



Android Bluetooth SDK

ID SCAN H3

iDTRONIC GmbH
Ludwig-Reichling-Straße 4
67059 Ludwigshafen
Germany/Deutschland

Issue 1.0
– 24 November 2025 –

Phone +49 621 6690094-0
Fax +49 621 6690094-9
E-Mail info@idtronic-rfid.com
Web <https://idtronic-rfid.com/>

Subject to alteration without prior notice.
© Copyright iDTRONIC GmbH 2025
Printed in Germany

Contents

1	Description.....	5
1.1	Prerequisites	5
2	Interface Description – Device Operation Library	6
2.1	ScannerService	6
2.1.1	Monitor System Bluetooth Status	6
2.2	ScannerBleService	6
2.2.1	Search Device (BLE)	6
2.2.2	Stop Search (BLE)	6
2.3	ScannerSppService	6
2.3.1	Search Device (SPP)	6
2.3.2	Stop Search (BLE)	7
2.4	ScannerDevice	7
2.4.1	Construct a ScannerDevice Object (BLE)	7
2.4.2	Construct a ScannerDevice Object (SPP)	7
2.4.3	Get Searched Bluetooth Name	7
2.4.4	Get Bluetooth MAC Address.....	7
2.4.5	Connect Devices	8
2.4.6	Disconnect Device.....	8
2.4.7	Monitor Device Data.....	8
2.4.8	Send Commands to Device	8
2.4.9	Monitor Device Status	8
2.4.10	Get Device Version Information	9
2.4.11	Get Device Power	9
2.4.12	Get Device Volume	9
2.4.13	Set Device Volume	9
2.4.14	Get Transmission Speed	10
2.4.15	Set Transfer Speed	10
2.4.16	Get Vibration Switch	10
2.4.17	Set Vibration Switch.....	10
2.4.18	Get Uppercase and Lowercase Letter Conversion.....	10
2.4.19	Set Uppercase and Lowercase Letter Conversion	11
2.4.20	Get End Symbol	11
2.4.21	Set End Symbol	11
2.4.22	Get Sleep Time.....	11
2.4.23	Set Sleep Time	12
2.4.24	Get Storage Mode Switch	12

2.4.25	Set Storage Mode Switch	12
2.4.26	Get Number of Storage Items	12
2.4.27	Get Storage Data	13
2.4.28	Clear Storage Data	13
2.4.29	Time Calibration	13
2.4.30	Shutdown	13
2.4.31	Restore Default Settings	13
3	Interface Description – Device Automatic Connection Library.....	14
3.1	BLE Device Automatic Connection.....	14
3.2	SPP Device Automatic Connection	14
3.3	HID Device Auto-Connect	15
3.4	HOG Device Auto-Connect	15
3.5	Stop Auto-Connect	16

1 Description

AndroidScannerDemo contains a sample project and an .apk file to install the project.

Library files can be found in libs and consists of

- lw_scanner_main-release-Vx.x.x.aar (Device operation library)
- lw_scanner_auto-release-Vx.x.x.aar (Device automatic connection library)

1.1 Prerequisites

The library files need to be placed in the project's libs folder.

The dependencies can be added for example in the gradle.build file

```
dependencies {  
    implementation files('libs/lw_scanner_main-release-vx.x.x.aar')  
    implementation files('libs/lw_scanner_auto-release-vx.x.x.aar')  
}
```

2 Interface Description – Device Operation Library

2.1 ScannerService

2.1.1 Monitor System Bluetooth Status

```
void onBluetoothState(BluetoothStateCallback callback)
```

This method responds to the status of the BluetoothAdapter:

onOpened	BluetoothAdapter.STATE_ON
onClosed	BluetoothAdapter.STATE_OFF
onOpening	BluetoothAdapter.STATE_TURNING_ON
onClosing	BluetoothAdapter.STATE_TURNING_OFF

Example:

```
ScannerService.onBluetoothState(new ScannerService.BluetoothStateCallback() {  
    @Override public void onOpened(){ }  
    @Override public void onClosed(){ }  
    @Override public void onOpening(){ }  
    @Override public void onClosing(){ }  
});
```

2.2 ScannerBleService

2.2.1 Search Device (BLE)

```
void search(ScannerResultListener listener)
```

Returns a scanning device of type ScannerDevice.

Example:

```
ScannerBleService.search((ScannerDevice scannerDevice) -> {  
    // TODO  
});
```

2.2.2 Stop Search (BLE)

```
void stopSearch()
```

Example:

```
ScannerBleService.stopSearch();
```

2.3 ScannerSppService

2.3.1 Search Device (SPP)

```
void search(ScannerResultListener listener)
```

Returns a scanning device of type ScannerDevice.

Example:

```
ScannerSppService.search((ScannerDevice scannerDevice) -> {
    // TODO
});
```

2.3.2 Stop Search (BLE)

```
void stopSearch()
```

Example:

```
ScannerSppService.stopSearch();
```

2.4 ScannerDevice

Note: The method 2.4.6 Disconnect Device and following methods must be called after the device is connected.

2.4.1 Construct a ScannerDevice Object (BLE)

```
ScannerDevice from (ScanResult scanResult)
```

Example:

```
scannerDevice scanner = scannerBleDevice.from(scanResult);
```

or

```
ScannerDevice from (BluetoothDevice bluetoothDevice)
```

Example:

```
scannerDevice scanner = scannerBleDevice.from(bluetoothDevice);
```

2.4.2 Construct a ScannerDevice Object (SPP)

```
ScannerDevice from (BluetoothDevice bluetoothDevice)
```

Example:

```
scannerDevice scanner = scannerBleDevice.from(bluetoothDevice);
```

2.4.3 Get Searched Bluetooth Name

```
String getName()
```

Example:

```
String name = scanner.getName();
```

2.4.4 Get Bluetooth MAC Address

```
String getAddress()
```

Example:

```
String address = scanner.getAddress();
```

2.4.5 Connect Devices

```
void connect(ScannerDevice.ConnectCallback callback)
```

Returns a boolean success to indicate successful connection and a message prompt.

Example:

```
scanner.connect((boolean success, String message) -> {  
    if (success) {  
        // TODO  
    }  
});
```

2.4.6 Disconnect Device

```
void close()
```

Example:

```
scanner.close();
```

2.4.7 Monitor Device Data

```
void onData(ScannerDevice.DataCallback callback)
```

Returns the scanned data in bytes (ASCII encoded) and the string converted from the bytes.

Example:

```
scanner.onData((byte[] bytes, String str) -> {  
    // TODO  
});
```

2.4.8 Send Commands to Device

```
void send(byte[] command)
```

Example: the command below shuts down the device. If a command is sent that requires the scanner to return data, the `onDataListener` needs to be called.

```
byte[] command = new byte[] {(byte) 0xBA, 0x05, 0x38};  
scanner.send(command);
```

2.4.9 Monitor Device Status

```
void onState(ScannerDevice.StateCallback callback)
```

This method responds to the status of `BluetoothDevice`:

<code>onConnected</code>	<code>BluetoothGatt.STATE_CONNECTED</code>
<code>onDisconnected</code>	<code>BluetoothGatt.STATE_DISCONNECTED</code>
<code>onConnecting</code>	<code>BluetoothGatt.STATE_CONNECTING</code>
<code>onDisconnecting</code>	<code>BluetoothGatt.STATE_DISCONNECTING</code>

Note: SPP only supports `onConnected` and `onDisconnected`.

Example:

```
scanner.onState(new ScannerDevice.StateCallback() {
    @Override public void onConnected() { }
    @Override public void onDisconnected() { }
    @Override public void onConnecting() { }
    @Override public void onDisConnecting() { }
});
```

2.4.10 Get Device Version Information

```
void getVersion(ScannerDevice.ResponseCallback<String> callback)
```

Returns a data string with the version information and an error message.

Example:

```
scanner.getVersion((String data, ScannerDevice.ResponseError error) -> {
    // TODO
});
```

2.4.11 Get Device Power

```
void getBatteryLevel(scannerDevice.ResponseCallback<Integer> callback)
```

Returns an integer of the battery level (value in %) and an error message.

Example:

```
scanner.getBatteryLevel((Integer data, ScannerDevice.ResponseError error) -> {
    // TODO
});
```

2.4.12 Get Device Volume

```
void getVolume(ScannerDevice.ResponseCallback<Volume> callback)
```

Returns the volume data and an error message.

Volume	
MUTE	Mute
LOW	low volume
MEDIUM	medium volume
HIGH	high volume

Example:

```
scanner.getVolume((Volume data, ScannerDevice.ResponseError error) -> {
    // TODO
});
```

2.4.13 Set Device Volume

```
void setVolume(Volume volume)
```

Example:

```
scanner.setVolume(Volume.MUTE);
```

2.4.14 Get Transmission Speed

```
void getSpeed(ScannerDevice.ResponseCallback<Speed> callback)
```

Returns the speed data and an error message.

Speed	
FAST	Fast transfer
MEDIUM	Medium-speed transmission
LOW	Low-speed transmission
ULTRA_LOW	Ultra-low speed transmission

Example:

```
scanner.getSpeed((Speed data, ScannerDevice.ResponseError error) -> {  
    // TODO  
});
```

2.4.15 Set Transfer Speed

```
void setSpeed(Speed speed)
```

Example:

```
scanner.setSpeed(Speed.FAST);
```

2.4.16 Get Vibration Switch

```
void getVibration(ScannerDevice.ResponseCallback<Boolean> callback)
```

Returns a boolean data (true indicates vibration is turned-on) and an error message.

Example:

```
scanner.getVibration((Boolean data, ScannerDevice.ResponseError error) -> {  
    // TODO  
});
```

2.4.17 Set Vibration Switch

```
void setVibration(boolean on)
```

Set boolean parameter “on” to true to enable vibration and false to disable vibration.

Example:

```
scanner.setVibration(true);
```

2.4.18 Get Uppercase and Lowercase Letter Conversion

```
void getCase(ScannerDevice.ResponseCallback<Case> callback)
```

Returns the case data and an error message.

Case

LOWER	Force capitalization
UPPER	Force lowercase
NONE	Do not convert
SWAP	Swap uppercase and lowercase

Example:

```
scanner.getCase((Case data, ScannerDevice.ResponseError error) -> {
    // TODO
});
```

2.4.19 Set Uppercase and Lowercase Letter Conversion

```
void setCase(Case case)
```

Example:

```
scanner.setCase(Case.LOWER);
```

2.4.20 Get End Symbol

```
void getEndSymbol(ScannerDevice.ResponseCallback<EndSymbol> callback)
```

Returns the end symbol data and an error message.

EndSymbol	
RETURN	Enter
LINE	Line break
RETURN_LINE	Carriage return and line feed
NONE	No addition
TAB	Tab

Example:

```
scanner.getEndSymbol((EndSymbol data, ScannerDevice.ResponseError error) -> {
    // TODO
});
```

2.4.21 Set End Symbol

```
void setEndSymbol(EndSymbol endSymbol)
```

Example:

```
scanner.setEndSymbol(EndSymbol.RETURN);
```

2.4.22 Get Sleep Time

```
void getSleepTime(ScannerDevice.ResponseCallback<SleepTime> callback)
```

Returns the sleep time data and an error message.

SleepTime	
SEC_30	30 seconds later
MIN_1	1 minute later
MIN_2	2 minutes later

MIN_5	5 minutes later
MIN_10	10 minutes later
MIN_30	30 minutes later
HOURL_1	1 hour later
HOURL_2	2 hours later
HOURL_3	3 hours later
HOURL_4	4 hours later
HOURL_5	5 hours later
HOURL_6	6 hours later
HOURL_12	12 hours later
NEVER	Never sleep

Example:

```
scanner.getSleepTime((SleepTime data, ScannerDevice.ResponseError error) -> {  
    // TODO  
});
```

2.4.23 Set Sleep Time

```
void setSleepTime(SleepTime sleepTime)
```

Example:

```
scanner.setSleepTime(SleepTime.MIN_5);
```

2.4.24 Get Storage Mode Switch

```
void getStoreMode(ScannerDevice.ResponseCallback<Boolean> callback)
```

Returns a boolean (true indicates storage mode is enabled) and an error message.

Example:

```
scanner.getStoreMode((Boolean data, ScannerDevice.ResponseError error) -> {  
    // TODO  
});
```

2.4.25 Set Storage Mode Switch

```
void setStoreMode(boolean storeMode)
```

Set the boolean parameter “storeMode” to true to enable storage mode or false to disable storage mode.

Example:

```
scanner.setStoreMode(true);
```

2.4.26 Get Number of Storage Items

```
void getStoreTotal(ScannerDevice.ResponseCallback<Integer> callback)
```

Returns an integer with the number of stored items and an error message.

Example:

```
scanner.getStoreTotal((Integer data, ScannerDevice.ResponseError error) -> {  
    // TODO  
});
```

2.4.27 Get Storage Data

```
void getStoreData(ScannerDevice.ResponseCallback<String> callback)
```

Returns a string of data and an error message.

Example:

```
scanner.getStoreData((String data, ScannerDevice.ResponseError error) -> {  
    // TODO  
});
```

2.4.28 Clear Storage Data

```
void clearStoreData()
```

Example:

```
scanner.clearStoreData();
```

2.4.29 Time Calibration

```
void timing()
```

Example:

```
scanner.timing();
```

2.4.30 Shutdown

```
void shutdown()
```

Example:

```
scanner.shutdown();
```

2.4.31 Restore Default Settings

```
void restoreSetting()
```

Example:

```
scanner.restoreSetting();
```

3 Interface Description – Device Automatic Connection Library

3.1 BLE Device Automatic Connection

String ble(FinishCallback callback, StepCallback stepCallback)

Returns:

Name	Description
Direct return value	Returns a string of characters that needs to be scanned by a barcode scanner in the form of a 1D or 2D code
success	Whether the device was found successfully
device	Bluetooth device objects can be connected using the ScannerBleDevice object obtained from the device operation library: ScannerBleDevice.from(device).connect(callback)
step	Processing stage

Example:

```
String code = ScannerAutoService.ble((success, device) -> {
    if(success){
        ScannerBleDevice.from(device).connect(((success1,message)->{
            // TODO
        }));
    }
},step -> {
    // TODO
});
Display 2D(code);
```

3.2 SPP Device Automatic Connection

String spp(FinishCallback callback, StepCallback stepCallback)

Returns:

Name	Description
Direct return value	Returns a string of characters that needs to be scanned by a barcode scanner in the form of a 1D or 2D code
success	Whether the device was found successfully
device	Bluetooth device objects can be connected using the ScannerSppDevice object obtained from the device operation library: ScannerSppDevice.from(device).connect(callback)
step	Processing stage

Example:

```
String code = ScannerAutoService.spp((success, device) -> {
    if(success){
        ScannerSppDevice.from(device).connect(((success1,message)->{
            // TODO
        }));
    }
},step -> {
```

```
// TODO
});
Display 2D(code);
```

3.3 HID Device Auto-Connect

```
String hid(FinishCallback callback, StepCallback stepCallback)
```

Returns:

Name	Description
Direct return value	Returns a string of characters that needs to be scanned by a barcode scanner in the form of a 1D or 2D code
success	Whether the device was connected successfully
device	The bluetooth device object that has been connected
step	Processing stage

Example:

```
String code = ScannerAutoService.hid((success, device) -> {
    if(success){
        // TODO
    }
},step -> {
    // TODO
});
Display 2D(code);
```

3.4 HOG Device Auto-Connect

```
String hog(FinishCallback callback, StepCallback stepCallback)
```

Returns:

Name	Description
Direct return value	Returns a string of characters that needs to be scanned by a barcode scanner in the form of a 1D or 2D code
success	Whether the device was connected successfully
device	The bluetooth device object that has been connected
step	Processing stage

Example:

```
String code = ScannerAutoService.hog((success, device) -> {
    if(success){
        // TODO
    }
},step -> {
    // TODO
});
Display 2D(code);
```

3.5 Stop Auto-Connect

```
void stop()
```

Example:

```
ScannerAutoService.stop();
```