

IDO-IPC8802-V1 Android软件使用手册

1 主板信息

2 外围功能接口

2.1 固件烧录

2.2 打开测试工具方法

2.3 指示灯

2.4 REC

2.5 耳机

2.6 耳麦录音

2.7 TF卡槽

2.8 HDMI

2.9 HDMI_IN

2.10 USB

2.11 TypeC 接口

2.11.1 Device从机模式

2.11.2 Host主机模式

2.11.3 DP模式

2.12 串口及IO接口

2.12.1 硬件接口介绍

2.12.2 串口测试方法

2.12.3 CAN口发送

2.12.4 CAN口接收

2.13 DI/DO口使用

2.13.1 DI 端口测试方法

2.13.2 DO 端口测试方法

2.13.3 APK读写DIDO方法

三、网络测试

3.1 Ethernet

3.1.1 LAN0

3.1.2 LAN1

3.1.3 以太网共享

3.1.4 以太网-WLAN共享

3.2 4G网络

3.2.1 4G网络

3.2.2 4G网络共享

3.3 WiFi

3.4 Bluetooth

3.5 天线接口



IDO-IPC8802-V1

Android软件使用手册

深圳触觉智能科技有限公司

www.industio.cn

文档修订历史

版本	PCBA版本号	修订内容	修订	审核	日期
V1.0	V1A	创建文档并填写文档	TBR	IDO	2024/07/18
V1.1	V1A	更改文档格式	TBR	IDO	2024/09/25

1 主板信息

基本参数		
PCB版本号	IDO-SBC3588-V1	
SOC	RockChip RK3588	
内存	LPDDR4, 8GB	
存储	eMMC, 64GB/128GB	
OS	Android12	
4G/5G	4G: EC20 5G: RG200U (NC)	
WiFi/BT	AP6275S	
Lan	数量: 2	Lan0 (1000M) : YT8531C Lan1 (1000M) : YT8531C
Display	数量: 2	HDMI2.1 x1: 支持最大分辨率: 8K@60Hz DP1.4 x1: 支持最大分辨率: 8K@30Hz
Input	数量: 1	HDMI-IN 视频输入
USB	数量: 5	1路 TYPE-C (OTG、Host、DP) 4路 USB3.0 标准USB-A
SD卡	数量: 1	SD卡座
耳机座	数量: 1	美标
按键	数量: 1	REC
串口	数量: 4	RS232 x 2 RS485 x 2
CAN	数量: 1	CAN 0
LED	数量: 1	SYS指示灯 x1
RTC	数量: 1	内置

DC

数量: 1

12V

整机实物图片



2 外围功能接口

2.1 固件烧录

固件地址:<https://pan.baidu.com/s/1C2q-PsldNRqX3QvYYk7o5Q?pwd=1234>

主板下载固件需要先将设备切换为 Loader 模式，Loader 模式的操作流程方法如下：

1. 断开主板的所有电源；
2. 使用TYPE-C数据线，连接PC端的USB接口（如下图红色框所示）；
3. 按住REC按键（如下图黄色框所示），并保持；
4. 主板供电（如下图绿色框所示）；



5. 烧录工具显示“发现一个LOADER设备”后，即可松开REC按键；
6. 点击【升级固件】界面的【固件】按键，在系统文件资源管理器中选择烧录固件；



7. 加载完成后，点击【升级】按键开始下载程序，烧录完成开发板将会重新启动。

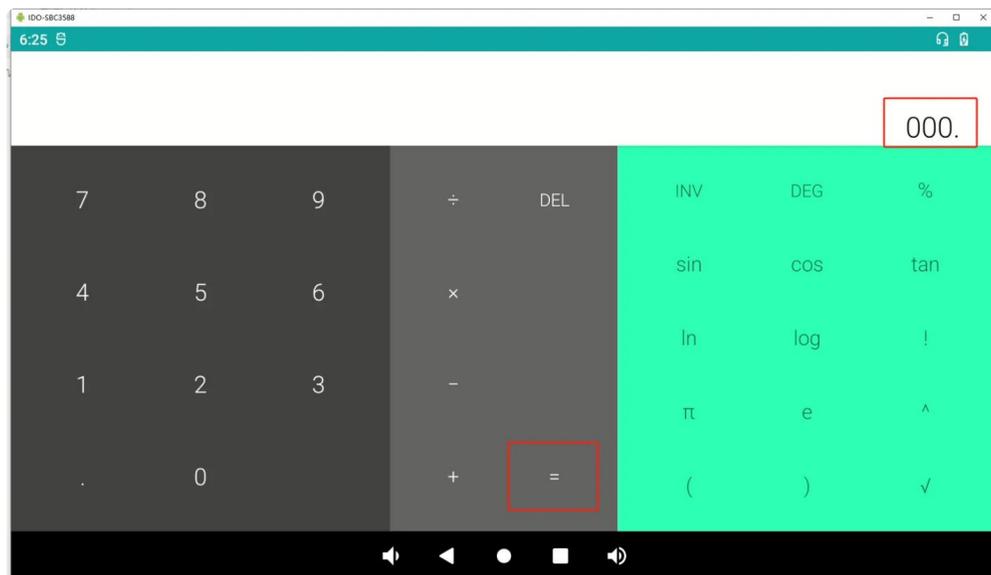


2.2 打开测试工具方法

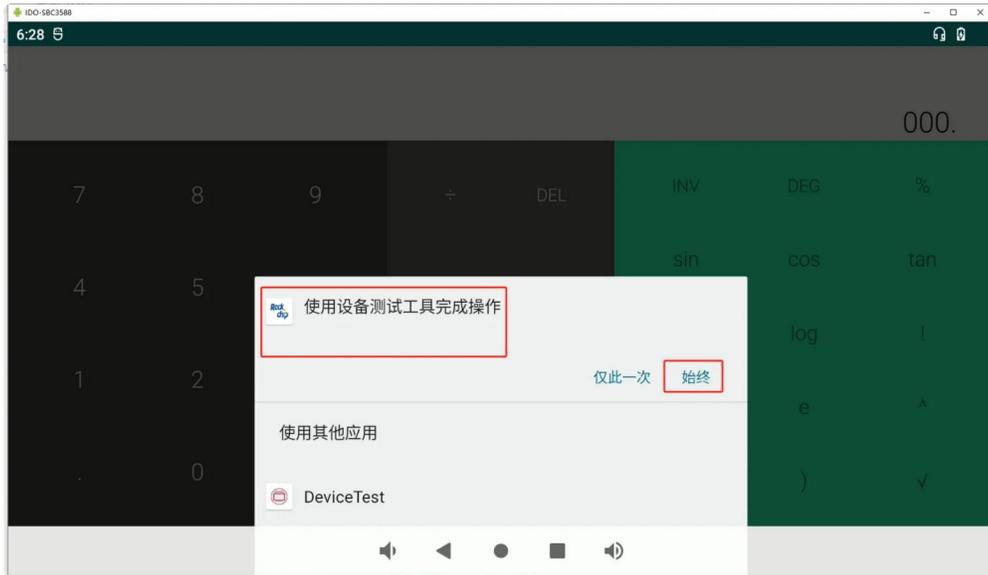
系统菜单找到【计算器】并打开，如下图所示：



输入【000.】然后按【=】号，如下图所示：



选择【设备测试工具】并进入，如下图所示：



进入到功能测试界面，如下图所示：



2.3 指示灯

设备上电后，绿色指示灯亮起，正常启动后以1秒2次频率闪动，如下图所示：



- 开机：绿灯闪烁
- 关机：绿灯熄灭
- 异常：绿灯常亮（或灭）

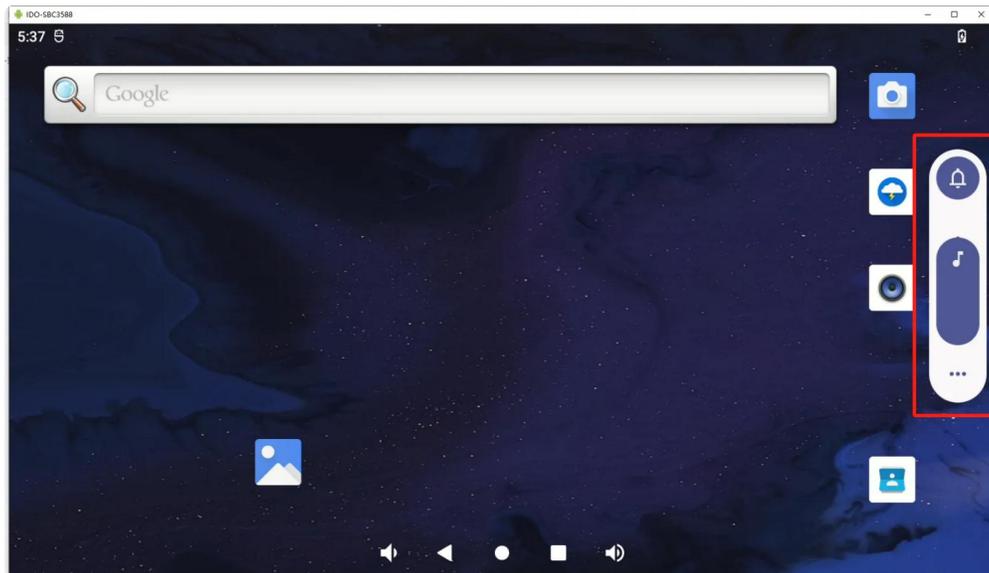
测试项目	要求	结果
系统指示灯	上电后1秒闪2次	✓

2.4 REC

【REC】按键，如下图所示：



系统正常开机状态下按下【REC】按键，系统上报音量弹窗，如下图所示：



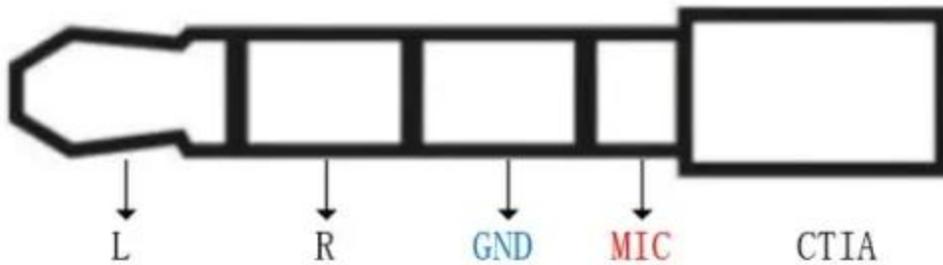
测试项目	要求	结果
REC按键	系统启动后上报的是音量+键值	✓

2.5 耳机

耳机接口，如下图所示：



CTIA标准四段式耳机，定义如下：



将耳机接入3.5mm耳机口，测试工具打开【喇叭】，如下图所示：



进入到扬声器测试界面，系统会播放双声道音乐，如下图所示：



声音通过3.5mm耳机孔输出到耳机，调节声音大小音乐清晰无杂音。

测试项目	要求	结果
3.5MM耳机孔	适配3.5耳机孔，出生音质无杂音	✓

2.6 耳麦录音

保持耳机接入，选择【录音】，如下图所示：



此时系统会通过耳机麦克风录音5秒，倒计时结束则录音结束，测试时需要在5秒倒计时里对着耳机麦克风以正常音量说出自己想要录音的内容，观察指针录音是否随声音摆动



录音结束后系统会自动播放刚才录音的内容，如果想要再次测试，可以点击【重测】按钮重新开始测试录音和放音，如下图所示：



测试项目	要求	结果
3.5MM耳机MIC录音	录音过程指针摆动正常	✓
	录制语音在播放时清晰无杂音	✓

2.7 TF卡槽

TF卡槽，如下图所示：

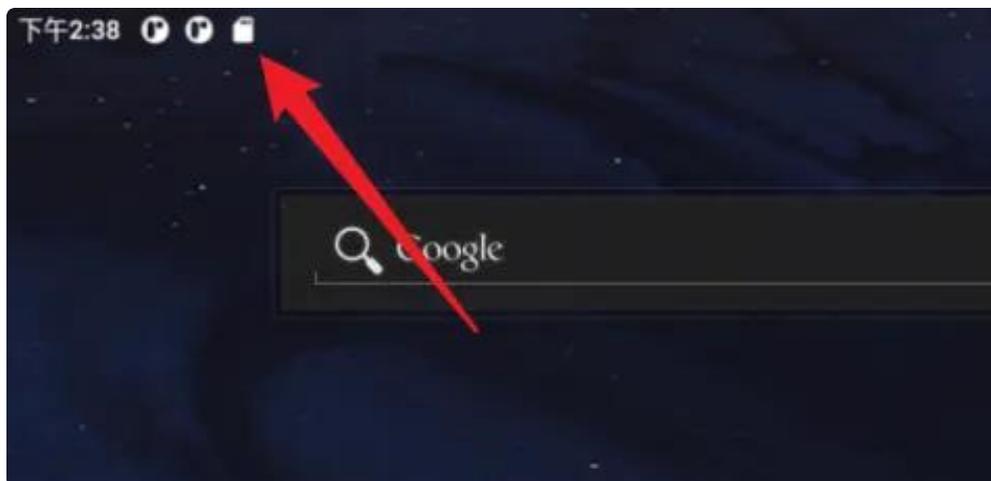


- 支持FAT32和NTFS格式分区自动挂载

TF卡正面朝下，触点朝上插入盒子TF卡槽，如下图所示：



插入TF卡后，安卓桌面/菜单栏界面界面会显示TF卡标识，如下图所示：



测试项目	要求	结果
TF卡	识别TF卡正常、读写文件正常	✓

2.8 HDMI

HDMI接口，如下图所示：



- 支持最大分辨率：8K@60fps

测试项目	要求	结果
显示	HDMI画面显示正常	✓
	HDMI音频输出正常	✓
	测试分辨率：4K@60fps	✓
	HDMI热插拔正常	✓

2.9 HDMI_IN

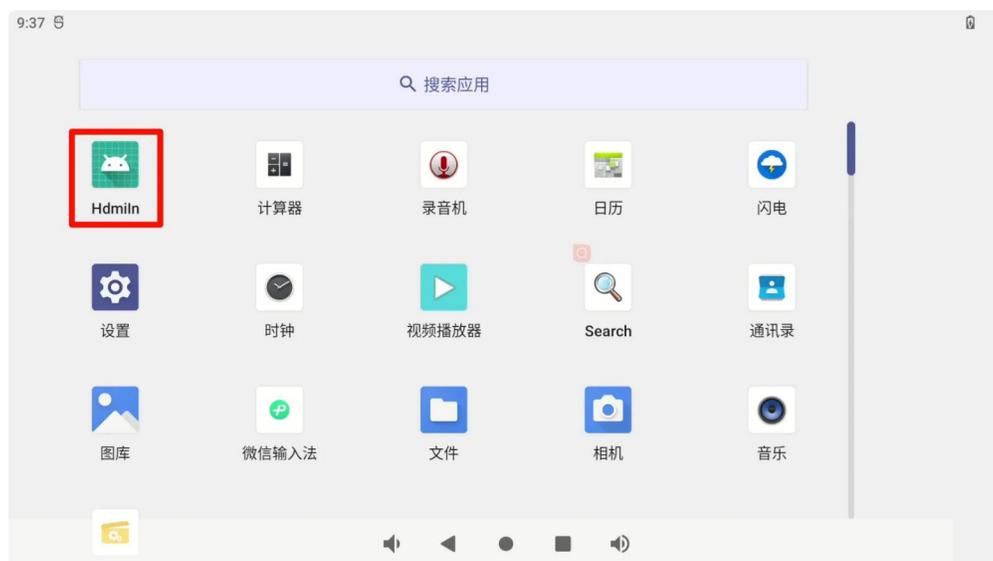
HDMI-IN接口，如下图所示：



播放设备使用 Micro-HDMI 数据线连接盒子HDMI-IN接口



打开【Hdmiln App】，如下图所示：



播放设备的HDMI画面就能接入到盒子显示，如下图所示：



测试项目	要求	结果
HDMI_IN	HDMI_IN输入画面可通过HDMI/DP接口输出到显示器上	✓
	HDMI_IN音频输入正常	✓
	支持最大分辨率：4K@30fps	✓

2.10 USB

USB接口，如下图所示：



测试项目	要求	结果
USB功能测试	USB1需要兼容U盘、键盘、鼠标	✓
	USB2需要兼容U盘、键盘、鼠标	✓
	USB3需要兼容U盘、键盘、鼠标	✓

	USB4需要兼容U盘、键盘、鼠标	✓
--	------------------	---

USB端口电源开关：

USB端口	动作	命令
USB1	关闭电源	echo 0 > /sys/devices/platform/leds/leds/usb_host1_pwr/brightness
	开启电源	echo 1 > /sys/devices/platform/leds/leds/usb_host1_pwr/brightness
USB2	关闭电源	echo 0 > /sys/devices/platform/leds/leds/usb_host2_pwr/brightness
	开启电源	echo 1 > /sys/devices/platform/leds/leds/usb_host2_pwr/brightness
USB3	关闭电源	echo 0 > /sys/devices/platform/leds/leds/usb_host3_pwr/brightness
	开启电源	echo 1 > /sys/devices/platform/leds/leds/usb_host3_pwr/brightness
USB4	关闭电源	echo 0 > /sys/devices/platform/leds/leds/usb_host4_pwr/brightness
	开启电源	echo 1 > /sys/devices/platform/leds/leds/usb_host4_pwr/brightness

测试项目	要求	结果
USB开关测试	USB1开关测试	✓
	USB2开关测试	✓
	USB3开关测试	✓
	USB4开关测试	✓

2.11 TypeC 接口

TypeC 接口，如下图所示：



- 支持Host、Device模式自动切换
- 支持DP显示输出

2.11.1 Device从机模式

使用TypeC数据线连接电脑，烧录工具能发现一个ADB设备，如下图所示：



测试项目	要求	结果
TYPE-C Device	烧录工具可发现ADB设备，可使用开发工具对盒子进行功能调试	✓

2.11.2 Host主机模式

接入TypeC设备，或通过TypeC to USB-A转接头接入USB外设，如下图所示：



测试项目	要求	结果
TYPE-C Host	可识别U盘、键盘、鼠标并正常使用	✓

2.11.3 DP模式

通过TypeC全功能数据线接入DP显示器，或通过TYPE-C to HDMI数据线连接HDMI显示器

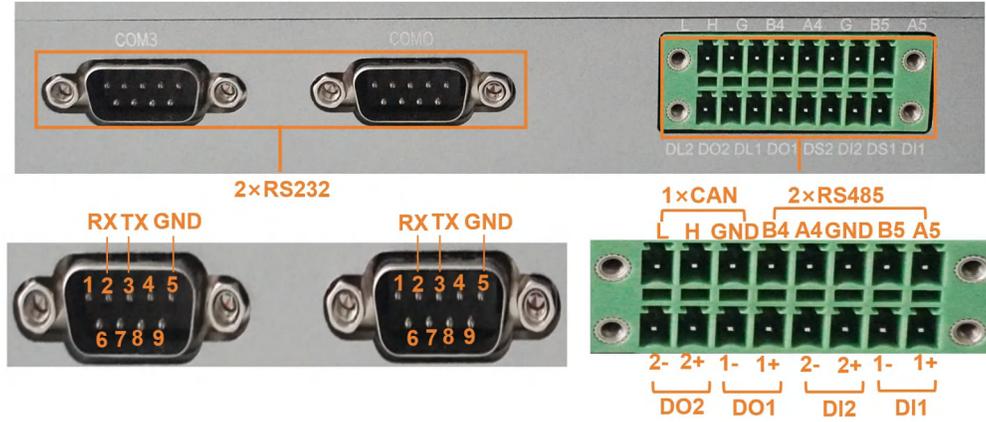


测试项目	要求	结果
TYPE-C DP	主板画面通过TYPE-C DP输出正常，DP声音输出正常	✓

2.12 串口及IO接口

2.12.1 硬件接口介绍

串口及IO接口位于盒子背面，如下图所示：



DB-9接口功能及节点，如下表所示：

标号	管脚功能	设备节点
COM0	RS232	dev/ttyS0
COM3	RS232	dev/ttyS3

凤凰端子上排引脚功能和节点，如下表所示：

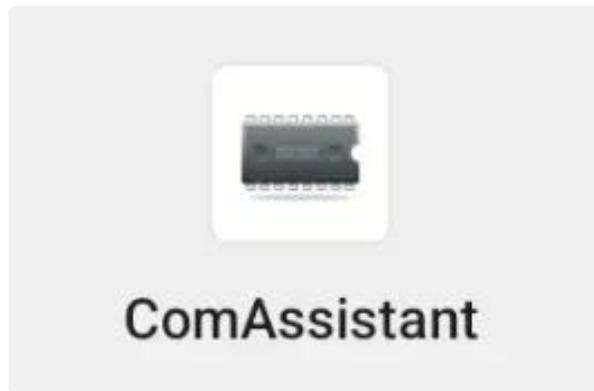
标号	管脚功能	设备节点
A5	RS485_A	/dev/ttyS5
B5	RS485_B	
G	GND	/
A4	RS485_A	/dev/ttyS4
B4	RS485_B	
G	GND	/
H	CAN_H	can0
L	CAN_L	

凤凰端子下排引脚功能和GPIO节点，如下表所示：

标号	管脚功能	GPIO节点	GPIO口模式
DI1	DI1	GPIO1_A3	输入
DS1	DI_SS1		
DI0	DI2	GPIO1_A0	输入
DS2	DI_SS2		
DO1	DO1	GPIO1_A1	输出
DL1	DO1_L		
DO2	DO2	GPIO1_A2	输出
DL2	DO2_L		

2.12.2 串口测试方法

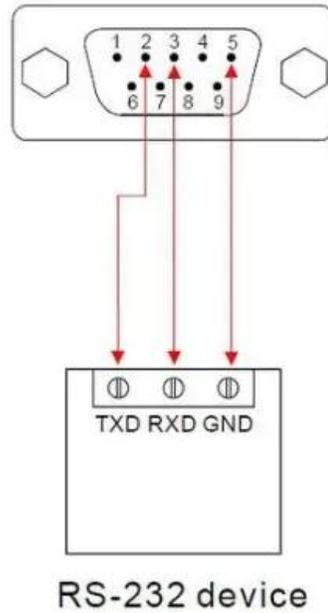
盒子安卓系统下，安装串口工具【ComAssistant.apk】：[ComAssistant.zip](#)



在电脑侧Windows系统下安装串口调试工具软件。通过两个串口工具收发来验证串口通信功能。

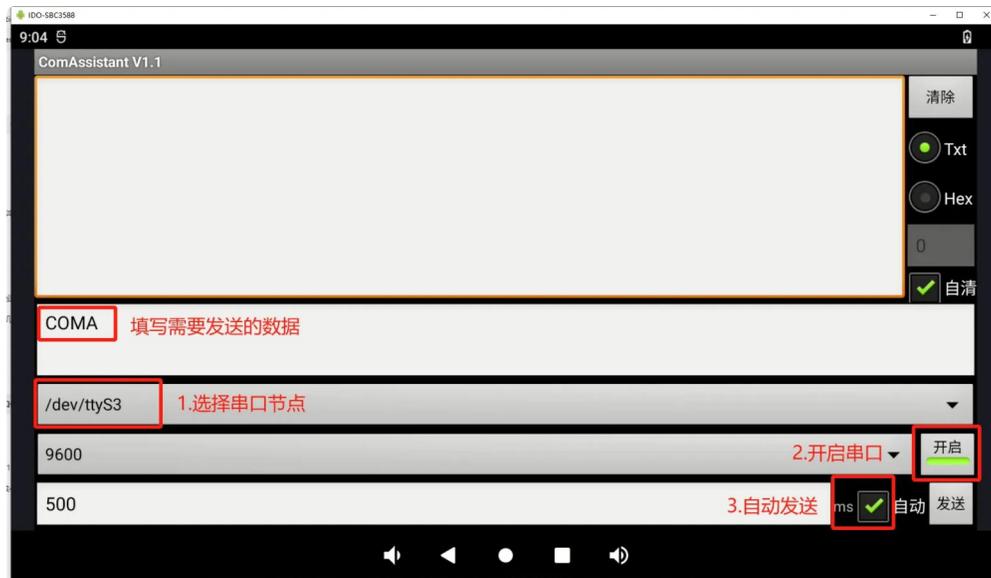
RS232接口 (COM0, COM3)测试方法

电脑使用RS232串口工具，连接盒子COM接口，RX-TX交叉连接，必须要共GND，如下图所示：



RS232发送

盒子安卓下串口工具选择对应的串口节点 (COM0: /dev/ttyS0, COM3: /dev/ttyS3), 波特率默认 9600, 发送" COMA "给电脑, 如下图所示:



从电脑串口工具可以看到盒子发过来的数据“ COMA ”, 如下图所示:



RS485接口测试方法

两路RS485，A4/B4接线对应串口 /dev/ttyS4 ， A5/B5接线对应串口/dev/ttyS5。

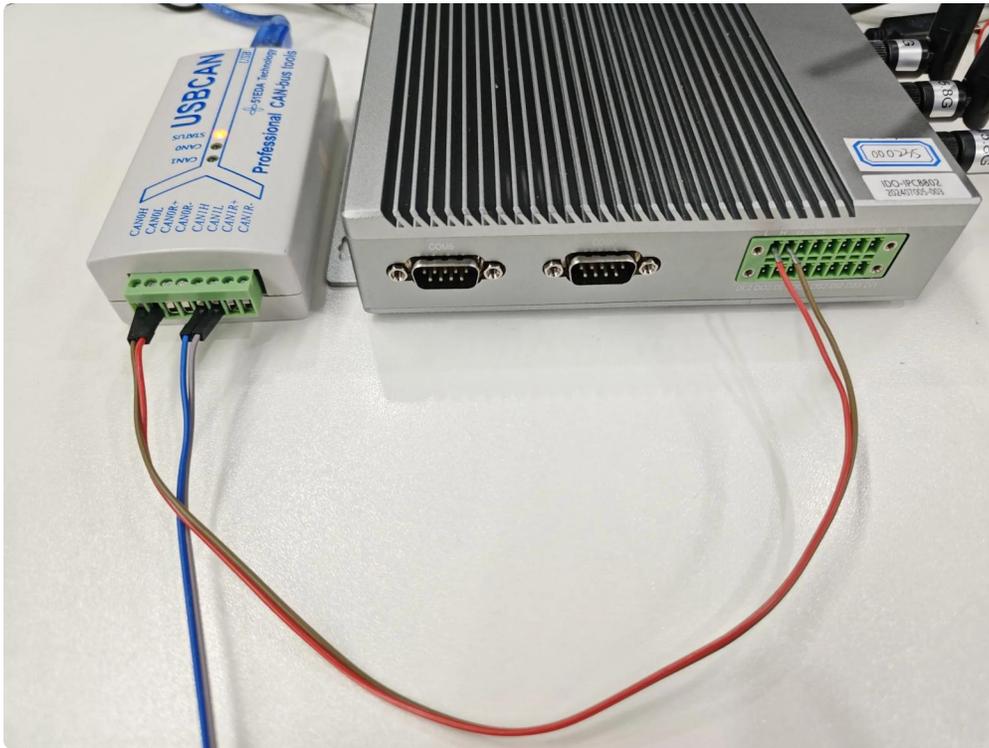
接线方法：USB转RS485模块连接到电脑侧，485总线A-A, B-B 直连即可，GND建议也连接在一起。

软件测试方法与RS232类似，只需要在安卓串口工具上选择对应的设备节点即可。

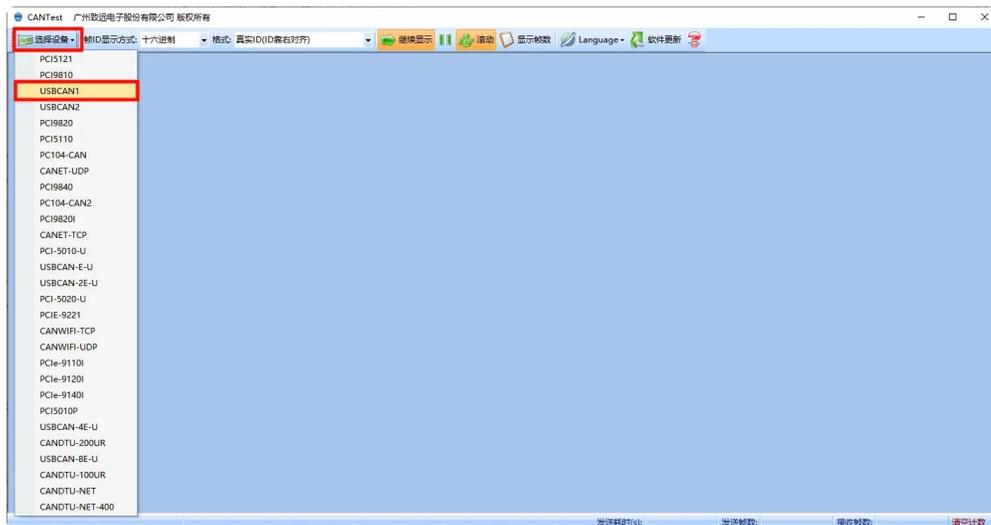
端口号	节点	类别	测试要求	结果
COM0	dev/ttyS0	✓ RS232	与电脑通讯正常，数据收发完整无丢包	✓ PASS
COM3	dev/ttyS3	✓ RS232	与电脑通讯正常，数据收发完整无丢包	✓ PASS
RS485 (A4/B4)	dev/ttyS4	✓ RS485	与电脑通讯正常，数据收发完整无丢包	✓ PASS
RS485 (A5/B5)	dev/ttyS5	✓ RS485	与电脑通讯正常，数据收发完整无丢包	✓ PASS

2.12.3 CAN口发送

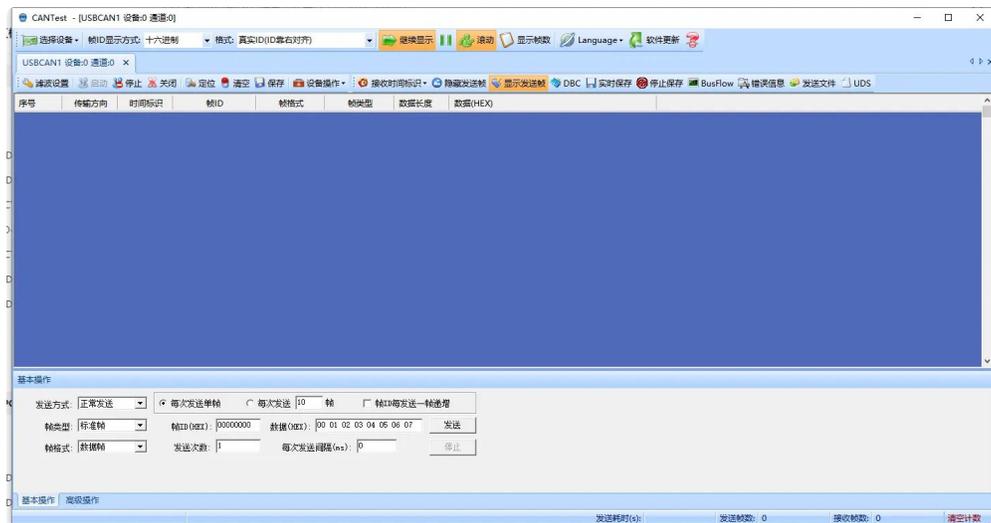
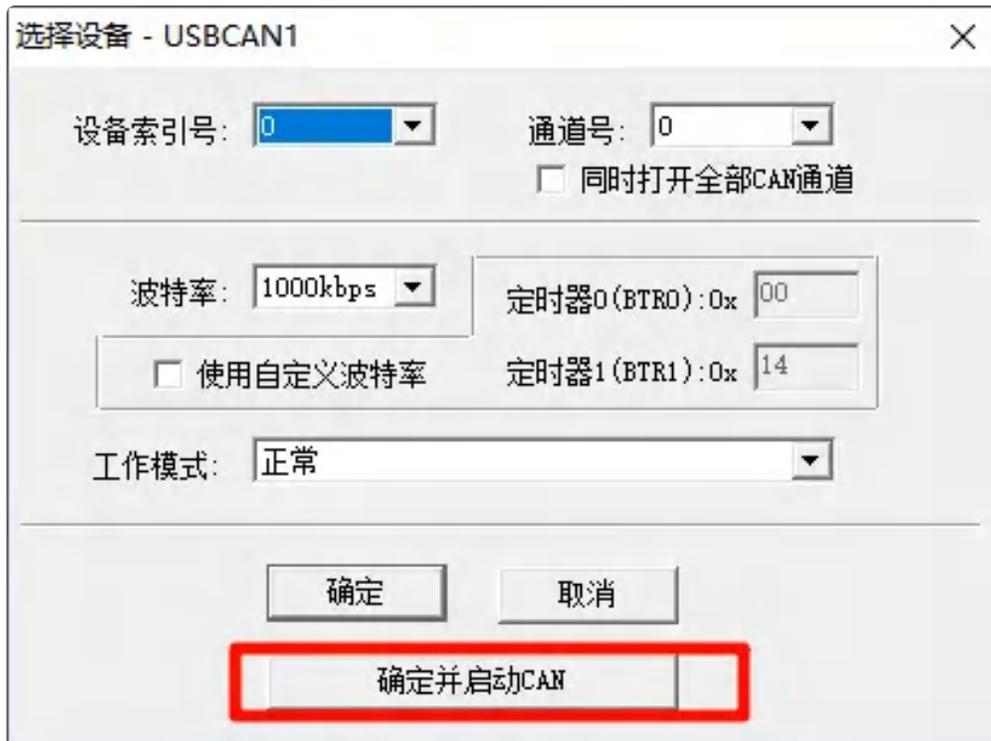
使用CAN转USB工具连接电脑跟盒子，CAN_H接CAN_H，CAN_L接CAN_L，如下图所示：



电脑打开【CANTest】工具，选择设备，打开【USBCAN1】选项，如下图所示：



默认设置波特率为1000kbps，点击【确认并启动CAN】，如下图所示：



打开CMD命令窗口，使用 `shell` 命令进入盒子，如下图所示：

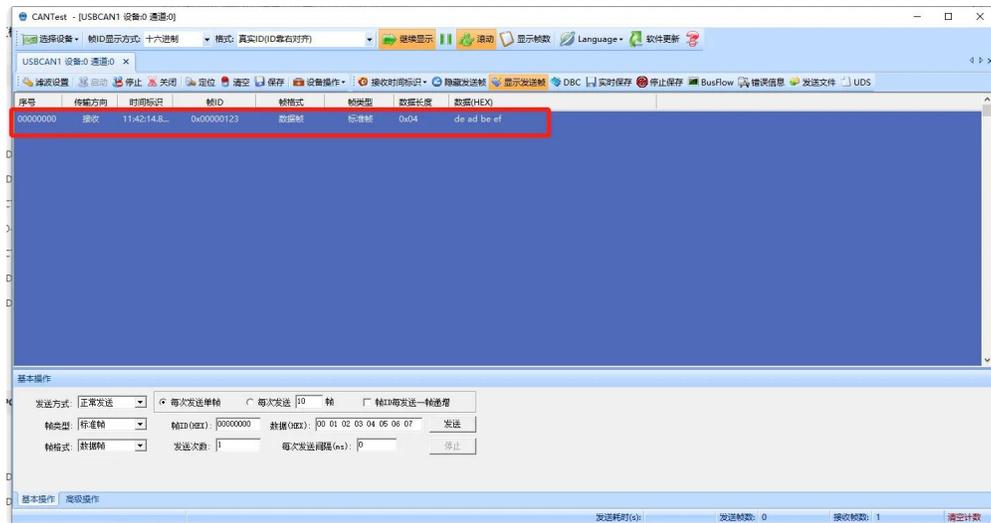
```
C:\Users\ID-Tbr>adb shell
IDO-SBC3588:/ $ su
IDO-SBC3588:/ #
```

依次输入下面命令，其中”DEADBEEF“为需要发送的数据

发送模式命令如下 Shell

- 1 IDO-SBC3588:/ # ip link set can0 down
- 2 IDO-SBC3588:/ # ip link set can0 type can bitrate 1000000 dbitrate 1000000 fd on
- 3 IDO-SBC3588:/ # ip link set can0 up
- 4 IDO-SBC3588:/ # cansend can0 123#DEADBEEF

电脑CANTest工具就能收到来自盒子can口发出来的数据，如下图所示：



2.12.4 CAN口接收

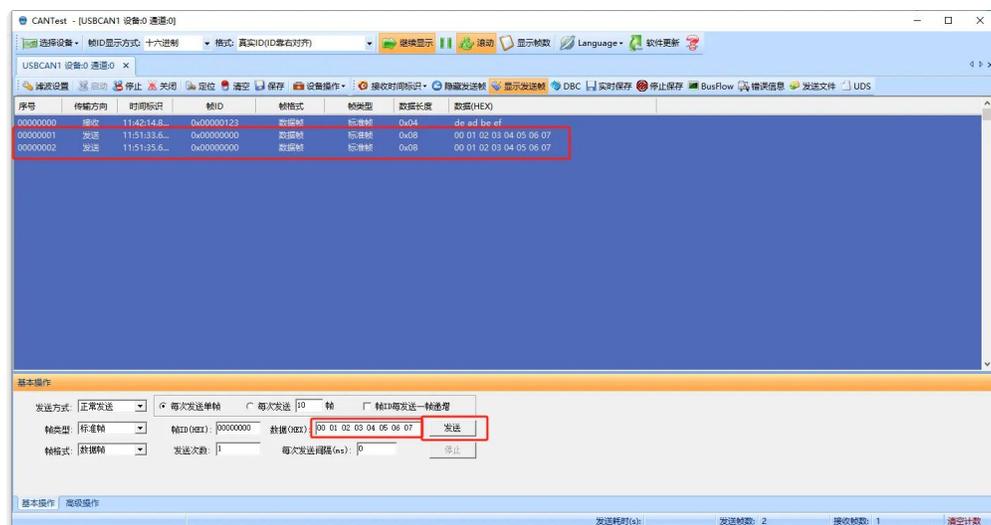
CMD窗口输入命令

接收模式命令如下

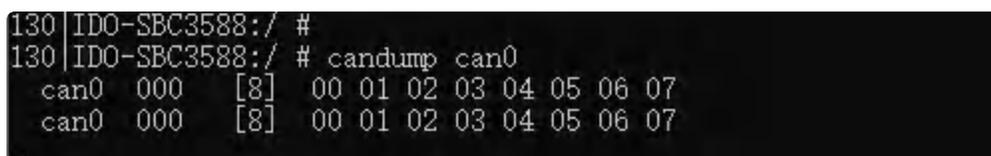
Shell

```
1 ID0-SBC3588:/ # candump can0
```

然后从CANTest工具发送数据过来，如下图所示：



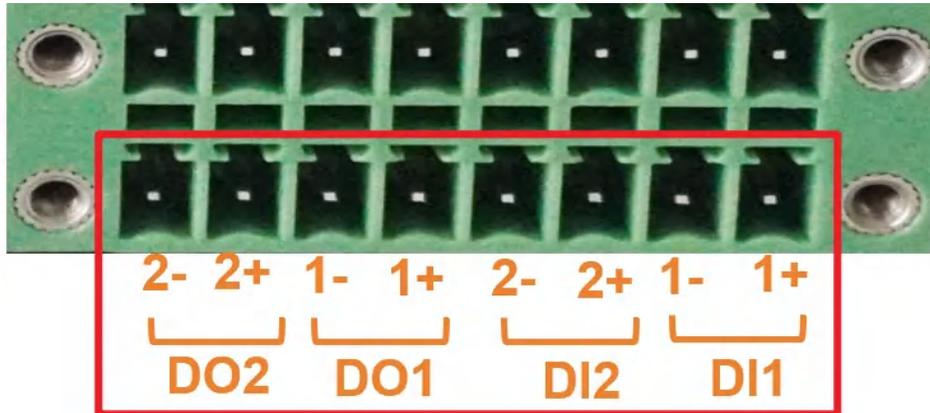
然后CMD窗口（盒子）能收到电脑发过来的数据，如下图所示：



测试项目	要求	结果
CAN数据通讯	盒子CAN口与电脑数据通讯正常，数据收发正常	✓

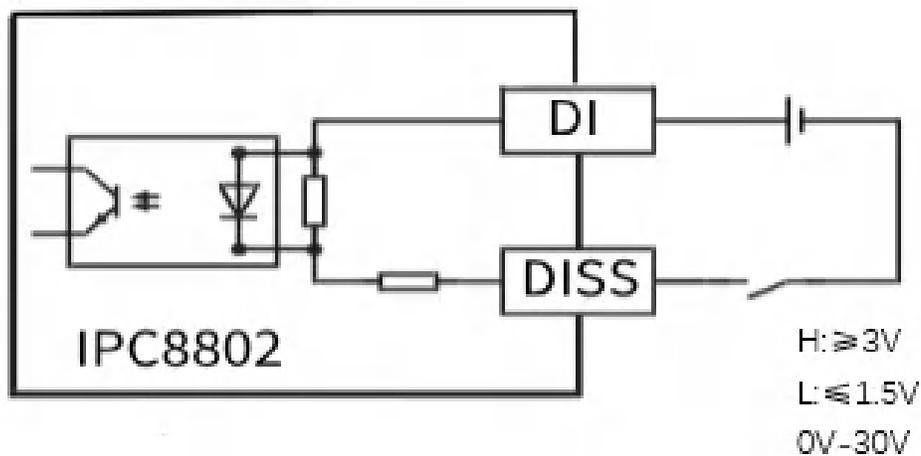
2.13 DI/DO口使用

凤凰端子下排引脚为DIDO接口，其中左边4个为DO口，右边4个为DI口，如下图所示：



2.13.1 DI 端口测试方法

DI接口的内部逻辑及接线示意图如下：



DI 接口通过一路光耦实现，外部接入电源 ($>3V$) /GND，可以使光耦导通。其中DI1 接正极，DIS1接负级。DI2 同。

对应的设备节点

DI	读取命令	逻辑结果
DI1	gpioget 1 3	1: 开路 0: 有源输入
DI2	gpioget 1 0	1: 开路 0: 有源输入

测试方法：

ADB命令行下执行 `gpioget` 指令，如下 读取DI1的状态：

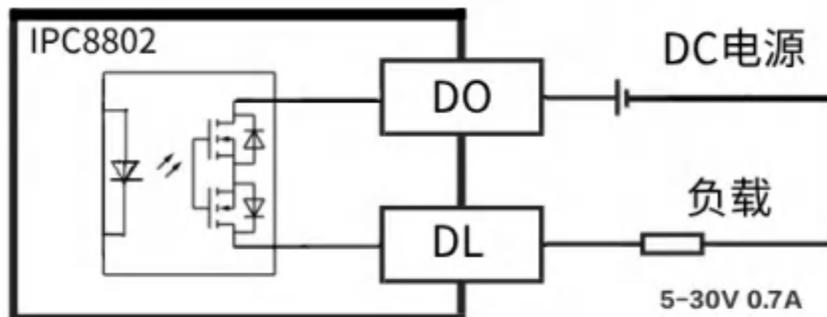
<p>▼ 读取DI1的命令</p> <pre>1 ID0-SBC3588:/ # gpioget 1 3</pre>	Shell
<p>▼ 读取DI2的命令</p> <pre>1 ID0-SBC3588:/ # gpioget 1 0</pre>	Shell

测试项目	要求	结果
DI1端口	DI1接入外部电源时，可读到 <code>gpioget 1 3</code> 值为 <code>0</code> （默认 <code>1</code> ）	✓
DI2端口	DI2接入外部电源时，可读到 <code>gpioget 1 0</code> 值为 <code>0</code> （默认 <code>1</code> ）	✓

2.13.2 DO 端口测试方法

DO的内部是采用了光固态继电器输出，支持48V内 0.5A 的导通输出能力。当IO输出1时，DO的输出两侧导通，当IO输出0时，DO的输出两侧开路。光固态继电器 DO 输出是双向的，没有正负极之分。

DO的内部逻辑及接线示意图如下：



对应的设备节点

DO	GPIO节点	逻辑
DO1	GPIO1_A1	1: 导通 0: 开路
DO2	GPIO1_A2	1: 导通 0: 开路

测试方法:

ADB命令行下执行 `gpioset` 指令, 如下 控制DO1的状态:

```
▼ 控制DO1的命令行 Shell |
1 #控制D01导通
2 ID0-SBC3588:/ # gpioset 1 1=1
3 #控制D01开路
4 ID0-SBC3588:/ # gpioset 1 1=0
```

```
▼ 控制DO2的命令行 Shell |
1 #控制D02导通
2 ID0-SBC3588:/ # gpioset 1 2=1
3 #控制D02开路
4 ID0-SBC3588:/ # gpioset 1 2=0
```

测试项目	要求	结果
DO1端口	命令 <code>gpioset 1 1=1</code> 控制DO1拉高, DO1和DL1被导通 (默认断开)	✓
DO2端口	命令 <code>gpioset 1 2=1</code> 控制DO2拉高, DO2和DL2被导通 (默认断开)	✓

2.13.3 APK读写DIDO方法

读取DI状态

执行命令: gpioget 1 3

Java

```
1  +import java.io.IOException;
2
3  public class MainActivity extends AppCompatActivity {
4      @Override
5      protected void onCreate(Bundle savedInstanceState) {
6          super.onCreate(savedInstanceState);
7          EdgeToEdge.enable(this);
8
9          // 执行命令并获取结果
10 +     String result = CommandGetGPIO("gpioget 1 3");    //获取DI1状态
11     }
12
13 +     private String CommandGetGPIO(String command) {
14 +         StringBuilder output = new StringBuilder();
15 +         try {
16 +             Process process = Runtime.getRuntime().exec(command);
17 +             BufferedReader reader = new BufferedReader(new InputStreamReader(process.getInputStream()));
18 +             String line;
19 +             while ((line = reader.readLine()) != null) {
20 +                 output.append(line).append("\n");
21 +             }
22 +             process.waitFor();
23 +         } catch (IOException | InterruptedException e) {
24 +             e.printStackTrace();
25 +         }
26 +         return output.toString().trim();
27 +     }
```

设置DO状态

执行命令行: gpio set 1 1=1

Java

```
1  +import java.io.IOException;
2
3  public class MainActivity extends AppCompatActivity {
4      @Override
5      protected void onCreate(Bundle savedInstanceState) {
6          super.onCreate(savedInstanceState);
7          EdgeToEdge.enable(this);
8      +    try {
9      +        Runtime.getRuntime().exec("gpio set 1 1=1");    //使能D01
10     +    } catch (IOException e) {
11     +        throw new RuntimeException(e);
12     +    }
13     }
```

旧固件使用上面方法没有权限，可以使用su命令执行

```
1 private void CommandGetGPIO() {
2     new Thread(() -> {
3         +     Process process = null;
4         +     DataOutputStream os = null;
5         try {
6             // 获取 root 权限
7         +     process = Runtime.getRuntime().exec("su");
8         +     os = new DataOutputStream(process.getOutputStream());
9         +
10        +     // 执行 gpioget 命令
11        +     os.writeBytes("gpioget 1 3\n");    //获取DI1状态
12        +     os.writeBytes("exit\n");
13        +     os.flush();
14
15        // 等待命令执行完成
16        int exitCode = process.waitFor();
17        if (exitCode == 0) {
18            updateUI("gpioget executed successfully.");
19        } else {
20            updateUI("Failed to execute gpioget.");
21        }
22    } catch (Exception e) {
23        e.printStackTrace();
24        updateUI("Error executing command: " + e.getMessage());
25    } finally {
26        if (os != null) {
27            try {
28                os.close();
29            } catch (IOException e) {
30                e.printStackTrace();
31            }
32        }
33        if (process != null) {
34            process.destroy();
35        }
36    }
37    }).start();
38 }
39
```

三、网络测试

3.1 Ethernet

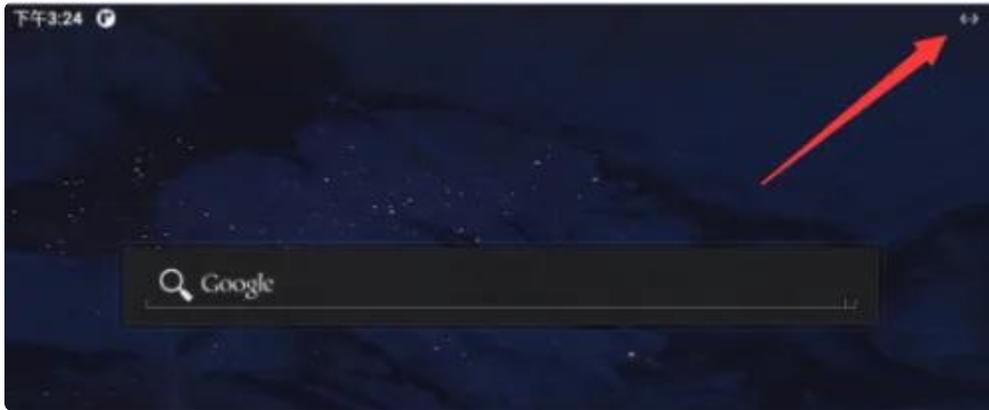
主板有两路千兆以太网接口LAN0和LAN1，如下图所示：



LAN0和LAN1分别对应系统端口为Ethernet0和Ethernet1，【菜单】->【设置】->【网络和互联网】，如下图所示：



以太网接口默认支持 HDCP，只需要将以太网接口连接路由器即可为主板动态分配 IP 地址。如下图所示：



3.1.1 LAN0

通过路由器连接外网，可在Ethernet0选项里查看相关信息，如下图所示：



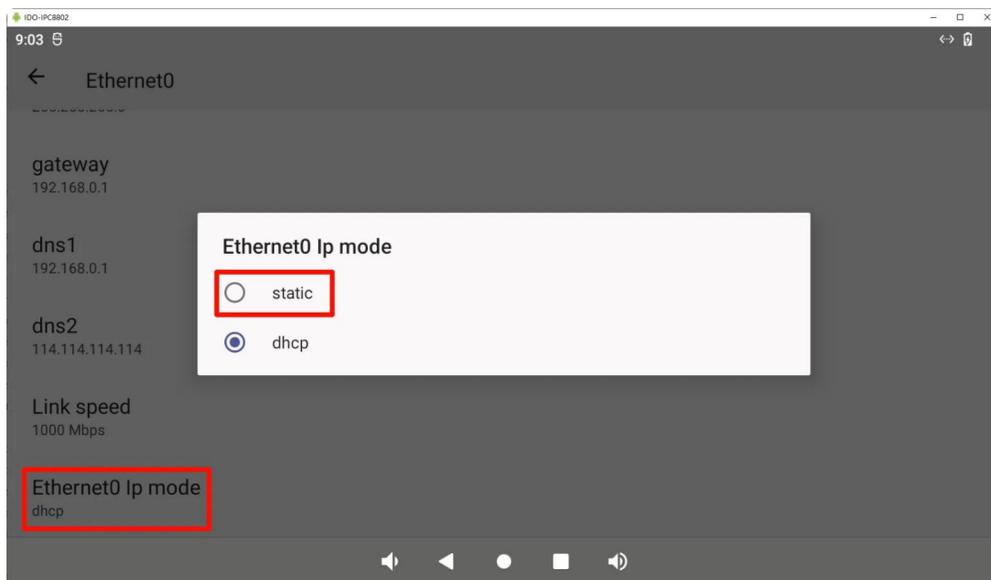
也可通过ADB命令ifconfig查看LAN0网络相关信息，如下图所示：

```
eth0      Link encap:Ethernet  HWaddr a2:5a:a0:70:10:63  Driver rk_gmac-dwmac
          inet addr:192.168.0.132  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::cffe:c69c:e687:1c9d/64 Scope: Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:168769  errors:0  dropped:0  overruns:0  frame:0
          TX packets:34922  errors:0  dropped:0  overruns:0  carrier:0
          collisions:0  txqueuelen:1000
          RX bytes:216074227  TX bytes:2407362
          Interrupt:89
```

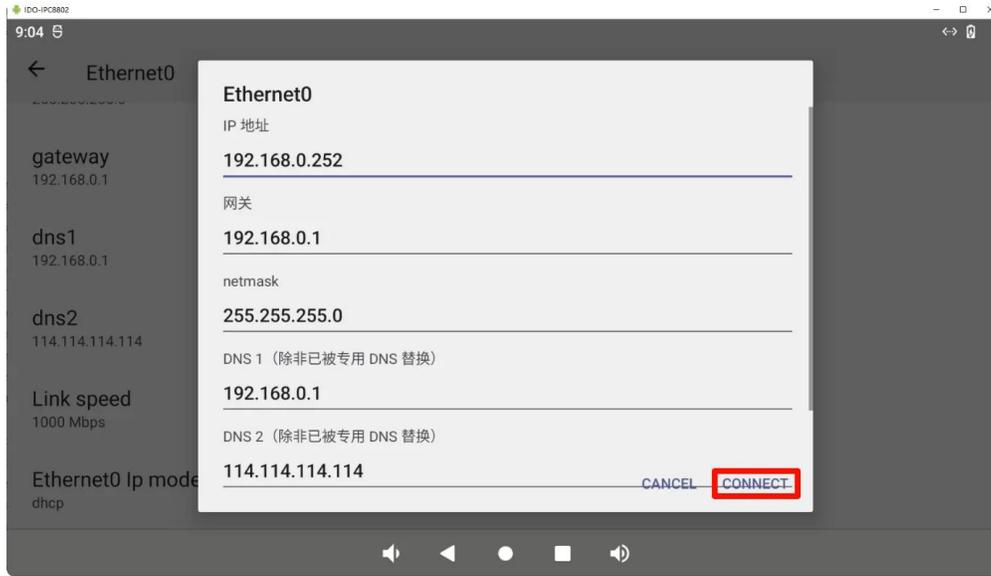
带网络连接速度100M/1000M自动识别功能，如下图所示：



Ethernet0静态IP设置： 点击【Ethernet0 Ip mode】，如下图所示：



选择【static】，在弹窗中输入想要设置的静态IP，输入完成后点击【CONNECT】，如下图所示：



系统会自动返回【Ethernet0】界面，此时可发现【Ethernet0 Ip mode】选项的值变成了【static】代表设置成功，如下图所示：



如想用回自动分配IP，【Ethernet0 Ip mode】选择【dhcp】。

3.1.2 LAN1

通过路由器连接外网，可在Ethernet1选项里查看相关信息，如下图所示：



也可通过ADB命令ifconfig查看LAN1网络相关信息，如下图所示：

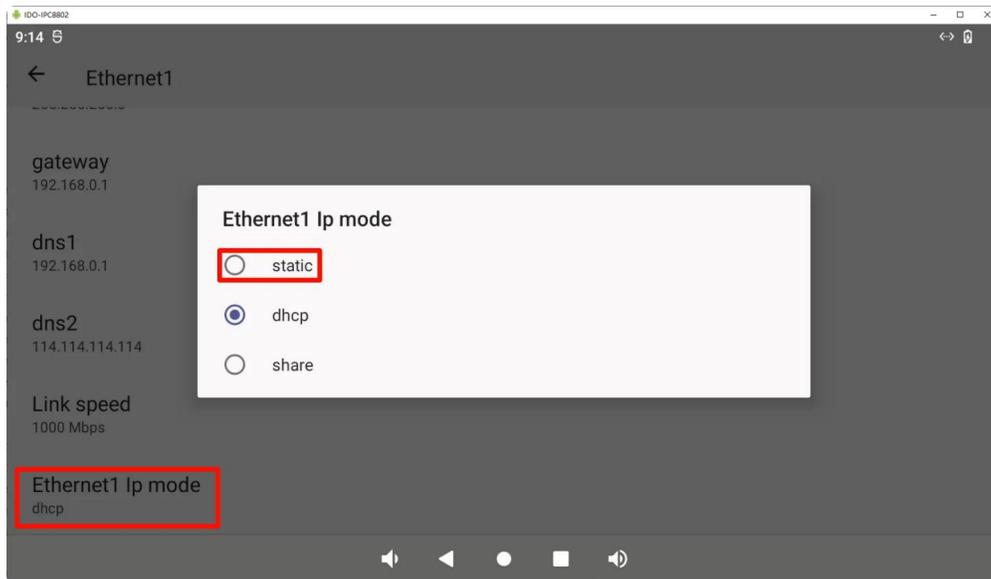
```
eth1      Link encap:Ethernet HWaddr ca:c1:79:68:45:ef Driver r8168
          inet addr:192.168.0.171 Bcast:192.168.0.255 Mask:255.255.255.0
          inet6 addr: fe80::7a83:883b:95f8:12f8/64 Scope: Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:286 errors:0 dropped:0 overruns:0 frame:0
          TX packets:50 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:28746 TX bytes:5922
          Interrupt:167 Base address:0x3000
```

带网络连接速度100M/1000M自动识别功能，如下图所示：

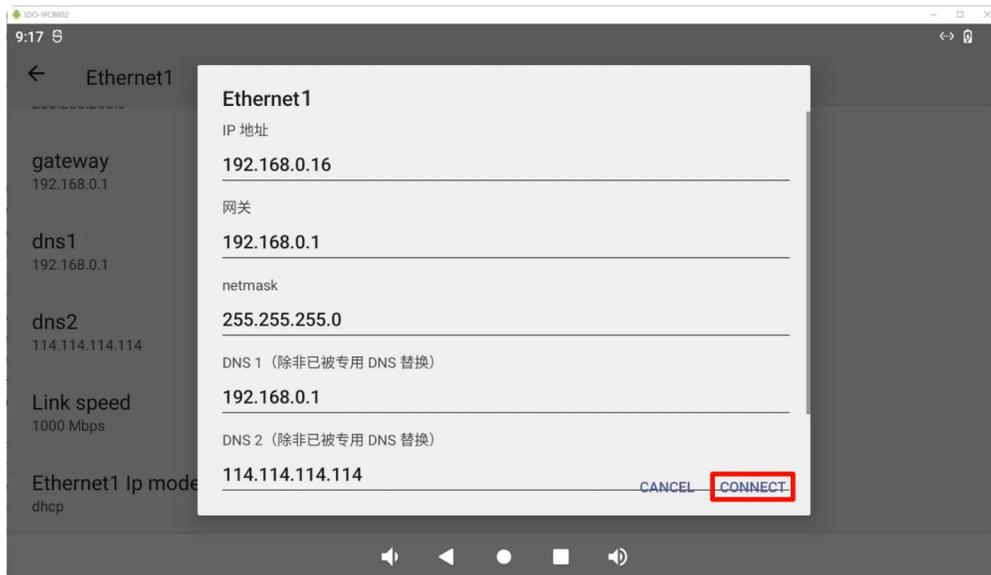




Ethernet1静态IP设置： 点击【Ethernet1 Ip mode】，如下图所示：



选择【static】，在弹窗中输入想要设置的静态IP，输入完成后点击【CONNECT】，如下图所示：



系统会自动返回【Ethernet1】界面，此时可发现【Ethernet1 Ip mode】选项的值变成了【static】代表设置成功，如下图所示：



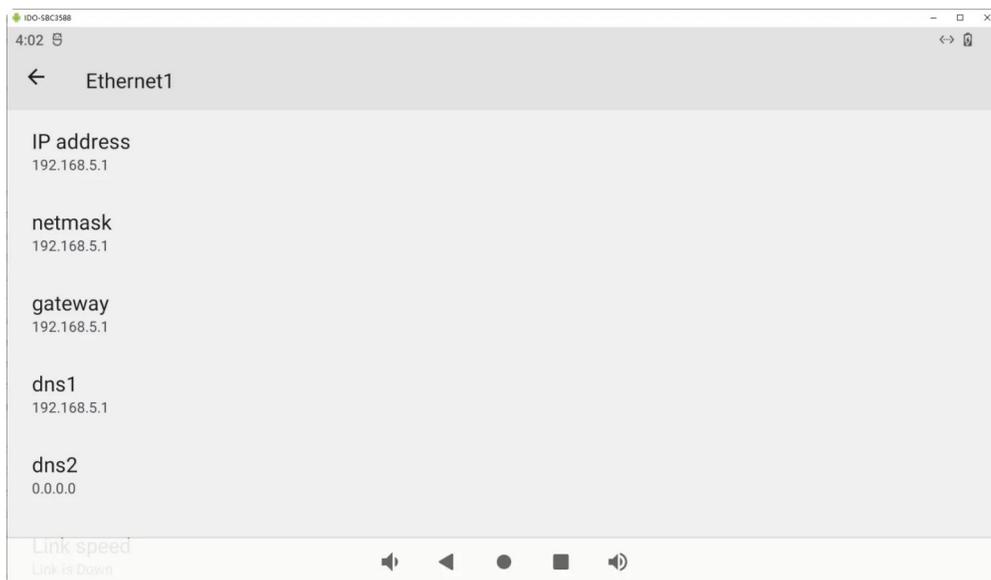
如想用回自动分配IP，【Ethernet1 Ip mode】选择【dhcp】。

3.1.3 以太网共享

当盒子LAN0通过路由器连接外网时，可通过以太网共享共享功能，【菜单】->【设置】->【网络和互联网】->【热点和网络共享】->【以太网共享】，把LAN1端口当作网络提供者连接下一个设备，把网络共享给下一个设备



可通过Ethernet1查看到共享网关等信息，如下图所示：



并且可以查看到当前LAN1处于share共享模式，如下图所示：



3.1.4 以太网-WLAN共享

盒子支持以太网络通过无线形式进行共享的功能，在已经接入以太网的状态下，在【菜单】->【设置】->【网络和互联网】->【热点和网络共享】->【WLAN热点】界面把【使用“WLAN热点”】开关打开，如下图所示：



并点击【热点密码】配置好热点密码，如下图所示：



手机或其他移动设备就可连接盒子的无线网络进行上网，如下图所示：



盒子可通过ifconfig命令查看共享信息，如下图所示：

```
wlan0    Link encap:Ethernet  HWaddr 54:78:c9:15:c0:b6  Driver bcm5dh_sdmmc
inet addr:192.168.5.1  Bcast:192.168.5.255  Mask:255.255.255.0
inet6 addr: fe80::5678:c9ff:fe15:c0b6/64  Scope: Link
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:645 errors:0 dropped:0 overruns:0 frame:0
TX packets:547 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:285898 TX bytes:182491
```

测试项目	要求	结果
------	----	----

网口	LAN0 LED 千兆为黄色闪烁	✓
	LAN1 LED 千兆为黄色闪烁	✓
	LAN0接入后 浏览器可以访问网络	✓
	LAN1接入后 浏览器可以访问网络	✓
	以太网共享	✓
	WLAN热点	✓

3.2 4G网络

3.2.1 4G网络

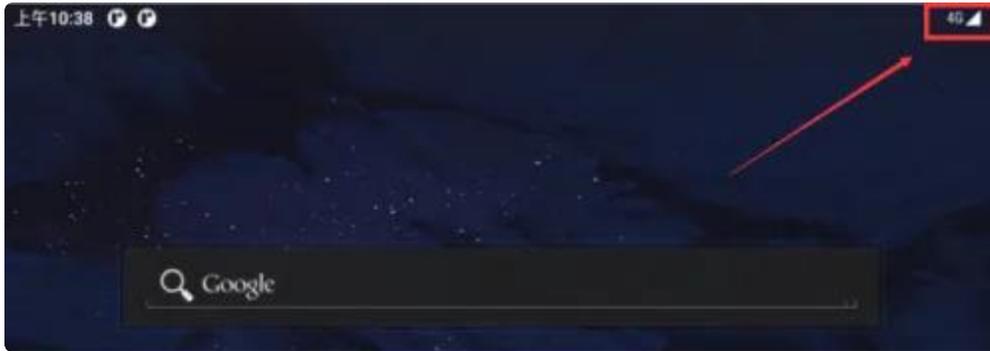
4G模块位于盒子内部，4G Micro SIM卡槽位，如下图所示：



将准备好的Micro SIM卡正面朝上，缺口朝外插入SIM卡槽，如下图所示：



开机后状态栏成功识别到 SIM卡，出现网络图标，如下图所示：



使用ifconfig命令可查看到相关网络信息，如下图所示：

```
wwan0  Link encap:Ethernet HWaddr 2e:0d:08:3d:6a:30 Driver qmi_wwan_q
inet addr:10.7.1.193 Mask:255.255.240.0
inet6 addr: fe80::2c0d:8ff:fe3d:6a30/64 Scope: Link
inet6 addr: 240e:47c:3000:8f14:793d:9edf:f228:7942/64 Scope: Global
inet6 addr: 240e:47c:3000:8f14:2c0d:8ff:fe3d:6a30/64 Scope: Global
UP RUNNING NOARP MTU:1500 Metric:1
RX packets:6830 errors:0 dropped:0 overruns:0 frame:0
TX packets:5394 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:4750783 TX bytes:1125747
```

3.2.2 4G网络共享

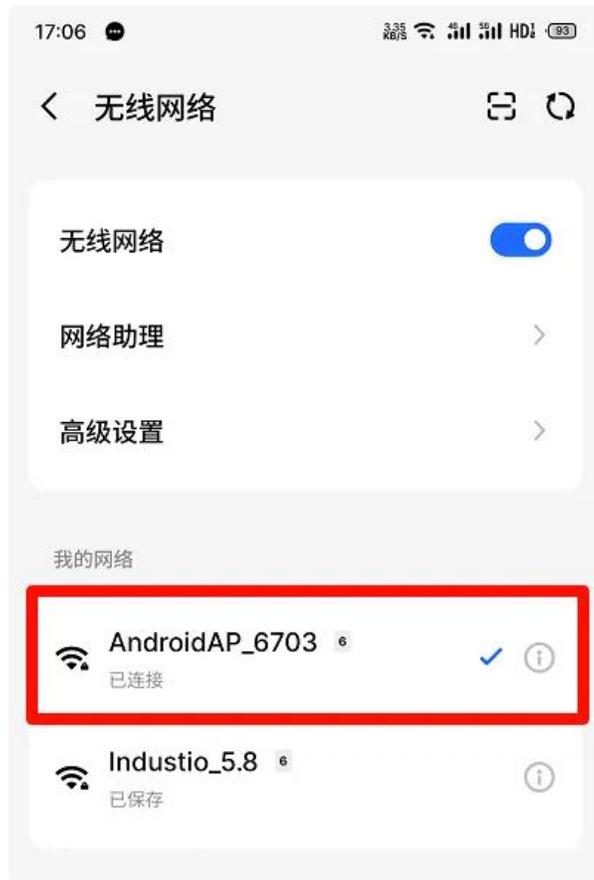
盒子支持4G网络共享功能，在3.2.1 4G网络功能正常的状态下，在【菜单】->【设置】->【网络和互联网】->【热点和网络共享】->【WLAN热点】界面把【使用“WLAN热点”】开关打开，如下图所示：



并点击【热点密码】配置好热点密码，如下图所示：



手机或其他移动设备就可连接盒子的无线网络进行上网，如下图所示：



盒子可通过ifconfig命令查看共享信息，如下图所示：

```
wlan0    Link encap:Ethernet  HWaddr 54:78:c9:15:c0:b6  Driver bcm57xx_sdmmc
inet addr:192.168.5.1  Bcast:192.168.5.255  Mask:255.255.255.0
inet6 addr: fe80::5678:c9ff:fe15:c0b6/64  Scope: Link
inet6 addr: 240e:47c:3000:8f14::8e/64  Scope: Global
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:4600  errors:0  dropped:0  overruns:0  frame:0
TX packets:6333  errors:0  dropped:0  overruns:0  carrier:0
collisions:0  txqueuelen:1000
RX bytes:992642  TX bytes:4618065
```

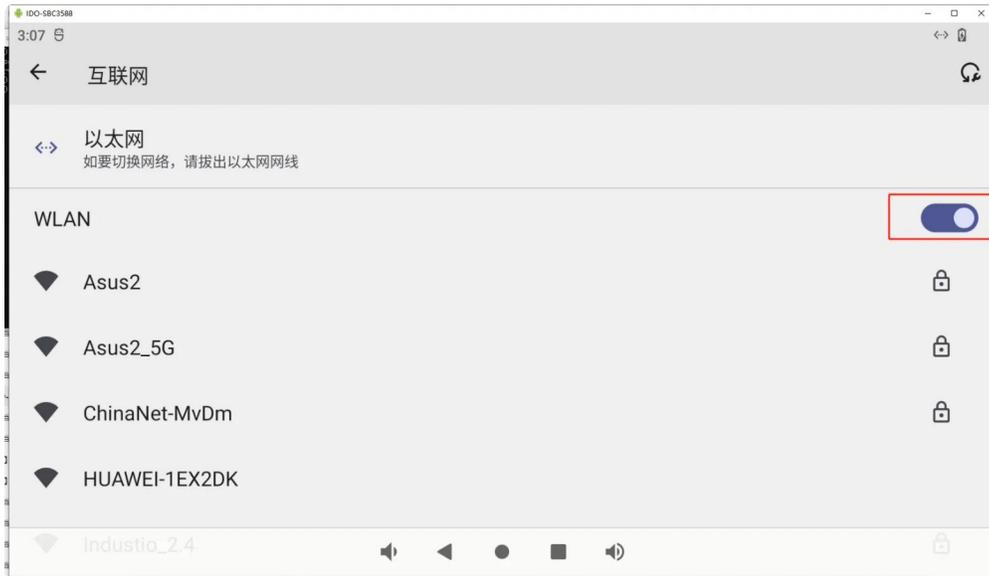
测试项目	要求	结果
SMI卡	可成功识别中国电信SMI卡	✓
	可成功识别中国联通SMI卡	✓
	可成功识别中国移动SMI卡	✓
	网络访问，打开浏览器可正常上网	✓

测试项目	要求	结果
WLAN共享	4G 网络 WLAN共享	✓

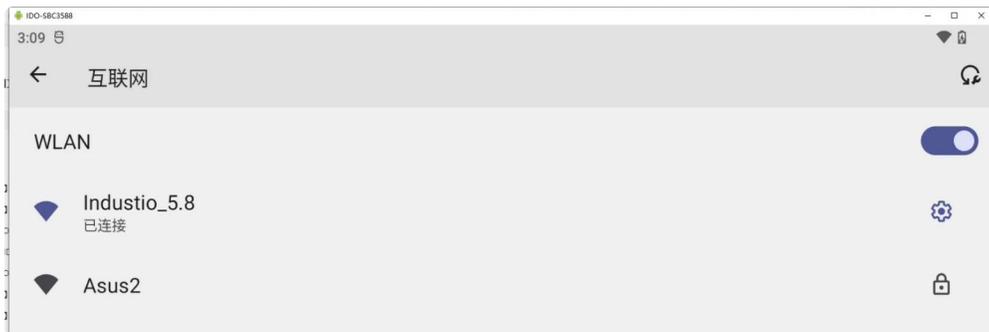
3.3 WiFi

【菜单】 -> 【设置】 -> 【网络和互联网】 -> 【互联网】 -> 【WLAN】

打开【WLAN】开关搜索路由器信号设备正常，如下图所示：



连接WiFi网络正常，浏览器测试WiFi上网正常，如下图所示：

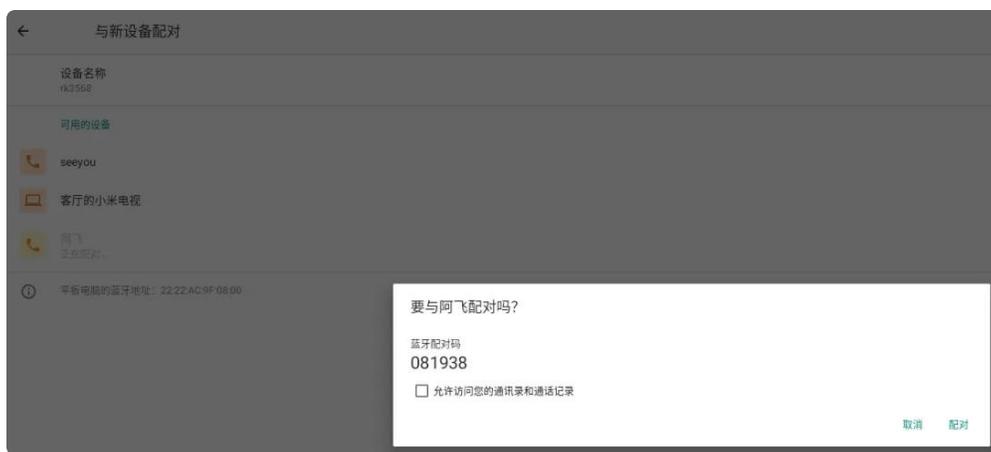


测试项目	要求	结果
WiFi	连接网络正常，连接网络后浏览器测试上网正常	✓

3.4 Bluetooth

【菜单】 -> 【设置】 -> 【已连接的设备】 -> 【与新设备配对】

即可扫描到附近的蓝牙设备，选择需要连接的设备即可根据配对信息进行连接，如下图所示：

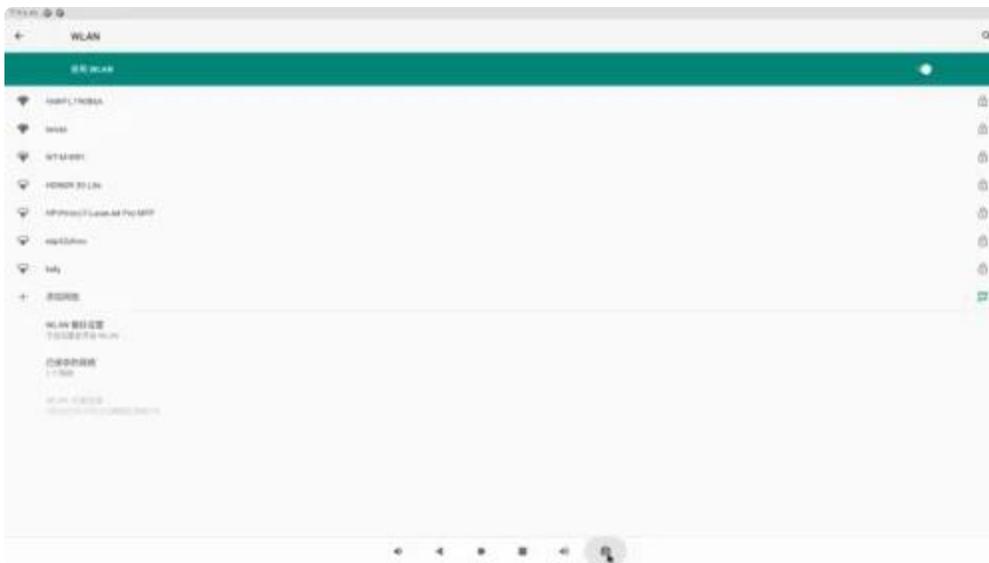


配对成功后主板即可通过蓝牙与手机相互传输文件

测试项目	要求	结果
蓝牙	配对可与连接设备收发文件	✓

3.5 天线接口

天线接口，如下图所示：



WiFi接上天线后，用WiFi魔盒扫描【industio5.8G】并查看 信号强度，记录接天线和不接天线的信号值，如下表所示：

测试项目	不接天线	接天线
4G天线	+CSQ:15,99	+CSQ:31,99
WiFi	-66db	-43db
WiFi天线二根	-73db	-43db