

iScan Pro SDK instruction

Current version: V1.2

Version status: Released

Document Update Instructions:

Version	Update	Version	content
v1.0	2020/07/23	V1.0	release
v1.1	2020/11/12	V1.1	1、update api
V1.2	2020/12/29	V1.2	1、update api

目录

iScan Pro SDK instruction	1
1、 Statement.....	2
2、 Development Instruction	2
3、 Interface list	4
3.1、 Open scan engine	4
3.2、 Power off scan engine	4
3.3、 Start Scan.....	5
3.4、 Stop Scan	5
3.5、 Disable Scan key	5
3.6、 Set the scan results output mode	5
3.7、 Set Scan prompt mode.....	6
3.8、 Set additional key	7
3.9、 Set barcode data processing rules.....	7
3.10、 Set the trigger mode	8
3.11、 Set the scan timeout	8
3.12、 Set the interval time for continuous scanning	8
3.13、 Set character encoding format.....	9
3.14、 Delete the existing contents of the edit box	9
3.15、 Restore default configuration parameters	9
3.16、 Set the decoding region mode.....	10
3.17、 Set the laser complement mode	10

3.18、	Set barcode type.....	10
3.19、	Set the image output mode	11
3.20、	Register the monitor object for scanning results.....	11
3.21、	Cancel the monitoring object of scanning results.....	12

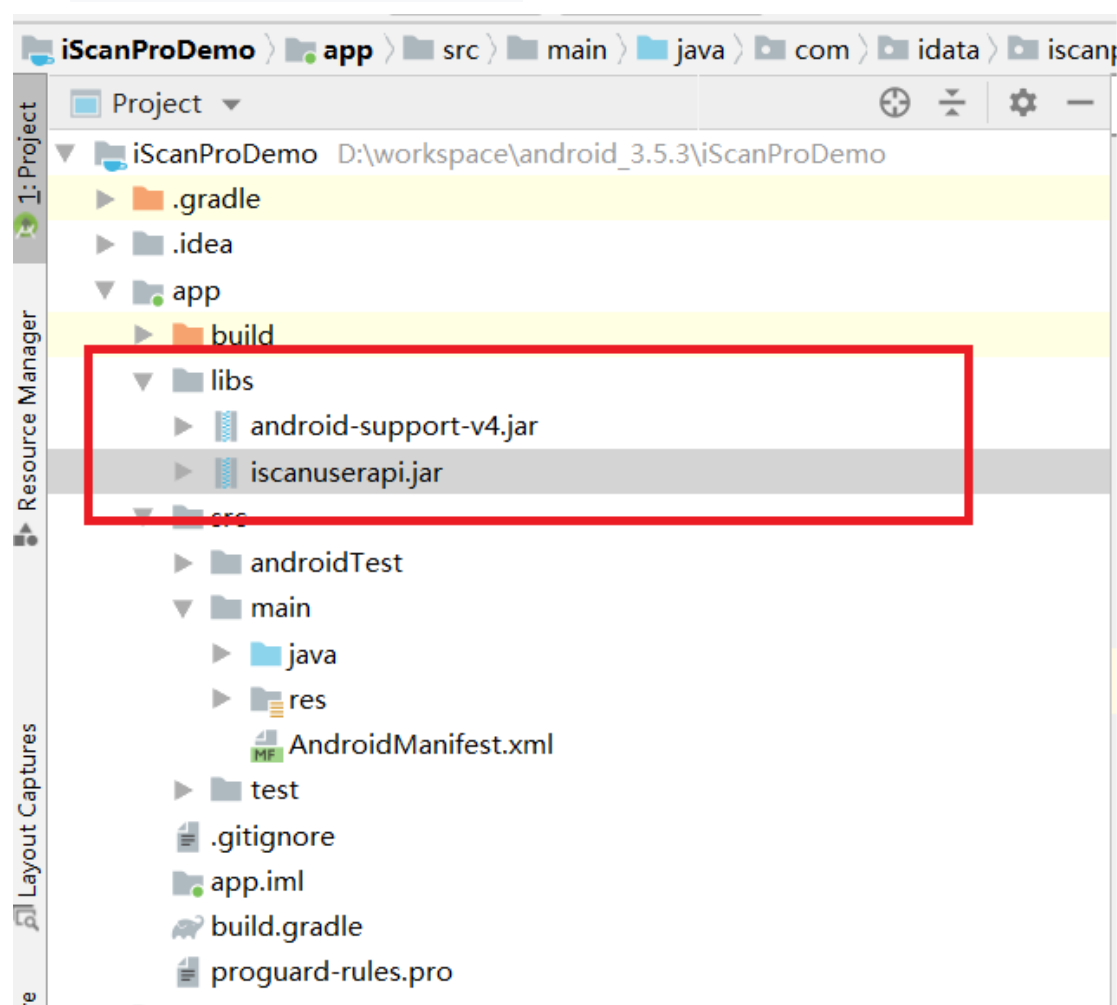
1、 Statement

This document mainly explains the code scanning development instructions of IDATA terminal equipment, for the reference of related industries and related developers of the integrator.

本文档主要说明 iData 终端设备扫码开发说明,供相关行业以及集成方相关开发人员参考。

2、 Development Instruction

- 1) Please copy "iscanuserapi.jar" from the libs directory to the libs directory of the project, as shown in the figure below:



- 2) Add "iscanUserApi.jar" to the project compilation path. After adding "build.gradle" file, it

is as follows:

```
dependencies {
    implementation fileTree(include: ['*.jar'], dir: 'libs')
    implementation 'androidx.appcompat:appcompat:1.0.2'
    implementation 'androidx.constraintlayout:constraintlayout:1.1.3'
    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'androidx.test.ext:junit:1.1.0'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.1.1'
    implementation files('libs/iscanuserapi.jar')
}
```

- 2) Initialize MisCanInterface in the onCreate method in the Context.
- 3) After initialization, MisCanInterface method is called for scanning operation.

```
//定义接口操作对象
private iScanInterface miScanInterface ;

//定义数据回调接口
private IScanListener scanResltListener = new IScanListener() {
    /**
     * @param data 条码数据
     * @param type 条码类型
     * @param decodeTime 解码耗时
     * @param keyDownTime 按键时间
     * @param imagePath 图片路径，注意：默认图片输出为关闭状态；
     * 如果需要条码图片需要打开图片输出选项。
     */
    @Override
    public void onScanResults(String data, int type, long decodeTime, long keyDownTime, String imagePath) {
        Log.d(TAG, msg: "onScanResults: data="+data);
        Log.d(TAG, msg: "onScanResults: type="+type);
        Log.d(TAG, msg: "onScanResults: decodeTime="+decodeTime);
        Log.d(TAG, msg: "onScanResults: keyDownTime="+keyDownTime);
        Log.d(TAG, msg: "onScanResults: imagePath="+imagePath);
    }
};

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    //初始化接口对象
    miScanInterface = new iScanInterface(context: this);
    //注册数据回调接口
    miScanInterface.registerScan(scanResltListener);
}
```

4) App exit destruction

```
@Override  
protected void onDestroy() {  
    super.onDestroy();  
    //应用关闭后注销回调  
    miScanInterface.unregisterScan(scanResltListener);  
}
```

3、 Interface list

3.1、 Open scan engine

void open()

Description: power on for scan engine. it is the must to set power on and then invoke the scan

Note: this interface is not called frequently. To save power, it is usually called once when the program is opened, or it is not called to iScan automatic control

Input parameter: null

Return value: null

3.2、 Power off scan engine

void close()

Description: power off for scan engine. In contrast with the open()

Note: this interface is not called frequently. To save power, it is usually called once when the program is opened, or it is not called to iScan automatic control

Input parameter: null

Return value: null

3.3、 Start Scan

```
void scan_start()
```

Description: start scan operation, with this operation the scan engine will off

Input parameter: null

Return value: null

3.4、 Stop Scan

```
void scan_stop()
```

Description: stop scan operation, with this operation the scan engine will off

Input parameter: null

Return value: null

3.5、 Disable Scan key

```
void lockScanKey(boolean enable)
```

Description: lock the scan key, to lock the scan key with iScan as scan function.

Noted: Please unlock(false) the Scan key when you exit the application. otherwise it can't scan after you exit the application

Input parameter: enable, Scan key disabled state.

true: Disable the scan key, after which users can develop the scan key by themselves.。

false: remove the disable, Can use the scan key to scan

Value: null

3.6、 Set the scan results output mode

```
void setOutputMode(int mode)
```

Description: scan result output mode

。

Input parameter:

mode 0: scan result send to the place appointed

mode 1: scan result send as broadcast. When you choose the broadcast mode, you can use Android broadcast mode to get the scan result. The default broadcast definition for scan results is as follows:

action: android.intent.action.SCANRESULT

extras: value, It's the scan result. When scan fail, the value will be null.

length, The length of the barcode

mode 2: scan result by key event

Return value: null

3.7、 Set Scan prompt mode

void enablePlayBeep(boolean enable)

Description: on/off beep after scan success

Input parameter:

enable true : play beep after scan success

enable false : not play beep after scan success

Value: null

void enableFailurePlayBeep(boolean enable)

Description: on/off beep after scan fail

Input parameter:

enable true : play beep after scan fail

enable false : not play beep after scan fail

Return value: null

void enablePlayVibrate(boolean enable)

Description: on/off vibration after scan success

Input parameter:

enable true : on vibration after scan success

enable false : off vibration after scan success

Return value: null

```
void lightSet (boolean enable )
```

Description: right LED control

Input parameter:

Enable : true show false not show

Return value: null

3.8、 Set additional key

```
void enableAddKeyValue(int value)
```

Description: addition mode, add additional ENTER, TAB, NONE or NEWLINE key etc.

Input parameter:

value 0 add NONE

value 1 add ENTER

value 2 add TAB key

value 3 add NEWLINE key

Return value: null

3.9、 Set barcode data processing rules

```
void addPrefix(String text)
```

Description: add additional prefix before scan result

Input parameter:

text: the test you hope to add as prefix before scan result

Return value: null

```
void addSuffix(String text)
```

Description: add additional suffix after scan result

Input parameter:

text: the test you hope to add as suffix after scan result

Return value: null

```
void filterCharacter (String text )
```

Description: filt appointed character

Input parameter:

text: appointed character

Return value: null

3.10、 Set the trigger mode

void continuousScan(boolean enable)

Description: set continuous scan

Input parameter:

enable, Continuous scanning state

true: open Continuous scanning state . **Explain:** You can continuous scan after you

clikc the scan key

false: Close the continuous scan.

value: null

void effortScan (boolean enable)

Description: saving mode, press the key until the barcode is decode or the scan overtime will stop scanning

Input parameter: enable, saving mode

true: open saving mode

false: close saving mode

value: null

3.11、 Set the scan timeout

void setTimeOut(int value)

Description: Set the timeout time, stop scanning after the timeout time

Input parameter: value, timeout time, Time: ms, Default: 30000

value: null

3.12、 Set the interval time for continuous scanning

void setIntervalTime(int value)

Description: Set the interval time for continuous scanning. In continuous scanning mode, the interval time after each scan is successful

Input parameter: value, the interval time for continuous scanning, Time: ms,

Default: 5000

Value: null

3.13、 Set character encoding format

void setEncodeFormart(int mode)

Description: Set the decoded character encoding format

Input parameter: mode, Encoding type value, the optional encoding is as follows:

- 0:** Auto
- 1:** GB2312
- 2:** GBK
- 3:** GB18030
- 4:** UTF-8
- 5:** ISO-8859-1
- 6:** BIG5
- 7:** SJIS
- 8:** EUC-JP

Value: null

3.14、 Delete the existing contents of the edit box

void setDelete(boolean enable)

Description: delete the contents of the edit box after scanning the barcode

Input parameter: enable, Automatically clears existing content state

true: After scanning the barcode, delete the contents of the edit box

false: Do not delete the contents of the edit box after scanning the bar code. Default.

Value: null

3.15、 Restore default configuration parameters

void resetScan()

Description: Restore the default Settings of scan parameters. After restoration, scan configuration parameters will be restored to factory state.

Input parameter: null

Value: null

3.16、 Set the decoding region mode

```
void setCenterMode(int mode)
```

Description: Set the decode region mode。

Input parameter: mode, decode region mode:

0: Regional mode, default。

1: center mode, 开 When turned on, the system only recognizes the bar code in the center area of the aiming light。

Value: null

3.17、 Set the laser complement mode

```
void setAimLightMode(int mode)
```

Description: Set up the mode of laser t

Input parameter: mode, 。

0: Fill light and laser light are bright

1: Only laser light

2: Only fill light

3: Both don't light

Value: null

3.18、 Set barcode type

```
void setBarcodeEnable(int barcodeId,boolean enable)
```

Description: Set barcode type

Input parameter: barcodeId, Barcode type, Support the following values:

0: Aztec

1: Codabar

2: Code11

3: Code128

4: Code39

6: Code93

8: DataMatrix

9: EAN8

10: EAN13

11: Interleaved 2 of 5

12: Maxicode

13: Micropdf

15: Pdf417

17: QR

19: UPCA

20: UPCE0

48: HANXIN

enable, State of the barcode, Support the following values:

true: Support specified barcode decoding

false: Close specified barcode decoding

Value:null

3.19、 Set the image output mode

```
void saveImageMode(int mode)
```

Description: Set the image output mode

Note: 1) This api for DS7000/DS7000pro Scan engine(iData)

2) Read permissions callback images need a device, the android version 10.0 in AndroidManifest. Add android: XML document Application requestLegacyExternalStorage = "true"

3) The APP must run on the main thread

Input parameter: mode, image output mode, Support the following values:

0: Close, default

1: output all the time

2: Decoded successfully output

3: Decoding failed output

Value:null

3.20、 Register the monitor object for scanning results

```
void registerScan(IScanListener mIScanListener)
```

Description: Register the listener, implement the listener callback method to get the saved image path

Note: 1) This api for DS7000/DS7000pro Scan engine(iData), If you want to get the image path, you need to call the saveImageMode method to start the image output.

2) The interface callback will return the barcode and the image. In addition, the barcode return of this interface is not affected by the setting parameters of the output mode.

3) After the picture output is turned on, please delete the picture after the use is finished to save storage space.

Input parameter: mIScanListener: Listener object, Sample list:

The callback is implemented as follows:

@Override

```
public void onScanResults(String data, int type, long decodeTime, long
keyDownTime, String imagePath) {
    // TODO Auto-generated method stub
    /**
    data:barcode result
    type:barcode type
    decodeTime:
    keyDownTime: key press time
    imagePath:
    */
}
```

Value:null

3.21、 Cancel the monitoring object of scanning results

void unregisterScan(IScanListener mIScanListener)

Description: Unlog the scan result listener, called when destruction is applied。

Input parameter: mIScanListener: Listener object, Listener object at initialization time:

Value:null