

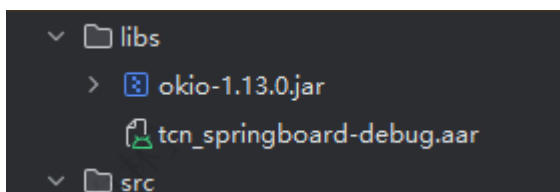
# Spring Machine SDK Description Document

## SDK Integration Steps (Step 1: Provide SDK)

### Development Preparation

#### 1. Environment setup:

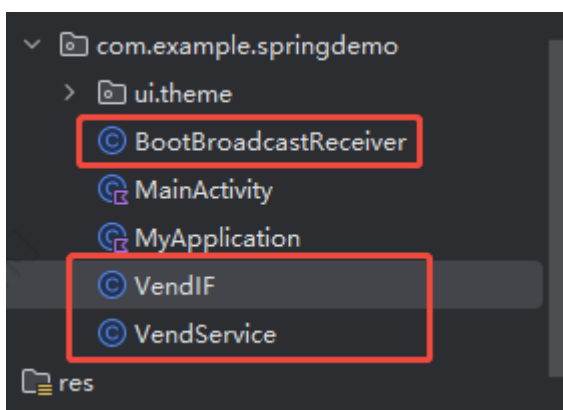
1. After creating a new project in Android studio, copy the package under the libs path in the demo to the target project directory.



2. After copying, add a reference to the build.gradle file of the corresponding Module and synchronize it:

```
dependencies {  
    implementation fileTree(dir: 'libs', include: ['*.jar', '*.aar'])  
}
```

3. The three important components VendService.java, VendIf.java, and BootBroadcastReceiver.java in the dome project should also be added to the target project. Don't forget that the components need to be declared in Androidmanifest.xml.



```

<receiver android:name="com.example.springdemo.BootBroadcastReceiver"
    android:exported="true">
    <intent-filter android:priority="999">
        <action android:name="android.intent.action.BOOT_COMPLETED" />
    </intent-filter>
</receiver>
<service
    android:name="com.example.springdemo.VendService"
    android:exported="true" />

```

At this point, the development environment is complete.

## Actual development

### 2. initialization

After the development environment is set up, it needs to be initialized in the custom application. Create a class that inherits Application and override the onCreate method. Initialize the SDK framework in the onCreate method

```

//首先初始化TcnVendIF
TcnVendIF.getInstance().run {
    init(context: this@MyApplication)
    setConfig()
    startWorkThread()
}

```

Set the motherboard serial port (according to the actual situation, the serial port that the channel is connected to will be initialized and changed)

```

TcnShareUseData.getInstance().boardSerPortFirst = "/dev/ttyS1"

```

### 3. Business Development

All business interfaces of the vending machine are integrated in TcnVendIF.java and called through a singleton.

The process and results of the business are callback through TcnVendIF.VendEventListener  
TcnVendIF.VendEventListener needs to be registered through  
TcnVendIF.getInstance().registerListener(listener).

TcnVendIF.getInstance().unRegisterListener(listener) is to unregister the listener.

example:

In Activity, according to actual development needs, you can register business listeners in the onCreate or onResume cycle phase, and unregister them in the onPause or onDestroy phase.

The business messages of after-sales machines are all encapsulated into VendEventInfo instances. In VendEventInfo, different business results are identified by this parameter:  
m\_iEventID: business code, all business codes can be queried in TcnVendEventID  
m\_lParam1,2,3,4,5 The result of the operation.

## 4. Required permissions

SDK permissions required: (The provided SDK already includes the following permissions)

```
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
2. <uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
3.    <uses-permission android:name="android.permission.WAKE_LOCK" />

<!-- Add permission to access phone location-->
4.    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<!-- Add permission to access phone status-->
5.    <uses-permission android:name="android.permission.READ_PHONE_STATE"/>

6.    <uses-permission android:name="android.permission.CHANGE_NETWORK_STATE" />

7.    <uses-permission android:name="android.permission.INTERNET" />
8.    <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
9.    <uses-permission android:name="android.permission.RESTART_PACKAGES" />
10.   <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
11.   <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
12.   <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />

<!-- Allow CWJ account group to access low-level information-->
13.   <uses-permission android:name="android.permission.CWJ_GROUP" />

<!-- Allow mTweak users to access advanced system permissions-->
14.   <uses-permission android:name="android.permission.MTWEAK_USER" />

<!-- Allow sound wave payment permission if sound wave related hardware is connected -->
14.   <uses-permission android:name="android.permission.RECORD_AUDIO"/>
14.   <uses-permission android:name="android.permission.MODIFY_AUDIO_SETTINGS" />

14.   <uses-permission android:name="android.permission.GET_TASKS" />
14.   <uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW" />
```

## 5. SDK authorization

### 5.1 Scan the QR code with your mobile phone and a

prompt will appear. Fill in the authorization form as required.

中国移动 96% 上午9:15

× SDK授权申请 ...

中吉售货机SDK授权

★ 申请机构 (公司)

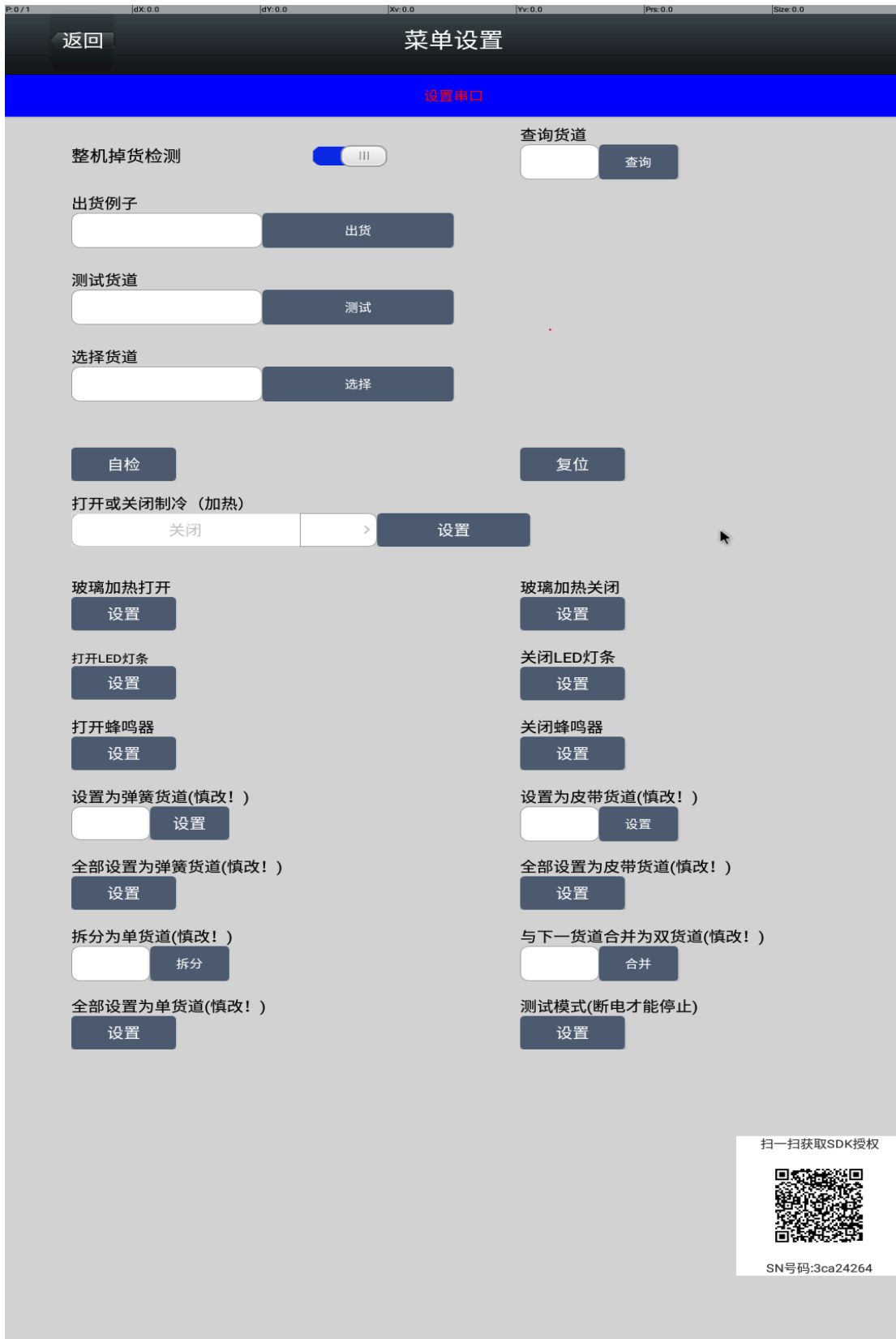
★ 电话号码

★ 对接业务经理

机器型号 5

申请授权

Detailed instructions for using SDK functions



1、 Whole machine drop detection: This switch is mainly used to turn on the detection function of whether goods and cargo fall into the delivery port .



The control function is in the com.tcn.sdk.springdemo.MainAct class, search for menu\_spr\_light\_check

TcnShareUseData.getInstance().setDropSensorCheck(isSwitchOn); //Set the switch status interface

TcnShareUseData.getInstance().isDropSensorCheck() //Get the switch status interface

Interface Description

Parameter name	Parameter Description
isSwitchOn	True turns on the switch
isSwitchOn	False turns off the switch

## 2、 Check the cargo channel: Check whether the cargo channel is faulty



The control function is in the com.tcn.sdk.springdemo.MainAct class, search for menu\_spr\_query\_slot

Enter the channel number and click query

TcnVendIF.getInstance().reqQuerySlotStatus(Integer.valueOf(strParam)); //Query method

Interface Description

Parameter name	Parameter Description
strParam	Enter the channel number

## 3、 Shipping example: simulate a normal purchase and shipment (including channel number, amount, order number, etc.)



The control function is in the com.tcn.sdk.springdemo.MainAct class, search for menu\_spr\_ship\_slot

Enter the channel number and click Ship.

TcnVendIF.getInstance().reqShip(slotNo,shipMethod,amount,tradeNo);//Test shipping interface

Shipping interface description

Parameter name	Parameter Description
slotNo	Cargo lane number
shipMethod	Payment Methods

amount	Payment amount
tradeNo	Order Number

#### 4、 Test cargo channel: Enter the cargo channel number to enable shipment from that channel.



The control function is in the `com.tcn.sdk.springdemo.MainAct` class, search for `menu_spr_ship_slot_test`

Enter the channel number and click Test

```
TcnVendIF.getInstance().reqShipTest(Integer.parseInt(strParam)); //Shipping channel interface
```

Interface Description

Parameter name	Parameter Description
strParam	Enter the channel number

#### 5、 Select aisle: Enter the aisle number to simulate a goods selection function.



The control function is in the `com.tcn.sdk.springdemo.MainAct` class, search for `menu_spr_reqselect`

Enter the channel number and click Select

```
TcnVendIF.getInstance().reqSelectSlotNo(Integer.valueOf(strParam)); //Selection method
```

Interface Description

Parameter name	Parameter Description
strParam	Enter the channel number

#### 6、 Self-examination.



#### 7. Reset: The position used by the machine is restored.



The control function is in the com.tcn.sdk.springdemo.MainAct class, search for menu\_spr\_reset;  
TcnVendIF.getInstance().reqReset(UIComBack.getInstance().getGroupSpringId(strParam))     reset  
method

Interface Description

Parameter name	Parameter Description
strParam	The parameters have been defined, just click the button

## 8. Control the hot and cold switches: select cooling, heating or off settings.

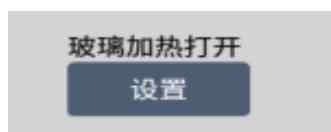


The control function is in the com.tcn.sdk.springdemo.MainAct class, search for menu\_spr\_set\_heat\_cool;  
TcnVendIF.getInstance().reqHeatSpring(-1,Integer.parseInt(temp),Integer.parseInt(startTime),Integer.parseInt(endTime));

Interface Description

Parameter name	Parameter Description
-1	Fixed value does not need to be modified
temp	Enter the set temperature value
startTime	Start time
endTime	End Time

## 9. Glass heating on: turn on glass heating.



The control function is in the com.tcn.sdk.springdemo.MainAct class, search for menu\_spr\_glass\_heat\_enable;

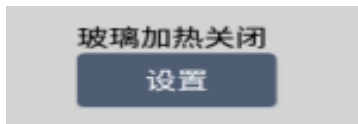
TcnVendIF.getInstance().reqSetGlassHeatEnable( -1 ,true); //Glass heating method

Interface Description

Parameter name	Parameter Description
-1	Fixed value does not need to be modified
true	Turn on the glass heating

## 10. Glass heating off: turn off glass heating.





The control function is in the `com.tcn.sdk.springdemo.MainAct` class, search for `menu_spr_glass_heat_disable`;

`TcnVendIF.getInstance().reqSetGlassHeatEnable( -1 , false );` //Disable heating interface  
Interface Description

Parameter name	Parameter Description
-1	Fixed value does not need to be modified
false	Turn off glass heating

#### 11. Turn on the LED light strip



The control function is in the `com.tcn.sdk.springdemo.MainAct` class, search for `menu_spr_open_led`;

`TcnVendIF.getInstance().reqSetLedOpen( -1 ,true);` //Open the LED light interface  
Interface Description

Parameter name	Parameter Description
-1	Fixed value does not need to be modified
true	Turn on the LED light strip

#### 12. Turn off the LED light strip

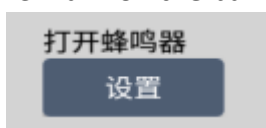


The control function is in the `com.tcn.sdk.springdemo.MainAct` class, search for `menu_spr_close_led`;

`TcnVendIF.getInstance().reqSetLedOpen( -1 , false );` //Close the LED light interface  
Interface Description

Parameter name	Parameter Description
-1	Fixed value does not need to be modified
False	Turn off LED light strip

#### 13. Turn on the buzzer

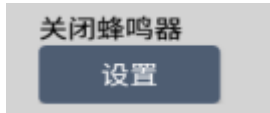


The control function is in the `com.tcn.sdk.springdemo.MainAct` class, search for `menu_spr_buzzer_open`;

TcnVendIF.getInstance().reqSetBuzzerOpen( -1 ,true); //Open the buzzer interface  
Interface Description

Parameter name	Parameter Description
-1	Fixed value does not need to be modified
true	Turn on the buzzer

#### 14. Turn off the buzzer

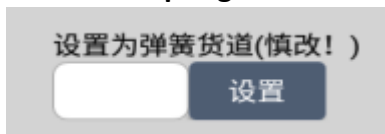


The control function is in the com.tcn.sdk.springdemo.MainAct class, search for menu\_spr\_buzzer\_close;

TcnVendIF.getInstance().reqSetBuzzerOpen( -1 , false ); //Method to close the buzzer  
Interface Description

Parameter name	Parameter Description
-1	Fixed value does not need to be modified
False	Turn off the buzzer

#### 15. Set the spring channel: Set the channel as the spring channel for shipping



The control function is in the com.tcn.sdk.springdemo.MainAct class, search for menu\_spr\_set\_slot\_spring

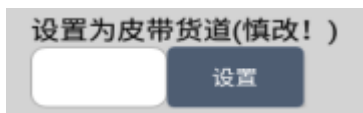
Enter the channel number you want to modify and click Set.

TcnVendIF.getInstance().reqSetSpringSlot(Integer.valueOf(data1)); //Set the spring channel interface

Interface Description

Parameter name	Parameter Description
data1	Enter the channel number

#### 16. Set the belt channel: Set the channel as the delivery mode of the belt channel



The control function is in the com.tcn.sdk.springdemo.MainAct class, search for menu\_spr\_set\_slot\_belts

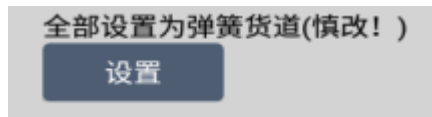
Enter the channel number you want to modify and click Set.

TcnVendIF.getInstance().reqSetBeltsSlot(Integer.valueOf(data1)); //Set the belt channel method

Interface Description

Parameter name	Parameter Description
data1	Enter the channel number

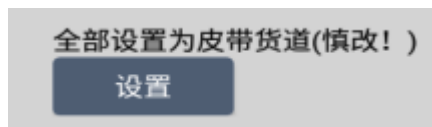
## 17. Set all to spring channels: Set all channels to the spring channel shipping method



The control function is in the `com.tcn.sdk.springdemo.MainAct` class, search for `menu_spr_set_slot_spring_all`  
`TcnVendIF.getInstance().reqSpringAllSlot(-1); //Set up the interface`  
Interface Description

Parameter name	Parameter Description
-1	Fixed value does not need to be modified

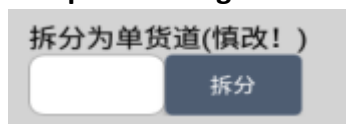
## 18. Set all as belt lanes: Set all lanes as belt lanes for shipment



The control function is in the `com.tcn.sdk.springdemo.MainAct` class, search for `menu_spr_set_slot_belts_all`  
`TcnVendIF.getInstance().reqBeltsAllSlot(-1); //Set the belt channel method`  
Interface Description

Parameter name	Parameter Description
-1	Fixed value does not need to be modified

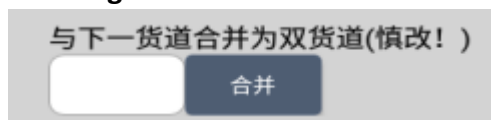
## 19. Split into single lanes: Split the merged lanes into single lanes



The control function is in the `com.tcn.sdk.springdemo.MainAct` class, search for `menu_spr_set_single_slot`  
Enter the channel number you want to split and click Split.  
`TcnVendIF.getInstance().reqSingleSlot(Integer.valueOf(data1)); //Call method`  
Interface Description

Parameter name	Parameter Description
data1	Enter the channel number

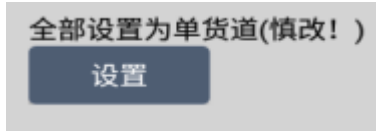
## 20. Merge with the next channel:



The control function is in the `com.tcn.sdk.springdemo.MainAct` class, search for `menu_spr_set_double_slot`  
Enter a channel number, it will merge with the next channel to form a double channel  
`TcnVendIF.getInstance().reqDoubleSlot(Integer.valueOf(data1)); //Merge method`  
Interface Description

Parameter name	Parameter Description
data1	Enter the channel number

## 21. Set all to single channel



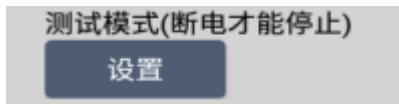
The control function is in the `com.tcn.sdk.springdemo.MainAct` class, search for `menu_spr_set_single_slot_all`

`TcnVendIF.getInstance().reqSingleAllSlot(-1); //Set single channel method`

Interface Description

Parameter name	Parameter Description
-1	Fixed value does not need to be modified

## 22. Test mode: All channels will ship one product.



The control function is in the `com.tcn.sdk.springdemo.MainAct` class, search for `menu_spr_test_mode`

`TcnVendIF.getInstance().reqTestMode(-1); //Simulation test`

Interface Description

Parameter name	Parameter Description
-1	Fixed value does not need to be modified

## SDK return parameter monitoring description

Register `VendListener` for monitoring

Main parameter description:

1、 `TcnVendEventID.CMD_QUERY_SLOT_STATUS`: Query the failure of the cargo lane

1、 Through the query interface

`TcnVendIF.getInstance().reqQuerySlotStatus(Integer.valueOf(strParam));` ( `strParam` : represents the channel number.)

Receive the returned fault in `TcnVendEventID.CMD_QUERY_SLOT_STATUS` in the `com.tcn.sdk.springdemo.MainAct` directory

Return Value	Parameter Description
cEventInfo.m_IParam4	0 represents normal, and other values display corresponding prompts.

## 2、 TcnVendEventID.CMD\_SELF\_CHECK: Query the self-check status

1. Click the self-check query button to call the TcnVendIF.getInstance().reqSelfCheck(-1) interface

;

Receive the return status in TcnVendEventID.CMD\_SELF\_CHECK in the com.tcn.sdk.springdemo.MainAct directory

Return Value	Parameter Description
cEventInfo.m_IParam4	Return error code word prompt

## 3、 TcnVendEventID.CMD\_RESET: Query reset status

1、 Click the Self-check Query button to call

TcnVendIF.getInstance().reqReset(UIComBack.getInstance().getGroupSpringId(strParam)) reset interface

Receive the return status in TcnVendEventID . CMD\_RESET in the com.tcn.sdk.springdemo.MainAct directory

Return Value	Parameter Description
cEventInfo.m_IParam4	Return error code text prompt

## 4、 TcnVendEventID.SET\_SLOTNO\_SPRING: Set the spring channel

1、 Set the spring slot through the TcnVendIF.getInstance().reqSetSpringSlot(Integer.valueOf(data1)); interface

Receive the return result in TcnVendEventID.SET\_SLOTNO\_SPRING in the com.tcn.sdk.springdemo.MainAct directory

Return Value	Parameter Description
cEventInfo.m_IParam4	Return error code text prompt

## 5、 TcnVendEventID.SET\_SLOTNO\_BELTS: Set the crawler cargo lane

1. Set the belt lane through TcnVendIF.getInstance().reqSetBeltsSlot(Integer.valueOf(data1)); interface

Receive the return result in TcnVendEventID.SET\_SLOTNO\_BELTS in the com.tcn.sdk.springdemo.MainAct directory

Return Value	Parameter Description
cEventInfo.m_IParam4	Return error code text prompt

## 6、 TcnVendEventID.SET\_SLOTNO\_ALL\_SPRING: Set all lanes to spring lanes

1、 Set all channels to spring channels through the TcnVendIF.getInstance().reqSpringAllSlot(-1); interface

Receive the returned result in TcnVendEventID. SET\_SLOTNO\_ALL\_SPRING in the com.tcn.sdk.springdemo.MainAct directory

Return Value	Parameter Description
cEventInfo.m_IParam4	Return error code text prompt

## 7、 TcnVendEventID.SET\_SLOTNO\_ALL\_BELT: Set all cargo lanes to track cargo

## lanes

1. Set the track lane through the `TcnVendIF.getInstance().reqBeltsAllSlot(-1);` interface  
Receive the returned result in `TcnVendEventID. SET_SLOTNO_ALL_BELT` in the `com.tcn.sdk.springdemo.MainAct` directory

Return Value	Parameter Description
<code>cEventInfo.m_IParam4</code>	Return error code text prompt

## 8、 TcnVendEventID.SET\_SLOTNO\_SINGLE: Set a single channel

1. Set a single channel through the `TcnVendIF.getInstance().reqSingleSlot(Integer.valueOf(data1))` interface  
Receive the returned result in `TcnVendEventID. SET_SLOTNO_ALL_BELT` in the `com.tcn.sdk.springdemo.MainAct` directory

Return Value	Parameter Description
<code>cEventInfo.m_IParam4</code>	Return error code text prompt

## 9、 TcnVendEventID.SET\_SLOTNO\_DOUBLE: Set double lanes

1. Set up dual channels through the `TcnVendIF.getInstance().reqDoubleSlot(Integer.valueOf(data1))` interface  
Receive the returned result in `TcnVendEventID. SET_SLOTNO_DOUBLE` in the `com.tcn.sdk.springdemo.MainAct` directory

Return Value	Parameter Description
<code>cEventInfo.m_IParam4</code>	Return error code text prompt

## 10、 TcnVendEventID.SET\_SLOTNO\_ALL\_SINGLE: Set all lanes to single lane

1. Set all lanes to single lane through `TcnVendIF.getInstance().reqSingleAllSlot(-1)` interface  
Receive the returned result in `TcnVendEventID. SET_SLOTNO_ALL_SINGLE` in the `com.tcn.sdk.springdemo.MainAct` directory

Return Value	Parameter Description
<code>cEventInfo.m_IParam4</code>	Return error code text prompt

## 11、 TcnVendEventID.CMD\_SET\_COOL : Set cooling

1. Set cooling through `TcnVendIF.getInstance().reqHeatSpring(-1,Integer.parseInt(temp),Integer.parseInt(startTime),Integer.parseInt(endTime));` interface  
Receive the returned result in `TcnVendEventID. SET_SLOTNO_ALL_SINGLE` in the `com.tcn.sdk.springdemo.MainAct` directory

Return Value	Parameter Description
<code>cEventInfo.m_IParam4</code>	Return error code text prompt

## 12、 TcnVendEventID.CMD\_SET\_HEAT: Set heating

1. Set heating through `TcnVendIF.getInstance().reqHeatSpring(-1,Integer.parseInt(temp),Integer.parseInt(startTime),Integer.parseInt(endTime));` interface.  
Receive the return result in `TcnVendEventID. CMD_SET_HEAT` in the

com.tcn.sdk.springdemo.MainAct directory

Return Value	Parameter Description
cEventInfo.m_IParam4	Return error code text prompt

### 13、 TcnVendEventID.CMD\_SET\_GLASS\_HEAT\_OPEN: Glass heating is turned

on

1. Turn on the glass heating through the TcnVendIF.getInstance().reqSetGlassHeatEnable( -1 , true) interface

Receive the return result in TcnVendEventID. CMD\_SET\_GLASS\_HEAT\_OPEN in the com.tcn.sdk.springdemo.MainAct directory

Return Value	Parameter Description
cEventInfo.m_IParam4	Return error code text prompt

### 14、 CMD\_SET\_GLASS\_HEAT\_CLOSE: Glass heating is turned off

1、 Turn off glass heating through the TcnVendIF.getInstance().reqSetGlassHeatEnable( -1 , false ) interface

Receive the return result in TcnVendEventID. CMD\_SET\_GLASS\_HEAT\_OPEN in the com.tcn.sdk.springdemo.MainAct directory

Return Value	Parameter Description
cEventInfo.m_IParam4	Return error code text prompt

### 15、 TcnVendEventID.CMD\_SET\_LIGHT\_OPEN: Turn on the LED light

1、 Through the TcnVendIF.getInstance().reqSetLedOpen( -1 , true) interface, turn on the LED light

Receive the return result in TcnVendEventID. CMD\_SET\_LIGHT\_OPEN in the com.tcn.sdk.springdemo.MainAct directory

Return Value	Parameter Description
cEventInfo.m_IParam4	Return error code text prompt

### 16、 TcnVendEventID.CMD\_SET\_LIGHT\_CLOSE: Turn off the LED light

1. Turn off the LED light through the TcnVendIF.getInstance().reqSetLedOpen( -1 , false ) interface.

Receive the return result in TcnVendEventID. CMD\_SET\_LIGHT\_CLOSE in the com.tcn.sdk.springdemo.MainAct directory

Return Value	Parameter Description
cEventInfo.m_IParam4	Return error code text prompt

### 17、 CMD\_SET\_BUZZER\_OPEN : Turn on the buzzer

1. Turn on the buzzer through the TcnVendIF.getInstance().reqSetBuzzerOpen( -1 , true) interface.

Receive the return result in TcnVendEventID. CMD\_SET\_BUZZER\_OPEN in the com.tcn.sdk.springdemo.MainAct directory

Return Value	Parameter Description
cEventInfo.m_IParam4	Return error code text prompt

### 18、 CMD\_SET\_BUZZER\_CLOSE : Turn off the buzzer

1. Turn off the buzzer through the TcnVendIF.getInstance().reqSetBuzzerOpen( -1 , false ) interface.

Receive the return result in TcnVendEventID. CMD\_SET\_BUZZER\_CLOSE in the com.tcn.sdk.springdemo.MainAct directory

Return Value	Parameter Description
cEventInfo.m_IParam4	Return error code text prompt

### 19、 TcnVendEventID.COMMAND\_SELECT\_GOODS //Goods

selection successful

1 Select the product and call TcnVendIF.getInstance().reqSelectSlotNo(Integer.valueOf(strParam)); product selection interface ( strParam : represents the channel number.)

2. In com.tcn.sdk.springdemo.MainAct, TcnVendEventID.COMMAND\_SELECT\_GOODS receives the successful selection method.

### 20、 TcnVendEventID.COMMAND\_INVALID\_SLOTNO //Invalid

channel

1. Select the product and call

TcnVendIF.getInstance().reqSelectSlotNo(Integer.valueOf(strParam)); product selection interface ( strParam : represents the channel number.)

2. TcnVendEventID.COMMAND\_INVALID\_SLOTNO in com.tcn.sdk.springdemo.MainAct receives this invalid channel method.

### 21、 TcnVendEventID.COMMAND\_SOLD\_OUT: Whether the channel is sold out

1、 Test shipment through the test

channel TcnVendIF.getInstance().reqShipTest(Integer.parseInt(strParam)) interface.

2. If the channel is empty, COMMAND\_SOLE\_OUT in com.tcn.sdk.springdemo.MainAct receives this method.

Return Value	Parameter Description
cEventInfo.m_IParam1	Return value > 0 means it is not sold out

### 22、 TcnVendEventID.COMMAND\_FAULT\_SLOTNO: Cargo lane failure

1、 Test shipment through the test channel TcnVendIF.getInstance().reqShipTest(Integer.parseInt(strParam)) interface.

2. If there is a fault in the cargo channel, COMMAND\_FAULT\_SLOTNO in com.tcn.sdk.springdemo.MainAct receives the return result.

Return Value	Parameter Description
cEventInfo.m_IParam4	Prompt cargo channel failure + description

### 23、 TcnVendEventID.COMMAND\_SHIPPING: Shipping in progress



1. Test shipment through the test channel  
TcnVendIF.getInstance().reqShipTest(Integer.parseInt(strParam)) interface.
2. In com.tcn.sdk.springdemo.MainAct, COMMAND\_SHIPPING receives the result returned by "shipping in progress".

Return Value	Parameter Description
cEventInfo.m_IParam1	Cargo lane number
cEventInfo.m_IParam4	"Shipping" text prompt

#### 24、 TcnVendEventID.COMMAND\_SHIPMENT\_SUCCESS: Shipment successful

1. Test shipment through the test channel  
TcnVendIF.getInstance().reqShipTest(Integer.parseInt(strParam)) interface.  
Receive the result of shipping in COMMAND\_SHIPMENT\_SUCCESS in  
com.tcn.sdk.springdemo.MainAct .

#### 25、 TcnVendEventID.COMMAND\_SHIPMENT\_FAILURE : Shipping failed

1. Test shipment through the test channel  
TcnVendIF.getInstance().reqShipTest(Integer.parseInt(strParam)) interface.  
Receive the result of "shipping in progress" in COMMAND\_SHIPMENT\_FAILURE in  
com.tcn.sdk.springdemo.MainAct .

#### 26、 TcnVendEventID.CMD\_READ\_DOOR\_STATUS: Door action

- 1、 Get the status through door open and door closed.  
Receive the main door status return result in CMD\_READ\_DOOR\_STATUS in  
com.tcn.sdk.springdemo.MainAct .

Return Value	Parameter Description
cEventInfo.m_IParam1	1: Door open
cEventInfo.m_IParam1	2: Door closed

#### 27、 TcnVendEventID.MDB\_RECIVE\_PAPER\_MONEY: Throwing paper money

- 1、 MDB\_RECIVE\_PAPER\_MONEY in com.tcn.sdk.springdemo.MainAct receives the banknote balance.

Return Value	Parameter Description
cEventInfo.m_IParam4	Amount invested

#### 28、 TcnVendEventID.MDB\_RECIVE\_COIN\_MONEY: coin toss

- Receive the coin balance in MDB\_RECIVE\_COIN\_MONEY in com.tcn.sdk.springdemo.MainAct .

Return Value	Parameter Description
cEventInfo.m_IParam4	Amount invested

#### 29、 TcnVendEventID.MDB\_BALANCE\_CHANGE //Balance change

- 1、 There are many ways to change the balance, such as throwing paper money and coins. Each time you throw it, it will be received in the MDB\_BALANCE\_CHANGE method in com.tcn.sdk.springdemo.MainAct. Get the balance through TcnVendIF.getInstance().getBalance()

#### 30、 TcnVendEventID.MDB\_PAYOUT\_PAPERMONEY: Find banknotes

1. After the purchase is completed, click on the refund button in the TcnVendIF.getInstance().reqSelectCancel()

method.

2. Receive the return result in the MDB\_PAYOUT\_PAPERMONEY method in com.tcn.sdk.springdemo.MainAct.

Return Value	Parameter Description
cEventInfo.m_IParam1	-1 Refund in progress
cEventInfo.m_IParam1	0
cEventInfo.m_IParam4	Coin refund status + amount display

### 31、 TcnVendEventID.MDB\_PAYOUT\_COINMONEY: Find coins

1. After the purchase is completed, click on the refund button in the **TcnVendIF.getInstance().reqSelectCancel()** method

2. Receive the return result in the MDB\_PAYOUT\_COINMONEY method in com.tcn.sdk.springdemo.MainAct.

Return Value	Parameter Description
cEventInfo.m_IParam1	-1 Refund in progress
cEventInfo.m_IParam1	0
cEventInfo.m_IParam4	Coin refund status + amount display

### 32、 TcnVendEventID.MDB\_SHORT\_CHANGE\_PAPER: Insufficient banknotes

1. Click on the coin refund button in the **TcnVendIF.getInstance().reqSelectCancel()** method. If the machine detects that the banknotes are insufficient,

Receive the return result in the MDB\_SHORT\_CHANGE\_PAPER method in com.tcn.sdk.springdemo.MainAct

Return Value	Parameter Description
cEventInfo.m_IParam4	Insufficient banknotes

### 33、 TcnVendEventID.MDB\_SHORT\_CHANGE\_COIN: Insufficient coins

1. Click on the coin refund button in the **TcnVendIF.getInstance().reqSelectCancel()** method if it is detected that the machine has insufficient coins .

2. Receive the return result in the MDB\_SHORT\_CHANGE\_COIN method in com.tcn.sdk.springdemo.MainAct.

Return Value	Parameter Description
cEventInfo.m_IParam4	Insufficient coins

### 34、 TcnVendEventID.MDB\_SHORT\_CHANGE: Insufficient change

1. Click to return the money in **TcnVendIF.getInstance().reqSelectCancel()** method, if it is detected that the machine does not have enough money

2. Receive the return result in the MDB\_SHORT\_CHANGE method in com.tcn.sdk.springdemo.MainAct.

Return Value	Parameter Description
cEventInfo.m_IParam4	Insufficient change

### 35、 TcnVendEventID.CMD\_CARD\_PAY\_FINISH: Card payment

1. Report the consumption amount when swiping the card  
TcnVendIF.getInstance().uploadMoneyToCard(soltNo,price,tradeNo);

2. Receive the return result in the CMD\_CARD\_PAY\_FINISH method in com.tcn.sdk.springdemo.MainAct.

Return Value	Parameter Description
cEventInfo.m_IParam1	0 means the card deduction is successful otherwise the card deduction fails
cEventInfo.m_IParam2	Cargo lane number

### 36、 TcnVendEventID.CMD\_CASH\_PAYOUT\_NORSP\_CONTIN: Continue to purchase or return the currency

1. After the consumption is completed, the result is returned in  
CMD\_CASH\_PAYOUT\_NORSP\_CONTIN of com.tcn.springboard.vend.VendControl .

Return Value	Parameter Description
--------------	-----------------------

cEventInfo.m_IParam4	Continue to purchase prompt
----------------------	-----------------------------

### 37、 TcnVendEventID.CMD\_STOPING\_CARD\_PAY: **Stop card swiping**

1. Receive the return result in the CMD\_STOPING\_CARD\_PAY method in com.tcn.sdk.springdemo.MainAct.

Return Value	Parameter Description
cEventInfo.m_IParam4	Stop card swipe prompt