replace signature.
isPaywaveCashPassProcessRestrict	Cash skip Process Restrict or not for Visa Paywave True: skip Process Restrict False: do process Restrict Default is false
isPaywaveCashBackPassProcessRestrict	CashBack skip Process Restrict or not for Visa Paywave True: skip Process Restrict False: do process Restrict Default is false

AmexTransDataEntity

Attributes	Description
Boolean isExpressPaySeePhoneTapCardAgain	express pay see phone test cases, the second tap should set the value true

UnionPayTransDataEntity

Attributes	Description
Boolean isForceOnline	Force online
Boolean isSupportCDCVM	Support CDCVM, default value is true
Boolean isQpbocForGlobal	if use China market, please set false, others please set true. Default value is true
Boolean isSupportContactlessQps	Support QPS
String contactlessQpsLimit	QPS limit

DPASTransDataEntity

Attributes	Description
DPASVersionEnum dpasVersion	DPAS_VERSION_1: DPAS 1.0 version; default value DPAS_VERSION_2: DPAS 2.0 version

DPASVersionEnum

Enumeration Name	Description
DPAS_VERSION_1	DPAS 1.0 version; default value
DPAS_VERSION_2	DPAS 2.0 version

EmvContactTerminalDecisionForFirstGAC

Enumeration Name	Description
DECISION_KERNEL	default value, decision by EMV kernel
DECISION_TERMINAL_AAC	terminal request AAC
DECISION_TERMINAL_ARQC	terminal request ARQC if kernel decision is not AAC

CpaceTransDataEntity

Attributes	Description
boolean isKernelTriggerPinEnter	false, App Process Pin Enter Depends on CVMR;

	true, Kernel Trigger Pin enter with callback onCardHolderInputPin public void onCardHolderInputPin(final boolean isOnlinePin, int leftTimes)

Return Value:

SdkResult.Success success execution listener callback

SdkResult.Param_In_Invalid illegal Parameter

3.8.23 onSetSelAppResponse

After executing the OnEMVProcessListener2. OnSelApp method, call it to notify the EMV kernel to continue the process.

Public void onSetSelAppResponse (int selResult);

Parameters:

Parameter	Description
selResult	After selecting the AID index number, the index starts at 1; the method is performed by onSelApp after obtained.

Return Value: None

3.8.24 onSetTransInitBeforeGPOResponse

After executing the OnEMVProcessListener2. onTransInitBeforeGPO method, call it to notify the EMV kernel to continue the process.

Public void onSetTransInitBeforeGPOResponse (boolean isSuccess);

Parameters:

Parameter	Description
isSuccess	Default value: true. Please note, application can not terminate the emv flow with this API.

Return Value: None

3.8.25 onSetConfirmCardNoResponse

After executing the OnEmvProcessListener2. OnConfirmCardNo method, call it to notify the EMV kernel to continue the process.

Public void onSetConfirmCardNoResponse (boolean isConfirm);

Parameters:

Parameter	Description
isConfirm	true: yes, false: no

Return Value: None

3.8.26 onSetPinInputResponse

After executing the OnEMVProcessListener2. OnCardHolderInputPin method, call it to notify the EMV kernel to continue the process.

Public void onSetPinInputResponse (boolean isConfirm, boolean isBypass);

Parameters:

Parameter	Description
isConfirm	Whether the Enter key is pressed
isBypass	If no password is entered, press the Enter key(Pinbypass)

Return Value: None

3.8.27 OnSetPinInputResponseV2

After executing the OnEMVProcessListener2. OnCardHolderInputPin method, call it to notify the EMV kernel to continue the process.

Public void onSetPinInputResponseV2 (int pinStatus, boolean isBypass);

Parameters:

Parameter	Description
pinStatus	PIN enter status 1: PIN enter OK 0: PIN cancel

	-1: PIN enter error, such as Pinpad not work, or not exist
isBypass	If no password is entered, press the Enter key(Pinbypass)

Return Value: None

3.8.28 onSetContactlessTapCardResponse

After executing OnEmvProcessListener2. onContactlessTapCardAgain method, call it to notify the EMV kernel to continue the process.

public void onSetContactlessTapCardResponse (boolean isSuccess);

Parameters:

Parameter	Description
isSuccess	isSuccess，true:yes，false:no

Return Value: None

3.8.29 onSetOnlineProcResponse

After executing the OnEmvProcessListener. OnOnlineProc method, call it to notify the EMV kernel to take the secondary authorization.

public void onSetOnlineProcResponse (int retCode, EmvOnlineResultEntity result);

Parameters:

Parameter	Description
retCode	SdkResult.Success: connect to the host successfully. SdkResult.Fail: unable connect to the host.
result	EmvOnlineResultEntity, EMV online results

EmvOnlineResultEntity

Attributes	Description
String rejCode	Host respond with transaction response codes
String authCode	Host respond with Transaction Authorization Code
Byte [] recvField55	Host respond 55 field data
EmvTerminalDecisionForSecondGAC	Terminal second GAC force TC regardless of host

TerminalDecisionSecondGAC	response code: DECISION_KERNEL: kernel process, default value DECISION_TERMINAL_AAC: terminal request AAC, not implement DECISION_TERMINAL_TC: terminal request TC when response code ！= 00 Note: If no special scenarios are used, this field can be left unconfigured; Processed by the kernel itself

EmvTerminalDecisionForSecondGAC

Enumeration Name	Description
DECISION_KERNEL	kernel process, default value
DECISION_TERMINAL_AAC	terminal request AAC, not implement
DECISION_TERMINAL_TC	terminal request TC when response code ！= 00

Return Value: None

3.8.30 onSetPromptResponse

After executing OnEmvProcessListener. onPrompt method，call it to notify the EMV kernel to continue the process.

public void onSetPromptResponse (boolean isSuccess);

Parameters:

Parameter	Description
isSuccess	isSuccess，true:yes，false:no

Return Value: None

3.8.31 onSetRemoveCardResponse

After executing OnEmvProcessListener. onRemoveCard method，call it to notify the EMV kernel to continue the process.

public void onSetRemoveCardResponse (boolean isSuccess);

Parameters:

Parameter	Description
isSuccess	isSuccess，true:yes，false:no

Return Value: None

3.8.32 EMVProcessCancel

Cancel EMV process.

public void emvProcessCancel ();

Parameters: None

Return Value: None

3.8.33 EMVProcessAbort

Force quite EMV process.

public void emvProcessAbort ();

Parameters: None

Return Value: None

3.8.34 getEmvContactlessMode

get EMV contactless flow mode, EMV mode or MSD mode, should be called in method onOnlineProc or onFinish method

public EmvModeEnum getEmvContactlessMode();

Return Value:

EmvModeEnum

Enumeration Name	Description
EMV	EMV mode
MSD	MSD mode

UNDEF	UNDEF mode
LEGACY	LEGACY mode

3.8.35 contactlessSetAidFirstSelect

set which AID first select for contactless transaction. It should be called before emvProcess.

public int contactlessSetAidFirstSelect (byte aidLen, byte[] aid);
parameter：

Attributes	Description
aidLen	AID length
aid	AID

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.36 setPureKernelCapab

set pure kernel capability. It should be called in method "onTransInitBeforeGPO"

int setPureKernelCapab(byte[] capab);
parameter：

Attributes	Description
capab	Capability, 5 bytes

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.37 setPureImplOptions

set pure implement options. It should be called in method "onTransInitBeforeGPO"

int setPureImplOptions (byte options);
parameter：

Attributes	Description
options	options

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.38 setPureMTOL

set pure MTOL. It should be called in method "onTransInitBeforeGPO"

int setPureMTOL(byte[] MTOL)
parameter：

Attributes	Description
MTOL	MTOL

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.39 setJcbContactlessTIP

set JCB contactless TIP. It should be called in method "onTransInitBeforeGPO"

void setJcbContactlessTIP(byte[] terminalInterchangeProfile);
parameter：

Attributes	Description
terminalInterchangeProfile	TIP

Return Value:

None

3.8.40 setRupayTransType

set Rupay contactless transaction type. It should be called in method "onTransInitBeforeGPO"

void setRupayTransType (RupayTransType transType);
parameter：

Attributes	Description
transType	Transaction Type

RupayTransType

Attributes	Description
RUPAY_TRANSTYPE_GOODS	GOODS, 0x00
RUPAY_TRANSTYPE_CASH	CASH, 0x01
RUPAY_TRANSTYPE_CASHBACK	CASHBACK, 0x19
RUPAY_TRANSTYPE_MONEYADD	MONEYADD, 0x28
RUPAY_TRANSTYPE_BALANCEENQUIRY	BALANCEENQUIRY,0x31
RUPAY_TRANSTYPE_VOID	VOID,0x34
RUPAY_TRANSTYPE_SERVICECREATION	SERVICE CREATION,0x83
RUPAY_TRANSTYPE_OTHER	DEFAULT, 0xff

Return Value:

None

3.8.41 getJcbContactlessTIP

get JCB contactless TIP.

byte[] getJcbContactlessTIP();

Return Value:

JCB contactless TIP

3.8.42 getSignNeed

get cvm result if need signature

public boolean getSignNeed();
Return Value:

ture	need signature
false	not need signature

3.8.43 getEmvCvmResult

get EMV CVM result

EmvCvmResultEnum getEmvCvmResult();
Return Value:

EmvCvmResultEnum

Enumeration Name	Description
EMV_CVMR_NA	CVM result is not specified, or the result is null
EMV_CVMR_NOCVM	No cvm required
EMV_CVMR_SIGNATURE	Signature
EMV_CVMR_ONLINEPIN	Online pin
EMV_CVMR_CONFVERIFIED	ID verify (not used)
EMV_CVMR_CDCVM	CDCVM
EMV_CVMR_OFFLINEPIN_PLAINTEXT	Offline plaintext pin
EMV_CVMR_OFFLINEPIN_ENCIPHER	Offline encipher pin
EMV_CVMR_OFFLINEPIN_PLAINTEXT_SIGNATURE	Offline plaintext pin & signature
EMV_CVMR_OFFLINEPIN_ENCIPHER_SIGNATURE	Offline encipher pin & signature
EMV_CVMR_SKIP_CVM	Skip cvm, used for MIR

3.8.44 getEmvCardDataInfo

get EMV card data, such as pan, track2 data

CardInfoEntity getEmvCardDataInfo();
Return Value:

CardInfoEntity

Attributes	Description
String cardNo	Card number; If devie is P2PE mode, return ciphertext
CardSlotTypeEnum cardExistslot	CardSlotType
RfCardTypeEnum rfCardType	RfCardTyp
String tk1	track 1; If devie is P2PE mode, return ciphertext
String tk2	tracks 2; If devie is P2PE mode, return ciphertext
String tk3	tracks 3; If devie is P2PE mode, return ciphertext
String expiredDate	Card expired date
String serviceCode	Service Code
boolean isTk1Valid	A track LRC is correct
boolean isTk2Valid	Two tracks LRC is correct
boolean isTk3Valid	Three tracks LRC is correct
boolean isICC	If mag card has chip flag
String csn	Card serial number, only returnd in OnEmvProcessListener.onConfirmCardNo
String maskCardNo	Masked card number, for example 456789******1234 Apply to device P2PE mode
boolean isInCardBin	Card is in whitelist Apply to device P2PE mode
boolean isAppInSredWhiteList	Application is in SRED whitelist Apply to device P2PE mode

CardSlotTypeEnum

Enumeration Name	Description
ICC1	Default IC card slot(Contact card solt)
ICC2	Unavailable
ICC3	Unavailable
PSAM1	PSAM slot 1
PSAM2	PSAM slot 2
PSAM3	Unavailable(only available for UN20)
PSAM4	Unavailable (only available for UN20)
RF	Conactless card slot

RfCardTypeEnum

Enumeration Name	Description
TYPE_A_CPU	
TYPE_B_CPU	
S50	
FELICA	
S70	
ULTRALIGHT	
MEMORY_OTHER	
S50_PRO	
S70_PRO	

3.8.45 getEmvContactlessKernelId

get EMV contactless kernel ID

byte[] getEmvContactlessKernelId();
Return Value:

Success return Kernel ID, otherwise return null.

3.8.46 contactlessAppendAidIntoKernel

For special contactless kernel, application can pass the specify AID to expect kernel process.

int contactlessAppendAidIntoKernel(EmvCardBrandEnum emvCardBrandEnum, byte aidLen, byte[] aid);

Parameters:

Parameter	Description
emvCardBrandEnum	Card brand, such as : VISA, MASTER, JCB, UNION PAY, AMEX, DIS, PURE…
aidLen	The length of the aid
aid	aid

EmvCardBrandEnum

Enumeration Name	Description
EMV_CARD_BRAND_VISA	VISA

EMV_CARD_BRAND_MASTER	MASTER
EMV_CARD_BRAND_AMEX	AMEX
EMV_CARD_BRAND_RUPAY	RUPAY
EMV_CARD_BRAND_UNIONPAY	UNION PAY(UPI)
EMV_CARD_BRAND_DPAS	DPAS: Discover, Diners
EMV_CARD_BRAND_JCB	JCB
EMV_CARD_BRAND_PURE	PURE
EMV_CARD_BRAND_MIR	MIR
EMV_CARD_BRAND_MB	MB
EMV_CARD_BRAND_NSICCS	NSICCS
EMV_CARD_BRAND_BANCOMAT	BANCOMAT
EMV_CARD_BRAND_CPACE	CPACE

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.47 contactlessConfigKernelId

For special contactless kernel, application can pass the specify Kernel ID to expect kernel process.

int contactlessConfigKernelId(EmvCardBrandEnum emvCardBrandEnum, byte kernelLen, byte[] kernelId);

Parameters:

Parameter	Description
emvCardBrandEnum	Card brand, such as: VISA, MASTER, JCB, UNION PAY, AMEX, DIS, PURE…
kernelLen	The length of the aid
kernelId	Kernel ID

EmvCardBrandEnum

Enumeration Name	Description

EMV_CARD_BRAND_VISA	VISA
EMV_CARD_BRAND_MASTER	MASTER
EMV_CARD_BRAND_AMEX	AMEX
EMV_CARD_BRAND_RUPAY	RUPAY
EMV_CARD_BRAND_UNIONPAY	UNION PAY(UPI)
EMV_CARD_BRAND_DPAS	DPAS: Discover, Diners
EMV_CARD_BRAND_JCB	JCB
EMV_CARD_BRAND_PURE	PURE
EMV_CARD_BRAND_MIR	MIR
EMV_CARD_BRAND_MB	MB
EMV_CARD_BRAND_NSICCS	NSICCS
EMV_CARD_BRAND_BANCOMAT	BANCOMAT
EMV_CARD_BRAND_CPACE	CPACE

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.48 getPayWaveResult

get the TVR and TSI of paywave flow.

PayWaveResultEntity getPayWaveResult();

Return :

PayWaveResultEntity

Parameter	Description
tvr	TVR
tsi	TSI

3.8.49 initReader

set contactless reader mode, internal reader, or external reader. It is used for Nexgo K110 Pinpad as contactless reader

void initReader(ReaderTypeEnum readerTypeEnum, int comNo);

parameter：

Attributes	Description
readerTypeEnum	INNER: internal reader, default OUTER: external reader
comNo	Com number, default value is 0; If N86 with K110, value should be 101

Return Value:

None

3.8.50 getEmvContactlessCardType

Get EMV contactless card type is physical card or mobile phone

EmvContactlessCardTypeEnum getEmvContactlessCardType();
Return Value:

EmvContactlessCardTypeEnum

Enumeration Name	Description
CARD_TYPE_NA	NA
CARD_TYPE_PHYSICALCARD	Physical card
CARD_TYPE_MOBILE	Mobile phone

3.8.51 configPrivateTag

application can config private tags to EMV kernel to store during EMV flow. And call getTlv method to get the private tags. Normally EMV kernel does not store the private(non-standard) tags.

int configPrivateTag(byte[] privateTag, byte privateTagLen);

parameter：

Attributes	Description
privateTag	Private(non-standard) tag
privateTagLen	Tag length

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.52 getLastApduCommand

get EMV last APDU command

EmvApduCommandEntity getLastApduCommand();

CardInfoEntity

Attributes	Description
byte[] apduCommandSend;	Send APDU command
byte[] apduCommandRecv;	Response APDU command

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.8.53 P2PEGetCustomizedCipherText

user can speicify the encryption rule. For example, combine the data of track 1 56 tag and track 2 57 tag according to the rules and return the encrypted result

byte[] P2PEGetCustomizedCipherText(P2PECustomizedEncryptionEntity encryptionEntity);

Parameters:

Parameter	Description
encryptionEntity	Encryption rule configuration

P2PECustomizedEncryptionEntity

attribute	Description
byte[] dataEncryptionRules	Data encryption rule; use TLV format to combine the encrypt data. Tag 01 = user data Tag 56 = track 1 Tag 57 = track 2
P2PEEncryptPaddingModeEnum encryptPaddingModeEnum	*PADDING_LEFT* *PADDING_RIGHT*
byte encryptPaddingCharacter = 0x00	Padding character, default is 0x00

Return Value:

Byte[]	Returns the encryption data
null	Failed

Example of dataEncryptionRules:

Card number = 6225111111118665

Track 1 = 4264281511112228^VARNAVA/SOLVEIGA^25121015432100000000000000000150

Track 2 = 6225111111118665=300920115941889

EMV Both tracks(track1 & track 2)

dataEncryptionRules = 0101%56000101?0101;57040101?

The expected encryption data =

%4264281511112228^VARNAVA/SOLVEIGA^25121015432100000000000000000150?;6225111111118665=300920115941889?

Secure firmware will combine and encrypt the expected data and return the cipher text data.

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.8.54 getPureKernelCapab

get pure kernel capability

byte[] getPureKernelCapab();

Return:

Pure kernel capability

3.9 EMV class(Emvhandler) Deprecated

Please note: All the Emvhandler method, do not recommend use it anymore.

The EMV class is responsible for managing the EMV operation of the POS.

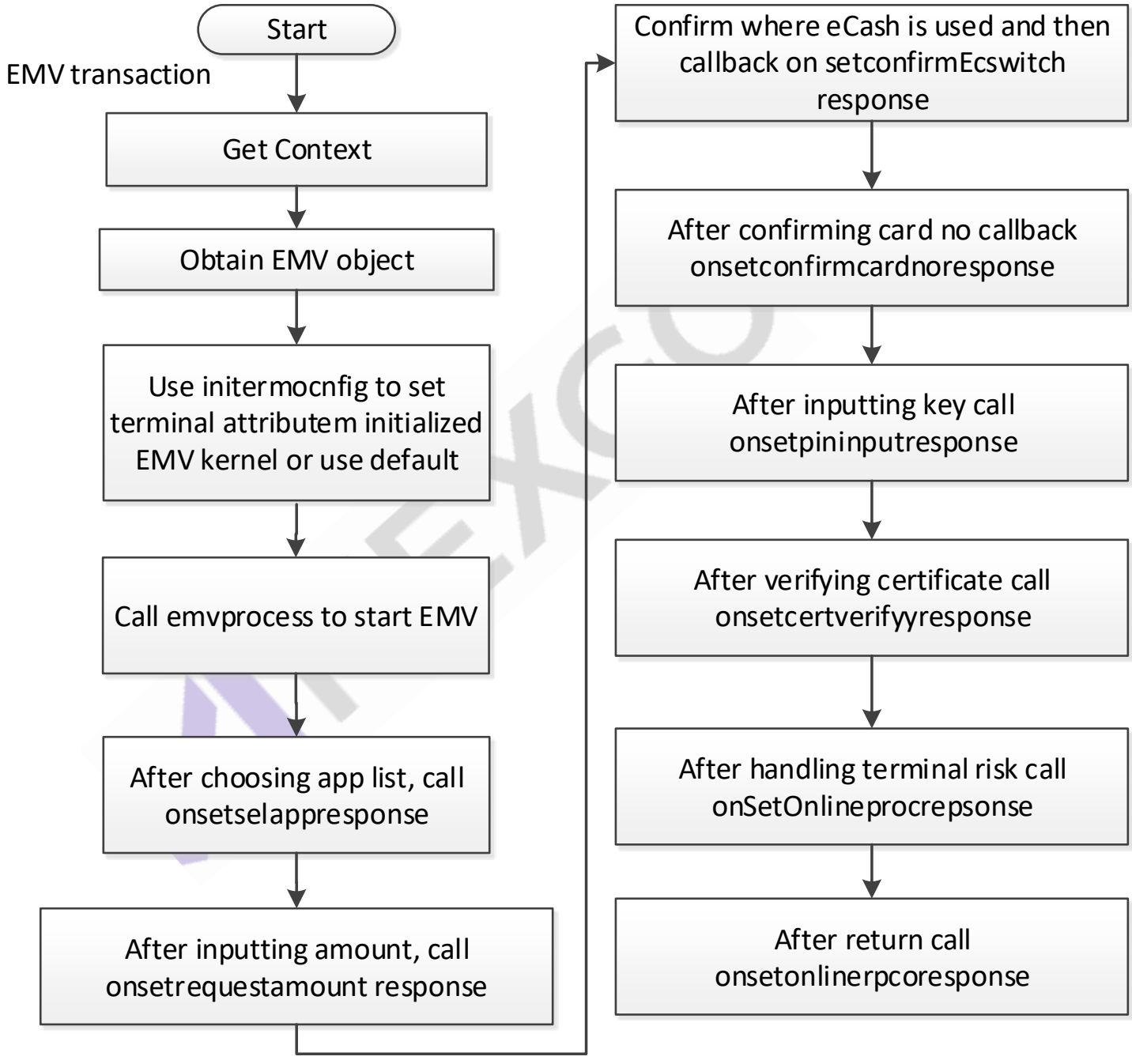
Get the object of the EMV class:

EmvHandler EmvHandler = deviceEngine getEmvHandler (String appId).;

Parameters:

Parameter	Description
appId	Application ID is mainly used to distinguish between aid and capk storage paths

This module operates using the basic flow chart:


Start
EMV transaction
Get Context
Obtain EMV object
Use initermocnfig to set terminal attributem initialized EMV kernel or use default
Call emvprocess to start EMV
After choosing app list, call onsetselappresponse
After inputting amount, call onsetrequestamount response
Confirm where eCash is used and then callback on setconfirmEcswitch response
After confirming card no callback onsetconfirmcardnoresponse
After inputting key call onsetpininputresponse
After verifying certificate call onsetcertverifyyresponse
After handling terminal risk call onSetOnlineprocrepsonse
After return call onsetonlinerpcoresponse

3.9.1 delAllAid

Remove all AIDs.

Public void delAllAid ();

Parameters: None

Return Value: None

3.9.2 delOneAid

Delete an AID.

Public boolean delOneAid (byte [] aid);

Parameters:

Parameter	Description
aid	Enter aid

Return Value:

True success

False failure

3.9.3 delAllCapk

Remove all CAPK.

Public void delAllCapk ();

Parameters: None

Return Value: None

3.9.4 delOneCapk

Delete a CAPK.

Public boolean delOneCapk (byte [] rid, int capkIdx);

Parameters:

Parameter	Description
rid	Enter rid
capkIdx	capk Index

Return Value:

True success

False failure

3.9.5 setAidParaList

Set the AID.

public int setAidParaList(List<AidEntity> aidParaTlvList);

Parameters:

Parameter	Description
aidParaTlvList	Aid list

AidEntity

attribute	Description
String aid	Application ID
int asi	Application selection indicator 0- needn't match exactly(partial match up to the length); 1- match exactly
String tacDefault	Terminal Action Code – Default
String tacOnline	Terminal Action Code – Online
String tacDenial	Terminal Action Code – Denial
String appVerNum	Application Version Number
String DDOL	DDOL
long threshold	Threshold value for biased random selection
int maxTargetPercent	The maximum target percentage to be used for biased random selection
int targetPercent	The target percentage to be used for random selection
int onlinePinCap	Terminal online Pin capability
long floorLimit	
long transLimit	
long contactlessCvmLimit	
long contactlessTransLimit	
long contactlessFloorLimit	

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.6 setAidParaList

Set the AID.

Public int setAidParaList (List <byte []> aidParaTlvList);

Parameters:

Parameter	Description
aidParaTlvList	Enter the number of aid data list, such as: aidParaTlvList.add(ByteUtils.hexString2ByteArray("9F0607A0000000043060DF010 1009F08020002DF1105FC5058A000DF1205F85058F800DF130504000000009F1B0 400000000DF150400000000DF160199DF170199DF14039F3704DF180101DF20060 00999999999"));

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.7 setAidParaList

Set the AID.

public int setAidParaList(List<String> aidParaTlvList);

Parameters:

Parameter	Description
aidParaTlvList	he number of aid data list, such as: aidParaTlvList.add("9F0607A0000000043060DF01 01009F08020002DF1105FC5058A000DF1205F850 58F800DF130504000000009F1B0400000000DF15 0400000000DF160199DF170199DF14039F3704DF 180101DF2006000999999999");

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.8 setCAPKList

Set CAPK.

public int setCAPKList(List<CapkEntity> capkTlvList);
Parameter：

Parameter	Description
capkTlvList	Capk list

CapkEntity

attribute	Description
String rid	Registered Application Identifier
int capkIdx	Unique CA public key index number
int hashInd	Cryptographic algorithm ID used to generate the CAPK
String modulus	CA Public Key modulus
String exponent	CA Public Key exponent
String checkSum	CA Public Key checkSum
String expireDate	CA Public Key expireDate(YYYYMMDD)

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.9 setCAPKList

Set CAPK.

Public int setCAPKList (List <byte []> capkTlvList);

Parameters:

Parameter	Description
capkTlvList	Enter multiple capk data list, such as: capkTlvList.add (ByteUtils.hexString2ByteArray ("9F0605A0000000659F220109DF05083230303931323331DF060101DF070101DF02 8180B72A8FEF5B27F2B550398FDCC256F714BAD497FF56094B7408328CB626AA6F0 E6A9DF8388EB9887BC930170BCC1213E90FC070D52C8DCD0FF9E10FAD36801FE93F C998A721705091F18BC7C98241CADC15A2B9DA7FB963142C0AB640D5D0135E77EB AE95AF1B4FEFADCF9C012366BDDA0455C1564A68810D7127676D493890BDDF0401 03DF03144410C6D51C2F83ADFD92528FA6E38A32DF048D0A"));

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.10 setCAPKList

Set CAPK.

public int setCAPKList(List<String> capkTlvList);

Parameters:

Parameter	Description
capkTlvList	Enter multiple capk data list, such as: capkTlvList.add("9F0605A0000000659F220109DF05083230303931323331DF060101DF070101DF028180B72A8FEF5B27F2B550398FDCC256F714BAD497FF56094B7408328CB626AA6F0E6A9DF8388EB9887BC930170BCC1213E90FC070D52C8DCD0FF9E10FAD36801FE93FC998A721705091F18BC7C98241CADC15A2B9DA7FB963142C0AB640D5D0135E77EBAE95AF1B4FEFADCF9C012366BDDA0455C1564A68810D7127676D493890BDDF040103DF03144410C6D51C2F83ADFD92528FA6E38A32DF048D0A");

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.11 setDynamicReaderLimitList

Set DRL for paywave

public int setDynamicReaderLimitList(List<DynamicReaderLimitEntity> drlEntityList)

Parameters:

Parameter	Description
drlEntityList	DRL list

DynamicReaderLimitEntity

attribute	Description
byte[] appProgID	
boolean statusCheck	
boolean authOfZeroCheck	
byte authOfZeroCheckOption;	
boolean readerContactlessTransLimitCheck;	
boolean readerCVMReqLimitCheck;	
boolean readerContactlessFloorLimitCheck;	
private boolean drlSupport;	
byte[] readerContactlessTransLimit;	
byte[] readerCVMReqLimit;	
byte[] readerContactlessFloorLimit;	

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.12 setDynamicReaderLimitListForExpressPay

Set DRL for Amex Expresspay

public int setDynamicReaderLimitListForExpressPay (List<DynamicReaderLimitEntity> drlEntityList)

Parameters:

Parameter	Description
drlEntityList	DRL list

DynamicReaderLimitEntity

attribute	Description
byte[] appProgID	Application Prog ID

boolean statusCheck	statusCheck
boolean authOfZeroCheck	
byte authOfZeroCheckOption;	
boolean readerContactlessTransLimitCheck;	
boolean readerCVMReqLimitCheck;	
boolean readerContactlessFloorLimitCheck;	
private boolean drlSupport;	
byte[] readerContactlessTransLimit;	
byte[] readerCVMReqLimit;	
byte[] readerContactlessFloorLimit;	

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.13 initTermConfig

Allows the user to set the terminal personalization attribute, initialize the EMV kernel, and use the EMV kernel default attribute if the user does not call it. (Not recommended, please use method setTlv instead of this method)

Public int initTermConfig (byte [] cfgTlv);

Parameters:

Parameter	Description
cfgTlv	Standard tlv data stream

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.14 emvProcess

Start emv process

Public int emvProcess (emvTransDataEntity transData, OnEmvProcessListener listener);

Parameters:

Parameter	Description
transData	EMV transactions Entity Info
listener	EMV flow monitor interfaces

EmvTransDataEntity

Attributes	Description
EmvAlgorithmTypeEnum algType	Type RSA algorithm or SM2, the default RSA
EmvTransFlowEnum procType	EMV Process Type
String traceNo	Serial number, length 8
String transAmt	Amount, length 12
String cashbackAmt	Cash back amount, length 12
String transDate	Transaction date yyMMDD, length 6
String transTime	Transaction date HHMMSS, length 6
byte[] merName	Business name
String merId	Business number, length 15
String termId	Terminal number, length 8
byte B9C	Transaction Type, sale-0x00, refund-0x20…
boolean isSupportEC	Whether support e-cash(only sued for union pay in china market)
EMVChannelTypeEnum channelType	Card approach, contact or contactless
boolean isQpbocForceLine	union pay whether to force go online
boolean isDefaultEC	When set to support EC is true, the default is e-cash; when false, then callback the method to let user to select whether to use electronic cash(only sued for union pay in china market)
isSupportCDCVM	union pay support CDCVM
isQpbocForGlobal	Union pay contactless check CVM limit for excute CVM method.(not force prompt online pin).

isPaywaveCashPassProcessRestrict	Cash skip Process Restrict or not for Visa Paywave True: skip Process Restrict False: do process Restrict Default is false
isPaywaveCashBackPassProcessRestrict	CashBack skip Process Restrict or not for Visa Paywave True: skip Process Restrict False: do process Restrict Default is false

EMVAlgorithmTypeEnum

Enumeration Name	Description
RSA	RSA
SM2	Country code

EMVTransFlowEnum

Enumeration Name	Description
FULL	Standard full process
SIMPLE	Simple processonly confirms the callback number, then directlyOnFinish ends EMV process
QPASS	qpboc flow only confirms the callback number, then directly OnFinish ends EMV process

EMVChannelTypeEnum

Enumeration Name	Description
FROM_ICC	Contact
FROM_PICC	Contactless

Return Value:

SdkResult.Success success execution listener callback

SdkResult.Param_In_Invalid illegal Parameter

3.9.15 onSetSelAppResponse

After executing the OnEMVProcessListener. OnSelApp method, call the EMV kernel to continue the process.

Public void onSetSelAppResponse (int selResult);

Parameters:

Parameter	Description

selResult	After selecting the AID index number, the index starts at 1; the method is performed by onSelApp after obtained.

Return Value: None

3.9.16 onSetAfterFinalSelectedAppResponse

After executing the OnEMVProcessListener. onAfterFinalSelectedApp method, call the EMV kernel to continue the process.

Public void onSetAfterFinalSelectedAppResponse (boolean isSuccess);

Parameters:

Parameter	Description
isSuccess	Default value: true. The result of final select application.

Return Value: None

3.9.17 onSetRequestAmountResponse

After executing the OnEmvProcessListener. OnRequestAmount method, call the EMV kernel to continue the process.

Public void onSetRequestAmountResponse (String amount);

Parameters:

Parameter	Description
amount	Amount length 12, prepend with 0s to make it 12 digits long.

Return Value: None

3.9.18 onSetConfirmEcSwitchResponse

After executing the OnEMVProcessListener. OnConfirmEcSwitch method, call the EMV kernel to continue the process.

Public void onSetConfirmEcSwitchResponse (boolean isConfirm);

Parameters:

Parameter	Description
isConfirm	Whether to use electronic cash, true: yes, false: no

Return Value: None

3.9.19 onSetConfirmCardNoResponse

After executing the OnEmvProcessListener. OnConfirmCardNo method, call the EMV kernel to continue the process.

Public void onSetConfirmCardNoResponse (boolean isConfirm);

Parameters:

Parameter	Description
isConfirm	Are you sure, true: yes, false: no

Return Value: None

3.9.20 onSetPinInputResponse

After executing the OnEMVProcessListener. OnCardHolderInputPin method, call the EMV kernel to continue the process.

Public void onSetPinInputResponse (boolean isConfirm, boolean isBypass);

Parameters:

Parameter	Description
isConfirm	Whether the Enter key is pressed
isBypass	If no password is entered, press the Enter key

Return Value: None

3.9.21 onsetCertVerifyResponse

After executing the OnEMVProcessListener. OnCertVerify method, call the EMV kernel to continue the process.

Public void onSetCertVerifyResponse (boolean isVerify);

Parameters:

Parameter	Description
isVerify	Are you sure, true: yes, false: no

Return Value: None

3.9.22 onSetReadCardAgainResponse

After executing OnEmvProcessListener. onReadCardAgain method，call the EMV kernel to continue the process.

public void onSetReadCardAgainResponse(boolean isSuccess);

Parameters:

Parameter	Description
isSuccess	isSuccess，true:yes，false:no

Return Value: None

3.9.23 onSetOnlineProcResponse

After executing the OnEmvProcessListener. OnOnlineProc method, call the EMV kernel to take the secondary authorization.

public void onSetOnlineProcResponse (int retCode, EmvOnlineResultEntity result);

Parameters:

Parameter	Description
retCode	SdkResult.Success: connect to the host successfully. SdkResult.Fail: unable connect to the host.
result	EmvOnlineResultEntity, EMV online results

EmvOnlineResultEntity

Attributes	Description
String rejCode	Host respond with transaction response codes
String authCode	Host respond with Transaction Authorization Code
Byte [] recvField55	Host respond 55 field data

Return Value: None

3.9.24 onSetPromptResponse

After executing OnEmvProcessListener. onPrompt method，call the EMV kernel to continue the process.

public void onSetPromptResponse (boolean isSuccess);

Parameters:

Parameter	Description
isSuccess	isSuccess，true:yes，false:no

Return Value: None

3.9.25 onSetRemoveCardResponse

After executing OnEmvProcessListener. onRemoveCard method，call the EMV kernel to continue the process.

public void onSetRemoveCardResponse (boolean isSuccess);

Parameters:

Parameter	Description
isSuccess	isSuccess，true:yes，false:no

Return Value: None

3.9.26 getTlv

Get tag.

Public byte [] getTlv (byte [] tag, EmvDataSourceEnum pathId);

Parameters:

Parameter	Description
tag	tag value
pathId	tag source

EmvDataSourceEnum

Enumeration Name	Description
FROM_KERNEL	Data sources kernel
FORM_CARD	Data sources cards

Return Value:

Tlv successful Return Value

Else return null

3.9.27 getTlvByTags

public String getTlvByTags(String[] tags);

Parameters:

Parameter	Description
tags	Tag such as: String[] TAGS = {"9f26", "9f27", "9f10", "9f37", "9f36", "95", "9a", "9c", "9f02", "5f2a", "82", "9f1a", "9f03","9f33", "9f34", "9f35", "9f1e", "9f09", "84", "9f41"}

Return Value:

Tlv successful Return string Value

Else return null

3.9.28 setTlv

Settings tag.

public int setTlv (byte [] tag, byte [] value);

Parameters:

Parameter	Description
tag	tag value
value	data

Return Value:

SdkResult.Success success

SdkResult.Fail failure

SdkResult.Param_In_Invalid Parameter error

3.9.29 getEMVCardLog

Read the log, this method is finished after the callback to onFinish method.

public int getEmvCardLog (EmvChannelTypeEnum channelType, OnEmvProcessListener listener);

Parameters:

Parameter	Description
channelType	Channel Type
listener	Callback

EmvChannelTypeEnum

Enumeration Name	Description
FROM_ICC	Contact
FROM_PICC	Contactless

Return Value:

SdkResult.Success successful execution of listener callback interface

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.30 Clear the Log

Clear the log.

public int clearLog();

Parameters: None

Return Value:

SdkResult.Success success

SdkResult.Fail failure

3.9.31 EMVGetEcBalance

Read electronic cash balance.

public int emvGetEcBalance (EmvChannelTypeEnum channelType, OnEmvProcessListener listener);

Parameters:

Parameter	Description
channelType	Channel Type
listener	Callback

EmvChannelTypeEnum

Enumeration Name	Description
FROM _ICC	Contact
FROM_PICC	Contactless

Return Value:

SdkResult.Success successful execution of listener callback interface

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.32 EMVProcessCancel

Cancel EMV process.

public void emvProcessCancel ();

Parameters: None

Return Value: None

3.9.33 emvDebugLog

enable EMV log, default false

public void emvDebugLog(boolean isEnable);

Parameters:

Parameter	Description
isEnable	True， false

Return Value: None

3.9.34 getEmvContactlessMode

get EMV contactless flow mode, EMV mode or MSD mode, should be called in method onOnlineProc or onFinish method

public EmvModeEnum getEmvContactlessMode();

Return Value:

EmvModeEnum

Enumeration Name	Description
EMV	EMV mode
MSD	MSD mode

3.9.35 getAidListNum

get aid list number

public int getAidListNum();

Return Value:

Number of aid list

3.9.36 getAidList

get aid list

public List<AidEntity> getAidList();

AidEntity

Attributes	Description
String aid	Application ID
int asi	Application selection indicator 0- needn't match exactly(partial match up to the length); 1- match exactly
String tacDefault	Terminal Action Code – Default
String tacOnline	Terminal Action Code – Online
String tacDenial	Terminal Action Code – Denial
String appVerNum	Application Version Number
String DDOL	DDOL
long threshold	Threshold value for biased random selection
int maxTargetPercent	The maximum target percentage to be used for biased random selection
int targetPercent	The target percentage to be used for random selection
int onlinePinCap	Terminal online Pin capability
long floorLimit	
long transLimit	
long contactlessCvmLimit	
long contactlessTransLimit	
long contactlessFloorLimit	

Return Value:

Success return aid list

Fail return null

3.9.37 getCapkListNum

get capk list number

public int getCapkListNum();

Return Value:

Number of capk list

3.9.38 getCapkList

get capk list

public List<CapkEntity> getCapkList();

CapkEntity

Attributes	Description
String rid	Registered Application Identifier
int capkIdx	Unique CA public key index number
int hashInd	Cryptographic algorithm ID used to generate the CAPK
String modulus	CA Public Key modulus
String exponent	CA Public Key exponent
String checkSum	CA Public Key checkSum
String expireDate	CA Public Key expireDate(MMYY)

Return Value:

Success return capk list

Fail return null

3.9.39 newDelAllAid

Pure , MIR kernel API, delete all the AID

public void newDelAllAid();

Parameters: None

Return Value: None

3.9.40 newDelOneAid

Pure , MIR kernel API, delete one AID

public boolean newDelOneAid(byte[] aid);
parameter：

Attributes	Description
byte[] aid	aid

Return Value:

ture delete success

false delete failed

3.9.41 newDelAllCapk

Pure , MIR kernel API, delete all CAPK

public void newDelAllCapk();

Parameters: None

Return Value: None

3.9.42 newDelOneCapk

Pure , MIR kernel API, delete one CAPK

public boolean newDelOneCapk(byte[] rid,int capkIdx);
parameter:

Attributes	Description
rid	rid
capkIdx	Capk index

Return Value:

ture delete success

false delete failed

3.9.43 newSetAidParaList

Pure , MIR kernel API, set AID list

public int newSetAidParaList(List<byte[]> aidParaTlvList);
parameter:

Attributes	Description
aidParaTlvList	Aid list: aidParaTlvList.add(ByteUtils.hexString2ByteArray("9F0607A0000000043060DF0101009F08020002DF1105FC5058A000DF1205F85058F800DF13050400000000 9F1B0400000000DF150400000000DF160199DF170199DF14039F3704DF180101DF2006000999999999"));

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.44 newSetAidParaList

Pure , MIR kernel API, set AID list

public int newSetAidParaList(List<String> aidParaTlvList);

parameter:

Attributes	Description
aidParaTlvList	Aid list: aidParaTlvList.add("9F0607A0000000043060DF0101009F08020002DF1105FC5058A000DF1205F85058F800DF130504000000009F1B0400000000DF150400000000DF160199DF170199DF14039F3704DF180101DF2006000999999999");

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.45 newSetAidParaList

Pure , MIR kernel API, set AID list

public int newSetAidParaList(List<AidEntity> aidParaTlvList);

parameter:

Attributes	Description
aidParaTlvList	Aid list

AidEntity

Attributes	Description
String aid	Application ID
int asi	Application selection indicator 0- needn't match exactly(partial match up to the length); 1- match exactly
String tacDefault	Terminal Action Code – Default
String tacOnline	Terminal Action Code – Online
String tacDenial	Terminal Action Code – Denial
String appVerNum	Application Version Number
String DDOL	DDOL
long threshold	Threshold value for biased random selection
int maxTargetPercent	The maximum target percentage to be used for biased random selection
int targetPercent	The target percentage to be used for random selection
int onlinePinCap	Terminal online Pin capability
long floorLimit	
long transLimit	
long contactlessCvmLimit	
long contactlessTransLimit	
long contactlessFloorLimit	

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.46 newSetCAPKList

Pure , MIR kernel API, set CAPK list

public int newSetCAPKList(List<byte[]> capkTlvList);

parameter：

Attributes	Description
capkTlvList	Capk list： capkTlvList.add(ByteUtils.hexString2ByteArray("9F0605A0000000659F220109DF05083230303931323331DF060101DF070101DF028180B72A8FEF5B27F2B550398FDCC256F714BAD497FF56094B7408328CB626AA6F0E6A9DF8388EB9887BC930170BCC1213E90FC070D52C8DCD0FF9E10FAD36801FE93FC998A721705091F18BC7C98241CADC15A2B9DA7FB963142C0AB640D5D0135E77EBAE95AF1B4FEFADCF9C012366BDDA0455C1564A68810D7127676D493890BDDF040103DF03144410C6D51C2F83ADFD92528FA6E38A32DF048D0A"));

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.47 newSetCAPKList

Pure , MIR kernel API, set CAPK list

public int newSetCAPKList(List<String> capkTlvList);

parameter：

Attributes	Description
capkTlvList	Capk list： capkTlvList.add("9F0605A0000000659F220109DF05083230303931323331DF060101DF070101DF028180B72A8FEF5B27F2B550398FDCC256F714BAD497FF56094B7408328CB626AA6F0E6A9DF8388EB9887BC930170BCC1213E90FC070D52C8DCD0FF9E10FAD36801FE93FC998A721705091F18BC7C98241CADC15A2B9DA7FB963142C0AB640D5D0135E77EBAE95AF1B4FEFADCF9C012366BDDA0455C1564A68810D7127676D493890BDDF040103DF03144410C6D51C2F83ADFD92528FA6E38A32DF048D0A");

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.48 newSetCAPKList

Pure , MIR kernel API, set CAPK list

public int newSetCAPKList(List<CapkEntity> capkTlvList);

parameter：

Attributes	Description
capkTlvList	Capk list

CapkEntity

Attributes	**Description**
String rid	Registered Application Identifier
int capkIdx	Unique CA public key index number
int hashInd	Cryptographic algorithm ID used to generate the CAPK
String modulus	CA Public Key modulus
String exponent	CA Public Key exponent
String checkSum	CA Public Key checkSum
String expireDate	CA Public Key expireDate(MMYY)

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.49 newGetAidListNum

Pure , MIR kernel API, get AID list number

public int getAidListNum();

Return Value:

Number of aid list

3.9.50 newGetAidList

Pure , MIR kernel API, get AID list number

public List<AidEntity> newGetAidList();

AidEntity

Attributes	Description
String aid	Application ID
int asi	Application selection indicator 0- needn't match exactly(partial match up to the length); 1- match exactly
String tacDefault	Terminal Action Code – Default
String tacOnline	Terminal Action Code – Online
String tacDenial	Terminal Action Code – Denial
String appVerNum	Application Version Number
String DDOL	DDOL
long threshold	Threshold value for biased random selection
int maxTargetPercent	The maximum target percentage to be used for biased random selection
int targetPercent	The target percentage to be used for random selection
int onlinePinCap	Terminal online Pin capability
long floorLimit	
long transLimit	
long contactlessCvmLimit	
long contactlessTransLimit	
long contactlessFloorLimit	

Return Value:

Success return aid list

Fail return null

3.9.51 newGetCapkListNum

Pure , MIR kernel API, get CAPK list number

public int newGetCapkListNum();

Return Value:

Number of capk list

3.9.52 newGetCapkList

Pure , MIR kernel API, get CAPK list

public List<CapkEntity> newGetCapkList();

CapkEntity

Attributes	Description
String rid	Registered Application Identifier
int capkIdx	Unique CA public key index number
int hashInd	Cryptographic algorithm ID used to generate the CAPK
String modulus	CA Public Key modulus
String exponent	CA Public Key exponent
String checkSum	CA Public Key checkSum
String expireDate	CA Public Key expireDate(MMYY)

Return Value:

Success return capk list

Fail return null

3.9.53 selectAidFirst

set which AID first select for contactless transaction

public int selectAidFirst(boolean enable, byte aidLen, byte[] aid);

parameter：

Attributes	Description
enable	Enable: true--first; false--default
aidLen	AID length
aid	AID

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.9.54 getSignNeed

get signature state

public boolean getSignNeed();

Return Value:

ture need signature

false not need signature

3.9.55 setPureKernelCapab

set pure kernel capability

int setPureKernelCapab(byte[] capab);

parameter：

Attributes	Description
capab	Capability, 5 bytes

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail failure

3.10 getDeviceInfo

get device information

public DeviceInfo getDeviceInfo();
DeviceInfo

Attributes	Description
String sn	Terminal serail number
String ksn	Custom Terminal serail number

String model	Terminal model, such as N5
String osVer	Os version, such as 5.1.1
String sdkVer	Sdkversion, 3.0.7
String firmWareVer	firmware version, base version such as v1.2.8
String kernelVer	linuxversion
String vendor	vendor, such as Nexgo
String firmWareFullVersion	Full firmware version, such as “v1.2.8_N50000001”
EmvKernelVersionInfo emvKernelVersionInfo	Emv L1, L2 kernel version
String spCoreVersion;	Secure chip Core version
String spBootVersion;	Secure chip boot version

EmvKernelVersionInfo

Attributes	Description
String emvContactL1KernelVersion;	EMV Contact L1
String emvContactlessL1KernelVersion;	EMV Contactless L1
String emvContactKernelVersion;	EMV Contact L2
String emvPayPassKernelVersion;	Paypass version
String emvPayWaveKernelVersion;	Paywave version
String emvExpressPayKernelVersion;	Amex Expresspay version
String emvDiscoverKernelVersion;	EMV DIS version
String emvJcbKernelVersion;	EMV JCB version
String emvUnionPayKernelVersion;	EMV UPI version

Return Value:

successful return DeviceInfo

else return null

3.11 Serial class

Serial class is responsible for managing POS serial port.

Get the serial class objects:

SerialPortDriver port = deviceEngine.getSerialPortDriver(int portNo);

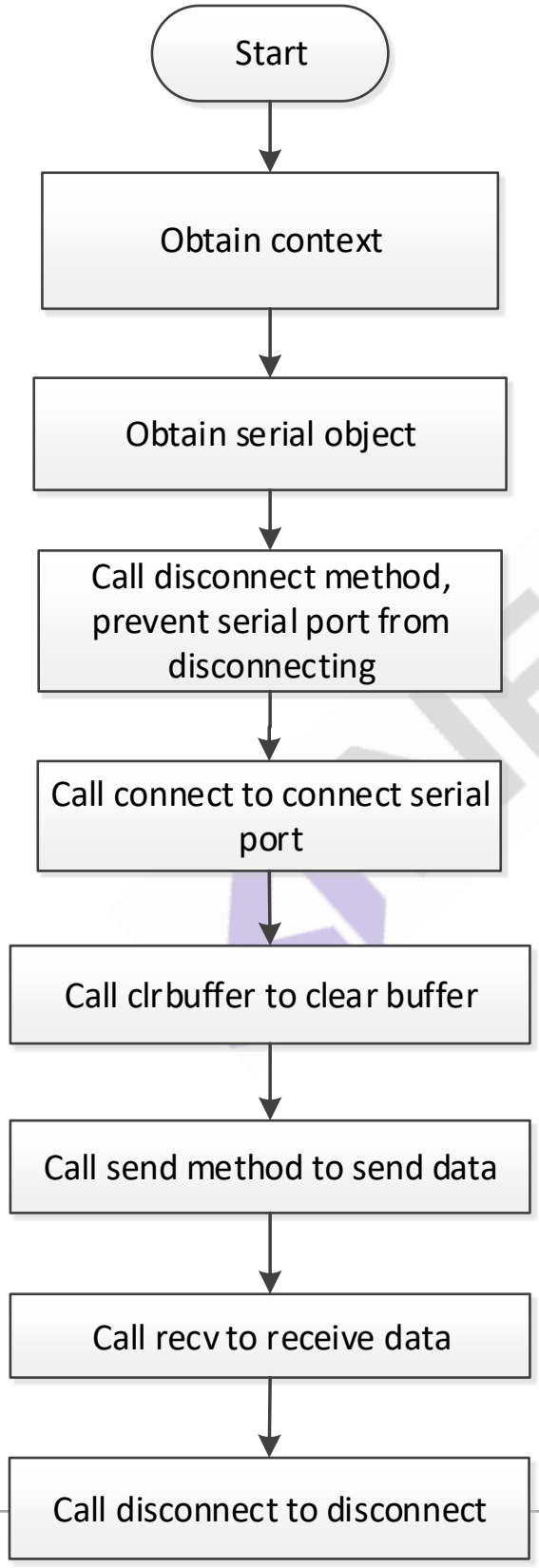
Parameter	Description
portNo	Serial No. N5, N3, N86, N6, P200 ,CT20/CT20P,N80, N96= 0; UN20 = 1, or 2, it depends on which serial port insert. RS232-A = 1; RS232-B = 2 N6(android 7) countertop, the portNo = 0. N6(android 10) countertop, the portNo = 0. CT20P-DB9 = 0 Check below table for better reference

Model	Type-C	RS232	USB-Host	USB-Slave
N5/N3	0	-	-	-
N5 Countertop	-	0	-	-
N6	0	-	-	-
N6 countertop	-	0	-	-
N86	0	-	-	-
UN20	0	RS232-A=1 RS232-B=2	-	-
N96	0	-	-	-
N82	0	-	-	-
N80	0	-	-	-
CT20	0	1	-	-
CT20P	0	-	-	-
N92	0	-	-	-
N86 Pro	0	-	-	-
N6 Pro	0	-	-	-
EF20	0	-	-	-

D20	-	10001	10003	10000
WIFI Base	-	10001	10003	10000

Note: if use Type-C, you need to enable the CDC mode in your device.

This module uses the basic flow chart:


Start
Obtain context
Obtain serial object
Call disconnect method, prevent serial port from disconnecting
Call connect to connect serial port
Call clrbuffer to clear buffer
Call send method to send data
Call recv to receive data
Call disconnect to disconnect

3.11.1 disconnect

Disconnect.

public int disconnect ();

Parameters: None

Return Value:

SdkResult.Success off successfully

SdkResult.SerialPort_Port_Not_Open serial port is not open

SdkResult.SerialPort_DisConnect_Fail serial chain disconnection failure

3.11.2 connect

Serial connection.

public int connect(SerialCfgEntity entity);

Parameters:

Parameter	Description
entity	SerialCfgEntity , Serial Info

SerialCfgEntity

Attributes	Description
int bauRate	The baud rate in the range of (bps): 110,300,600,1200,2400,4800, 9600,14400,56000,19200,38400,57600,115200,230400
int dataBits	Data Bits Range: 5, 6, 7, 8
char parity	Test methods in the range : 'o' odd , 'e' parity, 'n' no parity
int stopBits	Stop bit value range : 1, 2

Return Value:

SdkResult.Success serial connection success

SdkResult.Param_In_Invalid Parameter is null, illegal Parameter

SdkResult.SerialPort_Invalid_Communication_Parameter invalid communication Parameters

SdkResult.SerialPort_Connect_Fail serial connection failure

SdkResult.Fail other errors

3.11.3 clrBuffer

Clear the buffer.

public void clrBuffer ();

Parameters: None

Return Value: None

3.11.4 send

Send data.

public int send (byte [] data, int dataLen);

Parameters:

Parameter	Description
data	Input data
dataLen	Data length Range: 1-2048 bytes ; non-blocking send

Return Value:

SdkResult.Success sent successfully

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.SerialPort_Not_Open serial port is not open

SdkResult.SerialPort_Send_Fail serial data transmission failure

SdkResult.Fail other errors

3.11.5 recv

Receive data.

public int recv (byte [] buffer, int recvLen, long timeout);

Parameters:

Parameter	Description
buffer	Buffer to receive data
recvLen	The maximum length of buffer, which is 2048 bytes
timeout	Timeout in milliseconds; recommended value 3000

Return Value:

Successfully received returns the length of the received data

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.SerialPort_Not_Open serial port is not open

SdkResult.SerialPort_Timeout_Receiving_Data serial data receive timeout

SdkResult.Fail other errors

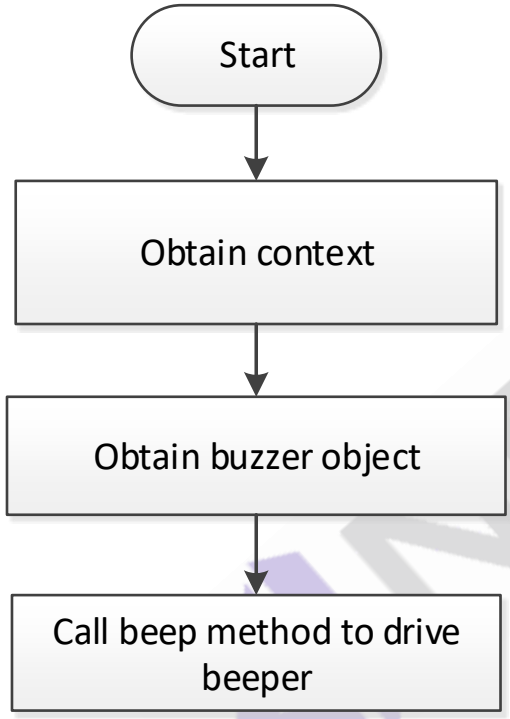
3.12 Buzzer class

Class is responsible for managing POS buzzer.

Get buzzer objects of class:

Beeper beep = deviceEngine.getBeeper();

This module uses the basic flow chart:



3.12.1 beep

Drive the buzzer sound duration specified length of time.

public void beep (int timeout);

Parameters:

Parameter	Description
timeout	Timeout in milliseconds. Zero immediately stops. Note: the timeout parameter is invalid.

Return Value: None

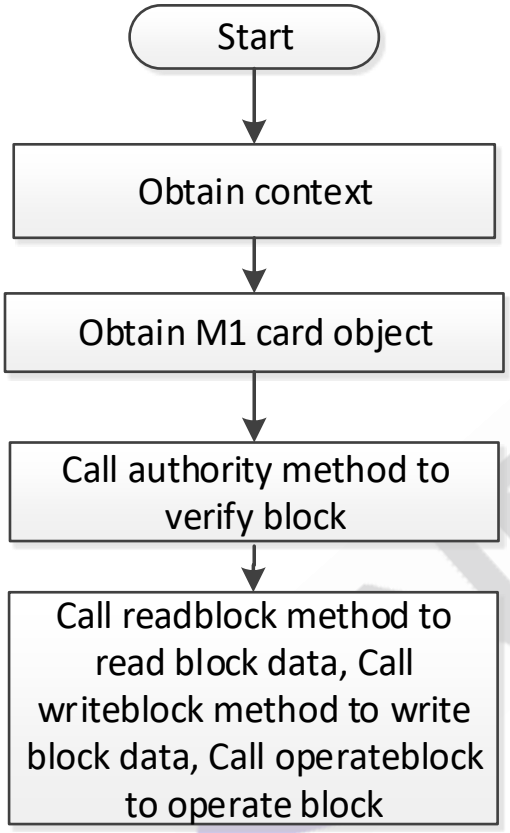
3.13 M1 Cards

M1 card class is responsible for managing M1 card (Mifare classic card).

Get M1 card class objects:

M1CardHandler m1Card = deviceEngine.getM1CardHandler();

This module uses the basic flow chart:



3.13.1 authority

Block certification.

public int authority (Authentity entity) ;

Parameters:

Parameter	Description
entity	AuthEntity, the authentication information class

AuthEntity

Attributes	Description

int blkNo	Block number
M1KeyTypeEnum keyType	Key type enumeration
byte [] pwd	Password authentication
String m1SN	M1 card uid

M1KeyTypeEnum

Enumeration Name	Description
KEYTYPE_A	KEY A
KEYTYPE_B	KEY B

Return Value:

SdkResult.Success success

SdkResult.Device_Not_Ready device is not ready

SdkResult.Param_In_InValid Parameter is not legitimate

SdkResult.M1Card_Verify_Err M1 card authentication failure

SdkResult.Fail other errors

3.13.2 readBlock

Read block data.

public int readBlock (Blockentity entity) ;

Parameters:

Parameter	Description
entity	BlockEntity block Info

BlockEntity

Attributes	Description
M1CardOperTypeEnum operType	Operation enumeration
byte [] blkData	Data to be operated
int BLKNO	Block number to be operated
int desBlkNo	Destination block number

M1CardOperTypeEnum

Enumeration Name	Description
INCREMENT	Increment
DECREMENT	Decrement operation
BACKUP	Backup

Return Value:

SdkResult.Success success

SdkResult.Device_Not_Ready device is not ready

SdkResult.Param_In_InValid Parameter is not legitimate

SdkResult.Fail other errors

3.13.3 readBlockValue

Read block value

public int readblockValue(Blockentity entity);

Parameters:

Parameter	Description
entity	BlockEntity block Info

BlockEntity

Attributes	**Description**
M1CardOperTypeEnum operType	Operation enumeration
byte[] blkData	Data to be operated
int blkValue	Read and write block data values in M1 card data format
int BLKNO	Block number to be operated
int desBlkNo	Destination block number

M1CardOperTypeEnum

Enumeration Name	**Description**
INCREMENT	Increment
DECREMENT	Decrement operation
BACKUP	Backup

Return Value:

SdkResult.Success success

SdkResult.Device_Not_Ready device is not ready

SdkResult.Param_In_InValid Parameter is not legitimate

SdkResult.Fail other errors

3.13.4 writeBlock

Write block data.

public int writeBlock (Blockentity entity) ;

Parameters:

Parameter	Description
entity	BlockEntity block Info

BlockEntity

Attributes	Description
M1CardOperTypeEnum operType	Operation enumeration
byte [] blkData	Data to be operated
int BLKNO	Block number to be operated
int desBlkNo	Destination block number

M1CardOperTypeEnum

Enumeration Name	Description
INCREMENT	Increment
DECREMENT	Decrement operation
BACKUP	Backup

Return Value:

SdkResult.Success success

SdkResult.Device_Not_Ready device is not ready

SdkResult.Param_In_InValid Parameter is not legitimate

SdkResult.Fail other errors

3.13.5 writeBlockValue

Write block value

public int writeblock(Blockentity entity);

Parameters:

Parameter	Description
entity	BlockEntity block Info

BlockEntity

Attributes	Description
M1CardOperTypeEnum operType	Operation enumeration
byte[] blkData	Data to be operated
int blkValue	Read and write block data values in M1 card data format
int BLKNO	Block number to be operated
int desBlkNo	Destination block number

M1CardOperTypeEnum

Enumeration Name	Description
INCREMENT	Increment
DECREMENT	Decrement operation
BACKUP	Backup

Return Value:

SdkResult.Success success

SdkResult.Device_Not_Ready device is not ready

SdkResult.Param_In_InValid Parameter is not legitimate

SdkResult.Fail other errors

3.13.6 operateBlock

Operation block data. If you want to increment or decrement operation, please call writeBlockValue interface first, it will change the block format, then can increment or decrement operation.

public int operateblock (Blockentity entity) ;

Parameters:

Parameter	Description
entity	BlockEntity block Info

BlockEntity

Attributes	Description
M1CardOperTypeEnum operType	Operation enumeration
byte [] blkData	Data to be operated
int BLKNO	Block number to be operated
int desBlkNo	Destination block number

M1CardOperTypeEnum

Enumeration Name	Description
INCREMENT	Increment
DECREMENT	Decrement operation
BACKUP	Backup

Return Value:

SdkResult.Success success

SdkResult.Device_Not_Ready device is not ready

SdkResult.Param_In_InValid Parameter is not legitimate

SdkResult.Fail other errors

3.14 MemoryCard

MemoryCard class is responsible for managing MemoryCard.

Get MemoryCard card class objects:

MemoryCard memoryCard = deviceEngine.getMemoryCardHandler (CardSlotTypeEnum slotType);

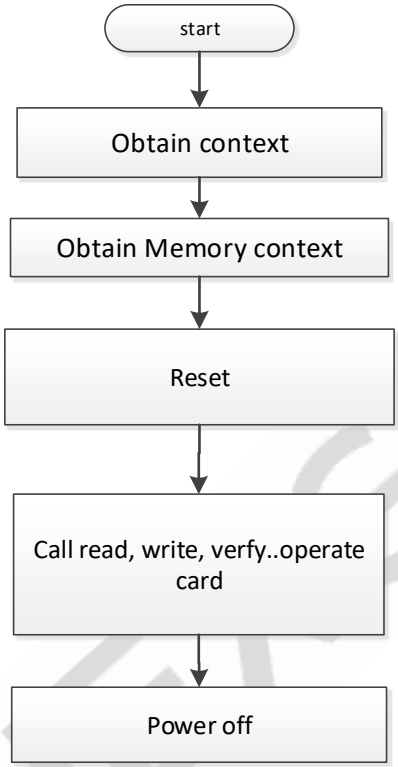
Parameters:

Parameter	Description
slotTypes	Slot enumerated type CardSlotTypeEnum; supports a variety of combinations of slots

CardSlotTypeEnum

Enumeration Name	Description
ICC1	Default IC card slot
ICC2	Unavailable
ICC3	Unavailable
PSAM1	PSAM slot 1
PSAM2	PSAM slot 2
PSAM3	Unavailable
RF	Non-access card slot
SWIPE	Magnetic stripe card slot

This module uses the basic flow chart:



3.14.1 reset

reset

public int reset(CardTypeEnum cardType);

Parameters:

Parameter	Description
cardType	Card type

CardTypeEnum

Enumeration Name	Description
AT24C01	
AT24C02	
AT24C04	
AT24C08	
AT24C16	
AT24C32	
AT24C64	

AT88SC101	
AT88SC102	
IS23SC1604	
AT88SC153	
AT88SC1608	
SLE4442	
SLE4428	

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.14.2 read

read

public byte[] read(ReadEntity read);

Parameters:

Parameter	Description
read	ReadEntity

ReadEntity

attribute	Description
CardTypeEnum cardType	Card type
int zone	SLE4428 means protection bit mode 0: no protection bit, 1: protection bit SLE4442 means the storage area, 0: main storage area, 1: protection area AT88SC153,AT88SC1608 means the partition number IS23SC1604 means area code
int address	The starting address, starting at 0
int readLen	Len of read data

CardTypeEnum

Enumeration Name	Description
AT24C01	
AT24C02	
AT24C04	
AT24C08	

AT24C16	
AT24C32	
AT24C64	
AT88SC101	
AT88SC102	
IS23SC1604	
AT88SC153	
AT88SC1608	
SLE4442	
SLE4428	

Return Value:

Success return byte[]

Fail return null

3.14.3 write

write

public int write(WriteEntity write);

Parameters:

Parameter	Description
write	WriteEntity

WriteEntity

attribute	Description
CardTypeEnum cardType	Card type
int zone	SLE4428 means protection bit mode 0: no protection bit, 1: protection bit SLE4442 means the storage area, 0: main storage area, 1: protection area AT88SC153,AT88SC1608 means the partition number IS23SC1604 means area code
int address	The starting address, starting at 0
byte[] writeData	Write data
int writeLen	Len of write data

CardTypeEnum

Enumeration Name	Description
AT24C01	
AT24C02	
AT24C04	

AT24C08	
AT24C16	
AT24C32	
AT24C64	
AT88SC101	
AT88SC102	
IS23SC1604	
AT88SC153	
AT88SC1608	
SLE4442	
SLE4428	

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.14.4 erase

erase

public int earse(EraseEntity erase);

Parameters:

Parameter	Description
erase	EraseEntity

EraseEntity

attribute	Description
CardTypeEnum cardType	Cardtype ,only support IS23SC1604,AT88SC101,AT88SC102
int address	The starting address, starting at 0
int eraseLen	Erase data length, unit byte
int zone	Zone number

CardTypeEnum

Enumeration Name	Description
AT24C01	
AT24C02	
AT24C04	
AT24C08	
AT24C16	
AT24C32	
AT24C64	

AT88SC101	
AT88SC102	
IS23SC1604	
AT88SC153	
AT88SC1608	
SLE4442	
SLE4428	

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid invaild param

SdkResult.Fail other errors

3.14.5 verify

verify card pin.

public int verify(VerifyEntity verify);

Parameters:

Parameter	Description
verify	VerifyEntity

VerifyEntity

attribute	Description
CardTypeEnum cardType	cardtype ,only support SLE4428,SLE4442,AT88SC153,AT88SC1608,IS23SC1604,AT88SC101,AT88SC102
byte[] pwd	password
int mode	AT88SC153,AT88SC1608 means check mode, 0: read check ;1: write check AT88SC101 AT88SC102, IS23SC1604 means check content 0: security code ;1: erase the password
int zone	AT88SC153,AT88SC1608 means password index AT88SC101 AT88SC102, IS23SC1604 means area ,0: the whole storage area; 1 ~ n: application code

CardTypeEnum

Enumeration Name	Description
AT24C01	
AT24C02	
AT24C04	
AT24C08	
AT24C16	
AT24C32	
AT24C64	
AT88SC101	
AT88SC102	
IS23SC1604	
AT88SC153	
AT88SC1608	
SLE4442	
SLE4428	

Return Value:

SdkResult.Success the remaining password verification times

SdkResult.Param_In_Invalid invaild param

SdkResult.Fail other errors

3.14.6 readEC

Read remaining password check times.

public int readEC(ReadECEntity readEC);

Parameters:

Parameter	Description
readEC	ReadECEntity

ReadECEntity

attribute	Description
CardTypeEnum cardType	card type , only support SLE4428,SLE4442,AT88SC153,AT88SC1608,IS23SC1604,AT88SC101,AT88SC102
int mode	AT88SC153,AT88SC1608 means check mode, 0: read check ;1: write check AT88SC101 AT88SC102, IS23SC1604 means check content 0: security code ;1: erase the password

int zone	AT88SC153,AT88SC1608 means password index AT88SC101 AT88SC102, IS23SC1604 means area ,0: the whole storage area; 1 ~ n: application code

CardTypeEnum

Enumeration Name	Description
AT24C01	
AT24C02	
AT24C04	
AT24C08	
AT24C16	
AT24C32	
AT24C64	
AT88SC101	
AT88SC102	
IS23SC1604	
AT88SC153	
AT88SC1608	
SLE4442	
SLE4428	

Return Value:

SdkResult.Success the remaining password verification times

SdkResult.Param_In_Invalid invaild param

SdkResult.Fail other errors

3.14.7 updateEC

Modify card password

public int updateEC(UpdateECEntity updateEC);

Parameters:

Parameter	Description
readEC	UpdateECEntity

UpdateECEntity

attribute	Description
CardTypeEnum cardType	Card type, only support SLE4428,SLE4442,AT88SC153,AT88SC1608,IS23SC160

	4,AT88SC101,AT88SC102
byte[] pwd	password
int mode	AT88SC153,AT88SC1608 means check mode, 0: read check ;1: write check AT88SC101 AT88SC102, IS23SC1604 means check content 0: security code ;1: erase the password
int zone	AT88SC153,AT88SC1608 means password index AT88SC101 AT88SC102, IS23SC1604 means area ,0: the whole storage area; 1 ~ n: application code

CardTypeEnum

Enumeration Name	Description
AT24C01	
AT24C02	
AT24C04	
AT24C08	
AT24C16	
AT24C32	
AT24C64	
AT88SC101	
AT88SC102	
IS23SC1604	
AT88SC153	
AT88SC1608	
SLE4442	
SLE4428	

Return Value:

SdkResult.Success

SdkResult.Param_In_Invalid invaild param

SdkResult.Fail other errors

3.14.8 powerOff

poweroff

public void powerOff();

Return Value:

None

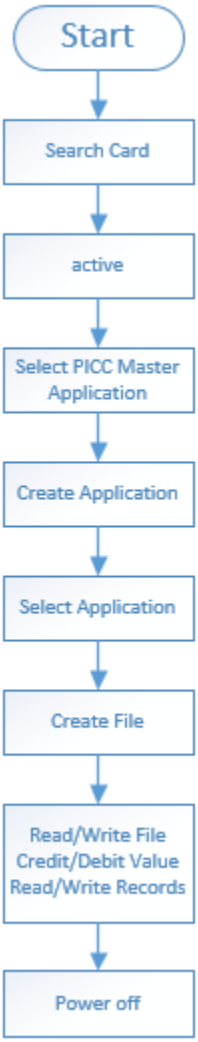
3.15 Desfire Cards

DesfireHandler is responsible for interacting with Desfire card.

Get Desfire card handler Object:

DesfireHandler desfireHandler = deviceEngine.getDesfireHandler();

This module uses the basic flow chart:


Start
Search Card
active
Select PICC Master Application
Create Application
Select Application
Create File
Read/Write File
Credit/Debit Value
Read/Write Records
Power off

3.15.1 Authenticate

Prototype	int authenticate(byte keyNo, byte[] key);	
Function	confirms that both entities (PICC and PCD) can trust each other, DES/3DES algorithm	
Parameters	keyNo	the key no used to authentication process
	Key	key used for authentication, 16bytes need. if the actual key is only 8bytes long, then should extended to 16bytes: key[0...7] || key[0...7].
Return	SdkResult.Success	Success
	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is Invalid
	SdkResult.TimeOut	TimeOut

3.15.2 AuthenticateIso

Prototype	int authenticateIso(byte keyNo, byte[] key);	
Function	confirms that both entities (PICC and PCD) can trust each other, DES/3DES /3KDES algorithm	
Parameters	keyNo	the key no used to authentication process
	Key	key used for authentication, 24bytes need. if the actual key is only 8bytes long, then should extended to 24bytes: key[0...7] || key[0...7] || key[0...7] if the actual key is only 16bytes long, then should extended to 24bytes: key[0...7] || key[8...15] || key[0...7]
Return	SdkResult.Success	Success

	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is Invalid
	SdkResult.TimeOut	TimeOut

3.15.3 AuthenticateAes

Prototype	int authenticateAes(byte keyNo, byte[] key);	
Function	confirms that both entities (PICC and PCD) can trust each other, AES algorithm	
Parameters	keyNo	the key no used to authentication process
	Key	key used for authentication, 16bytes need.
Return	SdkResult.Success	Success
	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is Invalid
	SdkResult.TimeOut	TimeOut

3.15.4 changeKeySettings

Prototype	int changeKeySettings(byte keySettings);	
Function	Changes the master key configuration settings depending on the currently selected AID.	
Parameters	keySettings	------ for PICC master key: bit7~bit4: 0000 RFU bit3: whether a change of the PICC master key settings is allowed bit2: whether PICC master key authentication is needed before Create- / DeleteApplication bit1: whether PICC master key authentication is needed

		for application directory access bit0: whether the PICC master key is changeable ------ for Application master key: bit7~bit4: hold the Access Rights for changing application keys (ChangeKey command) 0x0: Application master key authentication is necessary to change any key (default) 0x1~0x0D: Authentication with the specified key is necessary to change any key. 0x0E: Authentication with the key to be changed (same KeyNo) is necessary to change a key. 0x0F: All Keys (except application master key, see Bit0) within this application are frozen. bit3: whether a change of the application master key settings is allowed bit2: whether application master key authentication is needed before CreateFile / DeleteFile bit1: whether application master key authentication is needed for file directory access bit0: whether the application master key is changeable
Return	SdkResult.Success	Success
	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is Invalid
	SdkResult.TimeOut	TimeOut

3.15.5 getKeySettings

Prototype	KeySettingsEntity getKeySettings();	
Function	get configuration information on the PICC and application master key configuration settings, and get maximum number of keys which can be stored within the selected application. Depending on the master key settings, a preceding authentication with the master key is required.	
Parameters	null	
Return	KeySettingsEntity	getKeySettings current master key setting
		getMaxKeyNum maximum number of keys which can be stored within the selected application

3.15.6 changePiccMasterkey

Prototype	int changePiccMasterkey(KeyTypeEnum masterKeyType, byte[] key, byte aesVersion);	
Function	change PICC master key 1. according to PICC master key setting, a authentication with PICC master key is necessary 2. After a successful change of the key used to reach the current authentication status, this authentication is invalidated i.e. an authentication with the new key is necessary for subsequent operations.	
Parameters	masterKeyType	PICC master key type
	key	key information (16/24 bytes)
	aesVersion	key version, only valid when type = {@link KeyTypeEnum#AES}.
Return	SdkResult.Success	Success

	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is Invalid
	SdkResult.TimeOut	TimeOut

KeyTypeEnum

Enumeration Name	Description
DES_TDES_128	16bytes DES/3DES key
KEYTYPE_B	24bytes 3KDES key
AES	16bytes AES key

3.15.7 changeAppKey

Prototype	int changeAppKey(KeyTypeEnum appKeyType, byte keyNo, byte[] oldKey, byte[] newKey, byte aesVersion);	
Function	change application master key 1. according to application master key setting, a authentication with specified key is necessary 2. After a successful change of the key used to reach the current authentication status, this authentication is invalidated i.e. an authentication with the new key is necessary for subsequent operations.	
Parameters	appKeyType	app key type{ KeyTypeEnum}
	keyNo	the key to change
	oldKey	old key value
	newKey	new key value
	aesVersion	key version, only valid when type = {@link KeyTypeEnum#AES}.
Return	SdkResult.Success	Success
	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is Invalid

	SdkResult.TimeOut	TimeOut

3.15.8 getKeyVersion

Prototype	byte getKeyVersion(byte keyNo);	
Function	read out the current key version of any key stored on the PICC This command can be issued without valid authentication.	
Parameters	keyNo	key no
Return	key version of this key	

3.15.9 createApplication

Prototype	int createApplication(ApplicationEntity application);	
Function	create new applications on the PICC. 1. Application Identifier 0x00 00 00 is reserved as a reference to the PICC itself. 2. After application be created, All keys are initialised with a string consisting of 0x00 bytes 3. Before any setup of a file system, it is recommemded to configure the whole card using the command 'SetConfiguration'. This command will initialize all keys of any created application to a specified value which is taken out of the default key and default version from the 'SetConfiguration' command. Without this command all keys are consisting of 0x00 bytes.	
Parameters	application	{@link ApplicationEntity}
Return	SdkResult.Success	Success
	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is Invalid
	SdkResult.TimeOut	TimeOut

ApplicationEntity

Attributes	Description
byte[] aid	application identifier
byte[] isoFid	2 byte ISO/IEC 7816-4 File Identifies for files within the application
byte[]dfName	"DF-name" used in 7816-4 mode in combination with the ISO SELECT command
Byte masterKeySetting	Application master key setting
byte numberOfKey	Number of keys that can be stored within the application for cryptographic purposes. A maximum of 14 keys can be stored within an application of MIFARE DESFire EV1. One can also create an application with no keys
KeyTypeEnum keyType	the key type of application
boolean isSupFid	whether use File Identifies
boolean isSupIsoFid	whether use of 2 byte ISO/IEC 7816-4 File Identifies for files within the Application

3.15.10 deleteApplication

Prototype	int deleteApplication(byte[] aid);	
Function	allows to permanently deactivate applications on the PICC 1. Depending on the PICC master key settings, an PICC master key authentication is required. 2. The AID allocation is removed, therefore it is possible to create a new application with the deleted application's AID. However, the deleted memory blocks can only be recovered by using the FormatPICC command which erases the full user memory of the PICC.	
Parameters	aid	application Identifies (3bytes)
Return	SdkResult.Success	success
	SdkResult.Fail	fail
	SdkResult.Param_In_Invalid	Parameter is not legitimate

	SdkResult.TimeOut	timeout

3.15.11 getAids

Prototype	List<byte[]> getAids();	
Function	returns the Application IDentifiers of all active applications on a PICC. Depending on the PICC master key settings a successful authentication with the PICC master key might be required to execute this command.	
Parameters	null	
Return	if error return empty list.	

3.15.12 getDfNames

Prototype	List<DfNameEntity> getDfNames();	
Function	Returns the ISO/IEC 7816-4 DF-Names of all active applications on a PICC Depending on the PICC master key settings a successful authentication with the PICC master key might be required to execute this command.	
Parameters	null	
Return	if error return empty list.	

3.15.13 selectApplication

Prototype	int selectApplication(byte[] aid);
Function	select one specific application for further access. 1. each SelectApplication command invalidates the current authentication status. 2. If this pAID is 0x00 00 00, the PICC level is selected and any further operations (typically commands like CreateApplication, DeleteApplication) are related to this level.

	3. If an application with the specified AID is found in the application directory of the PICC, the subsequent commands interact with this application.	
Parameters	aid	Application Identifier (3bytes)
Return	SdkResult.Success	success
	SdkResult.Fail	fail
	SdkResult.Param_In_Invalid	Parameter is not legitimate
	SdkResult.TimeOut	timeout

3.15.14 formatPicc

Prototype	int formatPicc();	
Function	This command releases the PICC user memory. 1. The FormatPICC Command releases all allocated user memory on the PICC. 2. All applications are deleted and all files within those applications are deleted. 3. This command always requires a preceding authentication with the PICC master key. 4. The PICC master key and the PICC master key settings keep their currently set values, they are not influenced by this command.	
Parameters	null	
Return	SdkResult.Success	success
	SdkResult.Fail	fail
	SdkResult.Param_In_Invalid	Parameter is not legitimate
	SdkResult.TimeOut	timeout

3.15.15 getVersion

Prototype	VersionEntity getVersion();

Function	Return manufacturing related data of the PICC.	
Parameters	null	
Return	VersionEntity	{@link VersionEntity }

VersionEntity

Attributes	Description
byte hwVendorId	codes the vendor ID (0x04 for PHILIPS)
byte hwType	codes the type (here 0x01)
byte hwSubType	codes the subtype (here 0x01)
byte hwMajorVer	codes the major version number
byte hwMinorVer	codes the minor version number
byte hwSize	codes the storage size (here 0x1A = 8192 bytes)
byte hwProtocol	codes the communication protocol type (here 0x05 meaning ISO 14443-2 and -3)
byte swVendorId	codes the vendor ID (here 0x04 for PHILIPS)
byte swType	codes the type (here 0x01)
byte swSubType	codes the subtype (here 0x01)
byte swMajorVer	codes the major version
byte swMinorVer	codes the minor version
byte swSize	codes the storage size (here 0x1A = 8192 bytes)
byte swProtocol	codes the communication protocol type (here 0x05 meaning ISO 14443-3 and -4)
byte[] uid	code the unique serial number
byte[] batchNo	code the production batch number
byte weekOfProduction	codes the calendar week of production
byte yearOfProduction	codes the year of production

3.15.16 getFreeMemory

Prototype	int getFreeMemory();	
Function	Returns the available bytes on the PICC	
Parameters	null	
Return	SdkResult.Success	success
	SdkResult.Fail	fail
	SdkResult.Param_In_Invalid	Parameter is not legitimate
	SdkResult.TimeOut	timeout

3.15.17 setConfiguration

Prototype	int setConfiguration(byte option, byte[] info);	
Function	set PICC configuration PICC master key authentication on card level needs to be performed prior to this command.	
Parameters	option	configuration type, value as following 0x00: info data is the configuration byte 0x01: info data is the default key version and default key all applications will be personalized during creation with this default key and version instead of 0x00 0x02: info data is the user defined ATS 0xxx: RF
	info	configuration information, according to option if option = 0x00, the configuration byte showed as following: bit0 = 0 Format card enabled

		bit0 = 1 Format card disabled;can not be reset bit1 = 0 Random ID disabled bit1 = 1 Random ID enabled; can not be reset if option = 0x01, the *info should be 24bytes key and 1byte default version
Return	SdkResult.Success	success
	SdkResult.Fail	fail
	SdkResult.Param_In_Invalid	Parameter is not legitimate
	SdkResult.TimeOut	timeout

3.15.18 getCardUid

Prototype	byte[] getCardUid();	
Function	return the uid of PICC 1. An authentication with any key needs to be performed prior to this command 2. This command is only available when {@link DesfireHandler#authenticateIso(byte, byte[])} or {@link DesfireHandler#authenticateAes(byte, byte[])} called	
Parameters	null	
Return	uid information of PICC (7bytes)	

3.15.19 getFids

Prototype	List<byte[]> getFids();
Function	returns the File IDentifiers of all active files within the currently selected application. 1. Depending on the application master key settings, a preceding authentication with the application master key might be required.

	2. Each File ID is coded in one byte and is in the range from 0x00 to 0x1F.	
Parameters	null	
Return	if error return empty list.	

3.15.20 getIsoFids

Prototype	List<byte[]> getIsoFids();	
Function	Returns the 2 byte ISO/IEC 7816-4 File IDentifiers of all active files within the currently selected application 1. Depending on the application master key settings, a preceding authentication with the application master key might be required. 2. Each ISO File ID is coded in two byte .	
Parameters	null	
Return	if error return empty list.	

3.15.21 getFileSettings

Prototype	FileSettingsEntity getFileSettings(byte fileNo);	
Function	get information on the properties of a specific file. 1. This file number must be in the range between 0x00 and 0x1F. 2. Depending on the application master key settings, a preceding authentication with the application master key might be required. 3. After updating a value file's value but before issuing the CommitTransaction command, the GetFileSettings command will always retrieve the old, unchanged limit for the limited credit value.	
Parameters	fileNo	the specific file no, value 0~0x1F allowed
Return	FileSettingsEntity	

FileSettingsEntity

Attributes	Description
byte fileType	DESfire file type: 0x00 --- Standard Data Files 0x01 --- Backup Data Files 0x02 --- Value Files wih Backup 0x03 --- Linear record Files with Backup 0x04 --- Cyclic Record Files with Backup
byte commSettings	0x00 or 0x02 --- Plain communication 0x01 --- Plain communication secured by MACing 0x03 --- Fully enciphered communication
byte readAccessRightKeyNum	Access right capability, 0x0E means free access, and 0x0F means deny access. the reference number of the key which needs to be authenticated prior to Read Access and Read&Write Access
byte writeAccessRightKeyNum	the reference number of the key which needs to be authentication prior to Write Access and Read&Write Access
Byte readAndWriteAccessRightKeyNum	the reference number of the key which needs to be authentication prior to Read&Write Access
byte changeAccessRightKeyNum	the reference number of the key,which is necessary to be authenticated with in order to change the access rights for the file and to link each access right to key numbers
int fileSize	the user file size in bytes, only available when file_type = 0x00 or file_type = 0x01
int lowerLimit	lower limit of the value file ,only available when file_type = 0x02
int upperLimit	upper limit of the value file,only available when file_type = 0x02
the current maximum" limited credit" value	limitedCreditValue,only available when file_type = 0x02
boolean limitedCreditEnabled	if the LimitedCredit command is allowed for this file,only available when file_type = 0x03 or file_type = 0x04
int recordSize	the size of one single record (as deefined at file creation),only available when file_type = 0x03 or file_type = 0x04
int maxNumberOfRecords	the maximum number of records within the record file (as defined at file creation),only available when file_type = 0x03 or file_type = 0x04

int currentNumberOfRecords	the current number of records within the record file,only available when file_type = 0x03 or file_type = 0x04

3.15.22 changeFileSettings

Prototype	int changeFileSettings(byte fileNo, byte commSettings, byte newReadAccessKeyNum, byte newWriteAccessKeyNum, byte newReadAndWriteAccessKeyNum, byte newChangeAccessKeyNum);	
Function	changes the access parameters of an existing file 1. This change only succeeds if the current "Change Access Right" is different from "never", that is old_change_access_keyno != 0x0E 2. To guarantee that the ChangeFileSettings command is coming from the same party which did the preceding authentication, it is necessary to apply basically the same security mechanism as used with the ChangeKey command	
Parameters	fileNo	the specific file no, value 0~0x1F allowed
	commSettings	0x00 or 0x02 --- Plain communication 0x01 --- Plain communication secured by MACing 0x03--- Fully enciphered communication
	newReadAccessKeyNum	new Read Access Right Key No
	newWriteAccessKeyNum	new Write Access Right Key No
	newReadAndWriteAccessKeyNum	new Read and Write Access Right Key No
	newChangeAccessKeyNum	new Change Access Right Key No
Return	SdkResult.Success	success
	SdkResult.Fail	fail
	SdkResult.Param_In_Invalid	Parameter is not legitimate
	SdkResult.TimeOut	timeout

3.15.23 createStdDataFile

Prototype	int createStdDataFile(byte fileNo, DataFileEntity dataFile);	
Function	create files for the storage of plain unformatted user data within an existing application on the PICC	
Parameters	fileNo	the specific file no, value 0~0x1F allowed
	dataFile	file settings {@link DataFileEntity}
Return	SdkResult.Success	success
	SdkResult.Fail	fail
	SdkResult.Param_In_Invalid	Parameter is not legitimate
	SdkResult.TimeOut	timeout

DataFileEntity

Attributes	Description
boolean isoFidEnable	whether ISO/IEC 7816-4 File IDentifiers enabled (0x00 - disabled, 0x01-enabled)
byte[] isoFid	2bytes ISO/IEC 7816-4 File IDentifiers
commSettings	0x00 or 0x02 --- Plain communication 0x01 --- Plain communication secured by MACing 0x03--- Fully enciphered communication
byte readAccessRightKeyNum	Access right capability, 0x0E means free access, and 0x0F means deny access. the reference number of the key which needs to be authenticated prior to Read Access and Read&Write Access
byte writeAccessRightKeyNum	the reference number of the key which needs to be authentication prior to Write Access and Read&Write Access

Byte readAndWriteAccessRightKeyNum	the reference number of the key which needs to be authentication prior to Read&Write Access
byte changeAccessRightKeyNum	the reference number of the key,which is necessary to be authenticated with in order to change the access rights for the file and to link each access right to key numbers
int fileSize	the user file size in bytes, only available when file_type = 0x00 or file_type = 0x01

3.15.24 createBackupDatafile

Prototype	int createBackupDatafile(byte fileNo, DataFileEntity dataFile);	
Function	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 1. Due to the mirror image a BackupDataFile always consumes DOUBLE the NV-memory on the PICC compared to a StdDataFile with the same specified FileSize. 2. Every Write command is done in a independent mirror image of this file. To validate a write access to this file type, it is necessary to confirm it with a CommitTransaction command. If no CommitTransaction command is send by the PCD, only the mirror image is changed, the original data remains unchanged and valid.	
Parameters	fileNo	the specific file no, value 0~0x1F allowed
	dataFile	file settings {@link DataFileEntity}
Return	SdkResult.Success	success
	SdkResult.Fail	fail
	SdkResult.Param_In_Invalid	Parameter is not legitimate
	SdkResult.TimeOut	timeout

3.15.25 createValueFile

Prototype	int createValueFile(byte fileNo, ValueFileEntity valueFile);	
Function	create files for the storage and manipulation of 32bit signed integer values within an existing application on the PICC ValueFiles feature always the integrated backup mechanism. Therefore every access changing the value needs to be validated using the CommitTransaction command 1. It is necessary to validate the updated value with a CommitTransaction command. An AbortTransaction command will invalidate all changes 2. The value modifications of Credit, Debit and LimitedCredit commands are cumulated until a CommitTransaction command is issued.	
Parameters	fileNo	the specific file no, value 0~0x1F allowed
	valueFile	file settings {@link ValueFileEntity }
Return	SdkResult.Success	success
	SdkResult.Fail	fail
	SdkResult.Param_In_Invalid	Parameter is not legitimate
	SdkResult.TimeOut	timeout

ValueFileEntity

Attributes	Description
commSettings	0x00 or 0x02 --- Plain communication 0x01 --- Plain communication secured by MACing 0x03--- Fully enciphered communication
byte readAccessRightKeyNum	Access right capability, 0x0E means free access, and 0x0F means deny access. the reference number of the key which needs to be authenticated prior to Read Access and Read&Write Access
byte writeAccessRightKeyNum	the reference number of the key which needs to be authentication prior to Write Access and Read&Write Access

Byte readAndWriteAccessRightKeyNum	the reference number of the key which needs to be authentication prior to Read&Write Access
byte changeAccessRightKeyNum	the reference number of the key,which is necessary to be authenticated with in order to change the access rights for the file and to link each access right to key numbers
int lowerLimit	lower limit of the value file, only available when file_type = 0x02
int upperLimit	upper limit of the value file, only available when file_type = 0x02
int initValue	the initial value of this value file, only available when file_type = 0x02
boolean limitedCreditEnabled	if the LimitedCredit command is allowed for this file, only available when file_type = 0x02

3.15.26 createLinearRecordFile

Prototype	int createLinearRecordFile(byte fileNo, RecordFileEntity recordFile);	
Function	create files for multiple storage of structural data, for example for loyalty programs, within an existing application on the PICC 1. Once the file is filled completely with data records, further writing to the file is not possible unless it is cleared, see command ClearRecordFile. 2. Linear Record Files feature always the integrated backup mechanism. Therefore every access appending a record needs to be validated using the CommitTransaction command	
Parameters	fileNo	the specific file no, value 0~0x1F allowed
	valueFile	file settings {@link ValueFileEntity }
Return	SdkResult.Success	success
	SdkResult.Fail	fail
	SdkResult.Param_In_Invalid	Parameter is not legitimate
	SdkResult.TimeOut	timeout

RecordFileEntity

Attributes	Description
boolean isoFidEnable	whether ISO/IEC 7816-4 File IDentifiers enabled (0x00 - disabled, 0x01-enabled)
byte[] isoFid	2bytes ISO/IEC 7816-4 File IDentifiers
commSettings	0x00 or 0x02 --- Plain communication 0x01 --- Plain communication secured by MACing 0x03--- Fully enciphered communication
byte readAccessRightKeyNum	Access right capability, 0x0E means free access, and 0x0F means deny access. the reference number of the key which needs to be authenticated prior to Read Access and Read&Write Access
byte writeAccessRightKeyNum	the reference number of the key which needs to be authentication prior to Write Access and Read&Write Access
Byte readAndWriteAccessRightKeyNum	the reference number of the key which needs to be authentication prior to Read&Write Access
byte changeAccessRightKeyNum	the reference number of the key,which is necessary to be authenticated with in order to change the access rights for the file and to link each access right to key numbers
int recordSize	the size of one single record (as deefined at file creation), only available when file_type = 0x03 or file_type = 0x04
int maxNumberOfRecords	the maximum number of records within the record file (as defined at file creation) , only available when file_type = 0x03 or file_type = 0x04
byte specifiesRandomWriteAccessOption	whether specifies Random write access option, (0x00 - not, 0x01 - yes), only available when file_type = 0x03 or file_type = 0x04
boolean allowedRandomWriteAccess	whether allowed Random write access, only available when file_type = 0x03 or file_type = 0x04

3.15.27 createCyclicRecordFile

Prototype	int createCyclicRecordFile(byte fileNo, RecordFileEntity recordFile);	
Function	create files for multiple storage of structural data, for example for loyalty programs, within an existing application on the PICC 1. Once the file is filled completely with data records, further writing to the file is not possible unless it is cleared, see command ClearRecordFile. 2. Linear Record Files feature always the integrated backup mechanism. Therefore every access appending a record needs to be validated using the CommitTransaction command	
Parameters	fileNo	the specific file no, value 0~0x1F allowed
	valueFile	file settings {@link ValueFileEntity }
Return	SdkResult.Success	success
	SdkResult.Fail	fail
	SdkResult.Param_In_Invalid	Parameter is not legitimate
	SdkResult.TimeOut	timeout

3.15.28 deleteFile

Prototype	int deleteFile(byte fileNo);
Function	permanently deactivates a file within the file directory of the currently selected application. 1. The operation of this command invalidates the file directory entry of the specified file which means that the file can't be accessed anymore. 2. Depending on the application master key settings, a preceding authentication with the application master key is required. 3. Allocated memory blocks associated with the deleted file are not set free. The FileNo of the deleted file can be re-used to create a new file within that application.

	4. To release memory blocks for re-use, the whole PICC user NV-memory needs to be erased using the FormatPICC command.	
Parameters	fileNo	the file number within the file directory of the currently selected application.
Return	SdkResult.Success	success
	SdkResult.Fail	fail
	SdkResult.Param_In_Invalid	Parameter is not legitimate
	SdkResult.TimeOut	timeout

3.15.29 readData

Prototype	byte[] readData(byte fileNo, byte commSettings, int offset, int len);	
Function	Read data from Standard Data Files or Backup Data Files 1. This offset has to be in the range from 0 to file size -1. 2. If the len is coded as 0, the entire data file, starting from the position specified in the offset value, is read. 3. If Backup Data Files are read after writing to them, but before issuing the CommitTransaction command, the ReadData command will always retrieve the old, unchanged data stored in the PICC. All data written to a Backup Data File is validated and externally "visible" for a ReadData command only after a CommitTransaction command. 4. The Read command requires a preceding authentication either with the key specified for "Read" or "Read&Write" access	
Parameters	fileNo	the file number
	commSettings	0x00 or 0x02 --- Plain communication 0x01 --- Plain communication secured by MACing 0x03 --- Fully enciphered communication
	offset	the starting position for the read operation within the file

	len	the number of data bytes want to be read
Return	return the out data	

3.15.30 writeData

Prototype	int writeData(byte fileNo, byte commSettings, int offset, byte[] data);	
Function	Write data to Standard Data Files and Backup Data Files. 1. The Write command requires a preceding authentication either with the key specified for "Write" or "Read&Write" access. 2. If the WriteData operation is performed on a Backup Data File, it is necessary to validate the written data with a CommitTransaction command. An AbortTransaction command will invalidate all changes.	
Parameters	fileNo	the file number
	commSettings	0x00 or 0x02 --- Plain communication 0x01 --- Plain communication secured by MACing 0x03 --- Fully enciphered communication
	offset	the starting position for the write operation within the file
	data	Data to send
Return	SdkResult.Success	Success
	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is invalid
	SdkResult.TimeOut	timeout

3.15.31 getValue

Prototype	int getValue(byte fileNo, byte commSettings);

Function	Read the currently stored value from Value Files. 1. The GetValue command requires a preceding authentication with the key specified for Read, Write or Read&Write access 2. After updating a value file's value but before issuing the CommitTransaction command, the GetValue command will always retrieve the old, unchanged value which is still the valid one.	
Parameters	fileNo	the file number
	commSettings	0x00 or 0x02 --- Plain communication 0x01 --- Plain communication secured by MACing 0x03 --- Fully enciphered communication
Return	SdkResult.Success	Success
	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is invalid
	SdkResult.TimeOut	timeout

3.15.32 credit

Prototype	int credit(byte fileNo, byte commSettings, int value);	
Function	Increase a value stored in a Value File. 1. Credit commands do NEVER modify the Limited Credit Value of a Value file. However, if the Limited Credit Value needs to be set to 0, a LimitedCredit with value 0 can be used. 2. The Credit command requires a preceding authentication with the key specified for "Read&Write" access.	
Parameters	fileNo	the file number
	commSettings	0x00 or 0x02 --- Plain communication 0x01 --- Plain communication secured by MACing 0x03 --- Fully enciphered communication

	value	the value which will be subtracted from the current value stored in the file. Only positive values are allowed for the Credit command.
Return	SdkResult.Success	Success
	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is invalid
	SdkResult.TimeOut	timeout

3.15.33 debit

Prototype	int debit(byte fileNo, byte commSettings, int value);	
Function	Decrease a value stored in a Value File. 1. The Debit command requires a preceding authentication with one of the keys specified for Read, Write or Read&Write access. 2. If the usage of the LimitedCredit feature is enabled, the new limit for a subsequent LimitedCredit command is set to the sum of Debit commands within one transaction before issuing a CommitTransaction command. This assures that a LimitedCredit command can not re-book more values than a debiting transaction deducted before.	
Parameters	fileNo	the file number
	commSettings	0x00 or 0x02 --- Plain communication 0x01 --- Plain communication secured by MACing 0x03 --- Fully enciphered communication
	value	the value which will be subtracted from the current value stored in the file. Only positive values are allowed for the Credit command.
Return	SdkResult.Success	Success
	SdkResult.Fail	Fail

	SdkResult.Param_In_Invalid	Parameter is invalid
	SdkResult.TimeOut	timeout

3.15.34 limitedCredit

Prototype	int limitedCredit(byte fileNo, byte commSettings, int value);	
Function	Allows a limited increase of a value stored in a Value File without having full Read&Write permissions to the file. This feature can be enabled or disabled during value file creation. 1. The LimitedCredit command requires a preceding authentication with the key specified for "Write" or "Read&Write" access. 2. The value for LimitedCredit is limited to the sum of the Debit commands on this value file within the most recent transaction containing at least one Debit. After executing the LimitedCredit command the new limit is set to 0 regardless of the amount which has been re-booked. Therefore the LimitedCredit command can only be used once after a Debit transaction.	
Parameters	fileNo	the file number
	commSettings	0x00 or 0x02 --- Plain communication 0x01 --- Plain communication secured by MACing 0x03 --- Fully enciphered communication
	value	the value which will be subtracted from the current value stored in the file. Only positive values are allowed for the Credit command.
Return	SdkResult.Success	Success
	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is invalid
	SdkResult.TimeOut	timeout

3.15.35 writeRecord

Prototype	int writeRecord(byte fileNo, byte commSettings, int offset, int len, byte[] record);	
Function	The WriteRecord command allows to write data to a record in a Cyclic or Linear Record File. 1. The WriteRecord command appends one record at the end of the linear record file, it erases and overwrites the oldest record in case of a cyclic record file if it is already full. The entire new record is cleared before data is written to it. 2. If no CommitTransaction command is sent after a WriteRecord command, the next WriteRecord command to the same file writes to the already created record. After sending a CommitTransaction command, a new WriteRecord command will create a new record in the record file. An AbortTransaction command will invalidate all changes 3. After issuing a ClearRecordFile command, but before a CommitTransaction / AbortTransaction command, a WriteRecord command to the same record file will fail. 4. The WriteRecord command requires a preceding authentication either with the key specified for "Write" or "Read&Write" access.	
Parameters	fileNo	the file number
	commSettings	0x00 or 0x02 --- Plain communication 0x01 --- Plain communication secured by MACing 0x03 --- Fully enciphered communication
	offset	the offset within one single record, the value has to be in therange from 0 to record size - 1.
	len	the length of data which is to be written to the record file, the value has to be in the range from 1 to record size - offset.
	record	Record Information
Return	SdkResult.Success	Success
	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is invalid
	SdkResult.TimeOut	timeout

3.15.36 readRecords

Prototype	byte[] readRecords(byte fileNo, byte commSettings, int recordSize, int first, int num);	
Function	The ReadRecords command allows to read out a set of complete records from a Cyclic or Linear Record File. 1. In cyclic record files the maximum number of stored valid records is one less than the number of records specified in the CreateCyclicRecordFile command. 2. A ReadRecords command on an empty record file (directly after creation or after a committed clearance will result in an error. 3. The ReadRecords command requires a preceding authentication either with the key specified for "Read" or "Read&Write" access.	
Parameters	fileNo	the file number
	commSettings	0x00 or 0x02 --- Plain communication 0x01 --- Plain communication secured by MACing 0x03 --- Fully enciphered communication
	recordSize	the size of single record
	first	the first record which is read out. In case of 0x00 the latest record is read out. The value must be in the range from 0x00 to number of existing records - 1.
	num	the number of records to be read from the PICC. Records are always transmitted by the PICC in chronological order (= starting with the oldest, which is number of records ¨C 1 before the one addressed by the given offset). If this parameter is set to 0x00 then all records, from the oldest record up to and including the newest record(given by the offset parameter) are read.

Return	return Record Information

3.15.37 clearRecordFile

Prototype	int clearRecordFile(byte fileNo);	
Function	The ClearRecordFile command allows to reset a Cyclic or Linear Record File to the empty state. 1. After executing the ClearRecordFile command but before CommitTransaction, all subsequent WriteRecord commands will fail. 2. The ReadRecords command will return the old still valid records. 3. After the CommitTransaction command is issued, a ReadRecords command will fail, WriteRecord commands will be successful. 4. An AbortTransaction command (instead of CommitTransaction) will invalidate the clearance	
Parameters	fileNo	the file number
Return	SdkResult.Success	Success
	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is invalid
	SdkResult.TimeOut	timeout

3.15.38 commitTransaction

Prototype	int commitTransaction();
Function	The CommitTransaction command allows to validate all previous write access on Backup Data Files, Value Files and Record Files within one application. The CommitTransaction is typically the last command of a transaction before the ISO 14443-4 Deselect command or before proceeding with another application (SelectApplication command).

Parameters	null	
Return	SdkResult.Success	Success
	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is invalid
	SdkResult.TimeOut	timeout

3.15.39 abortTransaction

Prototype	int abortTransaction();	
Function	The AbortTransaction command allows to invalidate all previous write access on Backup Data Files, Value Files and Record Files within one application. This is useful to cancel a transaction without the need for re-authentication to the PICC, which would lead to the same functionality.	
Parameters	null	
Return	SdkResult.Success	Success
	SdkResult.Fail	Fail
	SdkResult.Param_In_Invalid	Parameter is invalid
	SdkResult.TimeOut	timeout

3.16 Mifare Ultralight card

The Ultralight module class is responsible for managing operate the Mifare Ultralight card.

UltralightCCardHandler ultralightCCardHandler = deviceEngine.getUltralightCCardHandler();

3.16.1 authority

Block certification.

public int authority(byte[] keyData);

Parameters:

Parameter	Description
keyData	Password authentication, 16 bytes(hex)

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.16.2 readBlock

Read block data.

public byte[] readBlock(byte blockNum);

Parameters:

Parameter	Description
blockNum	block number

Return Value:

Success, return block data

Failed, return null

3.16.3 writeBlock

Write block data.

public int writeBlock(byte blockNum, byte[] writeData);

Parameters:

Parameter	Description
blockNum	Block number
writeData	Write data

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.16.4 exchangeCmd

Exchange data with command(use original command communicate with the card directly)

public byte[] exchangeCmd(byte[] cmdData);

Parameters:

Parameter	Description
cmdData	Command data send to card

Return Value:

Success, return response data

Failed, return null

3.17 NTAG card

The NTAG module class is responsible for managing operate the NTAG card.

NTAGCardHandler cardHandler = deviceEngine.getNTAGCardHandler();

3.17.1 authority

Block certification.

public int authority(byte[] keyData);

Parameters:

Parameter	Description
keyData	Password authentication, 16 bytes(hex)

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.17.2 getCardVersion

Get the card version

public byte[] getCardVersion();

Return Value:

success: return the card version

failed: return null

3.17.3 read

Read block data.

public byte[] read (byte address);

Parameters:

Parameter	Description
address	block number

Return Value:

Success, return block data

Failed, return null

3.17.4 fastRead

Fast Read block data.

public byte[] fastRead (byte startAddress, byte endAddress);

Parameters:

Parameter	Description
startAddress	The start address which you want to read
endAddress	The end address which you want to read

Return Value:

Success, return block data

Failed, return null

3.17.5 write

Write block data.

public int writeBlock(byte address, byte[] writeData, boolean isCompatibility);

Parameters:

Parameter	Description
address	Block number
writeData	Write data
isCompatibility	true: Compatibility

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.17.6 readCNT

read counter

public byte[] readCNT(byte address);

Parameters:

Parameter	Description
address	Block number

Return Value:

success: return the card counter

failed: return null

3.17.7 exchangeCmd

Exchange data with command(use original command communicate with the card directly)

public byte[] exchangeCmd(byte[] cmdData);

Parameters:

Parameter	Description
cmdData	Command data send to card

Return Value:

Success, return response data

Failed ,return null

3.18 Platform

The platform module class is responsible for managing operate the device function, such as install application, uninstall application , reboot device, update firmware..etc.

Platform platform = deviceEngine.getPlatform();

3.18.1 initPlatform

Platform initialize.

After initialize success, can call all the other APIs, such as installApp, disableControlBar, etc

void initPlatform(OnPlatformInitListener listener);

Parameters:

Parameter	Description
listener	OnPlatformInitListener, to identify the platform initialization success

Return Value: None

Note:

If your application encounters a situation where the platform function does not work(SDK Throws exception), please make sure that the application first calls the initPlatform interface. After initialization is successful, then call other functional methods within the platform.

3.18.2 installApp

Install application.

public int installApp(String appFilePath, final OnAppOperatListener listener);

Parameters:

Parameter	Description
appFilePath	The path of the application.

	Note: the application must be signed if you want to install in production devices Example: Environment.getExternalStorageDirectory().getPath() + "/" + "demo.apk"
listener	Install result callback

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid parameter invalid

SdkResult.Fail other errors

Other error code, please refer to Appendix

3.18.3 unInstallApp

Uninstall application.

public int uninstallApp(String appPackageName, final OnAppOperatListener listener);

Parameters:

Parameter	Description
appPackageName	Thepackae name of the application which you want to uninstall
listener	Uninstall result callback

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid parameter invalid

SdkResult.Fail other errors

Other error code, please refer to Appendix

3.18.4 updateFirmware

update firmware. The firmware must be provided by Nexgo .

public int updateFirmware(String firmwareFilePath);

Parameters:

Parameter	Description
firmwareFilePath	The path of the firmware which you want to update Example: Environment.getExternalStorageDirectory().getPath() + "/"

	+"xx_custom_ en0002.zip" Note: the firmware must be provided by Nexgo

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid parameter invalid

SdkResult.Fail other errors

Other error code, please refer to Appendix

3.18.5 reboot

Reboot device

public int rebootDevice();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.6 shutDownDevice

power off device

public int shutDownDevice();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.7 enableMobileData

enable mobile data

public int enableMobileData ();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.8 disableMobileData

disable mobile data

public int disableMobileData ();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.9 enableDataRoaming

enable data roaming

public int enableDataRoaming ();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.10 disableDataRoaming

disable data roaming

public int disableDataRoaming ();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.11 enableAirplaneMode

enable airplane mode

public int enableAirplaneMode ();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.12 disableAirplaneMode

disable airplane mode

public int disableAirplaneMode ();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.13 getAirplaneModeStatus

get airplane mode status

public int getAirplaneModeStatus ();

Return Value:

True: enable the airplane mode

False: disable the airplane mode

3.18.14 enableHomeButton

enable home button

public int enableHomeButton();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.15 disableHomeButton

disable home button

public int disableHomeButton();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.16 enableTaskButton

enable task button

public int enableTaskButton();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.17 disableTaskButton

disable task button

public int disableTaskButton();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.18 enableControlBar

enable controlBar (dropdown menu)

public int enableControlBar();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.19 disableControlBar

disable controlBar (dropdown menu)

public int disableControlBar();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.20 enablePowerButton

enable power button

public int enablePowerButton;

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.21 disablePowerButton

disable power button

public int disablePowerButton();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.22 setBeepMode

set device beep mode, application can control the beep volume

public void setBeepMode(BeepVolumeModeEnum beepMode, int volume);

Parameters:

Parameter	Description
beepMode	BEEP_MODE_SYSTEM_DEFAULT: system default BEEP_MODE_CUSTOM: application set the beep volume with fix value BEEP_MODE_SYSTEM_VOLUME: beep volume will follow with system volume
volume	Beep volume, range 0 -100 (volume percentage)

BeepVolumeModeEnum

Enumeration Name	Description
BEEP_MODE_SYSTEM_DEFAULT	use 80% of the maximum volume of the system, regardless of the volume of the system
BEEP_MODE_CUSTOM	application set the beep volume, need to set the volume value
BEEP_MODE_SYSTEM_VOLUME	Beep sound following system volume

Return Value:

None

3.18.23 enableUsbCdc

enable usb cdc option

public int enableUsbCdc ();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.24 disableUsbCdc

disable usb cdc option

public int disableUsbCdc ();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.18.25 getUsbCdcStatus

get usb cdc status

public boolean getUsbCdcStatus ();

Return Value:

true enable

false disable

3.18.26 showNavigationBar

Show bottom Navigation Bar.

public void showNavigationBar ();

Return Value:

None

3.18.27 hideNavigationBar

Hide bottom Navigation Bar.

public void hideNavigationBar ();

Return Value:

None

3.18.28 switchMobileDataNetwork

Switch mobile network , Sim card 1 to sim card 2.

public void switchMobileDataNetwork(int mobileDataSlot);

Parameters:

Parameter	Description
mobileDataSlot	Sim card slot , Use SIM 1: mobileDataSlot = 0 Use SIM 2: mobileDataSlot = 1

Return Value:

None

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.18.29 setNetworkStatusListener

set network status listener

void setNetworkStatusListener(onMobileDataNetworkListener onMobileDataNetworkListener);

Parameters:

Parameter	Description
onMobileDataNetworkListener	Callback of the network status

Return Value:

None

3.18.30 stopNetworkStatusListener

stop network status listener

void stopNetworkStatusListener ();

Return Value:

None

3.18.31 customBlindPinPadAudioFile

custom blind pinpad audio files. These files must put in assets folder.

boolean customBlindPinPadAudioFile(BlindPinPadAudioEntity audioEntity);

Parameters:

Parameter	Description
audioEntity	BlindPinPadAudioEntity

BlindPinPadAudioEntity

Parameter	Description
number5AudioName	Optional, the audio file path of number 5 button
leftAreaAudioName	Optional, the audio file path of left area, outside the pinpad button area
centralAreaAudioName	Optional, the audio file path of central area, outside the pinpad button area
rightAreaAudioName	Optional, the audio file path of right area, outside the pinpad button area
cancelAudioName	Optional, the audio file path of cancel button
clearAudioName	Optional, the audio file path of clear button
okAudioName	Optional, the audio file path of OK button.

Return Value:

true, setup audio file success

false, setup audio file failed

3.18.32 removeAllBlindPinPadAudioFile

Remove all blind pinpad audio file, then the blind pinpad will use the default audio.

void removeAllBlindPinPadAudioFile();

Return Value:

None

3.18.33 executeGeneralMethod

execute general method

public int executeGeneralMethod(int cmd, byte[] inParam, byte[] otherParam, byte[] outData)

Parameters:

Parameter	Description
cmd	Cmd
byte[] inParam	Depends on the cmd
byte[] otherParam	Depends on the cmd
byte[] outData	Depends on the cmd

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid parameter invalid

SdkResult.Fail other errors

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.18.34 executeGeneralMethod

execute general method with callback

public int executeGeneralMethod(int cmd, byte[] inParam, byte[] otherParam, byte[] outData , PlatformGeneralCallback generalCallback);

Parameters:

Parameter	Description
cmd	Cmd , Nexgo will specify the value of the CMD
byte[] inParam	Depends on the cmd
byte[] otherParam	Depends on the cmd
byte[] outData	Depends on the cmd
generalCallback	

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid parameter invalid

SdkResult.Fail other errors

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.18.35 grantLogPermission

get log premission

int grantLogPermission(String packageName);

Parameters:

Parameter	Description
packageName	Application package name

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid parameter invalid

SdkResult.Fail other errors

Other error code, please refer to Appendix

3.18.36 forceStopApp

force stop application

int forceStopApp(String packageName);

Parameters:

Parameter	Description
packageName	Application package name

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid parameter invalid

SdkResult.Fail other errors

Other error code, please refer to Appendix

3.18.37 setSystemClock

Set the system time.

public void setSystemClock (String datetime);

Parameters:

Parameter	Description
datetime	Time format YYYYMMDDHHMMSS, the year in the range 1970-2049

Return Value: None

3.18.38 isSupportBlindPinPad

Check terminal support Blind PinPad or not

boolean isSupportBlindPinPad();

Return Value:

True: support

False: not support

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.18.39 isSupportRkl

Check terminal support RKL(remote ket loading) or not

boolean isSupportRkl();

Return Value:

True: support

False: not support

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.18.40 enableIccPsamPPSControl

enable terminal ICC(contact chip card) or PSAM PPS to improve the card reading speed.

PPS:Protocol and Parameters Selection

int enableIccPsamPPSControl(boolean isEnable, CardSlotTypeEnum cardSlotTypeEnum);

Parameters:

Parameter	Description
isEnable	Enable PPS or not; True: enable False:disable
cardSlotTypeEnum	CardSlotTypeEnum. ICC1: ICC chip CardSlotTypeEnum. PSAM1: PSAM

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid parameter invalid

SdkResult.Fail other errors

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.18.41 setIccPsamPPSBaudRate

config BaudRate of ICC chip card or PSAM card

int setIccPsamPPSBaudRate(byte baudRate, CardSlotTypeEnum cardSlotTypeEnum);

Parameters:

Parameter	Description
baudRate	0x11-0x18
cardSlotTypeEnum	CardSlotTypeEnum. ICC1: ICC chip CardSlotTypeEnum. PSAM1: PSAM

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid parameter invalid

SdkResult.Fail other errors

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.18.42 enablePinpadBeep

enable pinpad beep or not during pin entering, beep by default

int enablePinpadBeep(boolean isEnable);

Parameters:

Parameter	Description
isEnable	Enable beep or not; True: enable False:disable

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid parameter invalid

SdkResult.Fail other errors

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.19 Usb Serial

Usb Serial class is responsible for managing POS usb serial port.

Please note, need to use Nexgo special cable.

Get the usb serial class objects:

UsbSerial usbSerial = deviceEngine. getUsbSerial ();

3.19.1 open

Open usb serial port.

public int open(UsbSerialCfgEntity entity, OnUsbSerialReadListener listener);

Parameters:

Parameter	Description
entity	UsbSerialCfgEntity, usb Serial Info
listener	OnUsbSerialReadListener, the reading data will be sent by this listener.

UsbSerialCfgEntity

Attributes	Description
int vid	The verdor id of the usb serial port
int pid	The product id of the usb serial port
int bauRate	The baud rate in the range of (bps): 110,300,600,1200,2400,4800, 9600,14400,56000,19200,38400,57600,115200,230400
int dataBits	Data Bits Range: 5, 6, 7, 8
char parity	Test methods in the range : 'o' odd , 'e' parity, 'n' no parity
int stopBits	Stop bit value range : 1, 2

Return Value:

SdkResult.Success serial connection success

SdkResult.Fail other errors

3.19.2 close

Close the usb serial port.

public int close ();

Parameters: None

Return Value:

SdkResult.Success close successfully

SdkResult.Fail other errors

3.19.3 clrBuffer

Clear the buffer.

public void clrBuffer ();

Parameters: None

Return Value: None

3.19.4 write

Send data.

public int write (byte [] data, int dataLen);

Parameters:

Parameter	Description
data	Write data
dataLen	Data length

Return Value:

SdkResult.Success sent successfully

SdkResult.Fail other errors

3.19.5 write

Send data, application can specify the time of sending data.

public int write (byte [] data, int dataLen, int timeout);

Parameters:

Parameter	Description
data	Write data
dataLen	Data length
timeout	Timeout, ms

Return Value:

SdkResult.Success sent successfully

SdkResult.Fail other errors

3.19.6 read

Receive data.

public int read (byte [] buffer, int readLen);

Parameters:

Parameter	Description
buffer	Buffer to receive data
recvLen	The maximum length of buffer, which is 4096 bytes

Return Value:

Successfully received returns the length of the received data

SdkResult.Fail other errors

3.20 MDB LED

LED class is responsible for managing POS MDB LED lights.

Get the object of the LED class:

MdbLEDDriver ledDriver = deviceEngine. getMdbLEDDriver ();

3.20.1 SetLed

Drive POS red, green, yellow, blue light switch.

void setLed(MdbLightModeEnum light, boolean isOn);

Parameters:

Parameter	Description
light	Card reader light, contact reader, contactless reader, mag-stripe reader lights
isOn	True: on, false: off

MdbLightModeEnum

Enumeration Name	Description
RF_BLUE	Contactless reader - blue light;same function with LEDDriver
RF_YELLOW	Contactless reader - yellow light
RF_GREEN	Contactless reader - green light
RF_RED	Contactless reader - red light
IC_BLUE	Contact reader – blue light
IC_GREEN	Contact reader – green light
IC_RED	Contact reader – red light
MAG_BLUE	Mag-stripe reader – blue light
MAG_GREEN	Mag-stripe reader – green light
MAG_RED	Mag-stripe reader – red light
SYS_GREEN	System – green light
SYS_RED	System – red light

Return Value: None

3.21 MDB Serial

Serial class is responsible for managing MDB serial port for UN20 only.

Get the serial class objects:

MdbSerialPortDriver port = deviceEngine. getMdbSerialPortDriver ();

Note: only support slave mode.

3.21.1 mdbOpen

Open MDB port.

int mdbOpen(MdbModeEnum mdbMode);

Parameters:

Parameter	Description
mdbMode	MdbModeEnum： MDB_MODE_MASTER – master mode MDB_MODE_SLAVE – slave mode(default)

MdbModeEnum

Enumeration Name	Description
MDB_MODE_MASTER	master mode
MDB_MODE_SLAVE	slave mode(default)

Return Value:

SdkResult.Success MDB open success

SdkResult.SerialPort_Connect_Fail MDB open failed

SdkResult.Fail other errors

3.21.2 mdbWrite

Write data

int mdbWrite(byte[] buf);

Parameters:

Parameter	Description
buf	Input data

Return Value:

SdkResult.Success sent successfully

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.SerialPort_Send_Fail serial data transmission failure

SdkResult.Fail other errors

3.21.3 mdbMergerWrite

Merger write data

int mdbMergerWrite(byte[] buf);

Parameters:

Parameter	Description
buf	Input data

Return Value:

SdkResult.Success sent successfully

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.SerialPort_Send_Fail serial data transmission failure

SdkResult.Fail other errors

3.21.4 mdbRead

Receive data.

int mdbRead(byte[] buf);
Note: app need to process the read time.

Parameters:

Parameter	Description
buffer	Buffer to receive data

Return Value:

Successfully received returns the length of the received data

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail other errors

3.21.5 mdbClose

Close MDB port.

int mdbClose();

Return Value:

SdkResult.Success MDB close success

SdkResult. SerialPort_DisConnect_Fail MDB close failed

SdkResult.Fail other errors

3.21.6 mdbClearBuffer

Clear buffer

int mdbClearBuffer();

Return Value:

SdkResult.Success MDB clear buffer success

SdkResult. SerialPort_Other_Error other errors

3.21.7 mdbWrite

Write data

int mdbWrite(MdbModeEnum mdbMode, byte[] buf);

Parameters:

Parameter	Description
mdbMode	MdbModeEnum： MDB_MODE_MASTER – master mode MDB_MODE_SLAVE – slave mode(default)
buf	Input data

Return Value:

SdkResult.Success sent successfully

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.SerialPort_Send_Fail serial data transmission failure

SdkResult.Fail other errors

3.21.8 mdbMergerWrite

Merger write data. If MDB slave mode and matser mode work at same time, please use this API for operation

int mdbMergerWrite(MdbModeEnum mdbMode, byte[] buf);

Parameters:

Parameter	Description
mdbMode	MdbModeEnum： MDB_MODE_MASTER – master mode MDB_MODE_SLAVE – slave mode(default)
buf	Input data

Return Value:

SdkResult.Success sent successfully

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.SerialPort_Send_Fail serial data transmission failure

SdkResult.Fail other errors

3.21.9 mdbRead

Receive data. If MDB slave mode and matser mode work at same time, please use this API for operation

int mdbRead(MdbModeEnum mdbMode, byte[] buf);
Note: app need to process the read time.

Parameters:

Parameter	Description
mdbMode	MdbModeEnum： MDB_MODE_MASTER – master mode MDB_MODE_SLAVE – slave mode(default)
buffer	Buffer to receive data

Return Value:

Successfully received returns the length of the received data

SdkResult.Param_In_Invalid illegal Parameter

SdkResult.Fail other errors

3.21.10 mdbClose

Close MDB port. If MDB slave mode and matser mode work at same time, please use this API for operation

int mdbClose(MdbModeEnum mdbMode);

Parameters:

Parameter	Description
mdbMode	MdbModeEnum： MDB_MODE_MASTER – master mode MDB_MODE_SLAVE – slave mode(default)

Return Value:

SdkResult.Success MDB close success

SdkResult. SerialPort_DisConnect_Fail MDB close failed

SdkResult.Fail other errors

3.21.11 mdbClearBuffer

Clear buffer. If MDB slave mode and matser mode work at same time, please use this API for operation

int mdbClearBuffer(MdbModeEnum mdbMode);

Parameters:

Parameter	Description
mdbMode	MdbModeEnum： MDB_MODE_MASTER – master mode MDB_MODE_SLAVE – slave mode(default)

Return Value:

SdkResult.Success MDB clear buffer success

SdkResult. SerialPort_Other_Error other errors

3.21.12 mdbConfigDeviceType

config device type

int mdbConfigDeviceType(MdbDeviceTypeEnum mdbDeviceType);

Parameters:

Parameter	Description
mdbDeviceType	MdbDeviceTypeEnum： CASHLESS_DEVICE_1, cashless device 1 CASHLESS_DEVICE_2, cashless device 2 COIN_CHANGER, coin changer BILL_VALIDATOR, bill validator

Return Value:

SdkResult.Success, success

SdkResult. SerialPort_Other_Error other errors

3.21.13 mdbSetACKMode

set Poll ACD response mode

int mdbSetACKMode(MdbACKModeEnum mdbACKModeEnum);

Parameters:

Parameter	Description
mdbACKModeEnum	MdbACKModeEnum： ACK_RESPONSE_KERNEL, ACK is responsed by MDB kernel directly ACK_RESPONSE_APP, ACK is responsed by application

Return Value:

SdkResult.Success MDB, success

SdkResult. SerialPort_Other_Error other errors

3.21.14 mdbSetLogLevel

set log level

int mdbSetLogLevel(MdbLogLevelEnum mdbLogLevel);

Parameters:

Parameter	Description
mdbLogLevel	MdbLogLevelEnum： CRITICAL_LOG, critical logs DETAILED_LOG, detailed logs

Return Value:

SdkResult.Success MDB, success

SdkResult. SerialPort_Other_Error other errors

3.21.15 mdbSetCaptureLog

start or stop capture log

int mdbSetCaptureLog(MdbCaptureLogEnum mdbCaptureLog);

Parameters:

Parameter	Description
mdbCaptureLog	MdbCaptureLogEnum： STOP_CAPTURE, stop capture START_CAPTURE, start capture

Return Value:

SdkResult.Success MDB, success

SdkResult. SerialPort_Other_Error other errors

3.21.16 mdbGetLog

get log, the max log buffer is 16Kb

int mdbGetLog(int startPosition, int endPosition, byte[] log);

Parameters:

Parameter	Description
startPosition	Start position, recommend value 0
endPosition	End position.
log	The expect log

Return Value:

>0, the length of the log

SdkResult.Param_In_Invalid; parameter error

SdkResult. SerialPort_Other_Error other errors

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.22 GPIO Drive

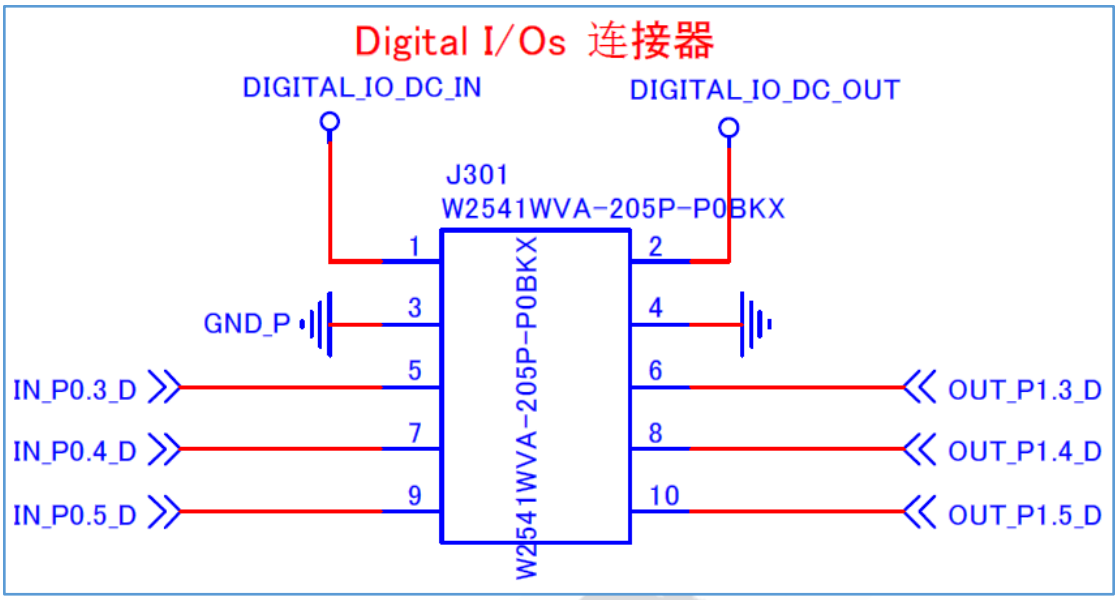
GPIO Drive class is responsible for managing GPIO for UN20 only.

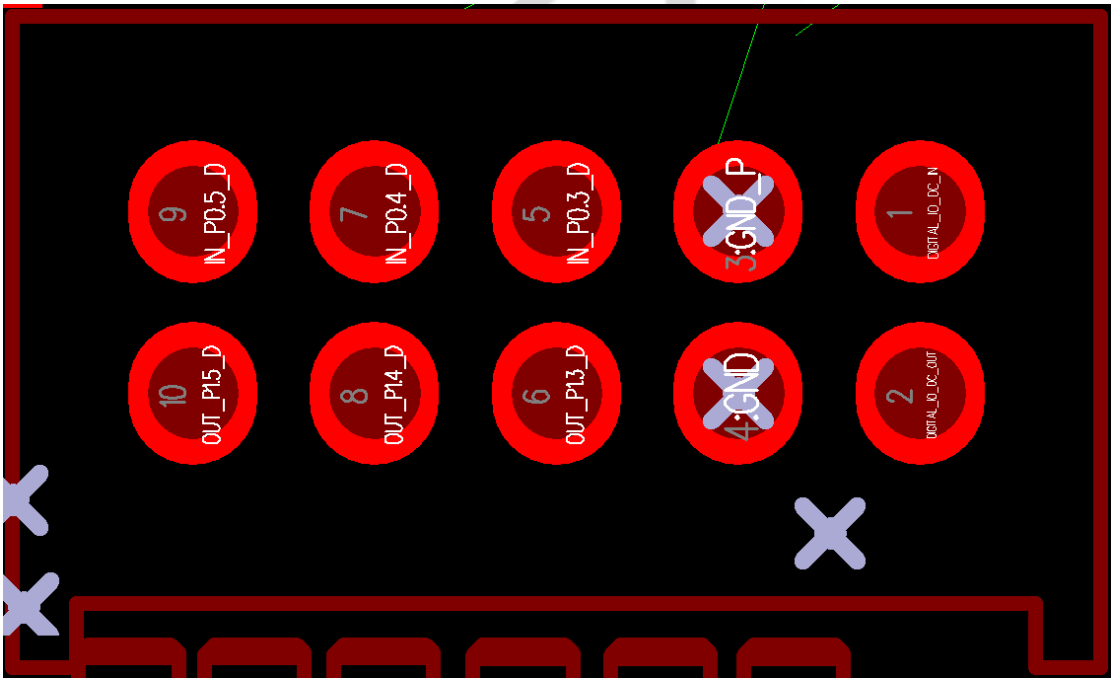
Note:

1. the input IO port is only valid for the input low level
2. the ouput IO port, if call the API to set 1 that will get the low level from ouput IO port
   if call the API set 0 that will get the high resistance state from output IO port

Get the GPIO Drive class objects:

GpioDriver gpio = deviceEngine. getGpioDriver ();





3.22.1 setOutPutGpio

Output GPIO

void setOutPutGpio(GpioInputEnum gpioInputEnum,int value);

Parameters:

Parameter	Description
gpioInputEnum	GPIO list
value	0: get the high resistance state from output IO port 1: low level from output IO port

GpioInputEnum

Enumeration Name	Description
GPIO_INPUT_0_0	Not support
GPIO_INPUT_0_1	Not support
GPIO_INPUT_0_2	Not support
GPIO_INPUT_0_3	OUT_P1.3_D
GPIO_INPUT_0_4	OUT_P1.4_D
GPIO_INPUT_0_5	OUT_P1.5_D
GPIO_INPUT_0_6	OUT_P1.6_D
GPIO_INPUT_0_7	OUT_P1.7_D

Return Value:

Null

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.22.2 getInPutGpio

get input IO

void getInPutGpio(GpioOutputEnum gpioOutputEnum, OnGpioOutPutListener listener);

Parameters:

Parameter	Description
gpioOutputEnum	GPIO list
listener	the callback interface after get input IO is complete

GpioInputEnum

Enumeration Name	Description
GPIO_OUTPUT_1_0	Not support
GPIO_OUTPUT_1_1	Not support
GPIO_OUTPUT_1_2	Not support
GPIO_OUTPUT_1_3	IN_P0.3_D
GPIO_OUTPUT_1_4	IN_P0.4_D
GPIO_OUTPUT_1_5	IN_P0.5_D
GPIO_OUTPUT_1_6	IN_P0.6_D
GPIO_OUTPUT_1_7	IN_P0.7_D

Return Value:

null

Note: This API is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.22.3 openGpio

open GPIO, UN10 need to call open GPIO API before GPIO process.

int openGpio(GpioInputEnum gpioInputEnum);

Parameters:

Parameter	Description
gpioInputEnum	GPIO list

GpioInputEnum

Enumeration Name	Description
GPIO_INPUT_0_0	Not support
GPIO_INPUT_0_1	Not support
GPIO_INPUT_0_2	Not support
GPIO_INPUT_0_3	OUT_P1.3_D
GPIO_INPUT_0_4	OUT_P1.4_D
GPIO_INPUT_0_5	OUT_P1.5_D
GPIO_INPUT_0_6	OUT_P1.6_D

GPIO_INPUT_0_7	OUT_P1.7_D

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid parameter invalid

SdkResult.Fail other errors

3.22.4 closeGpio

close GPIO, UN10 need to call close GPIO API after GPIO process.

int closeGpio(GpioInputEnum gpioInputEnum);

Parameters:

Parameter	Description
gpioInputEnum	GPIO list

GpioInputEnum

Enumeration Name	Description
GPIO_INPUT_0_0	Not support
GPIO_INPUT_0_1	Not support
GPIO_INPUT_0_2	Not support
GPIO_INPUT_0_3	OUT_P1.3_D
GPIO_INPUT_0_4	OUT_P1.4_D
GPIO_INPUT_0_5	OUT_P1.5_D
GPIO_INPUT_0_6	OUT_P1.6_D
GPIO_INPUT_0_7	OUT_P1.7_D

Return Value:

SdkResult.Success success

SdkResult.Param_In_Invalid parameter invalid

SdkResult.Fail other errors

3.23 ExtPinpad

This class is managing operate external pinpad, such as K110. Right now, K110 does not support AES AlgType

3.23.1 initExtPinPad

Initialize the external pinpad.

Public void initExtPinPad (int comNo, SerialCfgEntity serialCfgEntity);

Parameters:

Parameter	Description
comNo	Com number, default value is 0; If N86 with K110, value should be 101
serialCfgEntity	External pinpad configuration, can pass null

SerialCfgEntity

attribute	Description
baudRate	
dataBits	
parity	
stopBits	

Return Value:

None

3.23.2 writeMasterKey

Inject the master key(plaintext key)

Public int writeMasterKey (int mKeyIdx, ExtPinPadKeyAlgTypeEnum keyAlgType, byte [] keyData);

Parameters:

Parameter	Description
mKeyIdx	Master Key Index 0-19
keyAlgType	DES or AES(K110 does not support AES)
keyData	Plaintext master key data

ExtPinPadKeyAlgTypeEnum

Enumeration Name	Description
DES	DES, and TDES
SM4	China SM4
AES	AES

Return Value:

SdkResult.Success success

SdkResult.Param_In_InValid Parameter is not legitimate

SdkResult.PinPad_Dstkey_Idx_Error wrong key index

SdkResult.Fail other errors

3.23.3 writeMasterKey

Inject the master key(cipher key)

Public int writeMKey (int mKeyIdx, ExtPinPadKeyAlgTypeEnum keyAlgType, ExtPinPadCipherModeEnum cipherModeEnum, int decKeyIdx, byte[] iv, byte[] mkCipher);

Parameters:

Parameter	Description
mKeyId	Master Key Index 0-19
keyAlgType	DES, or AES(K110 does not support AES)
cipherModeEnum	ECB, or CBC
decMKeyIdx	Decrypt the master key index 0-19
iv	iv
mkCipher	Ciphertext master key data

ExtPinPadCipherModeEnum

Enumeration Name	Description
ECB	DES, and TDES
CBC	China SM4

Return Value:

SdkResult.Success success

SdkResult.Param_In_InValid Parameter is not legitimate

SdkResult.PinPad_Dstkey_Idx_Error wrong key index

SdkResult.PinPad_No_Key_Error key does not exist

SdkResult.Fail other errors

3.23.4 writeWorkKey

Inject work key.

Public int writeWorkKey (int mKeyIdx, WorkKeyTypeEnum wKeyType, ExtPinPadKeyAlgTypeEnum keyAlgType, ExtPinPadCipherModeEnum cipherModeEnum, byte[] iv, byte [] keyData);

Parameters:

Parameter	Description
mKeyIdx	Master key index number 0-19
wKeyType	Working key type, TAK, TPK,TDK
algMode	DES, or AES(K110 does not support AES)
cipherModeEnum	ECB, CBC
iv	iv
keyData	Working key cipher text data

WorkKeyTypeEnum

Enumeration Name	Description
PINKEY	PIN key
MACKEY	MAC key
TDKEY	Track key
ENCRYPTIONKEY	Data encryption key, providing encryption and decryption

Return Value:

SdkResult.Success success

SdkResult.Param_In_InValid Parameter is invalid

SdkResult.PinPad_Dstkey_Idx_Error wrong key index object; not within the scope

SdkResult.PinPad_Key_Len_Error wrong key length

SdkResult.Fail other errors

3.23.5 calcByWorkKey

public byte[] calcByWKey(int mKeyIdx, WorkKeyTypeEnum wKeyType, ExtPinPadKeyAlgTypeEnum keyAlgType, ExtPinPadCipherModeEnum cipherModeEnum , byte[] iv, byte[] data);
Parameters:

Parameter	Description
mKeyIdx	Master Key Index 0-19
wKeyType	Working key type
algMode	DES, or AES(K110 does not support AES)
cipherModeEnum	ECB, CBC
iv	iv
data	Data to be encrypted

Return:
Null error
Return calculate data

3.23.6 calcMac

Use TAK work key to Calculate MAC.

Public byte [] calcMac (int mKeyIdx, MacAlgorithmModeEnum macAlgMode, ExtPinPadCipherModeEnum cipherModeEnum , byte[] iv, byte [] data);

Parameters:

Parameter	Description
mKeyIdx	Master Key Index 0-19
macAlgMode	MAC algorithm approach
cipherModeEnum	ECB, CBC
iv	iv
data	Input data

MacAlgorithmModeEnum

Enumeration Name	Description
ECB	ECB Algorithm

X99	ANSI X9.9 Encryption Algorithm
X919	ANSI X9.19 Encryption Algorithm
MAC9606	MAC 9606, does not support
AES_CMAC	AES CMAC, does not support

Return Value:

Success returns the computed array

Failure, returning null

3.23.7 calcByMasterKey

Master key encryption.

Public byte [] encryptByMKey (int mKeyId, ExtPinPadKeyAlgTypeEnum keyAlgType,
ExtPinPadCipherModeEnum cipherModeEnum, byte[] iv, byte [] data);

Parameters:

Parameter	Description
mKeyId	Master Key Index 0-19
keyAlgType	DES, or AES(K110 does not support AES)
cipherModeEnum	ECB, CBC
iv	iv
data	Data, lack of an integer multiple of 8, after the meeting 0x00

Return Value:

Success returns the computed array

Failure, returning null

3.23.8 isKeyExist

Check if terminal master key or work key exist or not.

Public boolean isKeyExist(int mKeyIdx, WorkKeyTypeEnum wKeyType);

Parameters:

Parameter	Description
mKeyIdx	Master Key Index 0-19

wKeyType	Null, means check master key exist or not; Not null, check work key exist or not

Return Value:

True, key exist

False, not exist

3.23.9 inputPin

Enter the online PIN on external pinpad.

Public int inputPin
(int mKeyIdx, ExtPinPadKeyAlgTypeEnum algModeEnum, PinAlgorithmModeEnum pinAlg, int minLen, int maxLen, String cardNumber, int timeout, OnExtPinPadInputPinListener listener);

Parameters:

Parameter	Description
mKeyIdx	Master Key Index 0-19
algModeEnum	DES, AES(K110 does not support AES)
pinAlg	Format0
minLen	Min length of the PIN
maxLen	Max length of the PIN
cardNumber	Pan, or Card number
timeout	Enter a timeout in seconds; recommended value 60
listener	Monitor callback interface

Return Value:

SdkResult.Success successful execution listener callback interface

SdkResult.Param_In_InValid illegal Parameter

SdkResult.Fail other errors

3.23.10 inputOfflinePin

Enter the offline PIN(offline plaintext pin, or offline cipher pin) on external pinpad.

Public int inputOfflinePin (int minLen, int maxLen, int timeout, OnPinPadInputListener listener);
Parameters:

Parameter	Description
minLen	Min length of the PIN
maxLen	Max length of the PIN
timeout	Enter a timeout in seconds; recommended value 60
listener	Monitor callback interface

Return Value:

SdkResult.Success successful execution listener callback interface

SdkResult.Param_In_InValid illegal Parameter

SdkResult.Fail other errors

3.23.11 dukptKeyInject

Inject BDK(or IPEK) and KSN for DUKPT.

int dukptInjectKey(int keyIndex, DukptAlgorithmModeEnum algType, DukptAESGenerateKeyTypeEnum g
enerateKeyType, DukptKeyTypeEnum dukptKeyType, byte[] keyData, byte[] ksnData);
Parameters:

Parameter	Description
keyIndex	Key Index 0-19
algType	DES, AES(K110 does not support AES)
generateKeyType	DES/AES(K110 does not support AES)
dukptKeyType	BDK or IPEK
keyData	Key data
ksnData	KSN

DukptKeyTypeEnum

Enumeration Name	Description
BDK	BDK
IPEK	IPEK

DukptAlgorithmModeEnum

Enumeration Name	Description
DES	DES
AES	AES

DukptAESGenerateKeyTypeEnum

Enumeration Name	Description
DUKPT_MODE_KEY_TYPE_2TDEA	Double length DES
DUKPT_MODE_KEY_TYPE_3TDEA	Triple DES
DUKPT_MODE_KEY_TYPE_AES128	AES128
DUKPT_MODE_KEY_TYPE_AES192	AES192
DUKPT_MODE_KEY_TYPE_AES256	AES256

Return value:

SdkResult.Success,

SdkResult.Fail,

SdkResult.PinPad_KeyIdx_Error,

SdkResult.Param_In_Invalid

3.23.12 dukptKsnIncrease

Use it to increase ksn, otherwise the ksn will not change.

Public void dukptKsnIncrease(int mKeyIdx, DukptAlgorithmModeEnum algType);
Parameters:

Parameter	Description
mKeyIdx	Key Index 0-19
algType	DES or AES(K110 does not support AES)

3.23.13 dukptCurrentKsn

Get current Ksn value.

Public byte[] dukptCurrentKsn(int mKeyIdx, DukptAlgorithmModeEnum algType);
Parameters:

Parameter	Description
mKeyIdx	Key Index 0-19
algType	DES or AES(K110 does not support AES)

If key exist, it will return the KSN

Otherwise, it will return null.
This API can be used to check if the DUKPT key exists or not.

3.23.14 dukptEncrypt

Encrypt data in dukpt model.

Public byte[] dukptEncrypt(int keyIndex, DukptAlgorithmModeEnum algType, CipherModeEnum cipherModeEnum, byte[] iv, byte[] data)
Parameters:

Parameter	Description
keyIndex	Key Index 0-19
algType	DES, AES(K110 does not support AES)
cipherModeEnum	Encrypt model,ECB or CBC
iv	Iv, CBC need
data	data

CipherModeEnum

Enumeration Name	Description
ECB	
CBC	

Return value:

Bytes Array,

Null

3.23.15 dukptCalcMac

Use TAK work key to Calculate MAC.

Public byte [] calcMac (int
mKeyIdx , DukptAlgorithmModeEnum algType, DukptAESGenerateKeyTypeEnum keyType, MacAlgorithmModeEnum macAlgMode, byte [] data);

Parameters:

Parameter	Description
mKeyIdx	Master Key Index 0-19

algType	DES or AES(K110 does not support AES)
keyType	
macAlgMode	
data	Input data

DukptAESGenerateKeyTypeEnum

Enumeration Name	Description
DUKPT_MODE_KEY_TYPE_2TDEA	Double length DES
DUKPT_MODE_KEY_TYPE_3TDEA	Triple DES
DUKPT_MODE_KEY_TYPE_AES128	AES128
DUKPT_MODE_KEY_TYPE_AES192	AES192
DUKPT_MODE_KEY_TYPE_AES256	AES256

MacAlgorithmModeEnum

Enumeration Name	Description
ECB	
CBC	
X919	
MAC9606	MAC 9606, does not support
AES_CMAC	AES CMAC, does not support

Return Value:

Success returns the computed array

Failure, returning null

3.23.16 dukptInputPin

Enter the online PIN.

Public int dukptInputPin
(int keyIndex, DukptAlgorithmModeEnum algType, DukptAESGenerateKeyTypeEnum keyType,
PinAlgorithmModeEnum pinFormat, String cardNumber,
int minLen, int maxLen, int timeout, OnExtPinPadInputPinListener listener);

Parameters:

Parameter	Description

keyIndex	Master Key Index 0-19
algType	DES, or AES(K110 does not support AES)
keyType	
pinFormat	Currently only support fomat 0
cardNumber	Pan, or Card number
minLen	Min length of the PIN
maxLen	Max length of the PIN
timeout	Enter a timeout in seconds; recommended value 60
listener	Monitor callback interface

DukptAESGenerateKeyTypeEnum

Enumeration Name	Description
DUKPT_MODE_KEY_TYPE_2TDEA	Double length DES
DUKPT_MODE_KEY_TYPE_3TDEA	Triple DES
DUKPT_MODE_KEY_TYPE_AES128	AES128
DUKPT_MODE_KEY_TYPE_AES192	AES192
DUKPT_MODE_KEY_TYPE_AES256	AES256

Return Value:

SdkResult.Success successful execution listener callback interface

SdkResult.Param_In_InValid illegal Parameter

SdkResult.Fail other errors

3.23.17 getExtPinPadInfo

get external Pinpad versions.

public ExtPinPadInfoEntity getExtPinPadInfo ();

ExtPinPadInfoEntity

attribute	Description
appVersion	
bootVersion	
coreVersion	
hardwareVersion	

3.23.18 extPinPadGetSn

get external Pinpad SN.

Public String extPinPadGetSn();

Return SN of the external pinpad

3.23.19 setLedLight

set pinpad LED

public int setLedLight(boolean blue, boolean yellow, boolean green, boolean red);
Parameters:

Parameter	Description
blue	Turn on Blue led
yellow	Turn on yellow led
green	Turn on green led
red	Turn on red led

Return Value:

SdkResult.Success successful execution listener callback interface

SdkResult.Param_In_InValid illegal Parameter

SdkResult.Fail other errors

3.23.20 beep

External Pinpad beep

public void beep(boolean isSuccessBeep);
Parameters:

Parameter	Description
isSuccessBeep	Success beep or not

Return Value:

None

3.23.21 beep

External Pinpad custom beep

public void beep(int soundID);
Parameters:

Parameter	Description
soundID	The sound ID already exist in External PinPad, such as {33,"beep_ok"}, {34,"beep_err"}

Return Value:

None

3.23.22 setBeepVolume

External Pinpad custom beep volume

public int setBeepVolume(int volume);
Parameters:

Parameter	Description
volume	int value, 0-7

Return Value:

SdkResult.Success successful

SdkResult.Param_In_InValid illegal Parameter

SdkResult.Fail other errors

3.23.23 backToMainScreen

return to idle screen

Public void backToMainScreen ();

Return Value:

None

3.23.24 lcdShowText

external pinpad show text

Public int lcdShowText(int line, int startColumn, boolean isOpenLight, LcdDisplayModeEnum displayModeEnum, LcdAlignEnum alignEnum, byte[] content);

Parameters:

Parameter	Description
line	Start from 1-5, total 5 lines
startColumn	Start Column
isOpenLight	Is the backlight turned on
displayModeEnum	POSITIVE display text or INVERSE
alignEnum	Left , right or center
content	Text to be displayed

LcdDisplayModeEnum

Enumeration Name	Description
POSITIVE	POSITIVE display text
INVERSE	INVERSE display text

LcdAlignEnum

Enumeration Name	Description
LEFT	left
RIGHT	right
CENTER	center

Return Value:

SdkResult.Success successful execution

SdkResult.Param_In_InValid illegal Parameter

SdkResult.Fail other errors

3.23.25 lcdShowImage

external pinpad show image

public int lcdShowImage(int x, int y, int width, int height, String imageName);
Parameters:

Parameter	Description
x	Horizontal coordinate x
y	Vertical coordinate y
width	Image width
height	Image height
imageName	Image name

Return Value:

SdkResult.Success successful execution

SdkResult.Param_In_InValid illegal Parameter

SdkResult.Fail other errors

3.23.26 lcdClean

clean external pinpad screen

Public void lcdClean ();

Return Value:

None

3.23.27 inputAmount

external pinpad input amount

public void inputAmount(int line, int decimalDigit, LcdAlignEnum alignEnum, int timeout, OnExtPinPadInputAmountListener listener);

Parameters:

Parameter	Description
line	Start from 1-5, normally 4 or 5
decimalDigit	Decimal Digit
alignEnum	Left, right or center
timeout	Time out
listener	Listerer

LcdAlignEnum

Enumeration Name	Description
LEFT	left
RIGHT	right
CENTER	center

Return Value:

None

3.23.28 inputText

external pinpad input text

public void inputText (int line, LcdDisplayModeEnum displayModeEnum, LcdAlignEnum alignEnum, int timeout, OnExtPinPadInputTextListener listener);

Parameters:

Parameter	Description
line	Start from 1-5, normally 4 or 5
displayModeEnum	POSITIVE display text or INVERSE
alignEnum	Left, right or center
timeout	Time out
listener	OnExtPinPadInputTextListener

LcdDisplayModeEnum

Enumeration Name	Description
POSITIVE	POSITIVE display text
INVERSE	INVERSE display text

LcdAlignEnum

Enumeration Name	Description
LEFT	left
RIGHT	right
CENTER	center

Return Value:

None

3.23.29 extPinPadKeyEcho

external pinpad get press key value, it is used for item selected in external pinpad.

public void extPinPadKeyEcho(byte[] keyCodes, int timeout, OnExtPinPadKeyEchoListener listener);
Parameters:

Parameter	Description
byte[] keyCodes	Expected key value, refer to ExtPinPadKeyCode
timeout	Time out
listener	Listerer

ExtPinPadKeyCode:

	Description
KEYCODE_0 = 0x00	
KEYCODE_1 = 0x01	
KEYCODE_2 = 0x02	
KEYCODE_3 = 0x03	
KEYCODE_4 = 0x04	
KEYCODE_5 = 0x05	
KEYCODE_6 = 0x06	
KEYCODE_7 = 0x07	
KEYCODE_8 = 0x08	
KEYCODE_9 = 0x09	
KEYCODE_OK = 0x0a	
KEYCODE_CANCEL = 0x0b	
KEYCODE_CLEAR = 0x0c	

KEYCODE_PAGE_UP = 0x0d	
KEYCODE_PAGE_DOWN = 0x0e	
KEYCODE_F1 = 0x0f	
KEYCODE_F2 = 0x10	
KEYCODE_F3 = 0x11	

Return Value:

None

3.24 CardEmulationHandler

This class is managing Ndef Tag, Simulate POS into a card and write specific data into the card for mobile phones or other NFC card readers to read

Note: Only new models support Ndef Tag, such as N96, N92, N86PRO, N6PRO, N62.

Please check with NEXGO ADS(ads_support@xgd.com) team to confirm the firmware version for support Ndef Tag.

Note: All CardEmulationHandler APIs is related to the device model and firmware version. If you encounter any issues while using this API, please contact NEXGO ADS team(ads_support@xgd.com)

3.24.1 getNfcDataWithUri

generate NFC data containing URI

byte[] getNfcDataWithUri(NfcTagTypeEnum tagType, int uriProtocols, String uri);

Parameters:

Parameter	Description
tagType	Currently only support T2T
uriProtocols	URI protocol, range from 0-36
uri	URL

NfcTagTypeEnum

Enumeration Name	Description
T2T	T2T
T4T	T4T

Return Value:

Null, generate failed

byte[], generate NFC data which contains URI

3.24.2 getNfcDataWithAmt

generate NFC data containing amount

byte[] getNfcDataWithAmt(NfcTagTypeEnum tagType, String amt, String languageCodeAmt, String activityFullName, String activityData, String packageName);

Parameters:

Parameter	Description
tagType	Currently only support T2T
amt	Amount, for example “1112”
languageCodeAmt	Amount languae code, for example "0000,en,zh..."
activityFullName	Activity full name, for example “com.xgd.nfcUserMessage.nfcActivity"
activityData	Activity data, for example “com.xgd.nfcUserMessage.nfcActivity"
packageName	Package Name, for example "com.nexgo.tapbyphone"

NfcTagTypeEnum

Enumeration Name	Description
T2T	T2T
T4T	T4T

Return Value:

Null, generate failed

byte[], generate NFC data which contains amount

3.24.3 getNfcDataWithText

generate NFC data containing text

byte[] getNfcDataWithText(NfcTagTypeEnum tagType, String languageCode, String text);

Parameters:

Parameter	Description
tagType	Currently only support T2T
languageCodeAmt	Languae code, for example "0000,en,zh..."
text	User text

NfcTagTypeEnum

Enumeration Name	Description
T2T	T2T
T4T	T4T

Return Value:

Null, generate failed

byte[], generate NFC data which contains text

3.24.4 getNfcDataWithUserMessage

generate NFC data containing user message

byte[] getNfcDataWithUserMessage(NfcTagTypeEnum tagType, NdefUserMessageRecord ndefUserMessageRecord);

Parameters:

Parameter	Description
tagType	Currently only support T2T
ndefUserMessageRecord	NdefUserMessageRecord

NfcTagTypeEnum

Enumeration Name	Description
T2T	T2T
T4T	T4T

NdefUserMessageRecord

Enumeration Name	Description
textArray	NdefTextRtd[], Optional
uriArray	NdefUriRtd[], Optional
aarArray	NdefAarRtd[], Optional
customArray	NdefCustomRecord[], Optional

NdefTextRtd

Enumeration Name	Description
text	String, Text content
languageCode	String, Text language code, such as “0000,en,zh...”
encode	Int, Encode type, 0:UTF-8, 1:UTF-16

NdefUriRtd

Enumeration Name	Description
uri	String, URL
uiProtocol	int, URI protocol, range from 0-36

NdefAarRtd

Enumeration Name	Description
packageName	String, APP packagename

NdefCustomRecord

Enumeration Name	Description
tnf	int, Payload type
type	byte[], payload type name
id	byte[], payload ID
payLoad	byte[], payload content

Return Value:

Null, generate failed

byte[], generate NFC data which contains user message

3.24.5 openNfc

Write NFC data and Open NFC

int openNfc(NfcTagTypeEnum tagType, byte[] nfcData, int timeout, OnCardEmulationListener listener);

Parameters:

Parameter	Description
tagType	only support T2T
nfcData	NFC data which is generated by API: getNfcDataWithAmt GetNfcDataWithText GetNfcDataWithUri
timeout	Time out, for example 60s
listener	listener

NfcTagTypeEnum

Enumeration Name	Description
T2T	T2T
T4T	T4T

Return Value:

SdkResult.Success, success

SdkResult.Param_In_Invalid, parameter incorrect

3.24.6 closeNfc

Close NFC

void closeNfc();

Return Value:

None

3.24.7 setUid

Set custom uid.

int setUid(NfcTagTypeEnum tagType, byte[] uid);

Parameters:

Parameter	Description
tagType	NfcTagTypeEnum, Currently only support T2T
uid	byte[], fixed length 4 bytes

NfcTagTypeEnum

Enumeration Name	Description
T2T	T2T
T4T	T4T

Return Value:

Null, generate failed

byte[], generate NFC data which contains URI

3.25 ExtendLcd

This class is managing extend LCD, the extend lcd size maybe different for different model. For now, N86 extend lcd size is 192*32, N92 extend lcd size is 320*240.

Please note:

N86 with second display can work with blow API;

N92 with second display can work with blow API; But also support android native API(Presentation). We suggest to use android native API for developing the second display function.

N86 Pro with second display can work with blow API; But also support android native API(Presentation). We suggest to use android native API for developing the second display function.

3.25.1 open

open extend LCD

int open();

Return Value:

SdkResult.Success, success

SdkResult. Fail, failed

3.25.2 close

close extend LCD

int close();

Return Value:

SdkResult.Success, success

SdkResult. Fail, failed

3.25.3 showText

show text in specify area on Extend LCD

int showText(int x, int y, String strText);

Parameters:

Parameter	Description

x	x-coordinate, N86 extend LCD size is 192*32, x value range from 0-191 N92 extend LCD size is 320*240, x value range from 0-319
y	y-coordinate, N86 range from 0-15 N92 range from 0-239
strText	String text which will show in extend LCD

Return Value:

SdkResult.Success, success

SdkResult. Fail, failed

3.25.4 brushScreen

brush the extend LCD and show the cache data on the extend LCD

int brushScreen();

Return Value:

SdkResult.Success, success

SdkResult. Fail, failed

3.25.5 showBmp

show bitmap on the extend LCD with path

int showBmp(String path);

Parameters:

Parameter	Description
path	The path of the bitmap, format: *RGB888 24bit 192*32 BMP for N86*

Return Value:

SdkResult.Success, success

SdkResult. Fail, failed

3.25.6 showBmp

show bitmap on the extend LCD with bitmap

int showBmp(Bitmap bitmap);

Parameters:

Parameter	Description
bitmap	Bitmap, do not scaled. Below is example code for N86 Pro: BitmapFactory.Options options = new BitmapFactory.Options(); options.inScaled = false; Bitmap bitmap = BitmapFactory.decodeResource(getResources(), R.drawable.bmp284x76, options);

Return Value:

SdkResult.Success, success

SdkResult. Fail, failed

3.25.7 clearScreen

clear screen

int clearScreen();

Return Value:

SdkResult.Success, success

SdkResult. Fail, failed

3.25.8 setFontSize

set font size

int setFontSize(ExtendLcdFontSizeEnum fontSizeEnum);

Parameters:

Parameter	Description
fontSizeEnum	Font size, see ExtendLcdFontSizeEnum

ExtendLcdFontSizeEnum

Enumeration Name	Description
FONT_ASC8X16	
FONT_ASC12X24	
FONT_ASC16X24	
FONT_ASC16X32	
FONT_ASC16X48	Not support
FONT_BOLD_ASC8X16	
FONT_BOLD_ASC12X24	
FONT_BOLD_ASC16X24	
FONT_BOLD_ASC16X32	
FONT_ASC12X16	
FONT_BOLD_ASC12X16	
FONT_ASC20X40	
FONT_BOLD_ASC20X40	
FONT_ASC24X48	
FONT_BOLD_ASC24X48	
FONT_ASC28X56	
FONT_BOLD_ASC28X56	
FONT_ASC32X64	
FONT_BOLD_ASC32X64	
FONT_ASC36X72	
FONT_BOLD_ASC36X72	

Note:

a. Current FONT_ASC16X48 does support for N86 and N86 Pro with extend LCD.
b. N86 with extend LCD support from FONT_ASC8X16 to FONT_BOLD_ASC12X16
c. N86 Pro with extend LCD support from FONT_ASC8X16 to FONT_BOLD_ASC36X72

Return Value:

SdkResult.Success, success

SdkResult. Fail, failed

3.25.9 getFontSize

get font size

ExtendLcdFontSizeEnum getFontSize();

ExtendLcdFontSizeEnum

Enumeration Name	Description
FONT_ASC8X16	
FONT_ASC12X24	
FONT_ASC16X24	
FONT_ASC16X32	
FONT_ASC16X48	Not support
FONT_BOLD_ASC8X16	
FONT_BOLD_ASC12X24	
FONT_BOLD_ASC16X24	
FONT_BOLD_ASC16X32	
FONT_ASC12X16	
FONT_BOLD_ASC12X16	

Return Value:

ExtendLcdFontSizeEnum

3.25.10 setDisplayMode

set display mode on extend LCD

int setDisplayMode(ExtendLcdDisplayModeEnum displayModeEnum, int timeout);

Parameters:

Parameter	Description
displayModeEnum	See ExtendLcdDisplayModeEnum,
timeout	If displayModeEnum == NORMAL, Timeout corresponds to the timeout period of the customer display screen operation, and the screen will automatically turn off If displayModeEnum == AD, Timeout corresponds to the timeout of the customer display screen operation and entering the advertising carousel state

ExtendLcdDisplayModeEnum

Enumeration Name	Description
NORMAL	Default mode
AD	Advertising screen mode

Return Value:

SdkResult.Success, success

SdkResult. Fail, failed

3.25.11 adSetWorkingMode

Set up advertising rotation mode, which corresponds to different advertising rotation mechanisms for different functions and applications in different scenarios

int adSetWorkingMode(ExtendLcdAdWorkingModeEnum workingModeEnum);

Parameters:

Parameter	Description
workingModeEnum	See ExtendLcdAdWorkingModeEnum

ExtendLcdAdWorkingModeEnum

Enumeration Name	Description
AD_DEFAULT	Default mode, synchronized on/off
AD_AUTO	Insert external power and keep it constantly on
AD_FORCE	The advertising screensaver remains on continuously

Return Value:

SdkResult.Success, success

SdkResult. Fail, failed

3.25.12 adAddBmp

Add carousel advertisement images,; If the API is called multiple times to set the same BMP, the final display time set shall prevail. It will not be added multiple times, but will update the display time.

int adAddBmp(int second, String path);

Parameters:

Parameter	Description
second	Show image time, for example 10s
path	The path of the image, format: 320X240 RGB565 RGB888

Return Value:

SdkResult.Success, success

SdkResult. Fail, failed

3.25.13 adDeleteBmp

delete carousel advertisement images

int adDeleteBmp(String path);

Parameters:

Parameter	Description
path	The path of the image, format: 320X240 RGB565 RGB888

Return Value:

SdkResult.Success, success

SdkResult. Fail, failed

3.25.14 adClearList

delete all carousel advertisement images

int adClearList();

Return Value:

SdkResult.Success, success

SdkResult. Fail, failed

3.25.15 adGetListMaxSize

get the max number of carousel advertisement images

int adGetListMaxSize();

Return Value:

>0, the max number of the carousel advertisement images

SdkResult. Fail, failed

3.25.16 isExtendLcdExist

check extend LCD exist or not.

boolean isExtendLcdExist();

Return Value:

true, exist

false, not exist

3.25.17 getExtendLcdSize

get extend LCD screen size

boolean int[] getExtendLcdSize();

Return Value:

Screen size.

N86 with extend LCD: 192, 32

N86 Pro with extend LCD: 284, 76

N92 with extend LCD: 320, 240

3.26 MifareSamCardHandler

The MifareSamCard module class is responsible for managing operate theMifare SAM card.

MifareSamCardHandler cardHandler = deviceEngine.getMifareSamCardHandler();

3.26.1 authHostAV2

Mifare SAM Authenticate Host Function.

MifareSamCardEntity authHostAV2(PSAMCardSlotEnum slotEnum,byte[] samAuthKey,byte keyNo,byte keyVer,byte hostMode);

Parameters:

Parameter	Description
slotEnum	SAM slot
samAuthKey	SAM key
keyNo	SAM key number
keyVer	SAM key version
hostMode	SAM Protection Mode 0x00:Plain; 0x01:MAC Protection; 0x02:Full Protection

PSAMCardSlotEnum

Enumeration Name	Description
PSAM1	PSAM card slot 1
PSAM2	PSAM card slot 2
PSAM3	PSAM card slot 3
PSAM4	PSAM card slot 4

MifareSamCardEntity

Parameter	Description
result	SdkResult.Success, others failed
sessionKeyForEncryption	Session key for encryption
sessionKeyForMac	Session key for mac

Return Value:

MifareSamCardEntity, when this interface excute successful;

NULL, when this interface excute failed.

3.27 M1PlusCardHandler

The Ultralight module class is responsible for managing operate the Mifare Ultralight card.

UltralightCCardHandler ultralightCCardHandler = deviceEngine.getUltralightCCardHandler();

3.27.1 active

active the mifare card

boolean active();

Return Value:

True, active successfully

False, active failed

3.27.2 MPInit

Init mifare plus card.

int MPInit(int protectStatus);

Parameters:

Parameter	Description
protectStatus	Valid values: 3, PICC in ISO 14443-3 4, PICC in ISO 14443-4

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.27.3 MPGetVersion

Get Mifare Plus card version

byte[] MPGetVersion();

Return Value:

Card version

3.27.4 MPGetOriginalitySig

Get Mifare Plus card original signature data

byte[] MPGetOriginalitySig();

Return Value:

Mifare Plus card signature data

3.27.5 MPAuthFirst

Mifare plus card first authentication

int MPAuthFirst(short keyBN, byte[]authKey, int smMode);

Parameters:

Parameter	Description
keyBN	Block index
authKey	Authentication key, AES128, key length 16 bytes
smMode	Secure mode, 0-EV0, 1-EV1

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.27.6 MPAuthNonFirst

Mifare plus card non-first authentication

int MPAuthNonFirst(short keyBN, byte[] authKey);

Parameters:

Parameter	Description
keyBN	Block index
authKey	Authentication key, AES128, key length 16 bytes

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.27.7 MPResetAuth

Get Mifare Plus card reset authentication

int MPResetAuth();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.27.8 MPWritePersonal

Mifare plus card write private data

int MPWritePersonal(short keyBN, byte[] data);

Parameters:

Parameter	Description
keyBN	Block index
data	data

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.27.9 MPCommitPersonal

Mifare plus card commit private data, and set secure level

int MPCommitPersonal(int securityLevel);

Parameters:

Parameter	Description
securityLevel	1-SL1, 3-SL3

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.27.10 MPRead

Mifare plus card read block data

byte[] MPRead(short BNr, int blockNum, int cmdNoMac, int plain, int respMac);

Parameters:

Parameter	Description
BNr	Block index
blockNum	How many blocks you want to read
cmdNoMac	POS request data with MAC or not 0: With MAC 1: Without MAC
plain	Data transfer is plaintext or ciphertext. 0: Ciphertext 1: plaintext
respMac	Card response data with MAC or not 0: without MAC 1: with MAC

Return Value:

Data, length is 16* blockNum

3.27.11 MPWrite

Mifare plus card read block data

Int MPWrite(short BNr, int blockNum, int plain, int respMac, byte[] pData);

Parameters:

Parameter	Description
BNr	Block index
blockNum	How many blocks you want to write
plain	Data transfer is plaintext or ciphertext. 0: Ciphertext 1: plaintext
respMac	Card response data with MAC or not 0: without MAC 1: with MAC
pData	Data to be write

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.27.12 MPIncValue

Mifare plus card increate value

int MPIncValue(short BNr, int value, int transfer, int respMac);

Parameters:

Parameter	Description
BNr	Block index
value	Block value you want to increase
transfer	Store data immediately or not 0: store immediately 1: not store
respMac	Card response data with MAC or not 0: without MAC 1: with MAC

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.27.13 MPDecValue

Mifare plus card decrease value

int MPDecValue(short BNr, int value, int transfer, int respMac);

Parameters:

Parameter	Description
BNr	Block index
value	Block value you want to decrease
transfer	Store data immediately or not 0: store immediately 1: not store
respMac	Card response data with MAC or not

	0: without MAC 1: with MAC

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.27.14 MPTransfer

Mifare plus card data transer store

int MPTransfer(short BNr, int respMac);

Parameters:

Parameter	Description
BNr	Block index
respMac	Card response data with MAC or not 0: without MAC 1: with MAC

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.27.15 MPWriteValue

Mifare plus card write value

int MPWriteValue(short BNr, int value);

Parameters:

Parameter	Description
BNr	Block index
value	Write value into block

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.27.16 MPReadValue

Mifare plus card read value

int MPReadValue(short BNr);

Parameters:

Parameter	Description
BNr	Block index

Return Value:

Block value

3.27.17 MPVcSupportLast

Mifare plus config virtual card support options

int MPVcSupportLast(byte[] pIid, byte[] pRndq, char lenCap, byte[] pOut, byte[] VCMACKey, byte[] VCEncKey);

Parameters:

Parameter	Description
pIid	IID: Installation Identifier
pRndq	Random Number of the PCD
lenCap	set 00h
pOut	return the virtual card info
VCMACKey	The Mac Key of Virtual Card
VCEncKey	The Encrypted Key of Virtual Card

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

3.27.18 MPDeselect

Mifare plus cancel selection

int MPDeselect();

Return Value:

SdkResult.Success success

SdkResult.Fail other errors

4 Callback information

4.1 OnPrintListener

Responsible for managing the printer class callback interface.

4.1.1 onPrintResult

After executing the startPrint method, callback to print results.

public void onPrintResult (int retCode);

Parameters:

Parameter	Description
retCode	Print callback Results : SdkResult.Success success SdkResult.Printer_Print_Fail failed to print SdkResult.Printer_PaperLack out of paper SdkResult.Printer_UnFinished print unfinished SdkResult.Printer_TooHot printer is overheating

Return Value: None

4.2 OnPinPadInputListener

PIN pad class is responsible for the management callback interface.

4.2.1 onInputResult

After inputOnlinePin or inputOffinePin method is executed, it callback to show pin result.

public void onInputResult (int retCode, byte [] data);

Parameters:

Parameter	Description
retCode	Enter the result : SdkResult.Success success SdkResult.Fail failure SdkResult.PinPad_Input_Timeout input timeout SdkResult.PinPad_Input_Cancel cancel input SdkResulr.PinPad_No_Pin_Input no password entered, press Enter directly
data	When retCode == SdkResult.Success return Pinblock for OnlinePin. Offline Pin return 00 Otherwise return null

Return Value: None

4.2.2 onSendKey

After inputText, inputOnlinePin, inputOfflinePin method, this is executed, callback the key input, when input password, number key will return KEYCODE_STAR.

public void onSendKey (byte keyCode);

Parameters:

Parameter	Description
keyCode	Input key value

Key-Value

Constant Name	Constant Value	Description
KEYCODE_0	0x30 (byte)	0
KEYCODE_1	0x31 (byte)	1
KEYCODE_2	0x32 (byte)	2
KEYCODE_3	0x33 (byte)	3
KEYCODE_4	0x34 (byte)	4
KEYCODE_5	0x35 (byte)	5
KEYCODE_6	0x36 (byte)	6
KEYCODE_7	0x37 (byte)	7
KEYCODE_8	0x38 (byte)	8
KEYCODE_9	0x39 (byte)	9
KEYCODE_ the STAR	0x2a (byte)	*
KEYCODE_OCTOTHORPE	0x23 (byte)	#
KEYCODE_CANCEL	0x18 (byte)	Cancel key
KEYCODE_BACKSPACE	0x08 (byte)	Backspace

KEYCODE_CLEAR	0xfe (byte)	Clear key
KEYCODE_CONFIRM	0x0d (byte)	Enter

Return Value: None

4.3 OnScanner Listener

Responsible for managing the camera scan code results callback.

4.3.1 onInitResult

Initialize the camera configuration callback.

public void onInitResult (int retCode);

Parameters:

Parameter	Description
retCode	Enter the result: SdkResult.Success success SdkResult.Fail failure

Return Value: None

4.3.2 onScannerResult

Scan code results callback.

public void onScannerResult (int retCode, String data);

Parameters:

Parameter	Description
retCode	Enter the result : SdkResult.Success success SdkResult.Fail failure SdkResult.Param_In_Invalid Parameter error SdkResult. TimeOut scan code timeout SdkResult. Scanner_Customer_Exit voluntary user withdrawal
data	Scan code Results

Return Value: None

4.4 OnCardInfoListener

Reader class is responsible for managing the callback interface.

4.4.1 onCardInfo

After executing the searchCard method, callback reader results.

public void onCardInfo (int retCode, CardInfoEntity cardInfo);

Parameters:

Parameter	Description
retCode	Enter the result : SdkResult.Success success SdkResult.Fail failure SdkResult.TimeOut timeout
cardInfo	When retCode == SdkResult.Success return card information Otherwise return null

CardInfoEntity

Attributes	Description
String cardNo	Card number; if device P2PE mode, return cipher text
CardSlotTypeEnum cardExistslot	CardSlotType
RfCardTypeEnum rfCardType	RfCardTyp
String tk1	track 1; if device is P2PE mode, return cipher text
String tk2	tracks 2; if device is P2PE mode, return cipher text
String tk3	tracks 3; if device is P2PE mode, return cipher text
String expiredDate	Card expired date; if device is P2PE mode, return cipher text
String serviceCode	Service Code
boolean isTk1Valid	A track LRC is correct
boolean isTk2Valid	Two tracks LRC is correct
boolean isTk3Valid	Three tracks LRC is correct
boolean isICC	If mag card has chip flag
String csn	Card serial number, only returnd in OnEmvProcessListener.onConfirmCardNo
String maskCardNo	Masked card number, for example 456789******1234 Apply to device P2PE mode
boolean isInCardBin	Card is in whitelist Apply to device P2PE mode

boolean isAppInSredWhiteList	Application is in SRED whitelist Apply to device P2PE mode

CardSlotTypeEnum

Enumeration Name	Description
ICC1	Default IC card slot(Contact card solt)
ICC2	Unavailable
ICC3	Unavailable
PSAM1	PSAM slot 1
PSAM2	PSAM slot 2
PSAM3	Unavailable (only available for UN20)
PSAM4	Unavailable (only available for UN20)
RF	Conactless card slot
SWIPE	Magnetic stripe card slot

RfCardTypeEnum

Enumeration Name	Description
TYPE_A_CPU	
TYPE_B_CPU	
S50	
FELICA	
S70	
ULTRALIGHT	
MEMORY_OTHER	
S50_PRO	
S70_PRO	

Return Value: None

4.4.2 onSwipeIncorrect

After executing searchCard method, will be callback when a swipe error occurs. This callback is a process callback, not a result callback

```
public void onSwipeIncorrect();
```

Return Value: None

4.4.3 onMultipleCards

After executing searchCard method，will be callback when find multiple contactless cards. This callback is a process callback, not a result callback

public void onMultipleCards();

Return Value: None

4.5 OnEMVProcessListener2

Responsible for managing the EMV class callback interface.

4.5.1 onSelApp

After EmvProcess executed, if card have multi-application, onSelApp callback will be executed .it will show app-list to let the user to select the application. Then call EmvHandler2.onSetSelAppResponse.

public void onSelApp (List <String> appNameList, List <CandidateAppInfoEntity> appInfoList, boolean isFirstSelect);

Parameters:

Parameter	Description
appNameList	Application displays a list of names
appInfoList	Candidate application information card
isFirstSelect	Whether making the selection for the first time

CandidateAppInfoEntity

Attributes	Description
byte [] aid	AID
byte [] appLabel	Apply the label
byte [] preferName	Application Preferred Name
byte priority	Application Priority Indicator
byte [] langPrefer	The preferred language
byte icti	Issuer Code Table Index
byte[] rfu	extro EMV tags list, such as 42, 5F55 during the multi application selection list(onSelApp) for EMV contact. Application can select specify application by these tags.

Return Value: None

4.5.2 onTransInitBeforeGPO

After EmvProcess executed, before excute GPO, the EMV will callback this method. User can call setTlv method to set personalized tags. User can set it or not.(This is suit for both EMV contact and contactless flow), then call EmvHandler2. onSetTransInitBeforeGPOResponse.

public void onTransInitBeforeGPO ();

Return Value: None

4.5.3 onConfirmCardNo

After EmvProcess executed, confirm the card number, then call EmvHandler2.onSetConfirmCardNoResponse.

public void onConfirmCardNo (String cardNo);

Parameters:

Parameter	Description
cardNo	card number; if device P2PE mode, return ciphertext

Return Value: None

4.5.4 onCardHolderInputPin

EmvProcess executing the method, enter the password, to be called EmvHandler2.onSetPinInputResponse.

public void onCardHolderInputPin (boolean isOnlinePin, int leftTimes);

Parameters:

Parameter	Description
isOnlinePin	Is online password
leftTimes	Enter the remaining number of times for the offline PIN

Return Value: None

4.5.5 onContactlessTapCardAgain

EmvProcess executing the method, Callback the second read card .(When host response the script with contactless EMV transaction.) The application should re-search contactless card, then call

EmvHandler2. onSetContactlessTapCardResponse to notify EMV continue procress.

public void onContactlessTapCardAgain ();

Return Value: None

Note: for amex contactless, the method will called by kernel, when process case “please see phone”

4.5.6 onOnlineProc

EmvProcess executing the method, means EMV kernel request online process, Then the application should call method getTlv to get the EMV tags, then send request message to the host. After host response, the application should call EmvHandler2.onSetOnlineProcResponse to notify the EMV kernel to do the second auth.

public void onOnlineProc ();

Parameters: None

Return Value: None

4.5.7 onPrompt

EmvProcess executing the method, notify the application prompt information to the user, Then the application should call EmvHandler2. onPromptResponse to notify the EMV kernel to continue the flow.

public void onPrompt(PromptEnum prompt);

Parameters:

Parameters	Description
prompt	enum

PromptEnum

enum	Description
APP_SELECTION_IS_NOT_ACCEPTED	Application is not accepted, please try again
OFFLINE_PIN_INCORRECT_TRY_AGAIN	Offline pin incorrect, please try again
OFFLINE_PIN_INCORRECT	Offline pin incorrect
OFFLINE_PIN_CORRECT	Offline pin correct

Return Value: None

4.5.8 onRemoveCard

EmvProcess executing the method, notify the application contactless card can be remove from the card reader, Then call EmvHandler2. onSetRemoveCardResponse to notify the EMV kernel to continue the flow.

public void onRemoveCard ();

Parameters: None

Return Value: None

4.5.9 onFinish

EmvProcess executing the method, means all the EMV flow is finish. The retcode will indicate the EMV transaction result.

public void onFinish (int retCode, EmvProcessResultEntity entity);

Parameters:

Parameter	Description
retCode	Enter the result : SdkResult.Success success SdkResult.Fail failure Other return results please refer to the EMV class table 3.7 or consult Appendix 5
entity	When retCode == SdkResult.Success return data Otherwise return null

EmvProcessResultEntity

Attributes	Description
byte [] scriptResult	Script execution results
List <EMVCardLogEntity> EMVlog	Cards Blog List
byte [] ecBalance	Electronic cash balance

EmvCardLogEntity

Attributes	Description
boolean isAmtExist	Whether the amount of presence
String amt	Amount of money
boolean isOtherAmtExist	Whether the existence of other

String otherAmt	Other Amount
boolean isDateExist	Date of the transaction if there are
String transDate	transaction date
boolean isTimeExist	The existence of transactions
String transTime	transaction hour
boolean isCntCodeExist	Whether there is a country code
String cntCode	country code
boolean isCurExist	Currency code if there
String curCode	Currency code
boolean isAtcExist	The existence of the transaction counter
String atc	Transaction Counter
boolean is9CExist	9c transaction type whether there
String serveType	Transaction Type
boolean isMerNameExist	The existence of a business name
String merName	Business Name

Return Value: None

4.6 OnEMVProcessListener Deprecated

Please note: All the Emvhandler method, do not recommend use it anymore.

Responsible for managing the EMV class callback interface.

4.6.1 onSelApp

After EmvProcess executed, callback app-list to select the application, then call EmvHandler.onSetSelAppResponse.

public void onSelApp (List <String> appNameList, List <CandidateAppInfoEntity> appInfoList, boolean isFirstSelect);

Parameters:

Parameter	Description
appNameList	Application displays a list of names
appInfoList	Candidate application information card
isFirstSelect	Whether making the selection for the first time

CandidateAppInfoEntity

Attributes	Description
byte [] aid	AID
byte [] appLabel	Apply the label
byte [] preferName	Application Preferred Name
byte priority	Application Priority Indicator
byte [] langPrefer	The preferred language
byte icti	Issuer Code Table Index

Return Value: None

4.6.2 onAfterFinalSelectedApp

After EmvProcess executed, before excute GPO, the EMV will callback this method. User can call setTlv method to set personalized tags. User can set it or not.(This is suit for EMV contactless flow), then call EmvHandler. onSetAfterFinalSelectedAppResponse.

public void onAfterFinalSelectedApp ();

Return Value: None

4.6.3 onRequestAmount

After EmvProcess executed, callback request input amount (triggered when when the transaction amount has not been input), then call EmvHandler.onSetRequestAmountResponse.

public void onRequestAmount ();

Return Value: None

4.6.4 onConfirmEcSwitch

After EmvProcess method executed, whether use electronic cash (triggered when the transaction is set to support electronic cash, and the card also support e-cash), then call EmvHandler.onSetConfirmEcSwitchResponse.

public void onConfirmEcSwitch ();

Return Value: None

4.6.5 onConfirmCardNo

After EmvProcess executed, confirm the card number, then call EmvHandler.onSetConfirmCardNoResponse.

public void onConfirmCardNo (String cardNo);

Parameters:

Parameter	Description
cardNo	card number

Return Value: None

4.6.6 onCardHolderInputPin

EmvProcess executing the method, enter the password, to be called EmvHandler.onSetPinInputResponse.

public void onCardHolderInputPin (boolean isOnlinePin, int leftTimes);

Parameters:

Parameter	Description
isOnlinePin	Is online password
leftTimes	Enter the remaining number of times for the offline PIN

Return Value: None

4.6.7 onCertVerify

EmvProcess executing the method of confirming documents, later to be called EmvHandler.onSetCertVerifyResponse.

public void onCertVerify (String certName, String certInfo);

Parameters:

Parameter	Description
certName	the name of your ID
certInfo	identity information

Return Value: None

4.6.8 onReadCardAgain

EmvProcess executing the method, Callback the second read card .(When host response the script with contactless EMV transaction.) The application should re-search contactless card, then call

EmvHandler.onSetReadCardAgainResponse to notify EMV continue procress.

public void onReadCardAgain();

Return Value: None

Note: for amex contactless, the method will called by kernel, when process case “please see phone”

4.6.9 onOnlineProc

EmvProcess executing the method, means EMV kernel request online process, Then the application should call method getTlv to get the tags, then send request message to the host. After host response, the application should call EmvHandler.onSetOnlineProcResponse to notify the EMV kernel to do the second auth.

public void onOnlineProc ();

Parameters: None

Return Value: None

4.6.10 onPrompt

EmvProcess executing the method, notify the application prompt information to the user, Then the application should call EmvHandler. onPromptResponse to notify the EMV kernel to continue the flow.

public void onPrompt(PromptEnum prompt);

Parameters:

Parameters	Description
prompt	enum

PromptEnum

enum	Description
APP_SELECTION_IS_NOT_ACCEPTED	Application is not accepted, please try again
OFFLINE_PIN_INCORRECT_TRY_AGAIN	Offline pin incorrect, please try again
OFFLINE_PIN_INCORRECT	Offline pin incorrect
OFFLINE_PIN_CORRECT	Offline pin correct

Return Value: None

4.6.11 onRemoveCard

EmvProcess executing the method, notify the application contactless card can be remove from the card reader, Then call EmvHandler. onSetRemoveCardResponse to notify the EMV kernel to continue the flow.

public void onRemoveCard ();

Parameters: None

Return Value: None

4.6.12 onFinish

EmvProcess executing the method, means all the EMV flow is finish. The retcode will indicate the EMV transaction result.

public void onFinish (int retCode, EmvProcessResultEntity entity);

Parameters:

Parameter	Description
retCode	Enter the result : SdkResult.Success success SdkResult.Fail failure Other return results please refer to the EMV class table 3.7 or consult Appendix 5
entity	When retCode == SdkResult.Success return data Otherwise return null

EmvProcessResultEntity

Attributes	Description
byte [] scriptResult	Script execution results
List <EMVCardLogEntity> EMVlog	Cards Blog List
byte [] ecBalance	Electronic cash balance

EmvCardLogEntity

Attributes	Description
boolean isAmtExist	Whether the amount of presence
String amt	Amount of money
boolean isOtherAmtExist	Whether the existence of other
String otherAmt	Other Amount
boolean isDateExist	Date of the transaction if there are

String transDate	transaction date
boolean isTimeExist	The existence of transactions
String transTime	transaction hour
boolean isCntCodeExist	Whether there is a country code
String cntCode	country code
boolean isCurExist	Currency code if there
String curCode	Currency code
boolean isAtcExist	The existence of the transaction counter
String atc	Transaction Counter
boolean is9CExist	9c transaction type whether there
String serveType	Transaction Type
boolean isMerNameExist	The existence of a business name
String merName	Business Name

Return Value: None

4.7 OnAppOperatListener

Responsible for managing the apk install and uninstall result callback.

4.7.1 onOperatResult

Executing the installApp, after uninstallApp method is executed, the callback print the results.

public void onOperatResult (int result);

Parameters:

Parameter	Description
result	Enter the result : SdkResult.Success success SdkResult.Fail failure Other error code, please refer to Appendix

Return Value: None

4.8 OnUsbSerialReadListener

Responsible for receiving the usb serial port data.

4.8.1 onReadResult

Executing the open method, if pos receive the data from usb serial port , this callback will be triggered.

UsbSerial usbSerial = deviceEngine. getUsbSerial ();

usbSerial.open(UsbSerialCfgEntity entity, OnUsbSerialReadListener listener);

public void onReadResult(byte[] data);

Parameters:

Parameter	Description
data	The Data receive from the usb serial port.

Return Value: None

4.9 onMobileDataNetworkListener

Monitor the status of the mobile network

4.9.1 onStatusChanged

SIM cards network connection status

void onStatusChanged(Map StatusMap);

Parameters:

Parameter	Description
StatusMap	Key contains: "Sim_0" = SIM card 1 "Sim_1" = SIM card 2 Value contains: 0 = DATA DISCONNECTED 1 = DATA CONNECTING 2= DATA CONNECTED

Return Value: None

4.9.2 onStrengthChanged

SIM card Strength

void onStrengthChanged(Map StrengthMap) ;

Parameters:

Parameter	Description
StrengthMap	Key contains: "Sim_0_dbm" = dbm of SIM card 1 "Sim_0_asu" = asu of SIM card 1 "Sim_1_dbm" = dbm of SIM card 2 "Sim_1_asu" = asu of SIM card 2

Return Value: None

4.9.3 onServiceStatusChanged

operator status

void onServiceStatusChanged(int id, String operatorName);

Parameters:

Parameter	Description
id	Operator ID
operatorName	Operator name

Return Value: None

4.10 OnGpioOutPutListener

Responsible for managing the getInPutGpio class callback interface.

4.10.1 onGetHighLevel

Because the inputIO default is high level, when your peripheral circuit input the high level or don't input anything that will get the high level

public void onGetHighLevel ();

Return Value: None

4.10.2 onLowLevel

When your peripheral circuit input the low level that will get the low level

public void onLowLevel ();

Return Value: None

4.11 OnExtPinPadInputPinListener

4.11.1 onInputPinResult

After inputOnlinePin or inputOffinePin method is executed, it callback to show pin result.

public void onInputResult (int retCode, byte [] data);

Parameters:

Parameter	Description
retCode	Enter the result : SdkResult.Success success SdkResult.Fail failure SdkResult.PinPad_Input_Timeout input timeout SdkResult.PinPad_Input_Cancel cancel input SdkResulr.PinPad_No_Pin_Input no password entered, press Enter directly
data	When retCode == SdkResult.Success return Pinblock for OnlinePin. Offline Pin return 00 Otherwise return null

Return Value: None

4.12 OnExtPinPadInputAmountListener

4.12.1 onInputAmountResult

After inputAmount method is executed, it callback to show amount result.

public void onInputAmountResult (int retCode, string amount);

Parameters:

Parameter	Description
retCode	Enter the result : SdkResult.Success success SdkResult.Fail failure SdkResult.PinPad_Input_Timeout input timeout SdkResult.PinPad_Input_Cancel cancel input
amount	Amount

Return Value: None

4.13 OnExtPinPadKeyEchoListener

4.13.1 onKeyEchoResult

After inputOnlinePin or inputOffinePin method is executed, it callback to show pin result.

public void onInputResult (int retCode, byte keyCode);

Parameters:

Parameter	Description
retCode	Enter the result : SdkResult.Success success SdkResult.Fail failure SdkResult.PinPad_Input_Timeout input timeout SdkResult.PinPad_Input_Cancel cancel input
keyCode	keycode, refer to ExtPinPadKeyCode

Return Value: None

4.14 OnCardEmulationListener

4.14.1 onProcessResult

Card Emulation listener

void onProcessResult(int retCode, String msg);

Parameters:

Parameter	Description
retCode	Enter the result: SdkResult.Success success SdkResult.Fail failure SdkResult.PinPad_Input_Timeout input timeout
msg	Error message

Return Value: None

4.15 OnPlatformInitListener

4.15.1 onPlatformInitResult

Platform initialize listener

void onPlatformInitResult(int result);

Parameters:

Parameter	Description
result	Callback initialize result: SdkResult.Success success SdkResult.Fail failure SdkResult.Param_In_Invalid

Return Value: None

4.16 PlatformGeneralCallback

4.16.1 onCallbackMethod

Platform executeGeneralMethod API callback

void onCallbackMethod (byte[] data);

Parameters:

Parameter	Description
data	Data retuened by the system; the data format will be defined by NEXGO

Return Value: None

4.17 OnHoverListener

4.17.1 onHoverKey

onhover listener, will callback ok button and cancel button touch event of blind pinpad.

void onHoverKey(byte keyCode);

Parameters:

Parameter	Description
keyCode	Callback ok button and cancel button touch event: PinPadKeyCode.KEYCODE_CANCEL PinPadKeyCode.KEYCODE_CONFIRM

Return Value: None

4.18 OnExtPinPadInputTextListener

4.18.1 onInputTextResult

After inputText method is executed, it callback to show text result.

public void onInputTextResult (int retCode, string text);

Parameters:

Parameter	Description
retCode	Enter the result : SdkResult.Success success SdkResult.Fail failure SdkResult.PinPad_Input_Timeout input timeout SdkResult.PinPad_Input_Cancel cancel input
text	text

Return Value: None

4.19 OnPinPadManualInputListener

PIN pad class is responsible for the manual input callback interface.

4.19.1 onInputResult

After inputOnlinePin or inputOffinePin method is executed, it callback to show pin result.

public void onInputResult (int retCode, ManualInputInfoEntity manualInputInfoEntity);

Parameters:

Parameter	Description
retCode	Enter the result : SdkResult.Success success SdkResult.Fail failure SdkResult.PinPad_Input_Timeout input timeout SdkResult.PinPad_Input_Cancel cancel input SdkResulr.PinPad_No_Pin_Input no password entered, press Enter directly
manualInputInfoEntity	Ciphertext card data, such as ciphertext card number, expired date, CVV

ManualInputInfoEntity

Attributes	Description
String cardNo	return cipher text
String expiredDate	return cipher text

String maskCardNo	Masked card number, for example 456789******1234 Apply to device P2PE mode
String cvv	eturn cipher text
boolean isInCardBin	Card is in whitelist Apply to device P2PE mode
boolean isAppInSredWhiteList	Application is in SRED whitelist Apply to device P2PE mode

Return Value: None

4.19.2 onSendKey

After inputText, inputOnlinePin, inputOfflinePin, manualInput method, this is executed, callback the key input, when input password, number key will return KEYCODE_STAR.

public void onSendKey (byte keyCode);

Parameters:

Parameter	Description
keyCode	Input key value

Key-Value

Constant Name	Constant Value	Description
KEYCODE_0	0x30 (byte)	0
KEYCODE_1	0x31 (byte)	1
KEYCODE_2	0x32 (byte)	2
KEYCODE_3	0x33 (byte)	3
KEYCODE_4	0x34 (byte)	4
KEYCODE_5	0x35 (byte)	5
KEYCODE_6	0x36 (byte)	6
KEYCODE_7	0x37 (byte)	7
KEYCODE_8	0x38 (byte)	8
KEYCODE_9	0x39 (byte)	9
KEYCODE_ the STAR	0x2a (byte)	*

KEYCODE_OCTOTHORPE	0x23 (byte)	#
KEYCODE_CANCEL	0x18 (byte)	Cancel key
KEYCODE_BACKSPACE	0x08 (byte)	Backspace
KEYCODE_CLEAR	0xfe (byte)	Clear key
KEYCODE_CONFIRM	0x0d (byte)	Enter

Return Value: None

Appendix

Return Value Description

```
public class SdkResult {
        public final static int Success = 0;
        public final static int Fail = -1;
        public final static int Param_In_Invalid = -2;
        public final static int TimeOut = -3;
        / * Device not signed * /
        public final static int Device_Not_Ready = -4;
        // ---- Printer Error -----
        private final static int Printer_Base_Error = -1000;
        / ** * Print failed /
        public final static int Printer_Print_Fail = Printer_Base_Error -1;
        / * Failed to set string buffer * /
        public final static int Printer_AddPrnStr_Fail = Printer_Base_Error -2;
        / ** * Set picture buffer failure /
        public final static int Printer_AddImg_Fail = Printer_Base_Error -3;
        / ** * Printer Busy /
        public final static int Printer_Busy = Printer_Base_Error - 4;
        / ** * The printer is out of paper /
        public final static int Printer_PaperLack = Printer_Base_Error - 5;
        / ** * Wrong packet format print /
        public final static int Printer_Wrong_Package = Printer_Base_Error - 6;
```

```
/ ** * Printer Fault /
public final static int Printer_Fault = Printer_Base_Error - 7;
/ ** * Printer overheating /
public final static int Printer_TooHot = Printer_Base_Error - 8;
/ ** * Print the unfinished /
public final static int Printer_UnFinished = Printer_Base_Error - 9;
/ ** Other exception error * /
public final static int Printer_Other_Error = Printer_Base_Error-999;
// ---- Scanner Error -----
private final static int Scanner_Base_Error = -2000;
/ ** * Button to exit the user /
public final static int Scanner_Customer_Exit = Scanner_Base_Error-1;
/ ** Other exception error * /
public final static int Scanner_Other_Error = Scanner_Base_Error-999;
// ---- SerialPort Error -----
private final static int SerialPort_Base_Error = -4000;
/ ** * Serial connection failure /
public final static int SerialPort_Connect_Fail = SerialPort_Base_Error - 1;
/ ** * Serial data transmission failure /
public final static int SerialPort_Send_Fail = SerialPort_Base_Error - 2;
/ ** Fd error * /
public final static int SerialPort_Fd_Error = SerialPort_Base_Error - 3;
/ ** * Unopened serial /
public final static int SerialPort_Port_Not_Open = SerialPort_Base_Error - 4;
/ ** * Serial scission failure /
public final static int SerialPort_DisConnect_Fail = SerialPort_Base_Error - 5;
/ ** Transmit buffer is not empty (the remaining data to be transmitted) * /
public final static int SerialPort_Sending_Buf_IsNot_Null = SerialPort_Base_Error - 6;
/ ** Invalid channel number * /
public final static int SerialPort_Invalid_Channel = SerialPort_Base_Error - 7;
/ ** Channel is not open and no communication with any physical port * /
public final static int SerialPort_Channel_Isnot_Open = SerialPort_Base_Error - 8;
```

```
/ ** Transmit buffer error (continue 500ms at full state) * /
public final static int SerialPort_Sending_Buffer_Error = SerialPort_Base_Error - 9;
/ ** No available physical port * /
public final static int SerialPort_No_Available_Ports = SerialPort_Base_Error - 10;
/ ** Device enumeration and configuration process is not completed (USB DEV dedicated * /
public final static int SerialPort_Conf_Process_Error = SerialPort_Base_Error - 11;
/ ** Equipment de-energized and the host loses connection (USB DEV dedicated * /
public final static int SerialPort_Device_Lost_Power = SerialPort_Base_Error - 12;
/ ** From the host device and then plug plucking (USB DEV dedicated) * /
public final static int SerialPort_Unplug_Error = SerialPort_Base_Error - 13;
/ ** Device is off (USBDEV dedicated) * /
public final static int SerialPort_Device_Is_Off = SerialPort_Base_Error - 14;
/ ** * Data receive timeout /
public final static int SerialPort_Timeout_Receiving_Data = SerialPort_Base_Error - 15;
/ *** Channel is being occupied by the system * /
public final static int SerialPort_Channle_Is_Occupied = SerialPort_Base_Error - 16;
/ ** Invalid communication Parameters, communication Parameters do not meet the rules for strings or data beyond the normal range. * /
public final static int SerialPort_Invalid_Communication_Parameter = SerialPort_Base_Error - 17;
/ ** USB to serial device mounted unsuccessful (the Return Value only FIDI USB to serial use) * /
public final static int SerialPort_Usb_Mounted_Unsuccessful = SerialPort_Base_Error - 18;
/ ** Usb to serial device error (only FTDI USB serial port using the Return Value of re-exports * /
public final static int SerialPort_Reset_Usb_Error = SerialPort_Base_Error - 19;
/ ** Device USB to serial chip traffic congestion (only FIDI USB serial adapter used in the Return Value) * /
public final static int SerialPort_Devices_Error = SerialPort_Base_Error - 20;
/ ** Other exception error * /
public final static int SerialPort_Other_Error = SerialPort_Base_Error-999;
// ---- MagCardReader Error -----
private final static int MagCardReader_Base_Error = -5000;
/ ** * No credit card /
public final static int MagCardReader_No_Swiped = MagCardReader_Base_Error -1;
/ ** Other exception error * /
```

```
public final static int MagCardReader_Other_Error = MagCardReader_Base_Error -999;
// ---- IccCardReader Error -----
private final static int IccCardReader_Base_Error = -6000;
public final static int IccCardReader_Read_CardType_Error = IccCardReader_Base_Error-1;
public final static int IccCardReader_CardInit_Error = IccCardReader_Base_Error-2;
/ ** Other exception error * /
public final static int IccCardReader_Other_Error = IccCardReader_Base_Error-999;
// ---- PinPad Error ----
private final static int PinPad_Base_Error = -7000;
/ ** * Key does not exist /
public final static int PinPad_No_Key_Error = PinPad_Base_Error - 1;
/ ** Wrong key index, the index is not within the Parameters range * /
public final static int PinPad_KeyIdx_Error = PinPad_Base_Error - 2;
/ ** Did not enter PIN * /
public final static int PinPad_No_Pin_Input = PinPad_Base_Error - 3;
/ ** Cancel Enter PIN * /
public final static int PinPad_Input_Cancel = PinPad_Base_Error - 4;
/ ** * Key length wrong /
public final static int PinPad_Key_Len_Error = PinPad_Base_Error - 8;
/ ** Enter PIN Timeout * /
public final static int PinPad_Input_Timeout = PinPad_Base_Error - 9;
/ ** Open or close Pinpad failed * /
public final static int PinPad_Open_Or_Close_Error = PinPad_Base_Error - 10;
/ ** Pinpad process error * /
public final static int PinPad_Deal_Error = PinPad_Base_Error - 11;

/ ** Pinpad DUKPT KSN exceed limit * /
public final static int PinPad_KSN_Exceed_Limit = PinPad_Base_Error - 16;

/ ** Other exception error * /
public final static int PinPad_Other_Error = PinPad_Base_Error-999;
// ---- EMVHandler Error ----
```

```
private final static int EMVHandler_Base_Error = -8000;
/ ** <Try other communication interface * /
public final static int EMV_Other_Interface = EMVHandler_Base_Error - 1;
/ ** <Contactless transactions offline approved * /
public final static int EMV_Qpboc_Offline = EMVHandler_Base_Error - 2;
/ ** <Contactless union pay online transaction * /
public final static int EMV_Qpboc_Online = EMVHandler_Base_Error - 3;
/ ** <Contactless PBOC online transaction , Abolished * /
public final static int EMV_Pboc_Online = EMVHandler_Base_Error - 4;
/ ** <Contactless MSD online transaction, Abolished * /
public final static int EMV_MSD_Online = EMVHandler_Base_Error - 5;
/ ** <Offline electronic cash acceptance, Abolished * /
public final static int EMV_Ec_Accept = EMVHandler_Base_Error - 6;
/ ** <contact transaction Offline approved * /
public final static int EMV_Offline_Accept = EMVHandler_Base_Error - 7;
/ ** <Transaction card is removed * /
public final static int EMV_Card_Removed = EMVHandler_Base_Error -8;
/ ** <Reader failed * /
public final static int EMV_Command_Fail = EMVHandler_Base_Error -9;
/ ** <Card is Blocked * /
public final static int EMV_Card_Block = EMVHandler_Base_Error -10;
/ ** <Parameters wrong * /
public final static int EMV_PARA_ERR = EMVHandler_Base_Error -11;
/ ** <No common application * /
public final static int EMV_Candidatelist_Empty = EMVHandler_Base_Error -12;
/ ** <Application locked * /
public final static int EMV_App_Block = EMVHandler_Base_Error -13;
/ ** <Transaction fallback , need to swipe card* /
public final static int EMV_FallBack = EMVHandler_Base_Error -14;
/ ** <Data authentication has failed * /
public final static int EMV_Auth_Fail = EMVHandler_Base_Error -15;
/ ** <Application has not yet entered into force * /
```

```
public final static int EMV_App_Ineffect = EMVHandler_Base_Error -16;
/ ** <Application has expired * /
public final static int EMV_App_Expired = EMVHandler_Base_Error -17;
/ ** <Cardholder verification failed * /
public final static int EMV_Cvm_Fail = EMVHandler_Base_Error -18;
/ ** <* Transactions should the online,  Abolished /
public final static int EMV_Online = EMVHandler_Base_Error -19;
/ ** <Cancel the transaction * /
public final static int EMV_Cancel = EMVHandler_Base_Error -20;
/ ** <Transaction online decline * /
public final static int EMV_Declined = EMVHandler_Base_Error -21;
/ ** <Issuer Authentication failed * /
public final static int EMV_Arpc_Fail = EMVHandler_Base_Error -22;
/ ** <Issuer Script execution failed * /
public final static int EMV_Script_Fail = EMVHandler_Base_Error -23;
/ ** <Applications are not accepted, you can re-select * /
public final static int EMV_App_NoAccept = EMVHandler_Base_Error -24;
/ ** <Electronic cash offline decline * /
public final static int EMV_Ec_Decliend = EMVHandler_Base_Error -25;
/ ** <Successful transaction, Issuer Authentication failed * /
public final static int EMV_Sucess_Arpc_Fail = EMVHandler_Base_Error -26;
/ ** <plese see phone* /
public final static int Emv_Plz_See_Phone = EmvHandler_Base_Error - 27;
/ ** < Transaction Terminate * /
public final static int Emv_Terminate = EmvHandler_Base_Error - 28;
/ ** < Transaction Communicate Timeout * /
public final static int Emv_Communicate_Timeout = EmvHandler_Base_Error - 29;
/ ** < Use other card * /
public final static int Emv_USE_OTHER_CARD = EmvHandler_Base_Error - 30;
/ ** < Kernel end the application * /
public final static int Emv_CTLS_EndApplication = EmvHandler_Base_Error - 32;
```

```
/ ** < Torn * /
public final static int Emv_CTLS_Torn = EmvHandler_Base_Error - 33;


/ ** < Please try again * /
public final static int Emv_CTLS_TransTryAgain = EmvHandler_Base_Error - 34;


/ ** < Card in Exception * /
public final static int Emv_CTLS_CardInException = EmvHandler_Base_Error - 35;



/ ** * Other error exception /
public final static int EMV_Other_Error = EMVHandler_Base_Error -999;
// ---- CardHandler Error -----
// ---- ~ -10000 -19900 Allocated to card manipulation
// ---- -10.1 Thousand representatives contact CPU card -10200-- non-contact CPU card supports a total of 99 kinds of card types
private final static int CardHandler_Base_Error = -10000;
// ---- Contactless IC card return code segment
public final static int Icc_Base_Error = CardHandler_Base_Error -100;
/ ** Transaction card dialed * /
public final static int Icc_PullOut_Card = Icc_Base_Error - 1;
/ ** Parity error * /
public final static int Icc_Parity_Err = Icc_Base_Error - 2;
/ ** Select the channel error * /
public final static int Icc_Channel_Err = Icc_Base_Error - 3;
/ ** Transmit data too long (LC) * /
public final static int Icc_Data_Len_TooLong = Icc_Base_Error - 4;
/ ** Card Error protocol (T = 0 or not T = 1) * /
public final static int Icc_Protocol_Err = Icc_Base_Error - 5;
/ ** * Not reset the card /
public final static int Icc_No_Reset_Card = Icc_Base_Error - 6;
```

```
/ ** The dead can not communicate or * /
public final static int Icc_Not_Call = Icc_Base_Error - 7;
/ ** Other exception error * /
public final static int Icc_Other_Error = Icc_Base_Error - 99;
// ---- Contactless IC card return code segment
private final static int Picc_Base_Error = CardHandler_Base_Error -200;
/ ** * RF module is not turned on /
public final static int Picc_Not_Open = Picc_Base_Error - 1;
/ ** Not find the card (the sensor area no specific type of card) * /
public final static int Picc_Not_Searched_Card = Picc_Base_Error - 2;
/ ** Card induction area too (there is a communication conflict) * /
public final static int Picc_Card_Too_Many = Picc_Base_Error - 3;
/ ** Protocol error (data in violation of the agreement appears response card) * /
public final static int Picc_Protocol_Data_Err = Picc_Base_Error - 4;
/ ** * Card not active /
public final static int Picc_Card_No_Activation = Picc_Base_Error - 5;
/ ** * Conflict Doka /
public final static int Picc_Muti_Card_Err = Picc_Base_Error - 6;
/ ** Protocol error * /
public final static int Picc_Protocol_Err = Picc_Base_Error - 7;
/ ** Communications transmission error * /
public final static int Picc_Io_Err = Picc_Base_Error - 8;
/ ** * Card is still the sensor area /
public final static int Picc_Card_Sense_Err = Picc_Base_Error - 9;
/ ** Card status error (such as A / B card calling card interface M1, or M1 card call
PiccIsoCommand Interface) * /
public final static int Picc_Card_Status_Err = Picc_Base_Error - 10;
/ ** Interface chip does not exist or abnormal * /
public final static int Picc_Not_Call = Picc_Base_Error - 11;
/ ** * Other error exception /
public final static int Picc_Other_Error = Picc_Base_Error - 99;
/ ** M1 card section * /
```

```
public final static int M1Card_Base_Error = CardHandler_Base_Error -300;
/ ** M1 card authentication failed * /
public final static int M1Card_Verify_Err = M1Card_Base_Error - 1;
/ ** * Sector unauthenticated /
public final static int M1Card_Fan_Not_Verify = M1Card_Base_Error - 2;
/ ** Numeric data block format is wrong * /
public final static int M1Card_Data_Block_Err = M1Card_Base_Error - 3;
/ ** * Module unopened /
public final static int M1Card_Not_Open = M1Card_Base_Error - 4;
/ ** * Card not active /
public final static int M1Card_Card_Not_Activation = M1Card_Base_Error - 5;
/ ** Wrong type of card operations for operateBlock the Senate operType Check * /
public final static int M1Card_Card_OperType_Error = M1Card_Base_Error - 6;
/ ** * Other error exception /
public final static int M1Card_Other_Error = M1Card_Base_Error - 99;

//---- -20000 - 21000 Platform
private final static int Platform_Base_Error = -20000;

//20100-20199 install
public final static int Platform_Install_Base_Error = Platform_Base_Error - 100;

/** the package is already installed.*/
public final static int Platform_Install_Already_Exists = Platform_Install_Base_Error - 1;

/**the package archive file is invalid*/
public final static int Platform_Install_Invalid_Apk = Platform_Install_Base_Error - 2;

/**the URI passed in is invalid.*/
public final static int Platform_Install_Invalid_Uri = Platform_Install_Base_Error - 3;
```

/**the package manager service found that the device didn't have enough storage space to install the app.*/

public final static int Platform_Install_Insufficient_Storage = Platform_Install_Base_Error - 4;

/**package is already installed with the same name.*/

public final static int Platform_Install_Duplicate_Package = Platform_Install_Base_Error - 5;

/**the requested shared user does not exist*/

public final static int Platform_Install_No_Shared_User = Platform_Install_Base_Error - 6;

/**a previously installed package of the same name has a different signature than the new package (and the old package's data was not removed).*/

public final static int Platform_Install_Update_Incompatible = Platform_Install_Base_Error - 7;

/**the new package is requested a shared user which is already installed on the device and does not have matchingsignature*/

public final static int Platform_Install_Shared_User_Incompatible = Platform_Install_Base_Error - 8;

/**the new package uses a shared library that is not available*/

public final static int Platform_Install_Missing_Shared_Library = Platform_Install_Base_Error - 9;

/**the new package replace failed, case the delete failed.*/

public final static int Platform_Install_Replace_Delete_Failed = Platform_Install_Base_Error - 10;

/**the new package failed while optimizing and validating its dex files, either because there was not enough storage or the validation failed*/

public final static int Platform_Install_Dexopt = Platform_Install_Base_Error - 11;

//20200-20299 uninstall

public final static int Platform_Uninstall_Base_Error = Platform_Base_Error - 200;

/**the system failed to uninstall the apk for an unspecified reason*/

```
        public final static int Platform_Uninstall_Internal_Error = Platform_Uninstall_Base_Error - 1;

        /**the system failed to uninstall the apk because it is the active DevicePolicy manager.*/
        public final static int Platform_Uninstall_Device_Policy_Manager = Platform_Uninstall_Base_Error - 2;

        /**the system failed to uninstall the apk since the user is restricted*/
        public final static int Platform_Uninstall_User_Restricted = Platform_Uninstall_Base_Error - 3;

        /**the system failed to uninstall the apk because a profile or device owner has marked the package as uninstallable*/
        public final static int Platform_Uninstall_Owner_Blocked = Platform_Uninstall_Base_Error - 4;

        /**abort the uninstall*/
        public final static int Platform_Uninstall_Aborted = Platform_Uninstall_Base_Error - 5;

        //20300-20399 update logo & animation
        public final static int Platform_Update_Base_Error = Platform_Base_Error - 300;

        /**does not match, can not process*/
        public final static int Platform_Update_No_Match = Platform_Update_Base_Error - 1;

        /**update failed*/
        public final static int Platform_Update_Failed = Platform_Update_Base_Error - 2;
}
```