



# JS 外设调用接口文档

## 目录

修订记录 .....	9
导出设备说明 .....	10
导入设备示例 .....	10
<b>1. CardDispenser .....</b>	<b>11</b>
1.1 getCardUnitInfoAsync .....	11
1.2 setCardUnitInfoAsync .....	12
1.3 dispenseCardAsync .....	12
1.4 ejectCardAsync .....	13
1.5 cancelAsync.....	13
1.6 waitCardTakenAsync.....	13
1.7 getStatusAsync.....	13
1.8 retainCardAsync.....	14
<b>2. CardReader .....</b>	<b>14</b>
2.1 readCardTrack23Async .....	14
2.2 readCardChipAsync.....	15
2.3 ejectCardAsync .....	15
2.4 cancelAsync.....	15
2.5 retainCardAsync.....	15
2.6 waitCardTakenAsync.....	15
2.7 getStatusAsync.....	15
2.8 isCardAtPresentAsync.....	18
2.9 isCardAtEntranceAsync.....	18
2.10 isCardAtPresentOrAtEntranceAsync.....	18
2.11 powerUpAsync.....	18
2.12 powerDownAsync.....	18
2.13 pbocGetAppInfoByAIDAsync .....	18
2.14 pbocGetAppInfoByDebitAIDAsync.....	19
2.15 pbocGetAppInfoByCreditAIDAsync .....	19
2.16 pbocAppInitializeAsync.....	19
2.17 pbocAppInitializeByParamsAsync.....	20

2.18	pbocAppInitializeByDefaultParamsAsync.....	21
2.19	pbocReadAppDataAsync.....	21
2.20	pbocGetDataAsync .....	23
2.21	pbocGetECashLimitAsync .....	24
2.22	pbocGetECashTradeLimitAsync .....	24
2.23	pbocGetECashBalanceAsync.....	24
2.24	pbocGetECashResetThresholdAsync .....	24
2.25	pbocGetECashOffLineBalanceAsync.....	24
2.26	pbocGetECashCVMLimitAsync.....	24
2.27	pbocStartTradeAsync.....	24
2.28	pbocGenerateUnionpayField55Async .....	24
2.29	pbocGenerateUnionpayField55ByRequiredFieldAsync.....	24
2.30	pbocExternalAuthAsync.....	24
2.31	pbocFinishTradeAsync .....	25
2.32	pbocExcuteScriptAsync.....	25
2.33	pbocReadBinaryAsync .....	25
2.34	pbocUpdateBinaryAsync.....	25
2.35	pbocGetChallengeAsync .....	25
2.36	pbocGetChallengeOf4BAsync .....	25
2.37	pbocSelectByFileIdAsync .....	25
2.38	pbocSelectMFAsync.....	25
2.39	pbocChipIOAsync.....	25
2.40	pbocGetBalanceAsync .....	25
2.41	pbocInitForLoadAsync .....	25
2.42	pbocCreditForLoadAsync.....	26
2.43	getCapsAsync.....	27
2.44	pbocStartTradeAsync.....	29
2.45	pbocGetTerminalInfoAsync .....	30
3.	<b>FingerprintReader .....</b>	<b>33</b>
3.1	getSampleAsync.....	33
3.2	getFeatureImageAsync .....	33
3.3	getFeatureDataAsync .....	33

3.4	isFeatureMatchAsync .....	33
3.5	cancelAsync.....	33
3.6	getStatusAsync.....	33
4.	<b>ICCardReader</b> .....	36
4.1	readCardAsync .....	36
4.2	cancelAsync.....	36
4.3	getStatusAsync.....	36
4.4	pboc 操作 .....	36
4.5	getCapsAsync.....	36
5.	<b>IDCardReader</b> .....	36
5.1	readIdCardAsync.....	36
5.2	ejectCard.....	37
5.3	cancel.....	37
5.4	waitIDCardTaken .....	37
5.5	getStatus.....	37
5.6	getCapsAsync.....	37
6.	<b>PinKeypad</b> .....	37
6.1	getPinAsync.....	37
6.2	getDataAsync .....	39
6.3	waitKeyAsync.....	40
6.4	getPinBlockAsync.....	40
6.5	getPinBlockByCardNoAsync.....	41
6.6	cancelAsync.....	41
6.7	getStatusAsync.....	41
6.8	importMasterKeyAsync .....	42
6.9	importMasterKeyExAsync.....	42
6.10	importPinKeyAsync.....	42
6.11	importPinKeyExAsync .....	42
6.12	getPinBlockExAsync .....	43
6.13	getPinBlockExByCardNoAsync .....	43
6.14	cryptAsync .....	43
6.15	importKeyAsync.....	44

6.16	importKeyExAsync .....	45
6.17	importRsaPublicKeyExAsync.....	46
7.	<b>ReceiptPrinter</b> .....	47
7.1	printAsync.....	47
7.2	getStatusAsync.....	47
7.3	扩展调用说明 .....	47
7.4	printByFromNameAsync(data, formName, mediaName).....	47
7.5	printByRawFromAsync(data, formName, mediaName).....	48
8.	<b>UKeyDispenser</b> .....	48
8.1	getCardUnitInfoAsync .....	48
8.2	setCardUnitInfoAsync .....	48
8.3	dispenseCardAsync .....	49
8.4	ejectCardAsync .....	49
8.5	cancelAsync.....	49
8.6	waitUKeyTakenAsync .....	49
8.7	retainCardAsync.....	49
9.	<b>UKeyReader</b> .....	50
9.1	readAsync .....	50
9.2	cancelAsync.....	50
9.3	waitUKeyTakenAsync.....	50
9.4	getStatusAsync.....	50
9.5	retainCardAsync.....	50
10.	<b>Barcode</b> .....	50
10.1	readAsync .....	50
10.2	cancelAsync.....	50
10.3	getStatusAsync.....	50
11.	<b>StampPrinter</b> .....	51
11.1	printAsync.....	51
11.2	cancelAsync.....	51
11.3	getStatusAsync.....	51
11.4	ejectAsync.....	51
11.5	controlMediaAsync(ctrls).....	51

12.	Sensors .....	52
12.1	openCardUnitGuidLightAsync .....	52
12.2	closeCardUnitGuidLightAsync .....	52
12.3	openPinPadGuidLightAsync.....	52
12.4	closePinPadGuidLightAsync.....	52
12.5	openNotesDispenserGuidLightAsync .....	52
12.6	closeNotesDispenserGuidLightAsync .....	52
12.7	openSPCardReaderGuidLightAsync .....	52
12.8	closeSPCardReaderGuidLightAsync .....	52
12.9	openReceiptPrinterGuidLightAsync.....	52
12.10	closeReceiptPrinterGuidLightAsync.....	52
12.11	openPassbookPrinterGuidLightAsync.....	52
12.12	closePassbookPrinterGuidLightAsync.....	52
12.13	openUKeyGuidLightAsync .....	52
12.14	closeUKeyGuidLightAsync .....	52
12.15	openChequeUnitGuidLightAsync.....	53
12.16	closeChequeUnitGuidLightAsync.....	53
12.17	openRFCardGuidLightAsync .....	53
12.18	closeRFCardGuidLightAsync .....	53
12.19	openBillAcceptorGuidLightAsync .....	53
12.20	closeBillAcceptorGuidLightAsync .....	53
12.21	openDocumentPrinterGuidLightAsync.....	53
12.22	closeDocumentPrinterGuidLightAsync.....	53
12.23	openIDCardGuidLightAsync.....	53
12.24	closeIDCardGuidLightAsync.....	53
12.25	openScannerGuidLightAsync.....	53
12.26	closeScannerGuidLightAsync.....	53
12.27	openFingerGuidLightAsync.....	53
12.28	closeFingerGuidLightAsync.....	53
12.29	openSignatureGuidLightAsync .....	53
12.30	closeSignatureGuidLightAsync .....	53
12.31	openHRCameraGuidLightAsync.....	53

12.32	closeHRCameraGuidLightAsync.....	53
13.	PassbookPrinter .....	53
13.1	printAsync.....	53
13.2	printWithEmptyLinesAsync .....	54
13.3	readFormAsync.....	54
13.4	cancelAsync .....	54
13.5	getStatusAsync .....	54
14.	SignatureTablet .....	54
14.1	getSignatureAsync.....	54
14.2	candleSignatureAsync.....	54
15.	CashIn .....	54
15.1	LockAsync() .....	54
15.2	unlockAsync().....	54
15.3	cancelAsync() .....	54
15.4	getStatusAsync() .....	54
15.5	getCapabilitiesAsync() .....	58
15.6	getCashUnitInfoAsync() .....	61
15.7	getTellerInfoAsync(tellerInfo).....	70
15.8	getCurrencyExpAsync().....	71
15.9	getBanknoteTypesAsync() .....	71
15.10	getCashInStatusAsync() .....	73
15.11	getP6InfoAsync().....	73
15.12	getP6SignatureAsync(signature) .....	74
15.13	cashInStartAsync(cashInStart) .....	75
15.14	cashInAsync() .....	76
15.15	cashInEndAsync() .....	77
15.16	cashInRollbackAsync() .....	81
15.17	retractAsync(retract) .....	81
15.18	openShutterAsync(position).....	83
15.19	closeShutterAsync(position).....	83
15.20	setTellerInfoAsync(tellerUpdate) .....	83
15.21	setCashUnitInfoAsync(cashInfo).....	85

15.22	startExchangeAsync(startEx) .....	89
15.23	endExchangeAsync(cashInfo) .....	90
15.24	openSafeDoorAsync() .....	95
15.25	resetAsync(itemPosition) .....	95
15.26	configureCashInUnitsAsync(cashInType) .....	96
15.27	configureNoteTypesAsync(noteIDs) .....	97
15.28	createP6SignatureAsync() .....	97
16.	<b>CashDispenser</b> .....	98
16.1	lockAsync().....	98
16.2	unlockAsync().....	98
16.3	cancelAsync() .....	98
16.4	getStatusAsync() .....	98
16.5	getCapabilitiesAsync() .....	101
16.6	getCashUnitInfoAsync() .....	102
16.7	getTellerInfoAsync(tellerInfo).....	108
16.8	getCurrencyExpAsync().....	109
16.9	getMixTypesAsync().....	109
16.10	getMixTableAsync(mixNumber) .....	111
16.11	getPresentStatusAsync(position) .....	112
16.12	denominateAsync(denominate).....	113
16.13	dispenseAsync(dispense).....	114
16.14	countAsync(physicalCU) .....	115
16.15	presentAsync(position).....	115
16.16	rejectAsync().....	116
16.17	retractAsync(retract) .....	116
16.18	openShutterAsync(position).....	117
16.19	closeShutterAsync(position).....	118
16.20	setTellerInfoAsync(tellerUpdate) .....	119
16.21	setCashUnitInfoAsync(cashInfo).....	120
16.22	startExchangeAsync(startEx) .....	122
16.23	endExchangeAsync(cashInfo) .....	125
16.24	openSafeDoorAsync() .....	128



16.25	calibrateCashUnitAsync(calibrate) .....	128
16.26	setMixTableAsync(mixTable) .....	130
16.27	resetAsync(itemPosition) .....	130
16.28	testCashUintsAsync(itemPosition) .....	132

## 修订记录

变更日期	版本	章节	说明
2017/04/21	V0.1		初稿
2017/04/21	V0.11	2.6、2.7、2.8、2.9	新增读卡器获取状态的接口
2017/04/21	V0.12	2.10-2.31	增加 pboc 接口
2017/05/13	V0.13	2.2	新增读取芯片卡接口
2017/05/15	V0.14	6.4、6.5	添加获取密文说明，新增根据卡号获取秘钥方法
2017/05/18	V0.15	1.6、5.4、8.6、9.3	新增等待卡片取走方法
2017/05/21	V0.16	13	新增存折打印机接口
2017/06/02	V0.17	1-13	新增各个设备模块的获取状态，pboc 新增常用操作方法
2017/06/04	V0.18	6.8、6.9、6.10、6.11	新增导入秘钥方法
2017/06/04	V0.19	1.8、8.7、9.5	新增吞卡方法
2017/06/05	V0.20	2.40、2.41、2.42	新增圈存方法
2017/06/09	V0.21	14	新增签名板调用
2017/08/01	V0.22	2、4、5	新增 getCaps 方法获取设备支持信息
2017/08/04	V0.23	15、16	新增现金存入与现金发放设备模块接口
2017/08/11	V0.24	7.4、7.5	凭条打印新增可以指定 form 名称的接口
2017/08/16	V0.25	6.12、6.13、11.4、11.5， 导出设备说明	密码键盘新增国密调用接口，盖章打印新增控制媒介接口
2017/09/16	V0.26	2.44、2.45、6.14、6.15、 6.16	pboc 添加开始交易扩展方法和获取终端信息接口，密码键盘新增加解密和导入秘钥接口
2017/09/17	V0.27	6.15、6.16、6.17	更新导入秘钥方法说明，添加导入 RSA、SM2 秘钥的方法

## 导出设备说明

导出名称	设备名称	导出模块	备注
CardDispenser	发卡器	awp-plugin-device-card-dispenser	
CardReader	读卡器	awp-plugin-device-card-reader	
FingerprintReader	指纹仪	awp-plugin-device-fingerprint-reader	
ICCardReader	非接 IC 读卡器	awp-plugin-device-iccard-reader	
IDCardReader	身份证阅读器	awp-plugin-device-idcard-reader	
PinKeypad	密码键盘	awp-plugin-device-pinkeypad	使用密码控件
ReceiptPrinter	凭条打印机	awp-plugin-device-receipt-printer	
UKeyDispenser	UKey 发放器	awp-plugin-device-ukey-dispenser	
UKeyReader	Ukey 阅读器	awp-plugin-device-ukey-reader	
Barcode	扫码器	awp-plugin-device-barcode-reader	
StampPrinter	打印盖章机	awp-plugin-device-stamp-printer	
Sensors	指示灯	awp-plugin-device-sensors-indicators	
PassbookPrinter	存折打印机	awp-plugin-device-passbook-printer	
SignatureTablet	签名板	awp-plugin-device-signature-tablet	
CashIn	现金存入模块	awp-plugin-device-cash-in	
CashDispenser	现金发放模块	awp-plugin-device-cash-dispenser	

## 导入设备示例

```
import { ReceiptPrinter } from "awp-plugin-device-receipt-printer";
```

导入特殊说明：有些设备不方便直接调用会增加包装层或者控件，目前密码键盘使用控件，凭条打印机使用扩展。

# 1. CardDispenser

## 1.1 getCardUnitInfoAsync

函数说明：获取卡箱信息。

返回信息：所有卡箱信息。

输出示例：

```
[
  {
    "HardwareSensor": false,
    "Status": 0,
    "Threshold": 0,
    "RetainCount": 0,
    "Count": 0,
    "InitialCount": 0,
    "Type": 0,
    "CardName": null,
    "Number": 0
  },
  {
    "HardwareSensor": false,
    "Status": 0,
    "Threshold": 0,
    "RetainCount": 0,
    "Count": 0,
    "InitialCount": 0,
    "Type": 0,
    "CardName": null,
    "Number": 0
  }
]
```

字段说明：

字段	字段含义	备注
HardwareSensor	是否启动硬件传感器	启动硬件传感器后，当 Count 值达到 Threshold 数量时产生事件,具体参考 wosa 文档。
Status	卡箱状态	0:正常 1:卡少 2:卡箱空 3:卡箱不可用 4:卡箱不存在 5:卡量充足 6 卡箱满 7:状态不可用
Threshold	阈值数量	
RetainCount	吞卡数量	

Count	当前数量	
InitialCount	初始数量	
Type	卡箱类型	1:发卡箱, 2:吞卡箱
CardName	卡名称	
Number	卡箱号	

## 1.2 setCardUnitInfoAsync

函数说明：设置卡箱信息。

输入参数：

**cardUnitInfo**：卡箱信息数组

输入示例：

```
[
  {
    "HardwareSensor": false,
    "Status": 0,
    "Threshold": 0,
    "RetainCount": 0,
    "Count": 0,
    "InitialCount": 0,
    "Type": 0,
    "CardName": null,
    "Number": 0
  },
  {
    "HardwareSensor": false,
    "Status": 0,
    "Threshold": 0,
    "RetainCount": 0,
    "Count": 0,
    "InitialCount": 0,
    "Type": 0,
    "CardName": null,
    "Number": 0
  }
]
```

字段说明：同 1.1 字段说明。

## 1.3 dispenseCardAsync

函数说明：发卡。

输入参数：

**dispense**：发卡控制信息

输入示例：

```
{
  "Present": false,
```

"Number": 0

}

字段说明:

字段	字段含义	备注
Present	是否呈现给客户	false:卡发到卡道内;true:卡发到卡口
Number	发卡卡箱号	

#### 1.4 ejectCardAsync

函数说明: 退出卡片。

#### 1.5 cancelAsync

函数说明: 取消, 用来取消阻塞的操作。

#### 1.6 waitCardTakenAsync

函数说明: 等待卡片取走。

函数类型: 阻塞

#### 1.7 getStatusAsync

函数说明: 获取设备状态信息

返回信息:

```
{
  "AntiFraudModule": 0,
  "PowerSaveRecoveryTime": 0,
  "CrdDevicePosition": 0,
  "GuidLights": [],
  "Extra": {},
  "Shutter": 0,
  "Media": 0,
  "Transport": 0,
  "Dispenser": 0,
  "Device": 0
}
```

字段	字段含义	备注
AntiFraudModule	反欺骗模块	0: 不支持; 1: 状态正常; 2: 设备不可用 3: 检测到外部设备 4: 状态未知
PowerSaveRecoveryTime	节电唤醒时间	
CrdDevicePosition	设备位置	0: 设备处于正常操作位置 1: 设备被从它的正常操作位置移走 2: 由于硬件错误或者其它的情况, 读卡器的位置不能确定 3: 实际设备没有能力检测其位置
GuidLights	指示灯状态	指示灯状态数组 [{

		"Status": "0", "Color": "0", }
Extra	厂商自定义信息	
Shutter	卡口门状态	0: 关闭 1: 打开 2: 卡住 3: 未知 4: 不支持
Media	媒介状态	1: 媒介在设备内 2: 没有在设备内，也没有在出口位置 3: 媒介卡在设备内 4: 不支持 5: 未知 6: 出口位置 7: 吞卡箱
Transport	传送单元状态	0: 正常 1: 不可用 2: 未知 3: 不支持
Dispenser	发卡器卡单元状态	0: 所有单元正常 1: 至少一个单元可用，一个或多个单元是空的或者不可用状态 2: 所有单元是空的 3: 状态未知
Device	设备状态	0: 设备在线 1: 设备脱机 2: 设备断电或物理没连接 3: 设备不存在 4: 设备因硬件错误无法操作 5: 设备存在但有人正在阻止设备进行合适的操作 6: 设备忙当前不能处理指令 7: 设备存在但是检测到试图欺骗 8: 可能存在欺骗

## 1.8 retainCardAsync

函数说明：吞卡

输入信息：

CardBoxNumber: 吞卡卡箱号

## 2. CardReader

### 2.1 readCardTrack23Async

函数说明：读取卡片 23 磁道信息

函数类型：阻塞

返回信息：23 磁道信息结构

```

{
  "Track2": null,
  "Track3": null,
  "CardNumber": null
}

```

字段说明:

字段	字段含义	备注
Track2	二磁道信息	
Track3	三磁道信息	
CardNumber	卡号	

## 2.2 readCardChipAsync

函数说明: 读取芯片卡

返回信息: 卡 UID

## 2.3 ejectCardAsync

函数说明: 退卡。

## 2.4 cancelAsync

函数说明: 取消。

## 2.5 retainCardAsync

函数说明: 吞卡。

## 2.6 waitCardTakenAsync

函数说明: 等待卡片取走。

函数类型: 阻塞

## 2.7 getStatusAsync

函数说明: 获取设备状态。

返回信息:

```

{
  "AntiFraudModule": 0,
  "ParkingStationMedia": 0,
  "PowerSaveRecoveryTime": 0,
  "DevicePosition": 0,
  "BackImageModule": 0,
  "FrontImageModule": 0,
  "MagWriteModule": 0,
  "MagReadModule": 0,
  "ChipModule": 0,
  "GuidLights": [],
  "Extra": {},
  "ChipPower": 0,
  "Cards": 0,
  "Security": 0,
  "RetainBin": 0,
  "Media": 0,
  "Device": 0
}

```



}

字段说明：

字段	字段含义	备注
AntiFraudModule	反欺诈模块状态	0: 不支持; 1: 状态正常; 2: 设备不可用 3: 检测到外部设备 4: 状态未知
ParkingStationMedia	卡停留位置	1: 在卡口 2: 未在卡口 3: 卡住 4: 不支持 5: 未知 6: 正在进入 7: 锁住
PowerSaveRecoveryTime	从目前节电状态恢复到正常状态所需要的实际秒数	
DevicePosition	读卡器的位置	0: 读卡器处于正常操作位置 1: 读卡器被从它的正常操作位置移走 2: 由于硬件错误或者其它的情况，读卡器的位置不能确定 3: 实际设备没有能力检测其位置
BackImageModule	背面图像读取模块的状态	1: 模块处于良好的状态 2: 模块不可用 3: 模块的状态不能确定 4: 报告模块的状态不支持
FrontImageModule	正面图像读取模块的状态	同 BackImageModule
MagWriteModule	写磁模块的状态	同 BackImageModule
MagReadModule	读磁模块的状态	同 BackImageModule
ChipModule	芯片模块的状态	同 BackImageModule
GuidLights	指示灯的状态	指示灯状态数组 [[ "Status": "0", "Color": "0", ]] Status: 0: 指示灯不可用 1: 指示灯关闭 2: 指示灯开启 4: 灯在慢速闪动 8: 灯在中速闪动 16: 灯以快速闪动 128: 灯处于常亮状态 Color:

		0: 颜色不可用 256: 灯是红色 512: 灯是绿色 1024: 灯是黄色 2048: 灯是蓝色 4096: 灯是蓝绿色 8192: 灯是红紫色 16384: 灯是白色
Extra	厂商提供的特定信息或其它扩展信息	格式为对象，Key 值与厂商约定
ChipPower	IC 芯片的状态	0: 芯片存在，并通电，处于在线状态 1: 芯片存在，未通电 2: 芯片存在，并通电，但处于忙碌的状态 3: 卡在读卡器之中，但是卡上没有芯片 4: 芯片存在，但是由于硬件出错该芯片不可使用 5: 读卡器中没有 IC 芯片 6: 读卡器不支持报告芯片状态 7: 芯片的状态未知
Cards	吞卡数	
Security	读卡器内安全单元的状态	1: 没有可用的安全单元 2: 安全单元还未做好准备去处理卡或者它是不可操作的 3: 安全单元已打开并可操作
RetainBin	读卡器内回收盒的状态	1: 读卡器的回收盒未滿 2: 读卡器不支持回收卡功能 3: 读卡器的回收盒已滿 4: 读卡器的回收盒将滿 5: 读卡器的回收盒丢失
Media	读卡器内卡的状态	1: 卡在读卡器内 2: 卡不在读卡器内，也不在入卡口 3: 卡被夹住，需要操作员的干预 4: 卡状态不支持(例如刷卡式读卡器) 5: 卡的状态未知 6: 卡位于电动式读卡器的入卡口位置 7: 卡存在并被锁定。这表示可对卡的芯片进行操作
Device	读卡器的状态	0: 设备在线 1: 设备脱机 2: 设备断电或物理没连接 3: 设备不存在 4: 设备因硬件错误无法操作 5: 设备存在但有人正在阻止设备进行合适的操

		作 6: 设备忙当前不能处理指令 7: 设备存在但是检测到试图欺骗 8: 可能存在欺骗

### 2.8 isCardAtPresentAsync

函数说明: 卡是否在卡道内  
返回信息: Promise<boolean>

### 2.9 isCardAtEntranceAsync

函数说明: 卡是否在卡口  
返回信息: Promise<boolean>

### 2.10 isCardAtPresentOrAtEntranceAsync

函数说明: 卡是否在卡道内或者卡口  
返回信息: Promise<boolean>

### 2.11 powerUpAsync

函数说明: IC 卡上电

### 2.12 powerDownAsync

函数说明: IC 卡下电

### 2.13 pbocGetApplInfoByAIDAsync

函数说明: 通过 AID 获取应用信息  
输入参数: AID (应用 ID)  
返回信息:

```
{
  "DFName": null,
  "Label": null,
  "Priority": null,
  "PDOL": null,
  "Language": null,
  "BankCodeIndex": null,
  "PriorityName": null,
  "BankData": null,
  "AID": null,
  "LogEntry": null
}
```

字段	字段含义	备注
DFName	DF 名称	"A000000333010101"
Label	应用标签	应用名称需要转换显示成可读字符 "50424F43204445424954"
Priority	应用优先级	"01"
PDOL	待处理标签	应用初始化需要标签指定的数据 "9F7A019F02065F2A02"
Language	语言	"7A68"
BankCodeIndex	发卡行代码索引	"01"

PriorityName	应用优先级名称	"50424F43204445424954"
BankData	发卡行自定义数据	"9F4D020B0A"
AID	应用标识符	"A000000333010101"
LogEntry	交易日志入口	

#### 2.14 pbocGetApplInfoByDebitAIDAsync

函数说明：通过借记卡 ID 获取卡信息

返回信息：参考 2.12 返回信息

#### 2.15 pbocGetApplInfoByCreditAIDAsync

函数说明：通过贷记卡 ID 获取卡信息

返回信息：参考 2.12 返回信息

#### 2.16 pbocAppInitializeAsync

函数说明：通过所有参数初始化应用

输入参数：PDOL 值连接起来的字符串

返回信息：

```
{
  "AIP": {
    "AIP": "fAA=",
    "SAIP": "7C00",
    "IsSupportSDA": true,
    "IsSupportDDA": true,
    "IsSupportCardHolderVerification": true,
    "IsSupportTerminalRiskManagement": true,
    "IsSupportIssuingBankVerification": true,
    "IsSupportCDA": false
  },
  "AFLCollection": [
    {
      "SSFI": "08",
      "SFI": 8,
      "BeginRecordNum": 1,
      "EndRecordNum": 2,
      "OffLineAuthCount": 0
    },
    {
      "SSFI": "10",
      "SFI": 16,
      "BeginRecordNum": 1,
      "EndRecordNum": 4,
      "OffLineAuthCount": 1
    },
    {
      "SSFI": "20",
```

```

        "SFI": 32,
        "BeginRecordNum": 1,
        "EndRecordNum": 3,
        "OffLineAuthCount": 0
    },
    {
        "SSFI": "28",
        "SFI": 40,
        "BeginRecordNum": 1,
        "EndRecordNum": 1,
        "OffLineAuthCount": 0
    }
]
}

```

字段	字段含义	备注
AIP	应用交互特征	具体内容参照 AIP 字段定义
AFLCollection	文件定位器集合	具体内容参照 AFLCollection 字段定义

#### AIP 字段定义

字段	字段含义	备注
AIP	应用交互特征（原始值）	
SAIP	应用交互特征（字符串）	
IsSupportSDA	是否支持 SDA	
IsSupportDDA	是否支持 DDA	
IsSupportCardHolderVerification	是否支持持卡人认证	
IsSupportTerminalRiskManagement	是否支持终端风险管理	
IsSupportIssuingBankVerification	是否支持发卡行认证	
IsSupportCDA	是否支持 CDA	

#### AFLCollection 字段定义

字段	字段含义	备注
SSFI	短文件标识符（字符串）	
SFI	短文件标识符（原始值）	
BeginRecordNum	第 1 个要读出的记录号（不能为 0）	
EndRecordNum	最后一个要读出的记录号	
OffLineAuthCount	存放用于脱机数据认证的数据的连续记录个数	

#### 2.17 pbocAppInitializeByParamsAsync

方法说明：通过输入部分参数初始化应用

入参说明：

authAmount:授权金额，固定 12 位长度字符串，已分为单位。

tradeType:交易类型，常用类型：00:消费,01:现金,21:存款,31:余额,40:转账,60:圈存  
返回信息：参考 2.15

#### 2.18 pbocAppInitializeByDefaultParamsAsync

方法说明：通过默认参数初始化应用

返回信息：参考 2.15

#### 2.19 pbocReadAppDataAsync

函数说明：读取卡片数据

返回信息：

```
{
  "CardRawData": [...],
  "CardNumber": "6231250000008697",
  "CardSerial": "00",
  "Track2Data": "6231250000008697D221222091300000000000",
  "CDOL1": "9F02069F03069F1A0295055F2A029A039C019F37049F21039F4E14",
  "CDOL2": "8A029F02069F03069F1A0295055F2A029A039C019F37049F2103",
  "AppVersion": "0020",
  "AppExpiryDate": "221231",
  "AppEffectiveDate": "120916",
  "SAUC": "FF00",
  "AUC": {
    "B1b8_DomesticCash": true,
    "B1b7_InternationalCash": true,
    "B1b6_DomesticGoods": true,
    "B1b5_InternationalGoods": true,
    "B1b4_DomesticService": true,
    "B1b3_InternationalService": true,
    "B1b2_ATM": true,
    "B1b1_NonATM": true,
    "B2b8_DomesticCashBack": false,
    "B2b7_InternationalCashBack": false,
    "Value": "0000"
  },
  "SCVM": "000000000000000000002031F00",
  "CVM": {
    "AmountX": "00000000",
    "AmountY": "00000000",
    "CVMList": [
      {
        "Value": "0203",
        "Code": "02",
        "Condition": "03",
        "IsStandard": false,
        "IsContinueValidate": false,
        "CVMTType": 2
      }
    ]
  }
}
```

```

    },
    {
      "Value": "1F00",
      "Code": "1F",
      "Condition": "00",
      "IsStandard": false,
      "IsContinueValidate": false,
      "CVMType": 31
    }
  ]
},
"lacDefault": "2GCcqAA=",
"SlacDefault": "D8609CA800",
"lacReject": "ABAAAAA=",
"SlacReject": "0010000000",
"lacOnline": "2Gic+AA=",
"SlacOnline": "D8689CF800",
"AuthorizationCodeOfECIIssuer": null,
"IdatificationOfCardProduct": null,
"CountryCodeOfIssuerBank": "0156"
}

```

字段	字段含义	备注
CardRawData	卡原始数据	可以根据标签取值
CardNumber	卡号	
CardSerial	卡号序号	
Track2Data	等效二磁道信息	
CDOL1	动态数据对象列表 1	
CDOL2	动态数据对象列表 2	
AppVersion	应用版本号	
AppExpiryDate	应用失效日期	
AppEffectiveDate	应用生效日期	
SAUC	应用用途控制	
AUC	应用用途控制	参考 AUC 字段定义
SCVM	验证方法列表	
CVM	验证方法列表	参考 CVM 字段定义
lacDefault	发卡行行为代码-缺省	
SlacDefault	发卡行行为代码-缺省	
lacReject	发卡行行为代码-拒绝	
SlacReject	发卡行行为代码-拒绝	
lacOnline	发卡行行为代码-联机	
SlacOnline	发卡行行为代码-联机	
AuthorizationCodeOfECIIssuer	电子现金发卡行授权码	
IdatificationOfCardProduct	卡产品标识符	

CountryCodeOfIssuerBank	发卡行国家代码	

#### AUC 字段定义

字段	字段含义	备注
B1b8_DomesticCash	国内现金交易有效	
B1b7_InternationalCash	国际现金交易有效	
B1b6_DomesticGoods	国内商品有效	
B1b5_InternationalGoods	国际商品有效	
B1b4_DomesticService	国内服务有效	
B1b3_InternationalService	国际服务有效	
B1b2_ATM	ATM 有效	
B1b1_NonATM	非 ATM 有效	
B2b8_DomesticCashBack	国内现金交易有效	
B2b7_InternationalCashBack	国际现金交易有效	
Value	AUC 值	

#### CVM 字段定义

字段	字段含义	备注
AmountX	金额 X	
AmountY	金额 Y	
CVMList	验证方法列表	

#### CVMList 字段定义

字段	字段含义	备注
Value	CVM 值	
Code	CVM 代码	
Condition	CVM 条件	
IsStandard	是否规范值	
IsContinueValidate	是否继续验证	
CVMType	CVM 方法类型	<b>0: CVM 失败</b> <b>1: 卡片执行明文 PIN 核对</b> <b>2: 联机加密 PIN 验证</b> <b>3: 卡片执行明文 PIN 核对+签名（纸上）</b> <b>4: 签名（纸上）</b> <b>5: 无需 CVM</b> <b>6: 持卡人证件出示</b>

## 2.20 pbocGetDataAsync

方法说明：获取卡内标签值

输入参数：

p1: APDU 命令 p1 值



p2: APDU 命令 p2 值

返回信息: 标签值 (字符串)

2.21 pbocGetECashLimitAsync

方法说明: 获取电子现金限额

返回信息: 电子现金限额 (字符串)

2.22 pbocGetECashTradeLimitAsync

方法说明: 获取电子现金交易限额

返回信息: 电子现金交易限额 (字符串)

2.23 pbocGetECashBalanceAsync

方法说明: 获取电子现金余额

返回信息: 电子现金余额 (字符串)

2.24 pbocGetECashResetThresholdAsync

方法说明: 获取电子现金重置阈值

返回信息: 电子现金重置阈值 (字符串)

2.25 pbocGetECashOffLineBalanceAsync

方法说明: 获取电子现金脱机可用余额

返回信息: 电子现金脱机可用余额 (字符串)

2.26 pbocGetECashCVMLimitAsync

方法说明: 获取卡片 CVM 限额

返回信息: 卡片 CVM 限额 (字符串)

2.27 pbocStartTradeAsync

方法说明: 开始交易

返回信息: ARQC 或者 ACC 或者 TC (字符串)

2.28 pbocGenerateUnionpayField55Async

函数说明: 获取银联交易中 55 域信息

输入参数:

unionpayElements: 55 域可选字段列表

```
[
  {
    "Tag": "",
    "Value": ""
  }
]
```

字段	字段含义	备注
Tag	标签	
Value	值	

返回信息: 55 域信息 (字符串)

2.29 pbocGenerateUnionpayField55ByRequiredFieldAsync

函数说明: 获取银联交易中 55 域信息, 仅必填字段

返回信息: 55 域信息 (字符串)

2.30 pbocExternalAuthAsync

函数说明: 外部认证

### 2.31 pbocFinishTradeAsync

函数说明：结束交易，并执行脚本（如果入参中含有写卡脚本）

返回信息：ARQC 或者 ACC 或者 TC（字符串）

### 2.32 pbocExcuteScriptAsync

函数说明：执行脚本

输入信息：需要执行的脚本

### 2.33 pbocReadBinaryAsync

函数说明：读取二进制文件

输入信息：sif 短文件标识符

返回信息：文件内容

### 2.34 pbocUpdateBinaryAsync

函数说明：更新二进制文件

输入信息：更新的内容

### 2.35 pbocGetChallengeAsync

函数说明：获取随机数

输入参数：期望随机数长度（格式字符 2 位，例如 04）

返回信息：随机数

### 2.36 pbocGetChallengeOf4BAsync

函数说明：获取 4B 长度随机数

返回信息：随机数

### 2.37 pbocSelectByFileIdAsync

函数说明：根据文件 Id 选择应用

输入信息：文件 Id（例如昆山公交应用文件 Id 为：1001）

返回信息：文件信息

### 2.38 pbocSelectMFAsync

函数说明：选择 MF

返回信息：文件相关信息

### 2.39 pbocChipIOAsync

函数说明：使用原始指令操作 IC 卡

输入信息：CAPDU 请求命令

返回信息：RAPDU 响应数据

### 2.40 pbocGetBalanceAsync

函数说明：获取卡片余额

返回信息：余额（Int）

### 2.41 pbocInitForLoadAsync

函数说明：初始化圈存

输入信息：

amount：交易金额，以分为单位，自动左补 0 到 8 位长度

deviceNo：设备编号，自动左补 0 到 12 位长度

返回信息：

```
{  
    "CardAmount": ""  
    "TradeSequenceNo": ""  
    "KeyVersionsNo": ""  
}
```

```

        "AlgorithmId": ""
        "RandomNumber": ""
        "Mac1": ""
        "Value": ""
        "CardAmountInt": ""
    }

```

字段	字段含义	备注
CardAmount	卡片余额	
TradeSequenceNo	交易序号	
KeyVersionsNo	密钥版本号	
AlgorithmId	算法标识	
RandomNumber	随机数	
Mac1	MAC1	
Value	原始字符串值	
CardAmountInt	卡片余额(Int)	

## 2.42 pbocCreditForLoadAsync

函数说明：执行圈存

输入信息：

tradeDate: 交易日期  
tradeTime: 交易时间  
mac2: MAC2

返回信息：

```

{
    "Balance": ""
    "SequenceNo": ""
    "TradeAmount": ""
    "TradeType": ""
    "TerminalNo": ""
    "TradeDate": ""
    "TradeTime": ""
    "Value": ""
    "BalanceInt": 0
    "TradeAmountInt": 0
}

```

字段	字段含义	备注
Balance	新余额	
SequenceNo	交易序号	
TradeAmount	交易金额	
TradeType	交易类型	
TerminalNo	终端编号	
TradeDate	交易日期	
TradeTime	交易时间	
Value	原始圈存响应值	

BalanceInt	新余额(Int)	
TradeAmountInt	交易金额(Int)	

## 2.43 getCapsAsync

函数说明：获取设备支持能力信息

返回信息：

```
{
  "AntiFraudModule": false,
  "ParkingStation": 0,
  "PowerSaveControl": false,
  "EjectPosition": 0,
  "GuidLights": [],
  "MemoryChipProtocol": 0,
  "DIPMode": 0,
  "Extra": {},
  "ChipPower": 0,
  "WriteMode": 0,
  "ReadWriteAccessFollowingEject": false,
  "FluxSensorProgrammabl": false,
  "PowerOffOption": 0,
  "PowerOnOption": 0,
  "SecType": 0,
  "Cards": 0,
  "ChipProtocols": 0,
  "WriteTracks": 0,
  "ReadTracks": 0,
  "IsCompound": false,
  "Type": 0,
  "Class": 0
}
```

字段说明：

字段	字段含义	备注
AntiFraudModule	是否支持反欺诈模块	
ParkingStation	卡停留状态	1：处于可操作位置 2：没有在操作位置上 3：卡住 4：不支持 5：状态位置 6：正在进入 7：锁住
PowerSaveControl	是否支持节电模式	
EjectPosition	读卡器支持的退卡操作的位置	1：出口位置 2：内部传输装置
GuidLights	指示哪些指示灯可用	

MemoryChipProtocol	指定 SP 支持的记忆卡协议	1: Siemens4442 2: GPM896
DIPMode	刷卡式读卡器是否能在出口或入口读取磁道上的数据	0: 非刷卡式读卡器 0x0001: 未知是否可以读取 0x0002: 只能在出口读取磁道数据 0x0004: 只能在入口读取磁道数据 0x0008: 在出口和入口均可读取磁道数据
Extra	厂商提供的特定信息或其它扩展信息	
ChipPower	读卡器是否支持对 IC 卡芯片加电复位	0: 读卡器不支持对 IC 卡芯片加电复位 0x0002: 支持冷复位 0x0004: 支持热复位 0x0008: 读卡器支持对 IC 卡芯片掉电
WriteMode	写磁模式	0: 不支持写磁道 0x0002: 支持写低磁磁道 0x0004: 支持写高磁磁道 0x0008: SP 自动检测确定是写高磁磁道还是低磁磁道
ReadWriteAccessFollowingEject	卡退到取卡口时, 是否可以将卡重新吸入进行读卡或者写卡	
FluxSensorProgrammabl	读卡器上的磁通感应器是否是可用的	
PowerOffOption	断电设备操作类型	1: 开电时不对卡操作 2: 开电(断电) 时卡被退到入卡口 3: 开电(断电) 时卡被回收 4: 开电(断电) 时卡被退到入卡口, 在厂商规定的时间内卡没有被取走, 卡将被回收 5: 开电(断电) 时卡被移至读卡位置
PowerOnOption	开电设备操作类型	同 PowerOffOption。
SecType	安全单元的类型	1: 安全单元的类型 2: 读卡器的安全单元是 MMBBox 3: 读卡器的安全单元是 CIM86

Cards	回收盒可以容纳的最大卡张数	如果不可用则为 0
ChipProtocols	读卡器支持的 IC 芯片协议，可以是一个或多个	0: 不支持 IC 芯片协议 0x0001: 支持 T=0 协议 0x0002: 支持 T=1 协议 0x0004: 不需要应用程序来设置协议而可以与芯片卡通信 0x0008: TypeA Part3 协议 0x0010: TypeA Part4 协议 0x0020: TypeB 协议 0x0040: NFC 协议
WriteTracks	读卡器可以写入的磁道，可以是一个或多个	0: 不支持读磁道 0x0001: 可以读磁道 1 0x0002: 可以读磁道 2 0x0004: 可以读磁道 3 0x0008: IC 芯片 0x0010: 安全校验 0x0020: 无磁条 IC 芯片 0x0040: 内存芯片 0x0080: 可以读前磁道 1 0x0100: 读取卡的正面图像 0x0200: 读取卡的背面图像 0x0400: ID 卡 JIS1 Track1 0x0800: ID 卡 JIS1 Track3 0x4000: ID 卡动态数字识别的磁条 0x8000: 瑞典水印
ReadTracks	读卡器可以读取的磁道，可以是一个或多个	同 WriteTracks。
IsCompound	是否组合设备	
Type	读卡器类型	1: 电动式读卡器 2: 插卡式读卡器 3: 刷卡式读卡器 4: 非接触读卡器 5: 锁定直列式读卡器 6: 永久的荷载的芯片卡
Class	逻辑服务类	

#### 2.44 pbocStartTradeAsync

函数说明：开始交易（GAC 以及相关处理）

返回信息：

```
{
  "Value": "",
  "T9F27": "",
  "T9F36": "",
```

```

    "T9F26": "",
    "T9F10": "",
    "ArqcRawData": "",
}

```

字段	字段含义	备注
Value	ARQC、AAC 或者 TC	
T9F27	密文信息数据	10: ARQC
T9F36	应用交易计数器	
T9F26	应用认证密文	
T9F10	发卡行应用数据	
ArqcRawData	Arqc 原始数据	

## 2.45 pbocGetTerminalInfoAsync

函数说明：获取 pboc 终端信息

返回信息：

```

{
  "Version": "0030",
  "Type": {
    "HasWaiter": false,
    "Operator": 1,
    "NetworkingType": 1,
    "Value": "14"
  },
  "Performance": {
    "B1b8_IsSupportKeyboard": true,
    "B1b7_IsSupportMagneticStripeCards": true,
    "B1b6_IsSupportContactIC": true,
    "B2b8_IsSupportICCardPINVerification": false,
    "B2b7_IsSupportCryptPINOnlineVerification": true,
    "B2b6_IsSupportSignatureVerification": false,
    "B2b5_IsSupportCryptPINOfflineVerification": false,
    "B2b4_IsSupportNotRequiredCVM": false,
    "B2b1_IsSupportIDCardVerification": false,
    "B3b8_IsSupportSDA": false,
    "B3b7_IsSupportDDA": false,
    "B3b6_IsSupportRetainCard": false,
    "B3b4_IsSupportCDA": false,
    "Value": "E04000"
  },
  "AdditionalPerformance": {
    "B1b8_IsSupportCash": false,
    "B1b7_IsSupportGoods": true,
    "B1b6_IsSupportService": true,
    "B1b5_IsSupportCashback": true,

```

```

    "B1b4_IsSupportQuery": true,
    "B1b3_IsSupportTransfer": true,
    "B1b2_IsSupportPayment": true,
    "B1b1_IsSupportManage": true,
    "B2b8_IsSupportDepositTransaction": false,
    "B3b8_IsSupportDigitKey": true,
    "B3b7_IsSupportLetterAndSpecialCharacterKey": true,
    "B3b6_IsSupportCommandKey": true,
    "B3b4_IsSupportfunctionKey": true,
    "B4b8_IsSupportPrintToWaiter": true,
    "B4b7_IsSupportPrintToCardHolder": true,
    "B4b6_IsSupportDisplayToWaiter": true,
    "B4b5_IsSupportDisplayToCardHolder": true,
    "B4b2_IsSupportCodingSchedule10": false,
    "B4b1_IsSupportCodingSchedule9": false,
    "B5b8_IsSupportCodingSchedule8": false,
    "B5b7_IsSupportCodingSchedule7": false,
    "B5b6_IsSupportCodingSchedule6": false,
    "B5b5_IsSupportCodingSchedule5": false,
    "B5b4_IsSupportCodingSchedule4": false,
    "B5b3_IsSupportCodingSchedule3": false,
    "B5b2_IsSupportCodingSchedule2": false,
    "B5b1_IsSupportCodingSchedule1": false,
    "Value": "7F00E8F000"
  },
  "TradeDate": null,
  "Name": "0000000000000000000000000000000000000000",
  "TradeProperty": {
    "B1b7_IsSupportContactlessDebitCredit": true,
    "B1b6_IsSupportQPboc": true,
    "B1b5_IsSupportContactDebitCredit": true,
    "B1b4_IsSupportOffline": false,
    "B1b3_IsSupportOnlinePIN": true,
    "B1b2_IsSupportSignature": true,
    "B2b8_IsRequireOnlinePIN": true,
    "B2b7_IsRequireCVM": true,
    "Value": "76C00000"
  },
  "TVR": {
    "B1b8_OfflineDataAuthFailed": false,
    "B1b7_OfflineStaticDataAuthFailed": false,
    "B1b6_CardDataMiss": false,
    "B1b5_DataAbnormal": false,
    "B1b4_OfflineDynamicDataAuthFailed": false,

```



```

    "B1b3_DataAuthFailed": false,
    "B2b8_AppVersionDiff": false,
    "B2b7_ExpireApp": false,
    "B2b6_InvalidApp": false,
    "B2b5_InvalidService": false,
    "B2b4_NewCard": false,
    "B3b8_CardHolderVerifyFailed": false,
    "B3b7_UnknownCVM": false,
    "B3b6_PINNoRetryCount": false,
    "B3b5_PINRequireButDeviceNotWork": false,
    "B3b4_PINRequireButNotInput": false,
    "B3b3_PINRequireOnline": false,
    "B4b8_OverMinLimit": false,
    "B4b7_Limit": false,
    "B4b6_Limit": false,
    "B4b5_Limit": false,
    "B4b4_Limit": false,
    "B5b8_UseDefaultTDOL": false,
    "B5b7_IssuingBankVerificationFailed": false,
    "B5b6_ScriptExecuteFailedBeforeLastGAC": false,
    "B5b5_ScriptExecuteFailedAfterLastGAC": false,
    "Value": "0000000000"
  },
  "TSI": {
    "B1b8_OfflineDataAuthExecuted": false,
    "B1b7_CardHolderAuthExecuted": false,
    "B1b6_CardRiskManagementExecuted": false,
    "B1b5_IssuingBankVerificationExecuted": false,
    "B1b4_TerminalRiskManagementExecuted": false,
    "B1b3_ScriptExecuted": false,
    "Value": "0000"
  },
  "CVMResult": {
    "Value": "420300",
    "Code": "42",
    "Condition": "03",
    "Result": "00"
  },
  "AuthResponseCode": "5A31",
  "ScriptExcutedResultOfIssuerBank": null,
  "TacReject": "0000000000",
  "TacOnline": "0000000000",
  "TacDefault": "0000000000",
  "IsATM": false,

```

```
"AlgorithmFlag": "00"
}
```

字段说明：

字段	字段含义	备注
Version	终端应用版本	
Type	终端类型	
Performance	[9F33]终端性能	
AdditionalPerformance	[9F40]终端扩展性能	
TradeDate	[9A]交易日期	
Name	[9F4E]终端名称	
TradeProperty	[9F66]终端交易属性	
TVR	终端验证结果	
TSI	交易状态信息	
CVMResult	CVM 执行结果	
AuthResponseCode	[8A]授权响应码	
ScriptExcutedResultOfIssuerBank	发卡行脚本执行结果	
TacReject		
TacOnline		
TacDefault		
IsATM	指示终端是否 ATM	
AlgorithmFlag	算法标识	

### 3. FingerprintReader

#### 3.1 getSampleAsync

函数说明：获取指纹模版。

函数类型：阻塞。

返回信息：指纹模版信息。

#### 3.2 getFeatureImageAsync

函数说明：获取指纹图片。

函数类型：阻塞。

输入参数：

    ImagePath: 指纹图片保存路径。

返回信息：无

#### 3.3 getFeatureDataAsync

函数说明：获取指纹特征。

函数类型：阻塞。

返回信息：指纹特征数据（图像数据的 Base64 字符串）。

#### 3.4 isFeatureMatchAsync

函数说明：指纹特征是否匹配。

输入参数：

    templateData: 指纹模版数据。

#### 3.5 cancelAsync

函数说明：取消

#### 3.6 getStatusAsync

函数说明：获取设备状态

返回信息：

```
{
  "PowerSaveRecoveryTime": 0,
  "PTRDevicePosition": 3,
  "GuidLights": [
    {
      "Color": 0,
      "Status": 0
    }
  ],
  "Extra": {},
  "MediaOnStacker": 0,
  "RetractBins": [],
  "Lamp": 3,
  "Ink": 3,
  "Toner": 3,
  "Paper": [
    0,
    3,
    3
  ],
  "Media": 0,
  "Device": 0
}
```

字段说明：

字段	字段含义	备注
PowerSaveRecoveryTime	节电唤醒时间	
PTRDevicePosition	设备位置	0：设备处于正常操作位置 1：设备被从它的正常操作位置移走 2：由于硬件错误或者其它的情况，读卡器的位置不能确定 3：实际设备没有能力检测其位置
GuidLights	指示灯状态	指示灯状态数组 [{ "Status": "0", "Color": "0", }]
Extra	厂商自定义信息	
MediaOnStacker	堆放器上媒介的数量 仅适用于具有堆放能力的打印机	
RetractBins	回收盒信息	返回值为数组，代表多个回收盒 [

		{RetractBin :0, RetractCount :2}, {RetractBin :0, RetractCount :2} ] RetractBin 取值: 0: 正常 1: 满 2: 不支持 3: 未知 4: 满 5: 不在正常位置上
Lamp	灯状态	0: 灯正常 1: 必须更换灯泡 2: 灯状态不可用 3: 不支持 4: 未知
Ink	墨水状态	0: 满 1: 少 2: 用尽 3: 不支持 4: 未知
Toner	碳粉或墨水供应状态或打印色带的状态	0: 满 1: 少 2: 用尽 3: 不支持 4: 未知
Paper	纸张供应的情况	返回值为数组[0,0,0]，代表 3 个纸盒状态 0: 满 1: 少 2: 用尽 3: 不支持 4: 未知 5: 卡纸
Media	媒介状态	0: 媒介在打印位置 1: 媒介不在打印位置或堆放器上 2: 媒介卡在设备里 3: 不支持 4: 未知 5: 媒介位于设备的进/出口处 6: 媒介在复位操作过程中被回收
Device	设备状态	0: 设备在线 1: 设备脱机 2: 设备断电或物理没连接 3: 设备不存在 4: 设备因硬件错误无法操作

		<p>5: 设备存在但有人正在阻止设备进行合适的操作</p> <p>6: 设备忙当前不能处理指令</p> <p>7: 设备存在但是检测到试图欺骗</p> <p>8: 可能存在欺骗</p>
--	--	---

#### 4. ICCardReader

##### 4.1 readCardAsync

函数说明: 读取卡片。

函数类型: 阻塞。

返回信息: 卡号 (字符串)。

##### 4.2 cancelAsync

函数说明: 取消

```
{
    "PowerSaveRecoveryTime": 0,
    "PTRDevicePosition": 0,
    "GuidLights": [],
    "Extra": {},
    "MediaOnStacker": 0,
    "RetractBins": [],
    "Lamp": 0,
    "Ink": 0,
    "Toner": 0,
    "Paper": [],
    "Media": 0,
    "Device": 0
}
```

##### 4.3 getStatusAsync

同 2.7getStatusAsync。

##### 4.4 pboc 操作

pboc 相关操作同 CardReader2.11-2.42 的 pboc 操作。

##### 4.5 getCapsAsync

同 2.43 getCapsAsync。

#### 5. IDCardReader

##### 5.1 readIdCardAsync

函数说明: 读取身份证。

函数类型: 阻塞。

返回信息:

```
{
    "Name": null,
    "Sex": null,
    "Nation": null,
    "Birthday": null,
    "Address": null,
```

```

        "ID": null,
        "Department": null,
        "StartDate": null,
        "EndDate": null,
        "Reserve": null,
        "AppAddress1": null,
        "PhotoPath": null,
        "FrontImagePath": null,
        "BackImagePath": null
    }

```

字段	字段含义	备注
Name	姓名	
Sex	性别	字符串
Nation	民族	字符串
Birthday	生日	
Address	住址	
ID	证件号	
Department	发证机关	
StartDate	起始日期	
EndDate	终止日期	
Reserve	预留信息	
AppAddress1	预留地址	
PhotoPath	照片路径	
FrontImagePath	正面图片路径	
BackImagePath	背面图片路径	

## 5.2 ejectCard

函数说明：退出身份证。

## 5.3 cancel

函数说明：取消

## 5.4 waitIDCardTaken

函数说明：等待身份证被取走

函数类型：阻塞

## 5.5 getStatus

同 2.7 getStatus。

## 5.6 getCapsAsync

同 2.43 getCapsAsync。

## 6. PinKeypad

**特殊说明：**密码键盘一般不会直接调用，请使用密码控件。

### 6.1 getPinAsync

函数说明：密文模式调用起密码键盘。

函数类型：阻塞。

输入参数:

```
{
  "TerminateKeys": 3072,
  "TerminateFDKs": 0,
  "ActiveKeys": 16383,
  "ActiveFDKs": 0,
  "Echo": 0,
  "AutoEnd": true,
  "MaxLen": 6,
  "MinLen": 6
}
```

字段说明:

字段	字段含义	备注
TerminateKeys	终止按键掩码	所有终止按键或值
TerminateFDKs	终止功能键掩码	终止功能键或值
ActiveKeys	激活按键掩码	所有激活按键或值
ActiveFDKs	激活功能键掩码	所有激活功能键或值
Echo	回显字符	密文模式回显字符
AutoEnd	自动结束	达到最大按键长度是否自动结束输入， <b>true</b> :自动结束， <b>false</b> :不自动结束，需要按终止按键结束。
MaxLen	最大长度	输入最大长度
MinLen	最小长度	输入最小长度

按键说明:

按键	值	备注
0	0x00000001	
1	0x00000002	
2	0x00000004	
3	0x00000008	
4	0x00000010	
5	0x00000020	
6	0x00000040	
7	0x00000080	
8	0x00000100	
9	0x00000200	
ENTER	0x00000400	
CANCEL	0x00000800	
CLEAR	0x00001000	
BACKSPACE	0x00002000	
DECPOINT	0x00008000	小数点
00	0x00010000	
000	0x00020000	

返回信息： 按键信息

```
{
  "Completion": 0,
  "Digits": 0
}
```

字段说明：

字段	字段含义	备注
Completion	完成输入原因	0:自动结束 1:Enter 键 2:Cancel 键 6:继续输入 7:Clear 键 8:Backspace 键 9:FDK 10:Help 11:PFK 12:ContFDK
Digits	按键输入数量	

## 6.2 getDataAsync

函数说明： 明文模式调用起密码键盘。

函数类型： 阻塞。

输入参数：

```
{
  "TerminateKeys": 0,
  "TerminateFDKs": 0,
  "ActiveKeys": 0,
  "ActiveFDKs": 0,
  "AutoEnd": false,
  "MaxLen": 0
}
```

字段说明： 参考 6.1 字段说明。

返回信息：

```
{
  "Completion": 0,
  "Keys": [
    {
      "Key": 0,
      "Completion": 0
    },
    {
      "Key": 0,
      "Completion": 0
    }
  ]
}
```



```

    }
  ],
  "Count": 0
}

```

字段说明：

字段	字段含义	备注
Completion	按键原因	参考 6.1
Count	按键数量	
Keys	所有按键信息	

Keys 字段说明：

字段	字段含义	备注
Key	按键值	
Completion	按键原因	参考 6.1

### 6.3 waitKeyAsync

函数说明：等待密码键盘一个按键返回。

函数类型：阻塞。

返回信息：按键信息。

```

{
  "Key": 0,
  "Completion": 0
}

```

字段说明：参考 6.2 返回信息 Keys 字段说明。

### 6.4 getPinBlockAsync

函数说明：获取密文。

输入参数：

```

{
  "KeyEncKey": null,
  "Key": 'PinKey',
  "Format": 2,
  "Padding": 0,
  "XORData": null,
  "CustomerData": '6225888888888888'
}

```

字段	字段含义	备注
KeyEncKey	加密秘钥名称	取值 null
Key	秘钥名称	取值 PinKey
Format	PIN 格式	取值 2: FromAnsi
Padding	填充字符，范围 0x00 到 0x0F	取值 0
XORData	指定 PIN 块的格式	取值 null
CustomerData	客户数据	一般为卡号

返回信息：密文（字符串）。

#### 6.5 getPinBlockByCardNoAsync

函数说明：通过卡号获取密文

输入参数：

cardNo:卡号

返回信息：密文（字符串）。

#### 6.6 cancelAsync

函数说明：取消。

#### 6.7 getStatusAsync

函数说明：获取密码键盘状态

返回信息：

```
{
  "AntiFraudModule": 0,
  "PowerSaveRecoveryTime": 0,
  "PINDevicePosition": 0,
  "CertificateState": 0,
  "AutoBeepMode": 0,
  "GuidLights": [],
  "Extra": {},
  "EncStat": 0,
  "Device": 0
}
```

字段	字段含义	备注
AntiFraudModule	反欺骗模块	0: 不支持; 1: 状态正常; 2: 设备不可用 3: 检测到外部设备 4: 状态未知
PowerSaveRecoveryTime	节电唤醒时间	
PINDevicePosition	设备位置	0: 正常 1: 设备不在位置上 2: 位置未知 3: 未知状态不支持
CertificateState	公共的验证或加密 密钥在 PIN 证书模块 中的状态	0: 证书状态未知或设备没有此能力 1: 所有的预加载证书已经被加载 2: 一级校证书将不被接收, 并且只有二级 校证书才被接收 3: 证书模块没有准备好
AutoBeepMode	定义按键激活的时 候是否自动出声	1: 支持激活密钥按键的时候自动发出哔哔声 2: 密钥激活自动发声 3: 密钥未激活按键支持自动发声 4: 未激活密钥自动发声能够被应用控制

GuidLights	指示灯状态	指示灯状态数组 [[ "Status": "0", "Color": "0", ]]
Extra	厂商自定义信息	
EncStat	加密模块的状态	0: 加密模块已经初始化并准备好 1: 加密模块不可用或因硬件错误没准备好或通讯错误 2: 加密模块没有初始化(没有加载主密钥) 3: 加密模块忙(设备忙) 4: 加密模块状态未定义 5: 加密模块已经初始化 6: PIN 被篡改
Device	设备状态	0: 设备在线 1: 设备脱机 2: 设备断电或物理没连接 3: 设备不存在 4: 设备因硬件错误无法操作 5: 设备存在但有人正在阻止设备进行合适的操作 6: 设备忙当前不能处理指令 7: 设备存在但是检测到试图欺骗 8: 可能存在欺骗

#### 6.8 importMasterKeyAsync

函数说明: 导入主密钥

输入信息:

**masterKey**: 主密钥

返回信息: 校验值

#### 6.9 importMasterKeyExAsync

函数说明: 导入国密主密钥

输入信息:

**masterKey**: 主密钥

返回信息: 校验值

#### 6.10 importPinKeyAsync

函数说明: 导入工作密钥

输入信息:

**pinKey**: 工作密钥

返回信息: 校验值

#### 6.11 importPinKeyExAsync

函数说明: 导入国密工作密钥

输入信息:

**pinKey**: 工作密钥

返回信息：校验值

6.12 getPinBlockExAsync

函数说明：获取国密加密密文

输入信息：

```
{
    "Algorithm":4096,
    "KeyEncKey": null,
    "Key": 'PinKey',
    "Format": 2,
    "Padding": 0,
    "XORData": null,
    "CustomerData": '6225888888888888'
}
```

参数说明：

字段	字段含义	备注
Algorithm	算法	1: DESECB 2: DESCBC 4: DESCFB 8: RSA 16: ECMA 32: DESMAC 64: TriDESECB 128: TriDESCBC 256: TriDESCFB 512: TriDesMac 1024: MAAMac 4096: SM4(国密) 8192: SM4Mac
KeyEncKey	加密秘钥名称	取值 null
Key	秘钥名称	取值 PinKey
Format	PIN 格式	取值 2: FromAnsi
Padding	填充字符，范围 0x00 到 0x0F	取值 0
XORData	指定 PIN 块的格式	取值 null
CustomerData	客户数据	一般为卡号

返回信息：密文信息（字符串）。

6.13 getPinBlockExByCardNoAsync

函数说明：通过卡号获取国密密文

输入参数：

cardNo:卡号

返回信息：密文（字符串）。

6.14 cryptAsync

函数说明：通过密码键盘进行加解密

输入参数：

```

{
  "Algorithm": 4096
  "Compression": 0,
  "CryptData": "",
  "Key": 'MacKey',
  "KeyEncKey": null,
  "Mode": 1,
  "Padding": 0,
  "StartValue": null,
  "StartValueKey": null,
  "CryptDataEncoding": 'GB2312'
}

```

入参说明：

字段	字段含义	备注
Algorithm	算法	1: DESECB 2: DESCBC 4: DESCFB 8: RSA 16: ECMA 32: DESMAC 64: TriDESECB 128: TriDESCBC 256: TriDESCFB 512: TriDesMac 1024: MAAMac 4096: SM4(国密) 8192: SM4Mac
Compression	建立 MAC 之前数据是否被压缩	False: 未压缩, True: 压缩
CryptData	需要加密的数据	
Key	秘钥名称	
KeyEncKey	加密秘钥名称	
Mode	加解密模式	1: 加密 2: 解密 3: 取 8 字节随机数
Padding	填充字符, 范围 0x00 到 0x0F	取值 0
StartValue	初始化向量	一般取值 null
StartValueKey	解密初始向量秘钥名称	一般取值 null
CryptDataEncoding	CryptData 的字符编码	GB2312 UTF8 ..... HEX:16 进制编码

返回信息：Hex 字符串

## 6.15 importKeyAsync

函数说明：导入密钥

输入参数：

```
{
  "EncKey": "MasterKey"
  "Ident": null,
  "Key": "PinKey",
  "Use": 3,
  "Value": "",
}
```

入参说明：

字段	字段含义	备注
EncKey	加密密钥名称	
Ident	密钥所有者身份	一般取值 null
Key	密钥名称	
Use	密钥用途	0: Delete 1: Crypt 2: Function 4: MACing 32: KeyEncKey 64: NoDuplicate 128: SvEncKey 256: Construct 512: SecureConstruct 1024: ANSTR31Master
Value	密密钥值	

返回信息：密钥校验值

6.16 importKeyExAsync

函数说明：导入密钥扩展方法，一般用于国密密钥导入

输入参数：

```
{
  "ControlVector": null
  "EncKey": "MasterKey"
  "Key": "PinKey",
  "KeyCheckModes": 0,
  "KeyCheckValue": null,
  "Use": 3,
  "Value": "",
}
```

入参说明：

字段	字段含义	备注
ControlVector	加载密钥控制向量	
EncKey	加密密钥名称	
Key	密钥名称	

KeyCheckModes	秘钥检查模式	0: 不检查 1: Self 2: Zero
KeyCheckValue	秘钥校验值	
Use	秘钥用途	0: Delete 1: Crypt 2: Function 4: MACing 32: KeyEncKey 64: NoDuplicate 128: SvEncKey 256: Construct 512: SecureConstruct 1024: ANSTR31Master
Value	密秘钥值	

返回信息:

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明:

字段	字段含义	备注
ExcuteResult	函数执行结果	0: 函数成功执行
IsSuccess	函数执行是否成功	true: 执行成功 false: 执行失败

#### 6.17 importRsaPublicKeyExAsync

函数说明: 导入 Rsa 或者 SM2 秘钥

输入参数:

```
{
  "Key": "MasterKey"
  "RSASignatureAlgorithm": 0,
  "SigKey": 0,
  "Signature": null,
  "Use": 3,
  "Value": "",
}
```

字段说明:

字段	字段含义	备注
Key	秘钥名称	
RSASignatureAlgorithm	秘钥签名的算法	0: Na 1: SSA_PKCS_V1_5 2: SSA_PSS
SigKey	秘钥签名的秘钥名称	
Signature	导入秘钥的签名	

Use	秘钥用途	0: Delete 1: Crypt 2: Function 4: MACing 32: KeyEncKey 64: NoDuplicate 128: SvEncKey 256: Construct 512: SecureConstruct 1024: ANSTR31Master
Value	密秘钥值	

## 7. ReceiptPrinter

**特殊说明：**凭条打印机一般交易不会直接调用，请使用扩展调用。

### 7.1 printAsync

函数说明：打印凭条。

输入参数：

**data：**需要打印的数据。

输入示例：

“\\H 赞同科技测试凭条\\L 第一行测试数据\\L 第二行打印数据\\L 第三行打印数据”

指令说明：

指令	含义	备注
\\H	凭条头	后面文本为凭条头
\\L	新行	后面文本为凭条一行内容

### 7.2 getStatusAsync

同 3.6getStatusAsync。

### 7.3 扩展调用说明

扩展类位置：src\utils\receipt-printer.js

引入方式：import ReceiptPrinter from '../utils/receipt-printer';

打印函数：print

函数入参：

**data:**字符串数组对象，每个字符串会在凭条上打印一行。

调用示例：

```
ReceiptPrinter.print(["交易时间：2017/03/18", "服务机构：赞同科技", "客户姓名：张三", "卡号：1234567890123456", "交易名称 交易结果", "开卡 成功"]);
```

可以在此类中定义各种打印格式的小票，方便交易调用。

### 7.4 printByNameAsync(data, formName, mediaName)

函数说明：使用指定 form 打印凭条

调用示例：

```
printByNameAsync("\\H 赞同科技测试凭条\\L 第一行测试数
```



据","ReceiptForm","ReceiptMedia");

#### 7.5 printByRawFromAsync(data, formName, mediaName)

函数说明：使用指定 form 打印凭条

调用示例：

```
printByFromNameAsync("header=赞同科技测试凭条,Line1=第一行测试数据,Line2=
第二行测试数据","ReceiptForm","ReceiptMedia");
```

## 8. UKeyDispenser

### 8.1 getCardUnitInfoAsync

函数说明：获取卡箱信息。

返回信息：所有卡箱信息。

输出示例：

```
[
  {
    "HardwareSensor": false,
    "Status": 0,
    "Threshold": 0,
    "RetainCount": 0,
    "Count": 0,
    "InitialCount": 0,
    "Type": 0,
    "CardName": null,
    "Number": 0
  },
  {
    "HardwareSensor": false,
    "Status": 0,
    "Threshold": 0,
    "RetainCount": 0,
    "Count": 0,
    "InitialCount": 0,
    "Type": 0,
    "CardName": null,
    "Number": 0
  }
]
```

字段说明：参考 1.1 字段说明。

### 8.2 setCardUnitInfoAsync

函数说明：设置卡箱信息。

输入参数：

**cardUnitInfo**：卡箱信息数组

输入示例：

```
[
  {
```

```

        "HardwareSensor": false,
        "Status": 0,
        "Threshold": 0,
        "RetainCount": 0,
        "Count": 0,
        "InitialCount": 0,
        "Type": 0,
        "CardName": null,
        "Number": 0
    },
    {
        "HardwareSensor": false,
        "Status": 0,
        "Threshold": 0,
        "RetainCount": 0,
        "Count": 0,
        "InitialCount": 0,
        "Type": 0,
        "CardName": null,
        "Number": 0
    }
]

```

字段说明：参考 1.1 字段说明。

### 8.3 dispenseCardAsync

函数说明：发卡。

输入参数：

**dispense**：发卡控制信息

输入示例：

```

{
    "Present": false,
    "Number": 0
}

```

字段说明：参考 1.3 字段说明。

### 8.4 ejectCardAsync

函数说明：退出卡片。

### 8.5 cancelAsync

函数说明：取消，用来取消阻塞的操作。

### 8.6 waitUKeyTakenAsync

函数说明：等待 Ukey 被取走

函数类型：阻塞

### 8.7 retainCardAsync

函数说明：吞卡

输入信息：

**cardBoxNumber**：吞卡卡箱号

9. UKeyReader

9.1 readAsync

函数说明：读取 Ukey 盒子上的二维码/条码  
返回信息：二维码/条码数据（字符串）。

9.2 cancelAsync

函数说明：执行取消操作。

9.3 waitUKeyTakenAsync

函数说明：等待 Ukey 被取走  
函数类型：阻塞

9.4 getStatusAsync

同 2.7getStatusAsync。

9.5 retainCardAsync

函数说明：吞卡

10. Barcode

10.1 readAsync

函数说明：读取二维码/条码  
函数返回：二维码/条码数据（字符串）。

10.2 cancelAsync

函数说明：取消读取二维码

10.3 getStatusAsync

函数说明：获取二维码扫描设备的状态  
返回信息：

```
{
    "AntiFraudModule": 0,
    "PowerSaveRecoveryTime": 0,
    "BcrDevicePosition": 0,
    "Extra": {},
    "GuidLights": [],
    "BCRScanner": 0,
    "Device": 0
}
```

字段	字段含义	备注
AntiFraudModule	反欺骗模块	0: 不支持; 1: 状态正常; 2: 设备不可用 3: 检测到外部设备 4: 状态未知
PowerSaveRecoveryTime	节电唤醒时间	
GuidLights	指示灯状态	指示灯状态数组 [{ "Status": "0", "Color": "0", }]

Extra	厂商自定义信息	
BCRScanner	二维码扫描模块状态	0: 打开 1: 关闭 3: 不可用 4: 未知
Device	设备状态	0: 设备在线 1: 设备脱机 2: 设备断电或物理没连接 3: 设备不存在 4: 设备因硬件错误无法操作 5: 设备存在但有人正在阻止设备进行合适的操作 6: 设备忙当前不能处理指令 7: 设备存在但是检测到试图欺骗 8: 可能存在欺骗

## 11. StampPrinter

### 11.1 printAsync

函数说明：打印盖章

入参示例：

“PaperNum=3;PrintType=4;Stamp=1; File[0]=c:\\A4\_1.htm; WaitNum=1”

参数说明：

字段	字段含义	备注
PaperNum	打印张数	
PrintType	打印类型	
Stamp	是否盖章	1 为盖章，0 为不盖章
File[0]	打印文件路径	可以传多个文件，数组下标从 0 开始。
WaitNum	等待打印张数	WaitNum=1 等待打印 1 张完成后继续执行。注要针对批量打印。

### 11.2 cancelAsync

函数说明：取消阻塞操作

### 11.3 getStatusAsync

同 3.7getStatus。

### 11.4 ejectAsync

函数说明：开始执行打印指令

### 11.5 controlMediaAsync(ctrls)

函数说明：控制媒介

输入参数：

ctrls: 控制媒介枚举类型，参数可以进行与操作进行多个指令控制输入

入参说明：

值	含义	备注
0	Invalid	无含义指令
1	Eject	弹出媒介

2	Perforate	在媒介上打孔
4	Cut	切割媒介
8	Skip	跳过黑标
16	Flush	刷新缓存，根据请求打印
32	Retract	回收媒介
64	Stack	将媒介堆放整齐
128	PartialCut	部分切割
256	Alarm	发出警报
512	AtpForward	向前翻一页
1024	AtpBackword	向后翻一页
2048	TurnMedia	翻转媒介
4096	Stamp	盖章
8192	Park	将文档放入纸盒
16384	Expel	将媒介从出口吐出来
32768	EjectToTransport	将媒介移动到传送装置

## 12. Sensors

### 12.1 openCardUnitGuidLightAsync

函数说明：打开读卡器指示灯

### 12.2 closeCardUnitGuidLightAsync

函数说明：关闭读卡器指示灯

### 12.3 openPinPadGuidLightAsync

函数说明：打开密码键盘灯

### 12.4 closePinPadGuidLightAsync

函数说明：关闭密码键盘灯

### 12.5 openNotesDispenserGuidLightAsync

函数说明：打开纸币出钞单元指示灯

### 12.6 closeNotesDispenserGuidLightAsync

函数说明：关闭纸币出钞单元指示灯

### 12.7 openSPCardReaderGuidLightAsync

函数说明：打开刷卡器指示灯

### 12.8 closeSPCardReaderGuidLightAsync

函数说明：关闭刷卡器指示灯

### 12.9 openReceiptPrinterGuidLightAsync

函数说明：打开凭条指示灯

### 12.10 closeReceiptPrinterGuidLightAsync

函数说明：关闭凭条指示灯

### 12.11 openPassbookPrinterGuidLightAsync

函数说明：打开存折打印机指示灯

### 12.12 closePassbookPrinterGuidLightAsync

函数说明：关闭存折打印机指示灯

### 12.13 openUKeyGuidLightAsync

函数说明：打开 Ukey 指示灯

### 12.14 closeUKeyGuidLightAsync

函数说明：关闭 Ukey 指示灯

#### 12.15 openChequeUnitGuidLightAsync

函数说明：打开支票处理单元指示灯

#### 12.16 closeChequeUnitGuidLightAsync

函数说明：关闭支票处理单元指示灯

#### 12.17 openRFCardGuidLightAsync

函数说明：打开射频卡指示灯

#### 12.18 closeRFCardGuidLightAsync

函数说明：关闭射频卡指示灯

#### 12.19 openBillAcceptorGuidLightAsync

函数说明：打开货币出口指示灯

#### 12.20 closeBillAcceptorGuidLightAsync

函数说明：关闭货币出口指示灯

#### 12.21 openDocumentPrinterGuidLightAsync

函数说明：打开文档打印机指示灯

#### 12.22 closeDocumentPrinterGuidLightAsync

函数说明：关闭文档打印指示灯

#### 12.23 openIDCardGuidLightAsync

函数说明：打开身份证设备指示灯

#### 12.24 closeIDCardGuidLightAsync

函数说明：关闭身份证设备指示灯

#### 12.25 openScannerGuidLightAsync

函数说明：打开扫描仪指示灯

#### 12.26 closeScannerGuidLightAsync

函数说明：关闭扫描仪指示灯

#### 12.27 openFingerGuidLightAsync

函数说明：打开指纹仪指示灯

#### 12.28 closeFingerGuidLightAsync

函数说明：关闭指纹仪指示灯

#### 12.29 openSignatureGuidLightAsync

函数说明：打开签名设备指示灯

#### 12.30 closeSignatureGuidLightAsync

函数说明：关闭签名设备指示灯

#### 12.31 openHRCameraGuidLightAsync

函数说明：打开高拍仪指示灯

#### 12.32 closeHRCameraGuidLightAsync

函数说明：关闭高拍仪指示灯

### 13. PassbookPrinter

#### 13.1 printAsync

函数说明：打印存折数据

输入信息：

**data:**打印数据数组，每个数据项代表一行数据各个字段用|分割。

EmptyLineCount: 从顶部到打印内容的空行数

打印示例: `print(["F1|F2|F3|F4|F5","F1|F2|F3|F4|F5","F1|F2|F3|F4|F5"], 10)`

13.2 `printWithEmptyLinesAsync`

函数说明: 打印存折数据

data:打印数据数组, 每个数据项代表一行数据各个字段用|分割。

打印示例: `printWithEmptyLines(["","","","F1|F2|F3|F4|F5","F1|F2|F3|F4|F5"])`

13.3 `readFormAsync`

函数说明: 读取存折磁条信息

返回信息:

```
{
  "Track2": "1234567890",    //二磁道信息
  "Track3": "1234567890"    //三磁道信息
}
```

13.4 `cancelAsync`

函数说明: 取消阻塞操作

13.5 `getStatusAsync`

同 3.7`getStatusAsync`。

14. `SignatureTablet`

14.1 `getSignatureAsync`

函数说明: 获取签名数据

返回信息:

```
{
  "SignatureData": ""
  "SignatureImage": ""
}
```

字段	字段含义	备注
SignatureData	签名数据	文本格式
SignatureImage	签名图片	Base64 文本格式

14.2 `cancleSignatureAsync`

函数说明: 取消签名

15. `CashIn`

15.1 `LockAsync()`

函数说明: 锁定操作

15.2 `unlockAsync()`

函数说明: 解锁操作

15.3 `cancelAsync()`

函数说明: 取消操作

15.4 `getStatusAsync()`

函数说明: 获取设备状态

返回信息:

```
{
  "AntiFraudModule": 0,
  "MixedMode": 0,
  "PowerSaveRecoverTime": 0,
  "DevicePosition": 0,
  "GuidLights": [],
  "Extra": {
    "ErrorCode": "2202010200",
    "ErrErrDescribe": ""
  },
  "Positions": [
    {
      "TransportStatus": 0,
      "Transport": 0,
      "PositionStatus": 1,
      "Shutter": 4,
      "Position": 32
    },
    {
      "TransportStatus": 0,
      "Transport": 0,
      "PositionStatus": 1,
      "Shutter": 4,
      "Position": 8192
    }
  ],
  "DropBox": false,
  "BanknoteReader": 0,
  "StackerItems": 4,
  "IntermediateStacker": 0,
  "Acceptor": 0,
  "SafeDoor": 1,
  "Device": 0
}
```

字段说明:

字段	字段含义	备注
AntiFraudModule	反欺诈模块状态	0: 不支持; 1: 状态正常; 2: 设备不可用 3: 检测到外部设备 4: 状态未知
MixedMode	混合模式	0: 混合媒体模式不支持的设备或没有被激活; 1: 使用 CIM 和 IPM 接口被激活



PowerSaveRecoverTime	指定设备从当前节电模式恢复正常工作状态所需的实际秒数	如果没有激活省电模式或不支持省电控制，则此值为 0。
DevicePosition	读卡器的位置	0: 读卡器处于正常操作位置 1: 读卡器被从它的正常操作位置移走 2: 由于硬件错误或者其它的情况，读卡器的位置不能确定 3: 实际设备没有能力检测其位置
GuidLights	指示灯状态	指示灯状态数组 [ { “Status” : ” 0” , “Color” : ” 0” , } ] Status: 0: 指示灯不可用 1: 指示灯关闭 2: 指示灯开启 4: 灯在慢速闪动 8: 灯在中速闪动 16: 灯以快速闪动 128: 灯处于常亮状态 Color: 0: 颜色不可用 256: 灯是红色 512: 灯是绿色 1024: 灯是黄色 2048: 灯是蓝色 4096: 灯是蓝绿色 8192: 灯是红紫色 16384: 灯是白色
Extra	厂商提供的特定信息或其它扩展信息	这些信息作为一系列“键=值”字符串返回
Positions	输入输出位置信息集合	参考 Positions 字段说明
DropBox	指示废弃盒子是否有东西	
BanknoteReader	纸币读取设备状态	0: 验钞器状态良好 1: 验钞器不可操作 2: 硬件错误或其它情况导致验钞器状态无法确定 3: 物理设备没有验钞器
StackerItems	中间栈用户接入状态	0: 在中间堆栈的物件已经到过客户通道 如果是一个现金循环机，物件在那里可能是之前取款操作的结果 1: 在中间堆栈的物件未到客户通道 2: 不知道在中间堆栈的物件是否到过客户通道 4: 中间堆栈没有物件，或者物理设备没有中间堆栈

IntermediateStacker	中间栈状态	0:中间堆栈空 1:中间堆栈非空 2:中间堆栈满 4: 由于硬件错误或其他情况，无法确定中间堆栈的状态 5: 物理设备没有中间堆栈
Acceptor	接受钞箱状态	0: 所有现金单元存在且状态良好 1: 有一个存在的现金单元处于一个不正常的状态。接收器可操作 2: 由于现金单元故障，不可能接收。 接收器可以操作，但不接收任何物件 3: 由于硬件错误或其他情况，无法确定现金单元的状态
SafeDoor	保险箱门状态	1:物理设备没有保险箱门，或保险箱门状态不支持报告 2:保险箱门打开 3:保险箱门关闭 4:由于硬件错误或其他情况，保险箱门无法确定
Device	设备状态	0:设备在线 1: 设备脱机 2: 设备断电或物理没连接 3: 设备不存在 4: 设备因硬件错误无法操作 5: 设备存在但有人正在阻止设备进行合适的操作 6: 设备忙当前不能处理指令 7: 设备存在但是检测到试图欺骗 8: 可能存在欺骗

Positions 字段说明:

字段	字段含义	备注
TransportStatus	传送装置上物品状态	0:传输为空 1:传输非空，物件还没进入客户通道 2:客户接触过的物件在传输机构上 3:由于硬件错误或其它情况导致无法知道是否有物件在传输上 4:设备没有能力报告是否有物件在传输上
Transport	传送装置状态	0:传输机构状态良好 1:由于硬件故障或介质卡住导致传输不可操作 2:由于硬件错误或其它情况导致传输状态无法确定 3:物理设备没有传输或不能报告传输状态
PositionStatus	输出输出位置状态	0:该位置空 1:该位置非空 2:由于硬件错误或其它情况导致该位置状态无法确定

		3:设备没有能力报告物件是否在该位置 4:该位置检测到不相关的物件
Shutter	钞门状态	0: 钞门关闭 1: 钞门打开 2:钞门被卡住 3:因硬件错误和其它情况，无法确定钞门状态 4:物理设备没有钞门，或不支持报告钞门状态
Position	输入输出位置信息	0: 没有输入输出位置 1: 左输入位置 2: 右输入位置 4: 中间输入位置 8: 上输入位置 16: 下输入位置 32: 前输入位置 64: 后输入位置 128: 左输出位置 256: 右输出位置 512: 中间输出位置 1024: 上输出位置 2048: 下输出位置 4096: 前输出位置 8192: 后输出位置

### 15.5 getCapabilitiesAsync()

函数说明：获取设备支持能力

返回信息：

```

{
  "AntiFraudModule": false,
  "MixedDepositAndRollback": false,
  "MixedMode": 0,
  "DeviceLockControl": false,
  "CountActions": 0,
  "CashInLimit": 0,
  "Replenish": false,
  "PowerSaveControl": false,
  "CompareSignatures": false,
  "ItemInfoTypes": 0,
  "GuidLights": [],
  "Extra": {},
  "RetractStackerActions": 8,
  "RetractTransportActions": 2,
  "RetractAreas": 1,
  "ExchangeType": 1,
  "Positions": 8224,

```

```

    "ItemsInsteredSensor": false,
    "ItemsTakenSensor": true,
    "IntermediateStacker": 0,
    "Refill": false,
    "CashBox": false,
    "SafeDoor": false,
    "ShutterControl": true,
    "Shutter": false,
    "Compound": true,
    "MaxCashInItems": 5000,
    "Type": 1,
    "Class": 13
}

```

字段说明：

字段	字段含义	备注
AntiFraudModule	反欺诈模式	
MixedDepositAndRollback	是否支持回滚	
MixedMode	混合模式	0: 混合媒体模式不支持的设备或没有被激活 1:使用 CIM 和 IPM 接口被激活
DeviceLockControl	设备或者钞箱是否支持物理锁控制	
CountActions	指定 Count 命令	0: 不支持 1:个体 2:所有
CashInLimit	现金限制模式	0:不支持 1:通过数量限制 2:通过总额限制
Replenish	指示补充命令是否可用 WFS_INF_CIM_REPLENISH_TARGET WFS_CMD_CIM_REPLENISH	
PowerSaveControl	是否支持节电模式	
CompareSignatures	指示 SP 是否支持签名比对	
ItemInfoTypes	指定可以被检索的信息类型	1:物件的序列号 2:物件的 P6 签名
GuidLights	指示灯状态	指示灯状态数组 [[ "Status": "0", "Color": "0", ]] Status: 0: 指示灯不可用 1: 指示灯关闭 2: 指示灯开启

		<p>4: 灯在慢速闪动</p> <p>8: 灯在中速闪动</p> <p>16: 灯以快速闪动</p> <p>128: 灯处于常亮状态</p> <p>Color:</p> <p>0: 颜色不可用</p> <p>256: 灯是红色</p> <p>512: 灯是绿色</p> <p>1024: 灯是黄色</p> <p>2048: 灯是蓝色</p> <p>4096: 灯是蓝绿色</p> <p>8192: 灯是红紫色</p> <p>16384: 灯是白色</p>
Extra	厂商自定义信息	这些信息作为一系列“键=值”字符串返回
RetractStackerActions	回收到中间栈动作	<p>1: 物件可转移到出口位置</p> <p>2: 物件可被回收到回收现金单元</p> <p>4: CIM 没有从堆栈回收的能力</p> <p>8: 物件可被回收到拒绝现金单元</p> <p>16: 物件可被回收到现金钞箱</p>
RetractTransportActions	回收到传送装置动作	<p>1: 物件可转移到出口位置</p> <p>2: 物件可被回收到回收现金单元</p> <p>4: CIM 没有从堆栈回收的能力</p> <p>8: 物件可被回收到拒绝现金单元</p> <p>16: 物件可被回收到现金钞箱</p>
RetractAreas	指定哪些项目可以被回收	<p>1: 物件可被回收到回收钞箱</p> <p>2: 物件可被回收到传输</p> <p>4: 物件可被回收到中间堆栈</p> <p>8: 物件可被回收到物件盒, 即存款和循环钞箱</p> <p>16: CIM 没有能力从此位置回收</p> <p>32: 物件可被回收到拒绝钞箱</p>
ExchangeType	设备支持的钞箱交换类型	<p>1: 钞箱将被手工补给</p> <p>2: 物件将从补给容器转移到钞箱</p> <p>4: 支持从循环现金单元被转移到钞箱或输出位置</p> <p>8: 物件将从存款入口被转移到钞箱</p>
Positions	指示可用的输出输出位置	<p>0: 没有输入输出位置</p> <p>1: 左输入位置</p> <p>2: 右输入位置</p> <p>4: 中间输入位置</p> <p>8: 上输入位置</p> <p>16: 下输入位置</p> <p>32: 前输入位置</p>

		64: 后输入位置 128:左输出位置 256: 右输出位置 512:中间输出位置 1024:上输出位置 2048:下输出位置 4096: 前输出位置 8192: 后输出位置
ItemsInsteredSensor	是否支持插入传感器	
ItemsTakenSensor	是否支持取走传感器	
IntermediateStacker	指示中间栈可以存放的数量	
Refill	字段有没有使用	
CashBox	柜员 是否分配一个现金箱	
SafeDoor	是否支持安全门	
ShutterControl	钞门是否隐含被 SP 控制	
Shutter	是否支持钞门	
Compound	是否组合设备	
MaxCashInItems	可以存入的最大数量	
Type	服务类型	0: 柜员纸币接收器 1:自助服务纸币接收器 2: 柜员硬币接收器 3:自助服务硬币接收器
Class	指示逻辑服务类	

### 15.6 getCashUnitInfoAsync()

函数说明：获取现金箱信息

返回信息：

```
{
  "CashIn": [
    {
      "Minimum": 0,
      "RejectCount": 0,
      "RetractedCount": 0,
      "PresentedCount": 0,
      "DispensedCount": 0,
      "InitialCount": 0,
      "CashUnitName": null,
      "CDMType": 0,
      "NoteIds": [],
      "Extra": {},
      "Physical": [
        {
          "RejectCount": 0,
          "RetractedCount": 0,
```

```

        "PresentedCount": 0,
        "DispensedCount": 0,
        "InitialCount": 0,
        "Extra": {},
        "HardwareSensors": true,
        "PStatus": 0,
        "Maximum": 3000,
        "Count": 0,
        "CashInCount": 0,
        "UnitID": "PHTRA",
        "PhysicalPositionName": "PHTRA"
    }
],
"NumPhysicalCUs": 1,
"NoteNumberList": {
    "NoteNumber": [],
    "NumOfNoteNumbers": 0
},
"AppLock": false,
"Status": 0,
"Maximum": 3000,
"Count": 0,
"CashInCount": 0,
"Values": 0,
"CurrencyID": "CNY",
"UnitID": "LOG1",
"ItemType": 1,
"Type": 4,
"Number": 1
},
{
    "Minimum": 0,
    "RejectCount": 0,
    "RetractedCount": 0,
    "PresentedCount": 0,
    "DispensedCount": 0,
    "InitialCount": 0,
    "CashUnitName": null,
    "CDMType": 0,
    "NoteIDs": [],
    "Extra": {},
    "Physical": [
        {
            "RejectCount": 0,

```

```

        "RetractedCount": 0,
        "PresentedCount": 0,
        "DispensedCount": 0,
        "InitialCount": 0,
        "Extra": {},
        "HardwareSensors": true,
        "PStatus": 0,
        "Maximum": 3000,
        "Count": 0,
        "CashInCount": 0,
        "UnitID": "PH1",
        "PhysicalPositionName": "PhyPos1"
    },
    ],
    "NumPhysicalCUs": 1,
    "NoteNumberList": {
        "NoteNumber": [
            {
                "Count": 0,
                "NoteID": 7
            }
        ],
        "NumOfNoteNumbers": 1
    },
    },
    "AppLock": false,
    "Status": 0,
    "Maximum": 3000,
    "Count": 0,
    "CashInCount": 0,
    "Values": 100,
    "CurrencyID": "CNYd",
    "UnitID": "LOG2",
    "ItemType": 4,
    "Type": 1,
    "Number": 2
},
{
    "Minimum": 0,
    "RejectCount": 0,
    "RetractedCount": 0,
    "PresentedCount": 0,
    "DispensedCount": 0,
    "InitialCount": 0,
    "CashUnitName": null,

```



```
"CDMType": 0,
"NotelDs": [],
"Extra": {},
"Physical": [
  {
    "RejectCount": 0,
    "RetractedCount": 0,
    "PresentedCount": 0,
    "DispensedCount": 0,
    "InitialCount": 0,
    "Extra": {},
    "HardwareSensors": true,
    "PStatus": 0,
    "Maximum": 3000,
    "Count": 58,
    "CashInCount": 6,
    "UnitID": "PH2",
    "PhysicalPositionName": "PhyPos2"
  }
],
"NumPhysicalCUs": 1,
"NoteNumberList": {
  "NoteNumber": [
    {
      "Count": 58,
      "NoteID": 6
    }
  ],
  "NumOfNoteNumbers": 1
},
"AppLock": false,
"Status": 0,
"Maximum": 3000,
"Count": 58,
"CashInCount": 6,
"Values": 50,
"CurrencyID": "CNY2",
"UnitID": "LOG3",
"ItemType": 4,
"Type": 1,
"Number": 3
},
{
  "Minimum": 0,
```

```
"RejectCount": 0,
"RetractedCount": 0,
"PresentedCount": 0,
"DispensedCount": 0,
"InitialCount": 0,
"CashUnitName": null,
"CDMType": 0,
"NotelDs": [],
"Extra": {},
"Physical": [
  {
    "RejectCount": 0,
    "RetractedCount": 0,
    "PresentedCount": 0,
    "DispensedCount": 0,
    "InitialCount": 0,
    "Extra": {},
    "HardwareSensors": true,
    "PStatus": 0,
    "Maximum": 3000,
    "Count": 0,
    "CashInCount": 0,
    "UnitID": "PH3",
    "PhysicalPositionName": "PhyPos3"
  }
],
"NumPhysicalCUs": 1,
"NoteNumberList": {
  "NoteNumber": [
    {
      "Count": 0,
      "NoteID": 5
    }
  ],
  "NumOfNoteNumbers": 1
},
"AppLock": false,
"Status": 0,
"Maximum": 3000,
"Count": 0,
"CashInCount": 0,
"Values": 20,
"CurrencyID": "CNY\u0014",
"UnitID": "LOG4",
```

```
"ItemType": 4,
"Type": 1,
"Number": 4
},
{
  "Minimum": 0,
  "RejectCount": 0,
  "RetractedCount": 0,
  "PresentedCount": 0,
  "DispensedCount": 0,
  "InitialCount": 0,
  "CashUnitName": null,
  "CDMType": 0,
  "NoteIDs": [],
  "Extra": {},
  "Physical": [
    {
      "RejectCount": 0,
      "RetractedCount": 0,
      "PresentedCount": 0,
      "DispensedCount": 0,
      "InitialCount": 0,
      "Extra": {},
      "HardwareSensors": true,
      "PStatus": 0,
      "Maximum": 3000,
      "Count": 0,
      "CashInCount": 0,
      "UnitID": "PH4",
      "PhysicalPositionName": "PhyPos4"
    }
  ],
  "NumPhysicalCUs": 1,
  "NoteNumberList": {
    "NoteNumber": [
      {
        "Count": 0,
        "NoteID": 3
      }
    ],
    "NumOfNoteNumbers": 1
  },
  "AppLock": false,
  "Status": 0,
```

```

    "Maximum": 3000,
    "Count": 0,
    "CashInCount": 0,
    "Values": 10,
    "CurrencyID": "CNY\n",
    "UnitID": "LOG5",
    "ItemType": 4,
    "Type": 1,
    "Number": 5
  }
],
  "Count": 5
}

```

字段说明：

字段	字段含义	备注
CashIn	钞箱集合	参考 CashIn 字段说明
Count	CashIn 的数量	

CashIn 字段说明

字段	字段含义	备注
Minimum	可分发媒介物件的 CDM 现金单元	可参考 CDM 接口规范获取细节。
RejectCount	从这个现金单元被拒绝到拒绝箱的物件数目	可参考 CDM 接口规范获取细节
RetractedCount	回收到此物理现金单元的物件数目	
PresentedCount	从这个物理现金单元递交给客户的物件数	可参考 CDM 接口规范获取细节
DispensedCount	从这个物理现金单元分发的物件数目	可参考 CDM 接口规范获取细节
InitialCount	物理现金单元中包含的物件数目初始值	
CashUnitName	应用定义的名字，用于帮助标识现金单元的内容	这个值可以为 NULL
CDMType	与 CDM 接口现金单元相对应的现金单元的类型	可参考 CDM 接口规范以获取细节。对只用于 CIM 的现金单元这个值为 0。
NoteIDs	指向 0 结尾的无符号短整数列表，包含存款现金单元或循环现金单元可接受的钞票 ID。这个字段只应用于 WFS_CIM_CITYPINDIVIDUAL 盒子类型。	如果盒子没有定义钞票 ID 或盒子没有定义为 WFS_CIM_CITYPINDIVIDUAL，则 lpusNoteIDs 为 NULL。
Extra	指向厂商特定或任何其他扩展信息的列表	信息以“key=value”字符串的形式返回
Physical	指向一个指针数组，数组中的指针指向 WFSCIMPHCU	参考 Physical 字段说明

	结构	
NumPhysicalCUs	此值表示返回的物理现金单元结构的数目	它至少为 1
NoteNumberList	现金数量集合	参考 NoteNumberList 字段说明
AppLock	是否能被接收到现金单元	如果硬件不支持，忽略此参数。
Status	描述现金单元的状态	<p>0: 现金单元状态良好。</p> <p>1: 现金单元满</p> <p>2: 现金单元即将满</p> <p>3: 现金单元即将空</p> <p>4: 现金单元空</p> <p>5: 现金单元不可操作。</p> <p>6: 现金单元缺失。</p> <p>7: 现金单元的值不可用</p> <p>8: 没有关于现金单元中纸币的可参考的值</p> <p>9: 设备未处于交换状态时，现金单元被装入（包括取走之后重新装入）</p>
Maximum	当 ulCount 到达此值，阈值事件 WFS_USRE_CIM_CASHUNITTHRESHOLD (WFS_CIM_STATCUHIGH) 将会产生	如果此值非 0，则设备的硬件传感器不触发阈值事件。如果此值为 0 则硬件传感器可能触发阈值事件。
Count	此计数的含义依赖于现金单元的类型，报告现金单元中所有类型纸币的数量	
CashInCount	进入逻辑现金单元的物件计数	每次物件由于任何原因进入属于此逻辑现金单元的某个物理现金单元时，此计数增加。
Values	说明现金单元中单个物件的值。这个值表示取款最小分发单位	如果此现金单元的 cCurrencyID 字段为空或者现金单元配置为接受多个面值的纸币，那么这个字段为 0
CurrencyID	一个三字符数组存储着 ISO 格式的货币 ID	
UnitID	现金单元标识符	
ItemType	说明现金单元接受的物件类型	<p>1: 存款单元接受所有合格的钞票类型。</p> <p>2: 存款单元接受所有不合格的钞票</p> <p>4: 存款单元或循环现金单元接受在一个单独的列表中所列的所有合格类型的钞票。</p> <p>8: 所有为 Paragraph 6 等级 2 的纸币保存在此现金单元。</p> <p>16: 所有为 Paragraph 6 等级 3 的纸币保存在此</p>

		现金单元。
Type	现金单元的类型	1: 循环现金单元 2: 存款现金单元。 3: 补给容器。一个现金单元可以清空成为补给容器。 4: 回收现金单元。 5: 拒绝现金单元 6:只可应用于 CDM 接口的现金单元。此值用于报告下列类型的 CDM 现金单元： WFS_CDM_TYPENA, WFS_CDM_TYPEBILLCASSETTE, WFS_CDM_TYPECOINCYLINDER, WFS_CDM_TYPECOINDISPENSER, WFS_CDM_TYPECOUPON 和 WFS_CDM_TYPEDOCUMENT。 参考 usCDMType 字段以获取现金单元类型的详细信息。
Number	现金单元索引号。	1 代表第一个结构，后续的结构每个加 1。

NoteNumberList 字段说明：

字段	字段含义	备注
NoteNumber	钞箱包含的钞票数目的列表	参考 NoteNumber 字段说明
NumOfNoteNumbers	NoteNumber 的数量	

NoteNumber 字段说明

字段	字段含义	备注
Count	现金数	
NoteID	纸币类型	

Physical 字段说明

字段	字段含义	备注
RetractedCount	回收数量	
PresentedCount	呈现给客户数量	
DispensedCount	发放的数量	
HardwareSensor	是否触发传感器事件	
PStatus	物理钞箱状态	0: 钞箱状态良好 1: 钞箱已满 2: 钞箱将满 3: 钞箱将空 4: 钞箱已空或者残留不够将来分配操作 5: 钞箱未被激活 6: 钞箱丢失 7: 钞箱指定值（Values）不可用 8: 钞箱未被校正 9: 钞箱不处于激活状态
Maximum	最大存储数量	

RejectCount	废钞数量	
Count	当前计数	
InitialCount	初始数量	
UnitID	钞箱标识符	
PhysicalPositionName	物理位置名称	
CashInCount	存入数量	
Extra	厂商自定义信息	键值对集合

### 15.7 getTellerInfoAsync(tellerInfo)

函数说明：获取赋予柜员机的对每种货币的计数

输入参数：

```
{
  "CurrencyID": "CNY",
  "TellerID": 0
}
```

字段说明：

字段	字段含义	备注
CurrencyID	货币类型	
TellerID	柜员 id	

返回信息：

```
{
  "TellerTotals": [
    {
      "CashBoxDispensed": 0,
      "CashBoxReceived": 0,
      "CoinsDispensed": 0,
      "CoinsReceived": 0,
      "ItemsDispensed": 0,
      "ItemsReceived": 0,
      "CurrencyID": "CNY"
    }
  ],
  "OutputPosition": 0,
  "InputPosition": 0,
  "TellerID": 8888
}
```

参数说明：

字段	字段含义	备注
TellerTotals	柜员信息统计的集合	参考 TellerTotal 字段说明
OutputPosition	分配的现金输出位置	
InputPosition	分配的现金输入位置	
TellerID	柜员 id	

TellerTotal 字段说明

字段	字段含义	备注
CashBoxDispensed	现金盒里分配的货币的总金额	
CashBoxReceived	现金盒里接受的货币的总金额	
CoinsDispensed	分发的硬币的总金额,	
CoinsReceived	接受的硬币的总金额	
ItemsDispensed	分发货币的总金额	
ItemsReceived	受货币的总金额	
CurrencyID	货币类型	

### 15.8 getCurrencyExpAsync()

函数说明：获取货币指数

返回信息：

```
[
  {
    "Exponent": 0,
    "CurrencyID": "CNY"
  }
]
```

字段说明：

字段	字段含义	备注
Exponent	汇率指数	
CurrencyID	货币 Id	

### 15.9 getBanknoteTypesAsync()

函数说明：获取可被验钞检测的钞票类型信息

返回信息：

```
{
  "NoteTypes": [
    {
      "Configured": false,
      "Release": 5,
      "Values": 100,
      "CurrencyID": "CNY",
      "NoteID": 7
    },
    {
      "Configured": false,
      "Release": 5,
      "Values": 50,
      "CurrencyID": "CNY",
      "NoteID": 6
    },
    {
      "Configured": false,
```



```

        "Release": 5,
        "Values": 20,
        "CurrencyID": "CNY ",
        "NoteID": 5
    },
    {
        "Configured": false,
        "Release": 0,
        "Values": 10,
        "CurrencyID": "CNY",
        "NoteID": 3
    },
    {
        "Configured": false,
        "Release": 4,
        "Values": 5,
        "CurrencyID": "CNY ",
        "NoteID": 2
    },
    {
        "Configured": false,
        "Release": 4,
        "Values": 1,
        "CurrencyID": "CNY ",
        "NoteID": 1
    }
],
    "NumOfNoteTypes": 6
}

```

字段说明：

字段	字段含义	备注
NumOfNoteTypes	验钞支持的钞票类型数目	
NoteTypes	验钞支持的钞票类型列表	参考 NoteTypes 字段说明

NoteTypes 字段说明

字段	字段含义	备注
Configured	说明验钞是否认识此种纸币类型	
Release	钞票类型版本	更高的版本数表示更新的版本。0 表示这种钞票类型只有一个版本。这个值没有被标准化，因此一个钞票的版本号不一定在不同的系统中相同。
Values	单一物件最小分发单位的值。	
CurrencyID	货币类型	CNY

NoteID	纸币类型标识。	
--------	---------	--

#### 15.10 getCashInStatusAsync()

函数说明：获取最近一次存款交易信息

返回信息：

```
{
  "Extra": {
    "NG": "0"
  },
  "NoteNumberList": {
    "NoteNumber": [],
    "NumOfNoteNumbers": 0
  },
  "NumOfRefused": 0,
  "Status": 0
}
```

字段说明：

字段	字段含义	备注
Extra	指向一个厂商特定或任何其他扩展信息的列表	信息以一系列“key=value”形式的字符串返回
NoteNumberList	在存款交易中被放入、识别和接收的钞票的类型的列表。	请参考 NoteNumberList 说明
NumOfRefused	说明存款交易过程中被拒绝的物件数。	
Status	存款交易状态	0:存款交易完成 1:存款交易被回退 2:正在进行存款交易 3:存款交易终止于物件被回收 4:存款交易状态未知 5:存款交易终止于 reset 命令被执行。

NoteNumberList 字段说明：

字段	字段含义	备注
NoteNumber	钞箱包含的钞票数目的列表	参考 NoteNumber 字段说明
NumOfNoteNumbers	NoteNumber 的数量	

NoteNumber 字段说明

字段	字段含义	备注
Count	现金数	
NoteID	纸币类型	

#### 15.11 getP6InfoAsync()

函数说明：获取检测到的级别 2、级别 3 纸币的数量信息和创建的级别 2、级别 3 签名信息

返回信息：

```
{
```

```

    "NumOfSignatures": 0,
    "NoteNumberList": {
        "NoteNumber": [
            {
                "Count": 0,
                "NoteID": 0
            }
        ],
        "NumOfNoteNumbers": 0
    },
    "Level": 0
}

```

字段说明：

字段	字段含义	备注
NumOfSignatures	此存款交易中级别 x 签名的数量	如果为 0，没有签名可用。
NoteNumberList	可被识别为级别 x 的钞票类型的列表	请参考 NoteNumberList 说明
Level	说明纸币的级别	WFS_CIM_LEVEL_2 级别：2 纸币的信息 WFS_CIM_LEVEL_3 级别：3 纸币的信息。

NoteNumberList 字段说明：

字段	字段含义	备注
NoteNumber	钞箱包含的钞票数目的列表	参考 NoteNumber 字段说明
NumOfNoteNumbers	NoteNumber 的数量	

NoteNumber 字段说明

字段	字段含义	备注
Count	现金数	
NoteID	纸币类型	

## 15.12 getP6SignatureAsync(signature)

函数说明：获取一个特定的签名

输入参数：

```

{
    "Index": 0,
    "Level": 2
}

```

字段说明：

字段	字段含义	备注
Index	指明要求的签名的索引	从 0 开始
Level	签名的级别	2：级别 2 3：级别 3 4：级别 4

返回信息：

```
{
  "Signature": null,
  "Orientation": 0,
  "Length": 0,
  "NotelId": 0
}
```

字段说明：

字段	字段含义	备注
Signature	指向返回的签名。	应用必须在循环中多次调用此命令以获取所有的签名。
Orientation	钞票进入的方向	<p>1: 如果纸币以宽的一边作为前导，则纸币前面图像朝上，纸币的顶部边缘被首先放入。如果纸币以窄的一边作为前导，则纸币前面图像朝上，纸币左边边缘被首先放入</p> <p>2: 如果纸币以宽的一边作为前导，则纸币前面图像朝上，纸币的底部边缘被首先放入。如果纸币以窄的一边作为前导，则纸币图像朝上，纸币的右边边缘被首先放入。</p> <p>3: 如果纸币宽的一边作为前导边，则纸币背面图像朝上，纸币顶边首先被放入。如果纸币窄的一边作为前导放入，则纸币背面图像朝上，纸币左边被首先放入。</p> <p>4: 如果纸币宽的一边作为前导，则纸币背面图像朝上，纸币底边首先被放入。如果纸币窄的一边作为前导放入，则纸币背面图像朝上，纸币右边被首先放入。</p> <p>5: 放入的纸币无法确定方向</p> <p>6: 硬件没有确定方向的能力</p>
Length	签名字节长度	
NotelId	纸币类型标识	

### 15.13 cashInStartAsync(cashInStart)

函数说明：开始一个存款交易

输入参数：

```
{
  "InputPosition": 0,
  "OutputPosition": 0,
  "UseRecycleUnits": false,
  "TellerID": 0
}
```

字段说明：

字段	字段含义	备注
----	------	----

InputPosition	说明现金将被放入的位置	0:现金将从默认配置位置放入 1:现金将从左输入位置放入 2:现金将从左输入位置放入 4:现金将从中间输入位置放入 8:现金将从上输入位置放入 16:现金将从下输入位置放入 32:现金将从前输入位置放入 64:现金将从后输入位置放入
OutputPosition	在回退时递交物件给客户的输出位置	0: 物件将被递交于默认配置 128: 物件可被递交于左输出位置 256: 物件可被递交于右输出位置 512: 物件可被递交于中间输出位置 1024: 物件可被递交于上输出位置 2048: 物件可被递交于下输出位置 4096: 物件可被递交于前输出位置 8192: 物件可被递交于后输出位置
UseRecycleUnits	说明循环现金单元在现金存款交易过程中是否会被使用	如果没有循环现金单元或硬件不支持, 此参数将被忽略。
TellerID	柜员机标识符	此字段不被应用于自服务 CIM, 并被设置为 0

返回信息:

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明:

字段	字段含义	备注
ExcuteResult	函数执行结果	0: 函数成功执行
IsSuccess	函数执行是否成功	true: 执行成功 false: 执行失败

#### 15.14 cashInAsync()

函数说明: 将钞票从输入位置转移到设备内部

返回信息:

```
{
  "NoteNumber": [
    {
      "Count": 1,
      "NoteID": 6
    }
  ],
  "NumOfNoteNumbers": 1
}
```

}

字段说明:

字段	字段含义	备注
NoteNumber	钞箱包含的钞票数目的列表	参考 NoteNumber 字段说明
NumOfNoteNumbers	NoteNumber 的数量	

NoteNumber 字段说明

字段	字段含义	备注
Count	现金数	
NoteID	纸币类型	

#### 15.15 cashInEndAsync()

函数说明: 终止一个存款交易

返回信息:

```
{
  "CashIn": [
    {
      "Minimum": 0,
      "RejectCount": 0,
      "RetractedCount": 0,
      "PresentedCount": 0,
      "DispensedCount": 0,
      "InitialCount": 0,
      "CashUnitName": null,
      "CDMType": 0,
      "NoteIDs": [],
      "Extra": {},
      "Physical": [
        {
          "RejectCount": 0,
          "RetractedCount": 0,
          "PresentedCount": 0,
          "DispensedCount": 0,
          "InitialCount": 0,
          "Extra": {},
          "HardwareSensors": true,
          "PStatus": 0,
          "Maximum": 3000,
          "Count": 1,
          "CashInCount": 1,
          "UnitID": "PH2",
          "PhysicalPositionName": "PhyPos2"
        }
      ]
    },
    {
      "NumPhysicalCUs": 1,
```

```

    "NoteNumberList": {
      "NoteNumber": [
        {
          "Count": 1,
          "NoteID": 6
        }
      ],
      "NumOfNoteNumbers": 1
    },
    "AppLock": false,
    "Status": 0,
    "Maximum": 3000,
    "Count": 1,
    "CashInCount": 1,
    "Values": 50,
    "CurrencyID": "CNY2",
    "UnitID": "LOG3",
    "ItemType": 4,
    "Type": 1,
    "Number": 3
  }
],
"Count": 1
}

```

字段说明：

字段	字段含义	备注
CashIn	钞箱集合	参考 List 字段说明
Count	CashIn 的数量	

List 字段说明（钞箱字段说明）

字段	字段含义	备注
Minimum	可分发媒介物件的 CDM 现金单元	可参考 CDM 接口规范获取细节。
RejectCount	从这个现金单元被拒绝到拒绝箱的物件数目	可参考 CDM 接口规范获取细节
RetractedCount	回收到此物理现金单元的物件数目	
PresentedCount	从这个物理现金单元递交给客户的物件数	可参考 CDM 接口规范获取细节
DispensedCount	从这个物理现金单元分发的物件数目	可参考 CDM 接口规范获取细节
InitialCount	物理现金单元中包含的物件数目初始值	
CashUnitName	应用定义的名字，用于帮助标识现金单元的内容	这个值可以为 NULL

CDMType	与 CDM 接口现金单元相对应的现金单元的类型	可参考 CDM 接口规范以获取细节。对只用于 CIM 的现金单元这个值为 0。
NoteIDs	指向 0 结尾的无符号短整数列表，包含存款现金单元或循环现金单元可接受的钞票 ID。这个字段只应用于 WFS_CIM_CITYPINDIVIDUAL 盒子类型。	如果盒子没有定义钞票 ID 或盒子没有定义为 WFS_CIM_CITYPINDIVIDUAL，则 lpusNoteIDs 为 NULL。
Extra	指向厂商特定或任何其他扩展信息的列表	信息以“key=value”字符串的形式返回
Physical	指向一个指针数组，数组中的指针指向 WFSCIMPHCU 结构	参考 Physical 字段说明
NumPhysicalCUs	此值表示返回的物理现金单元结构的数目	它至少为 1
NoteNumberList	现金数量集合	参考 NoteNumberList 字段说明
AppLock	是否能被接收到现金单元	如果硬件不支持，忽略此参数。
Status	描述现金单元的状态	0: 现金单元状态良好。 1: 现金单元满 2: 现金单元即将满 3: 现金单元即将空 4: 现金单元空 5: 现金单元不可操作。 6: 现金单元缺失。 7: 现金单元的值不可用 8: 没有关于现金单元中纸币的可参考的值 9: 设备未处于交换状态时，现金单元被装入（包括取走之后重新装入）
Maximum	当 ulCount 到达此值，阈值事件 WFS_USRE_CIM_CASHUNITTHRESHOLD (WFS_CIM_STATCUHIGH) 将会产生	如果此值非 0，则设备的硬件传感器不触发阈值事件。如果此值为 0 则硬件传感器可能触发阈值事件。
Count	此计数的含义依赖于现金单元的类型，报告现金单元中所有类型纸币的数量	
CashInCount	进入逻辑现金单元的物件计数	每次物件由于任何原因进入属于此逻辑现金单元的某个物理现金单元时，此计数增加。
Values	说明现金单元中单个物件的值。这个值表示取款最小分发单位	如果此现金单元的 cCurrencyID 字段为空或者现金单元配置为接受多个面值的纸币，那么这个字段为 0
CurrencyID	一个三字符数组存储着 ISO 格式的货币 ID	



UnitID	现金单元标识符	
ItemType	说明现金单元接受的物件类型	<p>1: 存款单元接受所有合格的钞票类型。</p> <p>2: 存款单元接受所有不合格的钞票</p> <p>4: 存款单元或循环现金单元接受在一个单独的列表中所列的所有合格类型的钞票。</p> <p>8: 所有为 Paragraph 6 等级 2 的纸币保存在此现金单元。</p> <p>16: 所有为 Paragraph 6 等级 3 的纸币保存在此现金单元。</p>
Type	现金单元的类型	<p>1: 循环现金单元</p> <p>2: 存款现金单元。</p> <p>3: 补给容器。一个现金单元可以清空成为补给容器。</p> <p>4: 回收现金单元。</p> <p>5: 拒绝现金单元</p> <p>6:只可应用于 CDM 接口的现金单元。此值用于报告下列类型的 CDM 现金单元： WFS_CDM_TYPENA, WFS_CDM_TYPEBILLCASSETTE, WFS_CDM_TYPECOINCYLINDER, WFS_CDM_TYPECOINDISPENSER, WFS_CDM_TYPECOUPON 和 WFS_CDM_TYPEDOCUMENT。 参考 usCDMType 字段以获取现金单元类型的详细信息。</p>
Number	现金单元索引号。	1 代表第一个结构，后续的结构每个加 1。

NoteNumberList 字段说明：

字段	字段含义	备注
NoteNumber	钞箱包含的钞票数目的列表	参考 NoteNumber 字段说明
NumOfNoteNumbers	NoteNumber 的数量	

NoteNumber 字段说明

字段	字段含义	备注
Count	现金数	
NoteID	纸币类型	

Physical 字段说明

字段	字段含义	备注
RetractedCount	回收数量	
PresentedCount	呈现给客户数量	
DispensedCount	发放的数量	
HardwareSensor	是否触发传感器事件	

PStatus	物理钞箱状态	0: 钞箱状态良好 1: 钞箱已满 2: 钞箱将满 3: 钞箱将空 4: 钞箱已空或者残留不够将来分配操作 5: 钞箱未被激活 6: 钞箱丢失 7: 钞箱指定值（Values）不可用 8: 钞箱未被校正 9: 钞箱不处于激活状态
Maximum	最大存储数量	
RejectCount	废钞数量	
Count	当前计数	
InitialCount	初始数量	
UnitID	钞箱标识符	
PhysicalPositionName	物理位置名称	
CashInCount	存入数量	
Extra	厂商自定义信息	键值对集合

#### 15.16 cashInRollbackAsync()

函数说明：当放入数量和设备计数不同时可以回退

返回信息：

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明：

字段	字段含义	备注
ExcuteResult	函数执行结果	0: 函数成功执行
IsSuccess	函数执行是否成功	true: 执行成功 false: 执行失败

#### 15.17 retractAsync(retract)

函数说明：回收现金

输入参数：

```
{
  "Index": 0,
  "RetractArea": 0,
  "OutputPosition": 0
}
```

返回信息：

```
{
  "CashIn": [
    {
```

```
"Minimum": 0,
"RejectCount": 0,
"RetractedCount": 0,
"PresentedCount": 0,
"DispensedCount": 0,
"InitialCount": 0,
"CashUnitName": null,
"CDMType": 0,
"NoteIDs": [],
"Extra": {},
"Physical": [
  {
    "RejectCount": 0,
    "RetractedCount": 0,
    "PresentedCount": 0,
    "DispensedCount": 0,
    "InitialCount": 0,
    "Extra": {},
    "HardwareSensors": true,
    "PStatus": 0,
    "Maximum": 3000,
    "Count": 1,
    "CashInCount": 1,
    "UnitID": "PH2",
    "PhysicalPositionName": "PhyPos2"
  }
],
"NumPhysicalCUs": 1,
"NoteNumberList": {
  "NoteNumber": [
    {
      "Count": 1,
      "NoteID": 6
    }
  ],
  "NumOfNoteNumbers": 1
},
"AppLock": false,
"Status": 0,
"Maximum": 3000,
"Count": 1,
"CashInCount": 1,
"Values": 50,
"CurrencyID": "CNY2",
```

```

        "UnitID": "LOG3",
        "ItemType": 4,
        "Type": 1,
        "Number": 3
    }
],
    "Count": 1
}

```

#### 15.18 openShutterAsync(position)

函数说明：打开钞门

返回信息：

```

{
    "ExcuteResult": 0,
    "IsSuccess": true
}

```

字段说明：

字段	字段含义	备注
ExcuteResult	函数执行结果	0：函数成功执行
IsSuccess	函数执行是否成功	true：执行成功 false：执行失败

#### 15.19 closeShutterAsync(position)

函数说明：关闭钞门

返回信息：

```

{
    "ExcuteResult": 0,
    "IsSuccess": true
}

```

字段说明：

字段	字段含义	备注
ExcuteResult	函数执行结果	0：函数成功执行
IsSuccess	函数执行是否成功	true：执行成功 false：执行失败

#### 15.20 setTellerInfoAsync(tellerUpdate)

函数说明：为每种货币设置初始计数到柜员

输入参数：

```

{
    "TellerDetails": {
        "TellerTotals": [
            {
                "CashBoxDispensed": 0,

```

```

        "CashBoxReceived": 0,
        "CoinsDispensed": 0,
        "CoinsReceived": 0,
        "ItemsDispensed": 0,
        "ItemsReceived": 0,
        "CurrencyID": "CNY"
    }
},
    "OutputPosition": 0,
    "InputPosition": 0,
    "TellerID": 8888
},
    "Action": 0
}

```

字段说明:

字段	字段含义	备注
TellerDetails	柜员信息明细	参考 TellerDetails 字段说明
Action	将被执行的指令	1: 创建柜员信息 2: 修改柜员信息 3: 删除柜员信息

TellerDetails 字段说明:

字段	字段含义	备注
TellerTotals	柜员信息统计的集合	参考 TellerTotal 字段说明
OutputPosition	分配的现金输出位置	
InputPosition	分配的现金输入位置	
TellerID	柜员 Id	

TellerTotal 字段说明

字段	字段含义	备注
CashBoxDispensed	现金盒里分配的货币的总金额	
CashBoxReceived	现金盒里接受的货币的总金额	
CoinsDispensed	分发的硬币的总金额,	
CoinsReceived	接受的硬币的总金额	
ItemsDispensed	分发货币的总金额	
ItemsReceived	受货币的总金额	
CurrencyID	货币类型	

返回信息:

```

{
    "ExcuteResult": 0,
    "IsSuccess": true
}

```

字段说明:

字段	字段含义	备注
ExcuteResult	函数执行结果	0: 函数成功执行

IsSuccess	函数执行是否成功	true: 执行成功 false: 执行失败
-----------	----------	---------------------------

#### 15.21 setCashUnitInfoAsync(cashInfo)

函数说明：设置钞箱的状态和数据

输入参数：

```
{
  "CashIn": [
    {
      "Minimum": 0,
      "RejectCount": 0,
      "RetractedCount": 0,
      "PresentedCount": 0,
      "DispensedCount": 0,
      "InitialCount": 0,
      "CashUnitName": null,
      "CDMType": 0,
      "NoteIDs": [],
      "Extra": {},
      "Physical": [
        {
          "RejectCount": 0,
          "RetractedCount": 0,
          "PresentedCount": 0,
          "DispensedCount": 0,
          "InitialCount": 0,
          "Extra": {},
          "HardwareSensors": true,
          "PStatus": 0,
          "Maximum": 3000,
          "Count": 1,
          "CashInCount": 1,
          "UnitID": "PH2",
          "PhysicalPositionName": "PhyPos2"
        }
      ],
      "NumPhysicalCUs": 1,
      "NoteNumberList": {
        "NoteNumber": [
          {
            "Count": 1,
            "NoteID": 6
          }
        ]
      }
    }
  ]
}
```

```

    ],
    "NumOfNoteNumbers": 1
  },
  "AppLock": false,
  "Status": 0,
  "Maximum": 3000,
  "Count": 1,
  "CashInCount": 1,
  "Values": 50,
  "CurrencyID": "CNY2",
  "UnitID": "LOG3",
  "ItemType": 4,
  "Type": 1,
  "Number": 3
}
],
"Count": 1
}

```

字段说明：

字段	字段含义	备注
CashIn	钞箱集合	参考 CashIn 字段说明
Count	CashIn 的数量	

CashIn 字段说明

字段	字段含义	备注
Minimum	可分发媒介物件的 CDM 现金单元	可参考 CDM 接口规范获取细节。
RejectCount	从这个现金单元被拒绝到拒绝箱的物件数目	可参考 CDM 接口规范获取细节
RetractedCount	回收到此物理现金单元的物件数目	
PresentedCount	从这个物理现金单元递交给客户的物件数	可参考 CDM 接口规范获取细节
DispensedCount	从这个物理现金单元分发的物件数目	可参考 CDM 接口规范获取细节
InitialCount	物理现金单元中包含的物件数目初始值	
CashUnitName	应用定义的名字，用于帮助标识现金单元的内容	这个值可以为 NULL
CDMType	与 CDM 接口现金单元相对应的现金单元的类型	可参考 CDM 接口规范以获取细节。对只用于 CIM 的现金单元这个值为 0。
NoteIDs	指向 0 结尾的无符号短整数列表，包含存款现金单元或循环现金单元可接受的钞票 ID。这个字段只应用于	如果盒子没有定义钞票 ID 或盒子没有定义为 WFS_CIM_CITYINDIVIDUAL，则 IpusNoteIDs 为 NULL。

	WFS_CIM_CITYINDIVIDUAL 盒子类型。	
Extra	指向厂商特定或任何其他扩展信息的列表	信息以“key=value”字符串的形式返回
Physical	指向一个指针数组，数组中的指针指向 WFSCIMPHCU 结构	参考 Physical 字段说明
NumPhysicalCUs	此值表示返回的物理现金单元结构的数目	它至少为 1
NoteNumberList	现金数量集合	参考 NoteNumberList 字段说明
AppLock	是否能被接收到现金单元	如果硬件不支持，忽略此参数。
Status	描述现金单元的状态	0：现金单元状态良好。 1：现金单元满 2：现金单元即将满 3：现金单元即将空 4：现金单元空 5：现金单元不可操作。 6：现金单元缺失。 7：现金单元的值不可用 8：没有关于现金单元中纸币的可参考的值 9：设备未处于交换状态时，现金单元被装入（包括取走之后重新装入）
Maximum	当 ulCount 到达此值，阈值事件 WFS_USRE_CIM_CASHUNITTHRESHOLD (WFS_CIM_STATCUHIGH) 将会产生	如果此值非 0，则设备的硬件传感器不触发阈值事件。如果此值为 0 则硬件传感器可能触发阈值事件。
Count	此计数的含义依赖于现金单元的类型，报告现金单元中所有类型纸币的数量	
CashInCount	进入逻辑现金单元的物件计数	每次物件由于任何原因进入属于此逻辑现金单元的某个物理现金单元时，此计数增加。
Values	说明现金单元中单个物件的值。这个值表示取款最小分发单位	如果此现金单元的 cCurrencyID 字段为空或者现金单元配置为接受多个面值的纸币，那么这个字段为 0
CurrencyID	一个三字符数组存储着 ISO 格式的货币 ID	
UnitID	现金单元标识符	
ItemType	说明现金单元接受的物件类型	1：存款单元接受所有合格的钞票类型。 2：存款单元接受所有不合格的钞票 4：存款单元或循环现金单元接受在一个单独的列表中所列的所有合格类型的钞票。



		<p>8: 所有为 Paragraph 6 等级 2 的纸币保存在此 现金单元。</p> <p>16: 所有为 Paragraph 6 等级 3 的纸币保存在此 现金单元。</p>
Type	现金单元的类型	<p>1: 循环现金单元</p> <p>2: 存款现金单元。</p> <p>3: 补给容器。一个现金单元可以清空成为补给容器。</p> <p>4: 回收现金单元。</p> <p>5: 拒绝现金单元</p> <p>6: 只可应用于 CDM 接口的现金单元。此值用于报告下列类型的 CDM 现金单元： WFS_CDM_TYPENA, WFS_CDM_TYPEBILLCASSETTE, WFS_CDM_TYPECOINCYLINDER, WFS_CDM_TYPECOINDISPENSER, WFS_CDM_TYPECOUPON 和 WFS_CDM_TYPEDOCUMENT。 参考 usCDMType 字段以获取现金单元类型的详细信息。</p>
Number	现金单元索引号。	1 代表第一个结构，后续的结构每个加 1。

NoteNumberList 字段说明：

字段	字段含义	备注
NoteNumber	钞箱包含的钞票数目的列表	参考 NoteNumber 字段说明
NumOfNoteNumbers	NoteNumber 的数量	

NoteNumber 字段说明

字段	字段含义	备注
Count	现金数	
NoteID	纸币类型	

Physical 字段说明

字段	字段含义	备注
RetractedCount	回收数量	
PresentedCount	呈现给客户数量	
DispensedCount	发放的数量	
HardwareSensor	是否触发传感器事件	
PStatus	物理钞箱状态	<p>0: 钞箱状态良好</p> <p>1: 钞箱已满</p> <p>2: 钞箱将满</p> <p>3: 钞箱将空</p> <p>4: 钞箱已空或者残留不够将来分配操作</p> <p>5: 钞箱未被激活</p>

		6: 钞箱丢失 7: 钞箱指定值 (Values) 不可用 8: 钞箱未被校正 9: 钞箱不处于激活状态
Maximum	最大存储数量	
RejectCount	废钞数量	
Count	当前计数	
InitialCount	初始数量	
UnitID	钞箱标识符	
PhysicalPositionName	物理位置名称	
CashInCount	存入数量	
Extra	厂商自定义信息	键值对集合

返回信息:

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明:

字段	字段含义	备注
ExcuteResult	函数执行结果	0: 函数成功执行
IsSuccess	函数执行是否成功	true: 执行成功 false: 执行失败

## 15.22 startExchangeAsync(startEx)

函数说明: 设置设备进入交换状态, 可以进行清空、补给、取走或替换

输入参数:

```
{
  "Output": {
    "Number": 0,
    "Position": 0,
    "LogicalNumber": 0
  },
  "CUNumList": [],
  "Count": 0,
  "TellerID": 0,
  "ExchangeType": 0
}
```

字段说明:

字段	字段含义	备注
Output	现金单元将被清空时, 此字段被使用	参考 Output 字段说明
CUNumList	指向一个包含了将被交换的现金单元逻辑数的无符	

	号短整数数组。	
Count	被交换的现金单元数	
TellerID	柜员机标识符	
ExchangeType	指定现金单元交换操作的类型	<p>1: 现金单元将被手工补给, 通过用手填充或清空现金单元或通过替换现金单元。</p> <p>2: 物件将从补给容器转移到钞票现金单元。物件将被从钞票现金单元转移到补给容器。在一个循环机上, CDM 接口必须用于从补给容器转移物件。</p> <p>4: 物件将从循环现金单元被转移到现金单元或输出位置</p> <p>8: 物件将从存款入口被转移到钞票现金单元</p>

Output 字段说明:

字段	字段含义	备注
LogicalNumber	要清空逻辑钞箱号	
Number	要转移到的逻辑钞箱号	
Position	现金将要被转移的位置	<p>0: 转移物件到现金单元</p> <p>1: 转移物件到左输出位置</p> <p>2: 转移物件到右输出位置</p> <p>4: 转移物件到中间输出位置</p> <p>8: 转移物件到上输出位置</p> <p>16: 转移物件到下输出位置</p> <p>32: 转移物件到前输出位置</p> <p>64: 转移物件到后输出位置</p>

返回信息:

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明:

字段	字段含义	备注
ExcuteResult	函数执行结果	0: 函数成功执行
IsSuccess	函数执行是否成功	<p>true: 执行成功</p> <p>false: 执行失败</p>

### 15.23 endExchangeAsync(cashInfo)

函数说明: 结束交换状态

输入参数:

```
{
  "CashIn": [
    {
      "Minimum": 0,
```

```
"RejectCount": 0,
"RetractedCount": 0,
"PresentedCount": 0,
"DispensedCount": 0,
"InitialCount": 0,
"CashUnitName": null,
"CDMType": 0,
"NotelDs": [],
"Extra": {},
"Physical": [
  {
    "RejectCount": 0,
    "RetractedCount": 0,
    "PresentedCount": 0,
    "DispensedCount": 0,
    "InitialCount": 0,
    "Extra": {},
    "HardwareSensors": false,
    "PStatus": 0,
    "Maximum": 0,
    "Count": 0,
    "CashInCount": 0,
    "UnitID": null,
    "PhysicalPositionName": null
  }
],
"NumPhysicalCUs": 0,
"NoteNumberList": {
  "NoteNumber": [
    {
      "Count": 0,
      "NoteID": 0
    }
  ],
  "NumOfNoteNumbers": 0
},
"AppLock": false,
"Status": 0,
"Maximum": 0,
"Count": 0,
"CashInCount": 0,
"Values": 0,
"CurrencyID": null,
"UnitID": null,
```

```

        "ItemType": 0,
        "Type": 0,
        "Number": 0
    }
],
    "Count": 0
}

```

字段说明：

字段	字段含义	备注
CashIn	钞箱集合	参考 CashIn 字段说明
Count	CashIn 的数量	

CashIn 字段说明

字段	字段含义	备注
Minimum	可分发媒介物件的 CDM 现金单元	可参考 CDM 接口规范获取细节。
RejectCount	从这个现金单元被拒绝到拒绝箱的物件数目	可参考 CDM 接口规范获取细节
RetractedCount	回收到此物理现金单元的物件数目	
PresentedCount	从这个物理现金单元递交给客户的物件数	可参考 CDM 接口规范获取细节
DispensedCount	从这个物理现金单元分发的物件数目	可参考 CDM 接口规范获取细节
InitialCount	物理现金单元中包含的物件数目初始值	
CashUnitName	应用定义的名字，用于帮助标识现金单元的内容	这个值可以为 NULL
CDMType	与 CDM 接口现金单元相对应的现金单元的类型	可参考 CDM 接口规范以获取细节。对只用于 CIM 的现金单元这个值为 0。
NoteIDs	指向 0 结尾的无符号短整数列表，包含存款现金单元或循环现金单元可接受的钞票 ID。这个字段只应用于 WFS_CIM_CITYINDIVIDUAL 盒子类型。	如果盒子没有定义钞票 ID 或盒子没有定义为 WFS_CIM_CITYINDIVIDUAL，则 lpusNoteIDs 为 NULL。
Extra	指向厂商特定或任何其他扩展信息的列表	信息以“key=value”字符串的形式返回
Physical	指向一个指针数组，数组中的指针指向 WFSCIMPHCU 结构	参考 Physical 字段说明
NumPhysicalCUs	此值表示返回的物理现金单元结构的数目	它至少为 1
NoteNumberList	现金数量集合	参考 NoteNumberList 字段说明
AppLock	是否能被接收到现金单元	如果硬件不支持，忽略此参数。
Status	描述现金单元的状态	0：现金单元状态良好。

		1: 现金单元满 2: 现金单元即将满 3: 现金单元即将空 4: 现金单元空 5: 现金单元不可操作。 6: 现金单元缺失。 7: 现金单元的值不可用 8: 没有关于现金单元中纸币的可参考的值 9: 设备未处于交换状态时, 现金单元被装入 (包括取走之后重新装入)
Maximum	当 ulCount 到达此值, 阈值事件 WFS_USRE_CIM_CASHUNITTHRESHOLD (WFS_CIM_STATCUHIGH) 将会产生	如果此值非 0, 则设备的硬件传感器不触发阈值事件。如果此值为 0 则硬件传感器可能触发阈值事件。
Count	此计数的含义依赖于现金单元的类型, 报告现金单元中所有类型纸币的数量	
CashInCount	进入逻辑现金单元的物件计数	每次物件由于任何原因进入属于此逻辑现金单元的某个物理现金单元时, 此计数增加。
Values	说明现金单元中单个物件的值。这个值表示取款最小分发单位	如果此现金单元的 cCurrencyID 字段为空或者现金单元配置为接受多个面值的纸币, 那么这个字段为 0
CurrencyID	一个三字符数组存储着 ISO 格式的货币 ID	
UnitID	现金单元标识符	
ItemType	说明现金单元接受的物件类型	1: 存款单元接受所有合格的钞票类型。 2: 存款单元接受所有不合格的钞票 4: 存款单元或循环现金单元接受在一个单独的列表中所列的所有合格类型的钞票。 8: 所有为 Paragraph 6 等级 2 的纸币保存在此现金单元。 16: 所有为 Paragraph 6 等级 3 的纸币保存在此现金单元。
Type	现金单元的类型	1: 循环现金单元 2: 存款现金单元。 3: 补给容器。一个现金单元可以清空成为补给容器。 4: 回收现金单元。

		<p>5: 拒绝现金单元</p> <p>6:只可应用于 CDM 接口的现金单元。此值用于报告下列类型的 CDM 现金单元： WFS_CDM_TYPENA, WFS_CDM_TYPEBILLCASSETTE, WFS_CDM_TYPECOINCYLINDER, WFS_CDM_TYPECOINDISPENSER, WFS_CDM_TYPECOUPON 和 WFS_CDM_TYPEDOCUMENT。 参考 usCDMType 字段以获取现金单元类型的详细信息。</p>
Number	现金单元索引号。	1 代表第一个结构，后续的结构每个加 1。

NoteNumberList 字段说明：

字段	字段含义	备注
NoteNumber	钞箱包含的钞票数目的列表	参考 NoteNumber 字段说明
NumOfNoteNumbers	NoteNumber 的数量	

NoteNumber 字段说明

字段	字段含义	备注
Count	现金数	
NoteID	纸币类型	

Physical 字段说明

字段	字段含义	备注
RetractedCount	回收数量	
PresentedCount	呈现给客户数量	
DispensedCount	发放的数量	
HardwareSensor	是否触发传感器事件	
PStatus	物理钞箱状态	<p>0: 钞箱状态良好</p> <p>1: 钞箱已满</p> <p>2: 钞箱将满</p> <p>3: 钞箱将空</p> <p>4: 钞箱已空或者残留不够将来分配操作</p> <p>5: 钞箱未被激活</p> <p>6: 钞箱丢失</p> <p>7: 钞箱指定值（Values）不可用</p> <p>8: 钞箱未被校正</p> <p>9: 钞箱不处于激活状态</p>
Maximum	最大存储数量	
RejectCount	废钞数量	
Count	当前计数	
InitialCount	初始数量	
UnitID	钞箱标识符	
PhysicalPositionName	物理位置名称	

CashInCount	存入数量	
Extra	厂商自定义信息	键值对集合

返回信息：

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明：

字段	字段含义	备注
ExcuteResult	函数执行结果	0：函数成功执行
IsSuccess	函数执行是否成功	true：执行成功 false：执行失败

#### 15.24 openSafeDoorAsync()

函数说明：解锁保险门

返回信息：

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明：

字段	字段含义	备注
ExcuteResult	函数执行结果	0：函数成功执行
IsSuccess	函数执行是否成功	true：执行成功 false：执行失败

#### 15.25 resetAsync(itemPosition)

函数说明：重置

输入参数：

```
{
  "OutputPosition": 0,
  "RetractArea":
  {
    "Index":1,
    "RetractArea":1,
    "OutputPosition":
  },
  "Number": 0
}
```

参数说明：

字段	字段含义	备注
Index	钞箱索引	从 1 开始
RetractArea	回收位置	仅当钞箱类型为回收类型时可用,参考



		RetractArea 字段说明
OutputPosition	输出位置	0: 默认位置 1: 左边 2: 右边 4: 中间 64: 上边 128: 下边 2048: 前面 4096: 后面

RetractArea 字段说明

字段	字段含义	备注
Index	钞箱索引	从 1 开始
RetractArea	回收位置	1: 回收箱 2: 传送装置 4: 暂存 8: 现金箱 16: 不支持 32: 出口位置
OutputPosition	输出位置	0: 128: 256: 512: 1024: 2048: 4096: 8192:

返回信息:

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明:

字段	字段含义	备注
ExcuteResult	函数执行结果	0: 函数成功执行
IsSuccess	函数执行是否成功	true: 执行成功 false: 执行失败

#### 15.26 configureCashInUnitsAsync(cashInType)

函数说明: 改变一个存款单元或者循环单元可接受的钞票类型

输入参数:

```
{
  "NoteIDs": [],
  "Type": 0,
```

```
"Number": 0
}
```

字段说明:

字段	字段含义	备注
NotelDs	验钞可接受的纸币类型标识 Id	
Type	钞箱类型	
Number	逻辑钞箱号	

返回信息:

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明:

字段	字段含义	备注
ExcuteResult	函数执行结果	0: 函数成功执行
IsSuccess	函数执行是否成功	true: 执行成功 false: 执行失败

#### 15.27 configureNoteTypesAsync(notelDs)

函数说明: 配置存款时验钞将识别的类型 Id

输入参数: 类型标识 Id 数组, 例如:

[1,2]

返回信息:

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明:

字段	字段含义	备注
ExcuteResult	函数执行结果	0: 函数成功执行
IsSuccess	函数执行是否成功	true: 执行成功 false: 执行失败

#### 15.28 createP6SignatureAsync()

函数说明: 创建一个被检查和被认为伪造的签名

返回信息:

```
{
  "Signature": "",
  "Orientation": 0,
  "Length": 0,
  "NotelD": 0
}
```

字段说明:

字段	字段含义	备注
Signature	签名数据	
Orientation	钞票进入的方向	<p>1: 如果纸币以宽的一边作为前导, 则纸币前面图像朝上, 纸币的顶部边缘被首先放入。如果纸币以窄的一边作为前导, 则纸币前面图像朝上, 纸币左边边缘被首先放入。</p> <p>2: 如果纸币以宽的一边作为前导, 则纸币前面图像朝上, 纸币的底部边缘被首先放入。如果纸币以窄的一边作为前导, 则纸币图像朝上, 纸币的右边边缘被首先放入。</p> <p>3: 如果纸币宽的一边作为前导边, 则纸币背面图像朝上, 纸币顶边首先被放入。如果纸币窄的一边作为前导放入, 则纸币背面图像朝上, 纸币左边被首先放入。</p> <p>4: 如果纸币宽的一边作为前导, 则纸币背面图像朝上, 纸币底边首先被放入。如果纸币窄的一边作为前导放入, 则纸币背面图像朝上, 纸币右边被首先放入。</p> <p>5: 放入的纸币无法确定方向。</p> <p>6: 硬件没有确定方向的能力。</p>
Length	Signature 的 bytes 数据长度	
Noteld	纸币类型标识 Id	0~65535

## 16. CashDispenser

### 16.1 lockAsync()

函数说明: 锁定操作, 处理一个事务时使用

### 16.2 unlockAsync()

函数说明: 解锁操作

### 16.3 cancelAsync()

函数说明: 取消操作

### 16.4 getStatusAsync()

函数说明: 获取设备状态

返回信息:

```
{
  "AntiFraudModule": 0,
  "PowerSaveRecoveryTime": 0,
  "DevicePosition": 0,
  "GuidLights": [],
  "Extra": {
```

```

    "Name": "GWI_CRM_XFS300",
    "SPVersion": "Ver1.0.1.1",
    "DriverVersion": "XFS 3.0.0",
    "FirmwareVersion": "UR Boot:TS-P177-S120100 ,00020000;FPGA:URFPGA,000006  ",
    "LastErrCode": "9F00000000",
    "LastDesc": "Incorrect timing for issuing Firmware Version Read"
  },
  "Positions": [
    {
      "TransportStatus": 0,
      "Transport": 0,
      "PositionStatus": 0,
      "Shutter": 4,
      "Position": 2048
    }
  ],
  "IntermediateStacker": 0,
  "Dispenser": 1,
  "SafeDoor": 1,
  "Device": 0
}

```

字段说明:

字段	字段含义	备注
AntiFraudModule	反欺骗模块	0: 不支持; 1: 状态正常; 2: 设备不可用 3: 检测到外部设备 4: 状态未知
PowerSaveRecoveryTime	节电唤醒时间	
DevicePosition	设备位置	0: 设备处于正常操作位置 1: 设备被从它的正常操作位置移走 2: 由于硬件错误或者其它的情况, 读卡器的位置不能确定 3: 实际设备没有能力检测其位置
GuidLights	指示灯状态	
Extra	厂商自定义信息	键值对
Positions	位置	参考 Positions 字段说明
IntermediateStacker	暂存状态	0: 空的 1: 不是空的 2: 不是空的, 客户操作 3: 不是空的, 未知是否客户操作 4: 未知状态 5: 不支持状态查询
Dispenser	现金发放单元状态	0: 所有单元正常

		1: 至少一个单元不正常 2: 所有单元不可用 3: 状态未知
SafeDoor	保险门状态	1: 物理设备没有保险门或不支持状态报告 2: 保险箱门打开 3: 保险箱门关闭 4: 未知状态
Device	设备状态	0: 设备在线 1: 设备脱机 2: 设备断电或物理没连接 3: 设备不存在 4: 设备因硬件错误无法操作 5: 设备存在但有人正在阻止设备进行合适的操作 6: 设备忙当前不能处理指令 7: 设备存在但是检测到试图欺骗 8: 可能存在欺骗

Positions 字段说明:

字段	字段含义	备注
Position	输出位置	0: 默认位置 1: 左边 2: 右边 4: 中间 64: 上边 128: 下边 2048: 前面 4096: 后面
Shutter	钞门状态	0: 关闭 1: 打开 2: 卡住 3: 位置 4: 不支持状态查询
PositionStatus	输出位置状态	0: 空的 1: 不是空的 2: 未知 3: 不支持状态查询
Transport	传送装置状态	0: 正常 1: 不可用 2: 未知 3: 不支持状态查询
TransportStatus	传送装置上现金状态	0: 空的 1: 不是空的 2: 不是空的, 客户接入 3: 不是空位, 未知接入状态

		4: 不支持状态查询
--	--	------------

### 16.5 getCapabilitiesAsync()

函数说明：获取设备支持能力

返回信息：

```
{
  "AntiFraudModule": false,
  "PrepareDispense": false,
  "PowerSaveControl": false,
  "GuidLights": [],
  "Extra": {},
  "ExchangeType": 1,
  "MoveItems": 3,
  "Positions": 2048,
  "ItemsTakenSensor": true,
  "IntermediateStacker": true,
  "CashBox": false,
  "SafeDoor": false,
  "RetractStackerActions": 8,
  "RetractTransportActions": 8,
  "RetractAreas": 1,
  "ShutterControl": true,
  "Shutter": false,
  "Compound": true,
  "MaxDispenseItems": 300,
  "Type": 1,
  "Class": 3
}
```

字段说明：

字段	字段含义	备注
AntiFraudModule	是否支持饭欺骗模块	
PrepareDispense	是否支持预发放	
PowerSaveControl	是否支持节电控制	
GuidLights	指示灯支持情况	
Extra	厂商自定义信息	
ExchangeType	交换类型	1: 手动 2: 通过交换箱
MoveItems	移动位置	1: 从钞箱出来 2: 移动到钞箱 4: 移动到传送状态
Positions	指示哪些输出位置可用	0: 默认位置 1: 左边 2: 右边 4: 中间

		64: 上边 128: 下边 2048: 前面 4096: 后面
ItemsTakenSensor	是否支持取走传感器	
IntermediateStacker	是否支持暂存	
CashBox	指示柜员是否分配现金箱	仅当现金箱类型为 TellerBill 或 TellerCoin 时状态可用
SafeDoor	是否支持打开保险门命令	
RetractStackerActions		1: 出口位置 2: 回收 4: 拒绝 8: 不支持 16: 现金箱
RetractTransportActions		1: 出口位置 2: 回收 4: 拒绝 8: 不支持 16: 现金箱
RetractAreas	指示回收区域	1: 回收箱 2: 传送装置 4: 暂存 8: 拒绝箱 16: 不支持 32: 现金箱
ShutterControl	是否 sp 自动控制闸门	true: sp 控制; false: 应用控制
Shutter	是否支持打开、关闭闸门命令	
Compound	是否组合设备	
MaxDispenseItems	一次最大发放现金数量	
Type	自助机具类型	0: 柜员值守现金 1: 自助现金 2: 柜员值守硬币 4: 自助硬币
Class	逻辑设备分类	

#### 16.6 getCashUnitInfoAsync()

函数说明：获取钞箱信息

返回信息：

```
{
  "List": [
    {
      "RetractedCount": 0,
      "PresentedCount": 0,
```

```
"DispensedCount": 0,
"Physical": [
  {
    "RetractedCount": 0,
    "PresentedCount": 0,
    "DispensedCount": 0,
    "HardwareSensor": true,
    "PStatus": 0,
    "Maximum": 3000,
    "RejectCount": 0,
    "Count": 0,
    "InitialCount": 0,
    "UnitID": "PHTRA",
    "PhysicalPositionName": "PHTRA"
  }
],
"NumPhysicalCUs": 1,
"Status": 0,
"AppLock": false,
"Maximum": 3000,
"Minimum": 0,
"RejectCount": 0,
"Count": 0,
"InitialCount": 0,
"Values": 0,
"CurrencyID": "CNY",
"UnitID": "LOG1",
"CashUnitName": "LOG1",
"Type": 6,
"Number": 1
},
{
  "RetractedCount": 0,
  "PresentedCount": 0,
  "DispensedCount": 0,
  "Physical": [
    {
      "RetractedCount": 0,
      "PresentedCount": 0,
      "DispensedCount": 0,
      "HardwareSensor": true,
      "PStatus": 4,
      "Maximum": 3000,
      "RejectCount": 0,
```



```
        "Count": 0,
        "InitialCount": 0,
        "UnitID": "PH1",
        "PhysicalPositionName": "PhyPos1"
    }
],
    "NumPhysicalCUs": 1,
    "Status": 4,
    "AppLock": false,
    "Maximum": 3000,
    "Minimum": 20,
    "RejectCount": 0,
    "Count": 0,
    "InitialCount": 0,
    "Values": 100,
    "CurrencyID": "CNYd",
    "UnitID": "LOG2",
    "CashUnitName": "LOG2",
    "Type": 12,
    "Number": 2
},
{
    "RetractedCount": 0,
    "PresentedCount": 0,
    "DispensedCount": 0,
    "Physical": [
        {
            "RetractedCount": 0,
            "PresentedCount": 0,
            "DispensedCount": 0,
            "HardwareSensor": true,
            "PStatus": 3,
            "Maximum": 3000,
            "RejectCount": 0,
            "Count": 59,
            "InitialCount": 0,
            "UnitID": "PH2",
            "PhysicalPositionName": "PhyPos2"
        }
    ],
    "NumPhysicalCUs": 1,
    "Status": 3,
    "AppLock": false,
    "Maximum": 3000,
```

```
"Minimum": 20,
"RejectCount": 0,
"Count": 59,
"InitialCount": 0,
"Values": 50,
"CurrencyID": "CNY2",
"UnitID": "LOG3",
"CashUnitName": "LOG3",
"Type": 12,
"Number": 3
},
{
  "RetractedCount": 0,
  "PresentedCount": 0,
  "DispensedCount": 0,
  "Physical": [
    {
      "RetractedCount": 0,
      "PresentedCount": 0,
      "DispensedCount": 0,
      "HardwareSensor": true,
      "PStatus": 4,
      "Maximum": 3000,
      "RejectCount": 0,
      "Count": 0,
      "InitialCount": 0,
      "UnitID": "PH3",
      "PhysicalPositionName": "PhyPos3"
    }
  ],
  "NumPhysicalCUs": 1,
  "Status": 4,
  "AppLock": false,
  "Maximum": 3000,
  "Minimum": 20,
  "RejectCount": 0,
  "Count": 0,
  "InitialCount": 0,
  "Values": 20,
  "CurrencyID": "CNY\u0014",
  "UnitID": "LOG4",
  "CashUnitName": "LOG4",
  "Type": 12,
  "Number": 4
}
```

```

    },
    {
      "RetractedCount": 0,
      "PresentedCount": 0,
      "DispensedCount": 0,
      "Physical": [
        {
          "RetractedCount": 0,
          "PresentedCount": 0,
          "DispensedCount": 0,
          "HardwareSensor": true,
          "PStatus": 4,
          "Maximum": 3000,
          "RejectCount": 0,
          "Count": 0,
          "InitialCount": 0,
          "UnitID": "PH4",
          "PhysicalPositionName": "PhyPos4"
        }
      ],
      "NumPhysicalCUs": 1,
      "Status": 4,
      "AppLock": false,
      "Maximum": 3000,
      "Minimum": 20,
      "RejectCount": 0,
      "Count": 0,
      "InitialCount": 0,
      "Values": 10,
      "CurrencyID": "CNY\n",
      "UnitID": "LOG5",
      "CashUnitName": "LOG5",
      "Type": 12,
      "Number": 5
    }
  ],
  "Count": 5,
  "TellerID": 0
}

```

字段说明：

字段	字段含义	备注
List	钞箱集合	参考 List 字段说明
Count	钞箱数量	List 的数量
TellerID	柜员 id	此命令中未使用值为 0

## List 字段说明（钞箱字段说明）

字段	字段含义	备注
RetractedCount	回收数量	
PresentedCount	所有接入物理钞箱呈现给客户的数量	
DispensedCount	所有接入物理钞箱的发放数量	
Physical	物理钞箱集合	参考 Physical 字段说明
NumPhysicalCUs	物理钞箱数量	
Status	钞箱状态	0: 钞箱状态良好 1: 钞箱已满 2: 钞箱将满 3: 钞箱将空 4: 钞箱已空或者残留不够将来分配操作 5: 钞箱未被激活 6: 钞箱丢失 7: 钞箱指定值（Values）不可用 8: 钞箱未被校正 9: 钞箱不处于激活状态
AppLock	锁定钞箱标识	true:锁定钞箱，不能发放现金
Maximum	产生高阈值事件的数量	
Minimum	产生低阈值事件的数量	
RejectCount	放入废钞箱的属性	
Count	当前计数，含义依赖钞箱类型	
InitialCount	钞箱初始数量	
Values	钞箱单项值	最小分配单元
CurrencyID	货币类型	人民币 CNY
UnitID	钞箱标识符	
CashUnitName	钞箱名称	
Type	钞箱类型	1: 不可使用，典型为没有钞箱 2: 拒绝钞箱 3: 纸币钞箱 4: 硬币箱 5: 硬币发放箱 6: 回收钞箱 7: 票据箱 8: 文档箱 11: 补充箱 12: 循环箱
Number	钞箱索引	从 1 开始递增

## Physical 字段说明

字段	字段含义	备注
RetractedCount	回收数量	

PresentedCount	呈现给客户数量	
DispensedCount	发放的数量	
HardwareSensor	是否触发传感器事件	
PStatus	物理钞箱状态	0: 钞箱状态良好 1: 钞箱已满 2: 钞箱将满 3: 钞箱将空 4: 钞箱已空或者残留不够将来分配操作 5: 钞箱未被激活 6: 钞箱丢失 7: 钞箱指定值（Values）不可用 8: 钞箱未被校正 9: 钞箱不处于激活状态
Maximum	最大存储数量	
RejectCount	废钞数量	
Count	当前计数	
InitialCount	初始数量	
UnitID	钞箱标识符	
PhysicalPositionName	物理位置名称	

#### 16.7 getTellerInfoAsync(tellerInfo)

函数说明：获取针对每种货币的柜员信息

输入参数：null 为返回所有，或以下参数：

```
{
  "CurrencyID": "CNY",
  "TellerID": 0
}
```

参数说明：

字段	字段含义	备注
CurrencyID	货币类型	
TellerID	柜员 id	

返回信息：

```
{
  "TellerTotals": [
    {
      "CashBoxDispensed": 0,
      "CashBoxReceived": 0,
      "CoinsDispensed": 0,
      "CoinsReceived": 0,
      "ItemsDispensed": 0,
      "ItemsReceived": 0,
      "CurrencyID": "CNY"
    }
  ]
}
```

```

    ],
    "OutputPosition": 0,
    "InputPosition": 0,
    "TellerID": 0
}

```

参数说明：

字段	字段含义	备注
TellerTotals	柜员信息统计的集合	参考 TellerTotal 字段说明
OutputPosition	分配的现金输出位置	
InputPosition	分配的现金输入位置	
TellerID	柜员 Id	

TellerTotal 字段说明

字段	字段含义	备注
CashBoxDispensed	现金盒里分配的货币的总金额	
CashBoxReceived	现金盒里接受的货币的总金额	
CoinsDispensed	分发的硬币的总金额，	
CoinsReceived	接受的硬币的总金额	
ItemsDispensed	分发货币的总金额	
ItemsReceived	受货币的总金额	
CurrencyID	货币类型	

## 16.8 getCurrencyExpAsync()

函数说明：获取已知的每种货币的描述信息

```

[
  {
    "Exponent": 0,
    "CurrencyID": "CNY"
  },
  {
    "Exponent": 0,
    "CurrencyID": "CBB"
  }
]

```

参数说明：

字段	字段含义	备注
Exponent	汇率指数	
CurrencyID	货币 Id	

## 16.9 getMixTypesAsync()

函数说明：获取混合类型

返回信息：

```

[
  {
    "Name": "Min Notes",

```

```
"SubType": 1,
"MixType": 1,
"MixNumber": 1
},
{
  "Name": "Equ Emptying",
  "SubType": 2,
  "MixType": 1,
  "MixNumber": 2
},
{
  "Name": "Max Number",
  "SubType": 3,
  "MixType": 1,
  "MixNumber": 3
},
{
  "Name": "Max Total",
  "SubType": 32769,
  "MixType": 1,
  "MixNumber": 32769
},
{
  "Name": "CU Min Notes PU Max Remain Number",
  "SubType": 32770,
  "MixType": 1,
  "MixNumber": 32770
},
{
  "Name": "CU Equ Emptying PU Max Remain Number",
  "SubType": 32771,
  "MixType": 1,
  "MixNumber": 32771
},
{
  "Name": "CU Max Number PU Max Remain Number",
  "SubType": 32772,
  "MixType": 1,
  "MixNumber": 32772
},
{
  "Name": "CU Max Total PU Max Remain Number",
  "SubType": 32773,
  "MixType": 1,
```

```

    "MixNumber": 32773
  }
]

```

参数说明：

字段	字段含义	备注
Name	识别算法和结构表的数字定义	
SubType	包含识别算法或结构表类型的厂商定义的数值定义	1：选择需要的钞票的最小可能数的混合 2：尽可能的在相同比例下先清空钞箱 3：不同的逻辑钞箱被使用的最大数
MixType	设定是否复杂算法或结构表混合类型	1：混合算法 2：混合表
MixNumber	识别算法和结构表的数字定义	2：混合表 301：配额 302：发放

#### 16.10 getMixTableAsync(mixNumber)

函数说明：获取混合表信息

输入参数：混合表算法数字集合

返回信息：

```

{
  "MixRows": [
    {
      "Mixture": [
        1
      ],
      "Amount": 0
    }
  ],
  "MixHeader": [
    1
  ],
  "Cols": 0,
  "Rows": 0,
  "Name": null,
  "MixNumber": 0
}

```

字段说明：

字段	字段含义	备注
MixRows	混合表结构集合	参考 MixRows 字段说明
MixHeader	钞票相关值集合	
Cols	混合表列数	
Rows	混合表行数	
Name	表名称	
MixNumber	使用的混合表或者算法	



#### MixRows 字段说明

字段	字段含义	备注
Mixture	被使用的混合钞票面额集合	总和为 Amount
Amount	通过此混合行的配额总数	

#### 16.11 getPresentStatusAsync(position)

函数说明：获取最近一次尝试分配或者客户操作项的状态信息

输入参数：position:

字段	字段含义	备注
	呈现给客户的位置信息	0: 默认位置 1: 左边 2: 右边 4: 中间 64: 上边 128: 下边 2048: 前面 4096: 后面

返回信息：

```
{
  "Extra": {},
  "PresentState": 2,
  "Denomination": {
    "CashBox": 0,
    "Values": [],
    "Count": 0,
    "Amount": 100,
    "CurrencyID": null
  }
}
```

字段说明：

字段	字段含义	备注
Extra	厂商自定义信息	键值对集合
PresentState	最后分配的操作状态	1: 客户可以获得钞票 2: 客户不能获得钞票 3: 状态未知
Denomination	配额信息	参考 Denomination 字段说明

Denomination 字段说明：

字段	字段含义	备注
CashBox	将被从柜员钞箱分配的金额	
Values	从每一个钞箱带走的金额	
Count	Values 的数量	
Amount	配额的金额	
CurrencyID	货币类型	

16.12     denominateAsync(denominate)

函数说明： 配额

输入参数：

```
{
  "Denomination": {
    "CashBox": 1,
    "Values": [
      1
    ],
    "Count": 1,
    "Amount": 100,
    "CurrencyID": "CNY"
  },
  "MixNumber": 1,
  "TellerID": 0
}
```

参数说明：

字段	字段含义	备注
Denomination	配额信息	参考 Denomination 字段说明
MixNumber	使用的混合表或者算法	
TellerID	柜员 Id	

Denomination 字段说明：

字段	字段含义	备注
CashBox	将被从柜员钞箱分配的金额	
Values	从每一个钞箱带走的金额	
Count	Values 的数量	
Amount	配额的金额	
CurrencyID	货币类型	

返回信息：

```
{
  "CashBox": 0,
  "Values": [],
  "Count": 0,
  "Amount": 0,
  "CurrencyID": null
}
```

字段说明：

字段	字段含义	备注
CashBox	将被从柜员钞箱分配的金额	
Values	从每一个钞箱带走的金额	
Count	Values 的数量	
Amount	配额的金额	
CurrencyID	货币类型	

16.13     dispenseAsync(dispense)

函数说明：发放现金

输入参数：

```
{
  "Denomination": {
    "CashBox": 0,
    "Values": [],
    "Count": 0,
    "Amount": 0,
    "CurrencyID": null
  },
  "Present": false,
  "Position": 0,
  "MixNumber": 0,
  "TellerID": 0
}
```

参数说明：

字段	字段含义	备注
Denomination	配额信息	参考 Denomination 字段说明
Present	是否呈现给客户	
Position	输出位置	0：默认位置 1：左边 2：右边 4：中间 64：上边 128：下边 2048：前面 4096：后面
MixNumber	使用的混合表或者算法	
TellerID	柜员 Id	

Denomination 字段说明：

字段	字段含义	备注
CashBox	将被从柜员钞箱分配的金额	
Values	从每一个钞箱带走的金额	
Count	Values 的数量	
Amount	配额的金额	
CurrencyID	货币类型	

返回信息：

```
{
  "CashBox": 0,
  "Values": [],
  "Count": 0,
```

```

    "Amount": 0,
    "CurrencyID": null
}

```

Denomination 字段说明:

字段	字段含义	备注
CashBox	将被从柜员钞箱分配的金额	
Values	从每一个钞箱带走的金额	
Count	Values 的数量	
Amount	配额的金额	
CurrencyID	货币类型	

#### 16.14 countAsync(physicalCU)

函数说明: 清空钞箱到指定出钞口

输入参数:

```

{
    "PhysicalPositionName": "Center",
    "Position": 4,
    "EmptyAll": true
}

```

参数说明:

字段	字段含义	备注
PhysicalPositionName	物理位置名称	
Position	输出钱币位置, 取值为单一值或者多值或值	0: 默认位置 1: 左边 2: 右边 4: 中间 64: 上边 128: 下边 2048: 前面 4096: 后面
EmptyAll	是否清空所有钞箱	

返回信息:

```

{
    "ExcuteResult": 0,
    "IsSuccess": true
}

```

字段说明:

字段	字段含义	备注
ExcuteResult	函数执行结果	0: 函数成功执行
IsSuccess	函数执行是否成功	true: 执行成功 false: 执行失败

#### 16.15 presentAsync(position)

函数说明：移动现金到出口位置

输入参数：出口位置

字段	字段含义	备注
	呈现给客户的位置信息	0: 默认位置 1: 左边 2: 右边 4: 中间 64: 上边 128: 下边 2048: 前面 4096: 后面

返回信息：

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明：

字段	字段含义	备注
ExcuteResult	函数执行结果	0: 函数成功执行
IsSuccess	函数执行是否成功	true: 执行成功 false: 执行失败

#### 16.16 rejectAsync()

函数说明：从暂存移动现金到回收单元

返回信息：

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明：

字段	字段含义	备注
ExcuteResult	函数执行结果	0: 函数成功执行
IsSuccess	函数执行是否成功	true: 执行成功 false: 执行失败

#### 16.17 retractAsync(retract)

函数说明：回收

输入参数：

```
{
  "Index": 0,
  "RetractArea": 0,
  "OutputPosition": 0
}
```

参数说明：

字段	字段含义	备注
Index	回收钞箱索引	从 1 开始
RetractArea	回收位置	1: 回收箱 2: 传送装置 4: 暂存 8: 废钞箱 16: 不支持 32: 现金箱
OutputPosition	输出位置	0: 默认位置 1: 左边 2: 右边 4: 中间 64: 上边 128: 下边 2048: 前面 4096: 后面

返回信息：3.20 版本及以上可用，3.20 以下版本数值无效。

```
{
  "ItemNumber": [
    {
      "Number": 0,
      "Count": 0,
      "Release": 0,
      "Values": 0,
      "CurrencyID": null
    }
  ],
  "NumOfItemNumbers": 0
}
```

参数说明：

字段	字段含义	备注
ItemNumber	回收钞币项目的集合	参考 ItemNumber 字段说明
NumOfItemNumbers	ItemNumber 的数量	

ItemNumber 字段说明：

字段	字段含义	备注
Number	逻辑钞箱号	
Count	ItemNumber 的数量	
Release	钞票类型版本	
Values	回收钞币值	如果回收项目未知，值为 0
CurrencyID	货币类型	

#### 16.18 openShutterAsync(position)

函数说明：打开闸门

输入参数：闸门位置

字段	字段含义	备注
	闸门位置	0：默认位置 1：左边 2：右边 4：中间 64：上边 128：下边 2048：前面 4096：后面

输出参数：

```
{  
    "ExcuteResult": 0,  
    "IsSuccess": true  
}
```

字段说明：

字段	字段含义	备注
ExcuteResult	函数执行结果	0：函数成功执行
IsSuccess	函数执行是否成功	true：执行成功 false：执行失败

#### 16.19 closeShutterAsync(position)

函数说明：关闭闸门

输入参数：闸门位置

字段	字段含义	备注
	闸门位置	0：默认位置 1：左边 2：右边 4：中间 64：上边 128：下边 2048：前面 4096：后面

返回信息：函数执行结果

```
{  
    "ExcuteResult": 0,  
    "IsSuccess": true  
}
```

字段说明：

字段	字段含义	备注
ExcuteResult	函数执行结果	0：函数成功执行
IsSuccess	函数执行是否成功	true：执行成功 false：执行失败

## 16.20 setTellerInfoAsync(tellerUpdate)

函数说明：设置柜员信息

输入参数：

```
{
  "TellerDetails": {
    "TellerTotals": [],
    "OutputPosition": 0,
    "InputPosition": 0,
    "TellerID": 0
  },
  "Action": 0
}
```

字段说明：

字段	字段含义	备注
TellerDetails	柜员信息明细	参考 TellerDetails 字段说明
Action	将被执行的指令	1：创建柜员信息 2：修改柜员信息 3：删除柜员信息

TellerDetails 字段说明：

字段	字段含义	备注
TellerTotals	柜员统计明细	参考 TellerTotals 字段说明
OutputPosition	输出位置	
InputPosition	输入位置	
TellerID	柜员 Id	

TellerTotal 字段说明

字段	字段含义	备注
CashBoxDispensed	现金盒里分配的货币的总金额	
CashBoxReceived	现金盒里接受的货币的总金额	
CoinsDispensed	分发的硬币的总金额，	
CoinsReceived	接受的硬币的总金额	
ItemsDispensed	分发货币的总金额	
ItemsReceived	受货币的总金额	
CurrencyID	货币类型	

返回信息：函数执行结果

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明：

字段	字段含义	备注
ExcuteResult	函数执行结果	0：函数成功执行
IsSuccess	函数执行是否成功	true：执行成功 false：执行失败



## 16.21 setCashUnitInfoAsync(cashInfo)

函数说明：设置钞箱的状态和数据

输出参数：

```
{
  "List": [
    {
      "RetractedCount": 0,
      "PresentedCount": 0,
      "DispensedCount": 0,
      "Physical": [
        {
          "RetractedCount": 0,
          "PresentedCount": 0,
          "DispensedCount": 0,
          "HardwareSensor": false,
          "PStatus": 0,
          "Maximum": 0,
          "RejectCount": 0,
          "Count": 0,
          "InitialCount": 0,
          "UnitID": null,
          "PhysicalPositionName": null
        }
      ],
      "NumPhysicalCUs": 0,
      "Status": 0,
      "AppLock": false,
      "Maximum": 0,
      "Minimum": 0,
      "RejectCount": 0,
      "Count": 0,
      "InitialCount": 0,
      "Values": 0,
      "CurrencyID": null,
      "UnitID": null,
      "CashUnitName": null,
      "Type": 0,
      "Number": 0
    }
  ],
  "Count": 0,
  "TellerID": 0
}
```

字段说明：

字段	字段含义	备注
List	钞箱集合	参考 List 字段说明
Count	钞箱数量	List 的数量
TellerID	柜员 id	此命令中未使用值为 0

List 字段说明（钞箱字段说明）

字段	字段含义	备注
RetractedCount	回收数量	
PresentedCount	所有接入物理钞箱呈现给客户的数量	
DispensedCount	所有接入物理钞箱的发放数量	
Physical	物理钞箱集合	参考 Physical 字段说明
NumPhysicalCUs	物理钞箱数量	
Status	钞箱状态	0: 钞箱状态良好 1: 钞箱已满 2: 钞箱将满 3: 钞箱将空 4: 钞箱已空或者残留不够将来分配操作 5: 钞箱未被激活 6: 钞箱丢失 7: 钞箱指定值（Values）不可用 8: 钞箱未被校正 9: 钞箱不处于激活状态
AppLock	锁定钞箱标识	true:锁定钞箱，不能发放现金
Maximum	产生高阈值事件的数量	
Minimum	产生低阈值事件的数量	
RejectCount	放入废钞箱的属性	
Count	当前计数，含义依赖钞箱类型	
InitialCount	钞箱初始数量	
Values	钞箱单项值	最小分配单元
CurrencyID	货币类型	人民币 CNY
UnitID	钞箱标识符	
CashUnitName	钞箱名称	
Type	钞箱类型	1: 不可使用，典型为没有钞箱 2: 拒绝钞箱 3: 纸币钞箱 4: 硬币箱 5: 硬币发放箱 6: 回收钞箱 7: 票据箱

		8: 文档箱 11: 补充箱 12: 循环箱
Number	钞箱索引	从 1 开始递增

Physical 字段说明

字段	字段含义	备注
RetractedCount	回收数量	
PresentedCount	呈现给客户数量	
DispensedCount	发放的数量	
HardwareSensor	是否触发传感器事件	
PStatus	物理钞箱状态	0: 钞箱状态良好 1: 钞箱已满 2: 钞箱将满 3: 钞箱将空 4: 钞箱已空或者残留不够将来分配操作 5: 钞箱未被激活 6: 钞箱丢失 7: 钞箱指定值 (Values) 不可用 8: 钞箱未被校正 9: 钞箱不处于激活状态
Maximum	最大存储数量	
RejectCount	废钞数量	
Count	当前计数	
InitialCount	初始数量	
UnitID	钞箱标识符	
PhysicalPositionName	物理位置名称	

返回信息：函数执行结果

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明：

字段	字段含义	备注
ExcuteResult	函数执行结果	0: 函数成功执行
IsSuccess	函数执行是否成功	true: 执行成功 false: 执行失败

## 16.22 startExchangeAsync(startEx)

函数说明：设置设备进入交换状态，可以进行清空、补给、取走或替换

输出参数：

```
{
  "CUNumList": [],
  "Count": 0,
```

```
"TellerID": 0,  
"ExchangeType": 0  
}
```

字段说明：

字段	字段含义	备注
CUNumList	逻辑钞箱数集合	例如：[1,2,3,4,5,6]激活 1、2、3、4、5、6 号钞箱。
Count	CUNumList 数量	
TellerID	柜员 Id	
ExchangeType	设置钞箱激活的操作类型	1：手动或回收钞箱进行补满或清空 2：钱币将会从补充容器移动到钞箱

返回信息：

```
{  
  "List": [  
    {  
      "RetractedCount": 0,  
      "PresentedCount": 0,  
      "DispensedCount": 0,  
      "Physical": [  
        {  
          "RetractedCount": 0,  
          "PresentedCount": 0,  
          "DispensedCount": 0,  
          "HardwareSensor": false,  
          "PStatus": 0,  
          "Maximum": 0,  
          "RejectCount": 0,  
          "Count": 0,  
          "InitialCount": 0,  
          "UnitID": null,  
          "PhysicalPositionName": null  
        }  
      ],  
      "NumPhysicalCUs": 0,  
      "Status": 0,  
      "AppLock": false,  
      "Maximum": 0,  
      "Minimum": 0,  
      "RejectCount": 0,  
      "Count": 0,  
      "InitialCount": 0,  
      "Values": 0,  
      "CurrencyID": null,  
    }  
  ],  
  "NumPhysicalCUs": 0,  
  "Status": 0,  
  "AppLock": false,  
  "Maximum": 0,  
  "Minimum": 0,  
  "RejectCount": 0,  
  "Count": 0,  
  "InitialCount": 0,  
  "Values": 0,  
  "CurrencyID": null,  
}
```

```

        "UnitID": null,
        "CashUnitName": null,
        "Type": 0,
        "Number": 0
    }
],
"Count": 0,
"TellerID": 0
}

```

字段说明：

字段	字段含义	备注
List	钞箱集合	参考 List 字段说明
Count	钞箱数量	List 的数量
TellerID	柜员 id	此命令中未使用值为 0

List 字段说明（钞箱字段说明）

字段	字段含义	备注
RetractedCount	回收数量	
PresentedCount	所有接入物理钞箱呈现给客户的数量	
DispensedCount	所有接入物理钞箱的发放数量	
Physical	物理钞箱集合	参考 Physical 字段说明
NumPhysicalCUs	物理钞箱数量	
Status	钞箱状态	0: 钞箱状态良好 1: 钞箱已满 2: 钞箱将满 3: 钞箱将空 4: 钞箱已空或者残留不够将来分配操作 5: 钞箱未被激活 6: 钞箱丢失 7: 钞箱指定值（Values）不可用 8: 钞箱未被校正 9: 钞箱不处于激活状态
AppLock	锁定钞箱标识	true:锁定钞箱，不能发放现金
Maximum	产生高阈值事件的数量	
Minimum	产生低阈值事件的数量	
RejectCount	放入废钞箱的属性	
Count	当前计数，含义依赖钞箱类型	
InitialCount	钞箱初始数量	
Values	钞箱单项值	最小分配单元
CurrencyID	货币类型	人民币 CNY
UnitID	钞箱标识符	

CashUnitName	钞箱名称	
Type	钞箱类型	1: 不可使用, 典型为没有钞箱 2: 拒绝钞箱 3: 纸币钞箱 4: 硬币箱 5: 硬币发放箱 6: 回收钞箱 7: 票据箱 8: 文档箱 11: 补充箱 12: 循环箱
Number	钞箱索引	从 1 开始递增

#### Physical 字段说明

字段	字段含义	备注
RetractedCount	回收数量	
PresentedCount	呈现给客户的数量	
DispensedCount	发放的数量	
HardwareSensor	是否触发传感器事件	
PStatus	物理钞箱状态	0: 钞箱状态良好 1: 钞箱已满 2: 钞箱将满 3: 钞箱将空 4: 钞箱已空或者残留不够将来分配操作 5: 钞箱未被激活 6: 钞箱丢失 7: 钞箱指定值 (Values) 不可用 8: 钞箱未被校正 9: 钞箱不处于激活状态
Maximum	最大存储数量	
RejectCount	废钞数量	
Count	当前计数	
InitialCount	初始数量	
UnitID	钞箱标识符	
PhysicalPositionName	物理位置名称	

#### 16.23 endExchangeAsync(cashInfo)

函数说明: 结束交换状态

输入参数:

```
{
  "List": [
    {
      "RetractedCount": 0,
      "PresentedCount": 0,
      "DispensedCount": 0,
```

```

    "Physical": [
      {
        "RetractedCount": 0,
        "PresentedCount": 0,
        "DispensedCount": 0,
        "HardwareSensor": false,
        "PStatus": 0,
        "Maximum": 0,
        "RejectCount": 0,
        "Count": 0,
        "InitialCount": 0,
        "UnitID": null,
        "PhysicalPositionName": null
      }
    ],
    "NumPhysicalCUs": 0,
    "Status": 0,
    "AppLock": false,
    "Maximum": 0,
    "Minimum": 0,
    "RejectCount": 0,
    "Count": 0,
    "InitialCount": 0,
    "Values": 0,
    "CurrencyID": null,
    "UnitID": null,
    "CashUnitName": null,
    "Type": 0,
    "Number": 0
  }
},
"Count": 0,
"TellerID": 0
}

```

字段说明：

字段	字段含义	备注
List	钞箱集合	参考 List 字段说明
Count	钞箱数量	List 的数量
TellerID	柜员 id	此命令中未使用值为 0

List 字段说明（钞箱字段说明）

字段	字段含义	备注
RetractedCount	回收数量	
PresentedCount	所有接入物理钞箱呈现给客户的数量	

DispensedCount	所有接入物理钞箱的发放数量	
Physical	物理钞箱集合	参考 Physical 字段说明
NumPhysicalCUs	物理钞箱数量	
Status	钞箱状态	0: 钞箱状态良好 1: 钞箱已满 2: 钞箱将满 3: 钞箱将空 4: 钞箱已空或者残留不够将来分配操作 5: 钞箱未被激活 6: 钞箱丢失 7: 钞箱指定值 (Values) 不可用 8: 钞箱未被校正 9: 钞箱不处于激活状态
AppLock	锁定钞箱标识	true:锁定钞箱，不能发放现金
Maximum	产生高阈值事件的数量	
Minimum	产生低阈值事件的数量	
RejectCount	放入废钞箱的属性	
Count	当前计数，含义依赖钞箱类型	
InitialCount	钞箱初始数量	
Values	钞箱单项值	最小分配单元
CurrencyID	货币类型	人民币 CNY
UnitID	钞箱标识符	
CashUnitName	钞箱名称	
Type	钞箱类型	1: 不可使用，典型为没有钞箱 2: 拒绝钞箱 3: 纸币钞箱 4: 硬币箱 5: 硬币发放箱 6: 回收钞箱 7: 票据箱 8: 文档箱 11: 补充箱 12: 循环箱
Number	钞箱索引	从 1 开始递增

#### Physical 字段说明

字段	字段含义	备注
RetractedCount	回收数量	
PresentedCount	呈现给客户数量	
DispensedCount	发放的数量	
HardwareSensor	是否触发传感器事件	
PStatus	物理钞箱状态	0: 钞箱状态良好



		1: 钞箱已满 2: 钞箱将满 3: 钞箱将空 4: 钞箱已空或者残留不够将来分配操作 5: 钞箱未被激活 6: 钞箱丢失 7: 钞箱指定值（Values）不可用 8: 钞箱未被校正 9: 钞箱不处于激活状态
Maximum	最大存储数量	
RejectCount	废钞数量	
Count	当前计数	
InitialCount	初始数量	
UnitID	钞箱标识符	
PhysicalPositionName	物理位置名称	

返回信息：函数执行结果

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明：

字段	字段含义	备注
ExcuteResult	函数执行结果	0：函数成功执行
IsSuccess	函数执行是否成功	true：执行成功 false：执行失败

#### 16.24 openSafeDoorAsync()

函数说明：打开安全门

返回信息：函数执行结果

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明：

字段	字段含义	备注
ExcuteResult	函数执行结果	0：函数成功执行
IsSuccess	函数执行是否成功	true：执行成功 false：执行失败

#### 16.25 calibrateCashUnitAsync(calibrate)

函数说明：矫正物理单元接入到逻辑单元

输入参数：

```
{
```

```

"Position": {
  "OutputPosition": 0,
  "RetractArea": null,
  "Number": 0
},
"NumOfBills": 0,
"Number": 0
}

```

字段说明：

字段	字段含义	备注
Position	校正过程中钞票被移到的位置	0：默认位置 1：左边 2：右边 4：中间 64：上边 128：下边 2048：前面 4096：后面
NumOfBills	校正过程中被分配的钱币数	
Number	逻辑钞箱号	

返回信息：

```

{
  "Position": {
    "OutputPosition": 0,
    "RetractArea": null,
    "Number": 0
  },
  "NumOfBills": 0,
  "Number": 0
}

```

字段说明：

字段	字段含义	备注
Position	校正过程中钞票被移到的位置	0：默认位置 1：左边 2：右边 4：中间 64：上边 128：下边 2048：前面 4096：后面
NumOfBills	校正过程中被分配的钱币数	
Number	逻辑钞箱号	

16.26     setMixTableAsync(mixTable)

函数说明：设置混合表

输入参数：

```
{
  "MixRows": [
    {
      "Mixture": [],
      "Amount": 0
    }
  ],
  "MixHeader": [],
  "Cols": 0,
  "Rows": 0,
  "Name": null,
  "MixNumber": 0
}
```

字段说明：

字段	字段含义	备注
MixRows	混合表结构集合	参考 MixRows 字段说明
MixHeader	钞票相关值集合	
Cols	混合表列数	
Rows	混合表行数	
Name	表名称	
MixNumber	使用的混合表或者算法	

MixRows 字段说明

字段	字段含义	备注
Mixture	被使用的混合钞票面额集合	总和为 Amount
Amount	通过此混合行的配额总数	

返回信息：函数执行结果

```
{
  "ExcuteResult": 0,
  "IsSuccess": true
}
```

字段说明：

字段	字段含义	备注
ExcuteResult	函数执行结果	0：函数成功执行
IsSuccess	函数执行是否成功	true：执行成功 false：执行失败

16.27     resetAsync(itemPosition)

函数说明：重置设备

输入参数：

```
{
```

```

"OutputPosition": 0,
"RetractArea":
{
  "Index":1,
  "RetractArea":1,
  "OutputPosition":
},
"Number": 0
}

```

参数说明：

字段	字段含义	备注
Index	钞箱索引	从 1 开始
RetractArea	回收位置	仅当钞箱类型为回收类型时可用,参考 RetractArea 字段说明
OutputPosition	输出位置	0: 默认位置 1: 左边 2: 右边 4: 中间 64: 上边 128: 下边 2048: 前面 4096: 后面

RetractArea 字段说明

字段	字段含义	备注
Index	钞箱索引	从 1 开始
RetractArea	回收位置	1: 回收箱 2: 传送装置 4: 暂存 8: 现金箱 16: 不支持 32: 出口位置
OutputPosition	输出位置	0: 默认位置 1: 左边 2: 右边 4: 中间 64: 上边 128: 下边 2048: 前面 4096: 后面

返回信息：

```

{
  "ExcuteResult": 0,
  "IsSuccess": true
}

```

}

字段说明:

字段	字段含义	备注
ExcuteResult	函数执行结果	0: 函数成功执行
IsSuccess	函数执行是否成功	true: 执行成功 false: 执行失败

#### 16.28 testCashUintsAsync(itemPosition)

函数说明: 用于补充现金的时候测试钞箱

输入参数:

```
{  
  "OutputPosition": 0,  
  "RetractArea":  
  {  
    "Index":1,  
    "RetractArea":1,  
    "OutputPosition":  
  },  
  "Number": 0  
}
```

参数说明:

字段	字段含义	备注
Index	钞箱索引	从 1 开始
RetractArea	回收位置	仅当钞箱类型为回收类型时可用,参考 RetractArea 字段说明
OutputPosition	输出位置	0: 默认位置 1: 左边 2: 右边 4: 中间 64: 上边 128: 下边 2048: 前面 4096: 后面

RetractArea 字段说明

字段	字段含义	备注
Index	钞箱索引	从 1 开始
RetractArea	回收位置	1: 回收箱 2: 传送装置 4: 暂存 8: 现金箱 16: 不支持 32: 出口位置
OutputPosition	输出位置	0: 默认位置 1: 左边

		2: 右边 4: 中间 64: 上边 128: 下边 2048: 前面 4096: 后面
--	--	---

返回信息:

```
{
  "List": [
    {
      "RetractedCount": 0,
      "PresentedCount": 0,
      "DispensedCount": 0,
      "Physical": [
        {
          "RetractedCount": 0,
          "PresentedCount": 0,
          "DispensedCount": 0,
          "HardwareSensor": false,
          "PStatus": 0,
          "Maximum": 0,
          "RejectCount": 0,
          "Count": 0,
          "InitialCount": 0,
          "UnitID": null,
          "PhysicalPositionName": null
        }
      ],
      "NumPhysicalCUs": 0,
      "Status": 0,
      "AppLock": false,
      "Maximum": 0,
      "Minimum": 0,
      "RejectCount": 0,
      "Count": 0,
      "InitialCount": 0,
      "Values": 0,
      "CurrencyID": null,
      "UnitID": null,
      "CashUnitName": null,
      "Type": 0,
      "Number": 0
    }
  ]
}
```

```

    ],
    "Count": 0,
    "TellerID": 0
}

```

字段说明:

字段	字段含义	备注
List	钞箱集合	参考 List 字段说明
Count	钞箱数量	List 的数量
TellerID	柜员 id	此命令中未使用值为 0

List 字段说明（钞箱字段说明）

字段	字段含义	备注
RetractedCount	回收数量	
PresentedCount	所有接入物理钞箱呈现给客户的数量	
DispensedCount	所有接入物理钞箱的发放数量	
Physical	物理钞箱集合	参考 Physical 字段说明
NumPhysicalCUs	物理钞箱数量	
Status	钞箱状态	0: 钞箱状态良好 1: 钞箱已满 2: 钞箱将满 3: 钞箱将空 4: 钞箱已空或者残留不够将来分配操作 5: 钞箱未被激活 6: 钞箱丢失 7: 钞箱指定值（Values）不可用 8: 钞箱未被校正 9: 钞箱不处于激活状态
AppLock	锁定钞箱标识	true:锁定钞箱，不能发放现金
Maximum	产生高阈值事件的数量	
Minimum	产生低阈值事件的数量	
RejectCount	放入废钞箱的属性	
Count	当前计数，含义依赖钞箱类型	
InitialCount	钞箱初始数量	
Values	钞箱单项值	最小分配单元
CurrencyID	货币类型	人民币 CNY
UnitID	钞箱标识符	
CashUnitName	钞箱名称	
Type	钞箱类型	1: 不可使用，典型为没有钞箱 2: 拒绝钞箱 3: 纸币钞箱

		4: 硬币箱 5: 硬币发放箱 6: 回收钞箱 7: 票据箱 8: 文档箱 11: 补充箱 12: 循环箱
Number	钞箱索引	从 1 开始递增

Physical 字段说明

字段	字段含义	备注
RetractedCount	回收数量	
PresentedCount	呈现给客户数量	
DispensedCount	发放的数量	
HardwareSensor	是否触发传感器事件	
PStatus	物理钞箱状态	0: 钞箱状态良好 1: 钞箱已满 2: 钞箱将满 3: 钞箱将空 4: 钞箱已空或者残留不够将来分配操作 5: 钞箱未被激活 6: 钞箱丢失 7: 钞箱指定值（Values）不可用 8: 钞箱未被校正 9: 钞箱不处于激活状态
Maximum	最大存储数量	
RejectCount	废钞数量	
Count	当前计数	
InitialCount	初始数量	
UnitID	钞箱标识符	
PhysicalPositionName	物理位置名称	