IDO-EVB1126B-V1

Linux使用手册

深圳触觉智能科技有限公司

www.industio.cn

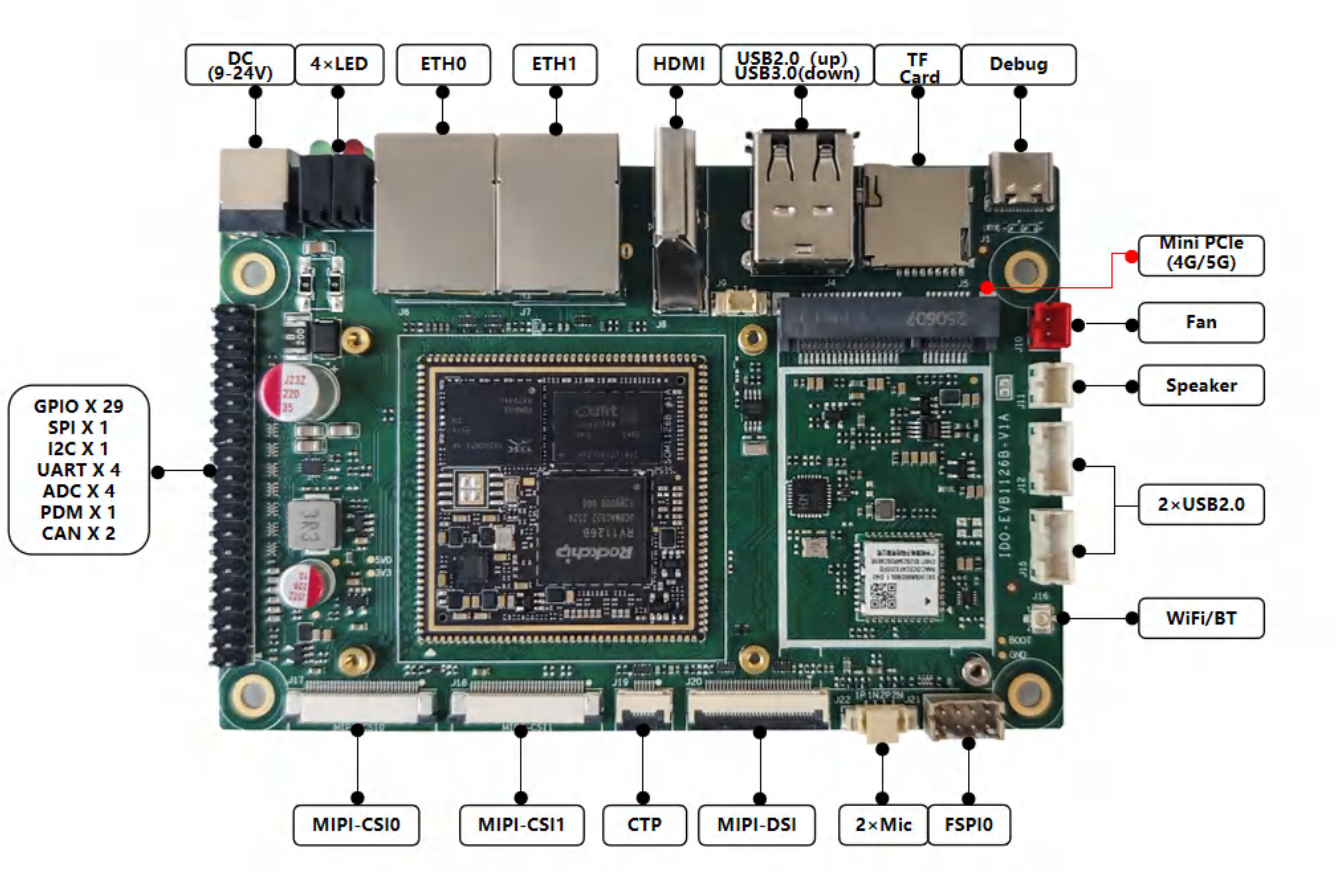
文档修订历史

版本	PCBA版本	修订内容	修订	审核	日期
V1.0	V1A	创建文档	LJS	IDO	2025/10/14
V1.1	V1B	以太网调整	LJS	IDO	2025/10/31

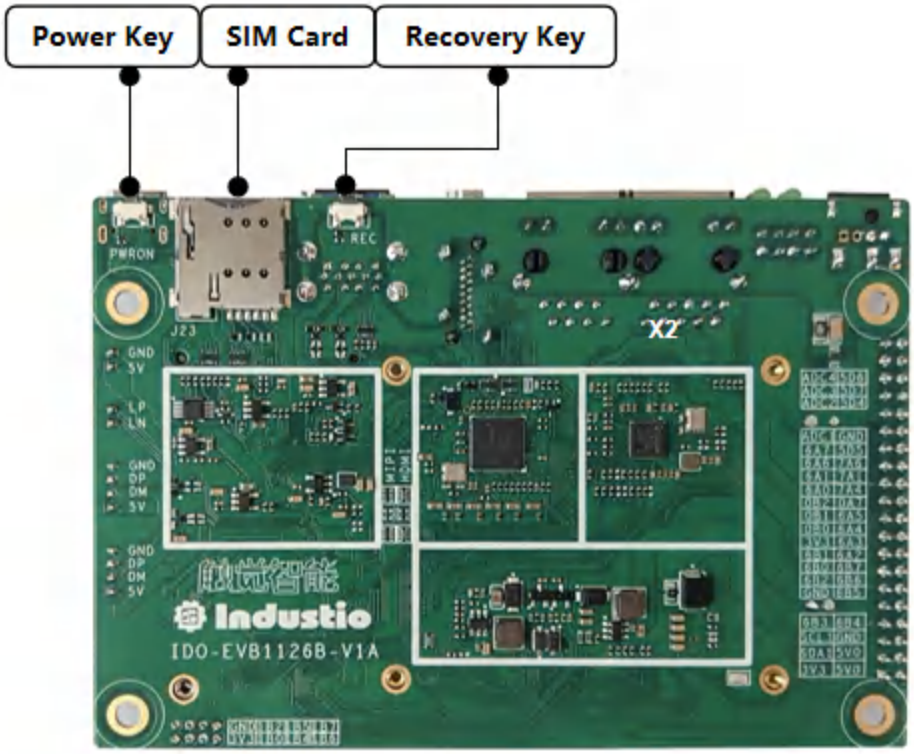
V1.2	V1C	补充PWRON说明	LJS	IDO	2026/01/14

1. 硬件资源概况

1.1. 主板照片



IDO-EVB1126B-V1A 正面实物图



IDO-EVB1126B-V1A 背面实物图

1.2. 硬件资源

序号	名称	描述
1	内核版本	Linux 6.1.141
2	系统版本	Buildroot2024/Debian12
3	内存	LPDDR4/4X（1/2/4G）
4	存储	Flash/eMMC（8/16/32GB）
5	供电	DC接口12V@2A
6	显示	HDMI-TX ×1 MIPI ×1 （二选一）
7	触摸	I2C-TP ×1

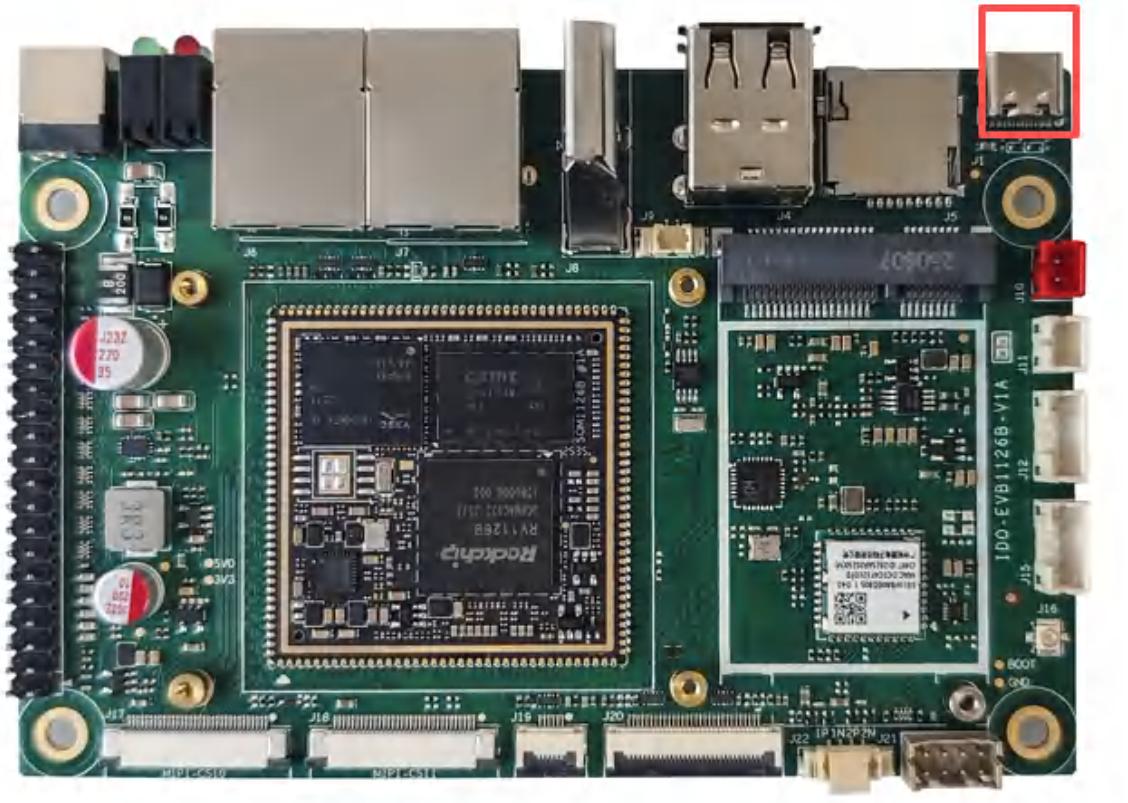
8	USB OTG	USB OTG Type-A ×1
9	USB HOST	USB2.0 HOST(Type-A) ×1 USB2.0 HOST(PH2.0) ×1
10	TF Card	SDIO3.0 TF Card ×1
11	以太网	千兆以太网 ×1 百兆以太网 x 1
12	WIFI/BT	AIC8800D80 ×1
13	扬声器	PH2.0-2P(4Ω5W) ×1
14	MIC	驻极体麦克风 ×1
15	Camera	IMX415 + IMX415
16	UART	TTL x 4 RS485 ×1 （转接板）
17	调试串口	TTL ×1
18	CAN口	CAN ×2
19	RTC	HYM8563 ×1
20	LED	系统指示灯 ×1
21	4G/5G	EC20 4G
22	按键	POWER ×1 REC ×1
23	Fan	5V PH2.0-2P ×1
24	GPIO	GPIO（预留） ×8
25	ADC	ADC（预留） ×4

2. IDO-EVB1126B-V1A-Buildroot系统

2.1. 功能及接口使用方法

2.2. 调试串口

调试串口位置J1，如下图所示：



使用USB Type-C数据线，连接PC端的USB接口。系统会识别到一个“USB-SERIAL CH340”端口设备。

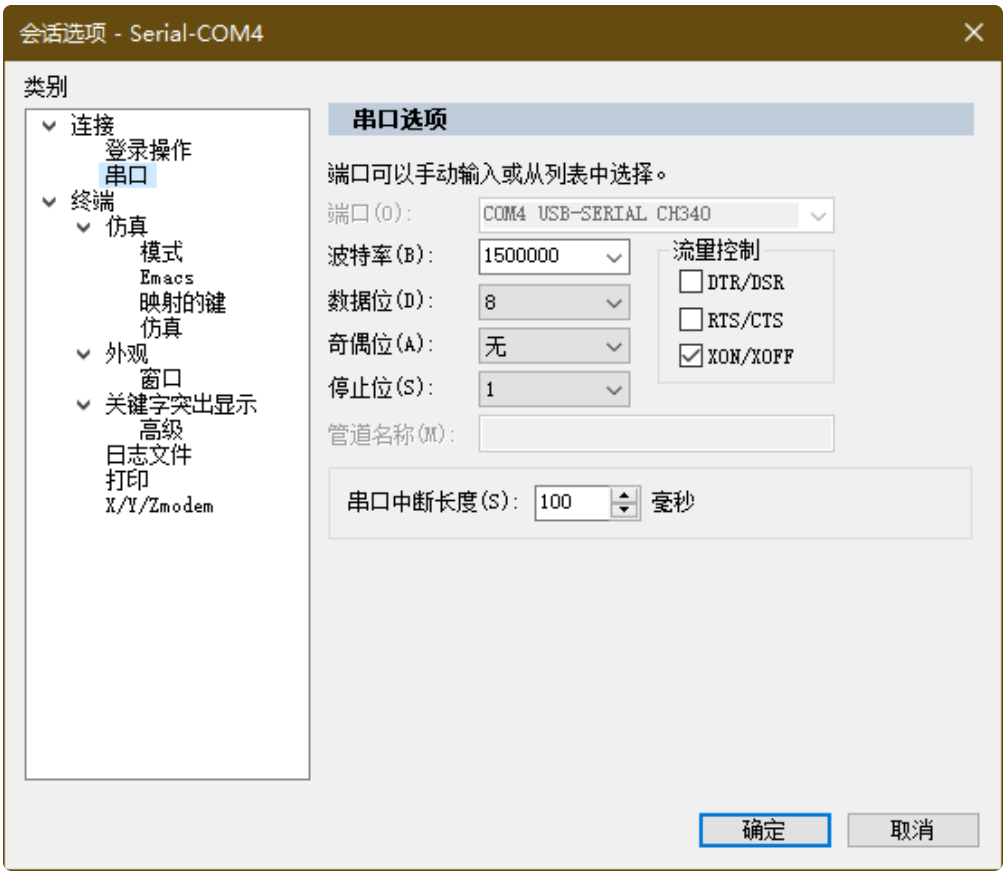
使用调试软件（MobaXterm、putty）等，设置参数如下：

波特率：1500000

数据位：8

奇偶校验位：无

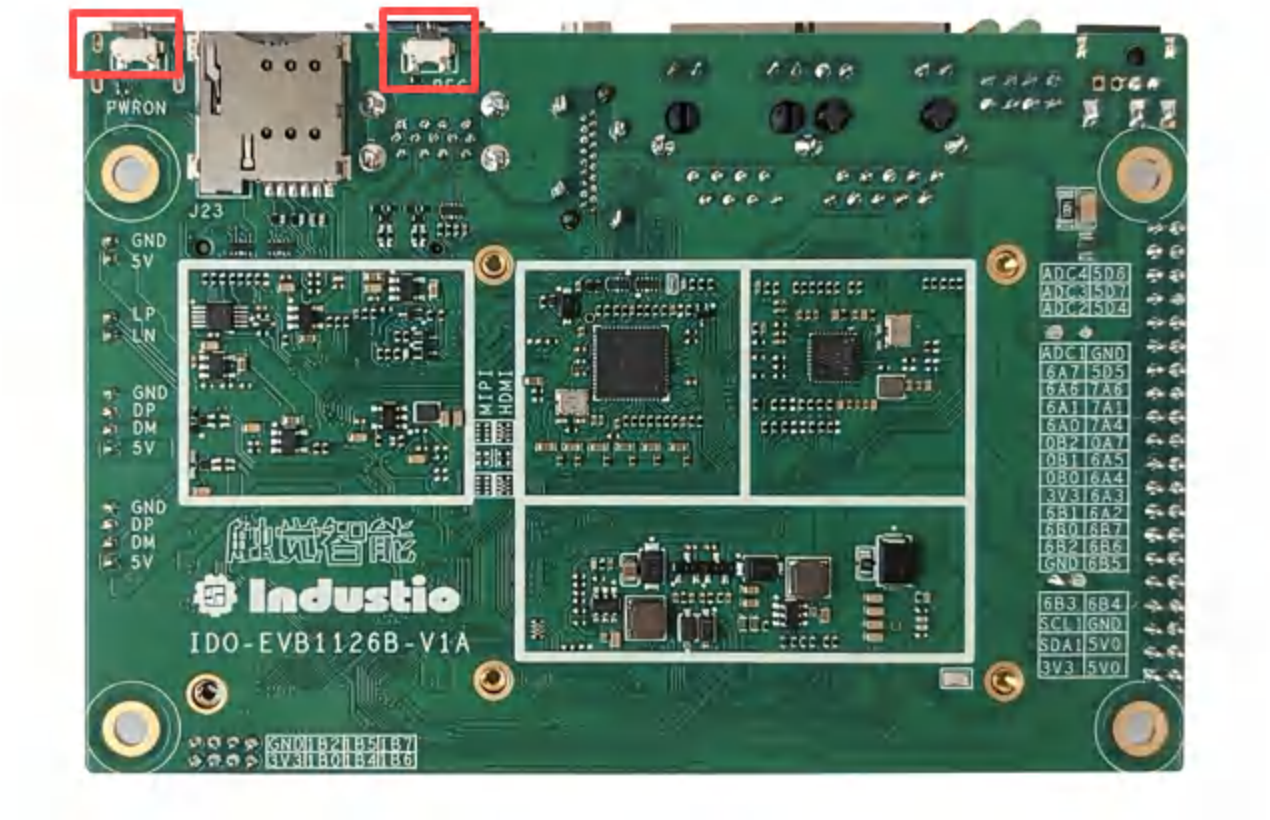
停止位：1



2.3. 按键

主板自带1个Recovery按键REC

Recovery按键，关机状态下，按住Recovery按键再给主板上电，主板会进入Loader模式。

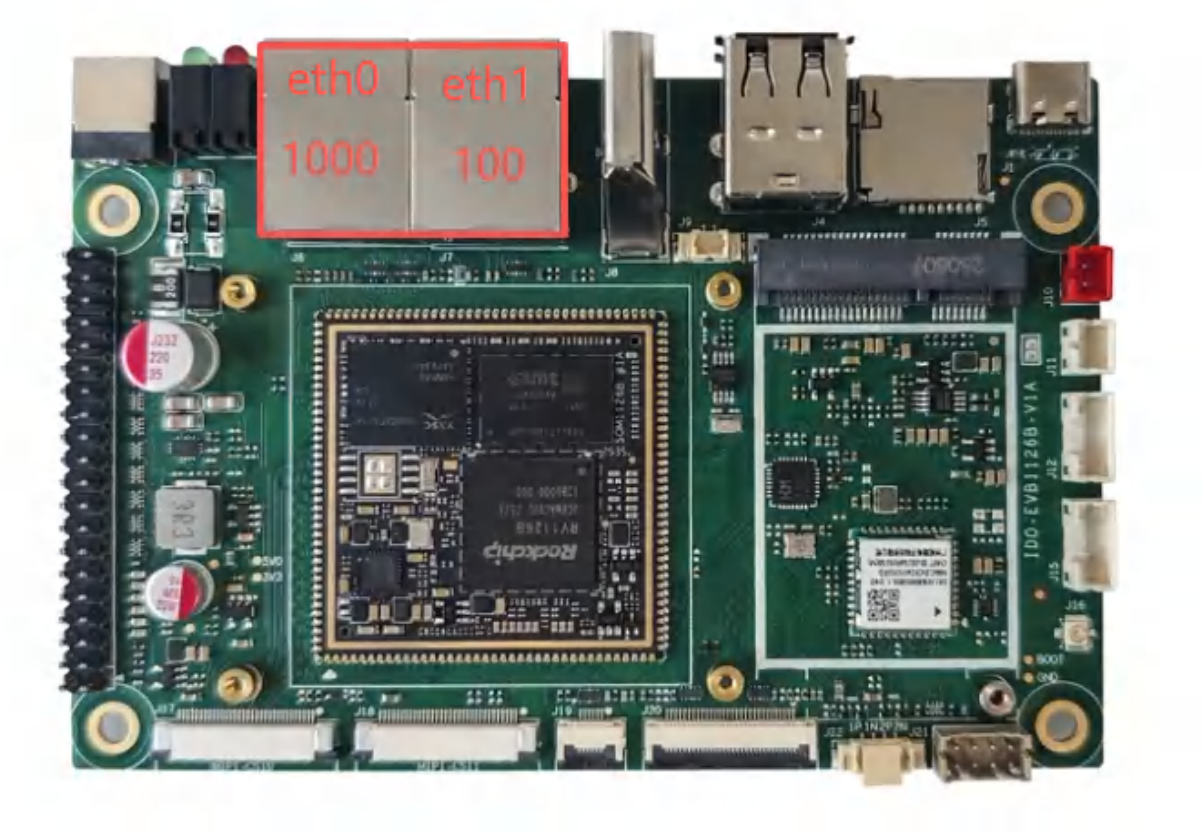


序号	定义	说明
1	PWRON_KEY	电源按键，用于开机/关机，休眠/唤醒；
2	RECOVERY_KEY	烧录按键，用于烧录

2.4. 网络

2.4.1. Ethernet

主板有 1 路千兆以太网，1路百兆以太网，如下图所示：




```Bash
# TCP
#server
iperf3 -s -i 1
#client
iperf3 -c server.ip   # 上行
iperf3 -c server.ip -R 下行

# UDP
#server
iperf3 -s -i 1
#client
iperf3 -c server.ip -u -b 1000M     # 上行
iperf3 -c server.ip -u -b 1000M  -R # 下行
```

可通过ifconfig指令设置临时静态IP，命令如下：

```Bash
$ ifconfig eth0 192.168.1.120
```

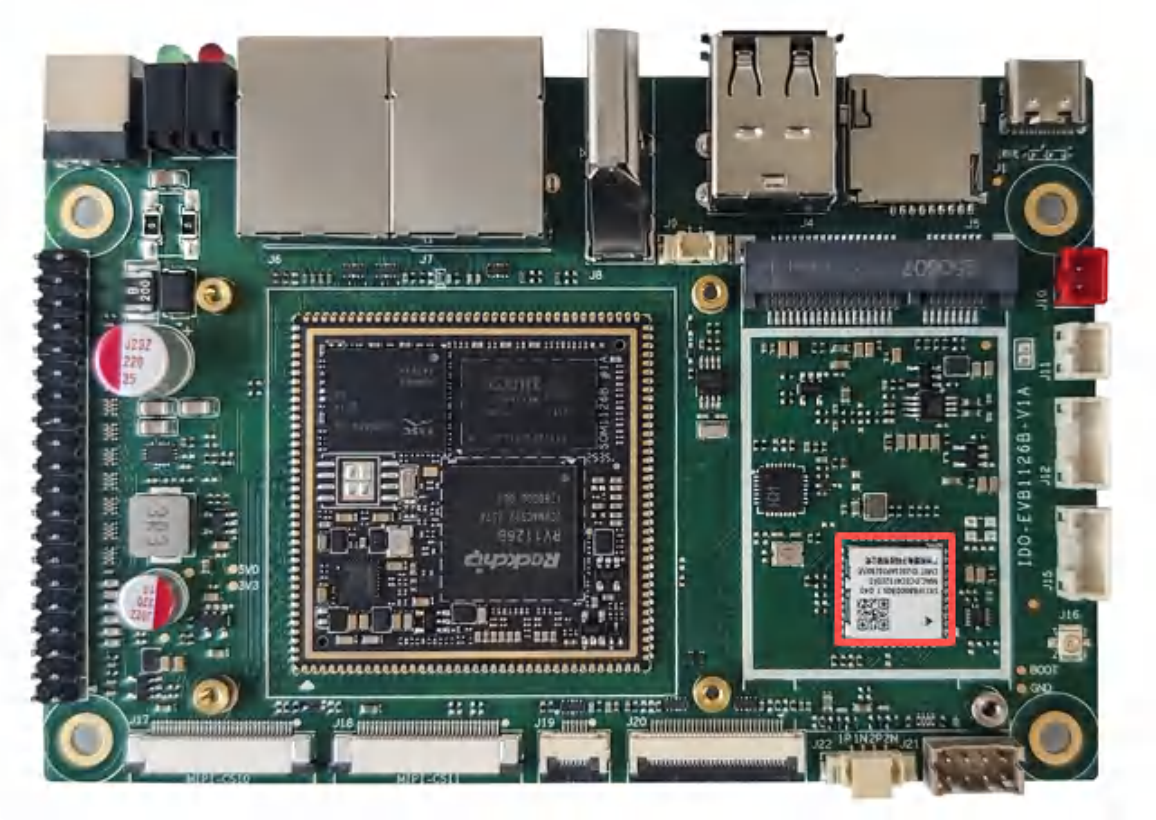
如果永久设置编辑文件 /etc/network/interfaces

```bash
root@rv1126b-buildroot:/#  vi /etc/network/interfaces
root@rv1126b-buildroot:/#  cat /etc/network/interfaces
# interface file auto-generated by buildroot

auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.3.123
netmask 255.255.255.0
gateway 192.168.3.1

source-directory /etc/network/interfaces.d
```

执行reboot重启系统后，可以看到我们设置的永久ip.

2.4.2. WIFI

使用WiFi/蓝牙时，需要连接天线以获得良好的信号，WiFi模块和天线座子J62，如下图所示：



```Bash
root@rv1126b-buildroot:/#  vim /etc/wpa_supplicant.conf
ctrl_interface=/var/run/wpa_supplicant
ap_scan=1

network={
        ssid="SSID"
        psk="PASSWORD"
        key_mgmt=WPA-PSK
}
```

注意：

ssid：账号

psk：密码

扫描附近WiFi：

```Bash
iw dev wlan0 scan | grep SSID
```

WiFi连接：

```Bash
wpa_supplicant -Dnl80211 -c /etc/wpa_supplicant.conf -i wlan0 &
```

查看连接结果：

```Bash
wpa_cli -i wlan0 -p /var/run/wpa_supplicant scan
```

2.4.3. Bluetooth

蓝牙节点：

```
 hciconfig -a
hci0: Type: Primary  Bus: UART
  BD Address: 4C:38:00:2B:C1:71  ACL MTU: 1021:8  SCO MTU: 64:1
  UP RUNNING
  RX bytes:878 acl:0 sco:0 events:62 errors:0
  TX bytes:3291 acl:0 sco:0 commands:62 errors:0
  Features: 0xbf 0xfe 0xcf 0xfe 0xdb 0xff 0x7b 0x87
  Packet type: DM1 DM3 DM5 DH1 DH3 DH5 HV1 HV2 HV3
  Link policy: RSWITCH SNIFF
  Link mode: PERIPHERAL ACCEPT
  Name: 'BlueZ 5.77'
  Class: 0x6c0414
  Service Classes: Rendering, Capturing, Audio, Telephony
  Device Class: Audio/Video, Loudspeaker
  HCI Version: 5.0 (0x9)  Revision: 0x50
  LMP Version: 5.0 (0x9)  Subversion: 0x6606
  Manufacturer: Broadcom Corporation (15)
```

注意：1. 当前SDK默认支持rkwifibt_app_test

2. bluetoothctl 和 rkwifibt_app_test 只能二选一使用

2.4.3.1. rkwifibt-app 测试（默认不支持）

rkwifibt_app_test 连接测试请参考文档 《IDO-EVB1126B-V1 蓝牙开发案例》

2.4.3.2. bluetoothctl 测试（默认支持）

由于命令 bluetoothctl 和 命令rkwifibt_app_test 只能二选一使用，当需要使用 bluetoothctl 命令的时候，

我们直接将这个补丁重命名备份，然后删除buildroot/output/rockchip_evb1126b-v1a/build/bluez5_utils-* , 最后重新编译buildroot 再烧录。

当需要使用rkwifibt_app_test 命令的时候，再将0001-bluez-modified-only-for-rockchip.patch 放到buildroot/package/bluez5_utils/ 目录下即可。

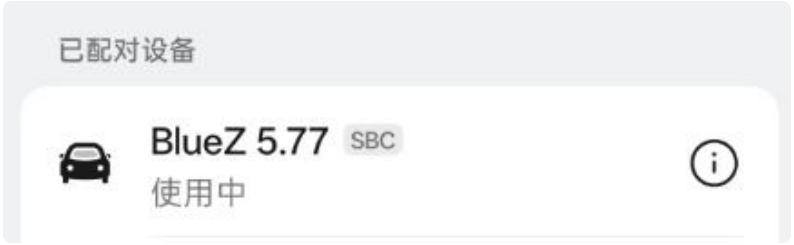
```Bash
$ mv buildroot/package/bluez5_utils/0001-bluez-modified-only-for-rockchip.patch.bak buildroot/package/bluez5_utils/0001-bluez-modified-only-for-rockchip.patch
$ rm -rf buildroot/output/rockchip_evb1126b-v1a/build/bluez5_utils-*
$ ./build.sh buildroot
```

使用bluetoothctl命令 测试蓝牙功能

```Bash

 bluetoothctl
 hci0 new_settings: powered bondable ssp br/edr le secure-conn
 Agent registered
[CHG] Controller 4C:38:00:2B:C1:71 Pairable: yes
[bluetooth]#
 #打开蓝牙
[bluetooth]# power on
 #扫描蓝牙
[bluetooth]# scan on
 # # 开启广播
[bluetooth]# discoverable on
 # 允许配对
[bluetooth]# pairable on
 # 使用 NoInputNoOutput (Just Works)
[bluetooth]# agent NoInputNoOutput
[bluetooth]# default-agent
 #查看蓝牙设备
[bluetooth]# devices
 #信任蓝牙设备
[bluetooth]# trust 7C:C1:80:09:DD:6C
 #蓝牙配对
[bluetooth]# pair 7C:C1:80:09:DD:6C
 #连接蓝牙
[bluetooth]# connect 7C:C1:80:09:DD:6C
```

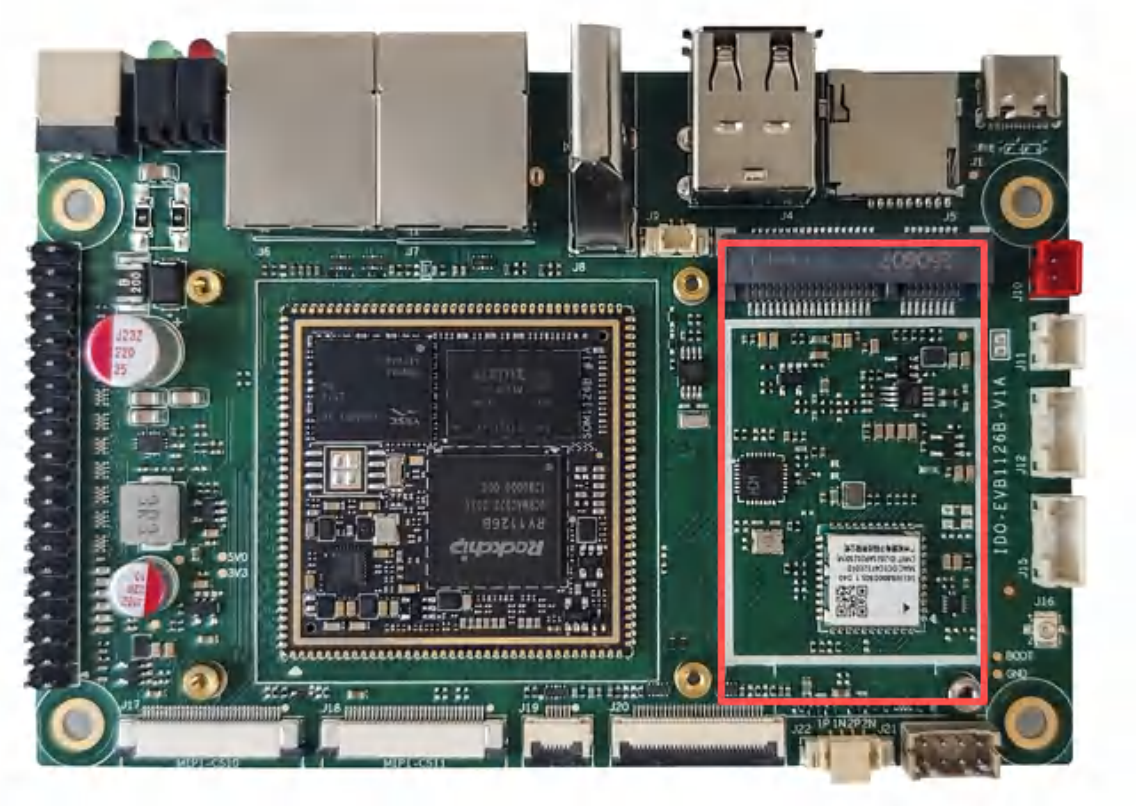
注意：如果出现无法配对的情况，请执行quit退出bluetoothctl ，重新启动蓝牙服务，再次执行以上连接命令

```Bash
[bluetooth]# quit
root@rv1126b-buildroot:/# /etc/init.d/S40bluetoothd restart
```

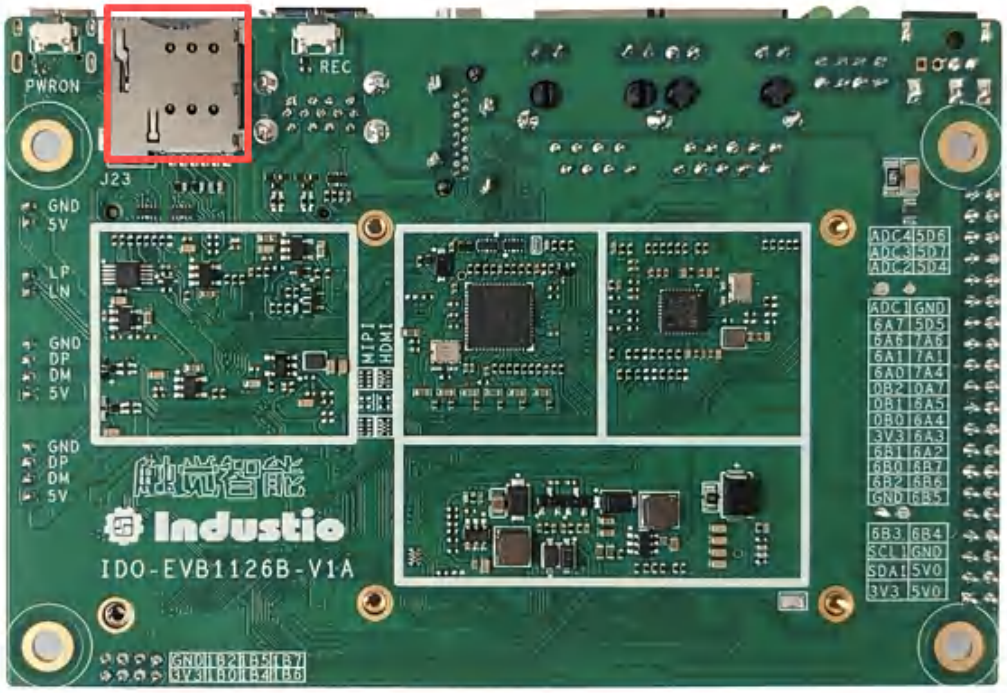


2.4.4. 4G/5G

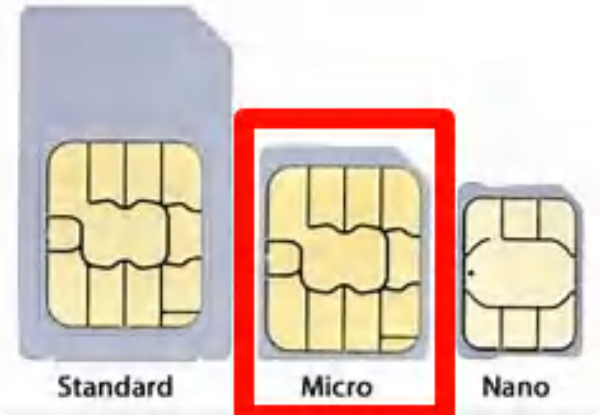
主板内置Mini PCIe 座子扩展 4G/5G模块，4G通信模块适配移远EC20、EC25等通用模组。如下图所示：



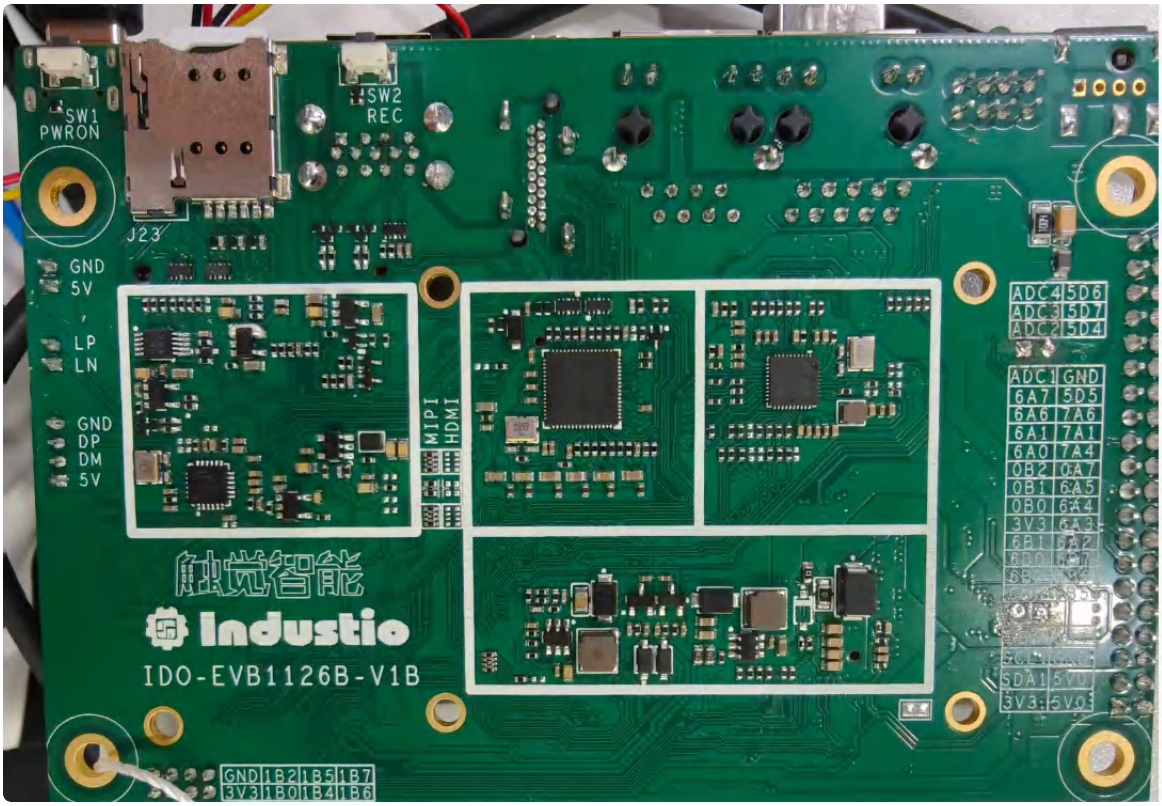
模块需要接上天线以确保获得更好的信号质量，SIM卡安装位J23位于主板背面，如下图所示：



使用Micro尺寸SIM卡，如下图所示：



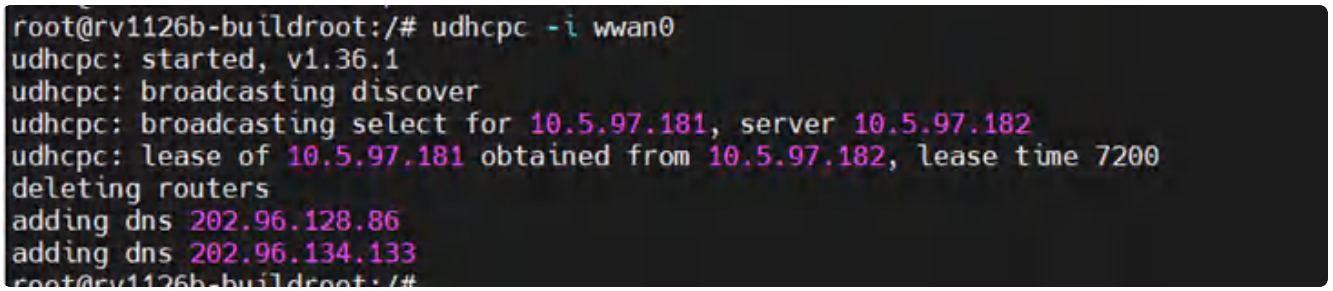
主板朝上时，SIM卡触点朝上，缺口朝外安装，如下图所示：



拨号工具使用quectel-CM

```bash
root@rv1126b-buildroot:/# quectel-CM &
wwan0     Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet addr:10.230.31.108  P-t-P:10.230.31.108  Mask:255.255.255.248
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:5 errors:0 dropped:0 overruns:0 frame:0
          TX packets:29 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1420 (1.3 KiB)  TX bytes:5702 (5.5 KiB)

root@rv1126b-buildroot:/# ping www.baidu.com -I wwan0
PING www.baidu.com (157.148.69.186) from 10.230.31.108 wwan0: 56(84) bytes of data.
64 bytes from 157.148.69.186: icmp_seq=1 ttl=52 time=29.2 ms
64 bytes from 157.148.69.186: icmp_seq=2 ttl=52 time=37.1 ms
64 bytes from 157.148.69.186: icmp_seq=3 ttl=52 time=35.4 ms
64 bytes from 157.148.69.186: icmp_seq=4 ttl=52 time=33.6 ms
64 bytes from 157.148.69.186: icmp_seq=5 ttl=52 time=31.7 ms
64 bytes from 157.148.69.186: icmp_seq=6 ttl=52 time=29.6 ms
64 bytes from 157.148.69.186: icmp_seq=7 ttl=52 time=37.7 ms
64 bytes from 157.148.69.186: icmp_seq=8 ttl=52 time=36.3 ms
^C
--- www.baidu.com ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 26063ms
rtt min/avg/max/mdev = 29.170/33.826/37.680/3.133 ms
```

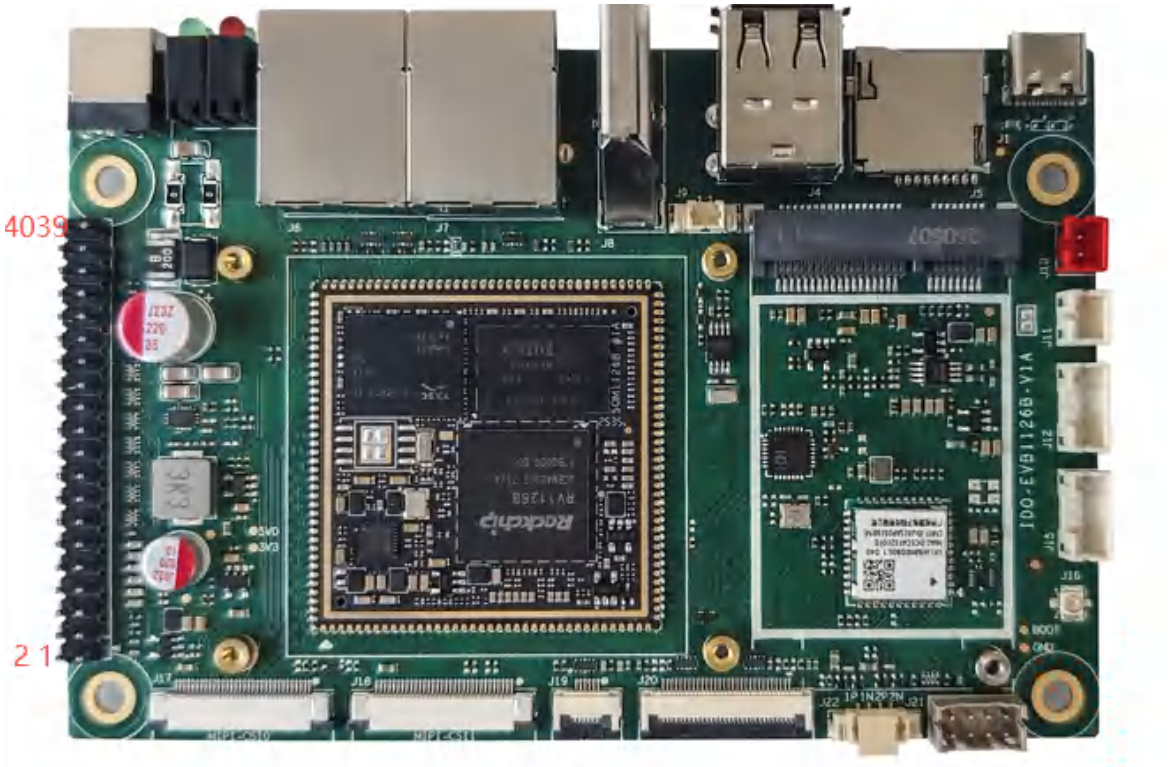
如果出现ping 不通外网，请查看是否是DNS 没有解析

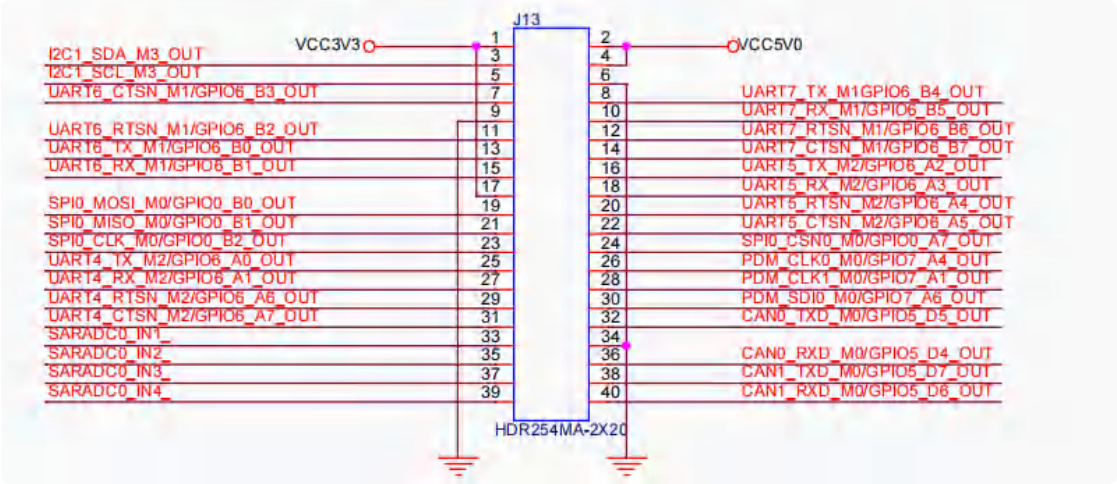



```bash
# 添加wwan0节点的dns
$ vi /etc/resolv.conf
nameserver 202.96.128.86
```

2.5. UART

IDO-EVB1126B-V1主板一共引出4路UART串口（不含调试串口），如下图所示：





UART串口设备节点列表如下：

序号	默认电平类型	设备节点
J13-25,J13-27	TTL	/dev/ttyS4
J3-16,J3-18	TTL	/dev/ttyS5
J3-13,J3-15	TTL	/dev/ttyS6
J3-8,J3-10	TTL	/dev/ttyS7

使用microcom进行串口测试：

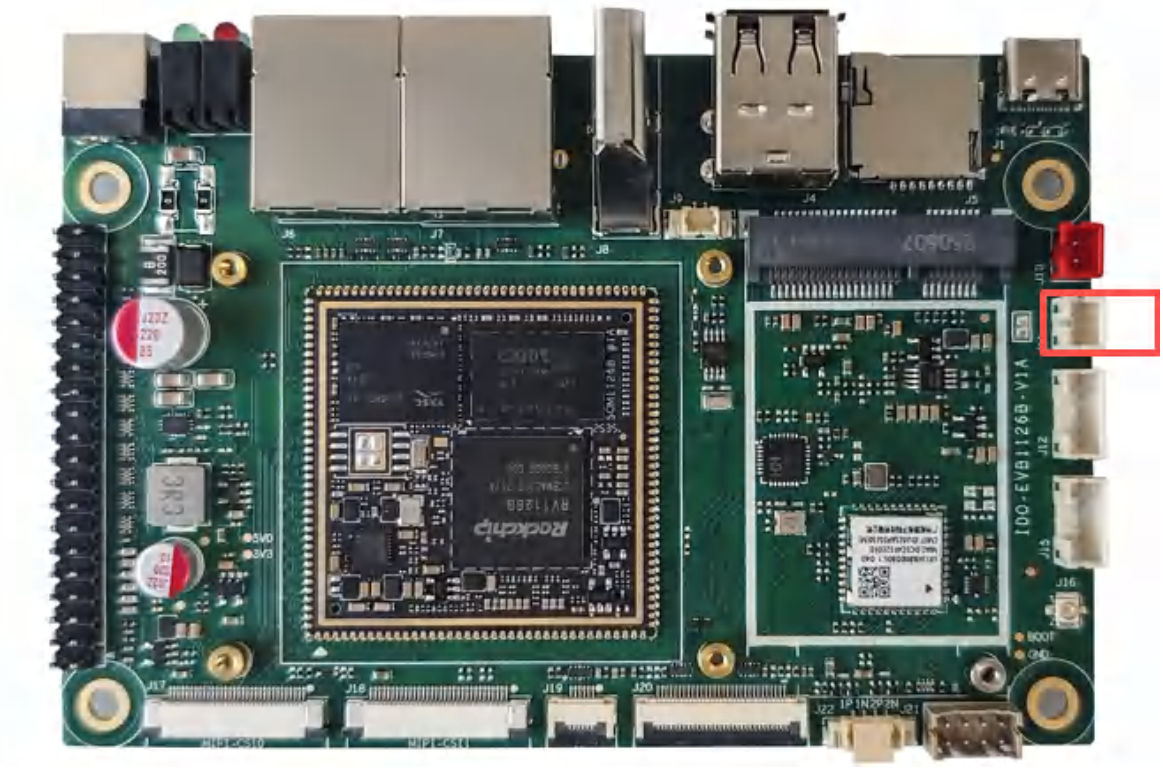
```Bash
# RX和TX短接，执行下面命令后，键盘输入，字符即时回显：
root@rv1126b-buildroot:/#  microcom /dev/ttyS7 -s 115200
```

注意：执行 Ctrl x 退出程序； RS485 需要使用转接板测试

2.6. Audio

2.6.1. Speaker

喇叭接口为J11，如下图所示：



声卡节点：

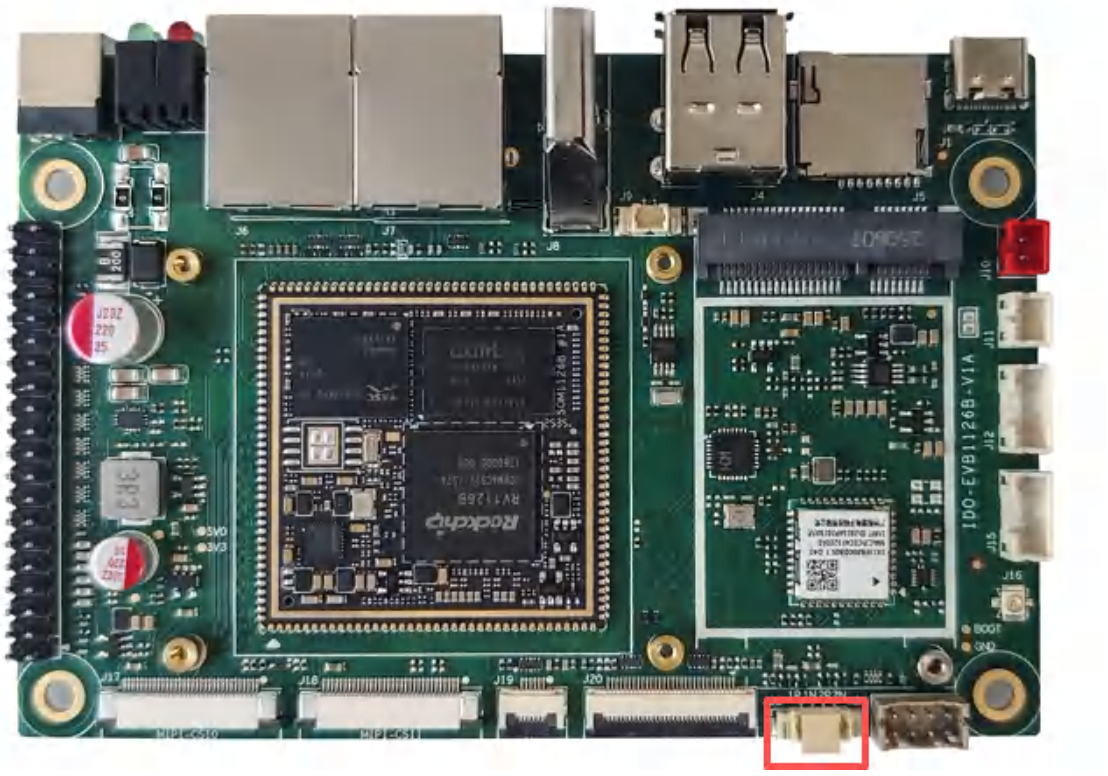
```Bash
root@rv1126b-buildroot:/# aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: rockchiprv1126b [rockchip,rv1126b-acodec], device 0: dailink-multicodecs multicodec-0 [dailink-multicodecs multicodec-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

音频播放：

```Bash
#音频播放到声卡
root@rv1126b-buildroot:/# aplay -D hw:0,0 test.wav
```

2.6.2. MIC

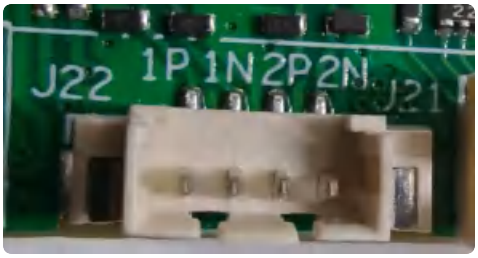
MX1.25T-2P麦克风接口J52，如下图所示：



注意：录出来音频文件需要导出到PC 上播放

测试命令：

```bash
# 一共两路mic ，mic0 录音命令
arecord -D hw:0,0 -f S16_LE -r 16000 -c 4 -t wav test_mic0.wav
# mic1 录音命令
arecord -D hw:0,0 -f S16_LE -r 16000 -c 4 -t  wav test_mic1.wav
```

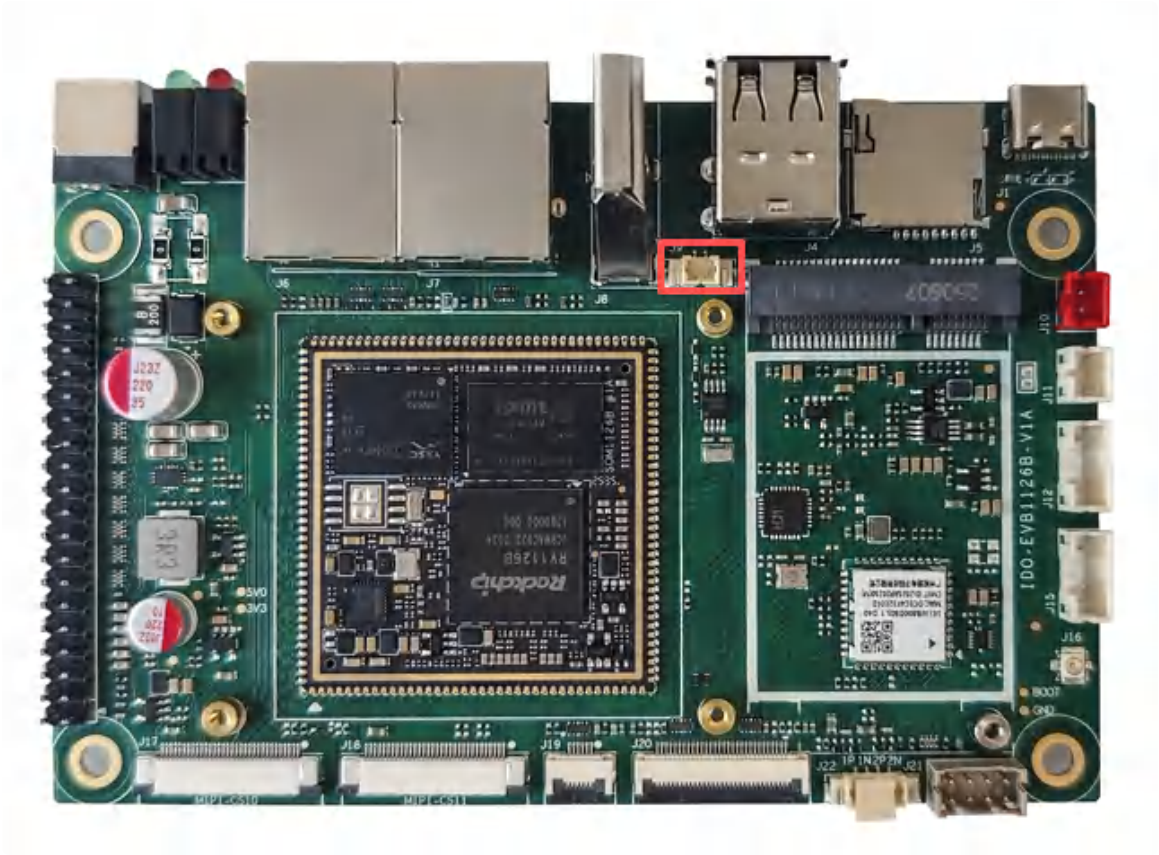


注意：如图上所示，2P,2N 是可以录制2line

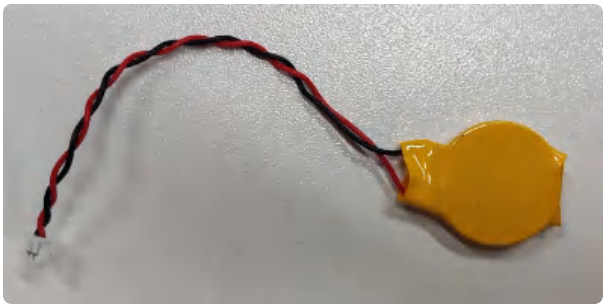
```bash
# 录音
arecord -D hw:0,0 -f S16_LE -r 16000 -c 2 -t wav test_mic.wav
# 播放
aplay -D hw:0,0  test_mic.wav
```

2.7. RTC

MX1.25T-2P RTC电池座J22，如下图所示：



连接3V 纽扣电池，RTC电池参考如下



系统默认使用HYM8563作为系统时钟，设备节点：/dev/rtc0，时间设置方法：

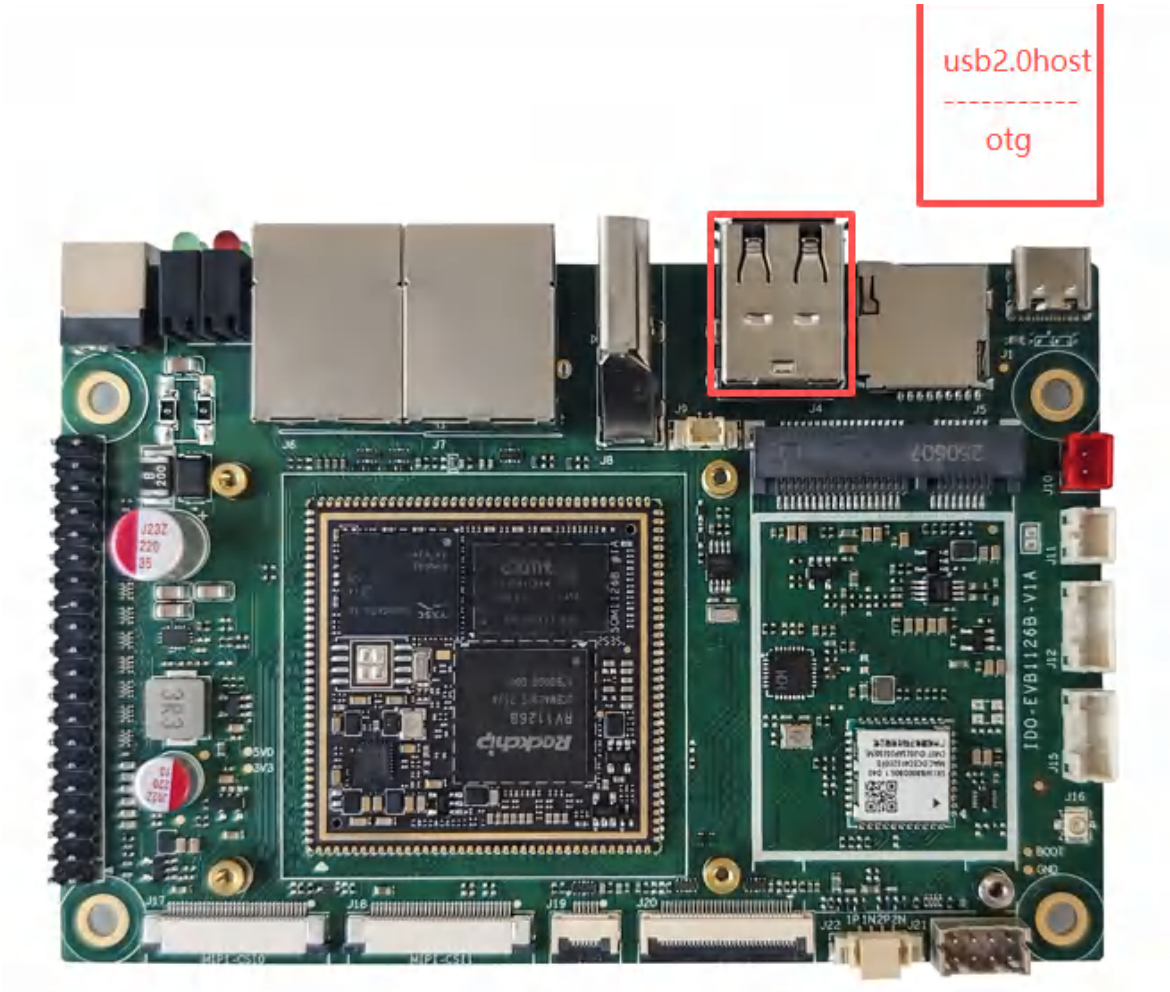
```Bash
#
root@rv1126b-buildroot:/# hwclock -w     # 向rtc写时间
root@rv1126b-buildroot:/# hwclock -r     # 向rtc读时间
```

2.8. USB

主板支持 1 个USB3-OTG，1 个 USB Type-A ,1 个USB2.0PH2.0-4P 接口，USB对外总供电应小于4A。

2.8.1. OTG

TypeA接口（USB3.0 OTG）J4，如下图所示：



EVB1126B 已实现上电启动自动切换OTG

1. 上电启动前接入OTG 通过双公头usb连接 PC ， 那么otg为device模式
2. 上电启动前接入U盘等usb设备，启动进入系统后otg为host模式

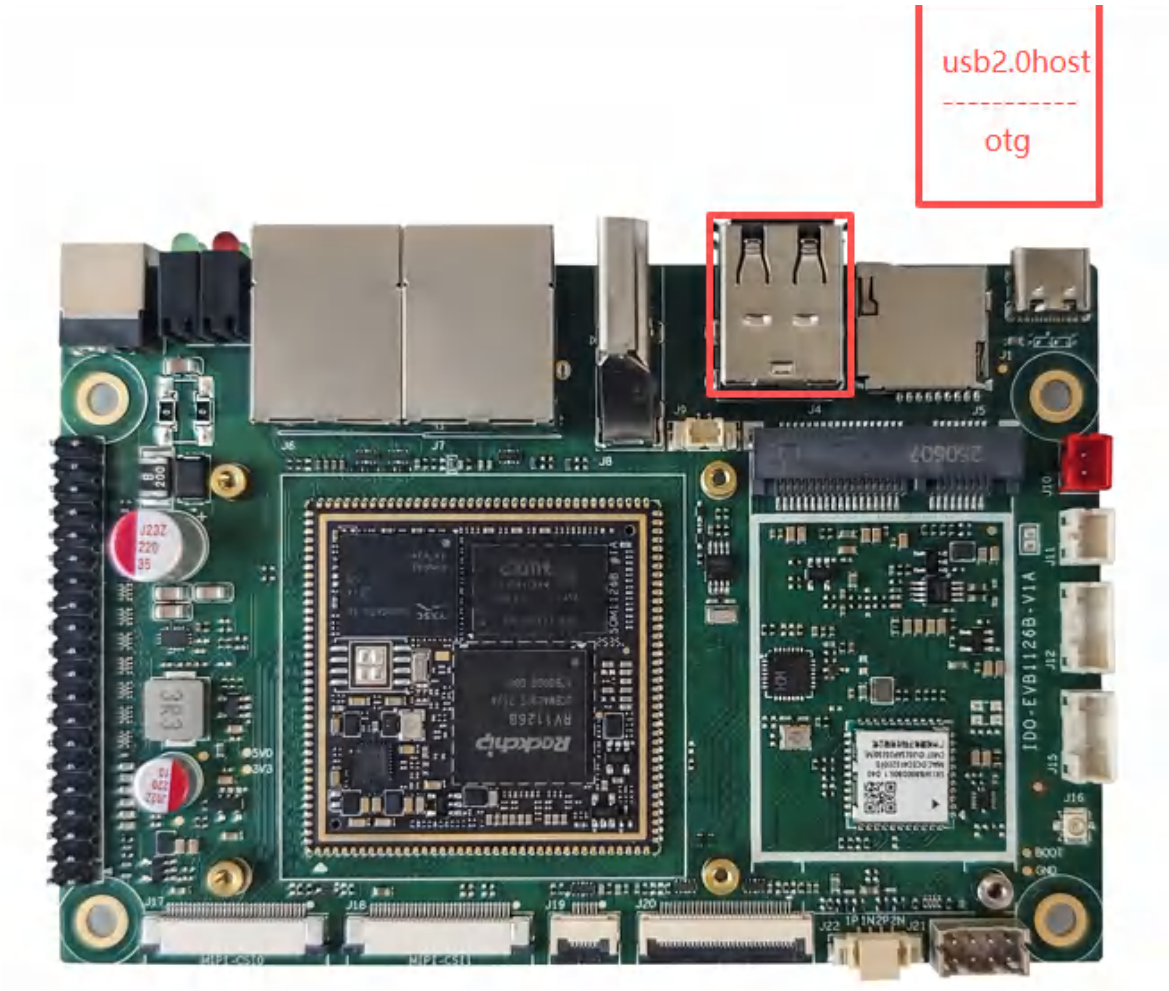
系统下otg切换命令：

命令	模式
gpio-otg HOST	HOST 模式
gpio-otg DEVICE	DEVICE 模式

```Bash
# host模式
root@rv1126b-buildroot:/# gpio-otg HOST
# device模式
root@rv1126b-buildroot:/# gpio-otg DEVICE
```

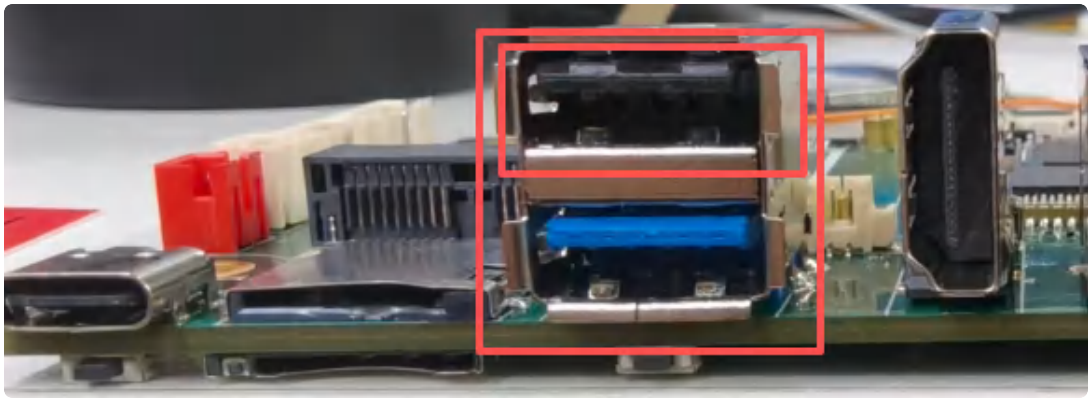
2.8.2. USB3.0

RV1126B 只有一个OTG3.0，



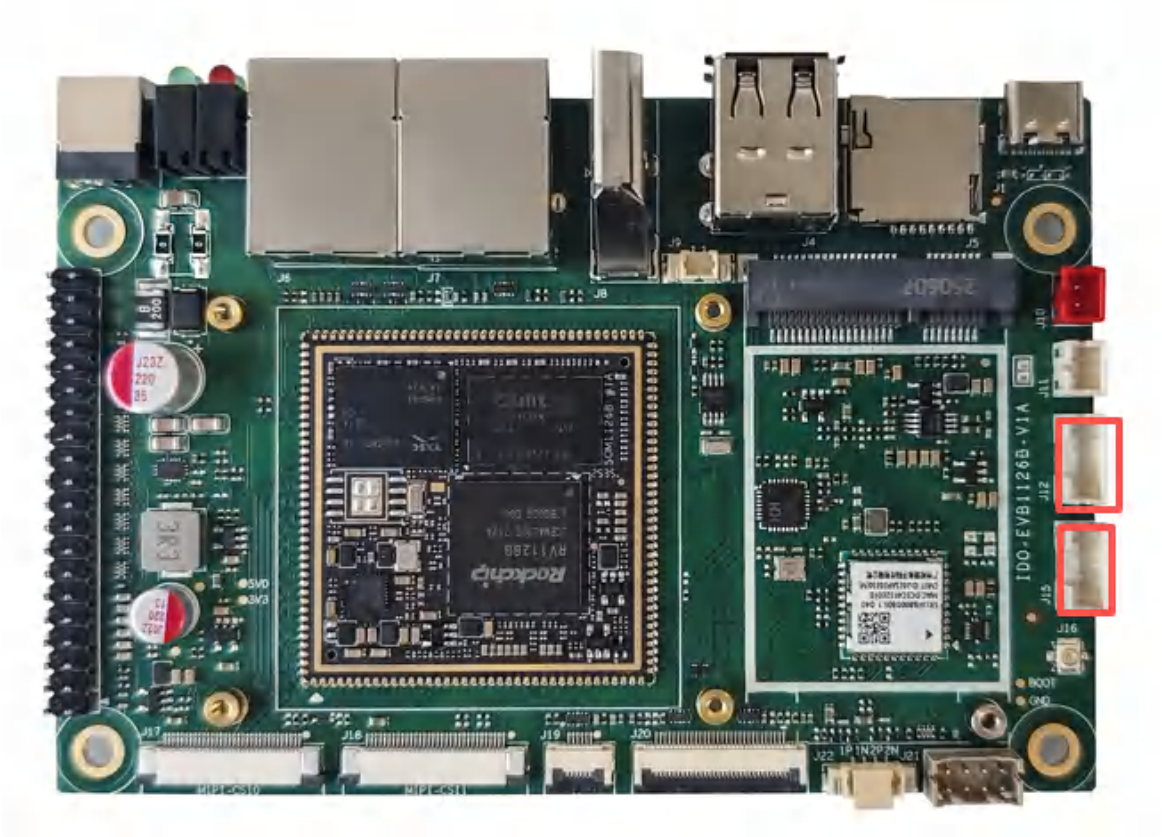
2.8.3. USB2.0-A

主板配置了1路USB2.0-A接口，如下图所示：



2.8.4. USB2.0-PH

主板配置了1路USB2.0 PH2.0-4P接口，USB接口均提供5V@1A的驱动能力，如下图所示：

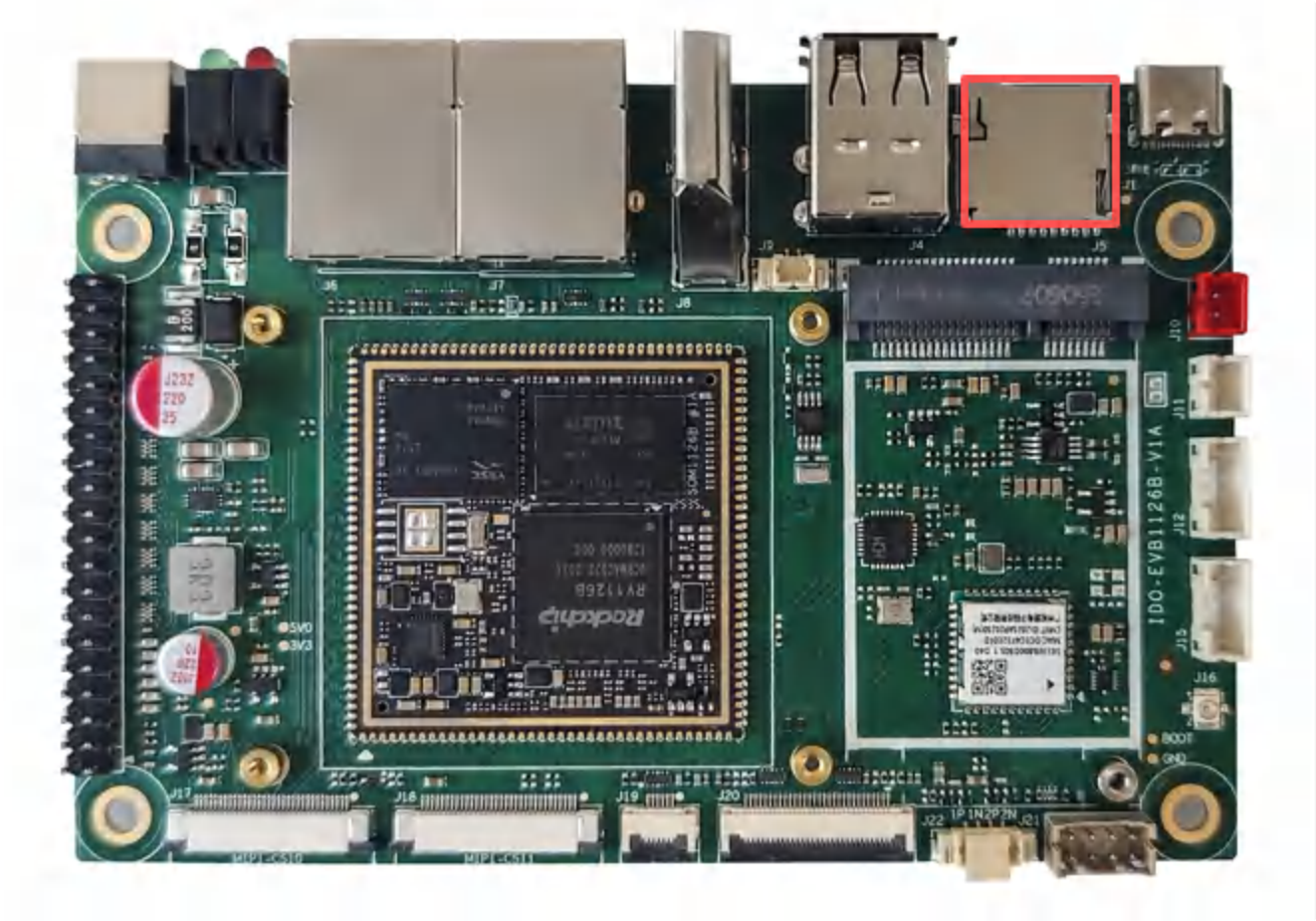


	J12
控制节点	/sys/class/leds/usb_vbus3_en/brightness

```bash
echo 1 > /sys/class/leds/usb_vbus3_en/brightness # 打开
echo 0 > /sys/class/leds/usb_vbus3_en/brightness # 关闭
```

2.9. TF Card

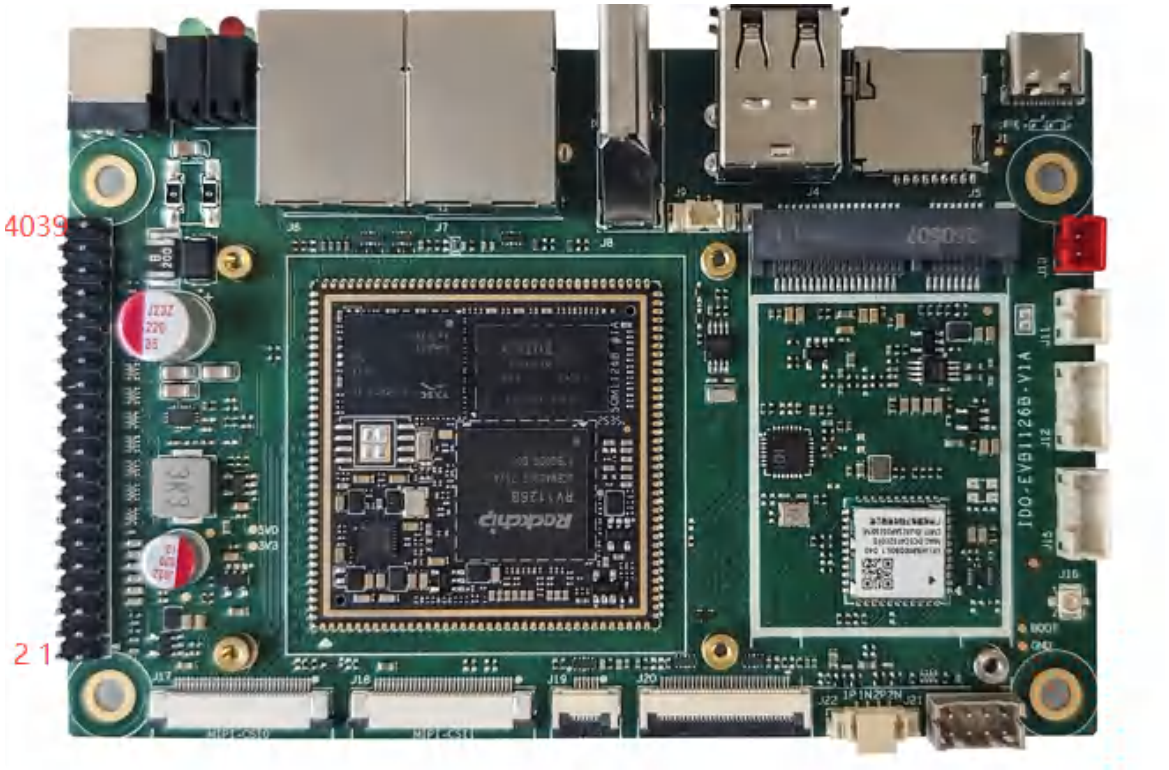
TF卡座支持SDIO3.0， 支持高速SD卡，接口位于主板背面J43，如下图所示：

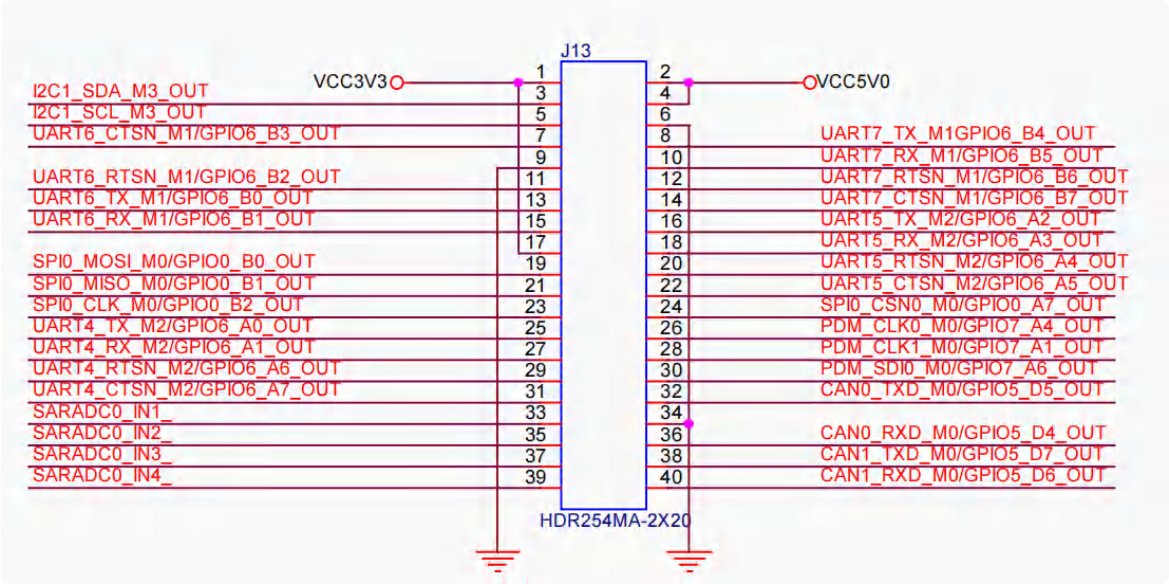


```
root@rv1126b-buildroot:/# fdisk -l
Found valid GPT with protective MBR; using GPT
...
Device       Boot StartCHS    EndCHS        StartLBA     EndLBA    Sectors  Size Id Type
/dev/mmcblk1p1    0,2,10      239,254,63         135    3858623    3858489 1884M  e Win95 FAT16 (LBA)

root@rv1126b-buildroot:/# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        56G  994M   52G   2% /
devtmpfs        1.9G  8.0K  1.9G   1% /dev
tmpfs           2.0G  140K  2.0G   1% /tmp
tmpfs           2.0G  468K  2.0G   1% /run
tmpfs           2.0G  244K  2.0G   1% /var/log
tmpfs           2.0G     0  2.0G   0% /dev/shm
/dev/mmcblk0p7  124M   12M  109M  10% /oem
/dev/mmcblk0p6  2.0G  616K  2.0G   1% /userdata
/dev/mmcblk1p1  32G   2G   30G     1% /mnt/sdcard
```

2.10. CAN

主板共有2路CAN接口如下图所示:



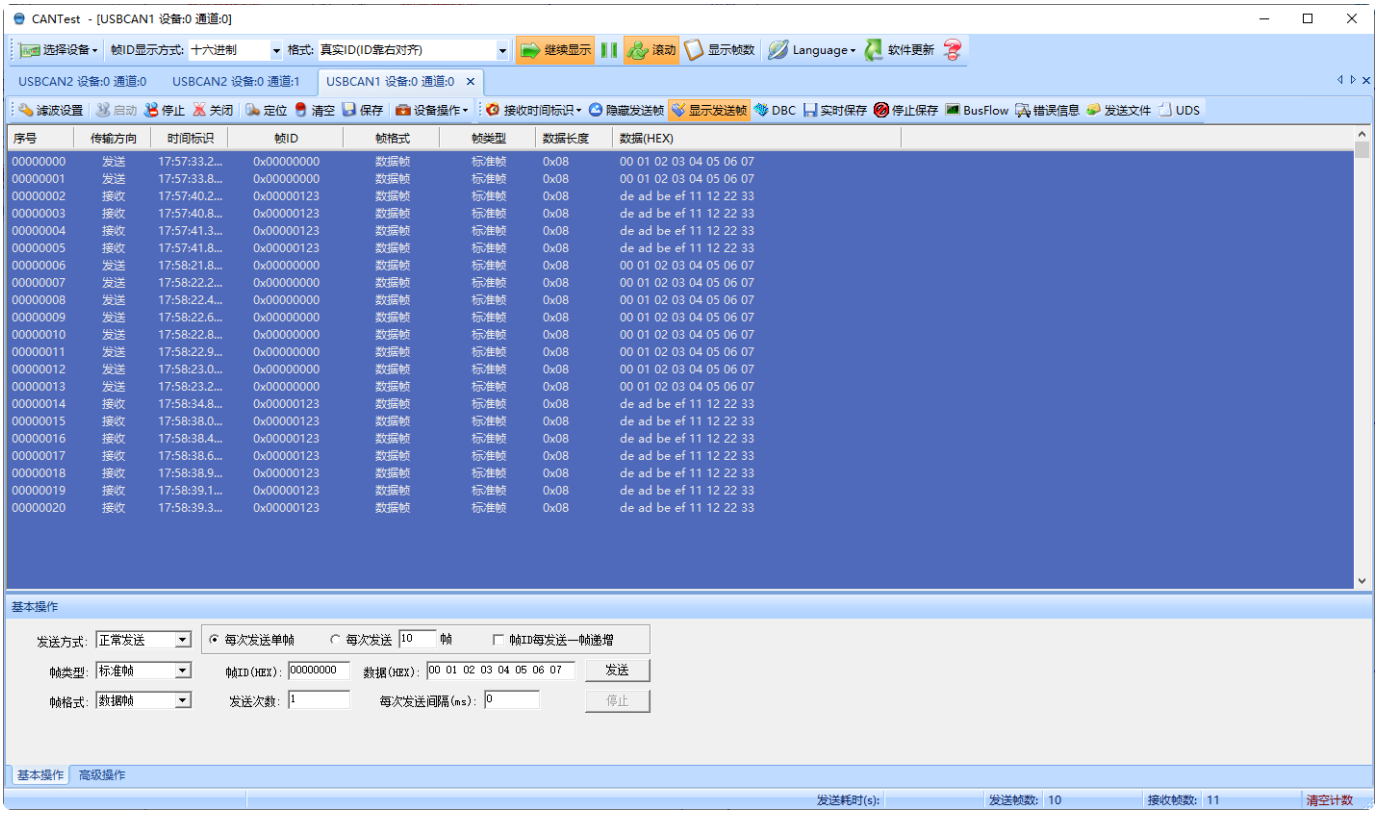


CAN	接线
can0	J13-32 ,J13-34
can1	J13-38 ,J13-40

分别对应系统节点名称为 can0、can1。使用can盒测试：

1. 设置CAN参数

```c
# can0节点发送数据
ip link set can0 down
ip link set can0 type can bitrate 1000000
ip link set can0 up
cansend can0 123#DEADBEEF
cansend can0 123#DEADBEEF11122233

# can0接收数据
ip link set can0 down
ip link set can0 type can bitrate 1000000
ip link set can0 up
candump can1

 # can1 接收数据
ip link set can1 down
ip link set can1 type can bitrate 1000000
ip link set can1 up
```



```
root@rv1126b-buildroot:/# candump can0
  can0  000   [8]  00 01 02 03 04 05 06 07
  can0  000   [8]  00 01 02 03 04 05 06 07
  can0  000   [8]  00 01 02 03 04 05 06 07
^Croot@rv1126b-buildroot:/# ^C
root@rv1126b-buildroot:/# ^C
root@rv1126b-buildroot:/# ^C
root@rv1126b-buildroot:/# cansend can0 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can0 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can0 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can0 123#DEADBEEF11122233
```

```
root@rv1126b-buildroot:/# candump can1
  can1  000   [8]  00 01 02 03 04 05 06 07
  can1  000   [8]  00 01 02 03 04 05 06 07
  can1  000   [8]  00 01 02 03 04 05 06 07
  can1  000   [8]  00 01 02 03 04 05 06 07
  can1  000   [8]  00 01 02 03 04 05 06 07
  can1  000   [8]  00 01 02 03 04 05 06 07
  can1  000   [8]  00 01 02 03 04 05 06 07
  can1  000   [8]  00 01 02 03 04 05 06 07
^Croot@rv1126b-buildroot:/# ^C
root@rv1126b-buildroot:/# ^C
root@rv1126b-buildroot:/# ^C
root@rv1126b-buildroot:/# cansend can0 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can1 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can1 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can1 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can1 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can1 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can1 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can1 123#DEADBEEF11122233
```

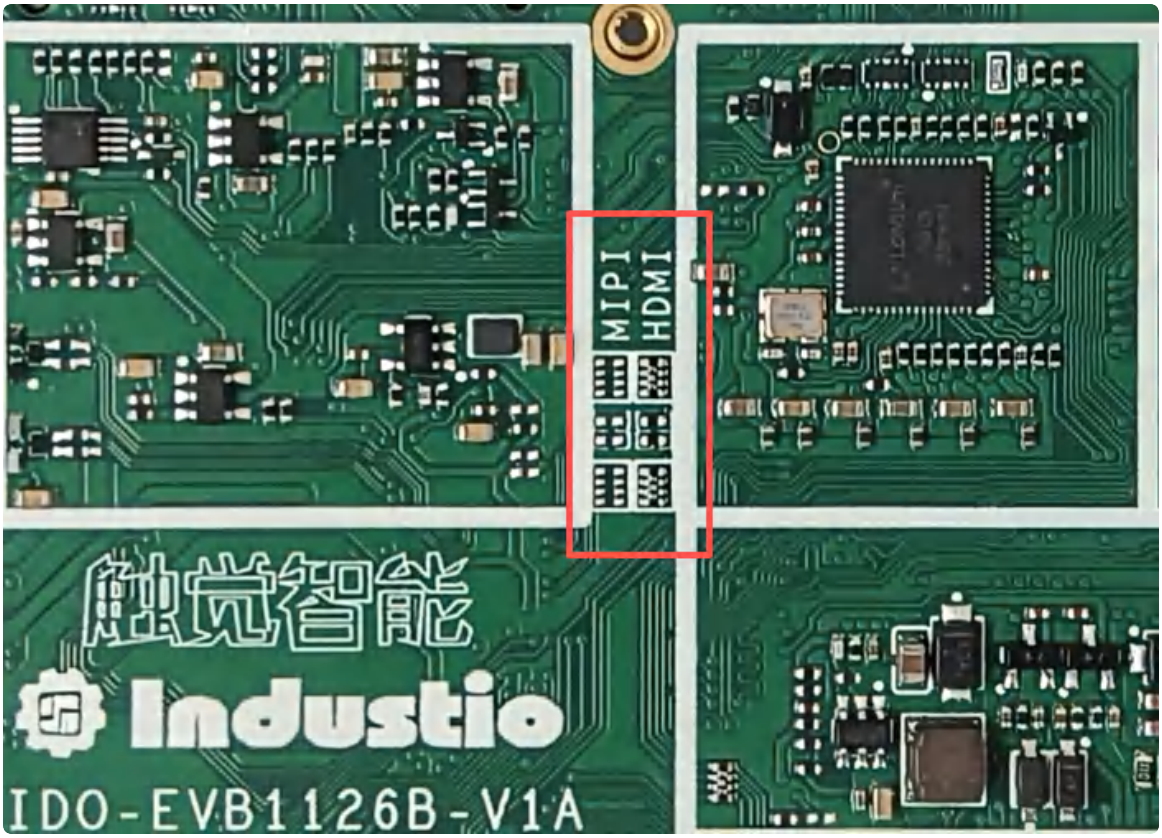
注意： 测试需要使用转接板。

2.11. LCD显示

2.11.1. HDMI-TX

注意：HDMI-TX 是由 MIPI-TX 接口 + 信号转换芯片 ， 二选一，硬件默认配置为 HDMI-TX。

HDMI-TX硬件确认，如下图所示：



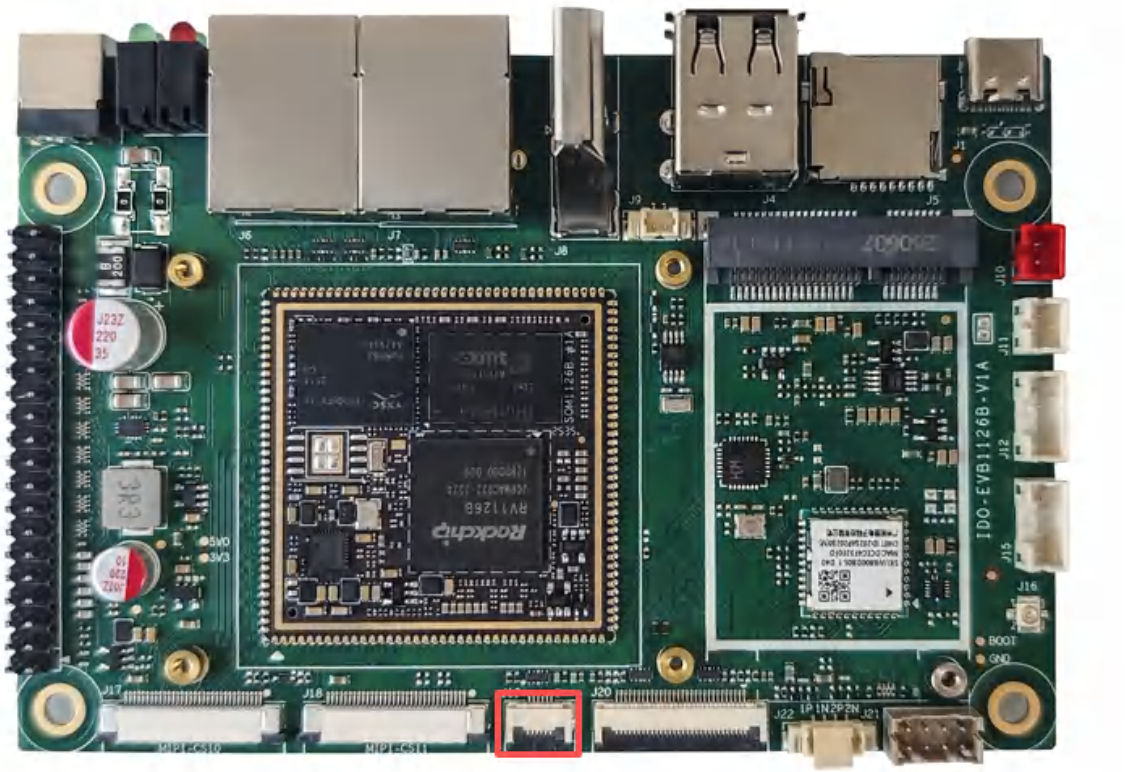
2.11.2. MIPI–TX

注意：MIPI–TX 与 HDMI 为复用关系，二选一，硬件默认配置为 HDMI。

要使用mipi的话，需要将下图中HDMI 列电阻移动至MIPI 列

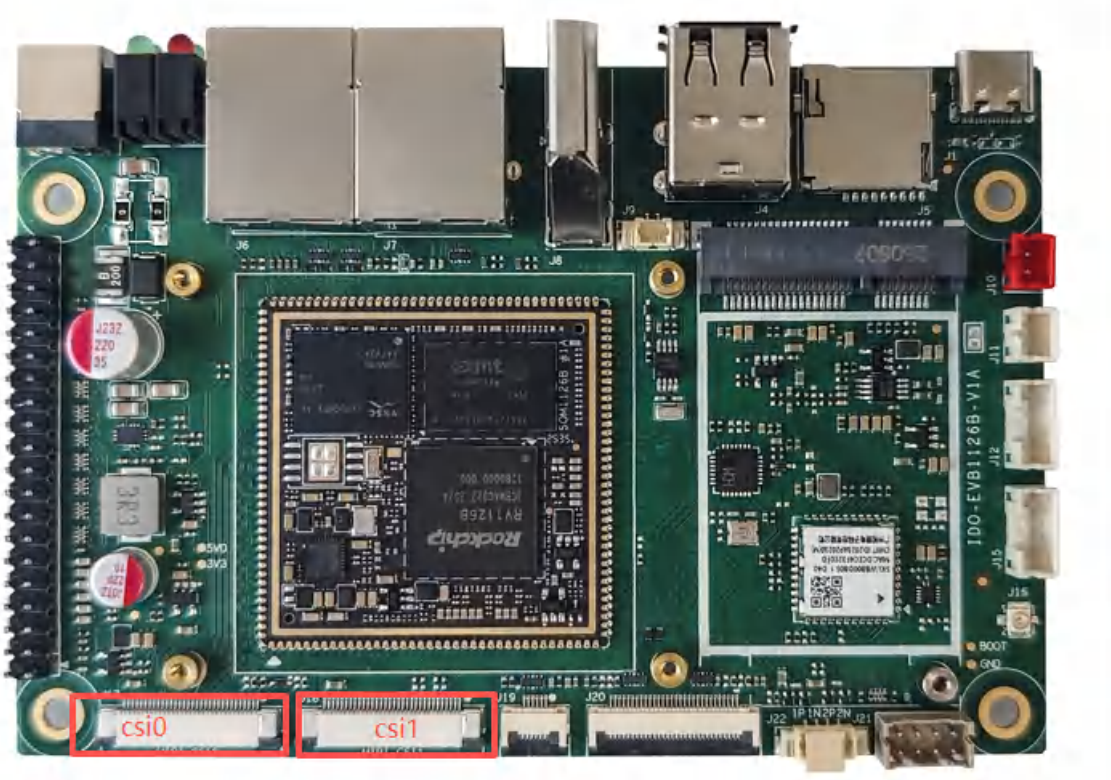
2.12. TP

I2C-TP接口为 6Pin FPC 0.5mm座子 (J13)，如下图所示：



2.13. MIPI Camera

主板内置两路MIPI Camer接口J17和J18，使用FPC05-PUW-32P抽拉式上接座子，如下图所示：



2.13.1. IMX415

J18 默认适配IMX415作为后置摄像头，接法如下图所示：

同时还支持的camera senor 有SC450AI,SC850SL,OV13855

J17	J18
/dev/video22	/dev/video30

2.13.1.1. 预览

```bash
gst-launch-1.0 v4l2src device=/dev/video22 ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! videoconvert ! autovideosink

gst-launch-1.0 v4l2src device=/dev/video30 ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! videoconvert ! autovideosink
```

2.13.1.2. 抓图

```bash
v4l2-ctl --verbose -d /dev/video22 --set-fmt-video=width=1920,height=1080,pixelformat='NV12' --stream-mmap=4 --stream-skip=3  --stream-to=./test-1920x1080-p50.yuv --stream-count=1 --stream-poll
```

2.13.1.3. 抓视频

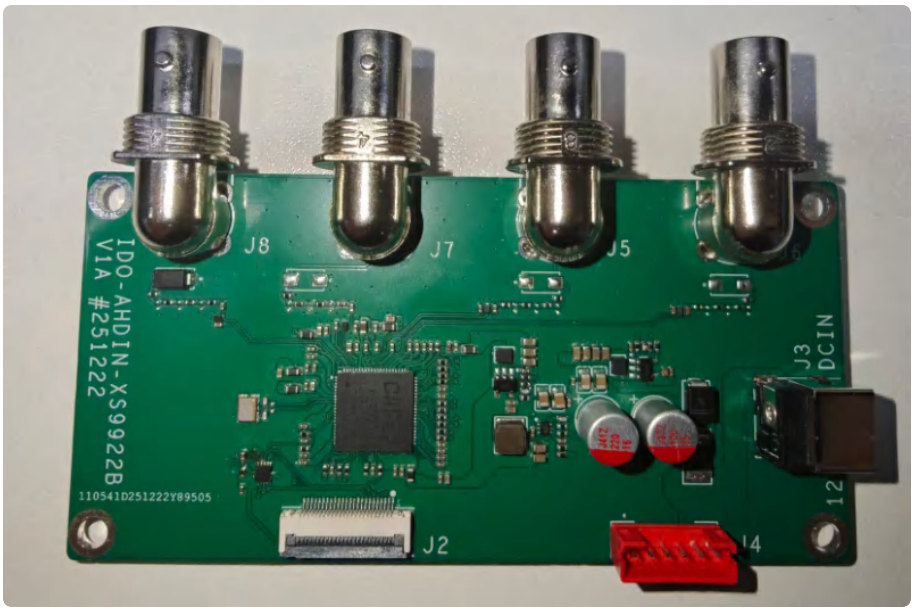
```bash
v4l2-ctl --stream-mmap=4  -d /dev/video22 --set-fmt-video=width=1920,height=1080,pixelformat='NV12' --stream-to=./output.yuv
```

2.13.2. TP2815/XS9922 AHD

TP2815


IDO-AHDIN-V1C

XS9922


IDO-AHDIN-XS9922B
V1A #251222
110541D251222Y89505
J8
J7
J5
J3
DCIN
J2
J4

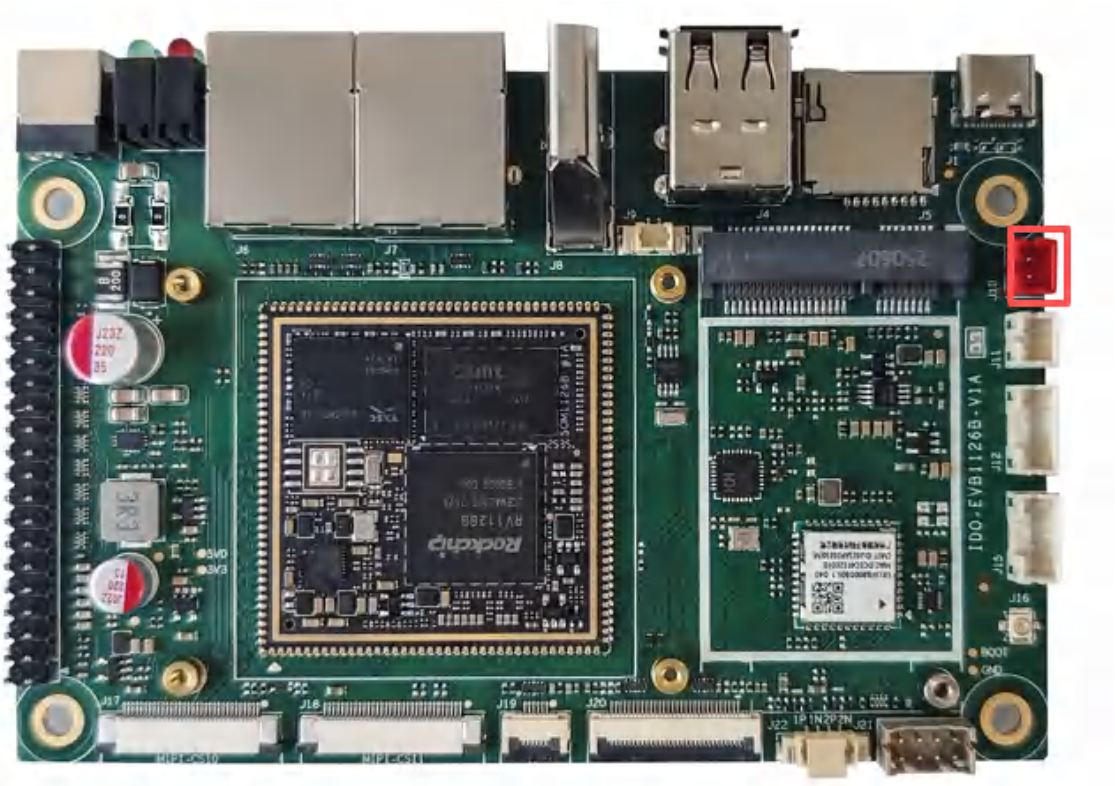
J17	J18
/dev/video0	/dev/video11
/dev/video1	/dev/video12
/dev/video2	/dev/video13
/dev/video3	/dev/video14

2.13.2.1. 预览

```bash
gst-launch-1.0 v4l2src device=/dev/video0 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink


gst-launch-1.0 v4l2src device=/dev/video1 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink


    gst-launch-1.0 v4l2src device=/dev/video2 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink


    gst-launch-1.0 v4l2src device=/dev/video3 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink


    gst-launch-1.0 v4l2src device=/dev/video11 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink

    gst-launch-1.0 v4l2src device=/dev/video12 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink


    gst-launch-1.0 v4l2src device=/dev/video13 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink


    gst-launch-1.0 v4l2src device=/dev/video14 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink

```

2.14. FAN 风扇

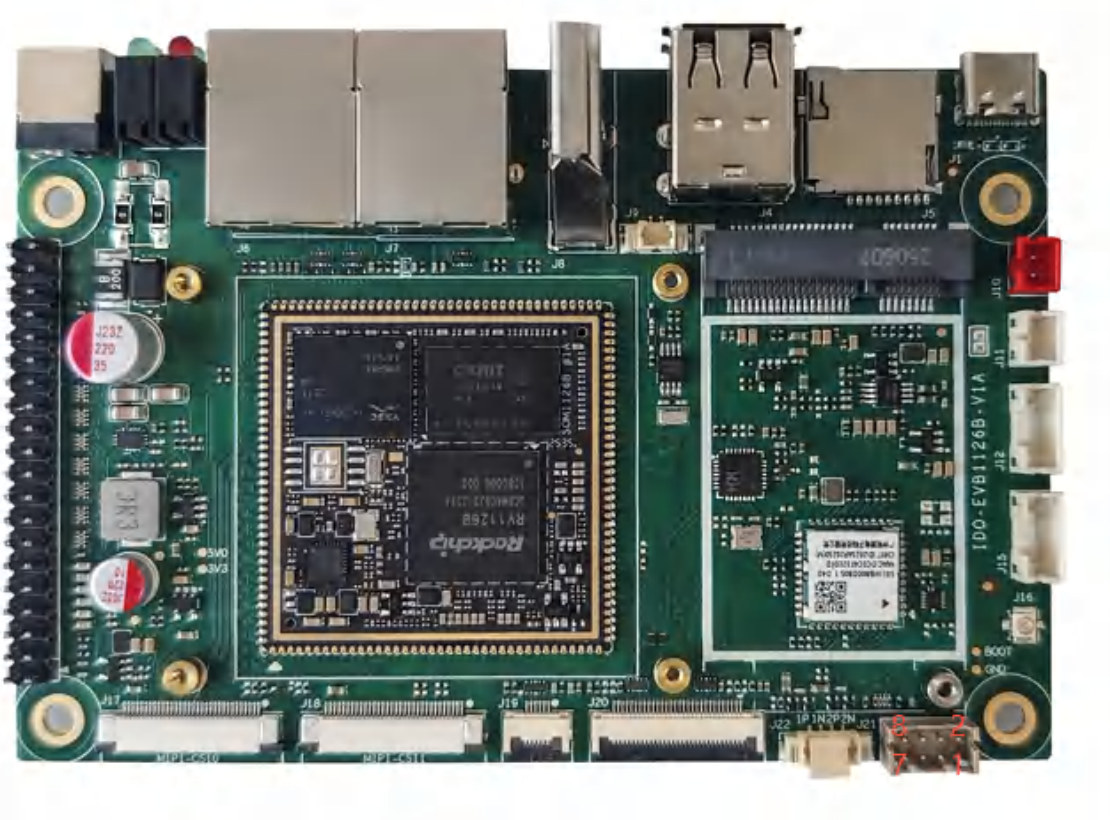
PH2 5V FAN风扇接口J3，如下图所示：

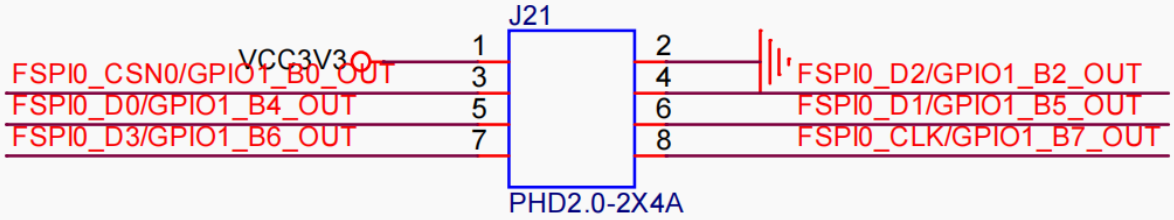


控制功能	控制命令
打开风扇	echo 1 > /sys/class/leds/fan_led/brightness
关闭风扇	echo 0 > /sys/class/leds/fan_led/brightness

2.15. GPIO

主板预留了8个GPIO接口J3，如下图所示：

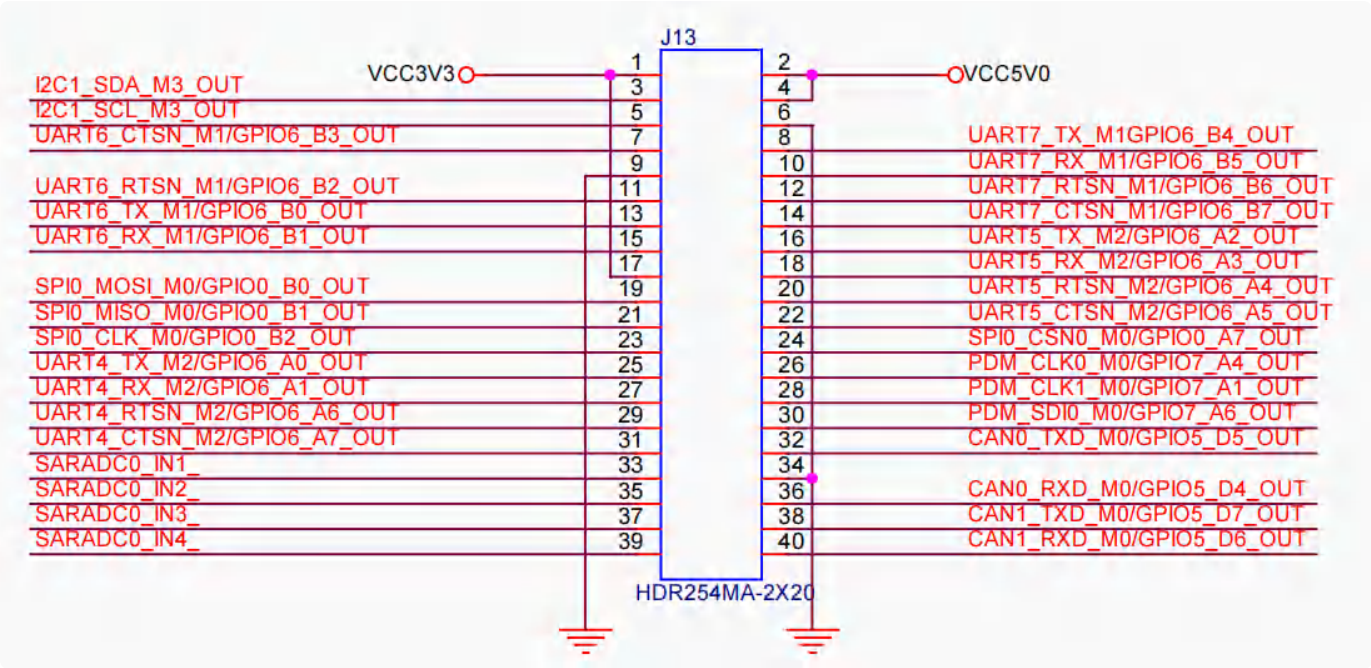




GPIO引脚定义描述，如下图所示：

序号	管脚标号	GPIO 序号
1	VCC3V3	/
2	GND	/
3	GPIO1_B0	40
4	GPIO1_B2	42
5	GPIO1_B4	44
6	GPIO1_B5	45
7	GPIO1_B6	46

8	GPIO1_B7	47



序号	管脚标号	GPIO 序号
7	GPIO6_B3	203
11	GPIO6_B2	202
19	GPIO0_B0	8
21	GPIO0_B1	9
23	GPIO0_B2	10
29	GPIO6_A6	198
31	GPIO6_A7	199
12	GPIO6_B6	206
14	GPIO6_B7	207
22	GPIO6_A5	197
24	GPIO0_A7	7
26	GPIO7_A4	228
28	GPIO7_A1	225

30	GPIO7_A6	230

执行以下脚本，使用万用表实际去量电压值，是否是高电平

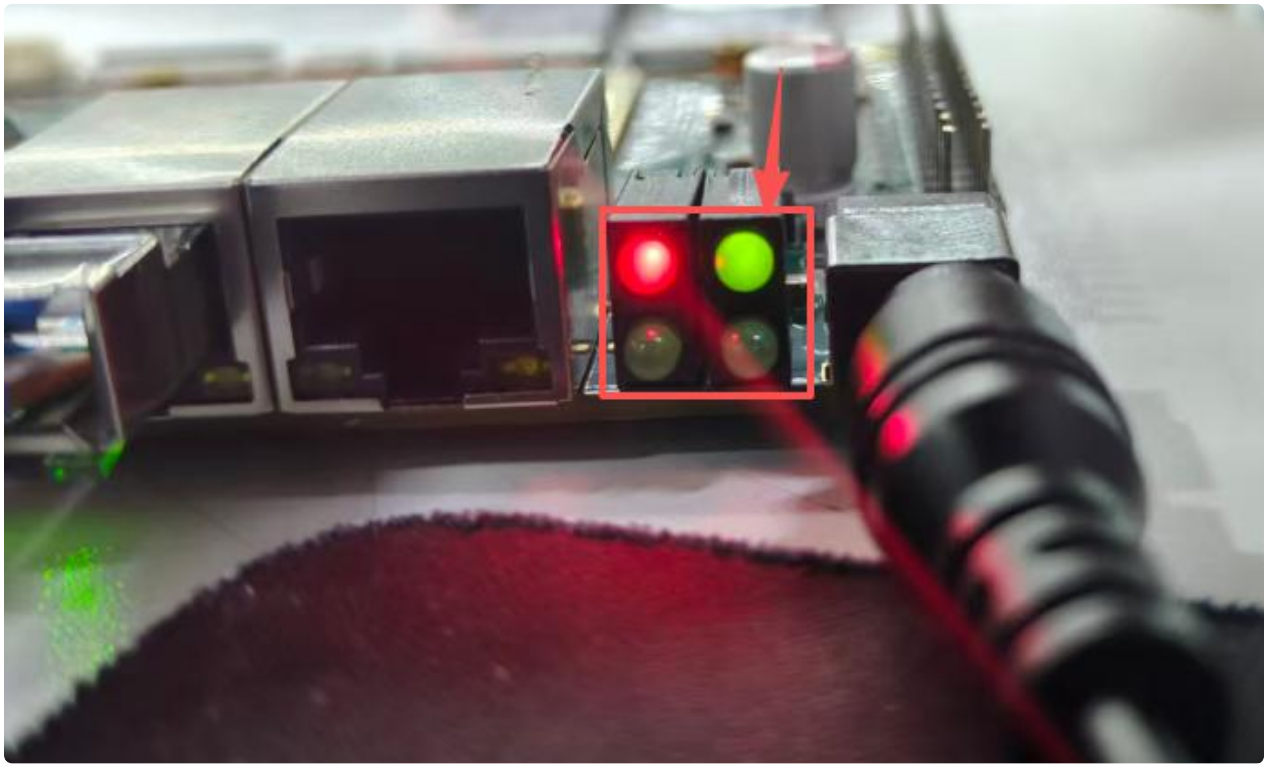
```C
#!/bin/bash

GPIO_NUM=(40 42 44 45 46 47  203 202 8 9 10 198 199 206 207 197 7 228 225 230)

for gpio in "${GPIO_NUM[@]}"; do
    echo "$gpio" > /sys/class/gpio/export 2>/dev/null
    if [ $? -eq 0 ]; then
        echo "GPIO $gpio exported successfully."
         echo out >  /sys/class/gpio/gpio$gpio/direction
        echo 1 >  /sys/class/gpio/gpio$gpio/value
        echo "GPIO $gpio Out High"
    else
        echo "GPIO $gpio already exported or failed to export."
    fi
done
```

执行以下脚本，使用万用表实际去量电压值，是否是低电平

test.sh Bash

```Bash
#!/bin/bash

GPIO_NUM=(40 42 44 45 46 47  203 202 8 9 10 198 199 206 207 197 7 228 225 230)

for gpio in "${GPIO_NUM[@]}"; do
        echo 0 >  /sys/class/gpio/gpio$gpio/value
        echo "GPIO $gpio in "

done
```

2.16. pwm



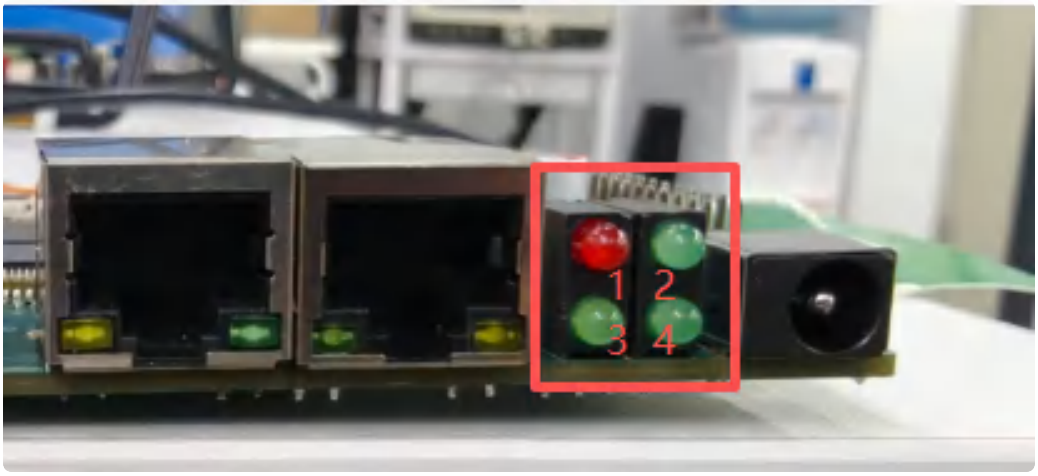
HDMI的 版本镜像pwm节点是 pwmchip1

```Bash
cd /sys/class/pwm/pwmchip1/
echo 0 > export
cd pwm0
echo 10000 > period
echo 5000 > duty_cycle
echo normal > polarity
echo 1 > enable
```

MIPI 的版本镜像 pwm节点是 pwmchip2

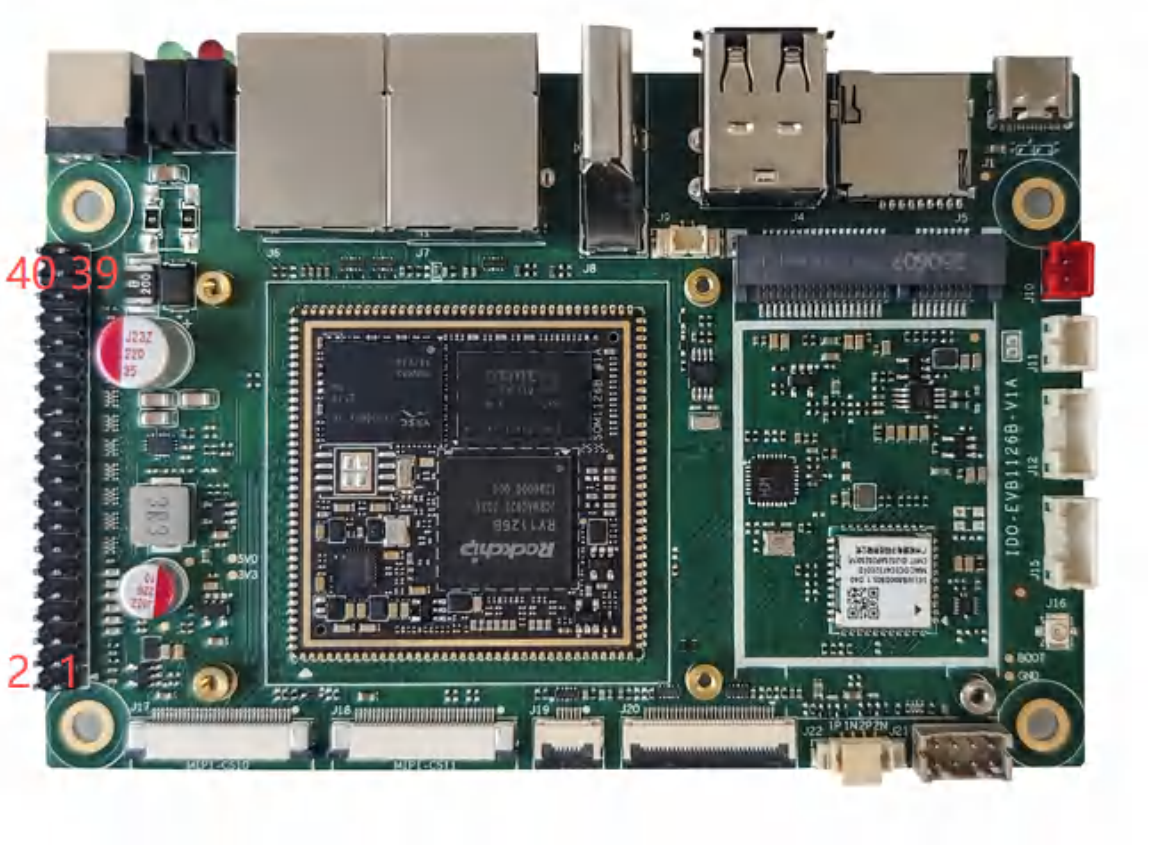
```Bash
cd /sys/class/pwm/pwmchip2/
echo 0 > export
cd pwm0
echo 10000 > period
echo 5000 > duty_cycle
echo normal > polarity
echo 1 > enable
```

2.17. LED



1. 电源指示灯
2. PWM–LED 灯
3. 系统运行指示灯
4. 4G模块指示灯

2.18. ADC



序号	ADC通道	读值

33	SARADC0_IN1_	cat /sys/bus/iio/devices/iio\:device0/in_voltage1_raw
35	SARADC0_IN2_	cat /sys/bus/iio/devices/iio\:device0/in_voltage2_raw
37	SARADC0_IN3_	cat /sys/bus/iio/devices/iio\:device0/in_voltage3_raw
39	SARADC0_IN4_	cat /sys/bus/iio/devices/iio\:device0/in_voltage4_raw

4路ADC精度为13位，参考电压为1.8V。

ADC输入电压计算公式：

$$V=(raw/8191)*1.8v$$

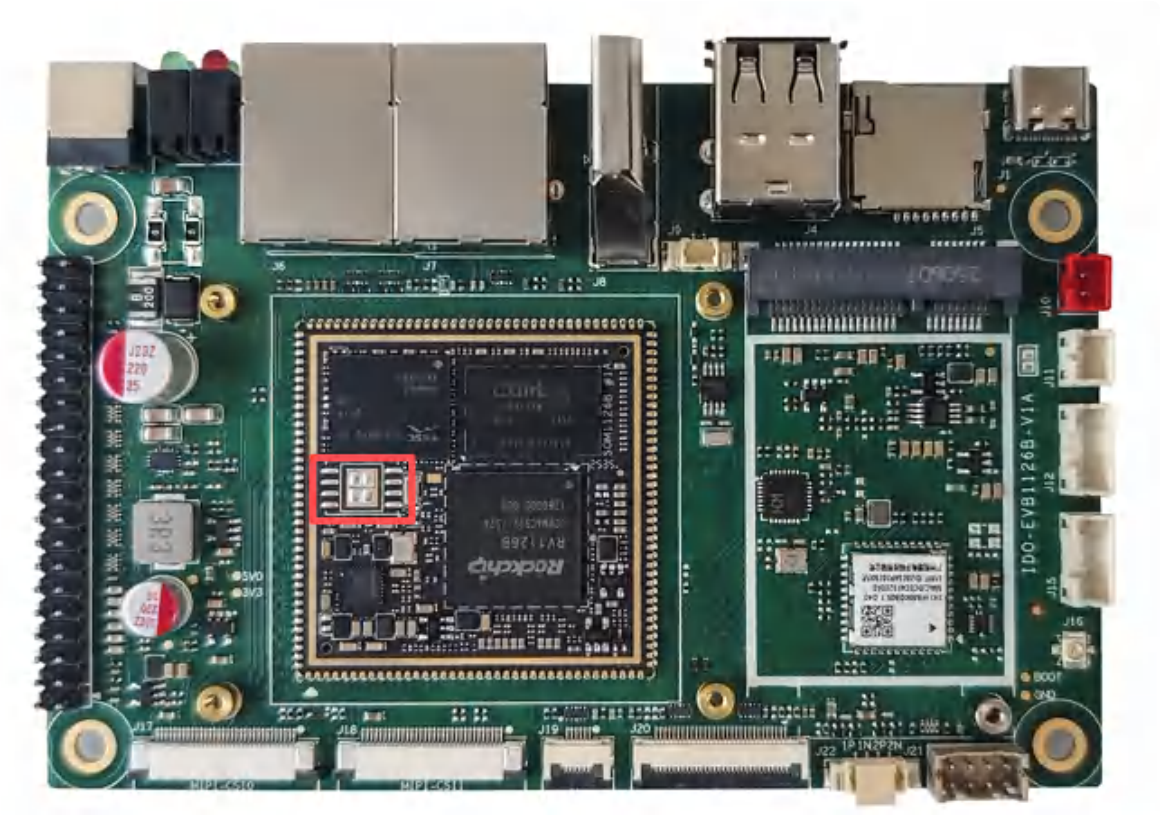
2.19. I2C

Bash

```
root@rv1126b-buildroot:/# i2cdetect -y -r 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:                         -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- 51 -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
root@rv1126b-buildroot:/#
```

注意： 测试需要使用转接板。

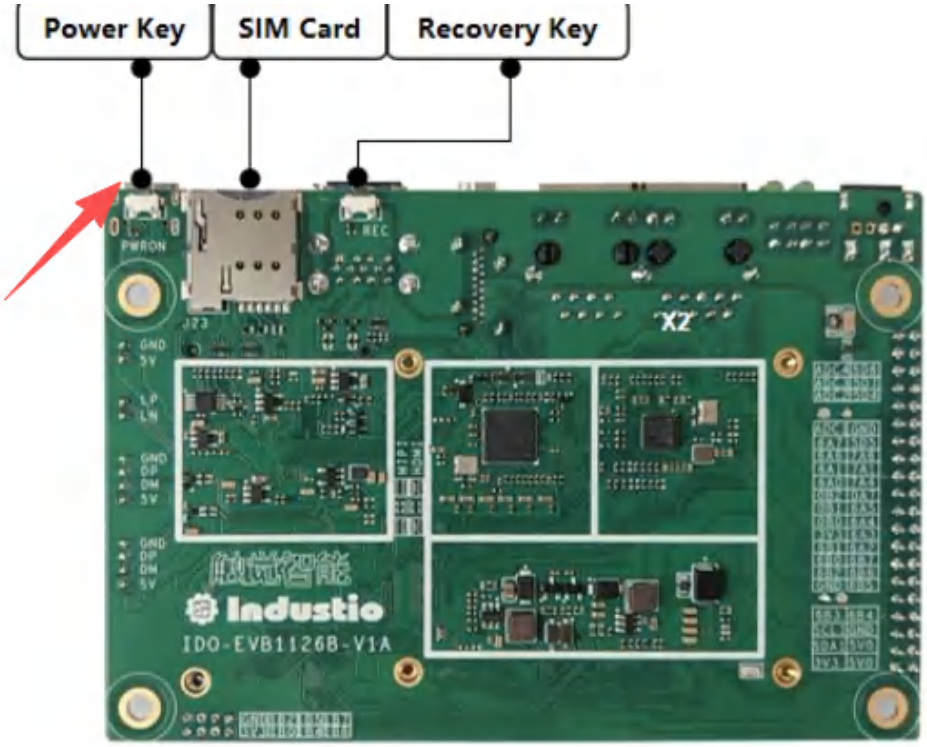
2.20. SPI NAND



Bash

```
root@rv1126b-buildroot:/# dmesg | grep spi
[    2.847504] spi-nand spi2.0: Winbond SPI NAND was found.
[    2.847513] spi-nand spi2.0: 512 MiB, block size: 128 KiB, page size: 2048, OOB size: 128
[    3.653787] mtdblock: MTD device 'spi-nand0' is NAND, please consider using UBI block devices instead.
[    4.464264] mtdblock: MTD device 'spi-nand0' is NAND, please consider using UBI block devices instead.
root@rv1126b-buildroot:/#
```

2.21. PWRON



按键	关机/开机	休眠/唤醒
PWRON	长按5S 关机/单击开机	单击休眠/单击唤醒

2.22. NPU

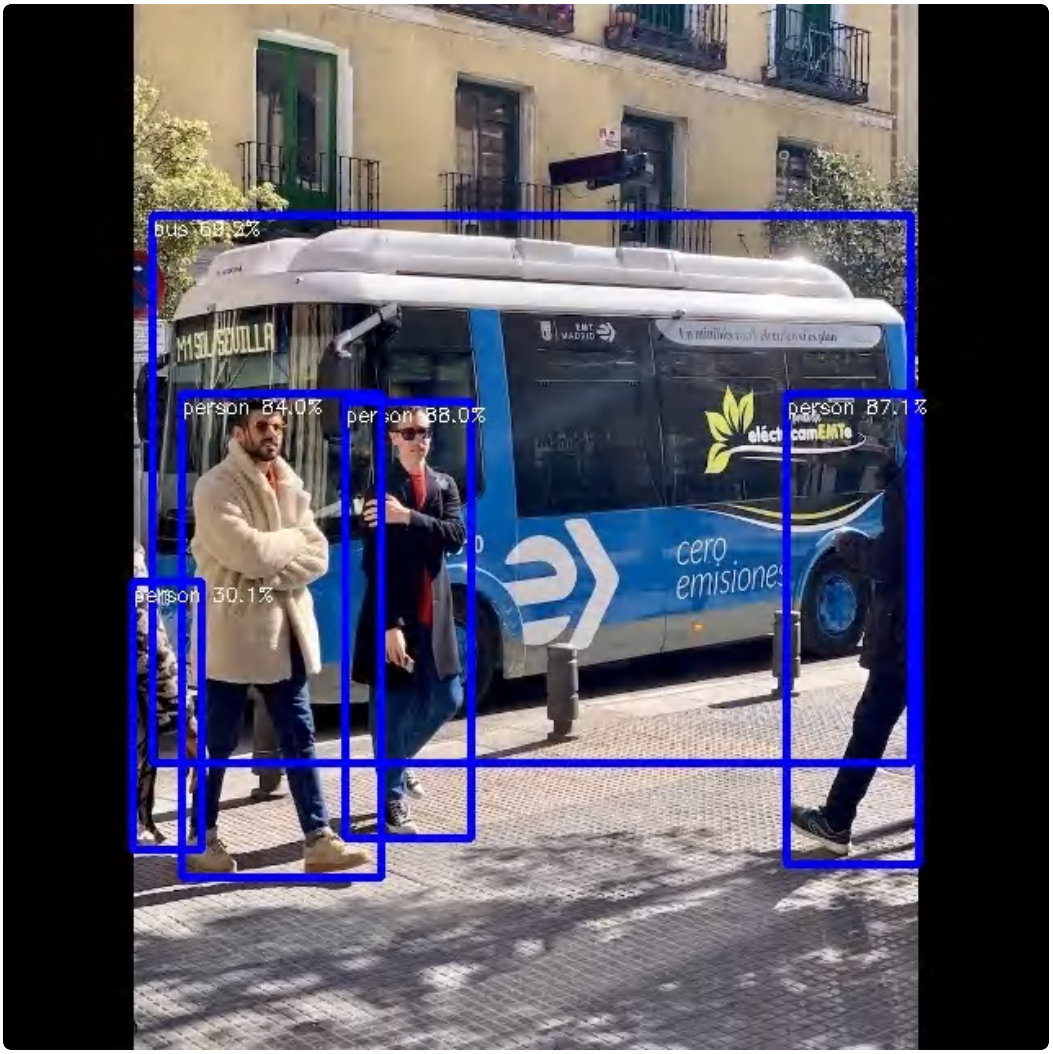
2.22.1. yolov5测试：

链接: https://pan.baidu.com/s/1N-vgqd1eH1Xcd7zxpfhZ4g 提取码: hnpm

2.22.1.1. rknn_yolov5_demo：

```shell
root@rv1126b-buildroot:/# unzip  rknn_yolov5_demo_Linux.zip
root@rv1126b-buildroot:/# cd rknn_yolov5_demo_Linux
root@rv1126b-buildroot:/# chmod a+x rknn_yolov5_demo
root@rv1126b-buildroot:/# ./rknn_yolov5_demo ./model/RV1126B/yolov5s-640-640.rknn ./model/bus.jpg
post process config: box_conf_threshold = 0.25, nms_threshold = 0.45
Loading mode...
sdk version: 2.3.2 (429f97ae6b@2025-04-09T09:09:27) driver version: 0.9.8
model input num: 1, output num: 3
  index=0, name=images, n_dims=4, dims=[1, 640, 640, 3], n_elems=1228800, size=1228800, w_stride = 640, size_with_stride=1228800, fmt=NHWC, type=INT8, qnt_type=AFFINE, zp=-128, scale=0.003922
  index=0, name=output0, n_dims=4, dims=[1, 255, 80, 80], n_elems=1632000, size=1632000, w_stride = 0, size_with_stride=1638400, fmt=NCHW, type=INT8, qnt_type=AFFINE, zp=-128, scale=0.003922
  index=1, name=286, n_dims=4, dims=[1, 255, 40, 40], n_elems=408000, size=408000, w_stride = 0, size_with_stride=491520, fmt=NCHW, type=INT8, qnt_type=AFFINE, zp=-128, scale=0.003922
  index=2, name=288, n_dims=4, dims=[1, 255, 20, 20], n_elems=102000, size=102000, w_stride = 0, size_with_stride=163840, fmt=NCHW, type=INT8, qnt_type=AFFINE, zp=-128, scale=0.003922
model is NHWC input fmt
model input height=640, width=640, channel=3
Read ./model/bus.jpg ...
img width = 640, img height = 640
once run use 37.585000 ms
loadLabelName ./model/coco_80_labels_list.txt
person @ (209 243 286 510) 0.879723
person @ (479 238 560 526) 0.870588
person @ (109 237 232 534) 0.828112
bus @ (93 129 553 464) 0.700761
person @ (79 353 122 517) 0.307297
save detect result to ./out.jpg
loop count = 10 , average run  33.047100 ms
```

执行完后会得到转换后的模型图片，如下所示：



2.22.1.2. rknn_yolov5_video_demo：

Shell

```
root@rv1126b-buildroot:/rknn_yolov5_demo_Linux# chmod a+x rknn_yolov5_video_demo
root@rv1126b-buildroot:/rknn_yolov5_demo_Linux# ./rknn_yolov5_video_demo ./model/RV1126B/yolov5s-640-640.rknn ./1280×720.h264 264

```

最终生成 out.h264

Shell

```
# 播放测试
gst-play-1.0 out.h264
```

2.22.2. yolo11 测试

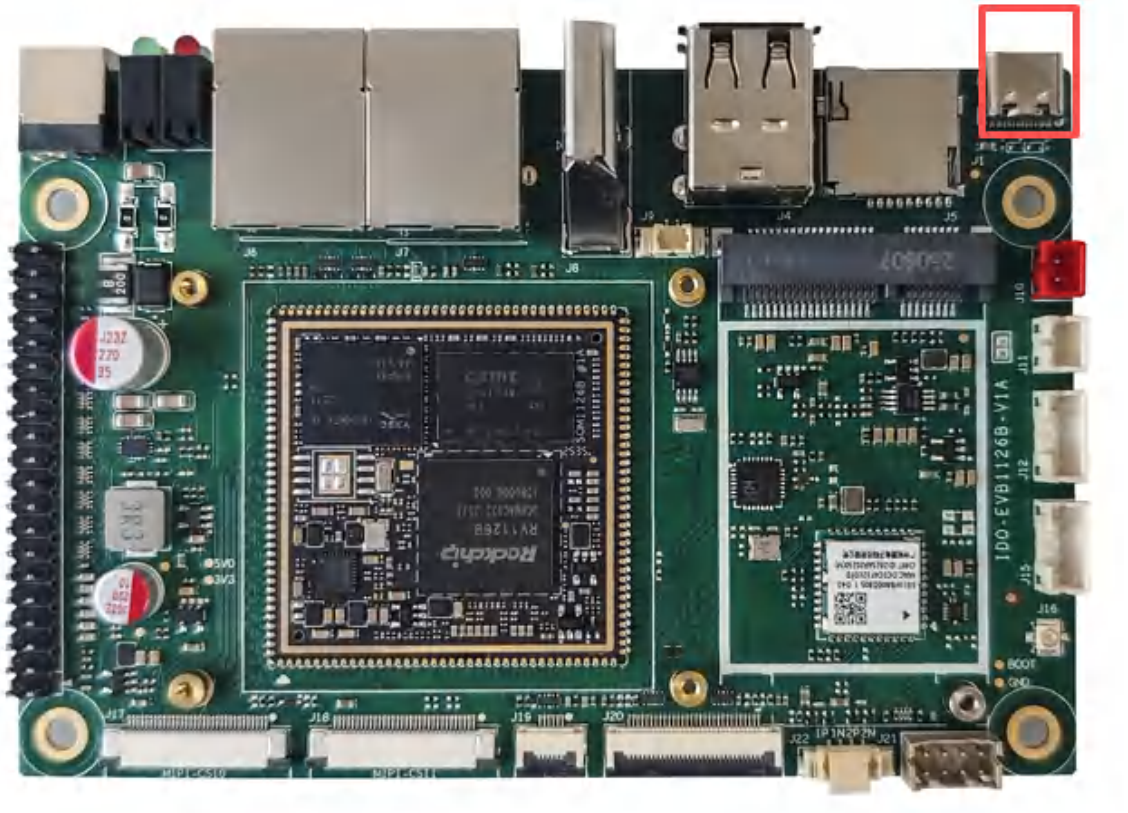
暂时不支持

3. IDO-EVB1126B-V1A-Debian12系统

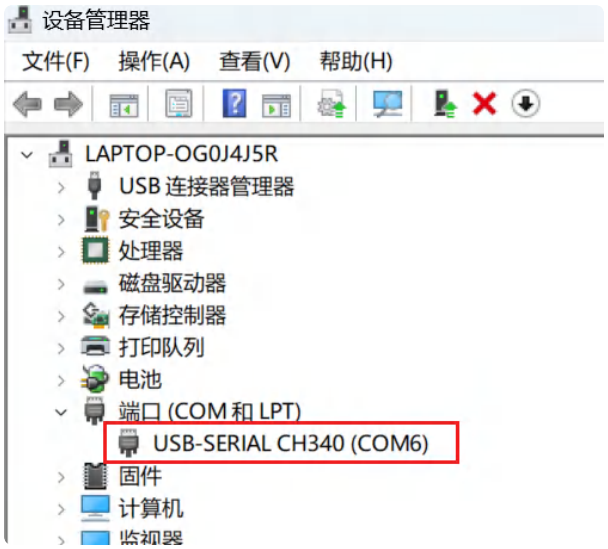
3.1. 功能及接口使用方法

3.2. 调试串口

调试串口位置J1，如下图所示：



使用USB Type-C数据线，连接PC端的USB接口。系统会识别到一个“USB-SERIAL CH340”端口设备。



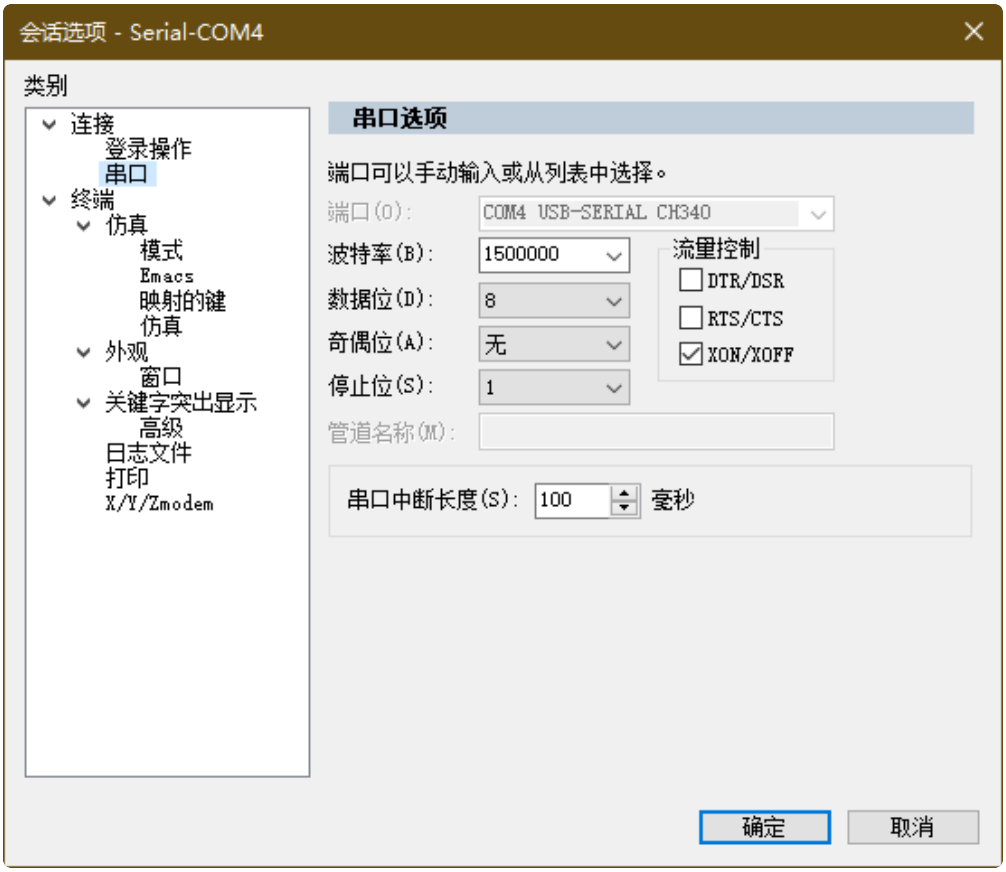
使用调试软件（MobaXterm、putty）等，设置参数如下：

波特率：1500000

数据位：8

奇偶校验位：无

停止位：1

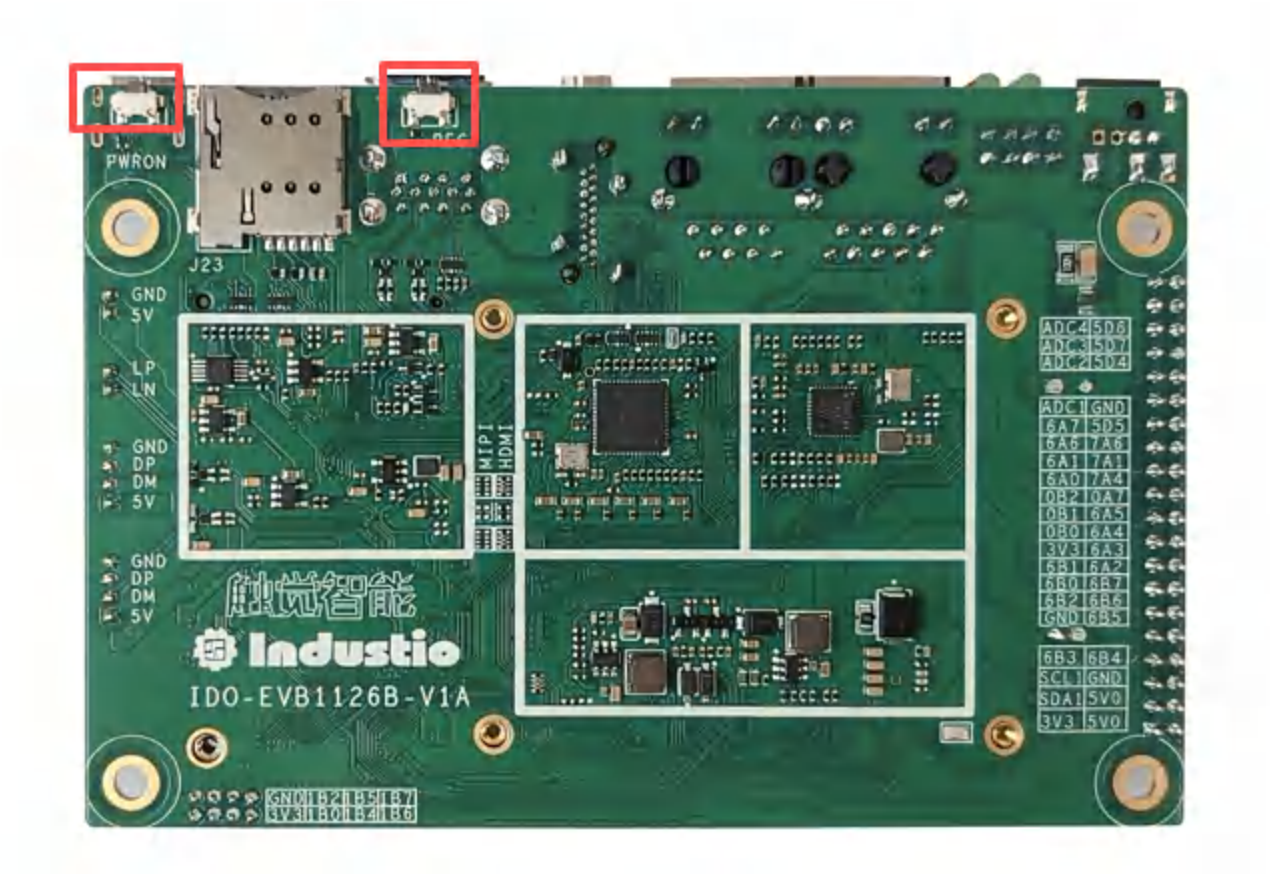


3.3. 按键

主板自带1个Recovery按键REC

Recovery按键，关机状态下，按住Recovery按键再给主板上电，主板会进入Loader模式。

power长按 强制关机，

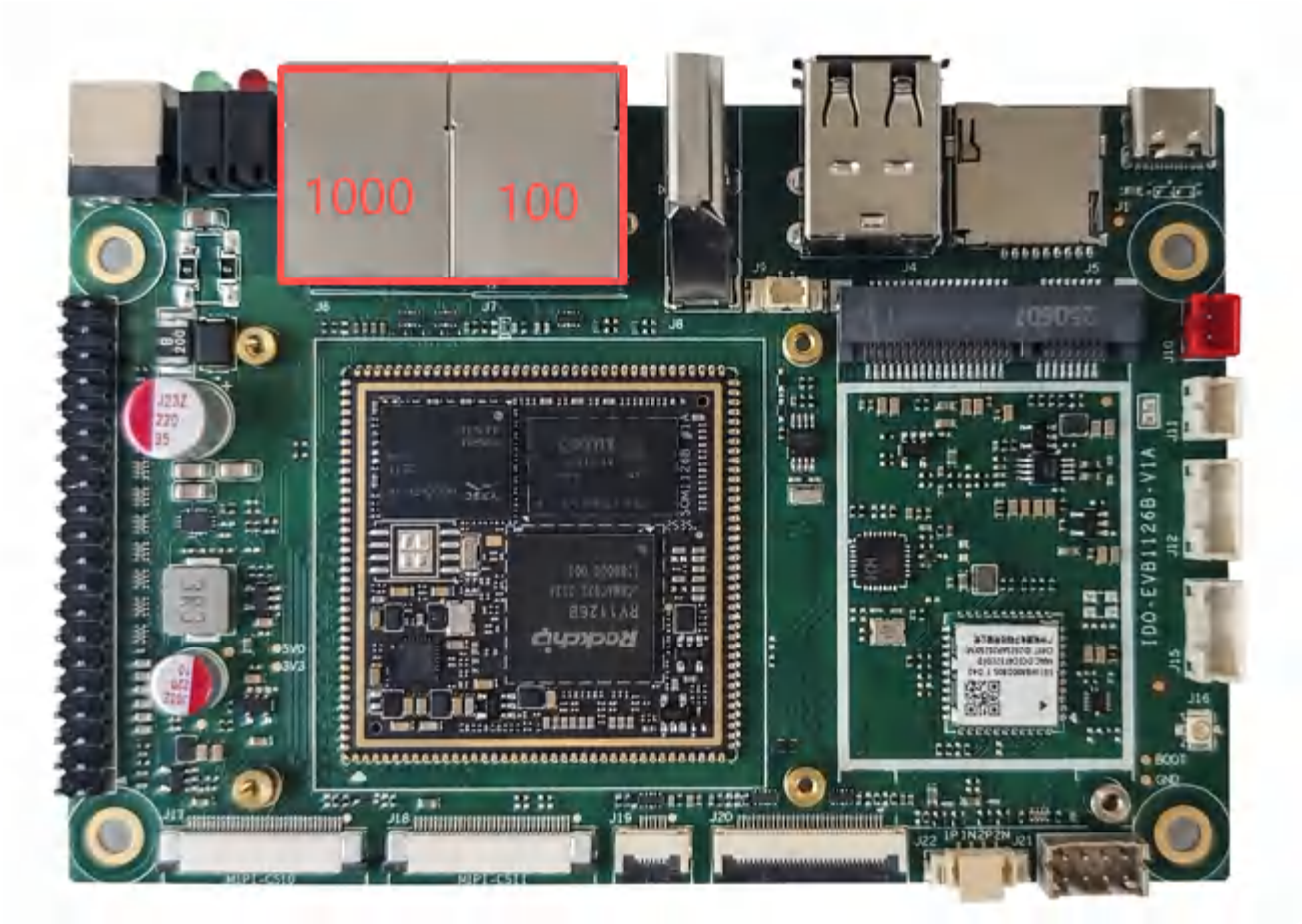


序号	定义	说明
1	PWRON_KEY	电源按键，用于开机/关机，休眠/唤醒；
2	RECOVERY_KEY	烧录按键，用于烧录

3.4. 网络

3.4.1. Ethernet

主板有 1 路千兆以太网，1路百兆以太网，如下图所示：




```Bash
# TCP
#server
iperf -s -i 1
#client
iperf -c server.ip   # 上行
iperf -c server.ip -R 下行

# UDP
#server
iperf -s -i 1
#client
iperf -c server.ip -u -b 1000M    # 上行
iperf -c server.ip -u -b 1000M  -R # 下行
```

可通过ifconfig指令设置临时静态IP，命令如下：

```Bash
$ ifconfig eth0 192.168.1.151 netmask 255.255.255.0 up
$ route add -net 192.168.1.0 netmask 255.255.255.0 dev eth0
$  route add default gw 192.168.1.1 dev eth0
$ echo "nameserver 8.8.8.8" >> /etc/resolv.conf
```

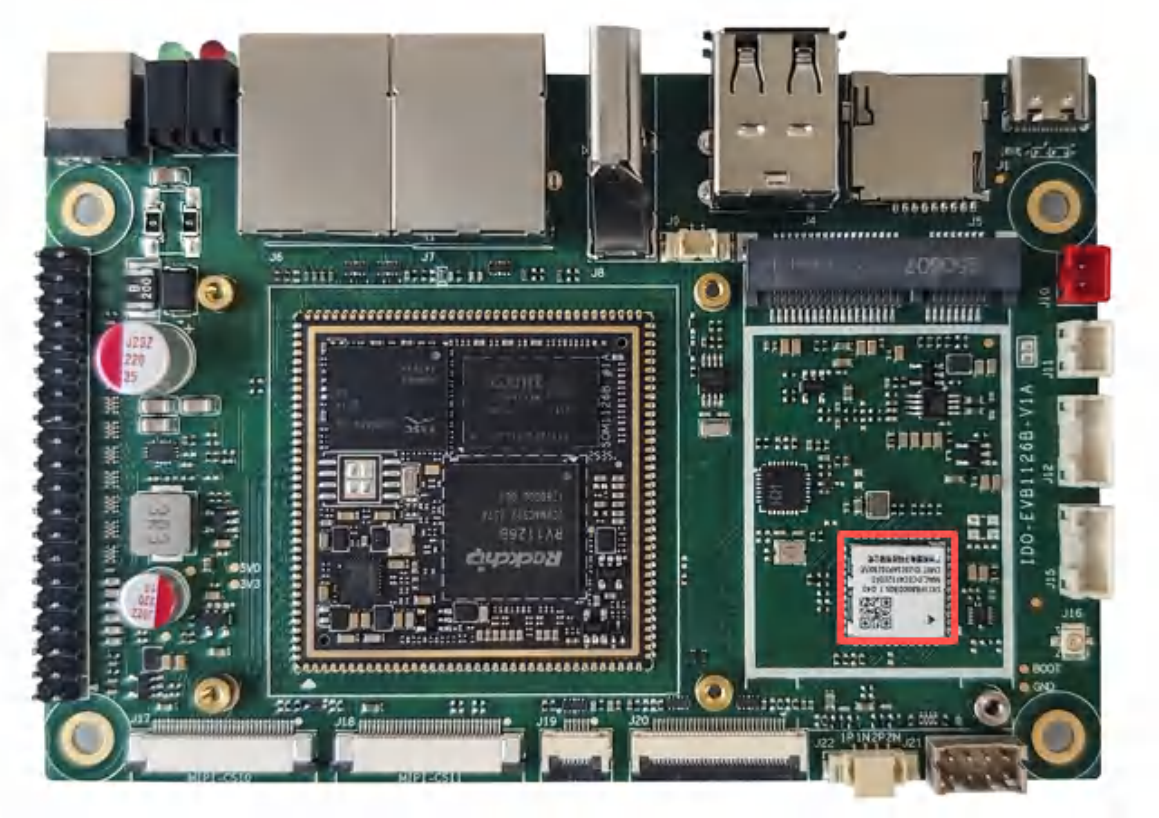
如果想让静态IP永久生效，编辑文件 /etc/network/interfaces

```Bash
root@linaro-alip:~# vi /etc/network/interfaces
root@linaro-alip:~# cat /etc/network/interfaces
# interface file auto-generated by buildroot

auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.3.123
netmask 255.255.255.0
gateway 192.168.3.1

source-directory /etc/network/interfaces.d
```

设置后，重启

3.4.2. WIFI

使用WiFi/蓝牙时，需要连接天线以获得良好的信号，WiFi模块和天线座子J62，如下图所示：



使用nmcli工具连接wifi热点：

```Bash
root@linaro-alip:~# nmcli dev wifi list #查询wifi列表
root@linaro-alip:~# nmcli dev wifi connect "industio" password "123456789" ifname wlan0 #连接热点
```

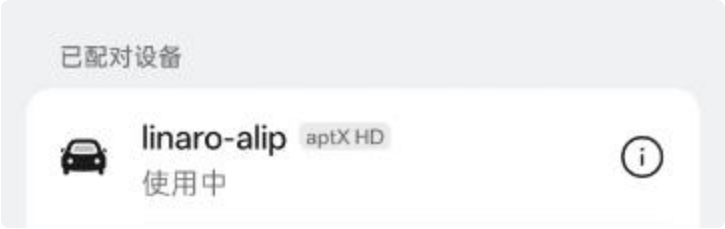
3.4.3. Bluetooth

蓝牙节点：

```Bash
root@linaro-alip:~# hciconfig -a
hci0: Type: Primary  Bus: UART
  BD Address: DC:EC:4F:32:E6:D4  ACL MTU: 1021:9  SCO MTU: 255:4
  UP RUNNING
  RX bytes:930 acl:0 sco:0 events:73 errors:0
  TX bytes:5493 acl:0 sco:0 commands:73 errors:0
  Features: 0xbf 0x2e 0x4d 0xfe 0xd8 0x3f 0x7b 0x87
  Packet type: DM1 DM3 DM5 DH1 DH3 DH5 HV1 HV3
  Link policy: RSWITCH SNIFF
  Link mode: PERIPHERAL ACCEPT
  Name: 'linaro-alip'
  Class: 0x7c0000
  Service Classes: Rendering, Capturing, Object Transfer, Audio, Telephony
  Device Class: Miscellaneous,
  HCI Version:  (0xd)  Revision: 0xb
  LMP Version:  (0xd)  Subversion: 0xb
  Manufacturer: not assigned (2875)

root@linaro-alip:~#
```

3.4.3.1. 命令行连接

```bash
bluetoothctl
hci0 new_settings: powered bondable ssp br/edr le secure-conn
Agent registered
[CHG] Controller 4C:38:00:2B:C1:71 Pairable: yes
[bluetooth]#
#打开蓝牙
[bluetooth]# power on
#扫描蓝牙
[bluetooth]# scan on
# # 开启广播
[bluetooth]# discoverable on
# 允许配对
[bluetooth]# pairable on
# 使用 NoInputNoOutput (Just Works)
[bluetooth]# agent NoInputNoOutput
[bluetooth]# default-agent
#查看蓝牙设备
[bluetooth]# devices
#信任蓝牙设备
[bluetooth]# trust 7C:C1:80:09:DD:6C
#蓝牙配对
[bluetooth]# pair 7C:C1:80:09:DD:6C
#连接蓝牙
[bluetooth]# connect 7C:C1:80:09:DD:6C
```



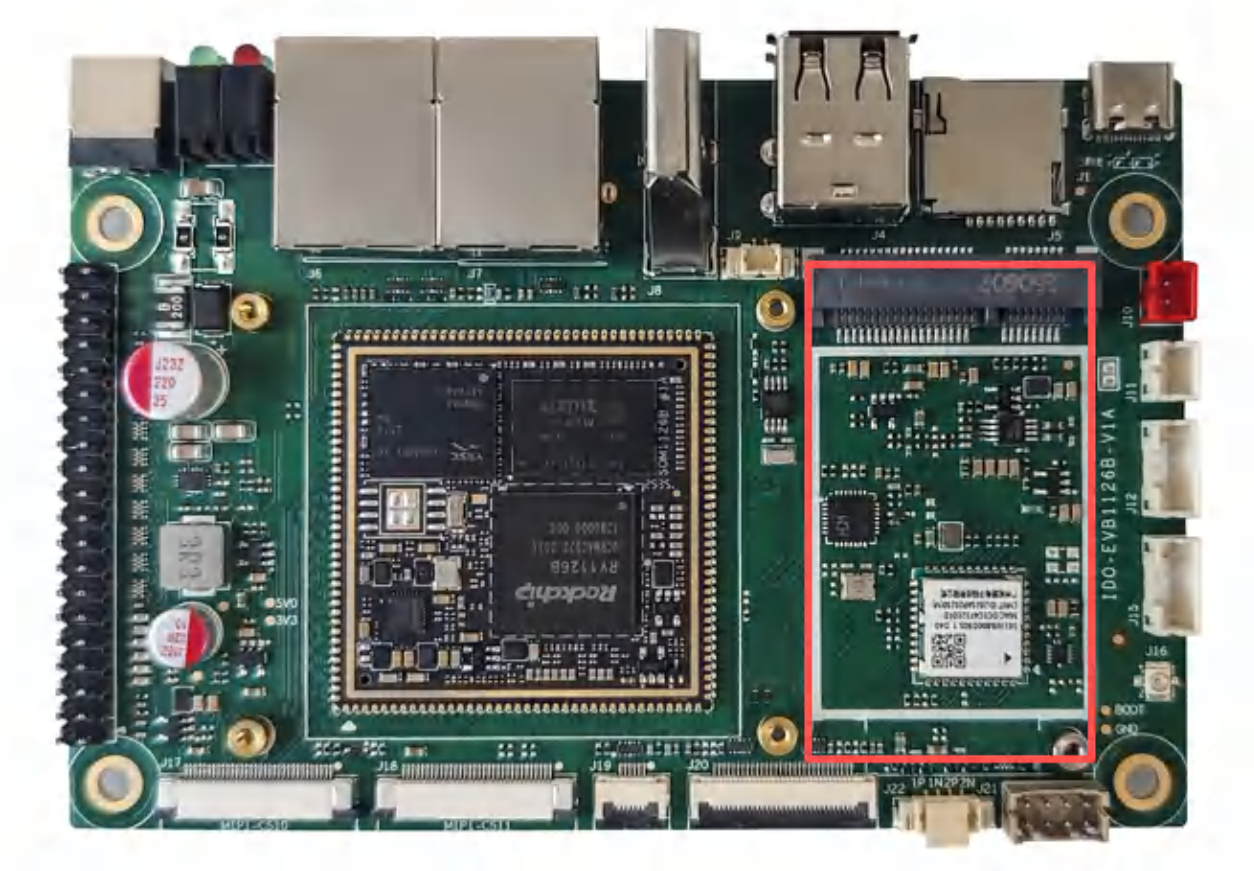

```
[bluetooth]# connect A8:C5:6F:AA:FC:A8
Attempting to connect to A8:C5:6F:AA:FC:A8
[CHG] Device A8:C5:6F:AA:FC:A8 Connected: yes
[NEW] Endpoint /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/sep1
[NEW] Endpoint /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/sep2
[NEW] Endpoint /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/sep3
[NEW] Endpoint /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/sep4
[NEW] Endpoint /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/sep5
[NEW] Transport /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/sep4/fd0
Connection successful
[NEW] Player /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/player0 [default]
[CHG] Device A8:C5:6F:AA:FC:A8 ServicesResolved: yes
[CHG] Player /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/player0 Type: Audio
[CHG] Player /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/player0 Subtype: None
[CHG] Player /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/player0 Status: paused
[CHG] Player /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/player0 Name: Bluetooth Player
[CHG] Player /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/player0 Repeat: off
[CHG] Player /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/player0 Shuffle: off
[CHG] Player /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/player0 Title: 凌晨的国道上，有赌徒，有破产的老板，有三教九流的大神，唯独没有
[CHG] Player /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/player0 TrackNumber: 0x00000001 (1)
[CHG] Player /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/player0 NumberOfTracks: 0x00000001 (1)
[CHG] Player /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/player0 Duration: 0x0000491b (18715)
[CHG] Player /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/player0 Artist: 摄影师-高原
[CHG] Player /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/player0 Position: 0x00000dec (3564)
[CHG] Player /org/bluez/hci0/dev_A8_C5_6F_AA_FC_A8/player0 Position: 0x00000dec (3564)
[CHG] Controller DC:EC:4F:1D:E7:C6 Discoverable: no
[OPPO Find X8]#
```

注意：如果出现无法配对的情况，请执行quit退出bluetoothctl ，重新启动蓝牙服务，再次执行以上连接命令

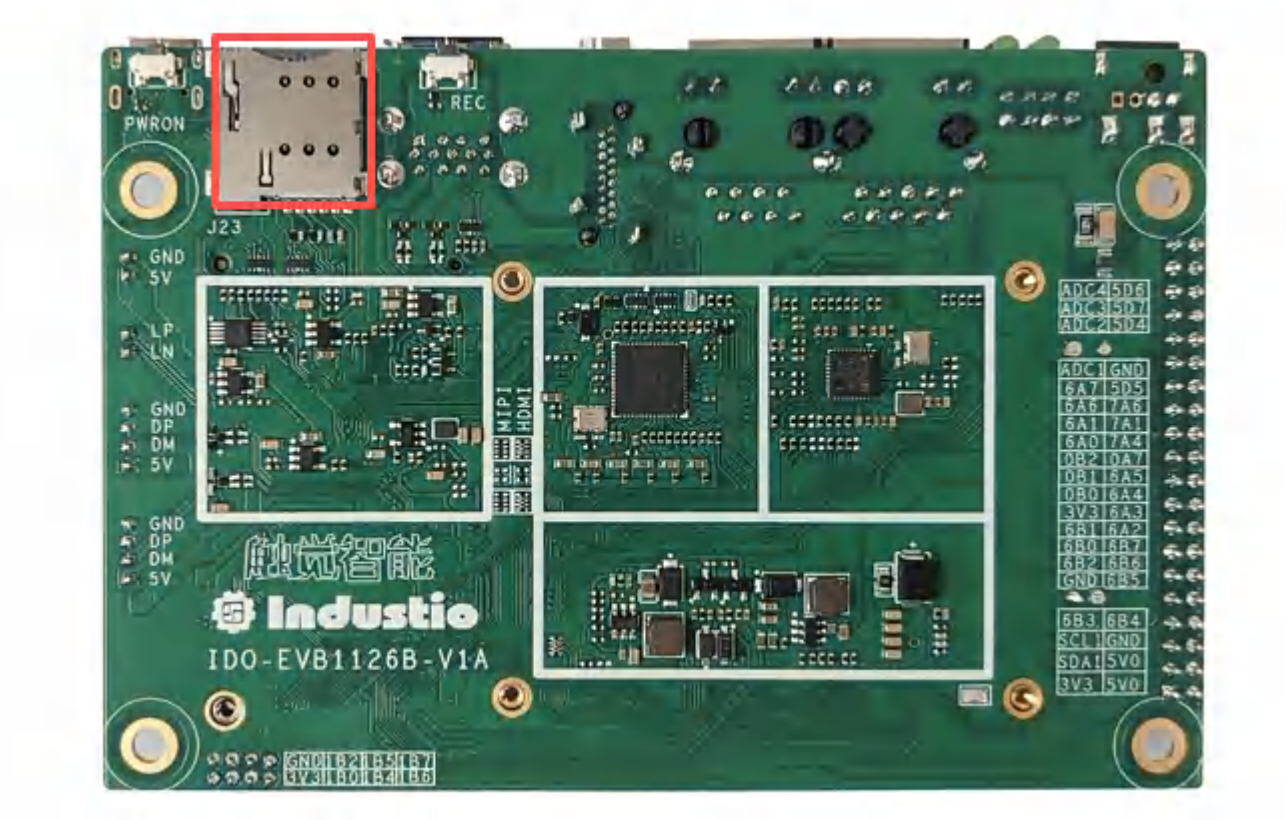
```Bash
[bluetooth]# quit
root@linaro-alip:/#  systemctl restart bluetooth
```

3.4.4. 4G/5G

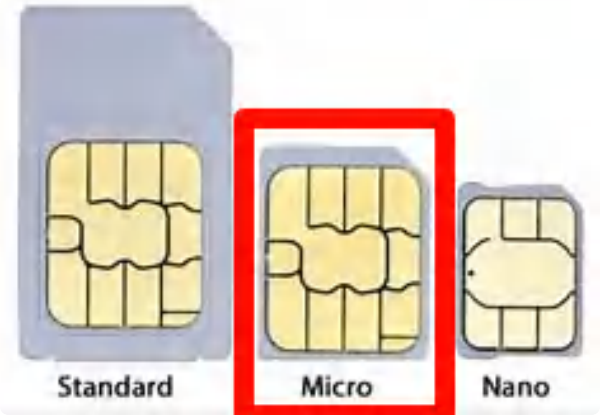
主板内置Mini PCIe 座子扩展 4G/5G模块，4G通信模块适配移远EC20、EC25等通用模组。如下图所示：

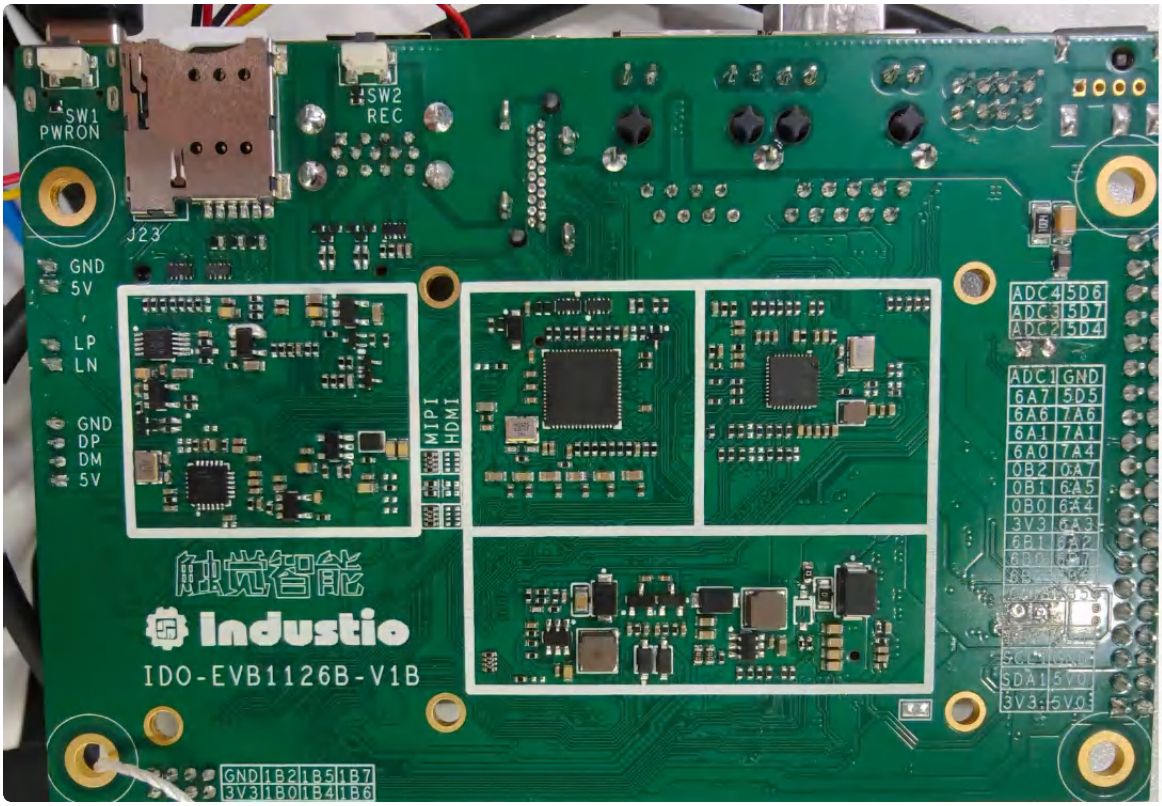


模块需要接上天线以确保获得更好的信号质量，SIM卡安装位J23位于主板背面，如下图所示：



使用Micro尺寸SIM卡，如下图所示：





EC20/25	ecm模式	rmnet模式
节点	usb0	wwan0
拨号方式	自动拨号	quectel-CM

ec20 模式切换。，切换后重启

```Bash
echo -e "AT+QCFG=\"usbnet\",1\r\n" > /dev/ttyUSB2   #节点usb0
echo -e "AT+QCFG=\"usbnet\",0\r\n" > /dev/ttyUSB2   #节点wwan0
```

测试模块EC20，ecm模式(usb0)

```Bash
echo -e "AT+CGDCONT=1.\"IP\",\"apn\"\r\n"  > /dev/ttyUSB2
echo -e "AT+QNETDEVCTL=1,1,1\r\n"  > /dev/ttyUSB2
```

```Bash
root@linaro-alip:/# ifconfig usb0
usb0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.225.30  netmask 255.255.255.0  broadcast 192.168.225.255
        inet6 fe80::771f:e144:67dd:d9ad  prefixlen 64  scopeid 0x20<link>
        inet6 240e:47c:3080:e:88ce:6089:1be:5ff4  prefixlen 64  scopeid 0x0<global>
        ether 36:b4:9e:ab:ab:01  txqueuelen 1000  (Ethernet)
        RX packets 20542  bytes 27780090 (26.4 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 10008  bytes 868483 (848.1 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@linaro-alip:/# ping baidu.com -I usb0
PING baidu.com (39.156.70.37) from 192.168.225.30 usb0: 56(84) bytes of data.
64 bytes from 39.156.70.37 (39.156.70.37): icmp_seq=1 ttl=47 time=208 ms
64 bytes from 39.156.70.37 (39.156.70.37): icmp_seq=2 ttl=47 time=74.8 ms
64 bytes from 39.156.70.37 (39.156.70.37): icmp_seq=3 ttl=47 time=82.8 ms
^C
--- baidu.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 74.768/121.737/207.605/60.807 ms

```

测试模块EC20，qmi(wwan0)

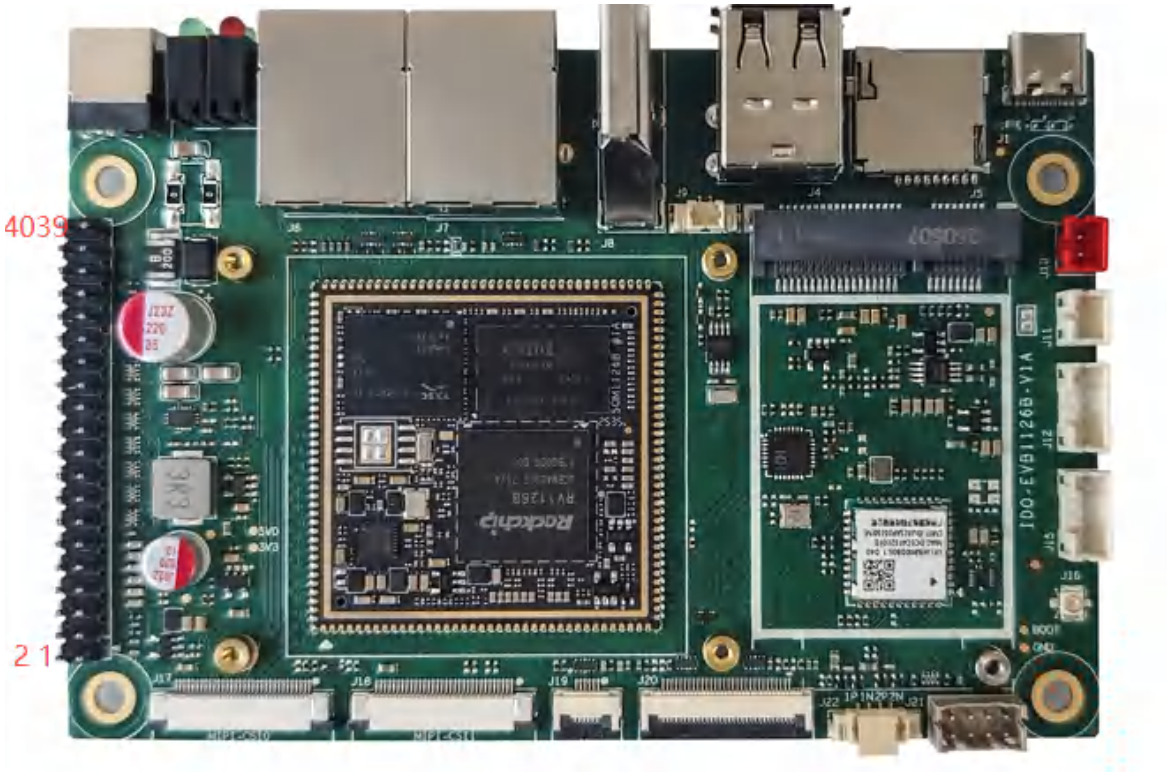
```Bash
# 模式修改
echo -e "AT+QCFG=\"usbnet\",0\r\n" > /dev/ttyUSB2   #节点wwan0
# 修改后需要重启系统，或者重启模块
#重启模块命令
echo -e "AT+CFUN=1,1\r\n" >/dev/ttyUSB2
```

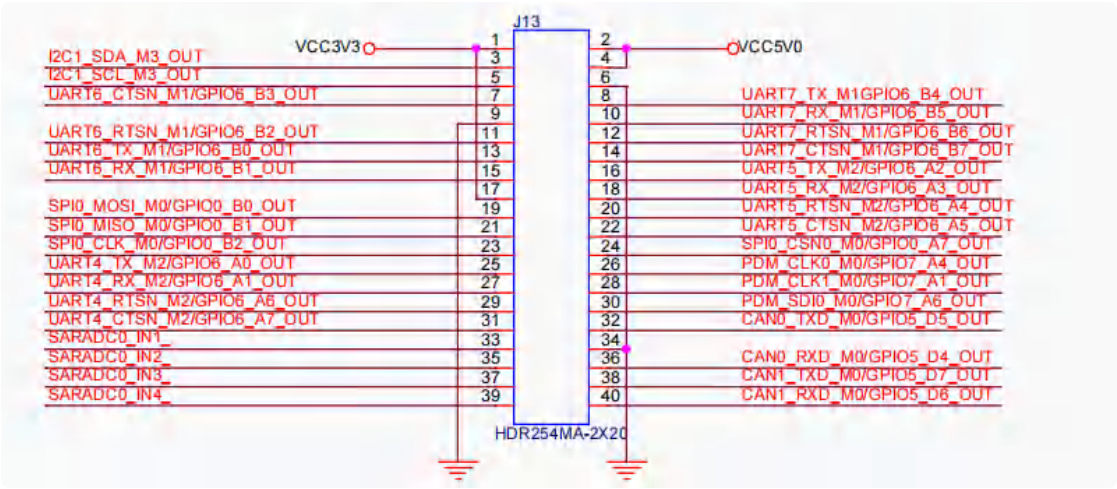
```
root@linaro-alip:/# quectel-CM &
[1] 1963
root@linaro-alip:/# [10-18_17:16:56:845] QConnectManager_Linux_V1.6.4
[10-18_17:16:56:847] Find /sys/bus/usb/devices/1-1.2 idVendor=0x2c7c idProduct=0x125, bus=0x001, dev=0x005
[10-18_17:16:56:848] Auto find qmichannel = /dev/cdc-wdm0
[10-18_17:16:56:848] Auto find usbnet_adapter = wwan0
[10-18_17:16:56:848] netcard driver = qmi_wwan, driver version = 6.1.141
[10-18_17:16:56:848] Modem works in QMI mode
[10-18_17:16:56:861] cdc_wdm_fd = 7
[10-18_17:16:56:947] Get clientWDS = 5
[10-18_17:16:56:979] Get clientDMS = 1
[10-18_17:16:57:010] Get clientNAS = 2
[10-18_17:16:57:043] Get clientUIM = 1
[10-18_17:16:57:075] Get clientWDA = 1
[10-18_17:16:57:107] requestBaseBandVersion EC25EUXGAR08A09M1G
[10-18_17:16:57:235] requestGetSIMStatus SIMStatus: SIM_READY
[10-18_17:16:57:267] requestGetProfile[1] ctlte///0/IPV4V6
[10-18_17:16:57:299] requestRegistrationState2 MCC: 460, MNC: 11, PS: Attached, DataCap: LTE
[10-18_17:16:57:330] requestQueryDataCall IPv4ConnectionStatus: DISCONNECTED
[10-18_17:16:57:331] ifconfig wwan0 0.0.0.0
[10-18_17:16:57:340] ifconfig wwan0 down
[10-18_17:16:57:395] requestSetupDataCall WdsConnectionIPv4Handle: 0x868a7ea0
[10-18_17:16:57:524] ifconfig wwan0 up
[10-18_17:16:57:533] busybox udhcpc -f -n -q -t 5 -i wwan0
udhcpc: started, v1.35.0
/etc/resolvconf/update.d/libc: Warning: /etc/resolv.conf is not a symbolic link to /run/resolvconf/resolv.conf
udhcpc: broadcasting discover
udhcpc: broadcasting select for 10.2.29.162, server 10.2.29.161
udhcpc: lease of 10.2.29.162 obtained from 10.2.29.161, lease time 7200
/etc/resolvconf/update.d/libc: Warning: /etc/resolv.conf is not a symbolic link to /run/resolvconf/resolv.conf



root@linaro-alip:/# ifconfig wwan0
wwan0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST>  mtu 1500
        inet 10.2.29.162  netmask 255.255.255.252  destination 10.2.29.162
        inet6 fe80::28fd:1308:143f:2ae9  prefixlen 64  scopeid 0x20<link>
        unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  txqueuelen 1000  (UNSPEC)
```

```
        RX packets 4  bytes 1224 (1.1 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 35  bytes 4960 (4.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@linaro-alip:/# ping baidu.com -I wwan0
PING baidu.com (39.156.70.37) from 10.2.29.162 wwan0: 56(84) bytes of dat
a.
64 bytes from 39.156.70.37 (39.156.70.37): icmp_seq=1 ttl=48 time=62.9 ms
64 bytes from 39.156.70.37 (39.156.70.37): icmp_seq=2 ttl=48 time=65.8 ms
64 bytes from 39.156.70.37 (39.156.70.37): icmp_seq=3 ttl=48 time=58.8 ms
64 bytes from 39.156.70.37 (39.156.70.37): icmp_seq=4 ttl=48 time=59.7 ms
64 bytes from 39.156.70.37 (39.156.70.37): icmp_seq=5 ttl=48 time=57.9 ms
^C64 bytes from 39.156.70.37: icmp_seq=7 ttl=48 time=58.3 ms

--- baidu.com ping statistics ---
7 packets transmitted, 6 received, 14.2857% packet loss, time 6003ms
rtt min/avg/max/mdev = 57.888/60.576/65.807/2.852 ms
```

3.5. UART

IDO-EVB1126B-V1主板一共引出6路UART串口（不含调试串口），如下图所示：





UART串口设备节点列表如下：

序号	默认电平类型	设备节点
J13–25,J13–27	TTL	/dev/ttyS4
J3–16,J3–18,J3–20	RS485	/dev/ttyS5
J3–13,J3–15	TTL	/dev/ttyS6
J3–8,J3–10	TTL	/dev/ttyS7

使用microcom进行串口测试：

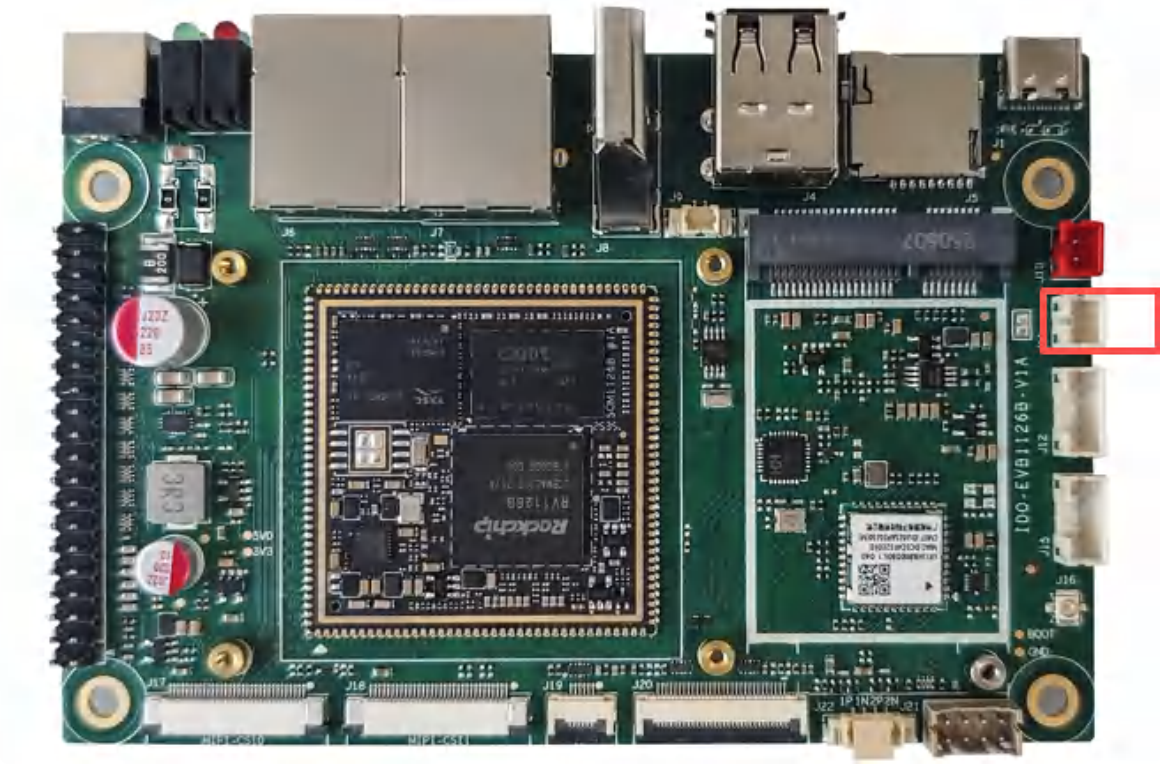
```Bash
root@linaro-alip:/# microcom /dev/ttyS7 -s 115200

```

注意：执行 Ctrl x 退出程序； RS485 需要使用转接板测试

3.6. Audio

3.6.1. Speaker

喇叭接口为PH2.0-P连接器J66，如下图所示：



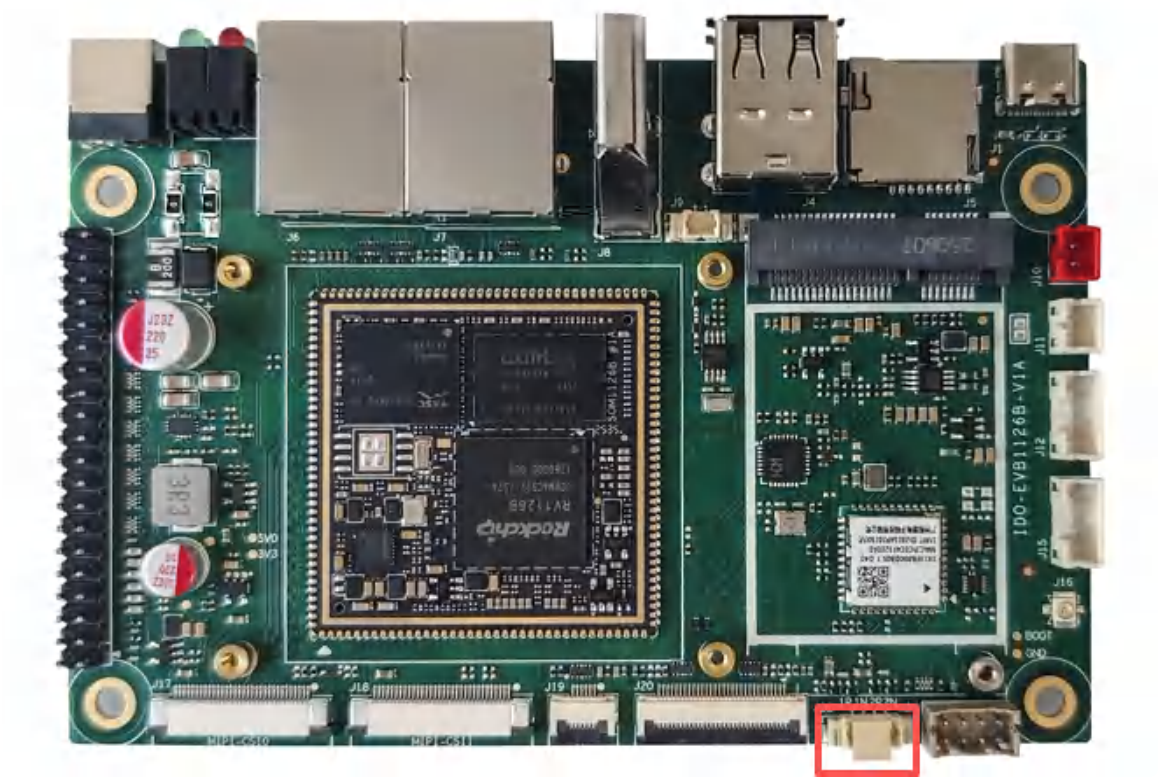
声卡节点：

```Bash
root@linaro-alip:/# aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: rockchiprv1126b [rockchip,rv1126b-acodec], device 0: dailink-multicodecs multicodec-0 [dailink-multicodecs multicodec-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
root@linaro-alip:/#
```

音频播放：

```Bash
root@linaro-alip:/# aplay -D hw:0,0  test.wav
```

3.6.2. MIC

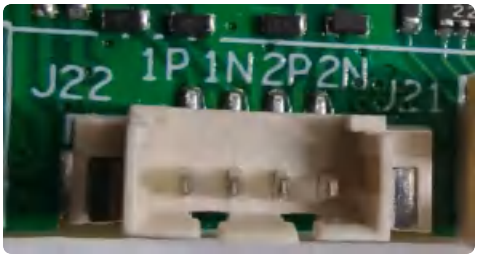


注意：录出来音频文件需要导出到PC 上播放

录音

```Bash
# 一共两路mic ， mic0 录音命令
arecord -D hw:0,0 -f S16_LE -r 16000 -c 4 -t wav test_mic0.wav
# mic1 录音命令
arecord -D hw:0,0 -f S16_LE -r 16000 -c 4 -t  wav test_mic1.wav
```

注意：录出来音频文件需要导出到PC 上播放

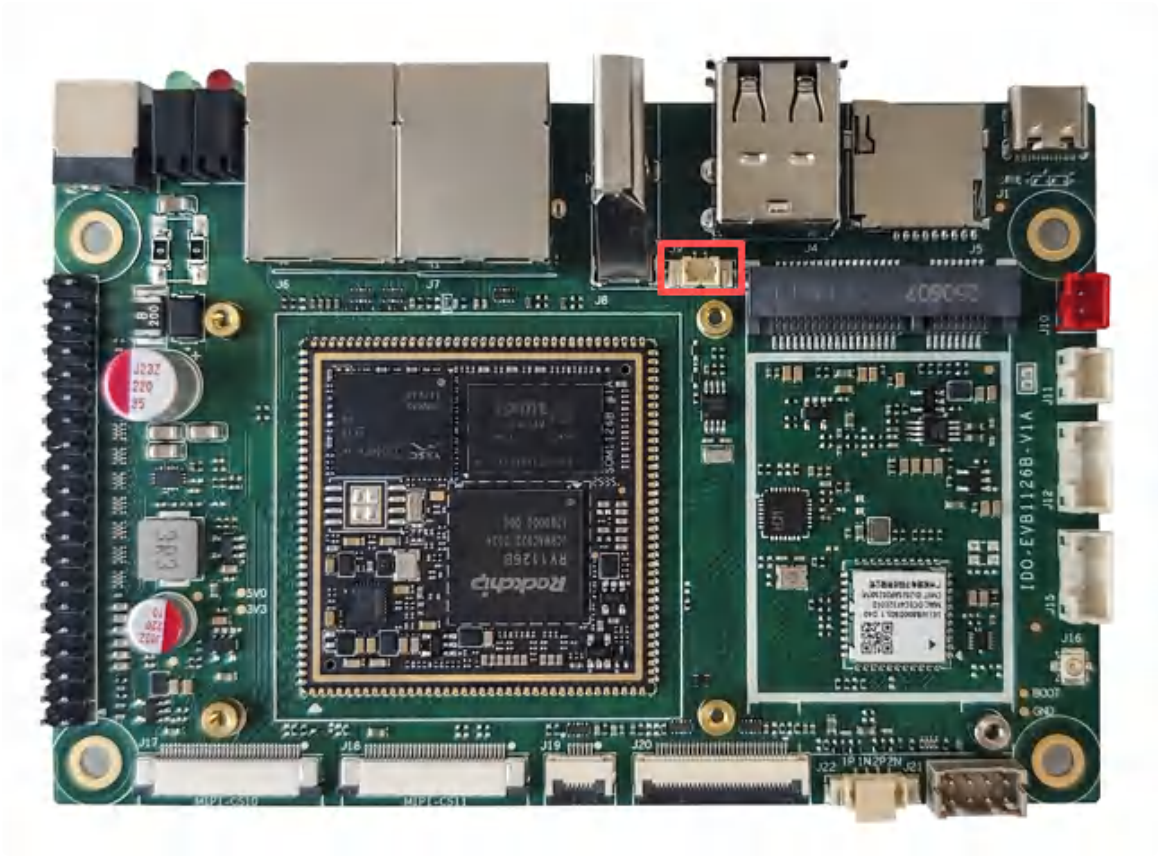


注意：如图上所示，2P,2N 是可以录制2line

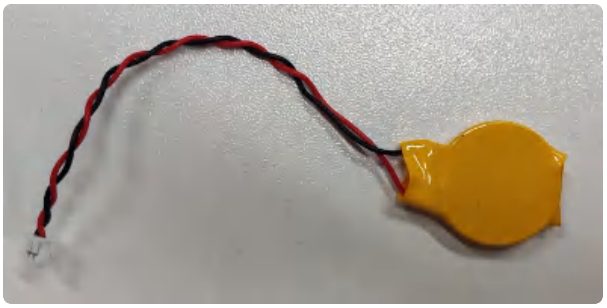
```Bash
# 录音
arecord -D hw:0,0 -f S16_LE -r 16000 -c 2 -t wav test_mic.wav
# 播放
aplay -D hw:0,0  test_mic.wav
```

3.7. RTC

MX1.25T-2P RTC电池座J9，如下图所示：



连接3V 纽扣电池，RTC电池参考如下



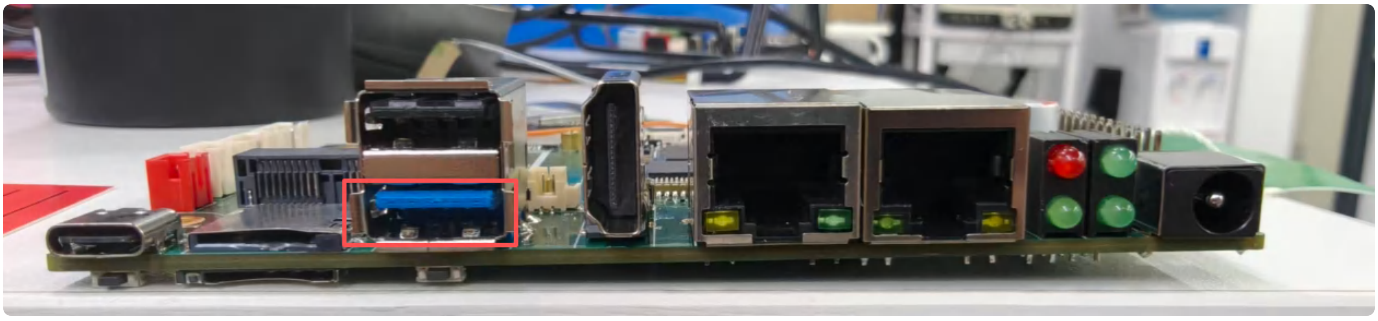
系统默认使用HYM8563作为系统时钟，设备节点：/dev/rtc0，时间设置方法：

```Bash
root@linaro-alip:/# hwclock -w      # 向rtc写时间
root@linaro-alip:/# hwclock -r      # 向rtc读时间
```

3.8. USB

主板支持1个TypeA接口（USB3.1 Gen1 OTG），支持1个USB3.0-otg接口，1个USB2.0PH2.0-4P 接口，一个USB2.0TypeA USB对外总供电应小于4A。

3.8.1. OTG



EVB1126B 已实现上电启动自动切换OTG

1. 上电启动前接入OTG 通过双公头usb连接 PC ，那么otg为device模式
2. 上电启动前接入U盘等usb设备，启动进入系统后otg为host模式

系统下otg模式切换命令：

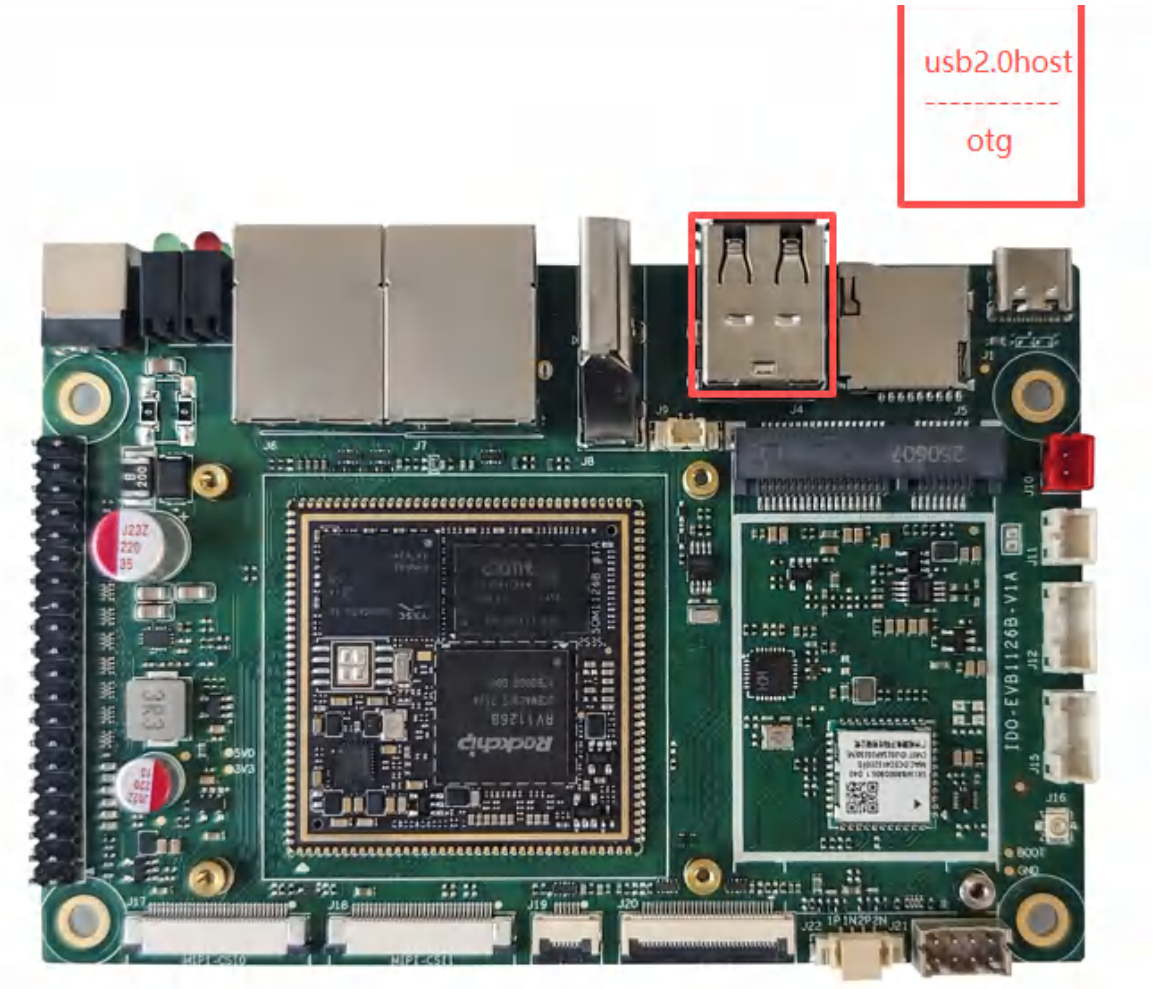
命令	模式
gpio-otg HOST	HOST 模式

gpio-otg DEVICE	DEVICE 模式

```bash
# host模式
root@linaro-alip:/# gpio-otg HOST
# device模式
root@linaro-alip:/# gpio-otg DEVICE
```

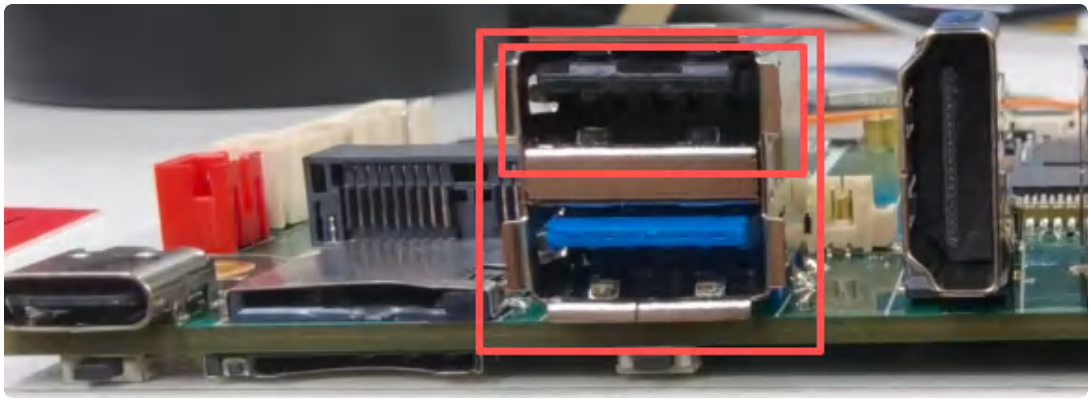
3.8.2. USB3.0

主板支持1个USB3.0-otg接口，接口为标准的双层A口J4，如下图所示：



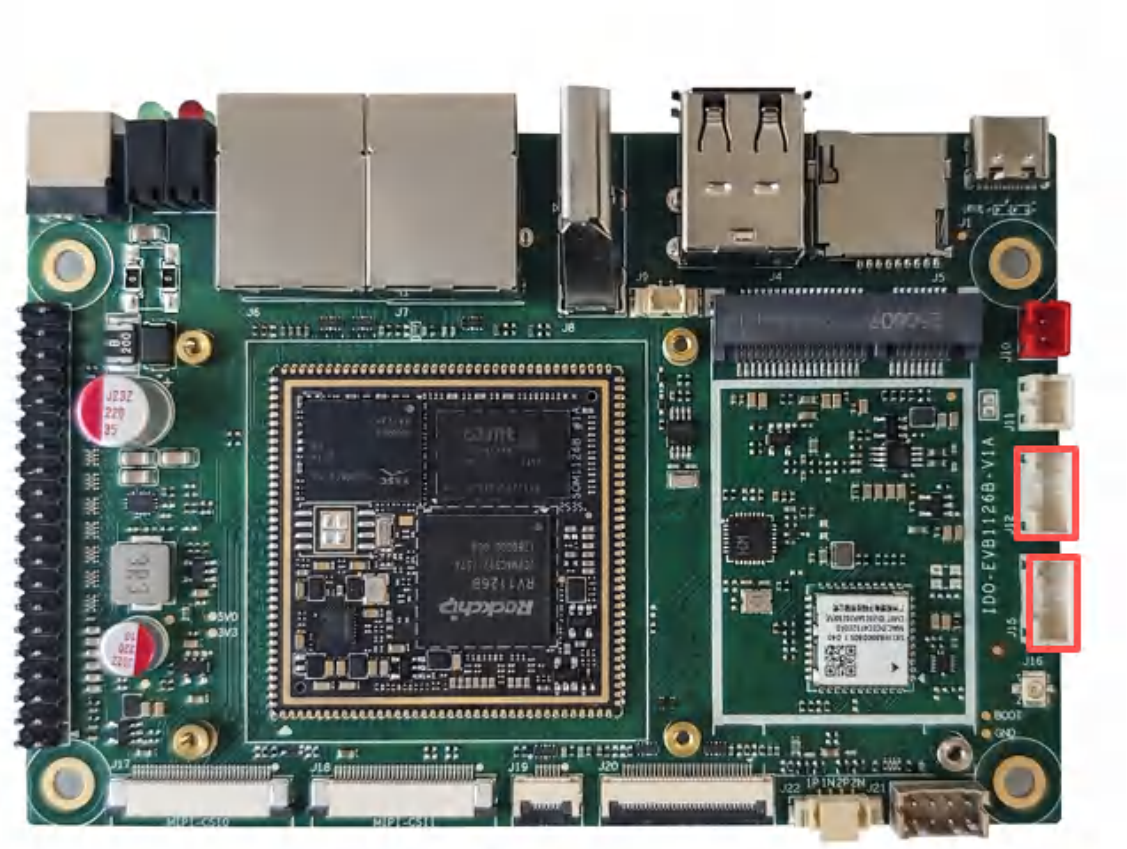
3.8.3. USB2.0-A

主板配置了1路USB2.0-A接口，如下图所示：



3.8.4. USB2.0-PH

主板配置了1路USB2.0 PH2.0-4P接口，USB接口均提供5V@1A的驱动能力，如下图所示：

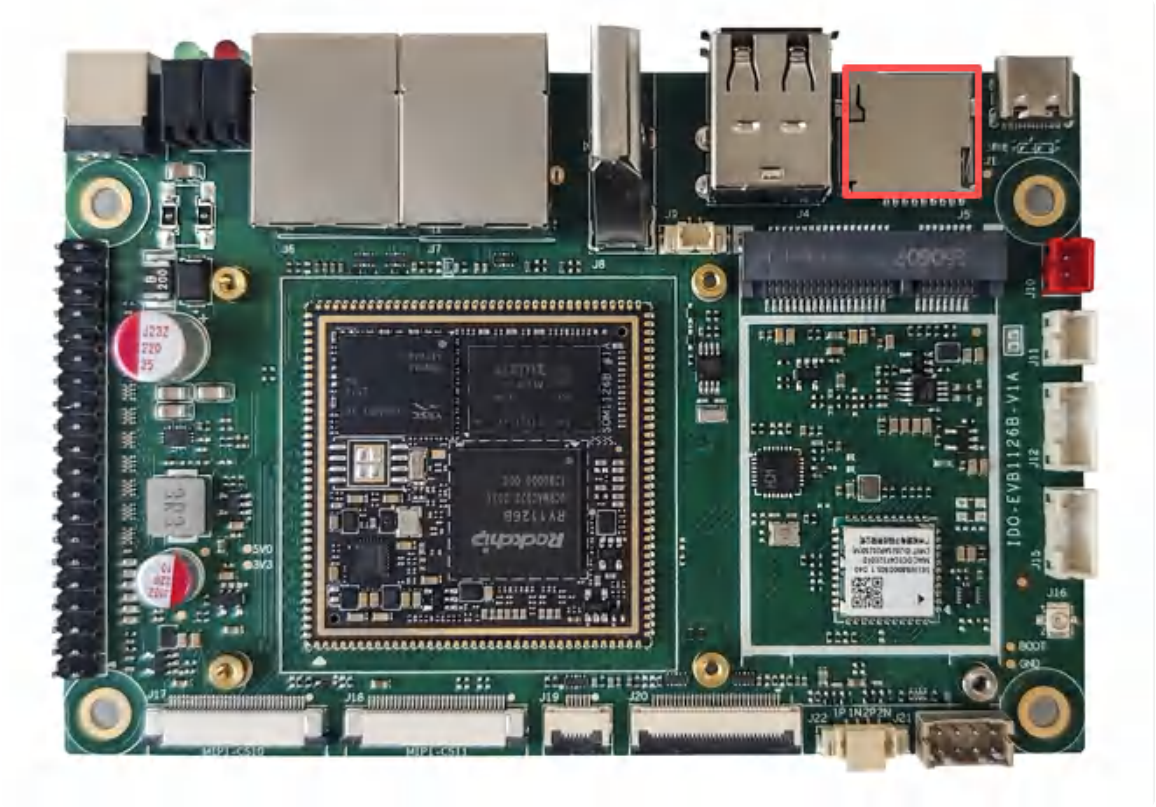


座子序号	控制节点
J12	/sys/class/leds/usb_vbus3_en/brightness

```bash
echo 1 > /sys/class/leds/usb_vbus3_en/brightness # 打开
echo 0 > /sys/class/leds/usb_vbus3_en/brightness # 关闭
```

3.9. TF Card

TF卡座支持SDIO3.0， 支持高速SD卡，接口位于主板背面J43，如下图所示：



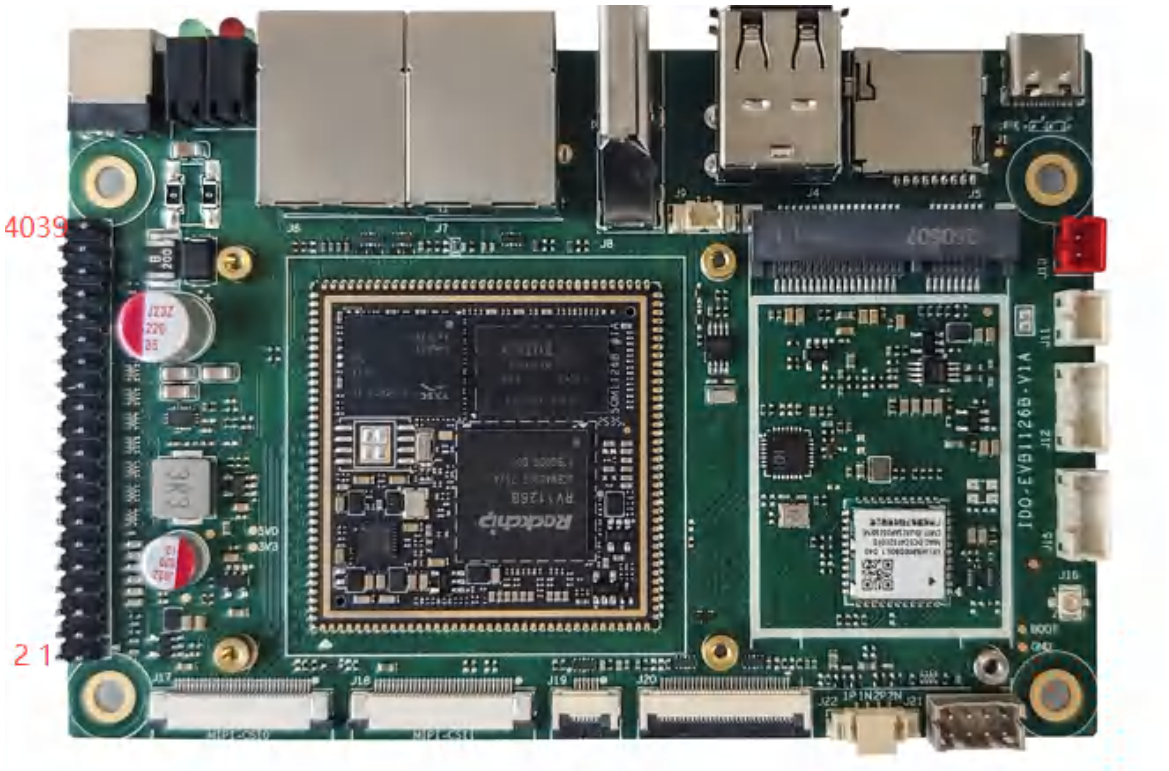
```
root@linaro-alip:/# fdisk -l
Disk /dev/ram0: 4 MiB, 4194304 bytes, 8192 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes


Disk /dev/mmcblk0: 116.48 GiB, 125074145280 bytes, 244285440 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: DB600000-0000-4377-8000-3FB20000370D

Device            Start       End   Sectors   Size Type
/dev/mmcblk0p1    16384     24575      8192     4M unknown
/dev/mmcblk0p2    24576     32767      8192     4M unknown
/dev/mmcblk0p3    32768    163839    131072    64M unknown
/dev/mmcblk0p4   163840    425983    262144   128M unknown
/dev/mmcblk0p5   425984    491519     65536    32M unknown
/dev/mmcblk0p6   491520  13074431  12582912     6G unknown
/dev/mmcblk0p7 13074432  13336575    262144   128M unknown
/dev/mmcblk0p8 13336576 244285406 230948831 110.1G unknown


Disk /dev/mmcblk0boot0: 4 MiB, 4194304 bytes, 8192 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/mmcblk0boot1: 4 MiB, 4194304 bytes, 8192 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/mmcblk1: 14.56 GiB, 15634268160 bytes, 30535680 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x00000000

root@linaro-alip:/# mount /dev/mmcblk1 /mnt
```

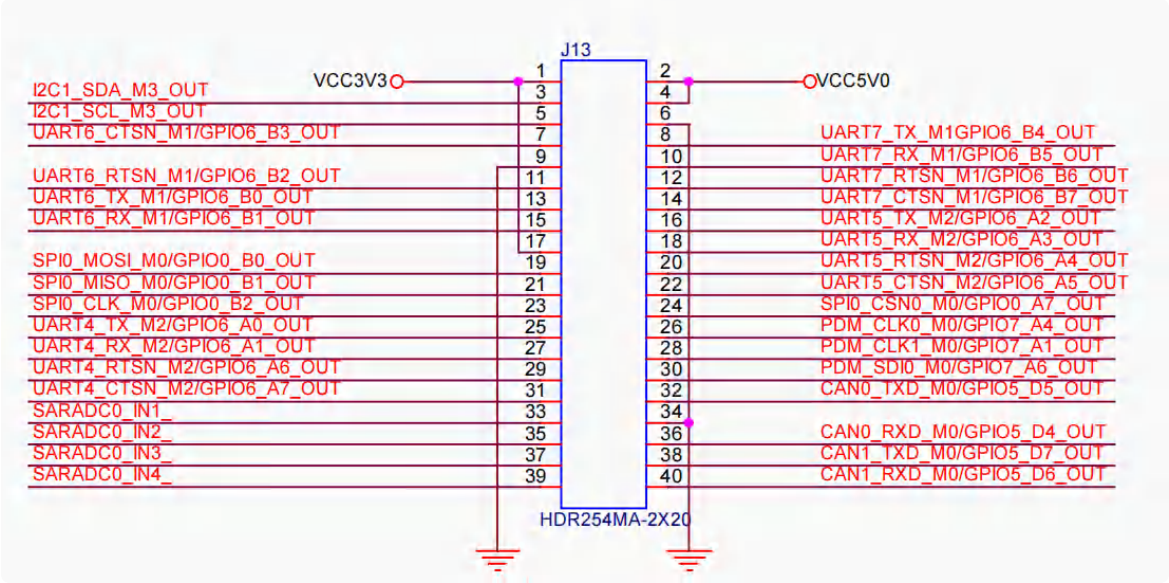
```
root@linaro-alip:/# df
Filesystem     1K-blocks    Used Available Use% Mounted on
/dev/root        3021512 2575920    271924  91% /
devtmpfs          981636       0    981636   0% /dev
tmpfs            1009988       0   1009988   0% /dev/shm
tmpfs             403996    1636    402360   1% /run
tmpfs               5120       8      5112   1% /run/lock
tmpfs             201996      24    201972   1% /run/user/1000
tmpfs             201996      20    201976   1% /run/user/103
/dev/mmcblk1    15267836 2114500  13153336  14% /mnt
root@linaro-alip:/#
```

默认挂在路径为 /media/mmcblk*

3.10. CAN

主板共有2路CAN，如下图所示:



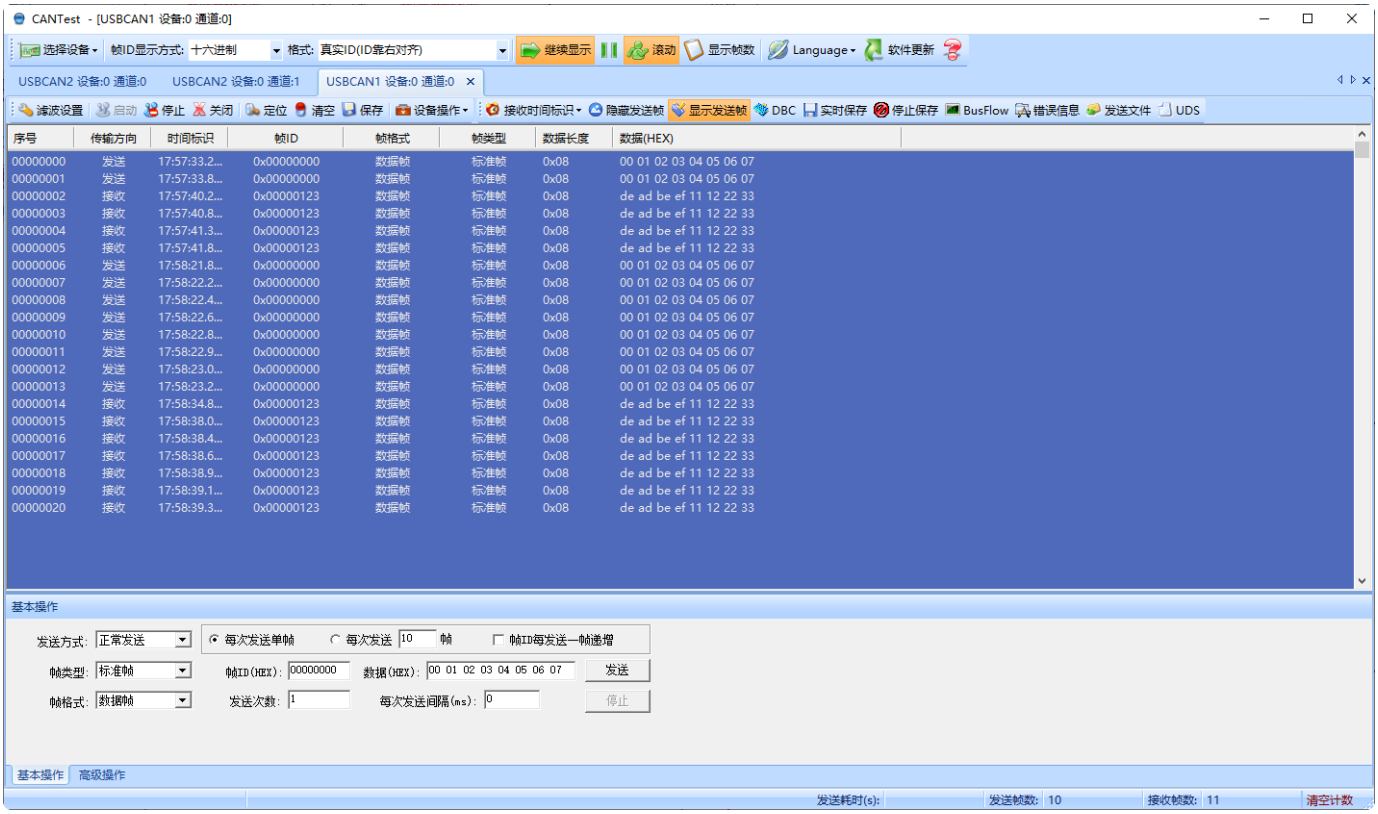


CAN	接线
can0	J13-32 ,J13-34
can1	J13-38 ,J13-40

分别对应系统节点名称为 can0、can1。使用can盒测试：

1. 设置CAN参数

```bash
# can0节点发送数据
ip link set can0 down
ip link set can0 type can bitrate 1000000
ip link set can0 up
cansend can0 123#DEADBEEF
cansend can0 123#DEADBEEF11122233

# can0接收数据
ip link set can0 down
ip link set can0 type can bitrate 1000000
ip link set can0 up
candump can1

 # can1 接收数据
ip link set can1 down
ip link set can1 type can bitrate 1000000
ip link set can1 up
```




```
root@rv1126b-buildroot:/# candump can0
  can0  000   [8]  00 01 02 03 04 05 06 07
  can0  000   [8]  00 01 02 03 04 05 06 07
  can0  000   [8]  00 01 02 03 04 05 06 07
^Croot@rv1126b-buildroot:/# ^C
root@rv1126b-buildroot:/# ^C
root@rv1126b-buildroot:/# ^C
root@rv1126b-buildroot:/# cansend can0 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can0 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can0 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can0 123#DEADBEEF11122233
```

```
root@rv1126b-buildroot:/# candump can1
  can1  000   [8]  00 01 02 03 04 05 06 07
  can1  000   [8]  00 01 02 03 04 05 06 07
  can1  000   [8]  00 01 02 03 04 05 06 07
  can1  000   [8]  00 01 02 03 04 05 06 07
  can1  000   [8]  00 01 02 03 04 05 06 07
  can1  000   [8]  00 01 02 03 04 05 06 07
  can1  000   [8]  00 01 02 03 04 05 06 07
  can1  000   [8]  00 01 02 03 04 05 06 07
^Croot@rv1126b-buildroot:/# ^C
root@rv1126b-buildroot:/# ^C
root@rv1126b-buildroot:/# ^C
root@rv1126b-buildroot:/# cansend can0 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can1 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can1 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can1 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can1 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can1 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can1 123#DEADBEEF11122233
root@rv1126b-buildroot:/# cansend can1 123#DEADBEEF11122233
```

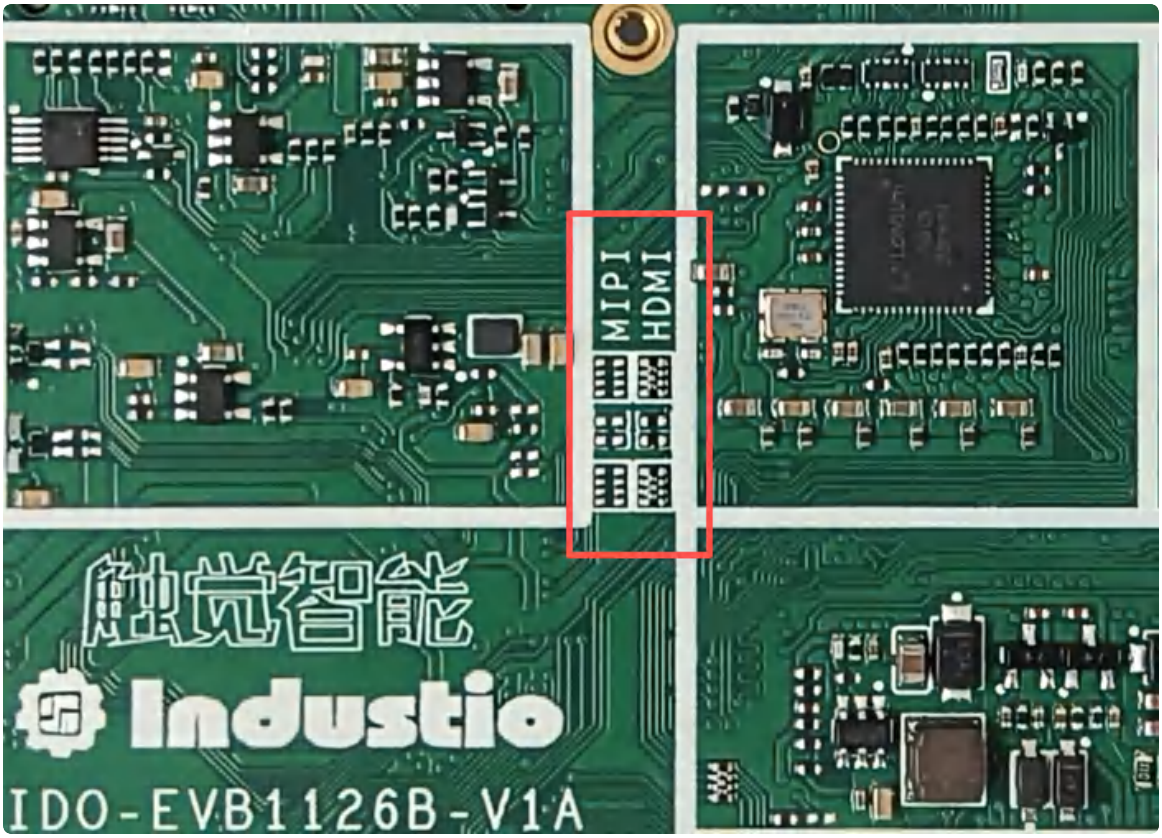
注意： 测试需要使用转接板。

3.11. LCD显示

3.11.1. HDMI-TX

注意：HDMI-TX 是由 MIPI-TX 接口 + 信号转换芯片 ， 二选一，硬件默认配置为 HDMI-TX。

HDMI-TX硬件确认，如下图所示：



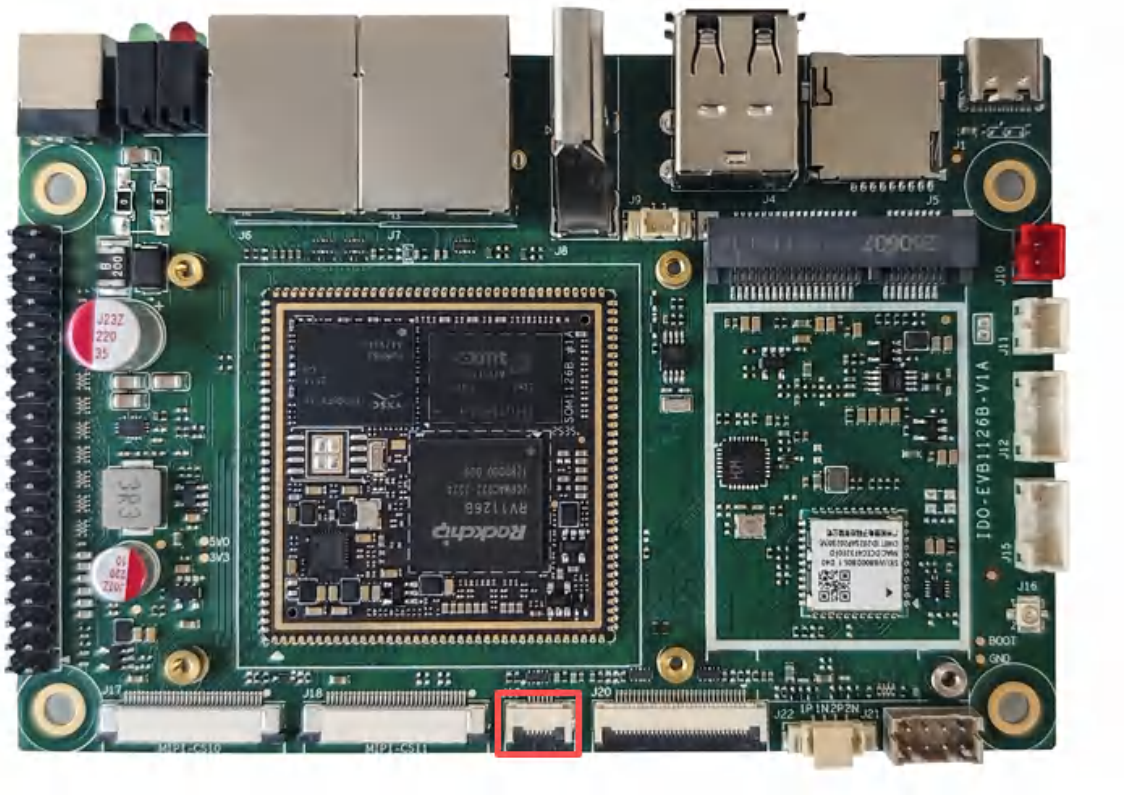
3.11.2. MIPI–TX

注意：MIPI–TX 与 HDMI 为复用关系，二选一，硬件默认配置为 HDMI。

要使用mipi的话，需要将下图中HDMI 列电阻移动至MIPI 列

3.12. TP

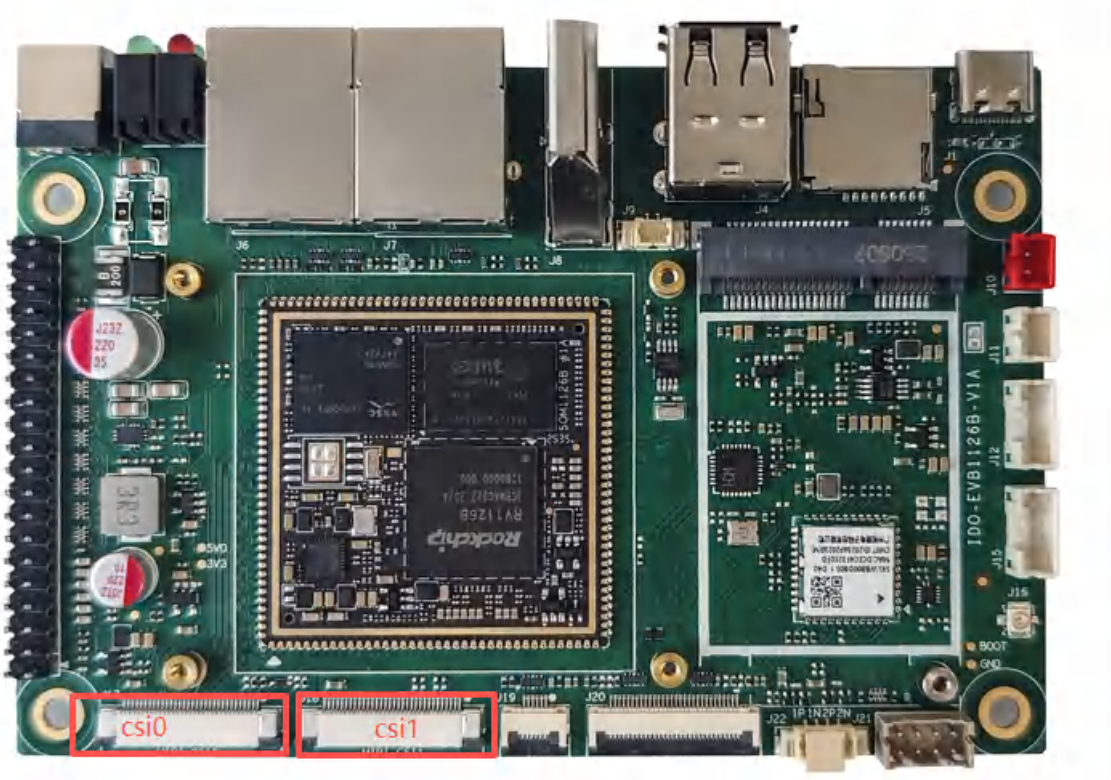
I2C–TP接口为 6Pin FPC 0.5mm座子 (J13)，如下图所示：



- 触摸 TP 接口接线方法为下接。

3.13. MIPI Camera

主板内置两路MIPI Camer接口J17和J18，使用FPC05-PUW-32P抽拉式上接座子，如下图所示：



3.13.1. IMX415

J18 默认适配IMX415作为后置摄像头，接法如下图所示：

同时还支持的camera senor 有SC450AI,SC850SL,OV13855

J17	J18
/dev/video22	/dev/video30

3.13.1.1. 预览

```bash
gst-launch-1.0 v4l2src device=/dev/video22 ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! videoconvert ! autovideosink

gst-launch-1.0 v4l2src device=/dev/video30 ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! videoconvert ! autovideosink
```

3.13.1.2. 抓图

```bash
v4l2-ctl --verbose -d /dev/video22 --set-fmt-video=width=1920,height=1080,pixelformat='NV12' --stream-mmap=4 --stream-skip=3  --stream-to=./test-1920x1080-p50.yuv --stream-count=1 --stream-poll
```

3.13.1.3. 抓视频

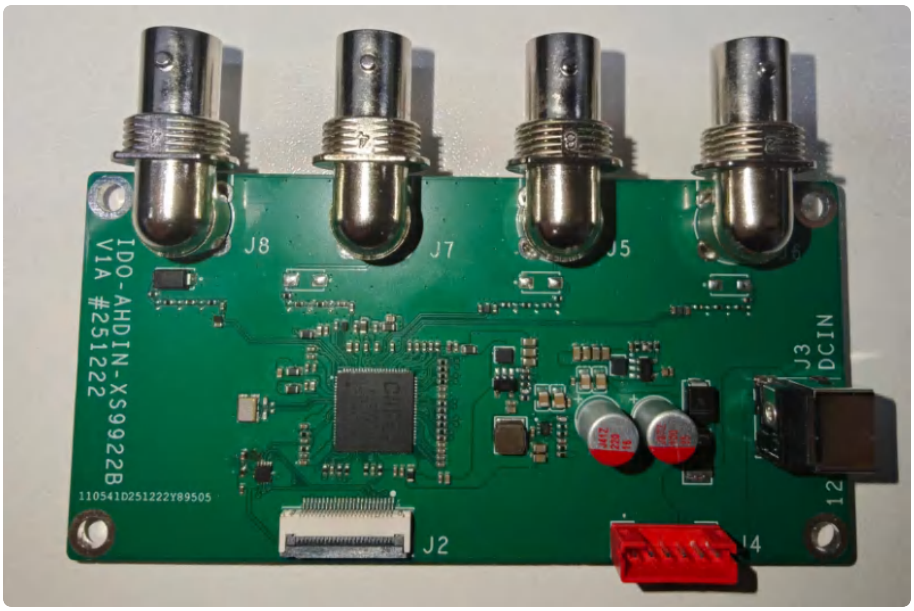
```bash
v4l2-ctl --stream-mmap=4  -d /dev/video22 --set-fmt-video=width=1920,height=1080,pixelformat='NV12' --stream-to=./output.yuv
```

3.13.2. TP2815/XS9922 AHD

TP2815


IDO-AHDIN-V1C

XS9922


IDO-AHDIN-XS9922B
V1A #251222
110541D251222Y89505
J8
J7
J5
J3
DCIN
J2
J4

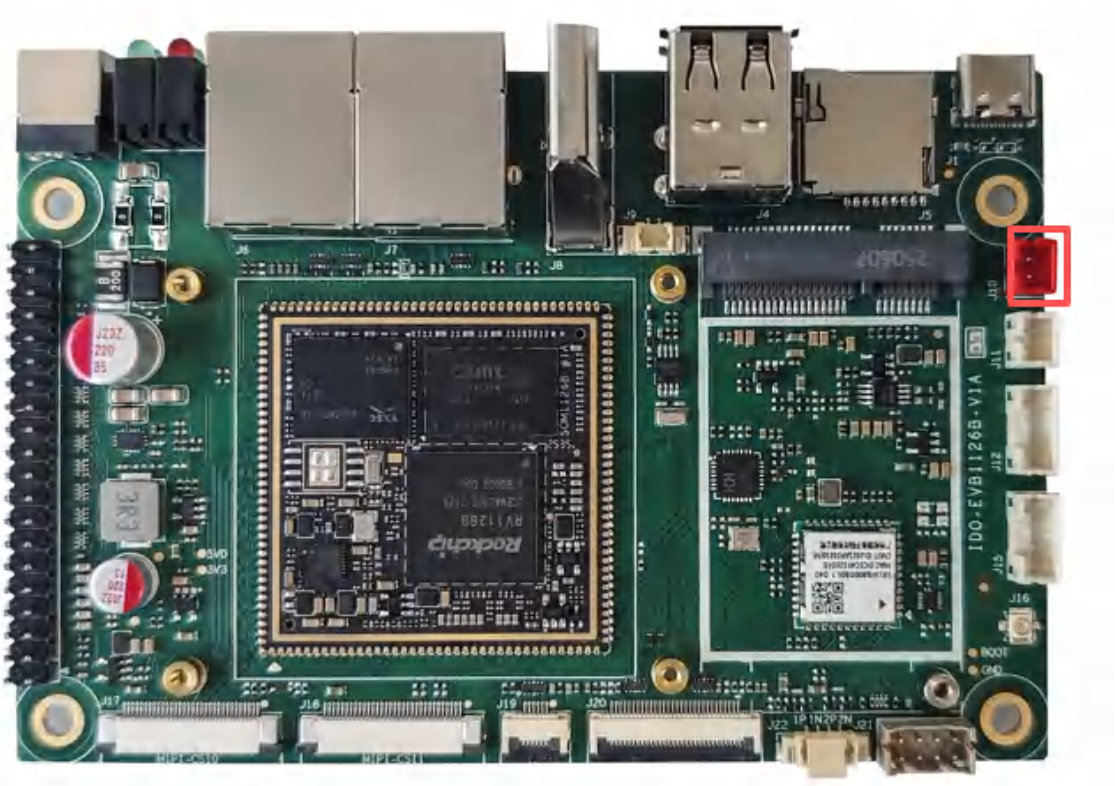
J17	J18
/dev/video0	/dev/video11
/dev/video1	/dev/video12
/dev/video2	/dev/video13
/dev/video3	/dev/video14

预览

```bash
gst-launch-1.0 v4l2src device=/dev/video0 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink


gst-launch-1.0 v4l2src device=/dev/video1 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink


    gst-launch-1.0 v4l2src device=/dev/video2 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink


    gst-launch-1.0 v4l2src device=/dev/video3 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink


    gst-launch-1.0 v4l2src device=/dev/video11 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink

    gst-launch-1.0 v4l2src device=/dev/video12 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink


    gst-launch-1.0 v4l2src device=/dev/video13 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink


    gst-launch-1.0 v4l2src device=/dev/video14 ! video/x-raw, format=NV12, width=1280, height=720, framerate=25/1 ! videoconvert ! autovideosink

```

3.14. FAN 风扇

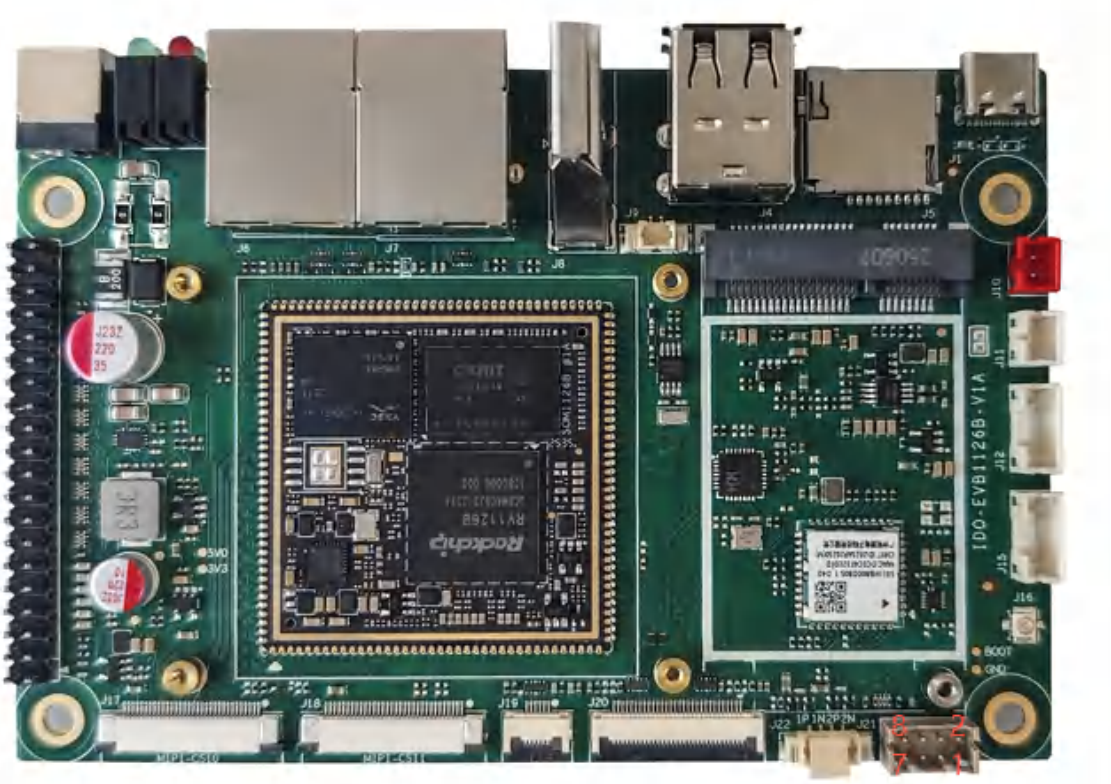
PH2 5V FAN风扇接口J3，如下图所示：

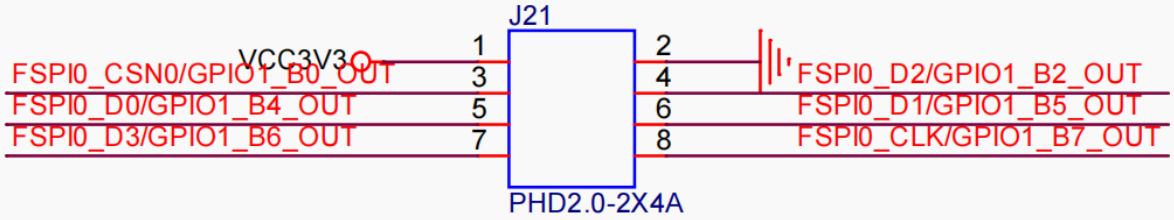


控制功能	控制命令
打开风扇	echo 1 > /sys/class/leds/fan_led/brightness
关闭风扇	echo 0 > /sys/class/leds/fan_led/brightness

3.15. GPIO

主板预留了8个GPIO接口J3，如下图所示：

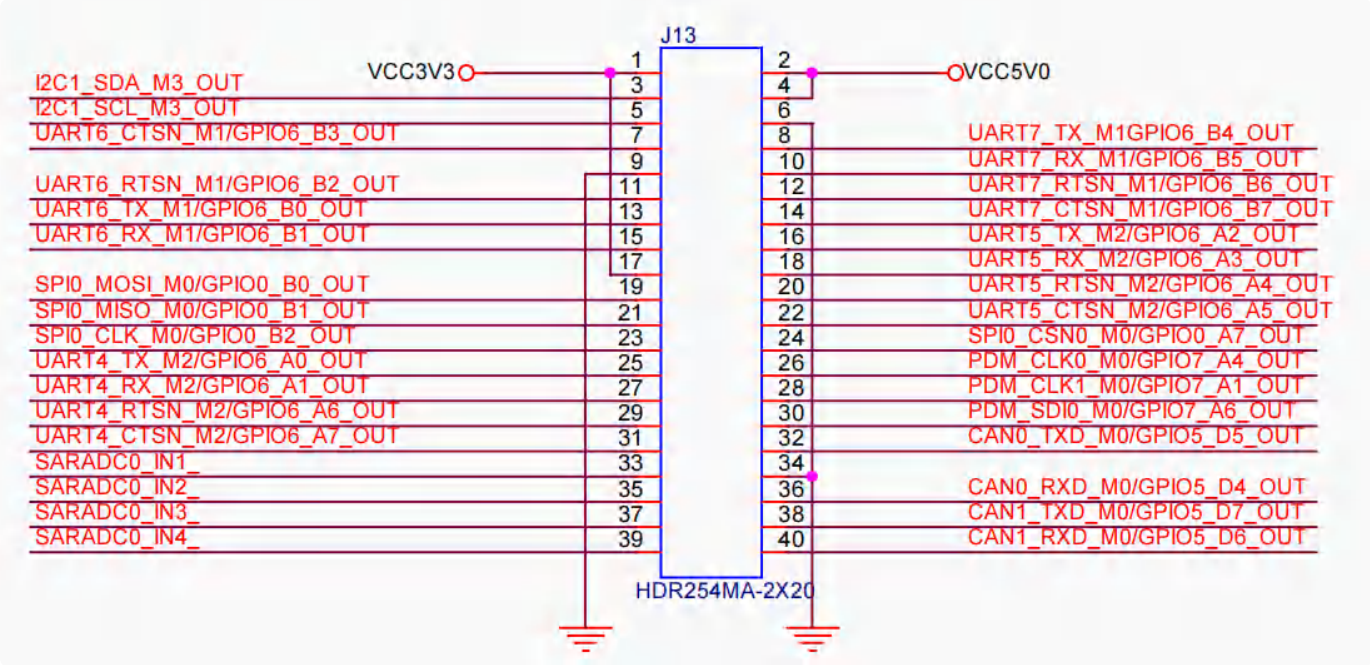




GPIO引脚定义描述，如下图所示：

序号	管脚标号	GPIO 序号
1	VCC3V3	/
2	GND	/
3	GPIO1_B0	40
4	GPIO1_B2	42
5	GPIO1_B4	44
6	GPIO1_B5	45
7	GPIO1_B6	46

8	GPIO1_B7	47



序号	管脚标号	GPIO 序号
7	GPIO6_B3	203
11	GPIO6_B2	202
19	GPIO0_B0	8
21	GPIO0_B1	9
23	GPIO0_B2	10
29	GPIO6_A6	198
31	GPIO6_A7	199
12	GPIO6_B6	206
14	GPIO6_B7	207
22	GPIO6_A5	197
24	GPIO0_A7	7
26	GPIO7_A4	228

28	GPIO7_A1	225
30	GPIO7_A6	230

执行以下脚本，使用万用表实际去量电压值，是否是高电平

test.sh Bash

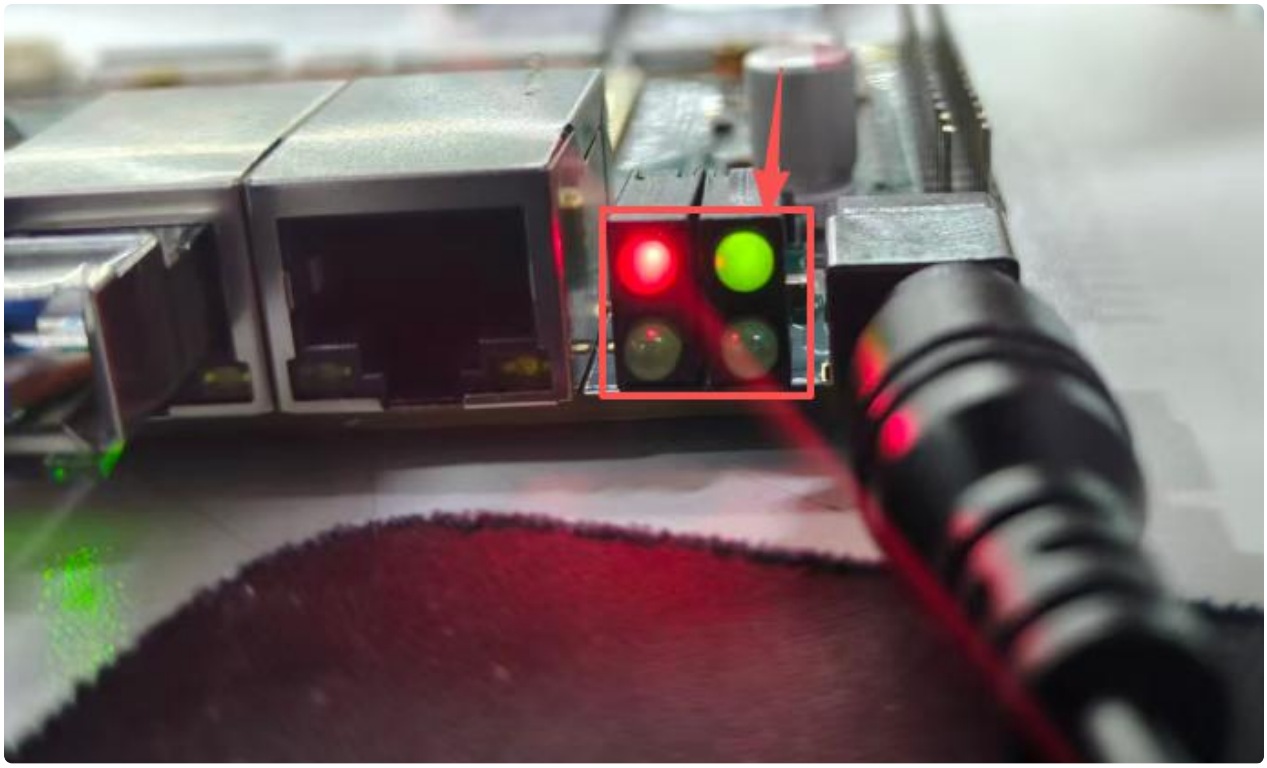
```bash
#!/bin/bash

GPIO_NUM=(40 42 44 45 46 47  203 202 8 9 10 198 199 206 207 197 7 228 225 230)

for gpio in  "${GPIO_NUM[@]}"; do
    echo "$gpio" > /sys/class/gpio/export 2>/dev/null
    if [ $? -eq 0 ]; then
        echo "GPIO $gpio exported successfully."
         echo out >  /sys/class/gpio/gpio$gpio/direction
        echo 1 >  /sys/class/gpio/gpio$gpio/value
        echo "GPIO $gpio Out High"
    else
        echo "GPIO $gpio already exported or failed to export."
    fi
    sleep 0.1
done
```

执行以下脚本，使用万用表实际去量电压值，是否是低电平

test.sh Bash

```bash
#!/bin/bash

GPIO_NUM=(40 42 44 45 46 47  203 202 8 9 10 198 199 206 207 197 7 228 225 230)

for gpio in "${GPIO_NUM[@]}"; do
        echo 0 >  /sys/class/gpio/gpio$gpio/value
        echo "GPIO $gpio Out Low"

done
```

3.16. pwm



pwm灯测试：

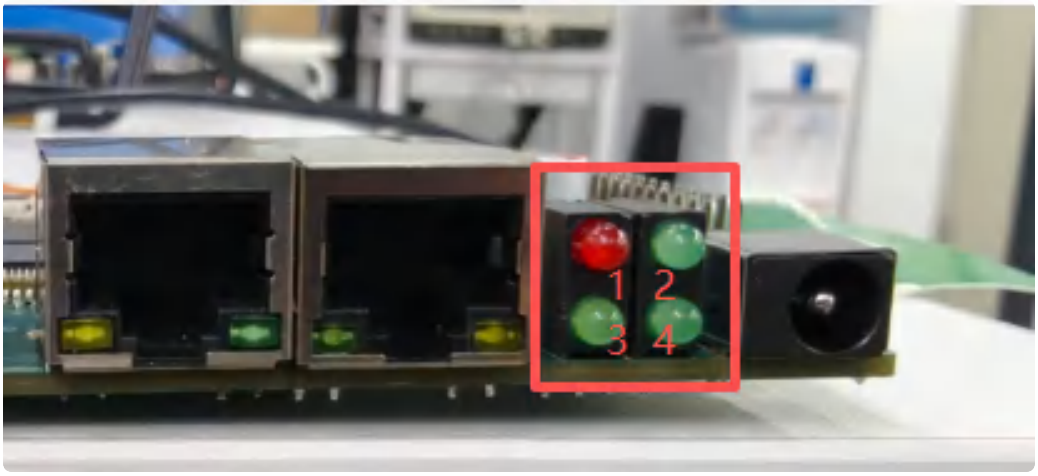
HDMI的 版本镜像pwm节点是 pwmchip1

```Bash
cd /sys/class/pwm/pwmchip1/
echo 0 > export
cd pwm0
echo 10000 > period
echo 5000 > duty_cycle
echo normal > polarity
echo 1 > enable
```

MIPI 的版本镜像 pwm节点是 pwmchip2

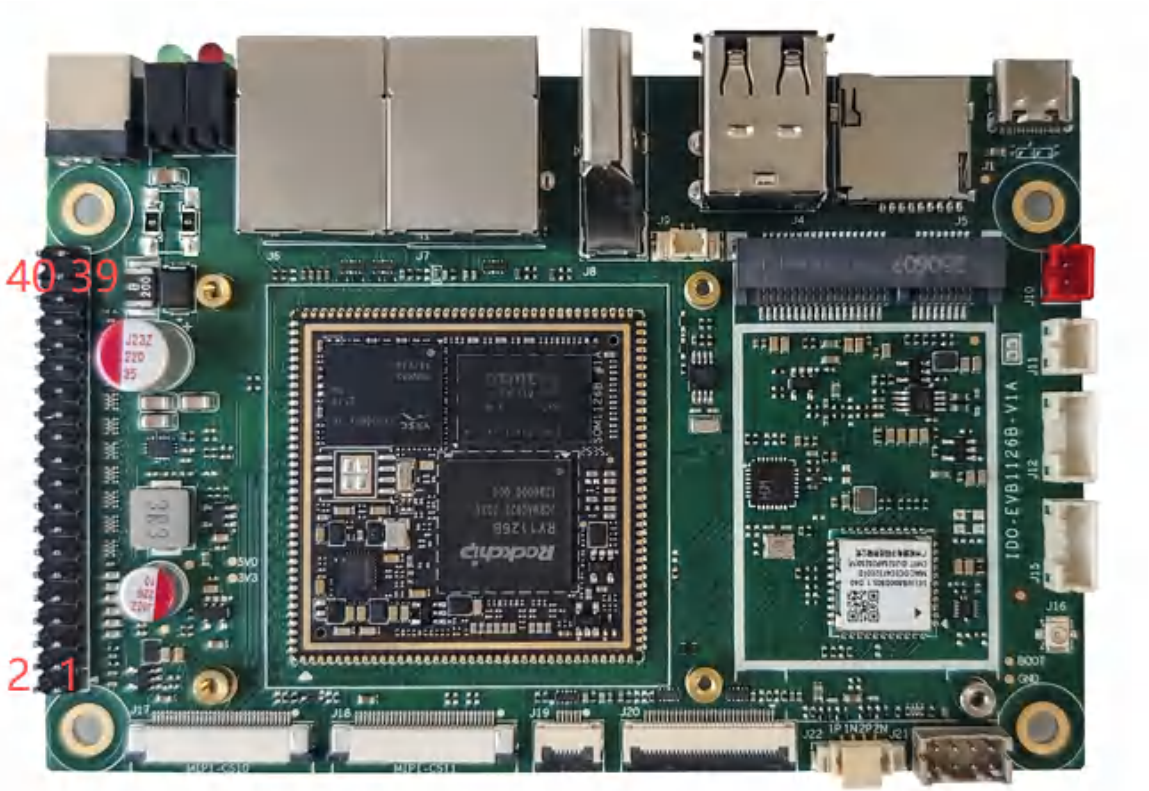
```Bash
cd /sys/class/pwm/pwmchip2/
echo 0 > export
cd pwm0
echo 10000 > period
echo 5000 > duty_cycle
echo normal > polarity
echo 1 > enable
```

3.17. LED



1. 电源指示灯
2. PWM–LED 灯
3. 系统运行指示灯
4. 4G模块指示灯

3.18. ADC



序号	ADC通道	读值
33	SARADC0_IN1_	cat /sys/bus/iio/devices/iio\:device0/in_voltage1_raw
35	SARADC0_IN2_	cat /sys/bus/iio/devices/iio\:device0/in_voltage2_raw
37	SARADC0_IN3_	cat /sys/bus/iio/devices/iio\:device0/in_voltage3_raw
39	SARADC0_IN4_	cat /sys/bus/iio/devices/iio\:device0/in_voltage4_raw

4路ADC精度为13位，参考电压为1.8V。

ADC输入电压计算公式：

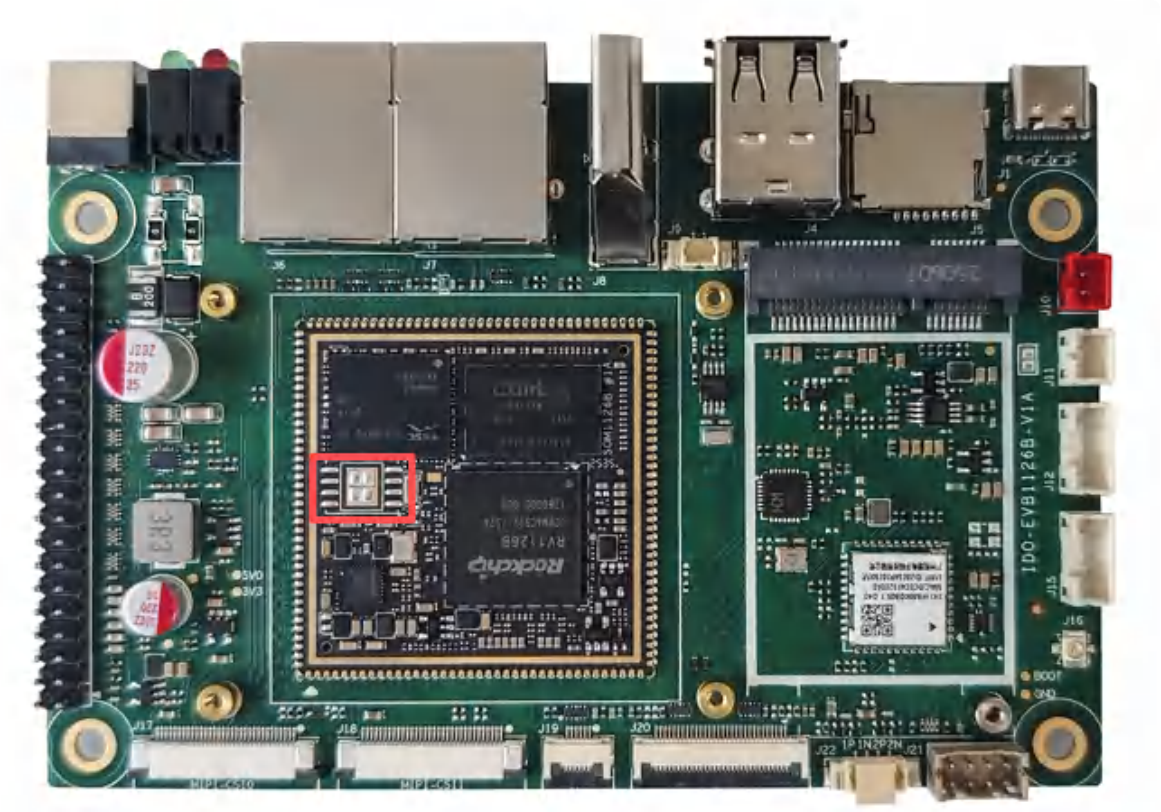
$$V=(raw/8191)*1.8v$$

3.19. I2C

```
root@linaro-alip:/#  i2cdetect -y -r 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:                         -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- 51 -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
root@linaro-alip:/#
```

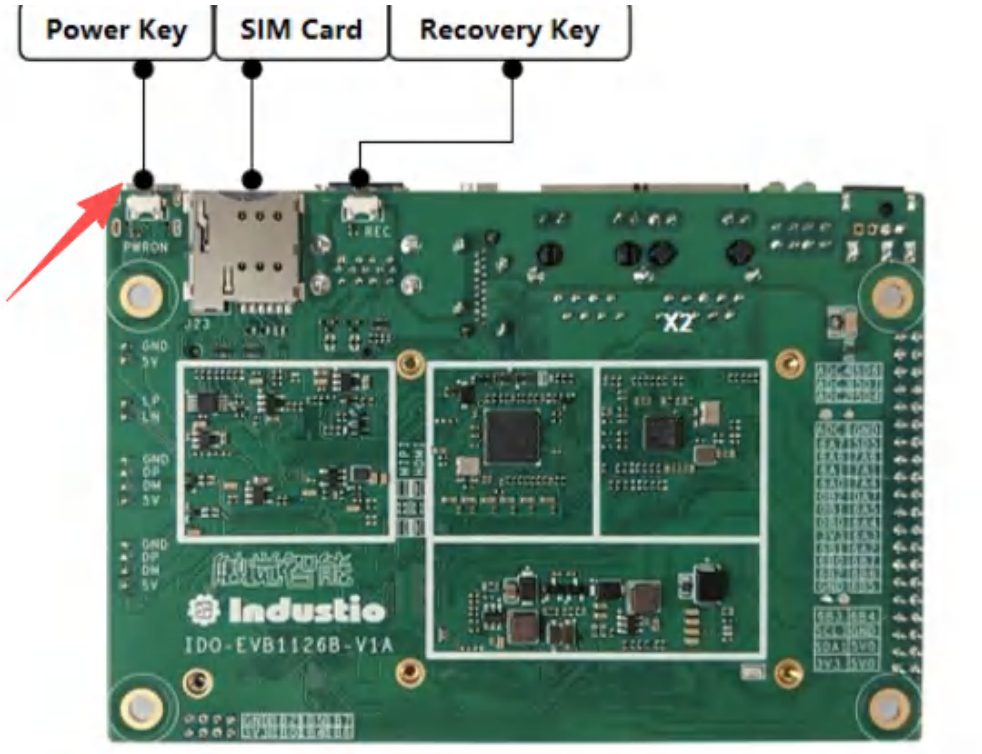
注意： 测试需要使用转接板。

3.20. SPI NAND



```
root@rv1126b-buildroot:/# dmesg | grep spi
[    2.847504] spi-nand spi2.0: Winbond SPI NAND was found.
[    2.847513] spi-nand spi2.0: 512 MiB, block size: 128 KiB, page size: 2048, OOB size: 128
[    3.653787] mtdblock: MTD device 'spi-nand0' is NAND, please consider using UBI block devices instead.
[    4.464264] mtdblock: MTD device 'spi-nand0' is NAND, please consider using UBI block devices instead.
root@rv1126b-buildroot:/#
```

3.21. PWRON



按键	关机/开机	休眠/唤醒
PWRON	长按5S 关机/单击开机	单击休眠/单击唤醒

3.22. NPU

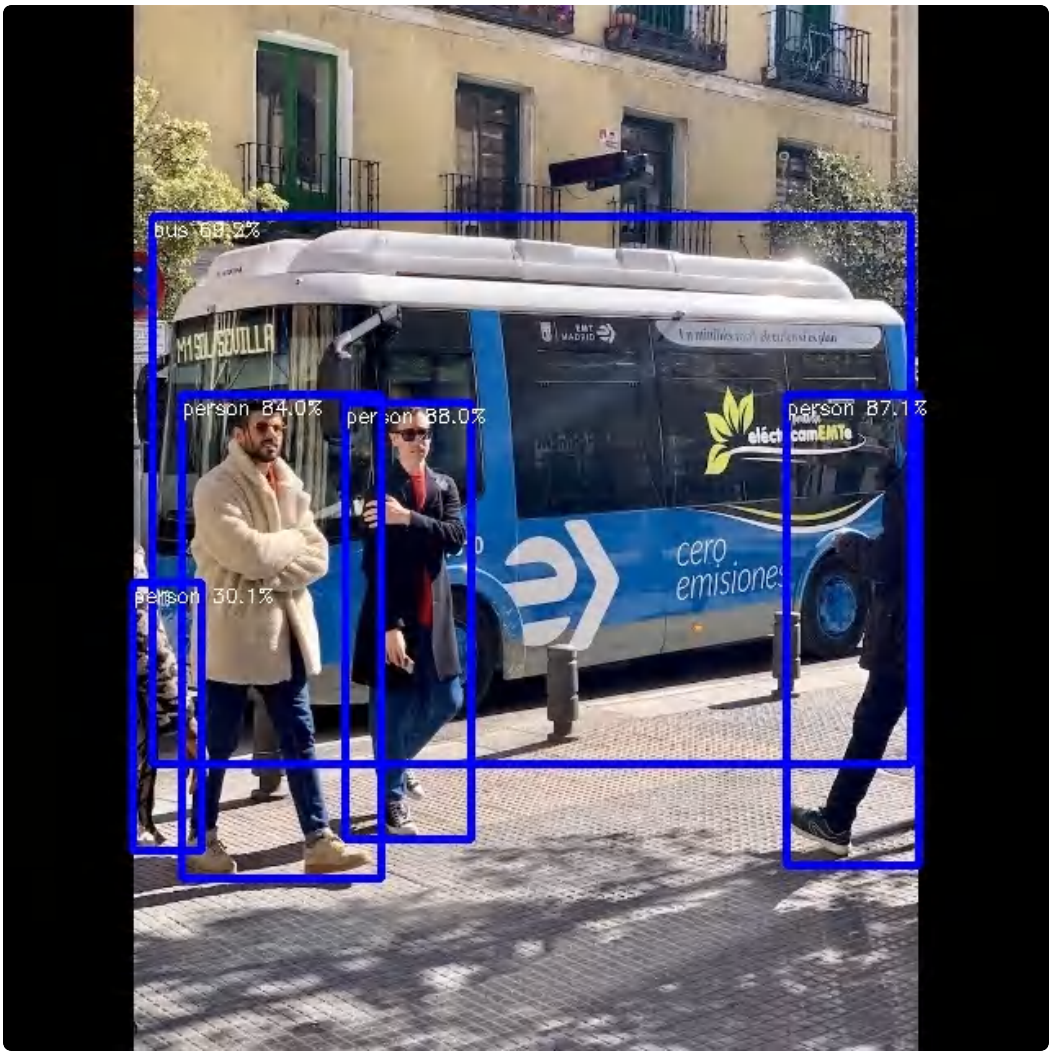
3.22.1. yolov5测试：

3.22.1.1. rknn_yolov5_demo：

通过网盘分享的文件：rknn_yolov5_demo_Linux.zip

链接: https://pan.baidu.com/s/1N-vgqd1eH1Xcd7zxpfhZ4g 提取码: hnpm

执行完后会得到转换后的模型图片，如下所示：



3.22.1.2. rknn_yolov5_video_demo：

Shell

```
root@rv1126b-buildroot:/rknn_yolov5_demo_Linux#
./rknn_yolov5_video_demo ./model/RV1126B/yolov5s-640-640.rknn ./1280x720.h264 264

```

最终生成 out.h264

```shell
# 播放测试
gst-play-1.0 out.h264
```

3.22.2. yolo11 测试

测试 摄像头+yolo11

yolo11_videocapture_demo ， yolo11_videocapture_demo_zero_copy

摄像头节点 22，30

```bash
# 22 为 /dev/video22 也可改成 30
yolo11_videocapture_demo /vendor/etc/firmware/yolo11n.rknn 22
# 或者
yolo11_videocapture_demo_zero_copy /vendor/etc/firmware/yolo11n.rknn 22
```