

IDO-EVB3588S-V1 Linux使用手册



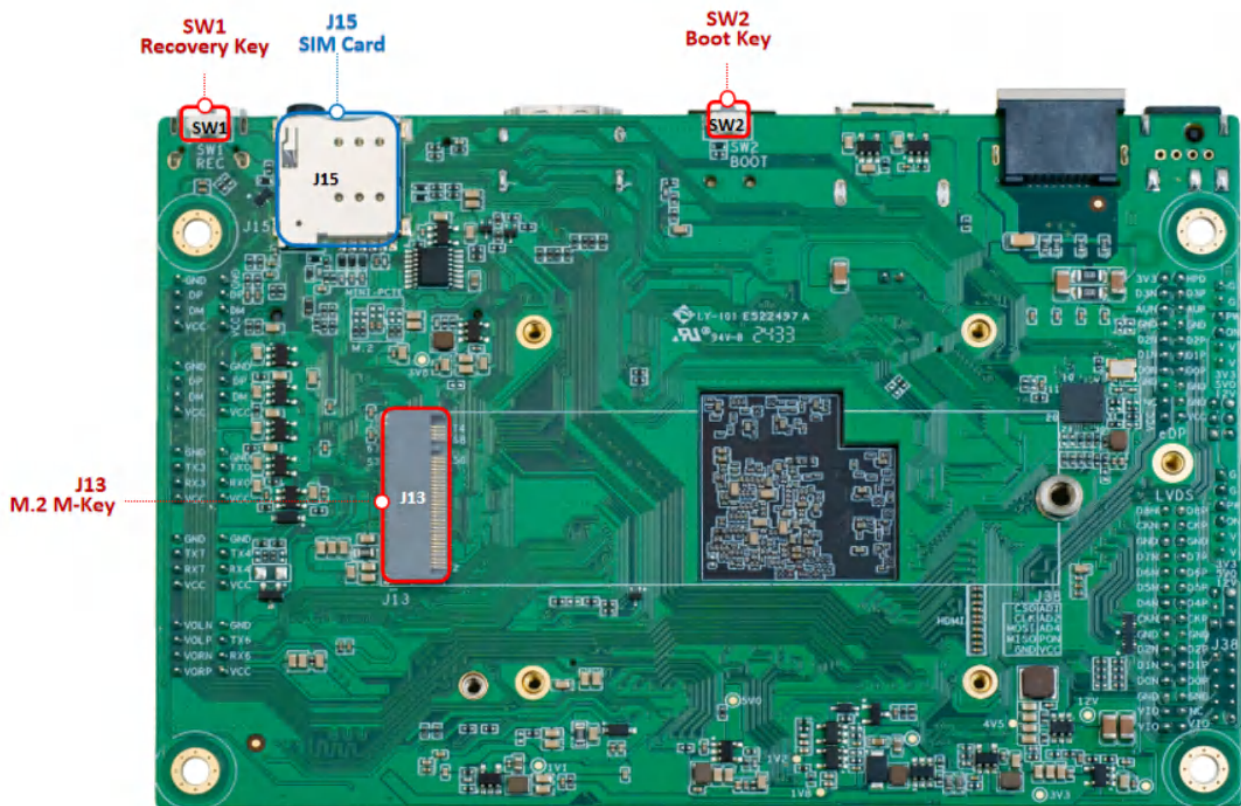
IDO-EVB3588S-V1 Linux使用手册

深圳触觉智能科技有限公司

www.industio.cn

文档修订历史

版本	修订内容	修订	审核	日期
V1.0	创建文档	MHK	IDO	2025/08/14



IDO-EVB3588S-V1C 背面实物图

1.2. 硬件资源

序号	名称	描述
1	内核版本	6.1.118
2	系统版本	Buildroot2024,Debian11,Ubuntu22.04
3	内存	LPDDR4, 8GB
4	存储	eMMC, 128GB
5	供电	默认12V/2A供电

6	显示	1x HDMI2.1接口, 支持 (8K/60fps) 输出 1x USB-C DP 接口, 支持 (8K/30fps) 输出 1x MIPI DSI接口, 支持4k@60fps输出 1x Dual LVDS接口, 支持1920x1080@60fps输出 1x eDP 接口, 支持 4K@60fps 输出
7	背光	eDP-backlight x1 LVDS-backlight x1
8	TP触摸	I2C-TP x1
9	USB OTG	USB-C3.0 x1
10	USB HOST	USB2.0 HOST(USB-A) x 1 USB3.0 HOST(USB-A) x 1 USB2.0 HOST(PH2.0-4A) x 4
11	TF Card	TF Card x 1
12	以太网	千兆以太网 x 1
13	WIFI/BT	AP6256
14	扬声器	4Pin 2.0 x1
15	耳机	3.5mm (CTIA) x1
16	Camera	OV13855 x 1 IMX415 x 2
17	串口	TTL x 2 RS232 x 2 RS485 x 1
18	调试串口	TTL x 1
19	RTC	HYM8563S
20	LED	电源指示灯 x 1
21	4G/5G	EC20、RG200U

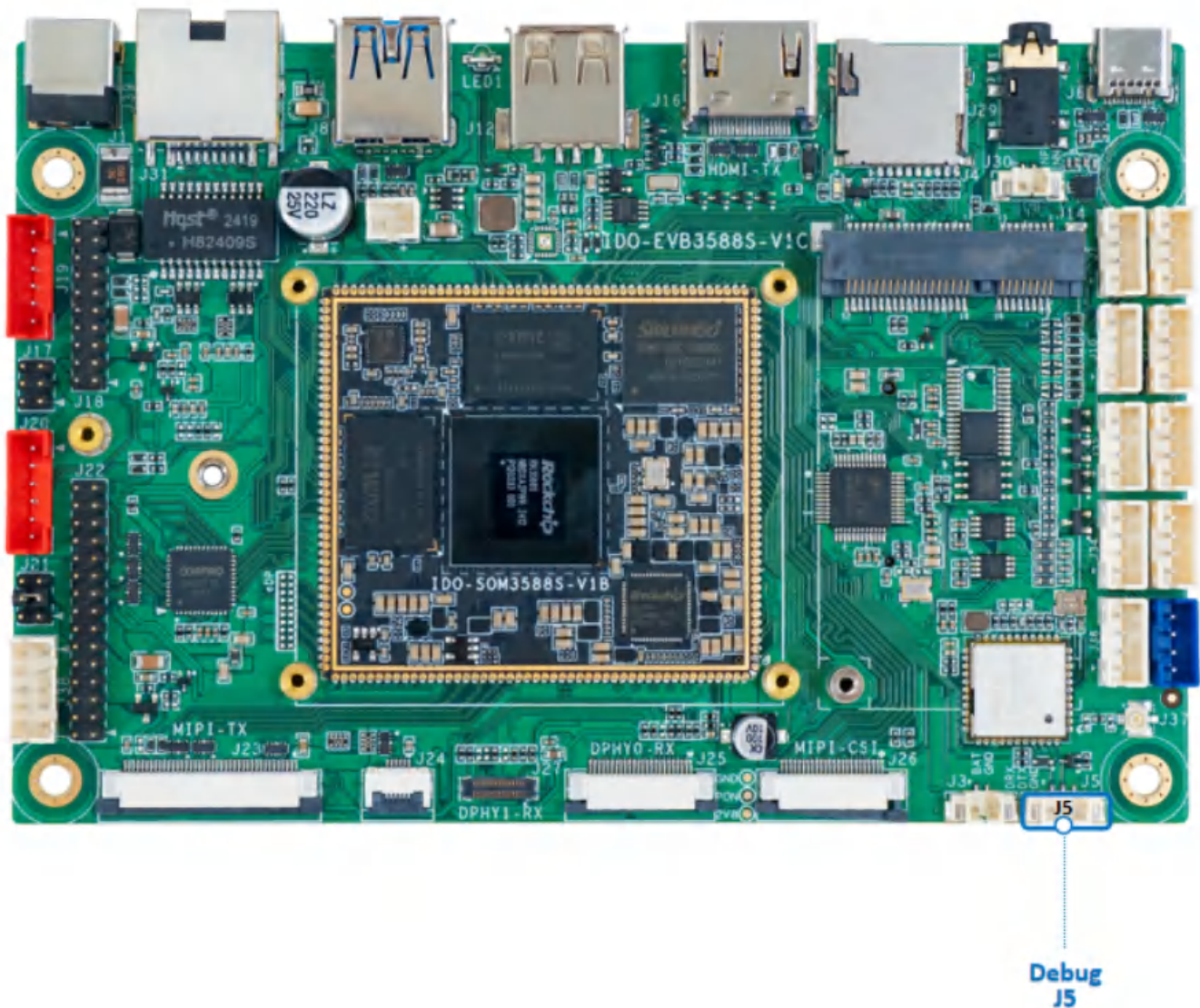
22	按键	Recovery按键 × 1 BOOT按键 × 1
23	MIC	2Pin 1.25 × 1
24	M2.0 SSD	NVME SSD × 1

2. IDO-EVB3588S-V1C-Debian系统

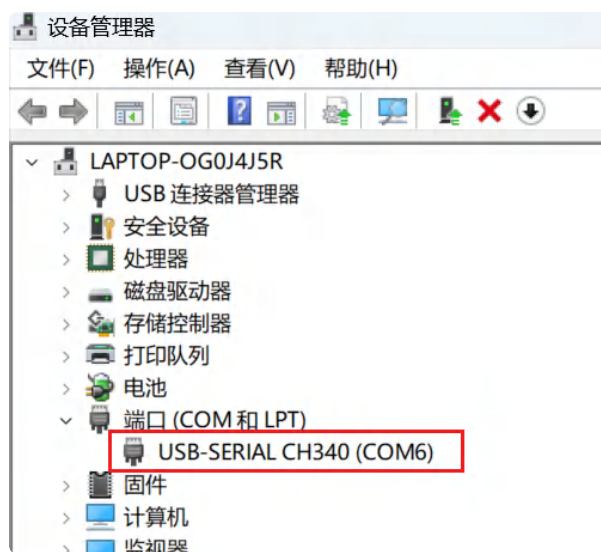
2.1. 功能及接口使用方法

2.1.1. DEBUG UART

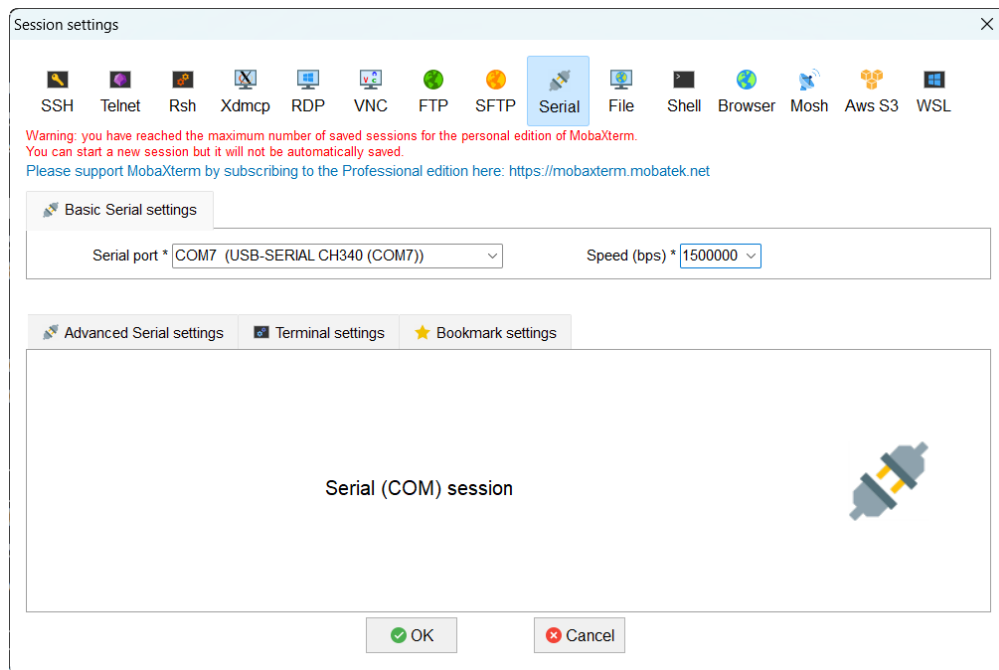
DEBUG UART位置J5，如下图所示：



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。



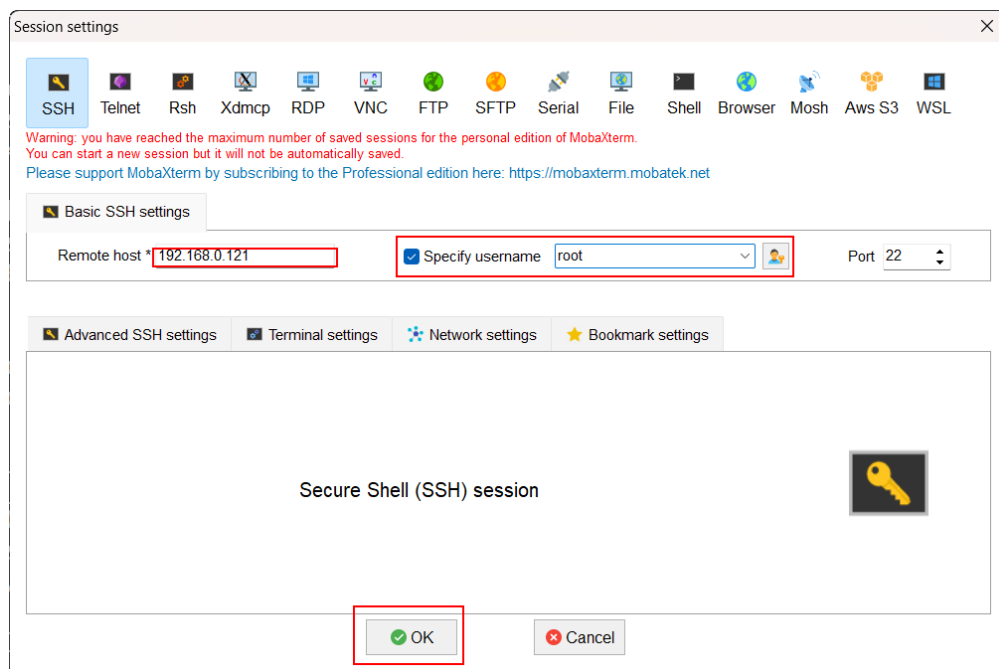
使用调试软件（MobaXterm、putty）等，以MobaXterm为例子，设置参数如下：



2.1.2. SSH登录

在知道IP的前提下，默认可以通过SSH登录进系统。

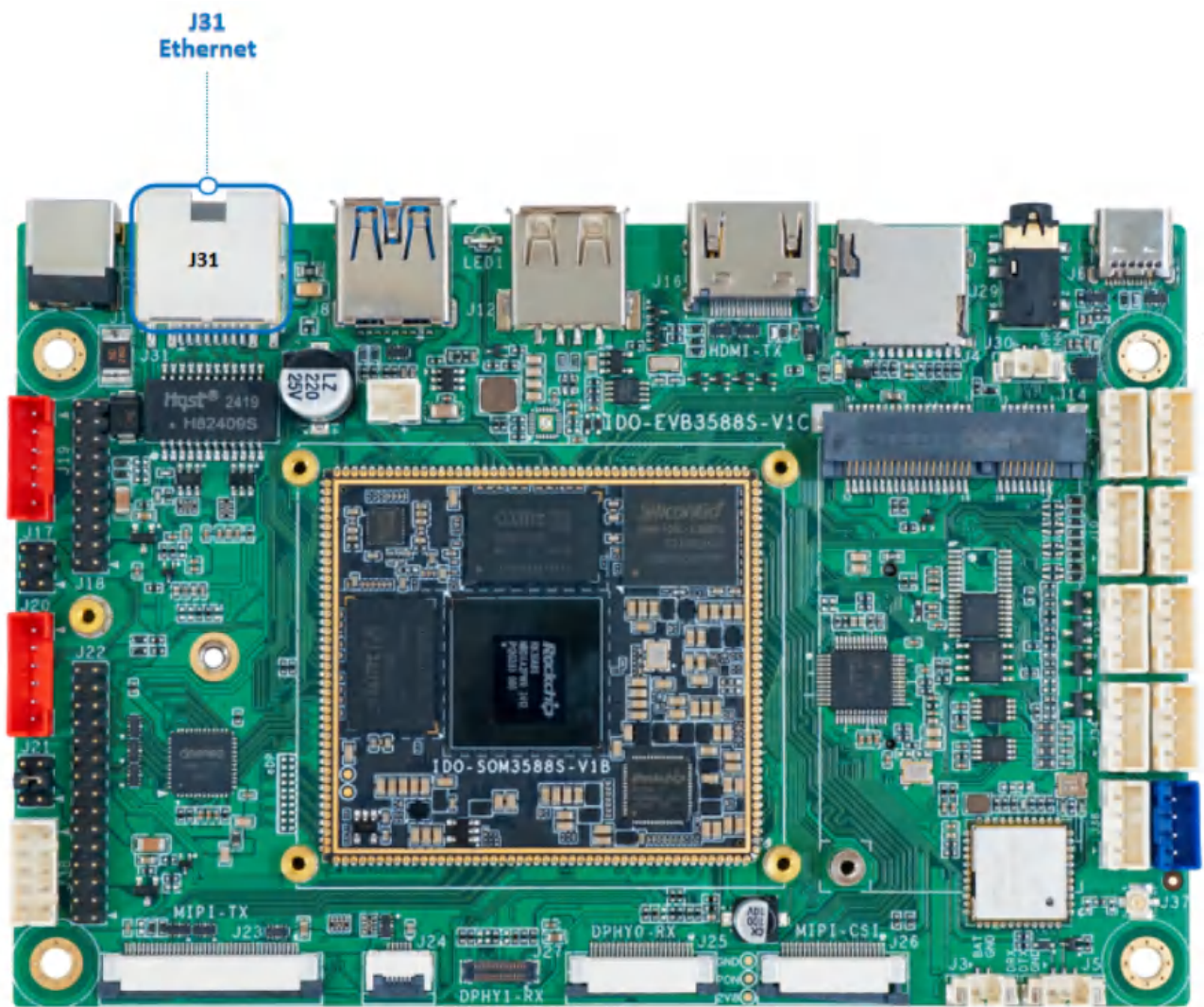
SSH登录账号密码：root/123456



2.1.3. 网络

2.1.3.1. 以太网

主板有一路千兆以太网接口位置J31，如下图所示：



设备节点分别为eth0，以太网接口默认支持DHCP，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。网络正常连接图标，如下图所示：

2.1.3.2. Ethernet动态IP设置

```
1 #eth0
2 root@linaro-alip:/# ifconfig eth0
3 eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
4     inet 192.168.0.90 netmask 255.255.255.0 broadcast 192.168.0.255
5     inet6 fe80::1b83:a156:cb16:c6b0 prefixlen 64 scopeid 0x20<link>
6     ether fa:c6:ba:c7:0c:60 txqueuelen 1000 (Ethernet)
7     RX packets 141 bytes 15129 (14.7 KiB)
8     RX errors 0 dropped 4 overruns 0 frame 0
9     TX packets 89 bytes 9306 (9.0 KiB)
10    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
11    device interrupt 121
12
```

2.1.3.3. Ethernet静态IP设置

```

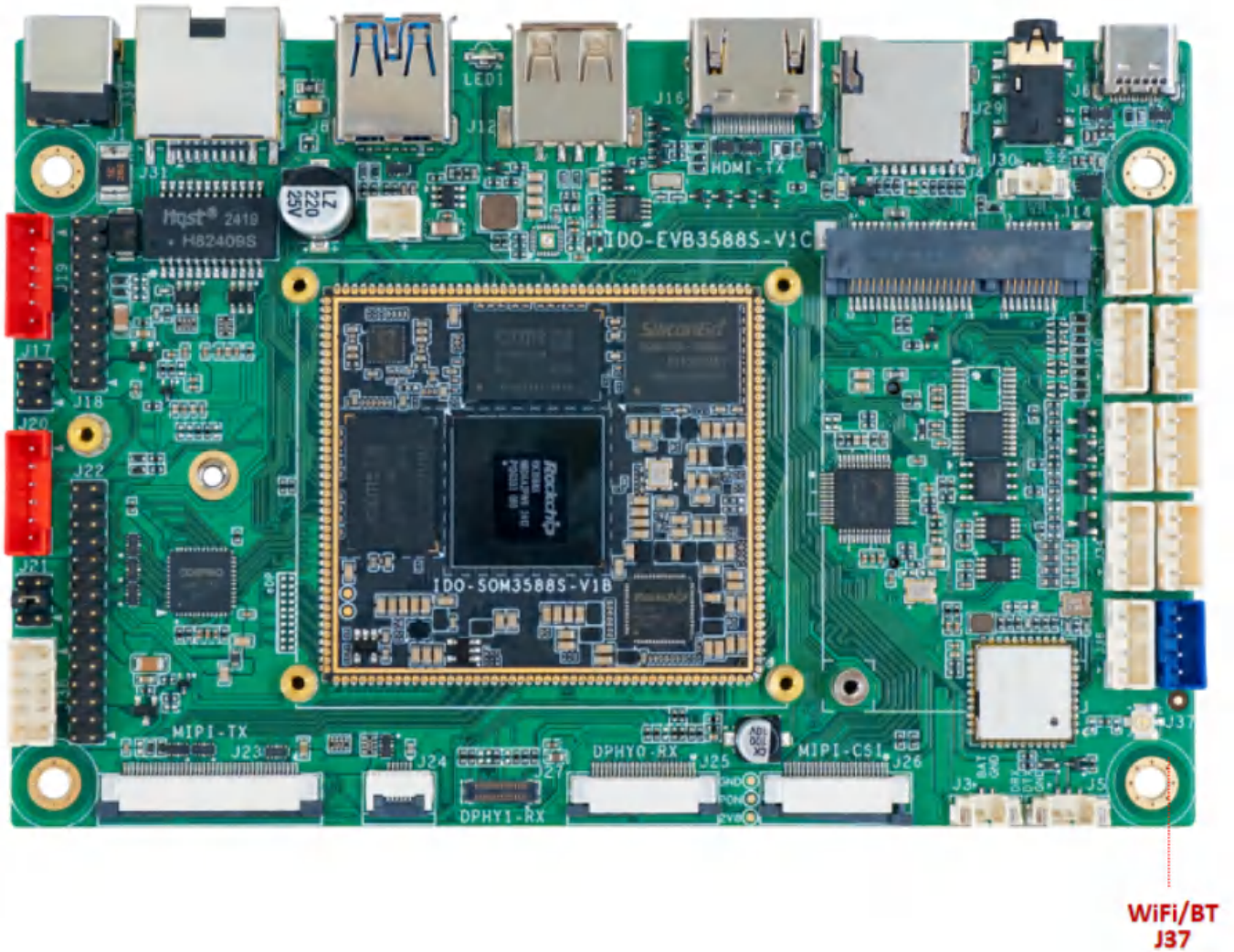
1 root@linaro-alip:/# touch /etc/netplan/01-netcfg.yaml
2 root@linaro-alip:/# vim /etc/netplan/01-netcfg.yaml
3 root@linaro-alip:/# cat /etc/netplan/01-netcfg.yaml
4 network:
5     version: 2
6     ethernet:
7         eth0:
8             dhcp4: no
9             addresses: [192.168.3.121/24]
10            nameservers:
11                addresses: [192.168.3.1, 8.8.8.8, 114.114.1
12                    14.114]
13            routes:
14                - to: 0.0.0.0/0
15                  via: 192.168.3.1
16                  metric: 100
17 #立即生效
18 root@linaro-alip:/# netplan apply
19 Cannot call openvswitch: ovsdb-server.service is not running.
20 root@linaro-alip:/# ifconfig eth0
21 eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
22     inet 192.168.3.121 netmask 255.255.255.0 broadcast 192.168.3.255
23     inet6 fe80::1b83:a156:cb16:c6b0 prefixlen 64 scopeid 0x20<link>
24     inet6 fe80::f8c6:baff:fec7:c60 prefixlen 64 scopeid 0x20<link>
25     ether fa:c6:ba:c7:0c:60 txqueuelen 1000 (Ethernet)
26     RX packets 4203 bytes 417790 (407.9 KiB)
27     RX errors 0 dropped 164 overruns 0 frame 0
28     TX packets 392 bytes 32299 (31.5 KiB)
29     TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
30     device interrupt 121

```

设备断电重启，此静态IP设置仍然生效。

2.1.4. 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J37，如下图所示：



命令行可以使用nmcli工具连接wifi热点：

```
1  #查看网络接口状态：
2  root@linaro-alip:~# nmcli device
3  #使用以下命令查看当前可用的 WiFi 网络：
4  root@linaro-alip:~# nmcli device wifi list
5  #连接到 WiFi 网络：
6  root@linaro-alip:~# nmcli device wifi connect 账号 password 密码
7  #确认连接状态：
8  root@linaro-alip:~# nmcli connection show --active
```

查看wlan0的IP地址，确认连接成功：

```
1 root@linaro-alip:/# ifconfig wlan0
2 wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
3     inet 192.168.0.118 netmask 255.255.255.0 broadcast 192.168.0.255
4     inet6 fe80::1036:80b4:5082:7440 prefixlen 64 scopeid 0x20<link>
5     ether c0:f5:35:12:ad:a2 txqueuelen 1000 (Ethernet)
6     RX packets 170 bytes 22149 (21.6 KiB)
7     RX errors 0 dropped 5 overruns 0 frame 0
8     TX packets 24 bytes 2683 (2.6 KiB)
9     TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
10
11 root@linaro-alip:/# ping www.baidu.com -I wlan0
12 PING www.wshifen.com (103.235.47.188) from 192.168.0.118 wlan0: 56(84) bytes of data.
13 64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=1 ttl=45 time=274 ms
14 64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=2 ttl=45 time=299 ms
15 64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=3 ttl=45 time=320 ms
16 64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=4 ttl=45 time=241 ms
```

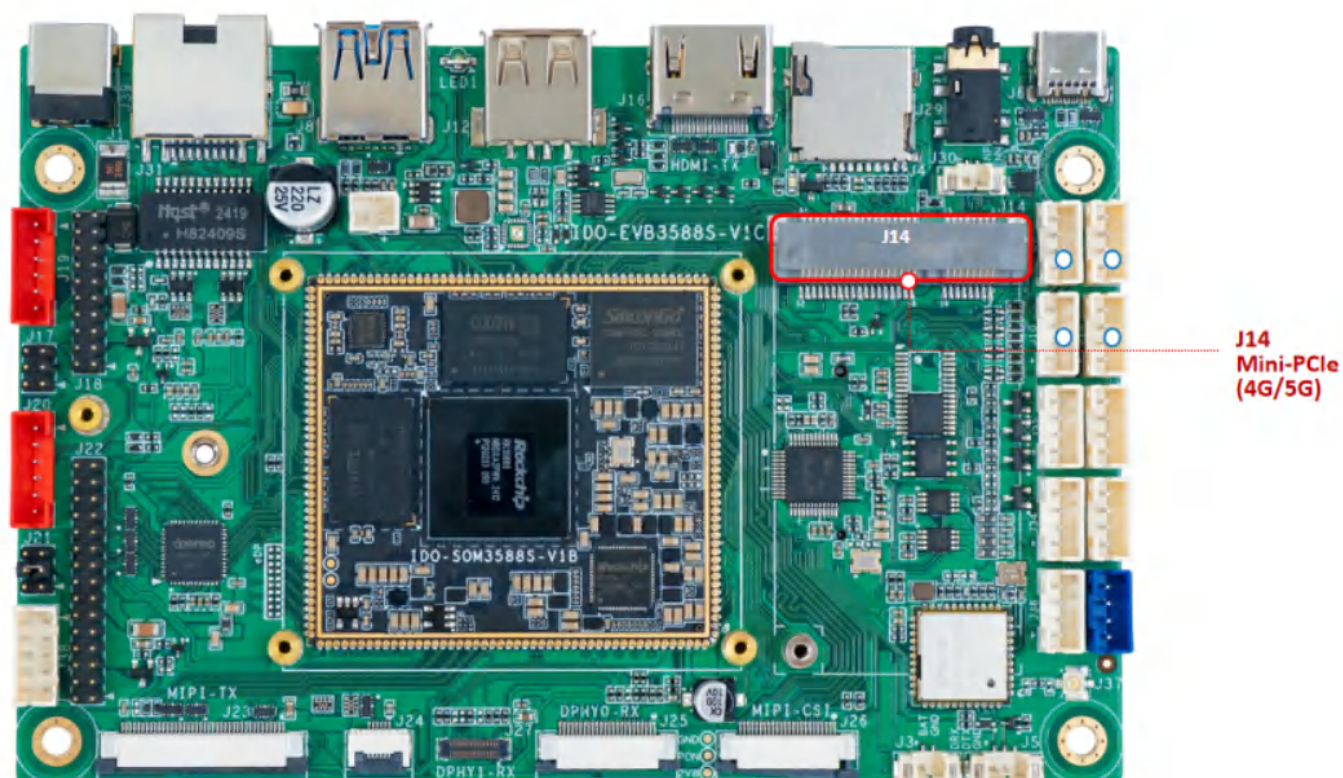
2.1.5. Bluetooth

2.1.5.1. 连接方法

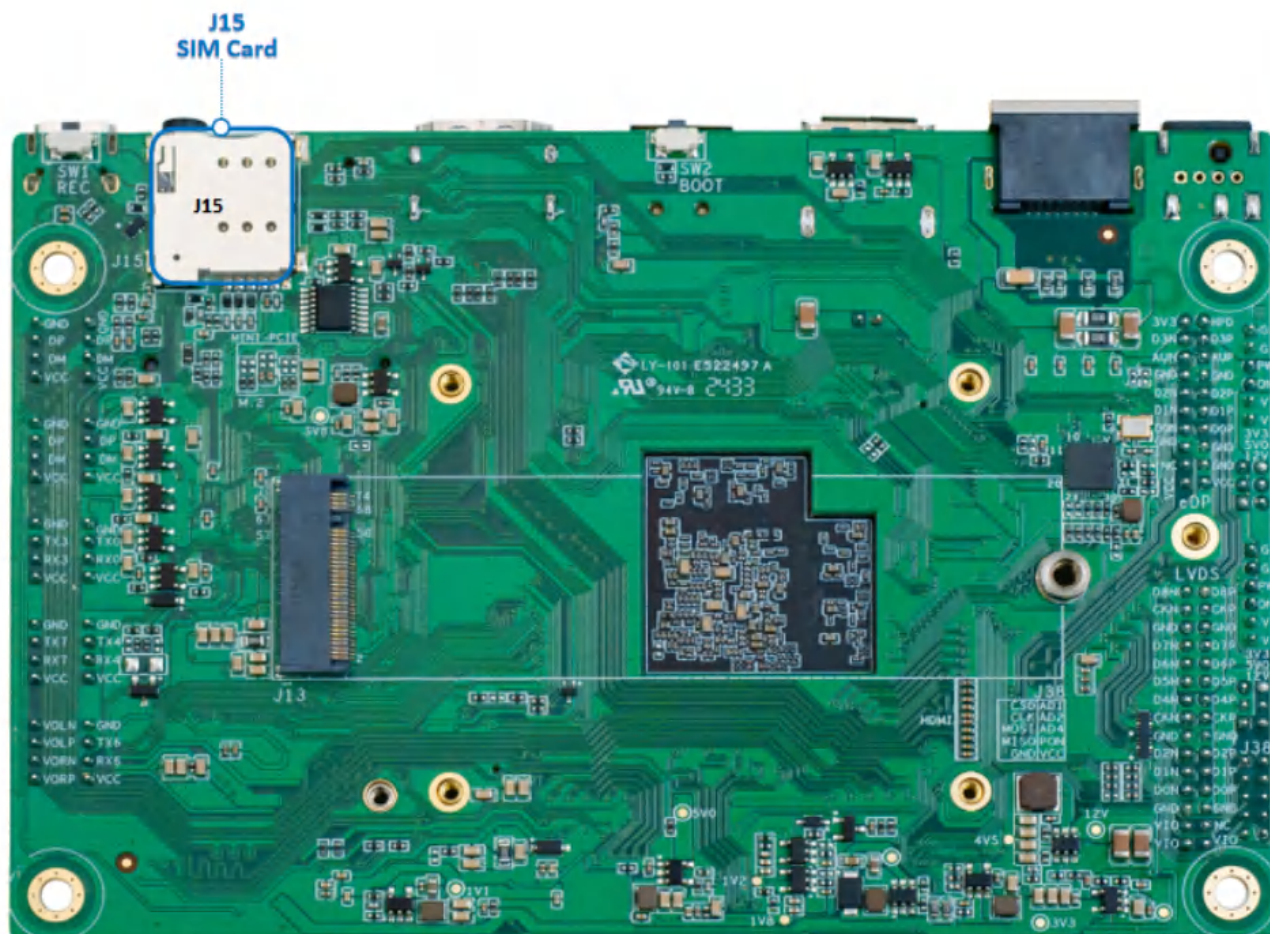
```
1 #打开蓝牙
2 root@linaro-alip:/# bluetoothctl
3 power on
4 #扫描蓝牙
5 scan on
6 #查看蓝牙设备
7 devices
8 #信任蓝牙设备
9 trust 98:EE:94:82:D5:71
10 #蓝牙配对
11 pair 98:EE:94:82:D5:71
12 #连接蓝牙
13 connect 98:EE:94:82:D5:71
```

2.1.6. 4G/5G

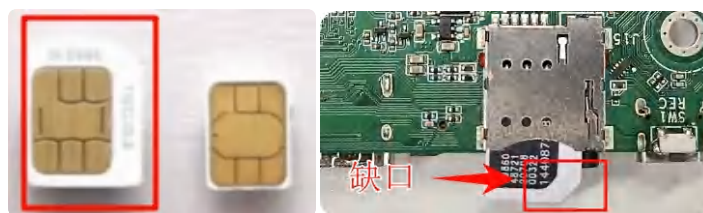
主板内置Mini PCIe 座子扩展 4G/5G模块，4G通信模块适配移远EC20、EC25等通用模组。5G通信模块适配移远RG200U-CN。模块安装位J14，如下图所示：



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J15位于主板背面，如下图所示：



使用Micro尺寸SIM卡，如下图所示：



4G使用 `ifconfig wwan0` 命令可查看到相关网络信息：

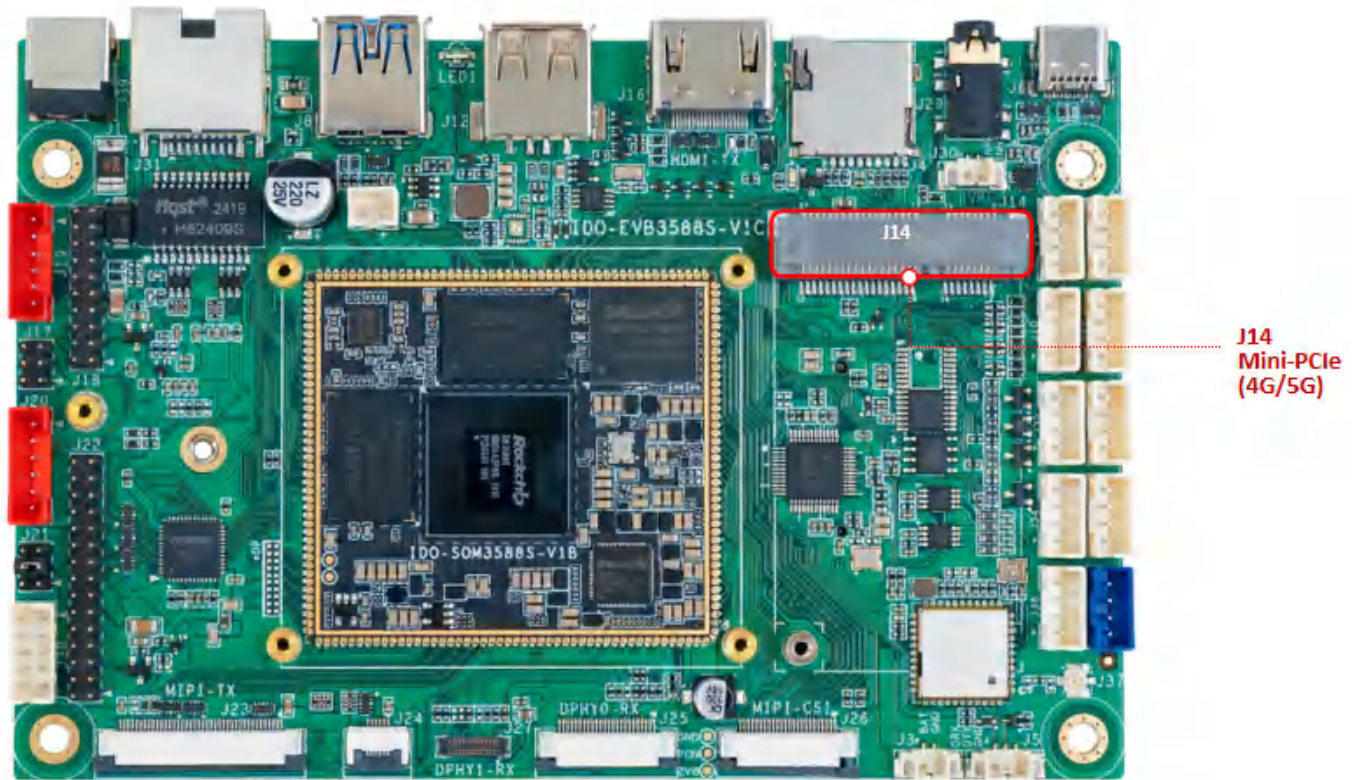
```
1 #拨号
2 root@linaro-alip:~# quectel-CM &
3
4 #查看网络
5 root@linaro-alip:~# ifconfig wwan0
6 wwan0: flags=193<UP,RUNNING,NOARP> mtu 1500
7     inet 10.101.61.51 netmask 255.255.255.248
8     inet6 fe80::fc:f6ff:fe8d:bab6 prefixlen 64 scopeid 0x20<link>
9     ether 02:fc:f6:8d:ba:b6 txqueuelen 1000 (Ethernet)
10    RX packets 42 bytes 7013 (6.8 KiB)
11    RX errors 0 dropped 0 overruns 0 frame 0
12    TX packets 57 bytes 4608 (4.5 KiB)
13    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
14
15 #测试通信
16 root@linaro-alip:~# ping www.baidu.com -I wwan0
```

5G使用 `ifconfig usb0` 命令可查看到相关网络信息:

```
1 #拨号
2 root@linaro-alip:~# quectel-CM &
3
4 #查看网络
5 root@linaro-alip:~# ifconfig usb0
6 usb0: flags=193<UP,RUNNING,NOARP> mtu 1500
7     inet 10.101.61.51 netmask 255.255.255.248
8     inet6 fe80::fc:f6ff:fe8d:bab6 prefixlen 64 scopeid 0x20<link>
9     ether 02:fc:f6:8d:ba:b6 txqueuelen 1000 (Ethernet)
10    RX packets 42 bytes 7013 (6.8 KiB)
11    RX errors 0 dropped 0 overruns 0 frame 0
12    TX packets 57 bytes 4608 (4.5 KiB)
13    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
14
15 #测试通信
16 root@linaro-alip:~# ping www.baidu.com -I usb0
```

2.1.7. mSATA

mSATA与4G/5G复用接口，如下图所示：

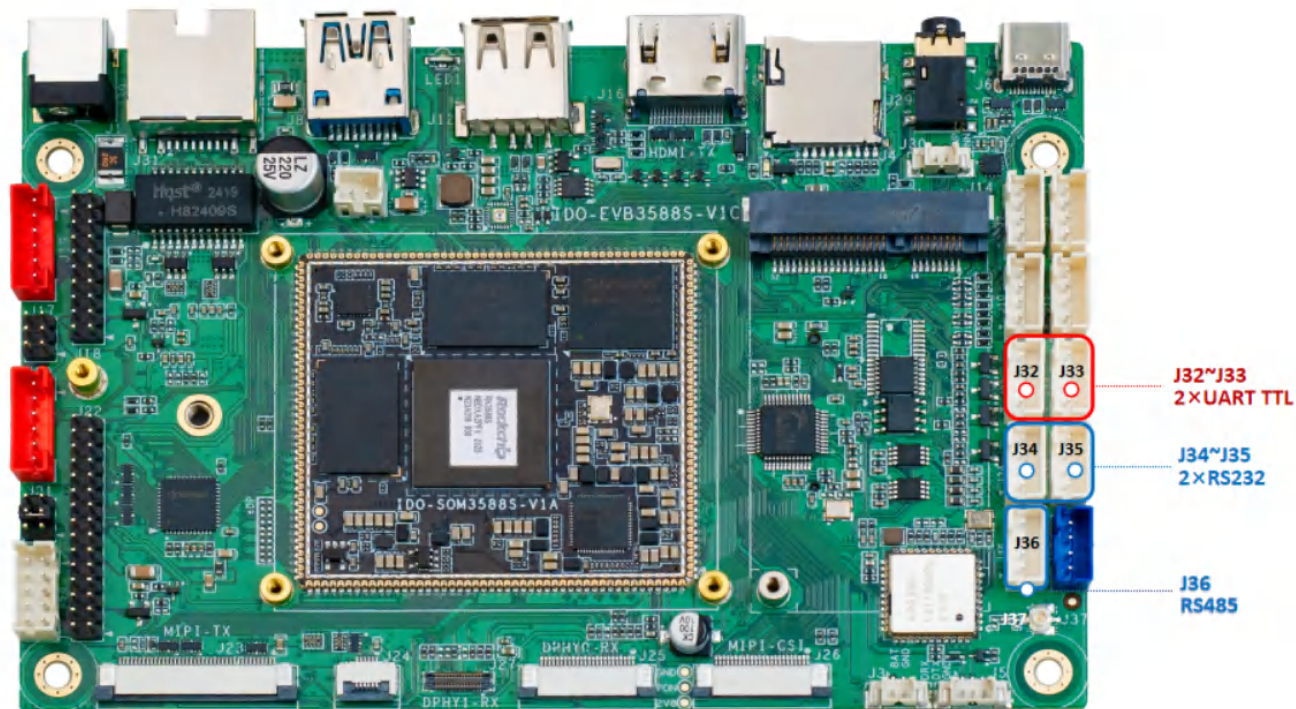


注意：

1. 默认配置为 4G，支持 mSATA 需要修改软硬件

2.1.8. UART

主板一共引出5路串口（不含调试串口）,如下图所示：



UART设备节点列表如下：

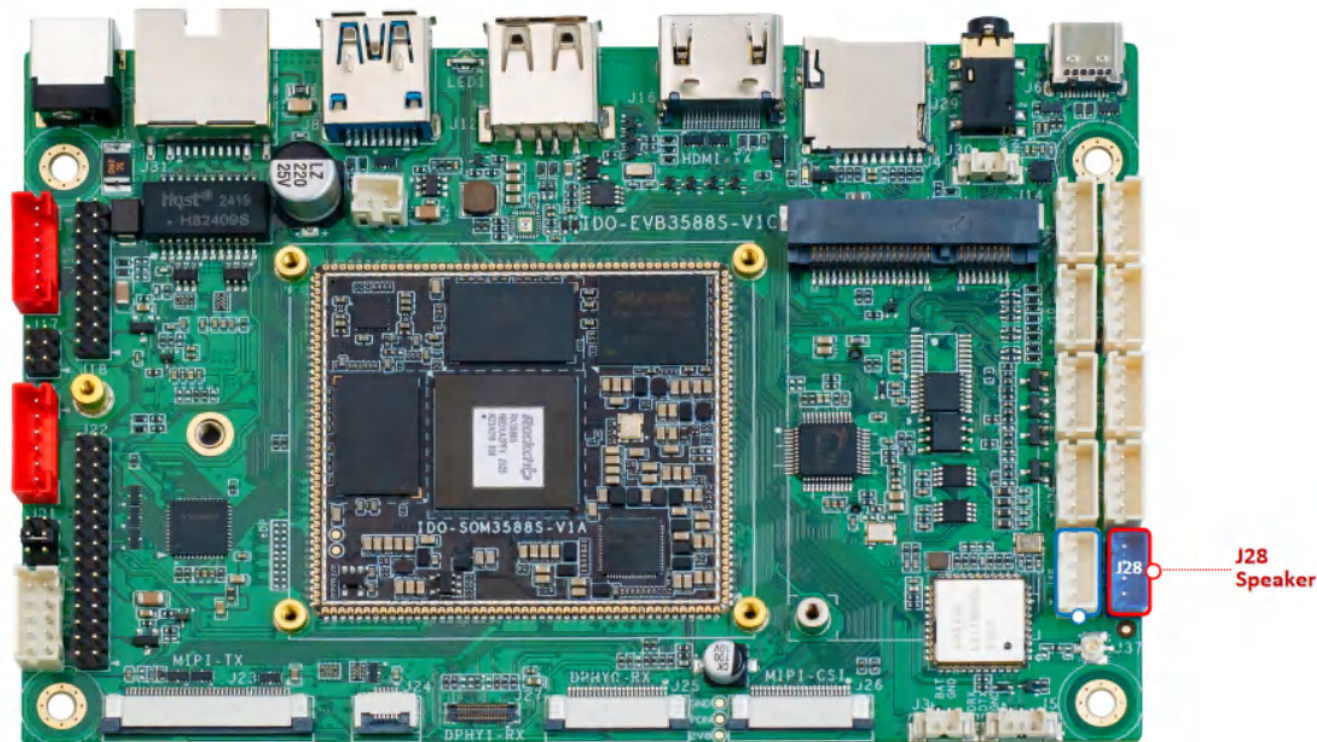
序号	功能	设备节点
0 (J32)	TTL	/dev/ttyS0
3 (J33)	TTL	/dev/ttyS3
4 (J34)	RS232	/dev/ttyS4
7 (J35)	RS232	/dev/ttyS7
6 (J36)	RS485	/dev/ttyS6

```
1 root@linaro-alip:~# microcom -p /dev/ttyS4 -s 115200
```

2.1.9. 声音

2.1.9.1. 喇叭

喇叭接口为PH2.0-4P连接器J28，如下图所示：



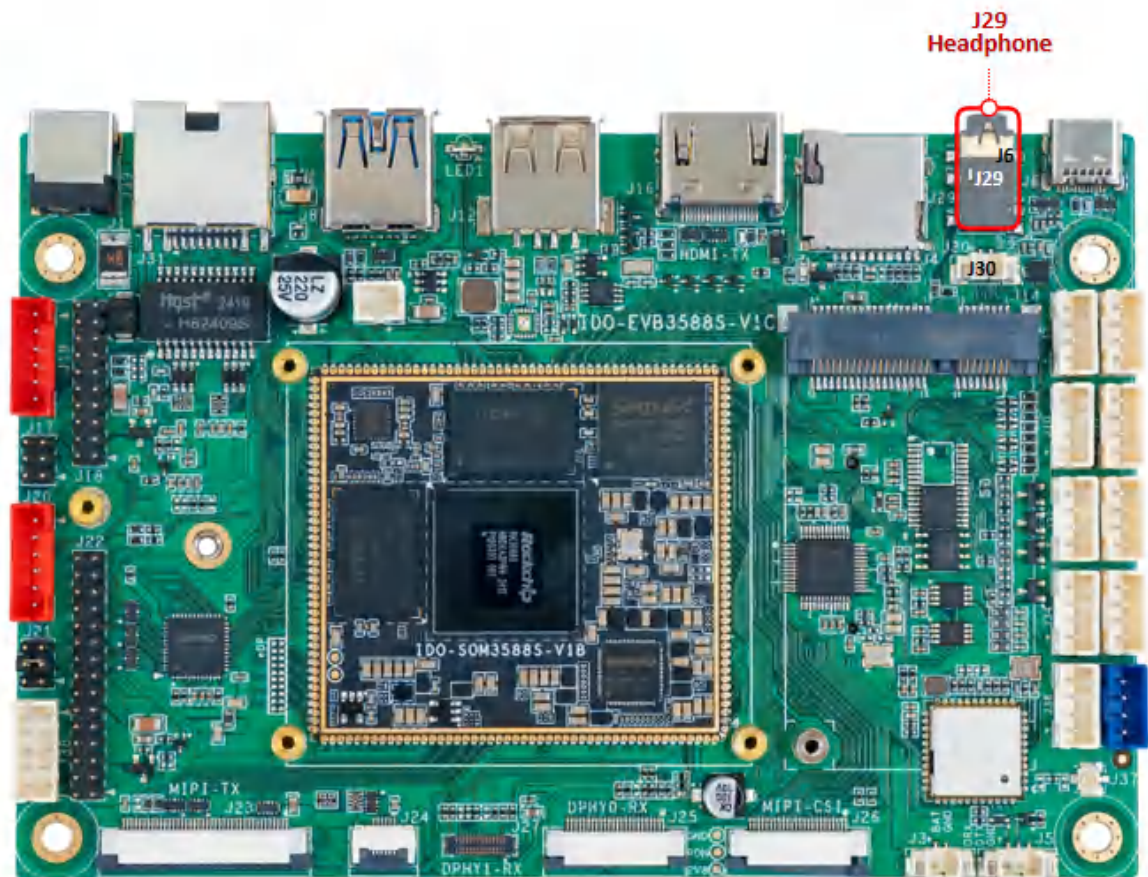
- 支持双声道，每个声道支持4Ω 3W输出

Plain Text

```
1 #查看声卡
2 root@linaro-alip:/# aplay -l
3
4 #音频播放到DP
5 root@linaro-alip:/# aplay -D hw:1,0 xihuangni.wav
6 Playing WAVE 'xihuangni.wav' : Signed 16 bit Little Endian, Rate 44100 H
z, Stereo
7
8 #音频播放到HDMI
9 root@linaro-alip:/# aplay -D hw:2,0 xihuangni.wav
10 Playing WAVE 'xihuangni.wav' : Signed 16 bit Little Endian, Rate 44100 H
z, Stereo
11
12 #音频播放到声卡
13 root@linaro-alip:/# aplay -D hw:0,0 xihuangni.wav
14 Playing WAVE 'xihuangni.wav' : Signed 16 bit Little Endian, Rate 44100 H
z, Stereo
```

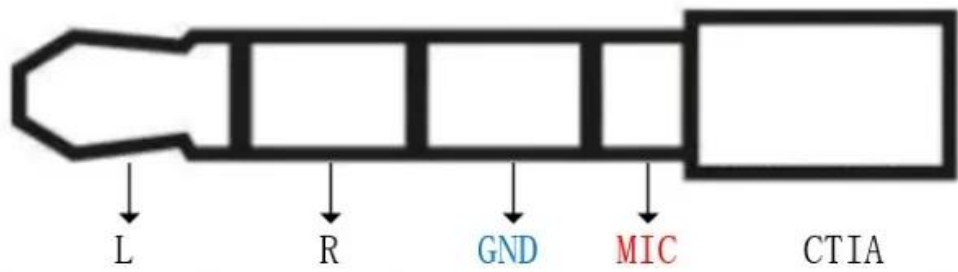
2.1.9.2. 耳机

主板支持1路3.5mm四节耳机座J29，如下图所示：



- 支持耳机检测
- 支持耳机录音

CTIA标准四段式耳机，定义如下：

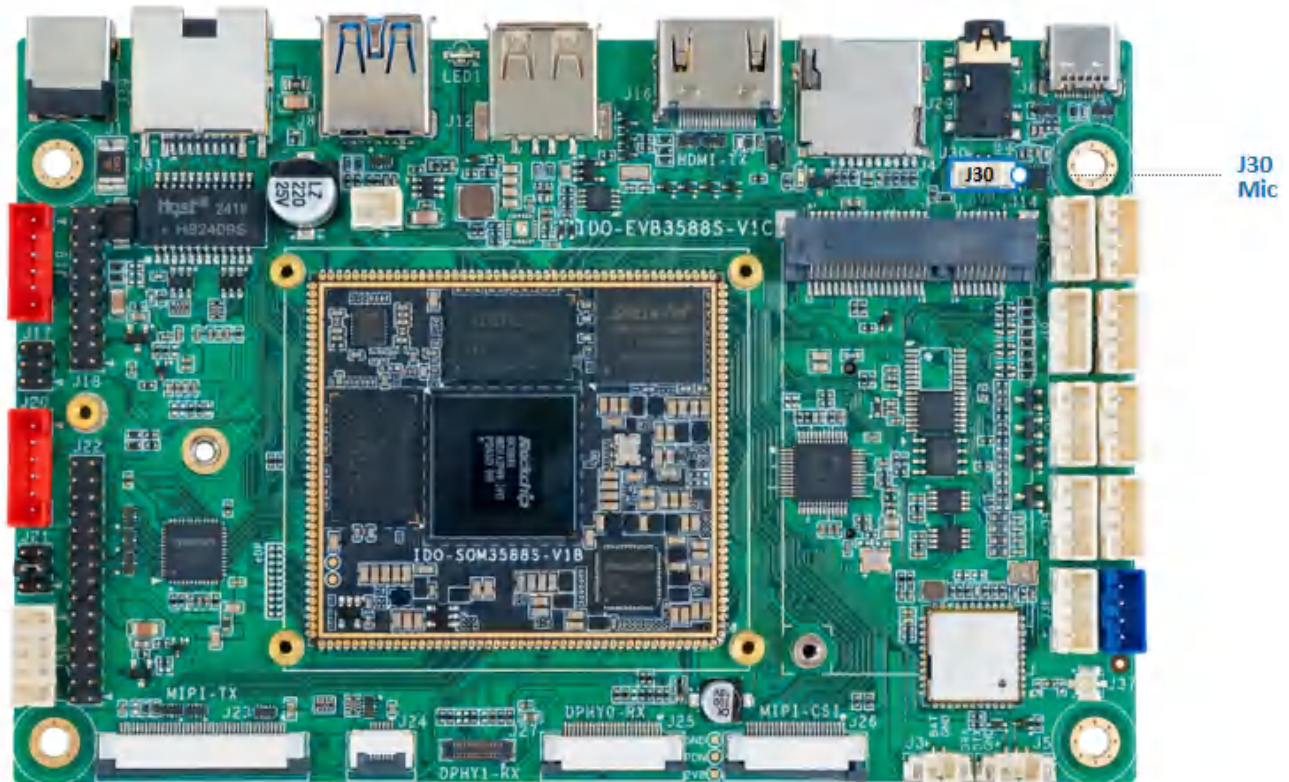


耳机录音：

```
1 #耳机录音
2 root@linaro-alip:/# amixer -c 0 cset numid=43 on
3 root@linaro-alip:/# echo -n "mic1" > /sys/class/es8388_mic/mic_channel
4
5 root@rk3588-buildroot:/# arecord -D hw:0,0 -f cd -r 44100 -c 2 test.wav
6 Recording WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

2.1.9.3. Mic

MX1.25T-2P麦克风接口J30，如下图所示：

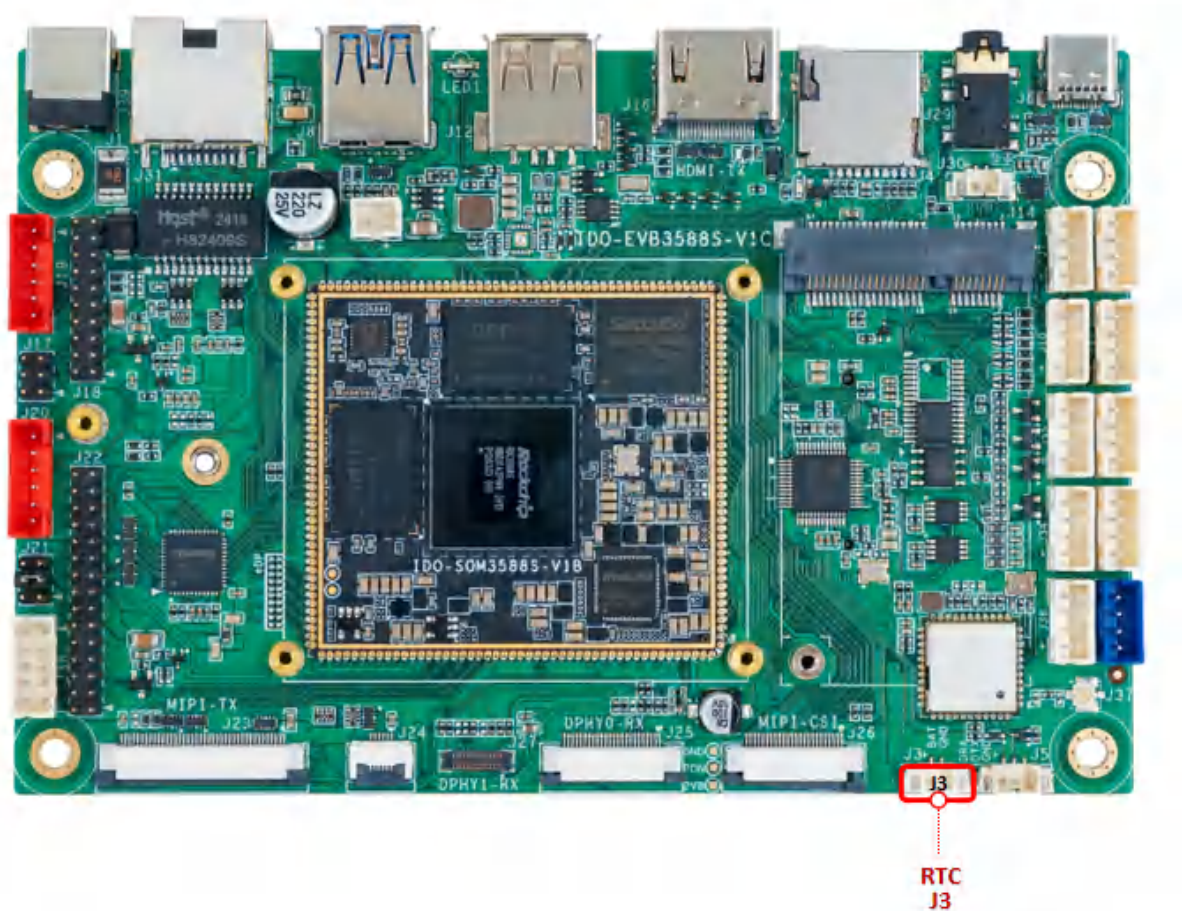


Mic录音：

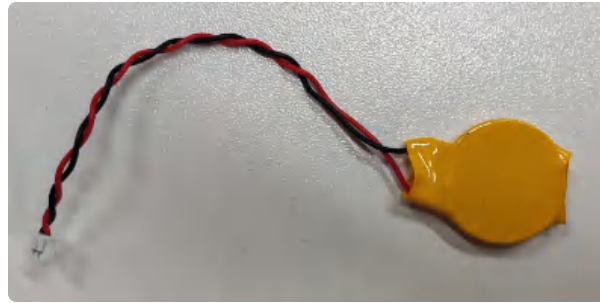
```
1 #Mic录音
2 root@linaro-alip:/# amixer -c 0 cset numid=43 on
3 root@linaro-alip:/# echo -n "mic2" > /sys/class/es8388_mic/mic_channel
4
5 root@linaro-alip:/# arecord -D hw:0,0 -f S16_LE -r 44100 -c 2 test.wav
6 Recording WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

2.1.10. RTC

MX1.25T-2P RTC电池座J3，如下图所示：



连接3V 纽扣电池，RTC电池参考如下



rtc时间设置方法：

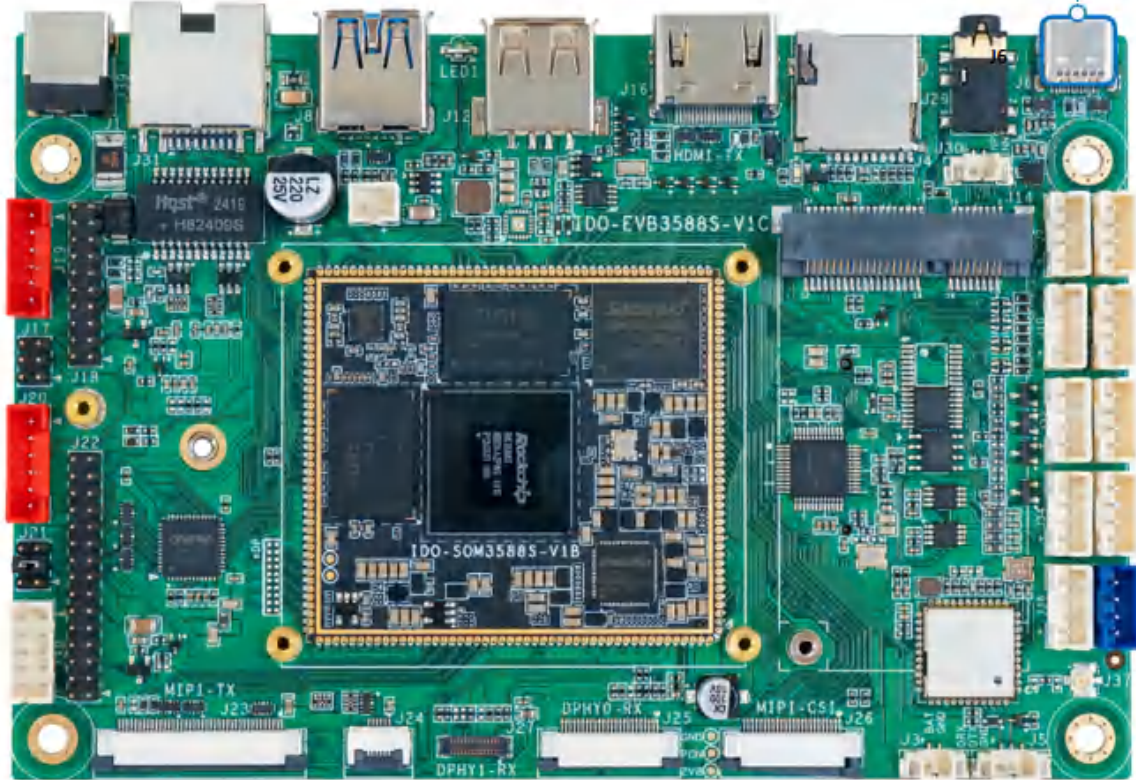
```
1  #设置时间
2  root@linaro-alip:/# date -s "2024-10-10 10:10:10"
3
4  #将rtc时钟调整为与目前的系统时钟一致
5  root@linaro-alip:/# hwclock -w
6
7  #获取硬件rtc当前时间（断电重启读取时间没有太大偏差）
8  root@linaro-alip:/# hwclock -r
9  2024-10-10 10:11:15.381196+08:00
```

2.1.11. USB接口

主板支持1个TypeC接口，支持2个USB-A，4个USB-PH2.0接口，USB对外总供电应小于3A。

2.1.11.1. TypeC 接口

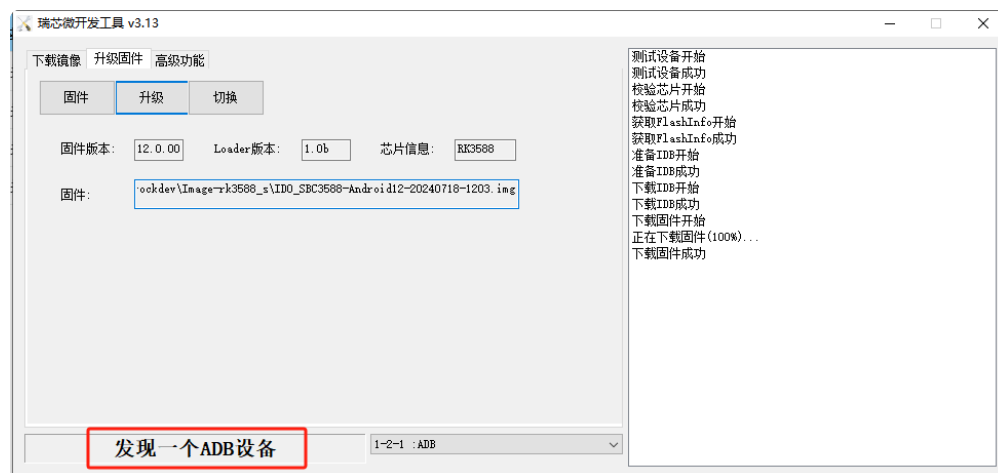
TypeC接口J6，如下图所示：



- 支持Host、Device模式自动切换
- 支持DP显示输出

2.1.11.1.1. 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：



- 可使用ADB相关开发工具对盒子进行功能调试

2.1.11.1.2. 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：



- 可识别U盘、键盘、鼠标并正常使用

2.1.11.1.3. DP模式

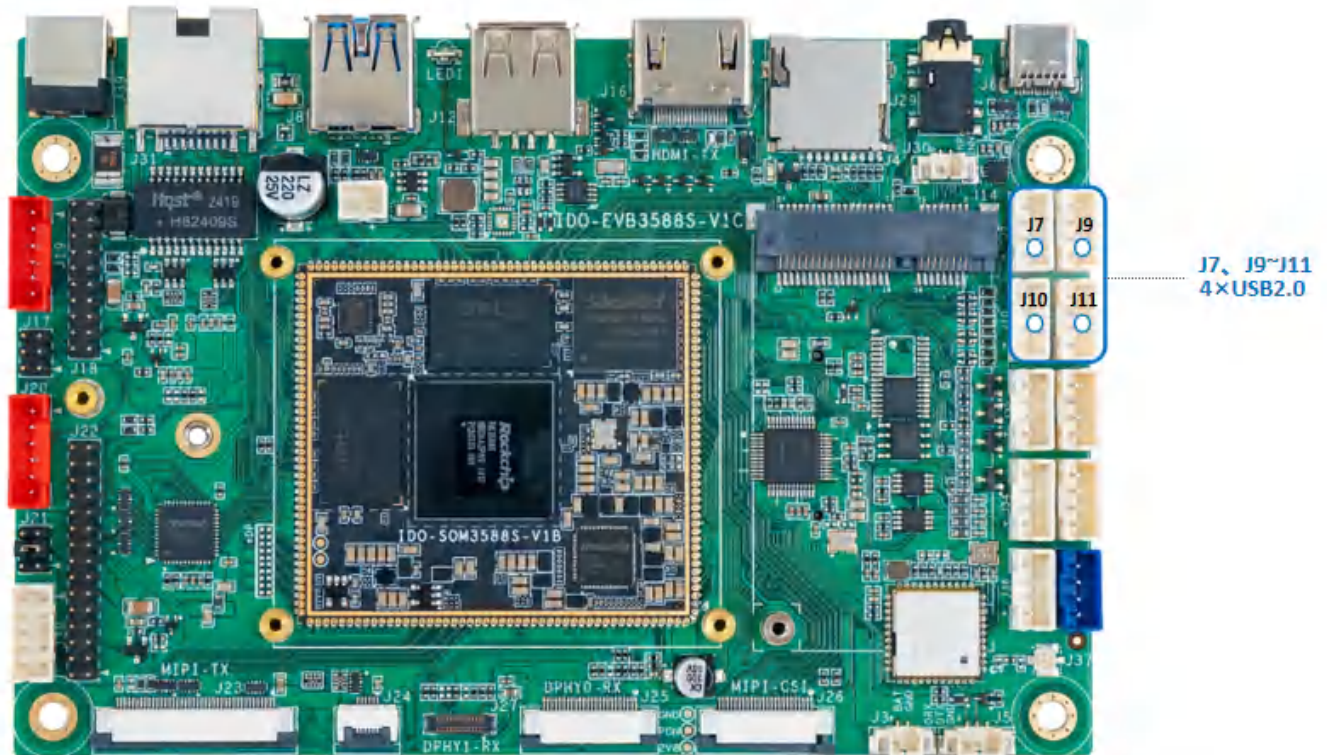
通过TypeC全功能数据线接入DP显示器，或通过TYPE-C to HDMI数据线连接HDMI显示器



- 主板画面通过TYPE-C DP输出正常，DP声音输出正常

2.1.11.2. USB PH2.0

支持四个USB PH2.0，如下图所示：



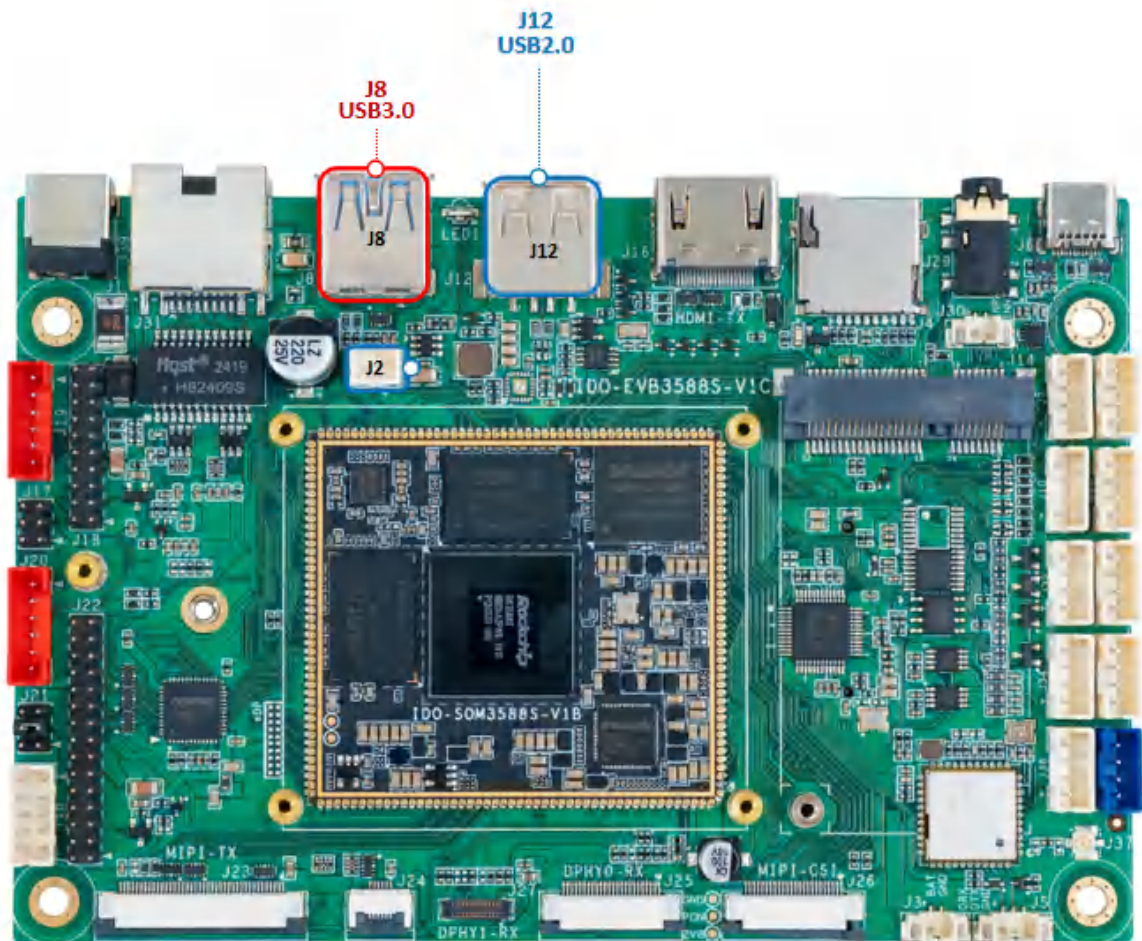
USB电源控制，如下表所示：

USB端口	默认状态	动作	命令
J7	开启	关闭电源	echo 0 > /sys/class/leds/usb_j7/brightness
		开启电源	echo 1 > /sys/class/leds/usb_j7/brightness
J9	开启	关闭电源	echo 0 > /sys/class/leds/usb_j9/brightness
		开启电源	echo 1 > /sys/class/leds/usb_j9/brightness
J10	开启	关闭电源	echo 0 > /sys/class/leds/usb_j10/brightness
		开启电源	echo 1 > /sys/class/leds/usb_j10/brightness
J11	开启	关闭电源	echo 0 > /sys/class/leds/usb_j11/brightness
		开启电源	echo 1 > /sys/class/leds/usb_j11/brightness

供电控制说明：设备节点写"0"关闭电源，写"1"开启电源。

2.1.11.3. USB接口

USB-A接口，如下图所示：



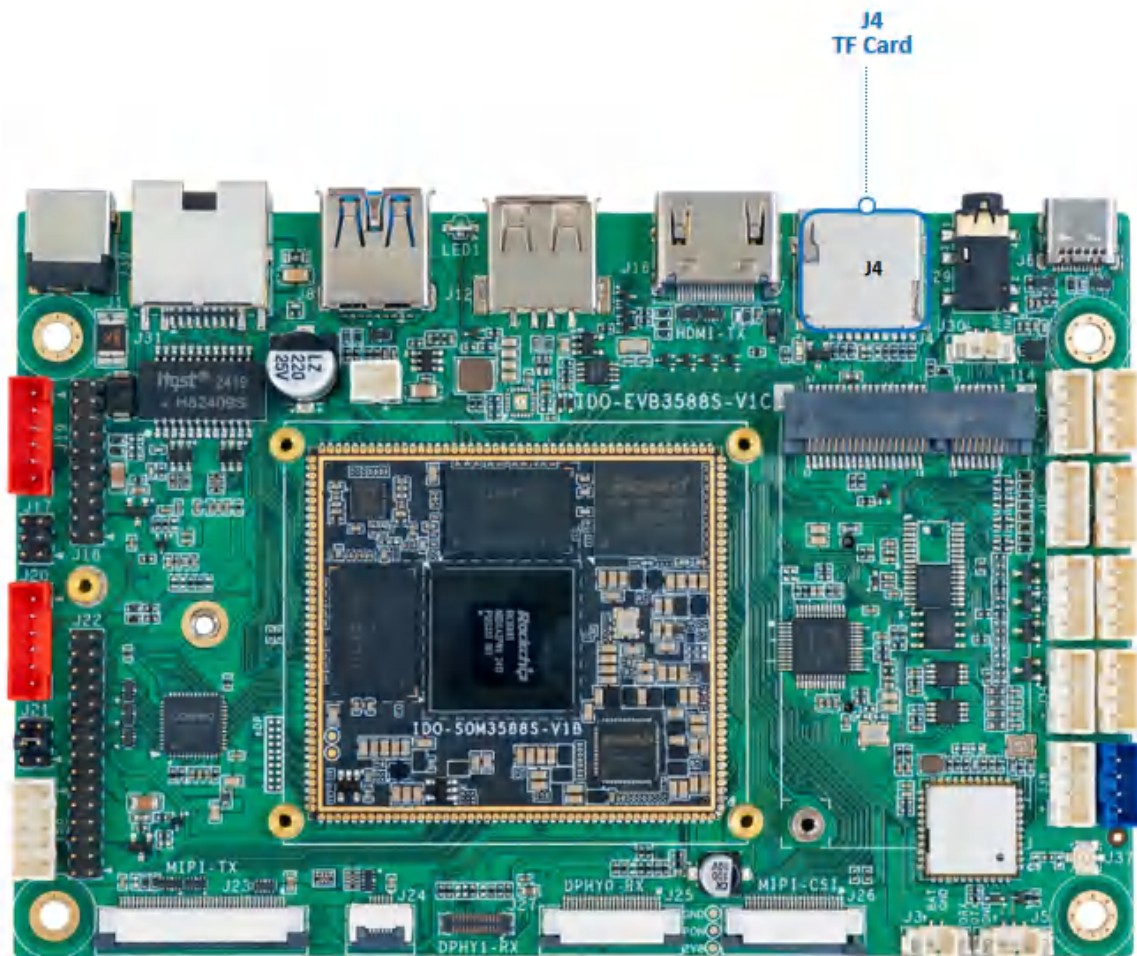
USB电源控制，如下表所示：

USB端口	默认状态	动作	命令
J8	开启	关闭电源	<code>echo 0 > /sys/class/leds/usb_j8/brightness</code>
		开启电源	<code>echo 1 > /sys/class/leds/usb_j8/brightness</code>
J12	开启	关闭电源	<code>echo 0 > /sys/class/leds/usb_j12/brightness</code>
		开启电源	<code>echo 1 > /sys/class/leds/usb_j12/brightness</code>

供电控制说明：设备节点写"0"关闭电源，写"1"开启电源。

2.1.12. TF Card

TF卡座接口位于J4，如下图所示：



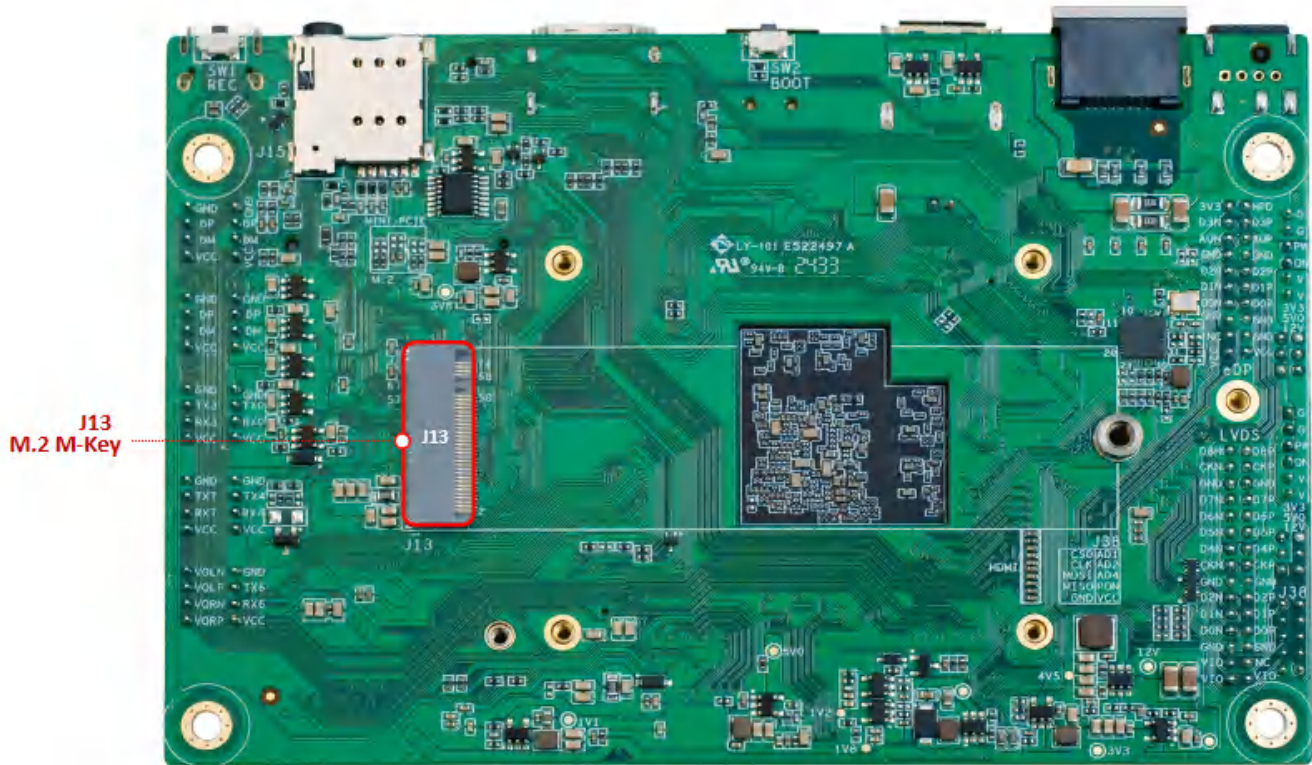
```

1 root@linaro-alip:/# fdisk -l
2 Disk /dev/ram0: 8 MiB, 8388608 bytes, 16384 sectors
3 Units: sectors of 1 * 512 = 512 bytes
4 Sector size (logical/physical): 512 bytes / 4096 bytes
5 I/O size (minimum/optimal): 4096 bytes / 4096 bytes
6 .....
7
8 Device Boot StartCHS      EndCHS          StartLBA      EndLBA        Sectors  Siz
  e Id Type
9 /dev/sda1    0,0,33      1023,254,63          32    30535679    30535648 14.5
  G  c Win95 FAT32 (LBA)
10
11 root@linaro-alip:/# df -h
12 Filesystem      Size  Used Avail Use% Mounted on
13 /dev/root        111G   1.1G  105G   1% /
14 devtmpfs         3.9G   8.0K   3.9G   1% /dev
15 tmpfs            3.9G  328K   3.9G   1% /tmp
16 tmpfs            3.9G  636K   3.9G   1% /run
17 .....
18 /dev/sda1        15G    32K   15G    1% /mnt/udisk/6C39-1D85

```

2.1.13. M.2接口

主板上使用标准M.2-M-key M.2接口连接座J13，如下图所示：



支持PCIe2.1通信，适用2280尺寸NVME固态硬盘。

▼

Shell

```

1  root@linaro-alip:/# fdisk -l
2  Disk /dev/nvme0n1: 233 GB, 250059350016 bytes, 488397168 sectors
3  30401 cylinders, 255 heads, 63 sectors/track
4  Units: sectors of 1 * 512 = 512 bytes
5
6  Device      Boot StartCHS      EndCHS          StartLBA      EndLBA      Sector
7  s  Size Id Type
8  /dev/nvme0n1p1  0,32,33      1023,254,63      2048  488397167  488395120
9  232G  7 HPFS/NTFS
10
11 root@linaro-alip:/# df -h
12 文件系统      大小  已用  可用  已用% 挂载点
13  Filesystem    Size  Used Avail Use% Mounted on
14  /dev/root      111G  1.1G  105G   1% /
15  .....
16  /dev/nvme0n1p1 233G   15G  219G   7% /mnt/m2

```

2.1.14. LCD显示

2.1.14.1. DP

DP接口（TYPE-C）J6，如下图所示：



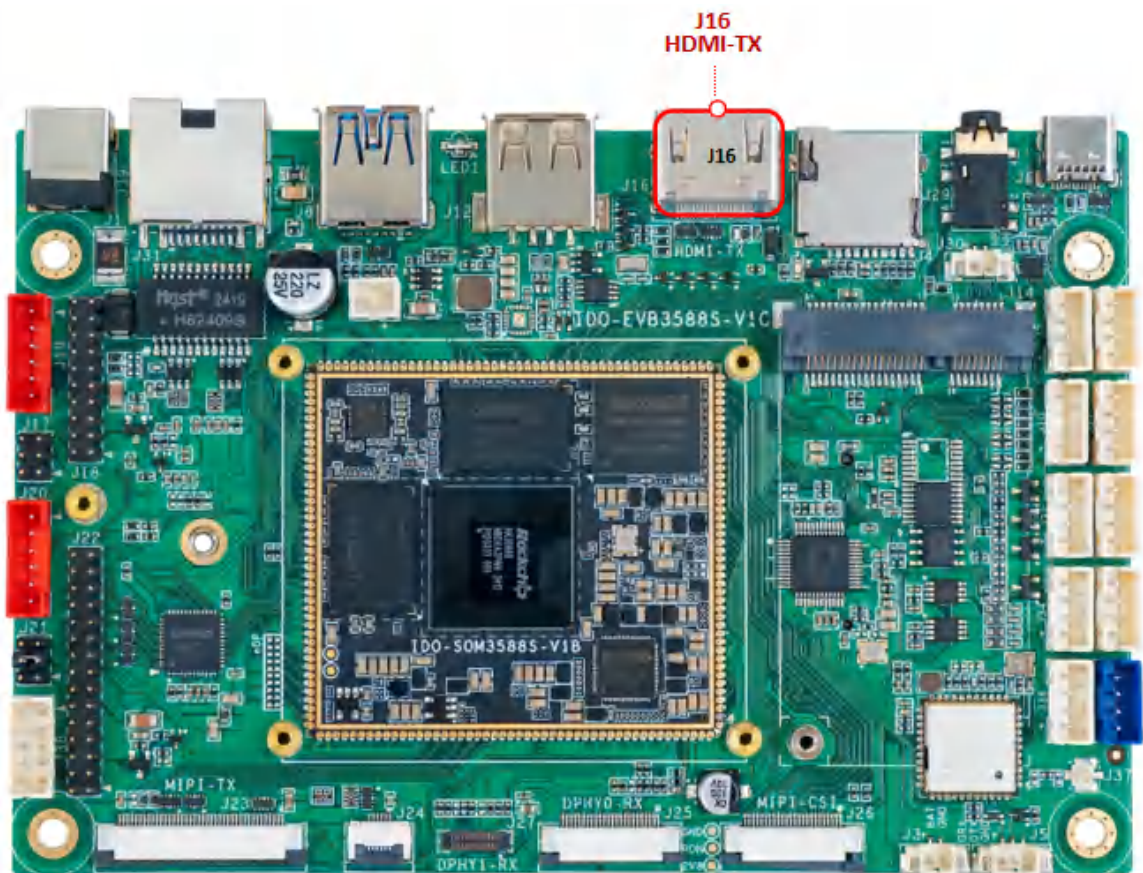
可以使用 USB-C 转 HDMI 高清线连接 HDMI 显示器输出画面，如下图所示：



- 最高支持8K@30fps输出

2.1.14.2. HDMI-TX

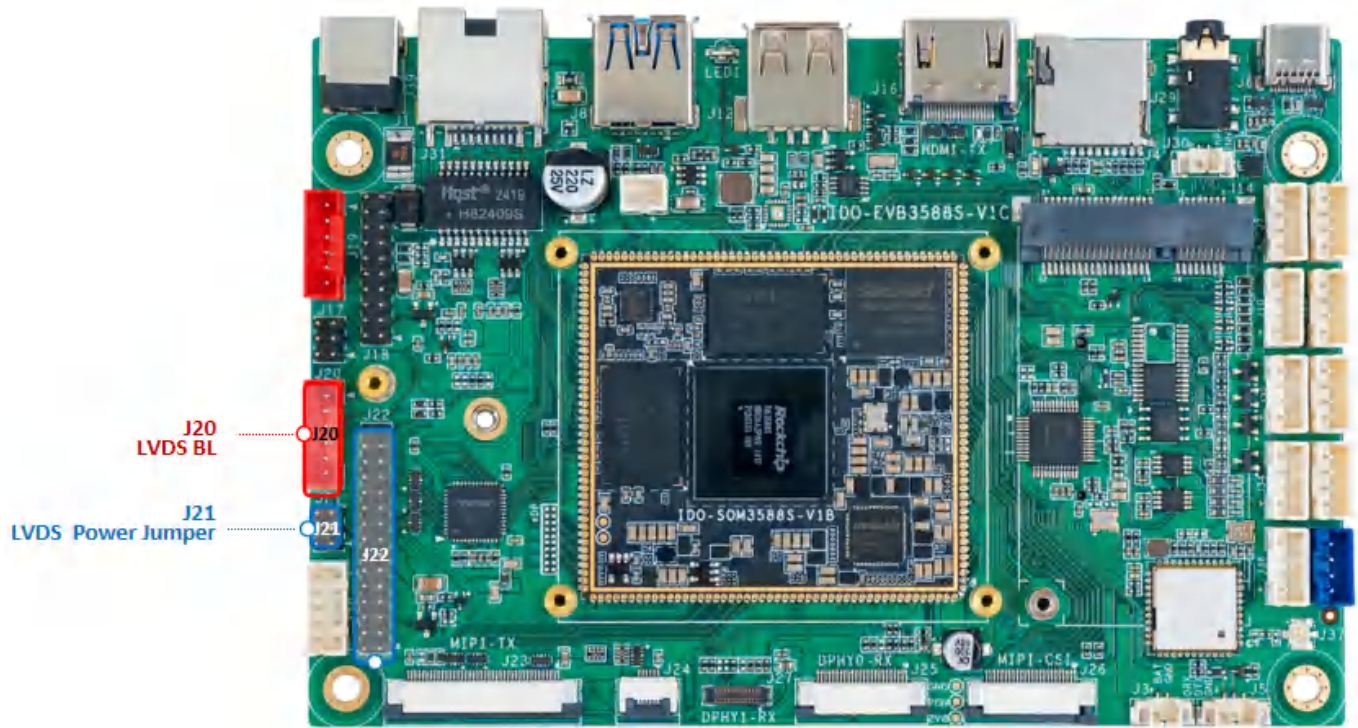
主板标准HDMI-A接口，如下图所示：



- 最高支持 HDMI2.1 8K@60fps 输出

2.1.14.3. Dual LVDS屏

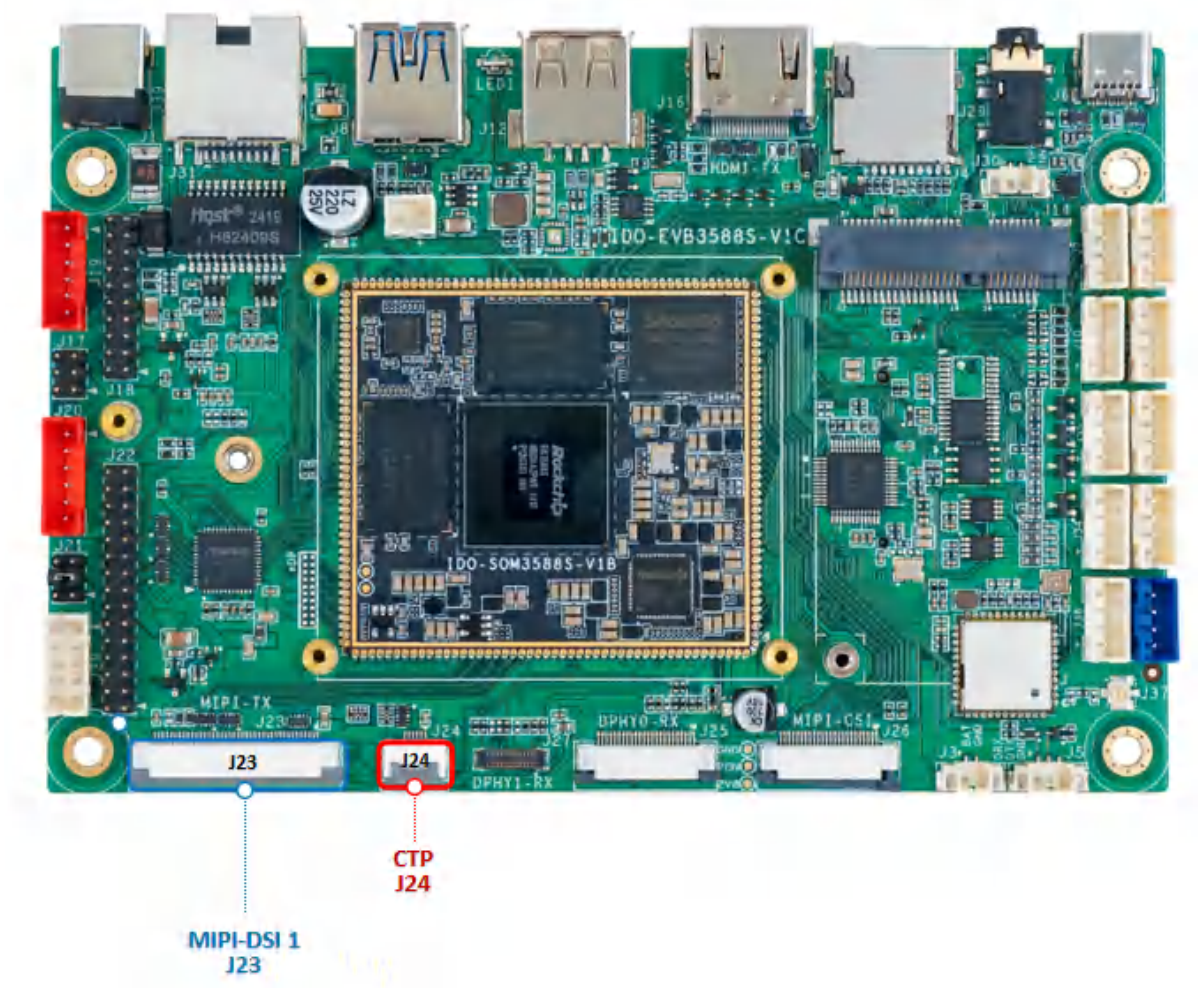
Dual LVDS接口J22，背光接 J20，屏幕供电电压选择接口J21，如下图所示：



J21屏幕供电电压选择接口，接屏前，需要根据具体的屏幕规格书来确认供电跳线帽接到3.3V、5V或12V，默认3.3V。

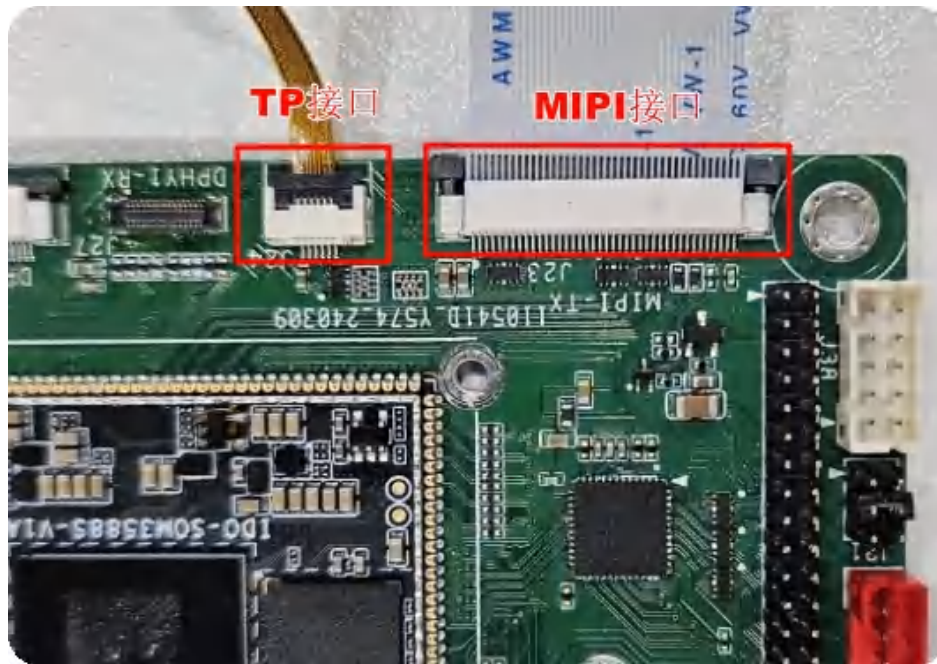
2.1.14.4. MIPI

MIPI接口J23，如下图所示：



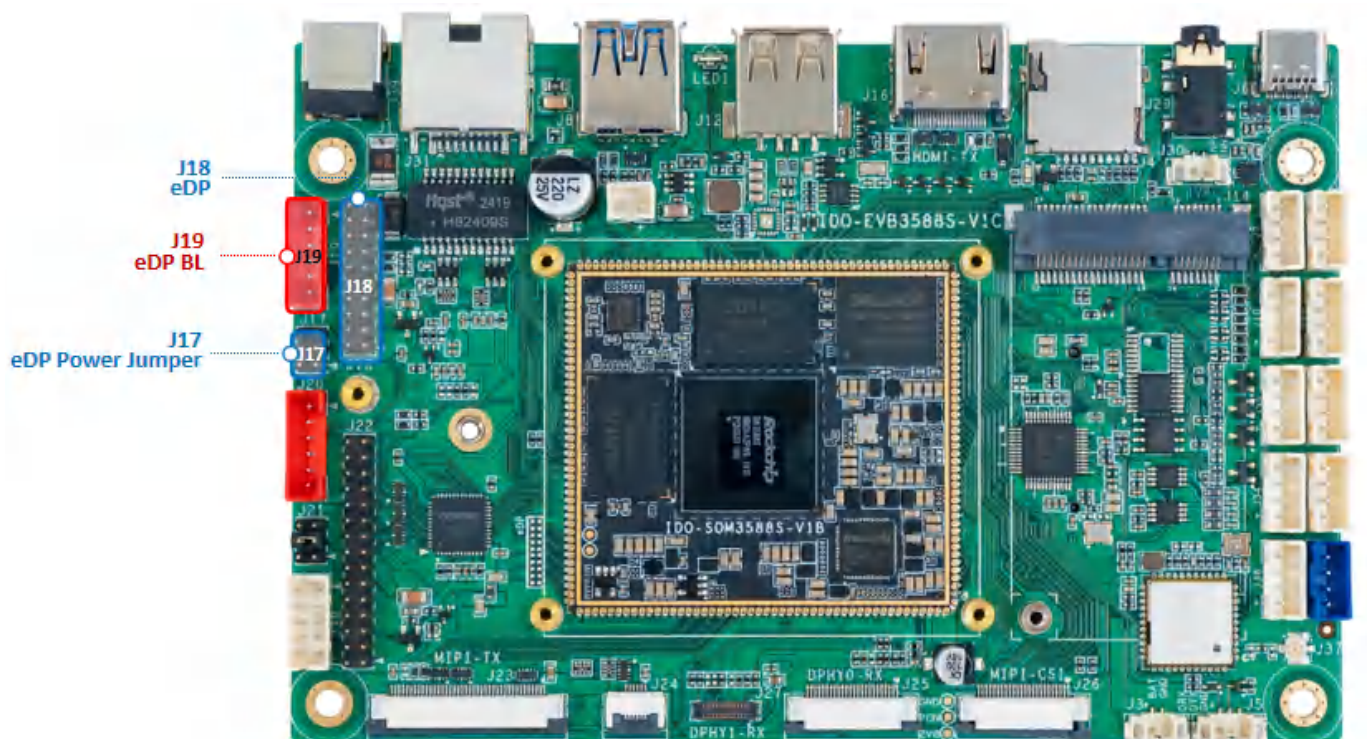
- MIPI供电为3.3V
- MIPI屏LCD排线接线：
- 触摸 TP 接口，下接
- 主板 LCD 排线接线（上接），40Pin FPC 屏座子

接法如下：



2.1.14.5. EDP

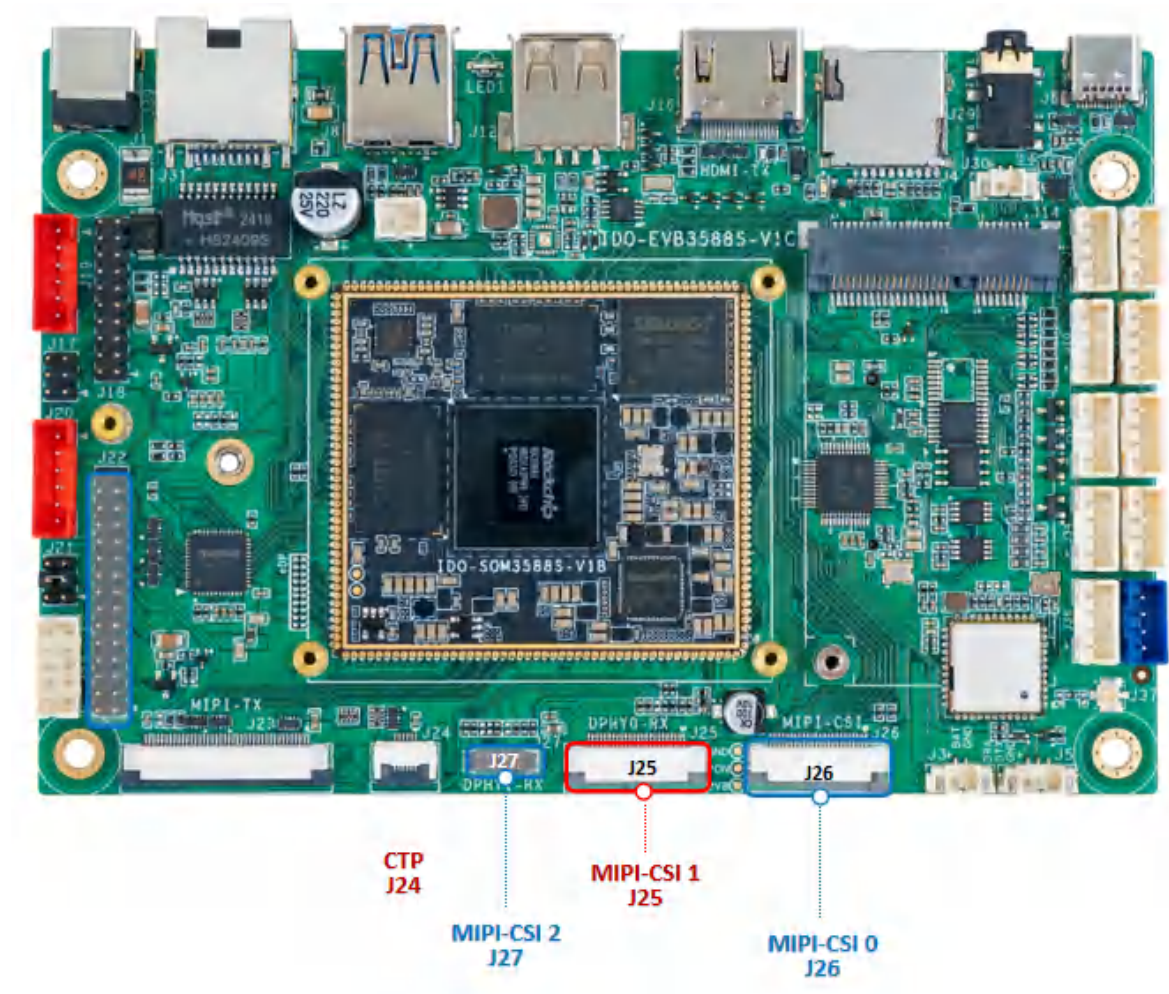
eDP接口，如下图所示：



注意：eDP与HDMI互斥，默认配置为 HDMI 功能，使用 eDP 需要更改硬件和软件

2.1.15. MIPI Camera

MIPI Camer接口J25、J26和J27，如下图所示：



MIPI摄像头接线，如下图所示：



2.1.15.1. MIPI-CSI1

2.1.15.1.1. 预览



Plain Text |

```
1  gst-launch-1.0 v4l2src device=/dev/video33 ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! videoconvert ! autovideosink sync=false
```

2.1.15.1.2. 抓图

```
1 root@linaro-alip:/# v4l2-ctl --verbose -d /dev/video33 --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
```

2.1.15.2. MIPI-CSI0

2.1.15.2.1. 预览

```
1 gst-launch-1.0 v4l2src device=/dev/video42 ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! videoconvert ! autovideosink sync=false
```

2.1.15.2.2. 抓图

```
1 root@linaro-alip:/# v4l2-ctl --verbose -d /dev/video42 --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
```

2.1.15.3. MIPI-CSI2

2.1.15.3.1. 预览

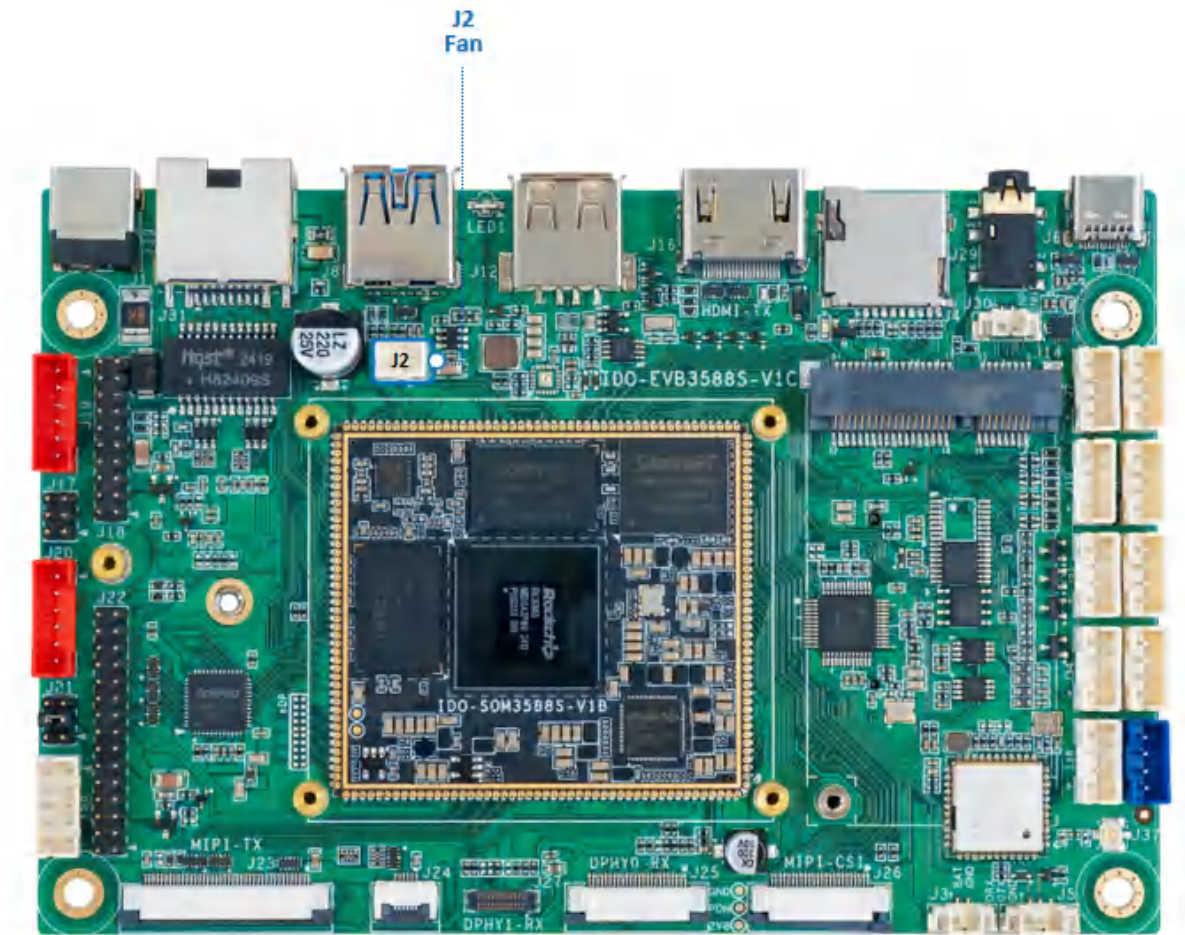
```
1 gst-launch-1.0 v4l2src device=/dev/video51 ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! videoconvert ! autovideosink sync=false
```

2.1.15.3.2. 抓图

```
1 root@linaro-alip:/# v4l2-ctl --verbose -d /dev/video51 --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
```

2.1.16. FAN 风扇

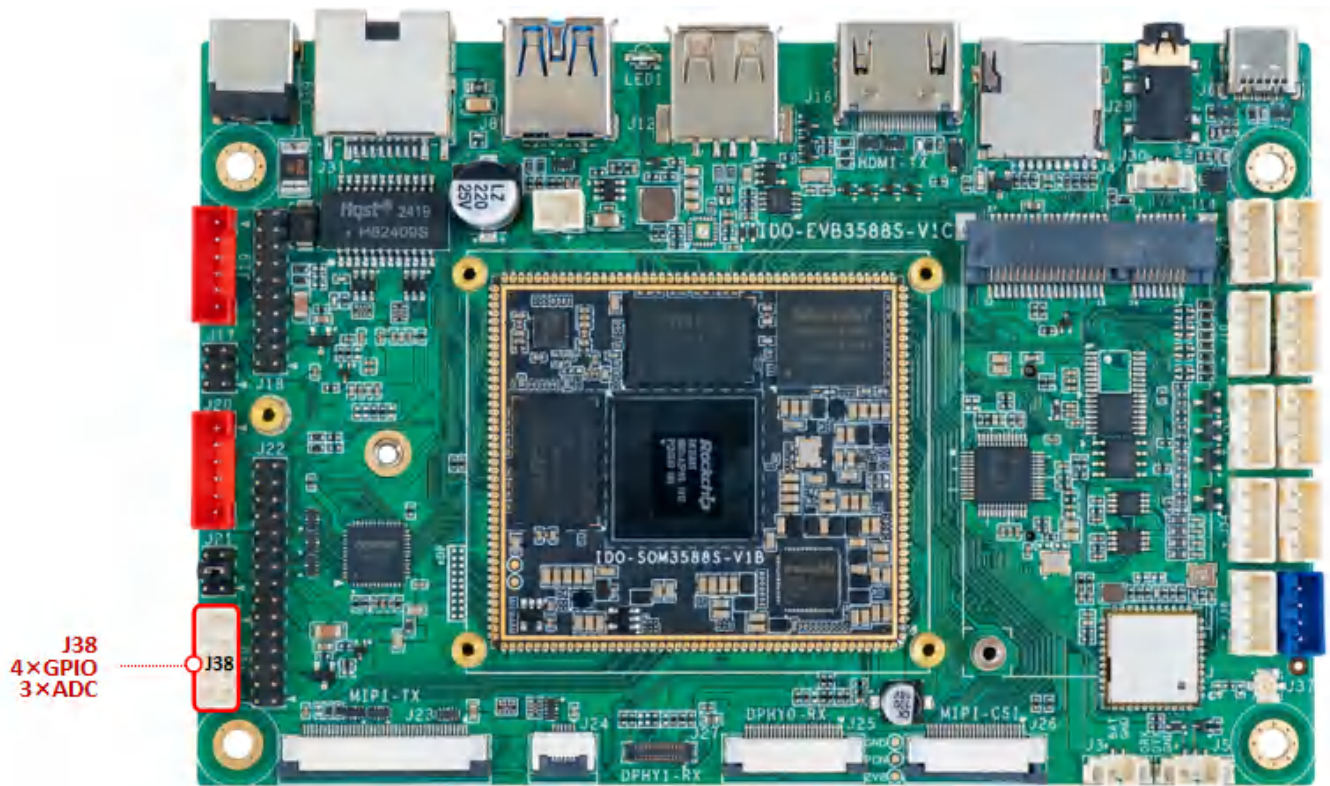
PH2 5V FAN风扇接口J2，如下图所示：



- 在CPU高于 55℃ 风扇自动开启，低于 50℃ 风扇自动关闭

2.1.17. GPIO

GPIO接口，如下图所示：



序号	管脚标号	GPIO 标号	GPIO 序号
1	CS0	GPIO1_A3	35
3	CLK	GPIO1_A2	34
5	MOSI	GPIO1_A1 (被占用，默认为 RS485)	33
7	MISO	GPIO1_A0 (被占用，默认为 RS485)	32
9	GND	地	/
2	AD1	recovery 键	/
4	AD2	ADC2	/
6	AD4	ADC4	/
8	PON	Power 键	/
10	VCC	3.3V 供电	/

GPIO 控制方式

Plain Text

```
1  输出:
2  # gpio1_A3 拉高
3  gpioset 1 3=1
4
5  # gpio1_A3 拉低
6  gpioset 1 3=0
7
8  # gpio2_A2 拉高
9  gpioset 1 2=1
10
11 # gpio2_A2 拉低
12 gpioset 1 2=0
13
14 输入:
15 # gpio2_A3 输入获取电平状态
16 gpioget 1 3
17
18 # gpio2_A2 输入获取电平状态
19 gpioget 1 2
```

2.2. GPU

执行glmark2-es2 开启gpu测试窗口:

```

1 root@linaro-alip:/# glmark2-es2
2 arm_release_ver: g24p0-00eac0, rk_so_ver: 6
3 =====
4     glmark2 2023.01
5     =====
6     OpenGL Information
7     GL_VENDOR:      ARM
8     GL_RENDERER:    Mali-G610
9     GL_VERSION:     OpenGL ES 3.2 v1.g24p0-00eac0.6b39a7384565dd0742f84d5a
    b1cdf9ef
10     Surface Config: buf=32 r=8 g=8 b=8 a=8 depth=24 stencil=0 samples=0
11     Surface Size:   800x600 windowed
12     =====
13 ▾ [build] use-vbo=false: FPS: 1039 FrameTime: 0.963 ms
14 ▾ [build] use-vbo=true: FPS: 1149 FrameTime: 0.871 ms
15 ▾ [texture] texture-filter=nearest: FPS: 1179 FrameTime: 0.848 ms
16 ▾ [texture] texture-filter=linear: FPS: 1149 FrameTime: 0.871 ms
17 ▾ [texture] texture-filter=mipmap: FPS: 1173 FrameTime: 0.853 ms
18 ▾ [shading] shading=gouraud: FPS: 1060 FrameTime: 0.944 ms
19 ▾ [shading] shading=blinn-phong-inf: FPS: 1057 FrameTime: 0.946 ms
20 ▾ [shading] shading=phong: FPS: 1032 FrameTime: 0.970 ms
21 ▾ [shading] shading=cel: FPS: 1054 FrameTime: 0.949 ms
22 ▾ [bump] bump-render=high-poly: FPS: 863 FrameTime: 1.160 ms
23 ▾ [bump] bump-render=normals: FPS: 1205 FrameTime: 0.830 ms
24 ▾ [bump] bump-render=height: FPS: 1177 FrameTime: 0.850 ms
25 ▾ [effect2d] kernel=0,1,0;1,-4,1;0,1,0;: FPS: 1206 FrameTime: 0.830 ms
26 ▾ [effect2d] kernel=1,1,1,1,1;1,1,1,1,1;1,1,1,1,1;: FPS: 964 FrameTime: 1.03
    8 ms
27 ▾ [pulsar] light=false:quads=5:texture=false: FPS: 1133 FrameTime: 0.883 ms
28 ▾ [desktop] blur-radius=5:effect=blur:passes=1:separable=true:windows=4: FPS
    : 779 FrameTime: 1.285 ms
29 ▾ [desktop] effect=shadow:windows=4: FPS: 970 FrameTime: 1.032 ms
30 ▾ [buffer] columns=200:interleave=false:update-dispersion=0.9:update-fractio
    n=0.5:update-method=map: FPS: 467 FrameTime: 2.142 ms
31 ▾ [buffer] columns=200:interleave=false:update-dispersion=0.9:update-fractio
    n=0.5:update-method=subdata: FPS: 448 FrameTime: 2.236 ms
32 ▾ [buffer] columns=200:interleave=true:update-dispersion=0.9:update-fraction
    =0.5:update-method=map: FPS: 531 FrameTime: 1.885 ms
33 ▾ [ideas] speed=duration: FPS: 964 FrameTime: 1.038 ms
34 ▾ [jellyfish] <default>: FPS: 1011 FrameTime: 0.990 ms
35 ▾ [terrain] <default>: FPS: 263 FrameTime: 3.812 ms
36 ▾ [shadow] <default>: FPS: 1012 FrameTime: 0.988 ms
37 ▾ [refract] <default>: FPS: 390 FrameTime: 2.567 ms
38 ▾ [conditionals] fragment-steps=0:vertex-steps=0: FPS: 1198 FrameTime: 0.835
    ms

```

```

39 [conditionals] fragment-steps=5:vertex-steps=0: FPS: 1179 FrameTime: 0.849
40 ms
41 [conditionals] fragment-steps=0:vertex-steps=5: FPS: 1144 FrameTime: 0.875
42 ms
43 [function] fragment-complexity=low:fragment-steps=5: FPS: 1126 FrameTime:
44 0.888 ms
45 [function] fragment-complexity=medium:fragment-steps=5: FPS: 1136 FrameTim
46 e: 0.881 ms
47 [loop] fragment-loop=false:fragment-steps=5:vertex-steps=5: FPS: 1173 Fram
48 eTime: 0.853 ms
[loop] fragment-steps=5:fragment-uniform=false:vertex-steps=5: FPS: 1122 F
rameTime: 0.892 ms
[loop] fragment-steps=5:fragment-uniform=true:vertex-steps=5: FPS: 1120 Fr
ameTime: 0.893 ms
=====
                                glmark2 Score: 983
=====

```

2.3. NPU

将rknn-toolkit2-master.tar.gz压缩包拷贝板子上，使用yolov5进行rknn模型转换测试：

2.3.1. 编译rknn_yolov5_demo

```

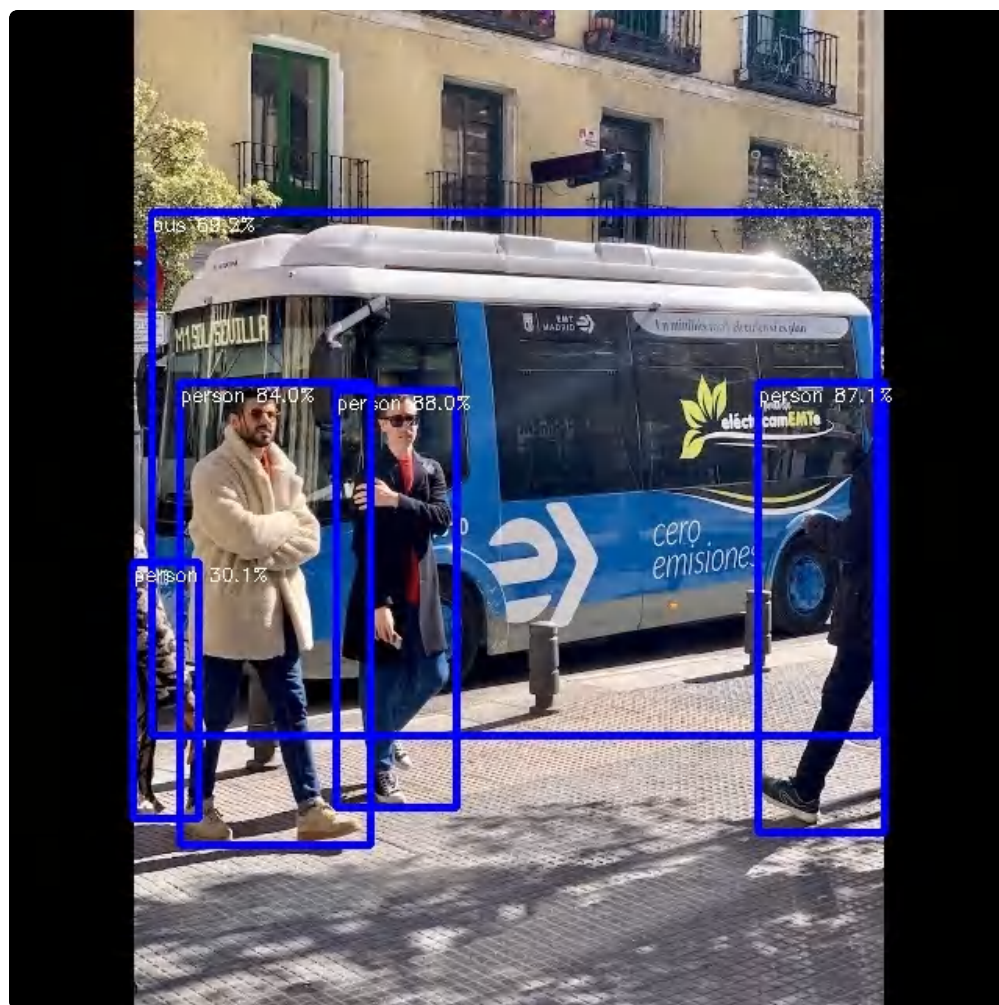
▼ Shell
1 root@linaro-alip:~$ export GCC_COMPILER=/usr/bin/aarch64-linux-gnu
2 root@linaro-alip:~$ tar -xvf rknn-toolkit2-master.tar.gz
3 root@linaro-alip:~$ cd rknn-toolkit2-master/rknpu2/examples/rknn_yolov5_demo
4 root@linaro-alip:~/rknn-toolkit2-master/rknpu2/examples/rknn_yolov5_demo$ c
hmod a+x build-linux_RK3588.sh
5 root@linaro-alip:~/rknn-toolkit2-master/rknpu2/examples/rknn_yolov5_demo$
./build-linux.sh -t rk3588 -a aarch64 -b Release

```

2.3.2. 测试

```
1 industio@ido:~/rknn-toolkit2-master/rknpu2/examples/rknn_yolov5_demo$ cd install/rknn_yolov5_demo_Linux/  
2 industio@ido:~/rknn-toolkit2-master/rknpu2/examples/rknn_yolov5_demo/install/rknn_yolov5_demo_Linux$ ./rknn_yolov5_demo ./model/RK3588/yolov5s-640-640.rknn ./model/bus.jpg  
3
```

执行完后会得到转换后的模型图片，如下所示：

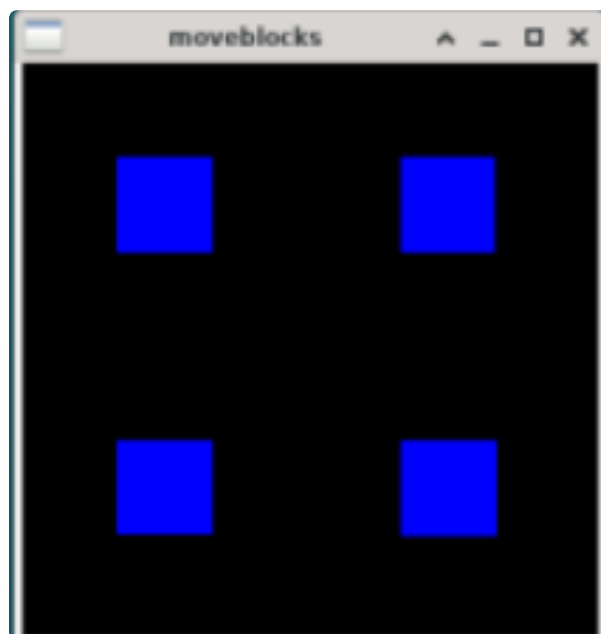


2.4. QT5

支持qt5.15，测试方法如下：

```
1 root@linaro-alip:/# source /usr/local/qt5/env.sh
2 root@linaro-alip:/# chmod a+x ./moveblocks
3 root@linaro-alip:/# ./moveblocks
```

结果如下所示：

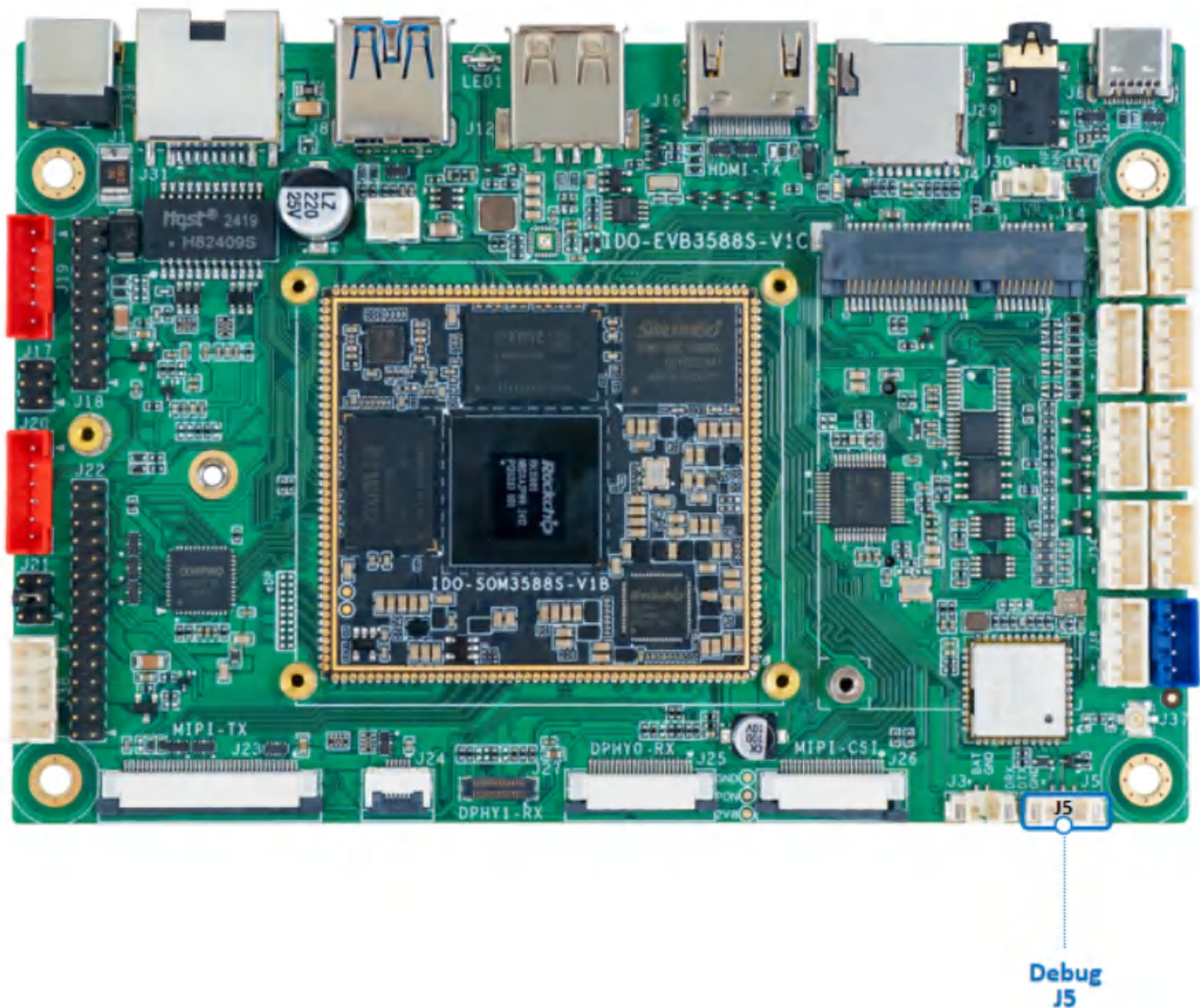


3. IDO-EVB3588S-V1C-Ubuntu系统

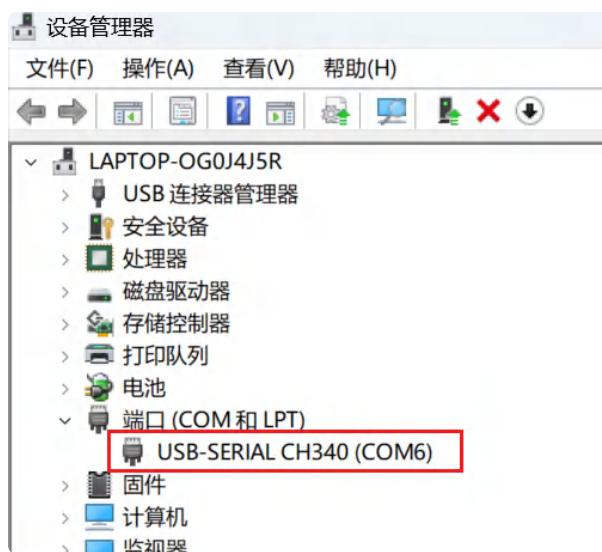
3.1. 功能及接口使用方法

3.1.1. DEBUG UART

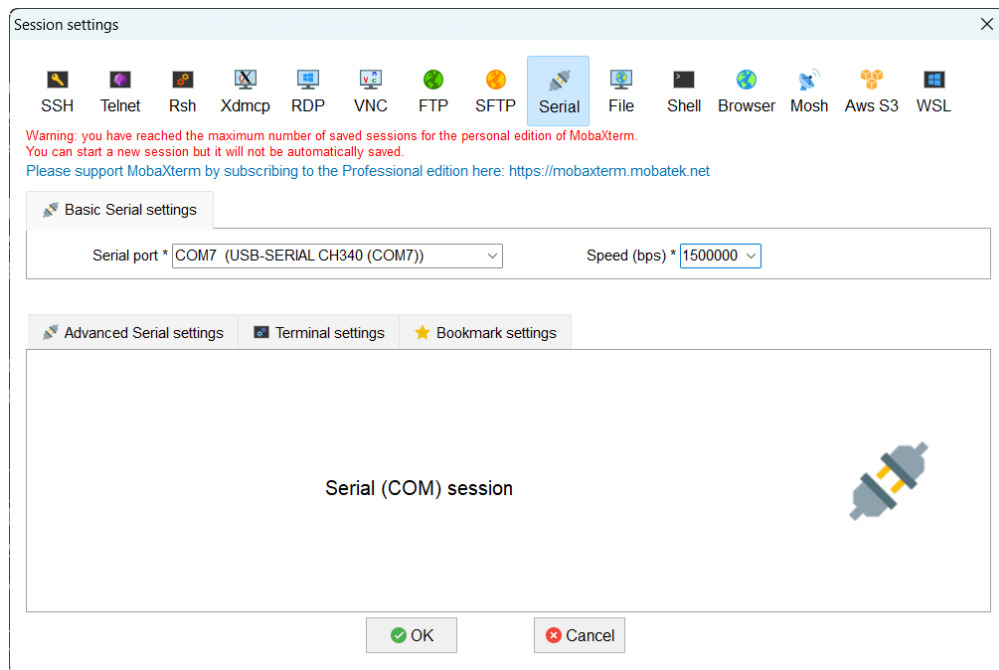
DEBUG UART位置J5，如下图所示：



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。



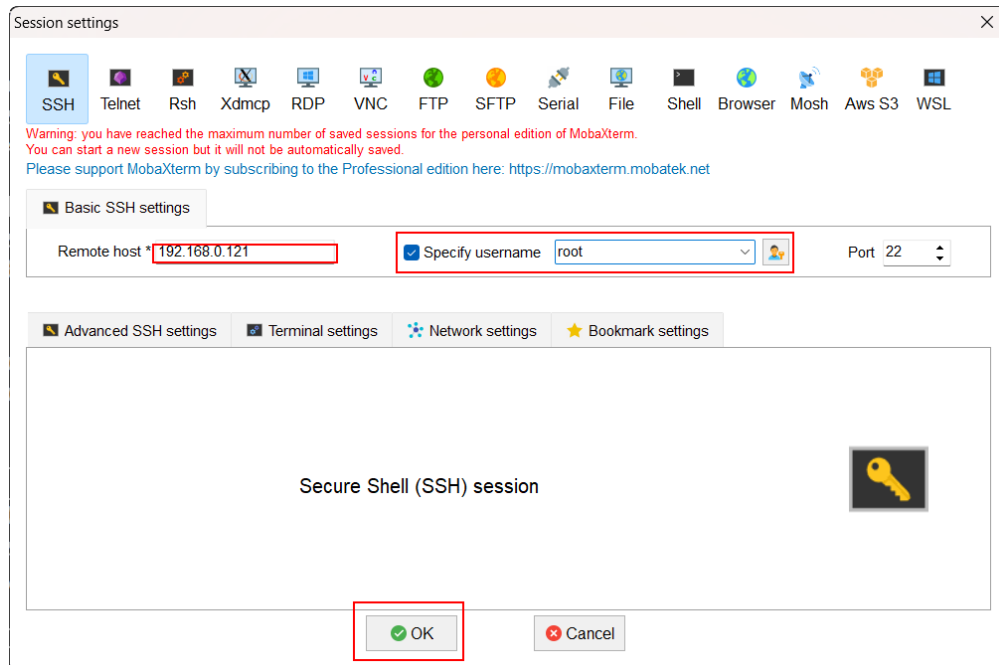
使用调试软件（MobaXterm、putty）等，以MobaXterm为例子，设置参数如下：



3.1.2. SSH登录

在知道IP的前提下，默认可以通过SSH登录进系统。

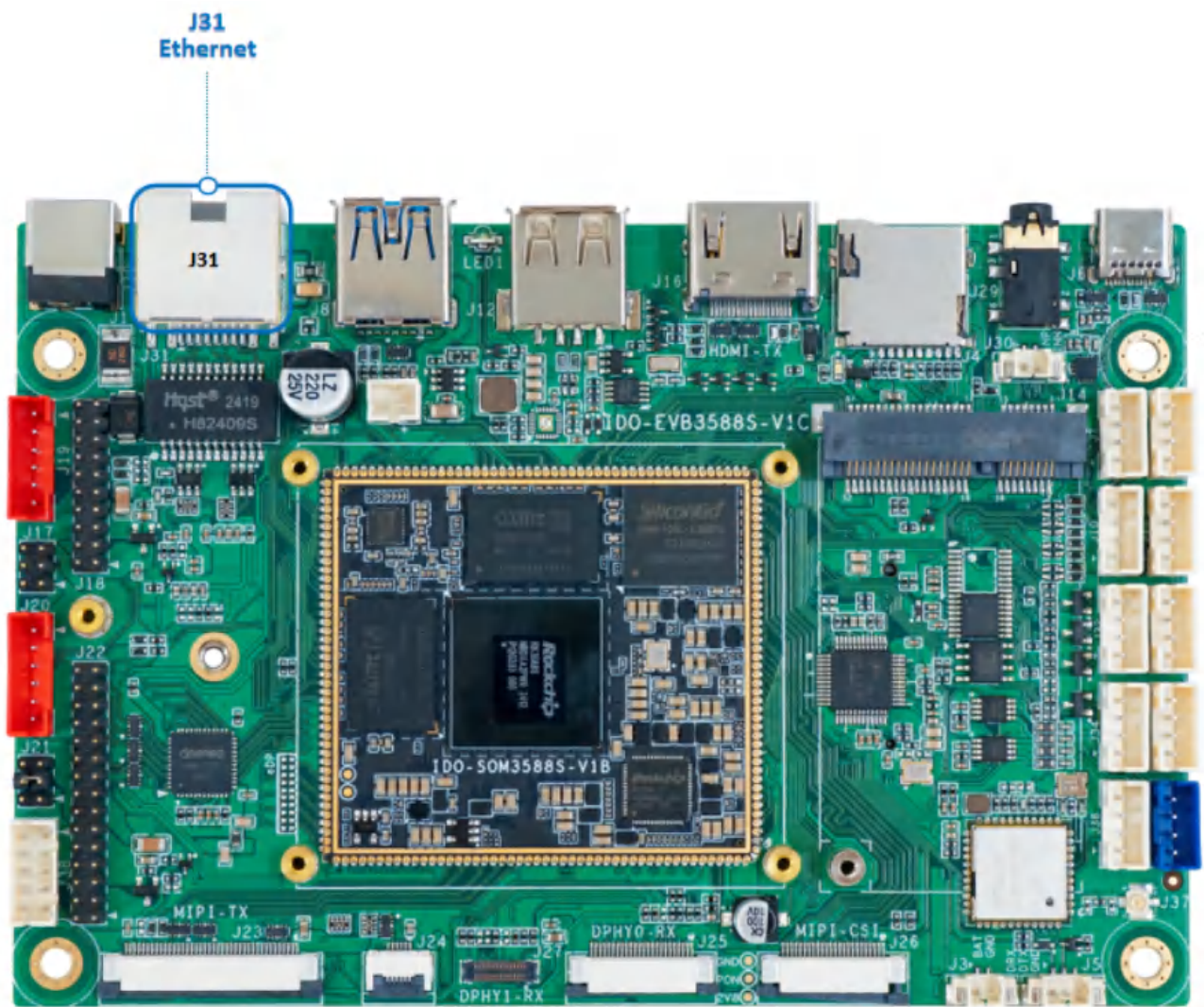
SSH登录账号密码：industio/123456



3.1.3. 网络

3.1.3.1. 以太网

主板有一路千兆以太网接口位置J31，如下图所示：



设备节点分别为eth0，以太网接口默认支持DHCP，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。网络正常连接图标，如下图所示：

3.1.3.2. Ethernet动态IP设置

```
1 #eth0
2 industio@ido:~$ ifconfig eth0
3 eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
4     inet 192.168.0.90 netmask 255.255.255.0 broadcast 192.168.0.255
5     inet6 fe80::1b83:a156:cb16:c6b0 prefixlen 64 scopeid 0x20<link>
6     ether fa:c6:ba:c7:0c:60 txqueuelen 1000 (Ethernet)
7     RX packets 141 bytes 15129 (14.7 KiB)
8     RX errors 0 dropped 4 overruns 0 frame 0
9     TX packets 89 bytes 9306 (9.0 KiB)
10    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
11    device interrupt 121
12
```

3.1.3.3. Ethernet静态IP设置

```

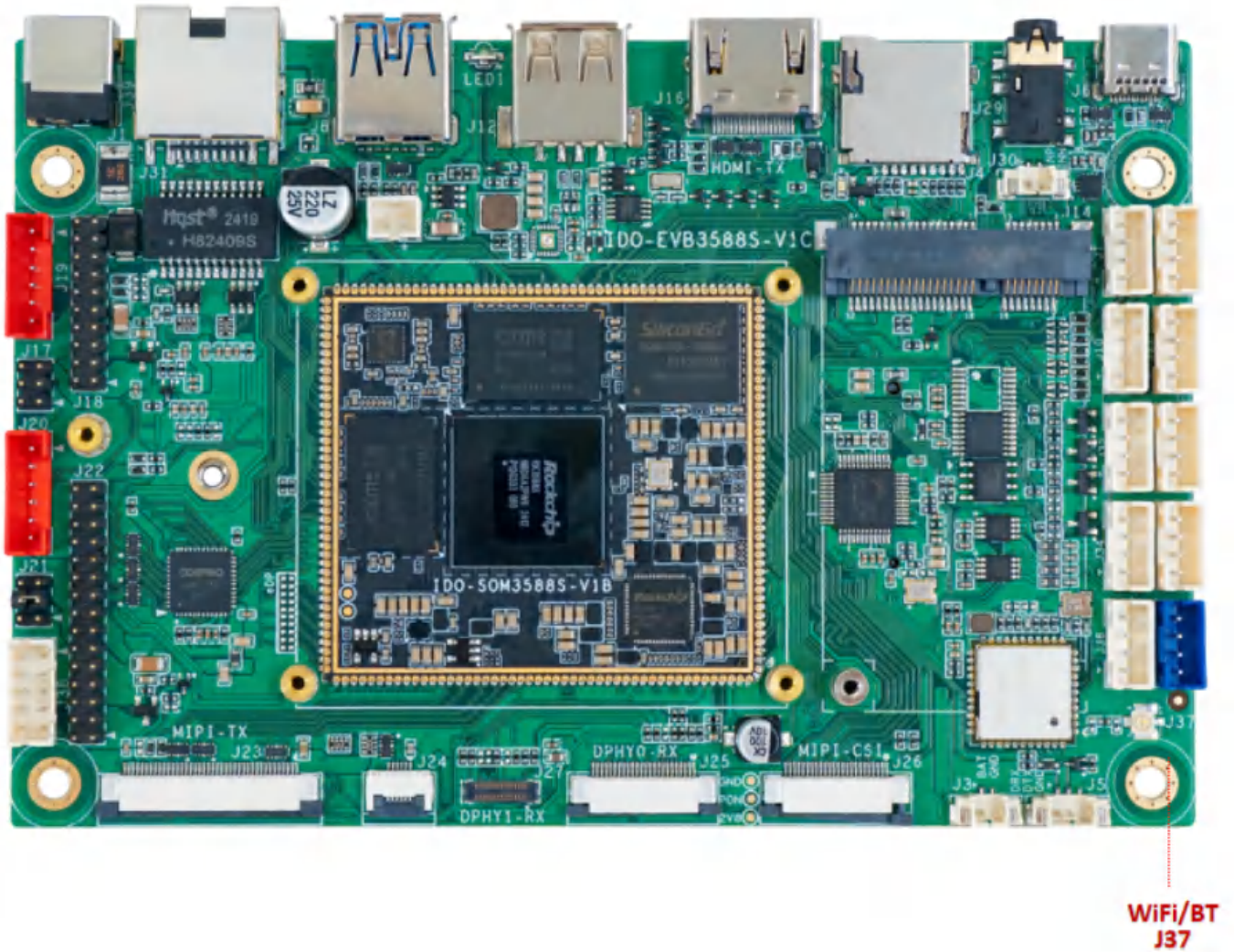
1  industio@ido:~$ touch /etc/netplan/01-netcfg.yaml
2  industio@ido:~$ vim /etc/netplan/01-netcfg.yaml
3  industio@ido:~$ cat /etc/netplan/01-netcfg.yaml
4  network:
5      version: 2
6      ethernet:
7          eth0:
8              dhcp4: no
9              addresses: [192.168.3.121/24]
10             nameservers:
11                 addresses: [192.168.3.1, 8.8.8.8, 114.114.1
12                        14.114]
13             routes:
14                 - to: 0.0.0.0/0
15                   via: 192.168.3.1
16                   metric: 100
17 #立即生效
18 industio@ido:~$ netplan apply
19 Cannot call openvswitch: ovsdb-server.service is not running.
20 industio@ido:~$ ifconfig eth0
21 eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
22     inet 192.168.3.121 netmask 255.255.255.0 broadcast 192.168.3.255
23     inet6 fe80::1b83:a156:cb16:c6b0 prefixlen 64 scopeid 0x20<link>
24     inet6 fe80::f8c6:baff:fec7:c60 prefixlen 64 scopeid 0x20<link>
25     ether fa:c6:ba:c7:0c:60 txqueuelen 1000 (Ethernet)
26     RX packets 4203 bytes 417790 (407.9 KiB)
27     RX errors 0 dropped 164 overruns 0 frame 0
28     TX packets 392 bytes 32299 (31.5 KiB)
29     TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
30     device interrupt 121

```

设备断电重启，此静态IP设置仍然生效。

3.1.4. 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J37，如下图所示：



命令行可以使用nmcli工具连接wifi热点：

```
1  #查看网络接口状态：
2  industio@ido:~$ nmcli device
3  #使用以下命令查看当前可用的 WiFi 网络：
4  industio@ido:~$ nmcli device wifi list
5  #连接到 WiFi 网络：
6  industio@ido:~$ nmcli device wifi connect 账号 password 密码
7  #确认连接状态：
8  industio@ido:~$ nmcli connection show --active
```

查看wlan0的IP地址，确认连接成功：

```
1  industio@ido:~$ ifconfig wlan0
2  wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
3          inet 192.168.0.118  netmask 255.255.255.0  broadcast 192.168.0.255
4          inet6 fe80::1036:80b4:5082:7440  prefixlen 64  scopeid 0x20<link>
5          ether c0:f5:35:12:ad:a2  txqueuelen 1000  (Ethernet)
6          RX packets 170  bytes 22149 (21.6 KiB)
7          RX errors 0  dropped 5  overruns 0  frame 0
8          TX packets 24  bytes 2683 (2.6 KiB)
9          TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
10
11  industio@ido:~$ ping www.baidu.com
12  PING www.wshifen.com (103.235.47.188) from 192.168.0.118 wlan0: 56(84) bytes of data.
13  64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=1 ttl=45 time=274 ms
14  64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=2 ttl=45 time=299 ms
15  64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=3 ttl=45 time=320 ms
16  64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=4 ttl=45 time=241 ms
```

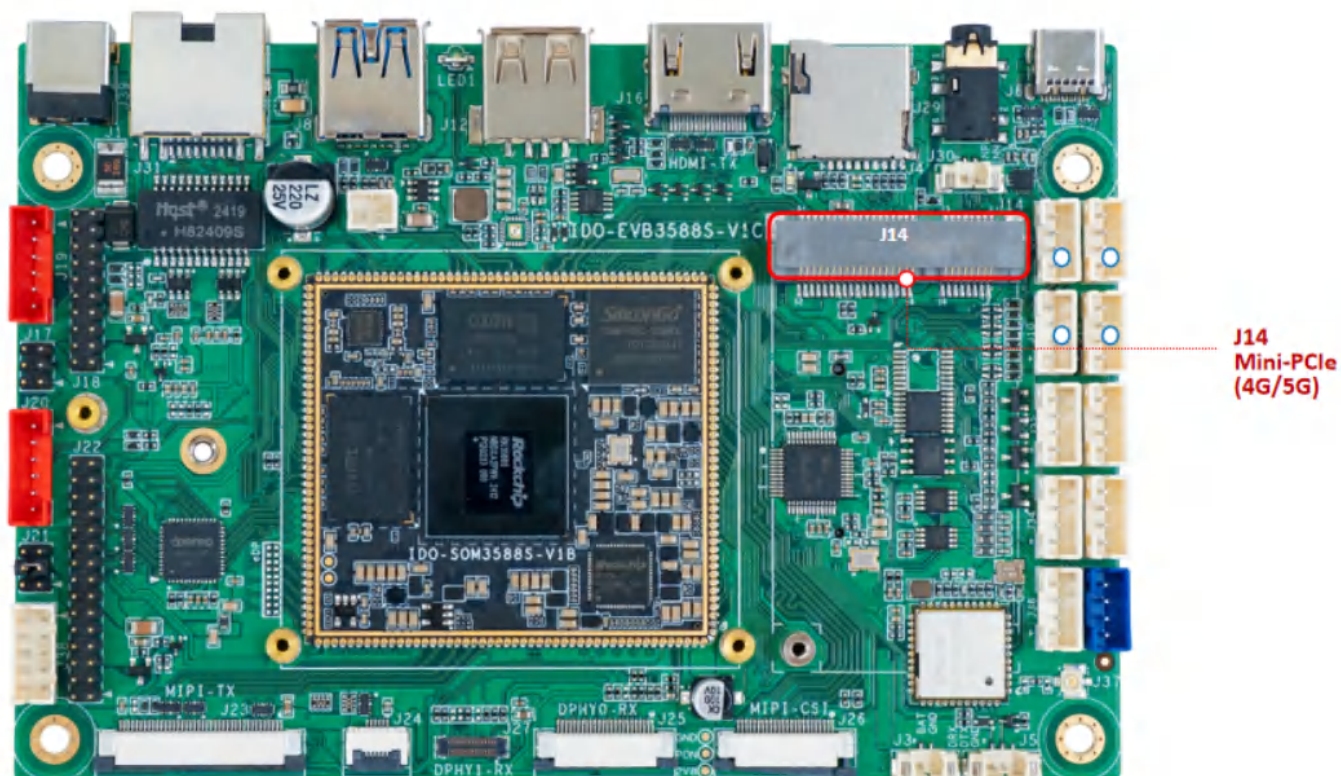
3.1.5. Bluetooth

3.1.5.1. 连接方法

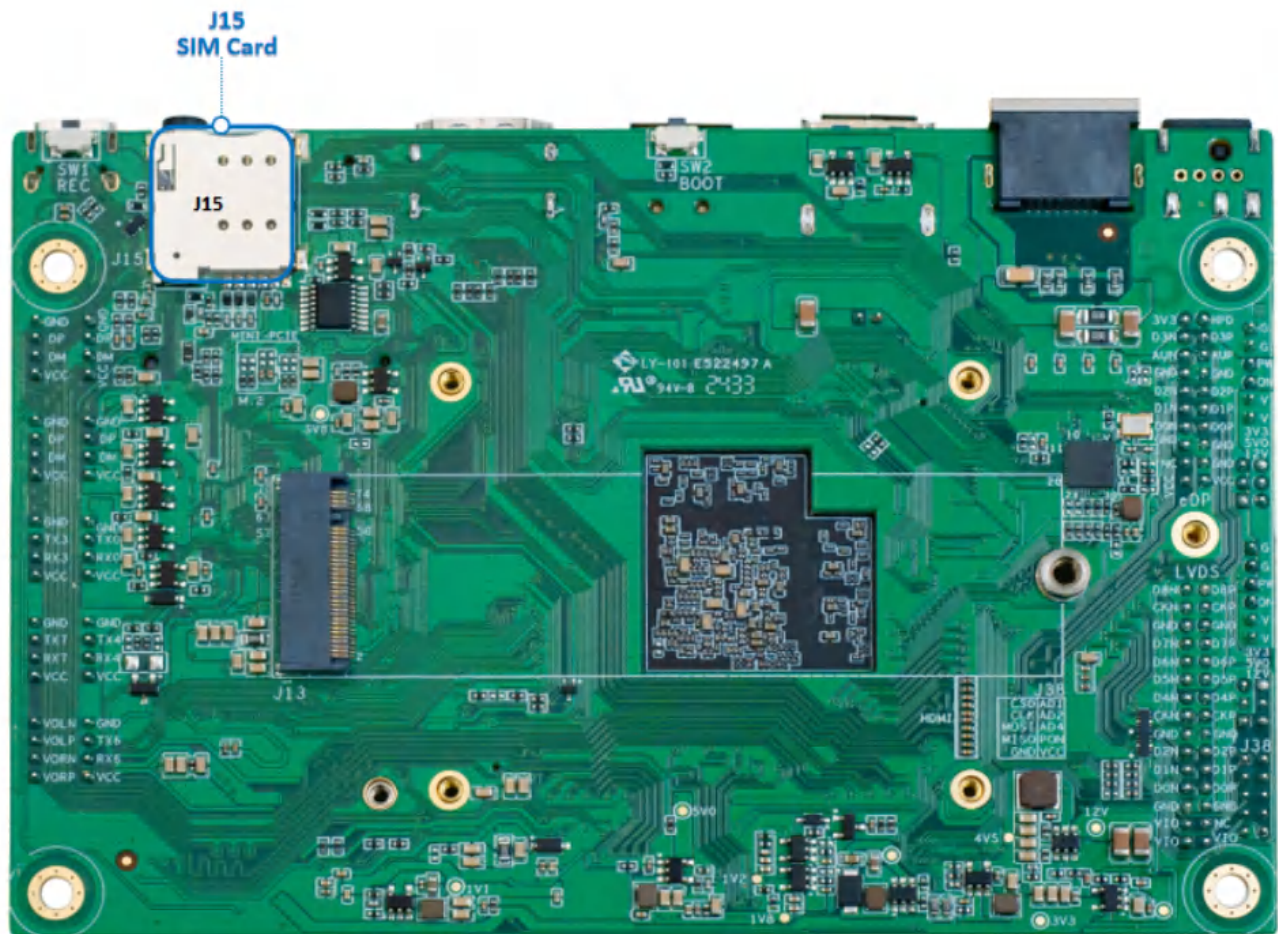
```
1  #打开蓝牙
2  industio@ido:~$ bluetoothctl
3  power on
4  #扫描蓝牙
5  scan on
6  #查看蓝牙设备
7  devices
8  #信任蓝牙设备
9  trust 7C:C1:80:09:DD:6C
10 #蓝牙配对
11 pair 7C:C1:80:09:DD:6C
12 #连接蓝牙
13 connect 7C:C1:80:09:DD:6C
```

3.1.6. 4G/5G

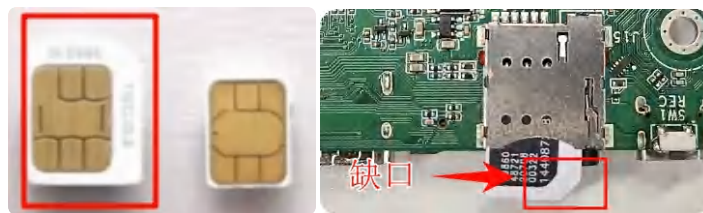
主板内置Mini PCIe 座子扩展 4G/5G模块，4G通信模块适配移远EC20、EC25等通用模组。5G通信模块适配移远RG200U-CN。模块安装位J14，如下图所示：



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J15位于主板背面，如下图所示：



使用Micro尺寸SIM卡，如下图所示：



4G使用 `ifconfig wwan0` 命令可查看到相关网络信息：

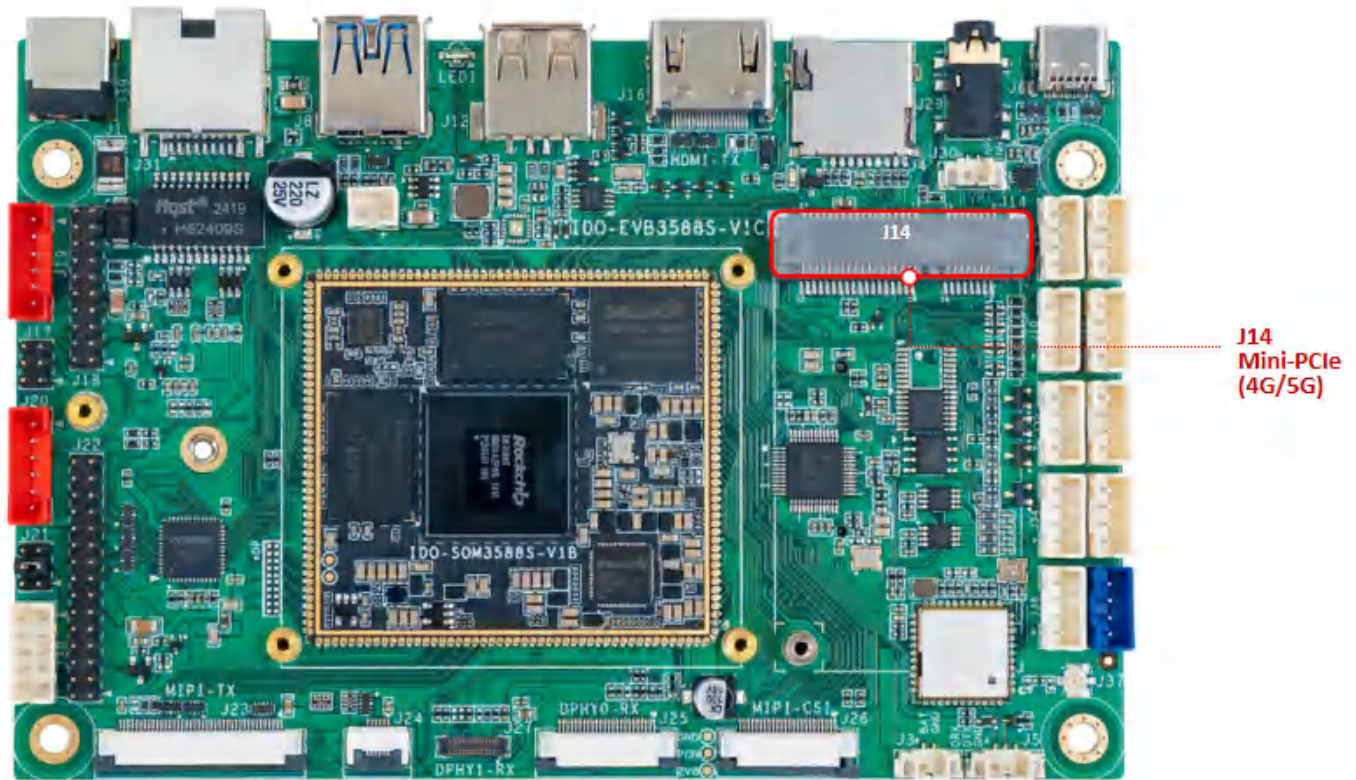
```
1  #拨号,需要进入root用户执行如下命令
2  root@idoroot@ido:~$ quectel-CM &
3
4  #查看网络
5  root@ido:~$ ifconfig wwan0
6  wwan0: flags=193<UP,RUNNING,NOARP>  mtu 1500
7          inet 10.101.61.51  netmask 255.255.255.248
8          inet6 fe80::fc:f6ff:fe8d:bab6  prefixlen 64  scopeid 0x20<link>
9          ether 02:fc:f6:8d:ba:b6  txqueuelen 1000  (Ethernet)
10         RX packets 42  bytes 7013 (6.8 KiB)
11         RX errors 0  dropped 0  overruns 0  frame 0
12         TX packets 57  bytes 4608 (4.5 KiB)
13         TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
14
15  #测试通信
16  root@ido:~$ ping www.baidu.com -I wwan0
```

5G使用 `ifconfig usb0` 命令可查看到相关网络信息:

```
1  #拨号
2  industio@ido:~$ quectel-CM &
3
4  #查看网络
5  industio@ido:~$ ifconfig usb0
6  usb0: flags=193<UP,RUNNING,NOARP>  mtu 1500
7          inet 10.101.61.51  netmask 255.255.255.248
8          inet6 fe80::fc:f6ff:fe8d:bab6  prefixlen 64  scopeid 0x20<link>
9          ether 02:fc:f6:8d:ba:b6  txqueuelen 1000  (Ethernet)
10         RX packets 42  bytes 7013 (6.8 KiB)
11         RX errors 0  dropped 0  overruns 0  frame 0
12         TX packets 57  bytes 4608 (4.5 KiB)
13         TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
14
15  #测试通信
16  industio@ido:~$ ping www.baidu.com -I usb0
```

3.1.7. mSATA

mSATA与4G/5G复用接口，如下图所示：

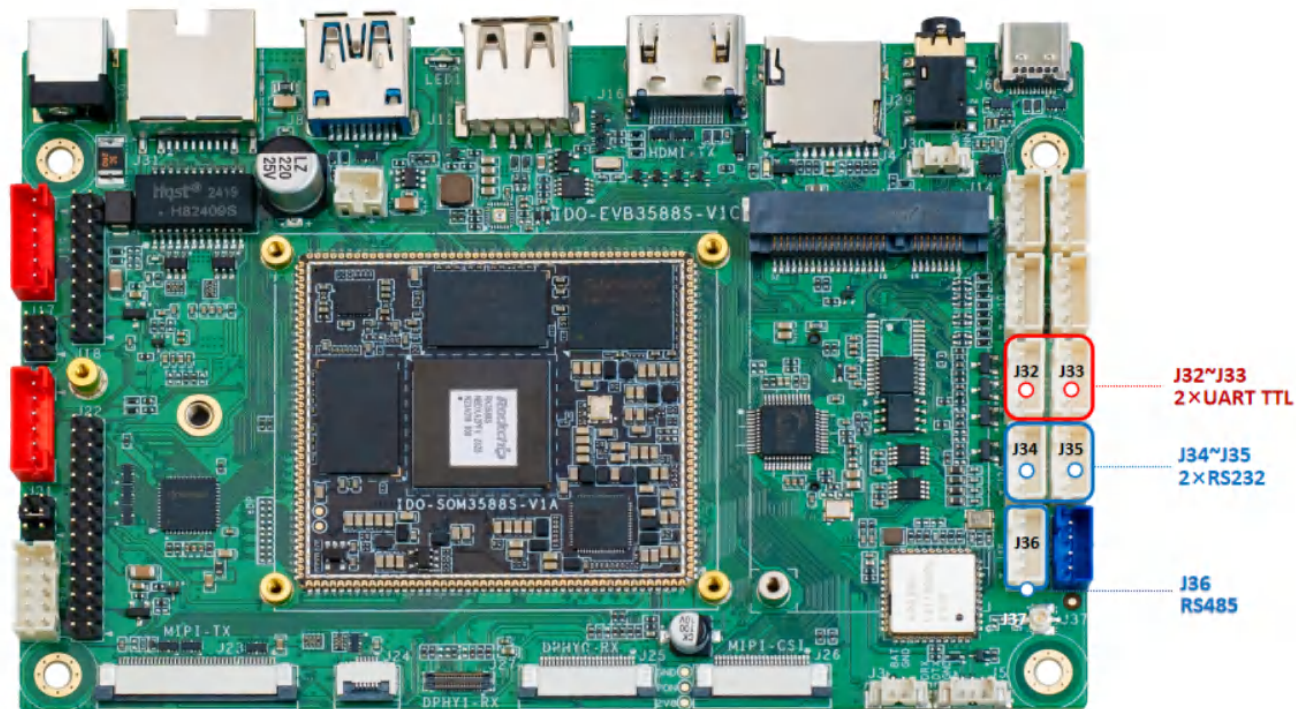


注意：

1. 默认配置为 4G，支持 mSATA 需要修改软硬件

3.1.8. UART

主板一共引出5路串口（不含调试串口）,如下图所示：



UART设备节点列表如下：

序号	功能	设备节点
0 (J32)	TTL	/dev/ttyS0
3 (J33)	TTL	/dev/ttyS3
4 (J34)	RS232	/dev/ttyS4
7 (J35)	RS232	/dev/ttyS7
6 (J36)	RS485	/dev/ttyS6

```

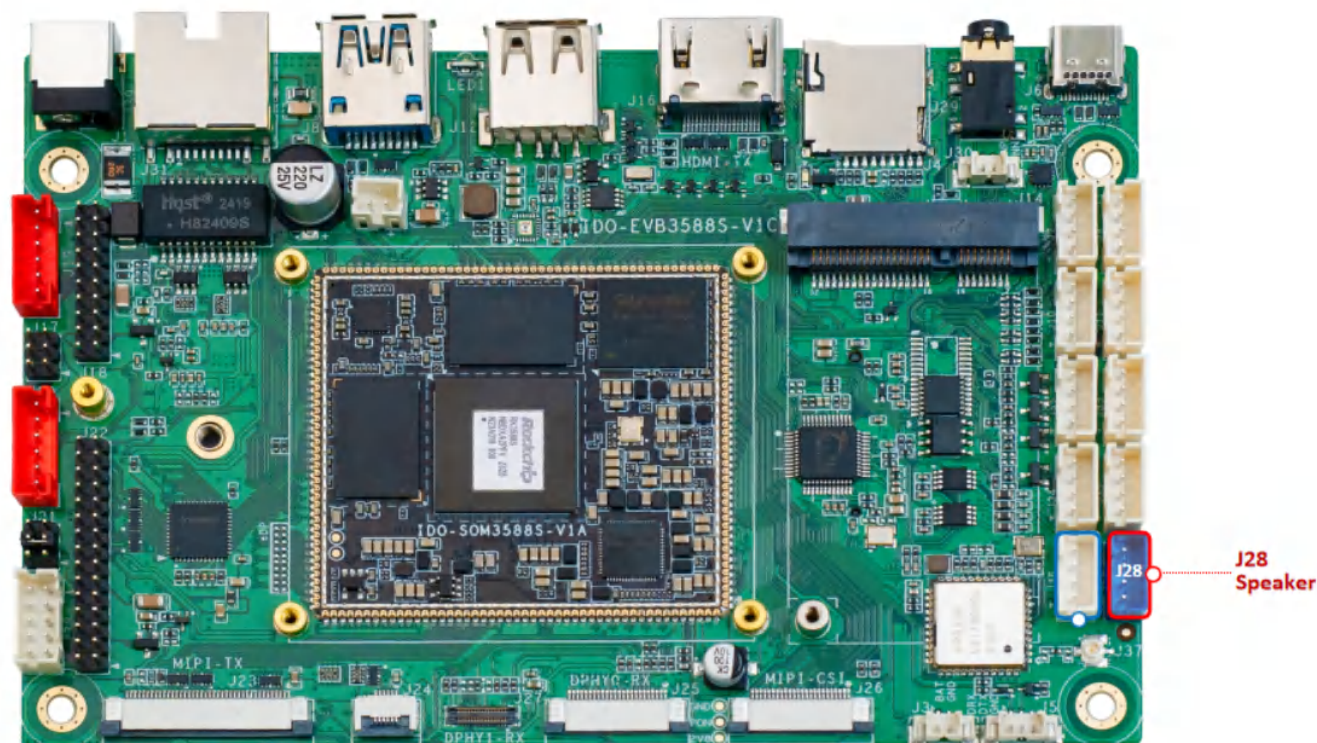
1  industio@ido:~$ microcom -p /dev/ttyS4 -s 115200

```

3.1.9. 声音

3.1.9.1. 喇叭

喇叭接口为PH2.0-4P连接器J28，如下图所示：



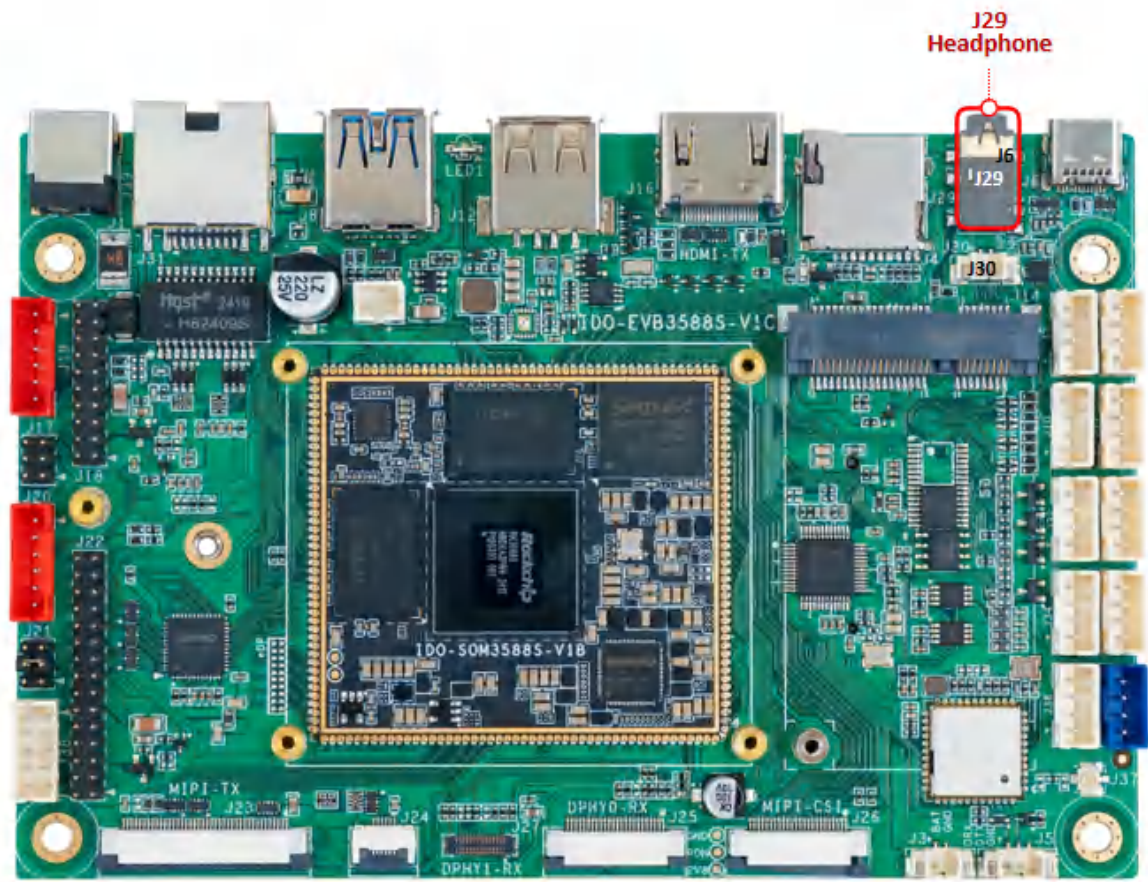
- 支持双声道，每个声道支持4Ω 3W输出

Plain Text

```
1 #查看声卡
2 industio@ido:~$ aplay -l
3
4 #音频播放到DP
5 industio@ido:~$ aplay -D hw:1,0 xihuangni.wav
6 Playing WAVE 'xihuangni.wav' : Signed 16 bit Little Endian, Rate 44100 H
  z, Stereo
7
8 #音频播放到HDMI
9 industio@ido:~$ aplay -D hw:2,0 xihuangni.wav
10 Playing WAVE 'xihuangni.wav' : Signed 16 bit Little Endian, Rate 44100 H
  z, Stereo
11
12 #音频播放到声卡
13 industio@ido:~$ aplay -D hw:0,0 xihuangni.wav
14 Playing WAVE 'xihuangni.wav' : Signed 16 bit Little Endian, Rate 44100 H
  z, Stereo
```

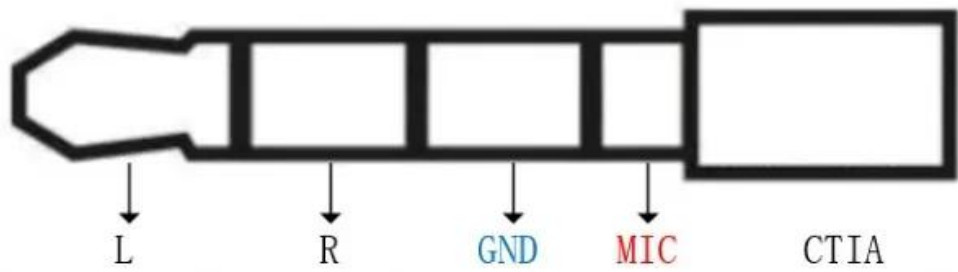
3.1.9.2. 耳机

主板支持1路3.5mm四节耳机座J29，如下图所示：



- 支持耳机检测
- 支持耳机录音

CTIA标准四段式耳机，定义如下：

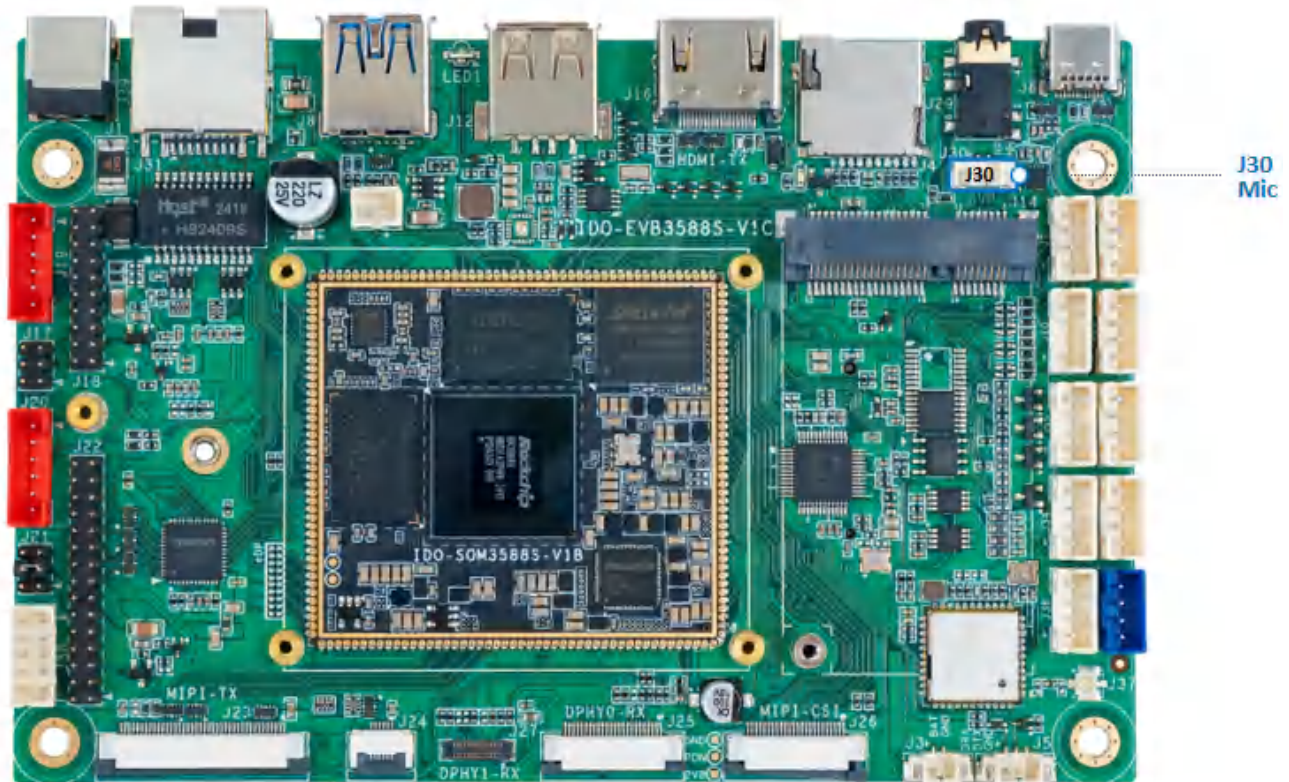


耳机录音：

```
1 #耳机录音
2 root@ido:/# amixer -c 0 cset numid=43 on
3 root@ido:/# echo -n "mic1" > /sys/class/es8388_mic/mic_channel
4
5 root@ido:/# arecord -D hw:0,0 -f cd -r 44100 -c 2 test.wav
6 Recording WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

3.1.9.3. Mic

MX1.25T-2P麦克风接口J30，如下图所示：

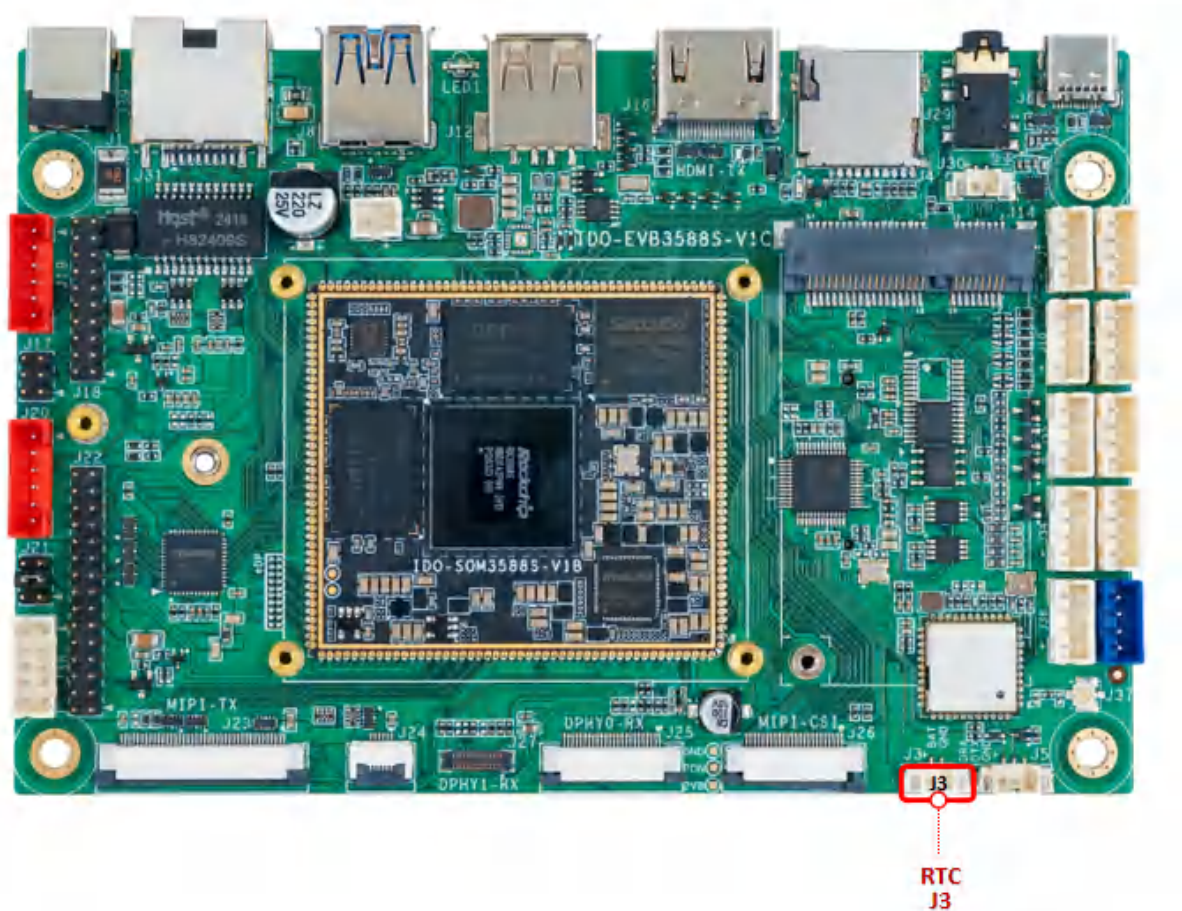


Mic录音：

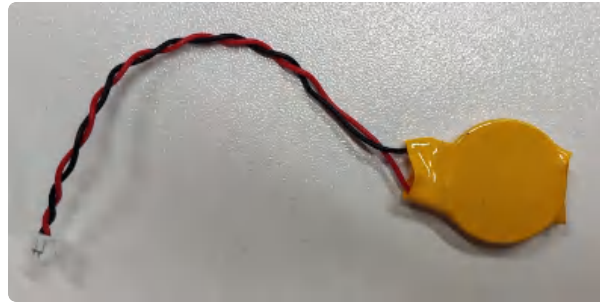
```
1 #Mic录音
2 root@ido:/# amixer -c 0 cset numid=43 on
3 root@ido:/# echo -n "mic2" > /sys/class/es8388_mic/mic_channel
4
5 root@ido:/# arecord -D hw:0,0 -f S16_LE -r 44100 -c 2 test.wav
6 Recording WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

3.1.10. RTC

MX1.25T-2P RTC电池座J3，如下图所示：



连接3V 纽扣电池，RTC电池参考如下



rtc时间设置方法：

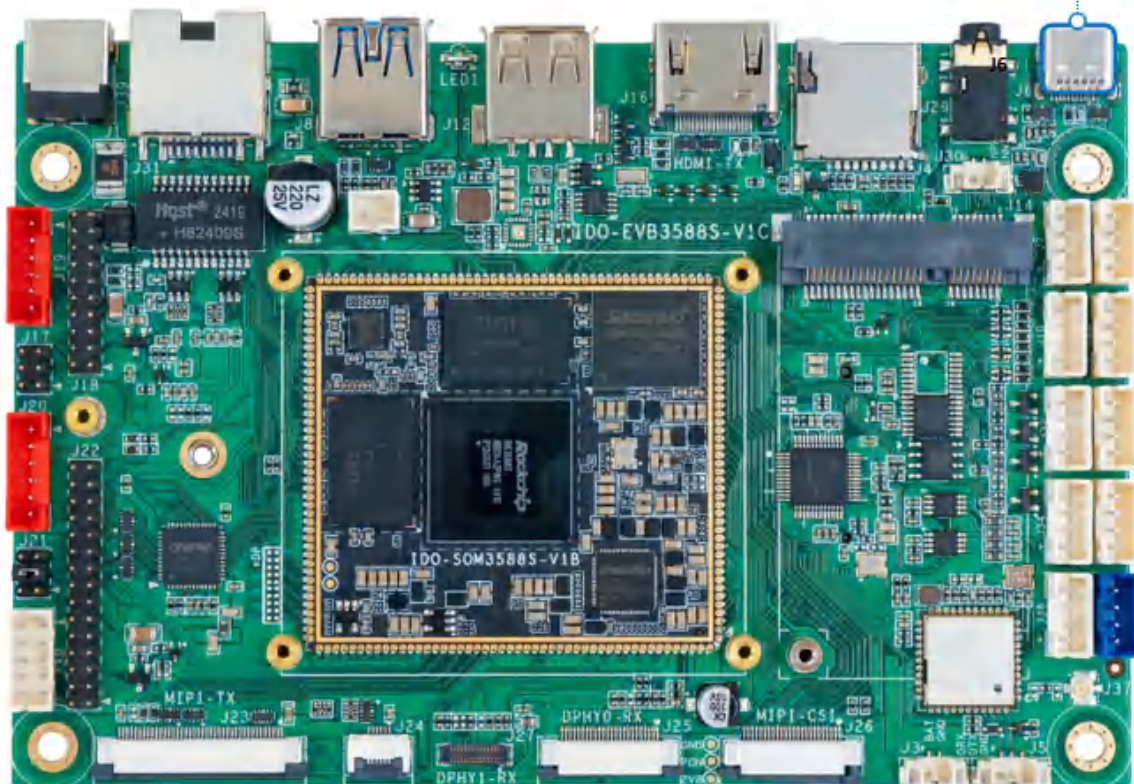
```
1  #设置时间
2  industio@ido:~$ sudo date -s "2024-10-09 14:02:30"
3
4  #将rtc时钟调整为与目前的系统时钟一致
5  industio@ido:~$ sudo hwclock -w
6
7  #获取硬件rtc当前时间（断电重启读取时间没有太大偏差）
8  industio@ido:~$ sudo hwclock -r
9  2024-10-09 14:02:35.945604+00:00
```

3.1.11. USB接口

主板支持1个TypeC接口，支持2个USB-A，4个USB-PH2.0接口，USB对外总供电应小于3A。

3.1.11.1. TypeC 接口

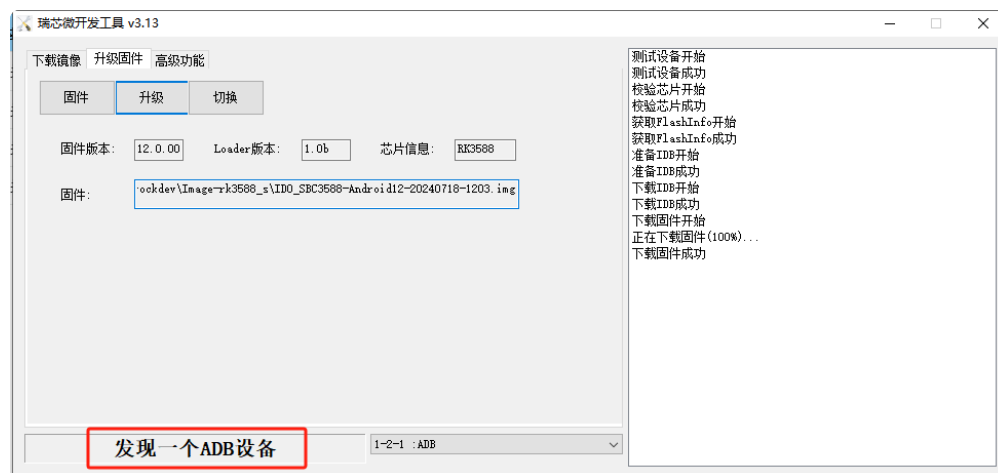
TypeC接口J6，如下图所示：



- 支持Host、Device模式自动切换
- 支持DP显示输出

3.1.11.1.1. 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：



- 可使用ADB相关开发工具对盒子进行功能调试

3.1.11.1.2. 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：



- 可识别U盘、键盘、鼠标并正常使用

3.1.11.1.3. DP模式

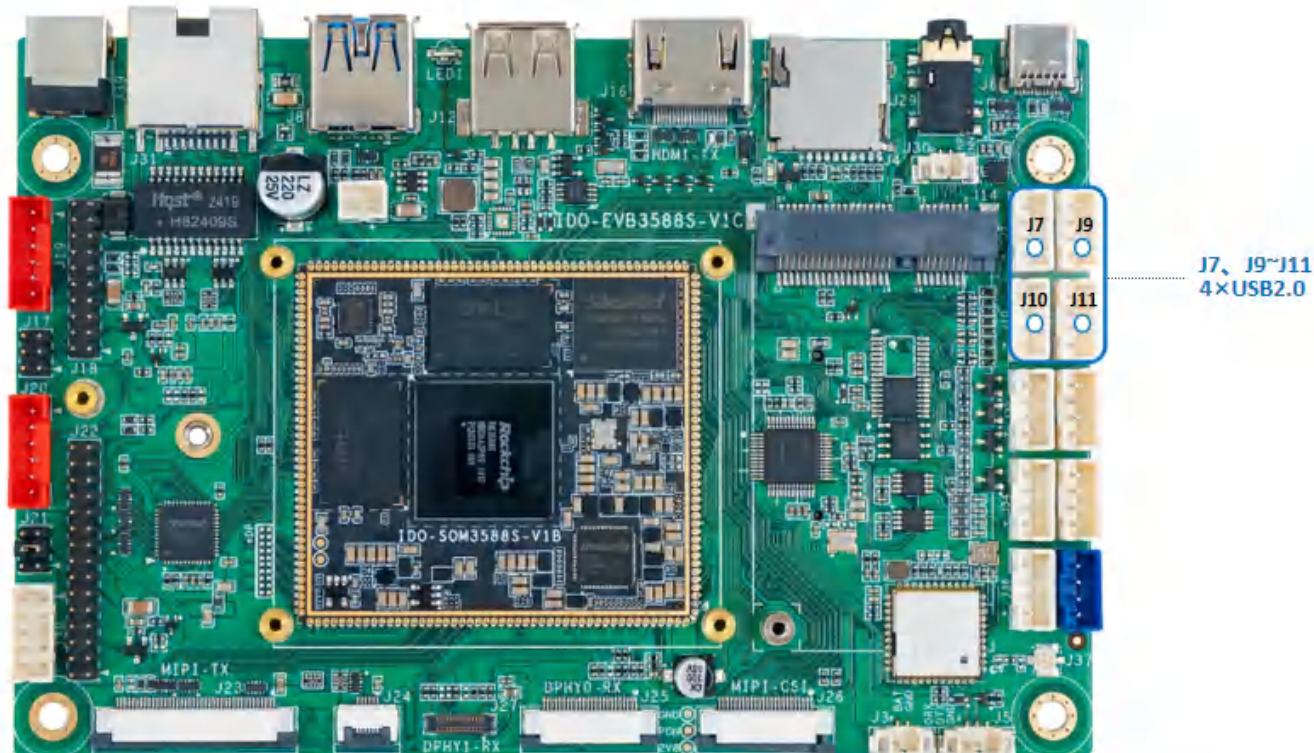
通过TypeC全功能数据线接入DP显示器，或通过TYPE-C to HDMI数据线连接HDMI显示器



- 主板画面通过TYPE-C DP输出正常，DP声音输出正常

3.1.11.2. USB PH2.0

支持四个USB PH2.0，如下图所示：



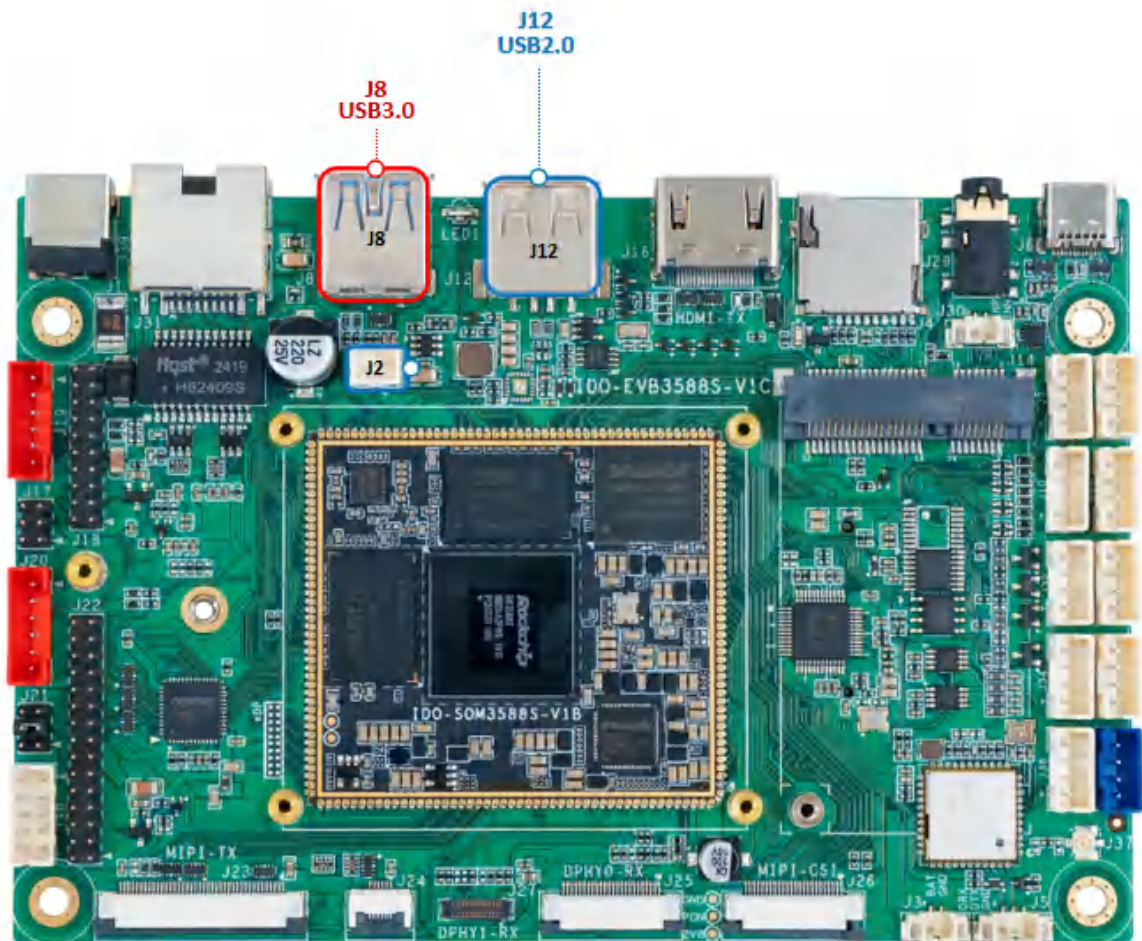
USB电源控制，如下表所示：

USB端口	默认状态	动作	命令
J7	开启	关闭电源	echo 0 > /sys/class/leds/usb_j7/brightness
		开启电源	echo 1 > /sys/class/leds/usb_j7/brightness
J9	开启	关闭电源	echo 0 > /sys/class/leds/usb_j9/brightness
		开启电源	echo 1 > /sys/class/leds/usb_j9/brightness
J10	开启	关闭电源	echo 0 > /sys/class/leds/usb_j10/brightness
		开启电源	echo 1 > /sys/class/leds/usb_j10/brightness
J11	开启	关闭电源	echo 0 > /sys/class/leds/usb_j11/brightness
		开启电源	echo 1 > /sys/class/leds/usb_j11/brightness

供电控制说明：设备节点写"0"关闭电源，写"1"开启电源。

3.1.11.3. USB接口

USB-A接口，如下图所示：



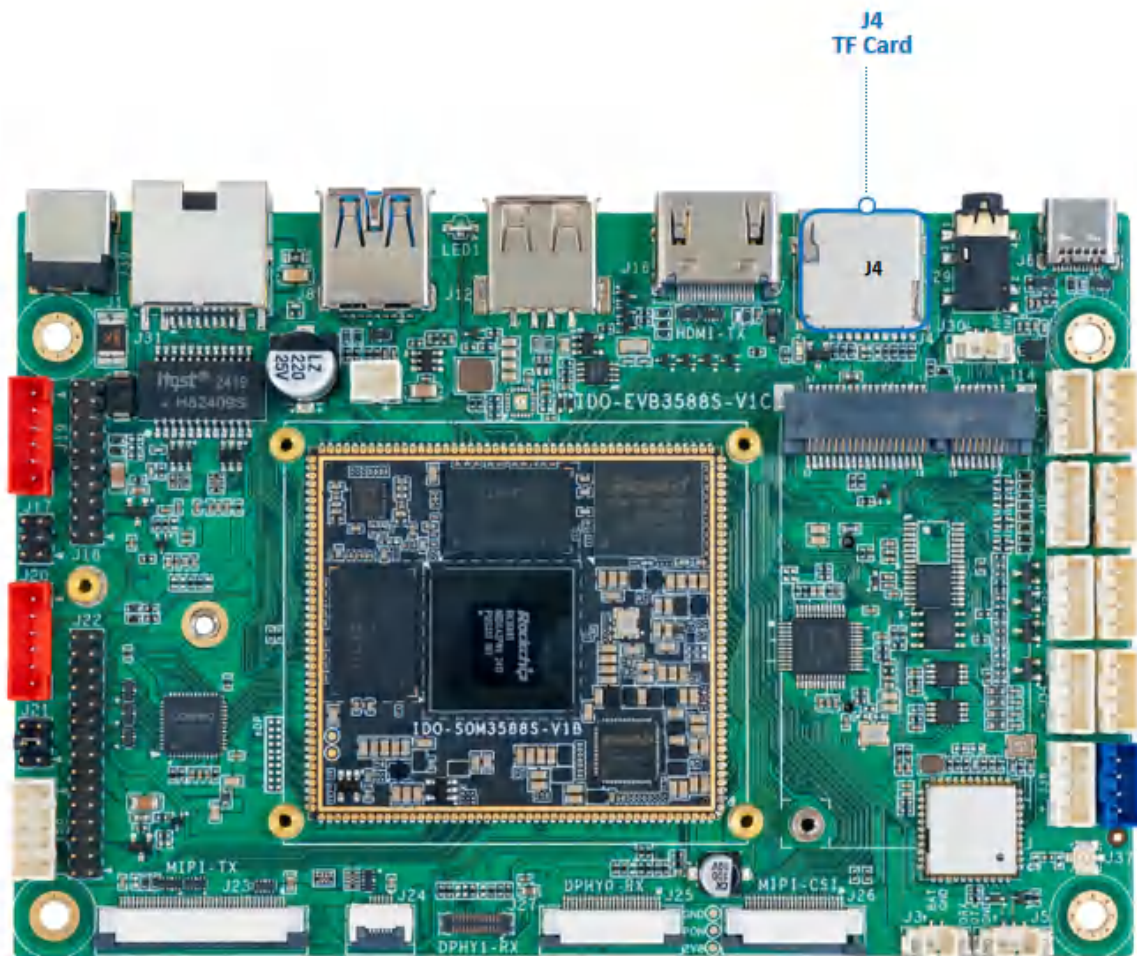
USB电源控制，如下表所示：

USB端口	默认状态	动作	命令
J8	开启	关闭电源	<code>echo 0 > /sys/class/leds/usb_j8/brightness</code>
		开启电源	<code>echo 1 > /sys/class/leds/usb_j8/brightness</code>
J12	开启	关闭电源	<code>echo 0 > /sys/class/leds/usb_j12/brightness</code>
		开启电源	<code>echo 1 > /sys/class/leds/usb_j12/brightness</code>

供电控制说明：设备节点写"0"关闭电源，写"1"开启电源。

3.1.12. TF Card

TF卡座接口位于J4，如下图所示：



```

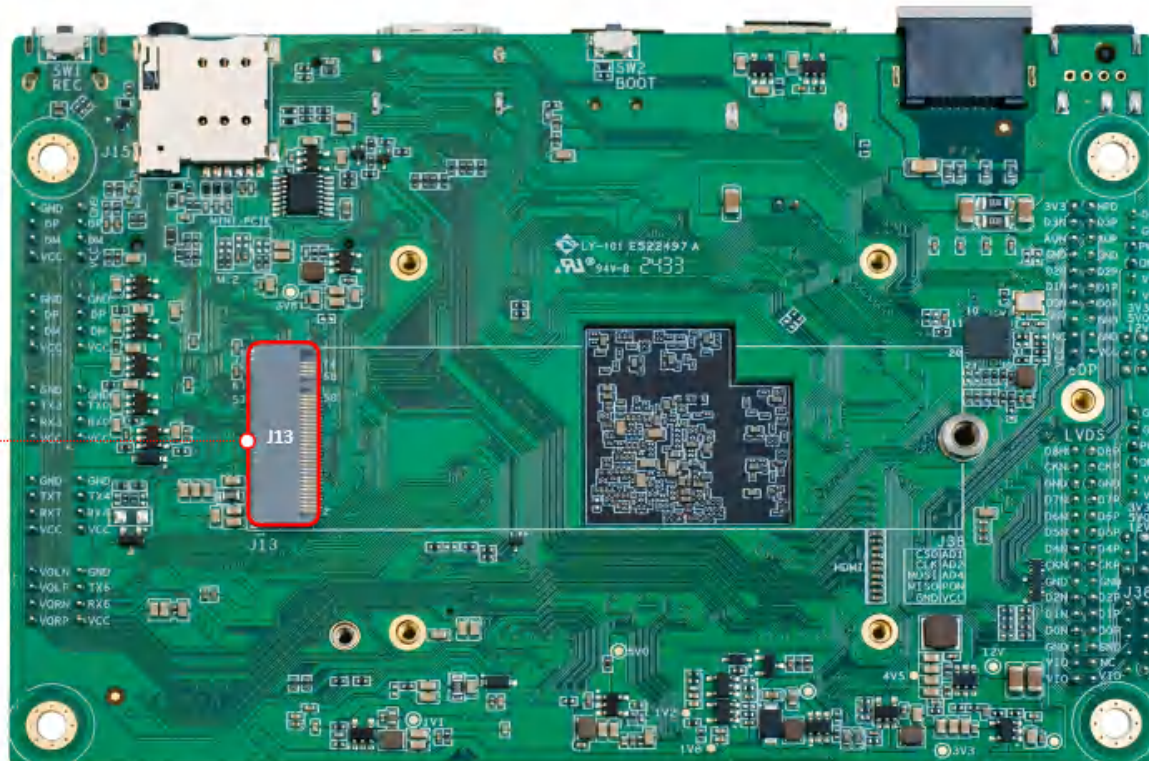
1  industio@ido:~$ sudo fdisk -l
2  Disk /dev/ram0: 8 MiB, 8388608 bytes, 16384 sectors
3  Units: sectors of 1 * 512 = 512 bytes
4  Sector size (logical/physical): 512 bytes / 4096 bytes
5  I/O size (minimum/optimal): 4096 bytes / 4096 bytes
6  .....
7
8  设备          启动  起点      末尾      扇区  大小  Id  类型
9  /dev/mmcblk1p1      16 60579839 60579824 28.9G  7  HPFS/NTFS/exFAT
10
11 industio@ido:~$ df -h
12 文件系统      大小  已用  可用  已用% 挂载点
13 /dev/root      56G  8.1G  48G   15% /
14 tmpfs          3.9G   0  3.9G   0% /dev/shm
15 tmpfs          1.6G  4.5M  1.6G   1% /run
16 tmpfs          5.0M  4.0K  5.0M   1% /run/lock
17 tmpfs          3.9G  8.0K  3.9G   1% /tmp
18 /dev/mmcblk0p7  2.0G  292K  2.0G   1% /userdata
19 /dev/mmcblk0p6 124M   12M  109M  10% /oem
20 tmpfs          792M   88K  792M   1% /run/user/1000
21 /dev/mmcblk1p1  29G   64M   29G   1% /media/industio/DECABB50CABB23A5

```

3.1.13. M.2接口

主板上使用标准M.2-M-key M.2接口连接座J13，如下图所示：

J13
M.2 M-Key



支持PCIe2.1通信，适用2280尺寸NVME固态硬盘。

```

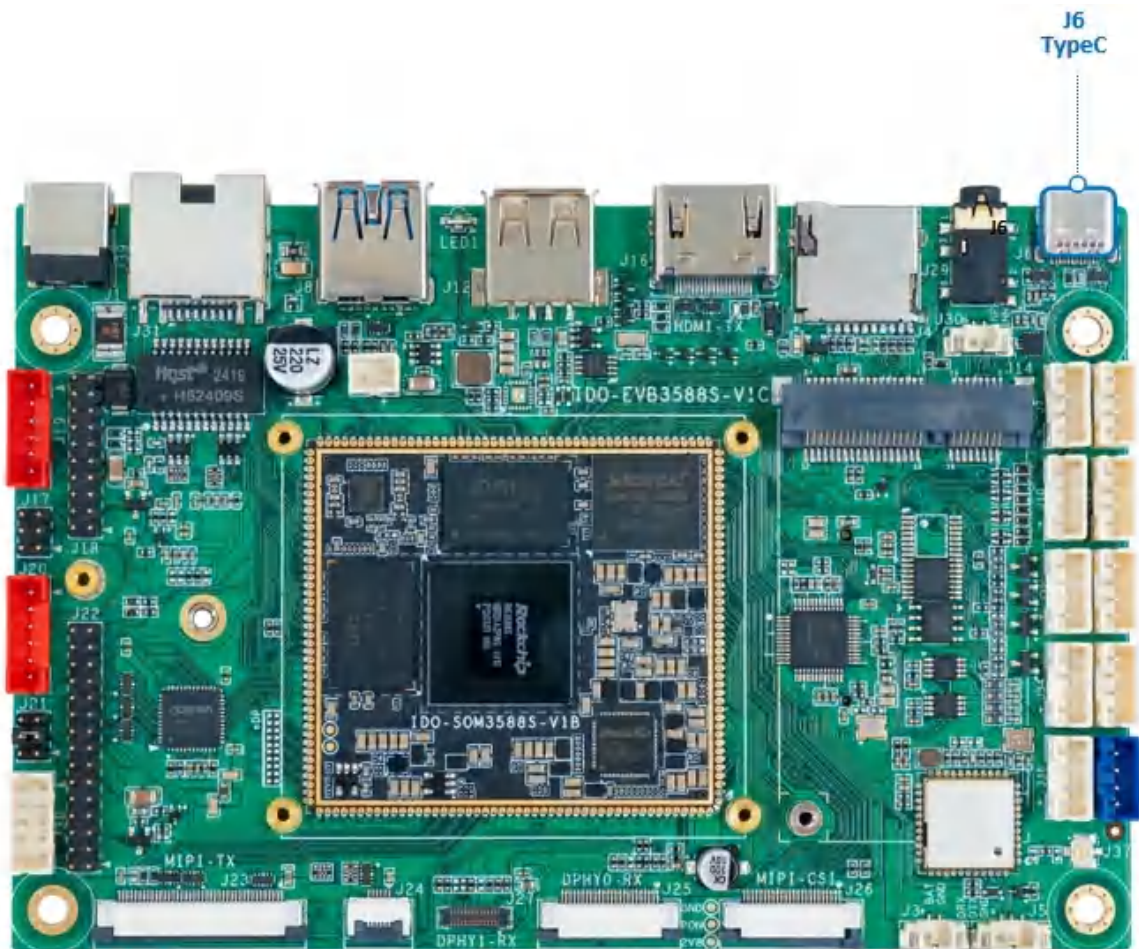
1  industio@ido:~$ sudo fdisk -l
2  Disk /dev/ram0: 8 MiB, 8388608 bytes, 16384 sectors
3  Units: sectors of 1 * 512 = 512 bytes
4  Sector size (logical/physical): 512 bytes / 4096 bytes
5  I/O size (minimum/optimal): 4096 bytes / 4096 bytes
6
7  Disk /dev/ram1: 8 MiB, 8388608 bytes, 16384 sectors
8  Units: sectors of 1 * 512 = 512 bytes
9  Sector size (logical/physical): 512 bytes / 4096 bytes
10 I/O size (minimum/optimal): 4096 bytes / 4096 bytes
11 ...
12 Disk /dev/nvme0n1: 476.94 GiB, 512110190592 bytes, 1000215216 sectors
13 Disk model: XPG GAMMIX S11L
14 Units: sectors of 1 * 512 = 512 bytes
15 Sector size (logical/physical): 512 bytes / 512 bytes
16 I/O size (minimum/optimal): 512 bytes / 512 bytes
17 Disklabel type: dos
18 Disk identifier: 0x00000000
19
20 Device          Boot Start          End      Sectors   Size Id Type
21 /dev/nvme0n1p1      2048 1000215182 1000213135 476.9G  c W95 FAT32 (LBA)
22
23 industio@ido:~$ df -h
24 文件系统      大小  已用  可用  已用% 挂载点
25 /dev/root      56G   8.1G   48G   15% /
26 tmpfs          3.9G     0   3.9G    0% /dev/shm
27 tmpfs          1.6G   4.5M   1.6G    1% /run
28 tmpfs          5.0M   4.0K   5.0M    1% /run/lock
29 tmpfs          3.9G   8.0K   3.9G    1% /tmp
30 /dev/mmcblk0p6  124M   12M   109M   10% /oem
31 /dev/mmcblk0p7   2.0G  292K   2.0G    1% /userdata
32 /dev/nvme0n1p1 233G   14G  219G    6% /media/industio/m2
33 tmpfs          792M   68K   792M    1% /run/user/1000

```

3.1.14. LCD显示

3.1.14.1. DP

DP接口 (TYPE-C) J6, 如下图所示:



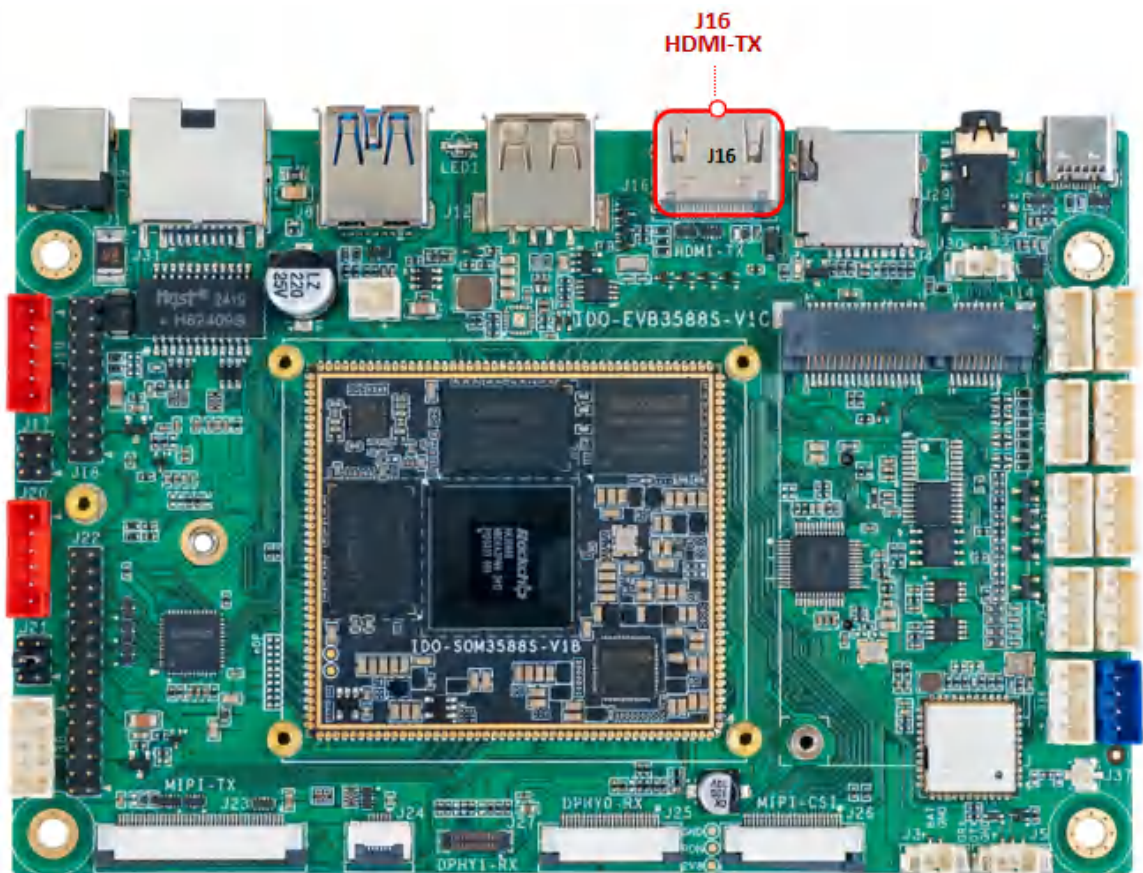
可以使用 USB-C 转 HDMI 高清线连接 HDMI 显示器输出画面，如下图所示：



- 最高支持8K@30fps输出

3.1.14.2. HDMI-TX

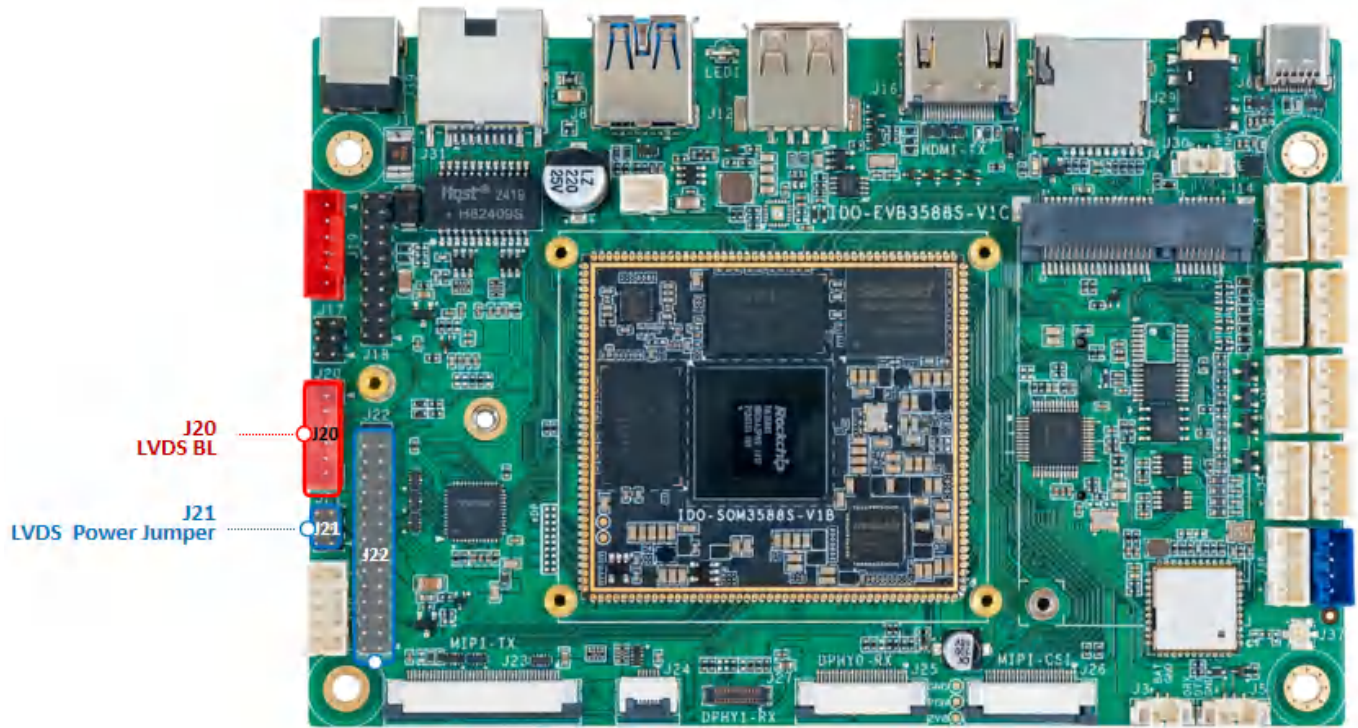
主板标准HDMI-A接口，如下图所示：



- 最高支持 HDMI2.1 8K@60fps 输出

3.1.14.3. Dual LVDS屏

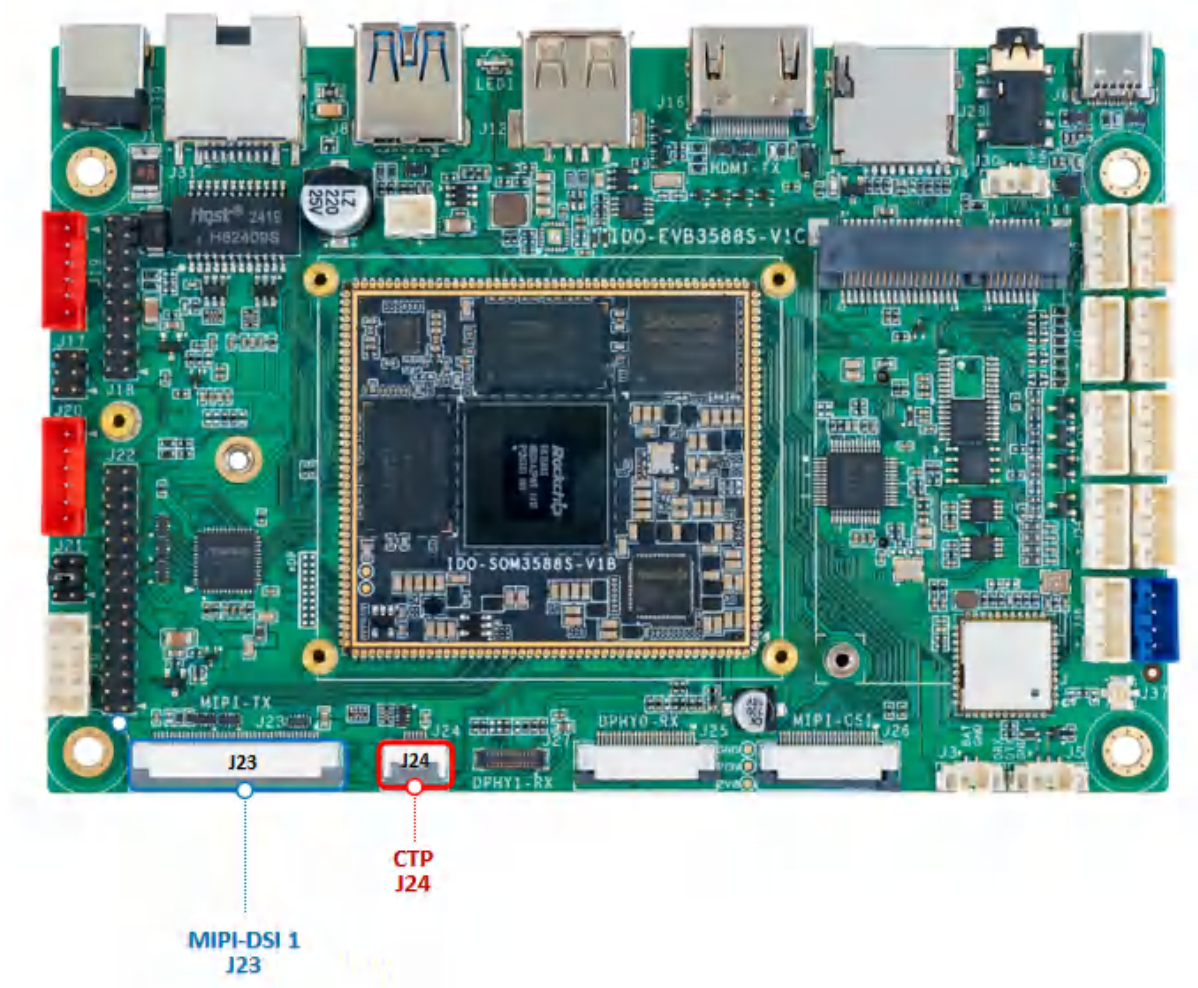
Dual LVDS接口J22，背光接 J20，屏幕供电电压选择接口J21，如下图所示：



J21屏幕供电电压选择接口，接屏前，需要根据具体的屏幕规格书来确认供电跳线帽接到3.3V、5V或12V，默认3.3V。

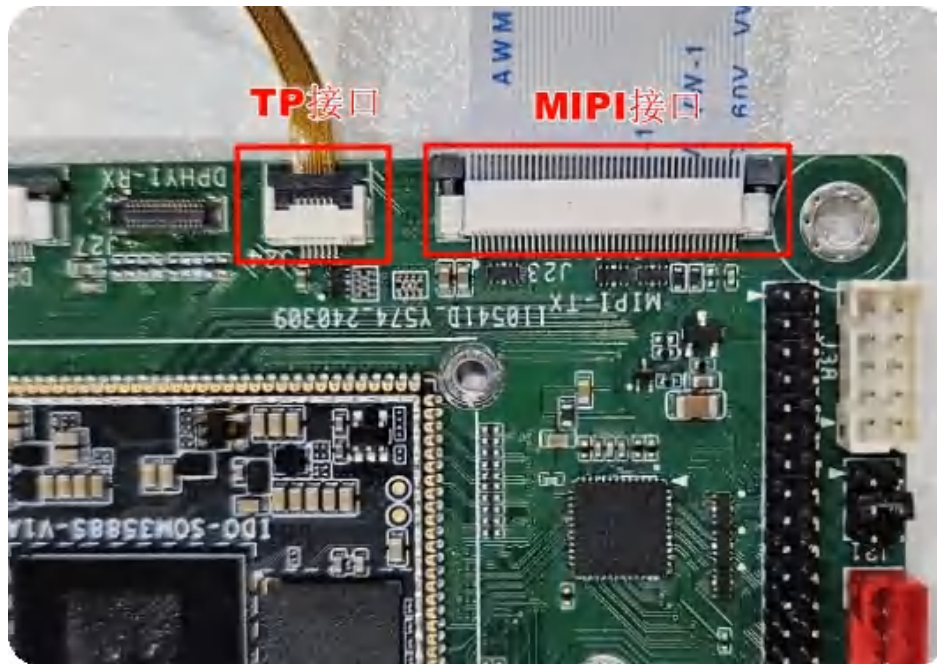
3.1.14.4. MIPI

MIPI接口J23，如下图所示：



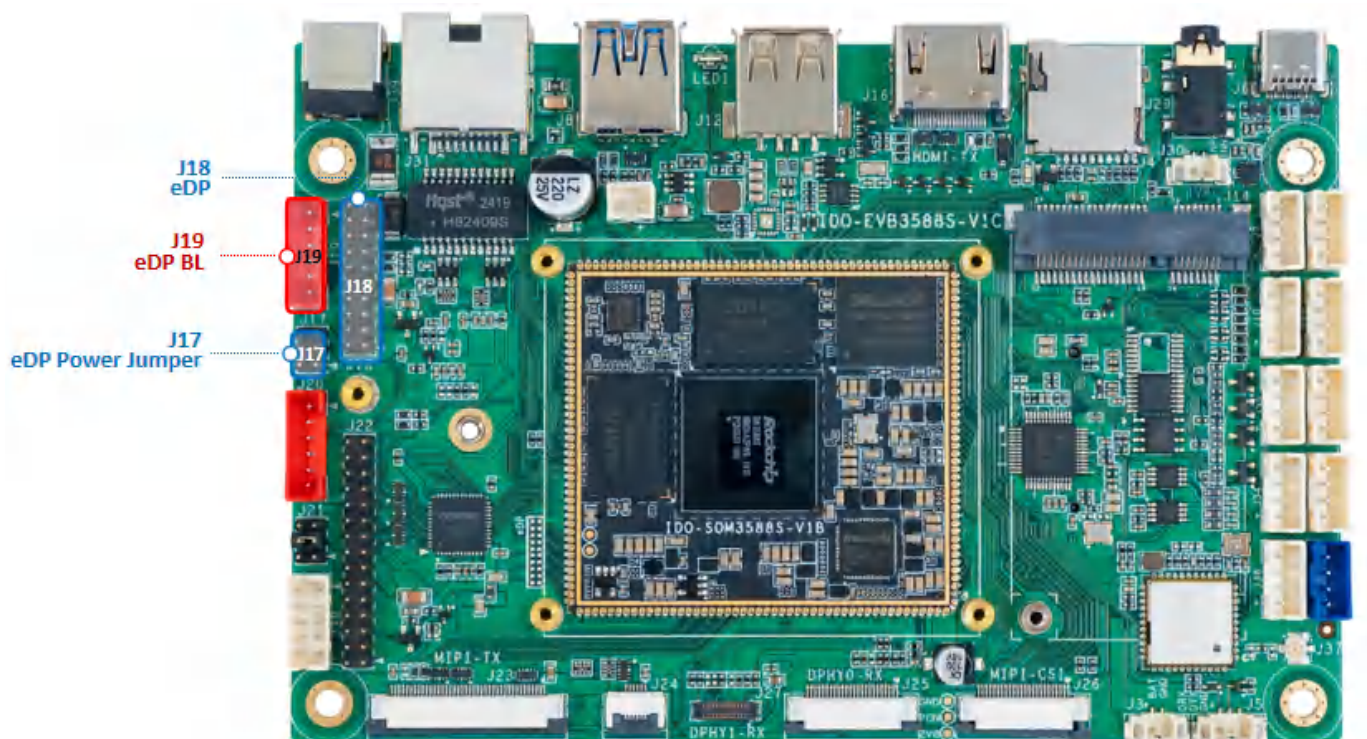
- MIPI供电为3.3V
- MIPI屏LCD排线接线：
- 触摸 TP 接口，下接
- 主板 LCD 排线接线（上接），40Pin FPC 屏座子

接法如下：



3.1.14.5. EDP

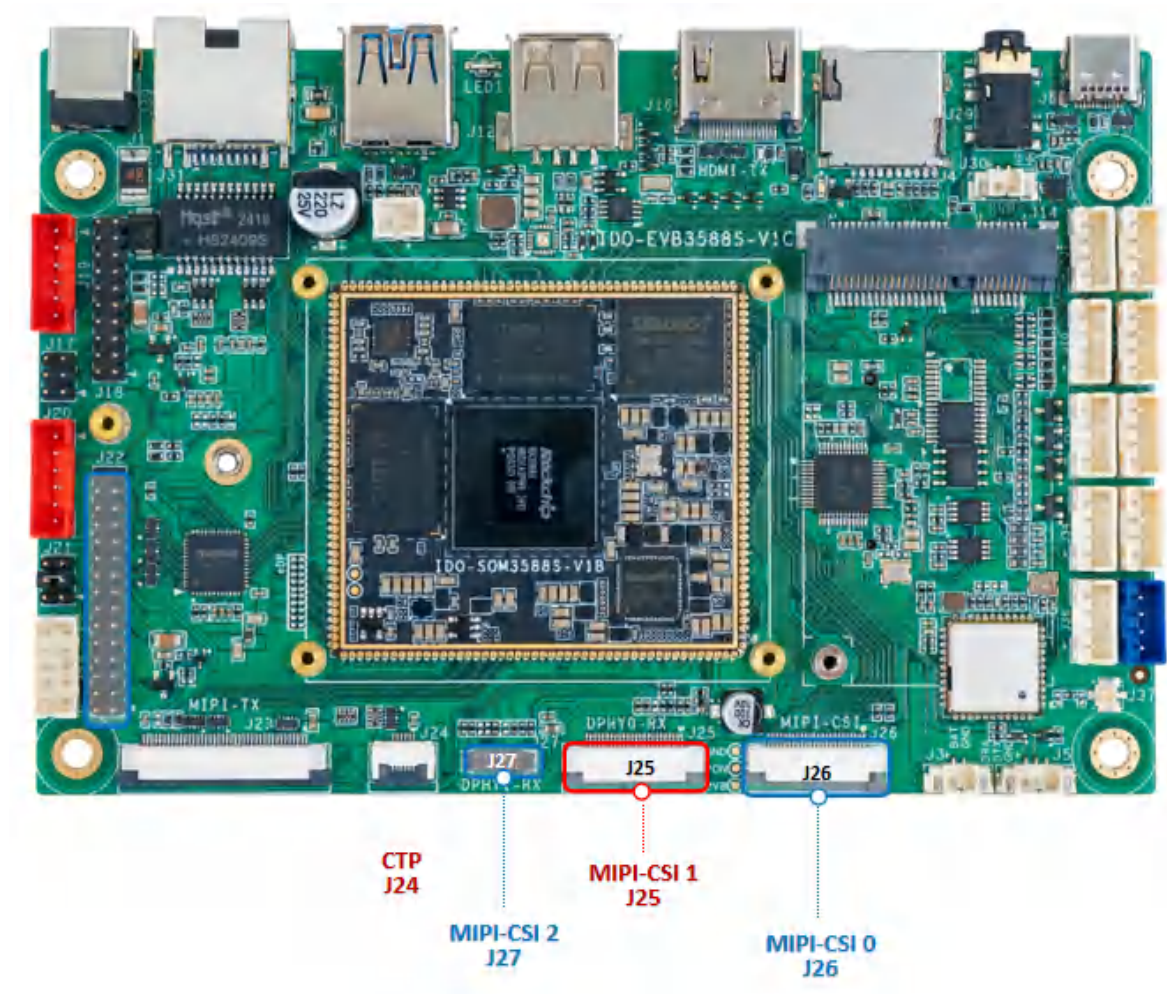
eDP接口，如下图所示：



注意：eDP与HDMI互斥，默认配置为 HDMI 功能，使用 eDP 需要更改硬件和软件

3.1.15. MIPI Camera

MIPI Camer接口J25、J26和J27，如下图所示：



MIPI摄像头接线，如下图所示：



3.1.15.1. MIPI-CSI1

3.1.15.1.1. 预览

▼ Plain Text |

```
1  industio@ido:~$ gst-launch-1.0 v4l2src device=/dev/video33 ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! videoconvert ! autovideodeosink sync=false
```

3.1.15.1.2. 抓图

```
1  industio@ido:~$ v4l2-ctl --verbose -d /dev/video33 --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
```

3.1.15.2. MIPI-CSI0

3.1.15.2.1. 预览

```
1  industio@ido:~$ gst-launch-1.0 v4l2src device=/dev/video42 ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! videoconvert ! autovideosink sync=false
```

3.1.15.2.2. 抓图

```
1  industio@ido:~$ v4l2-ctl --verbose -d /dev/video42 --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
```

3.1.15.3. MIPI-CSI2

3.1.15.3.1. 预览

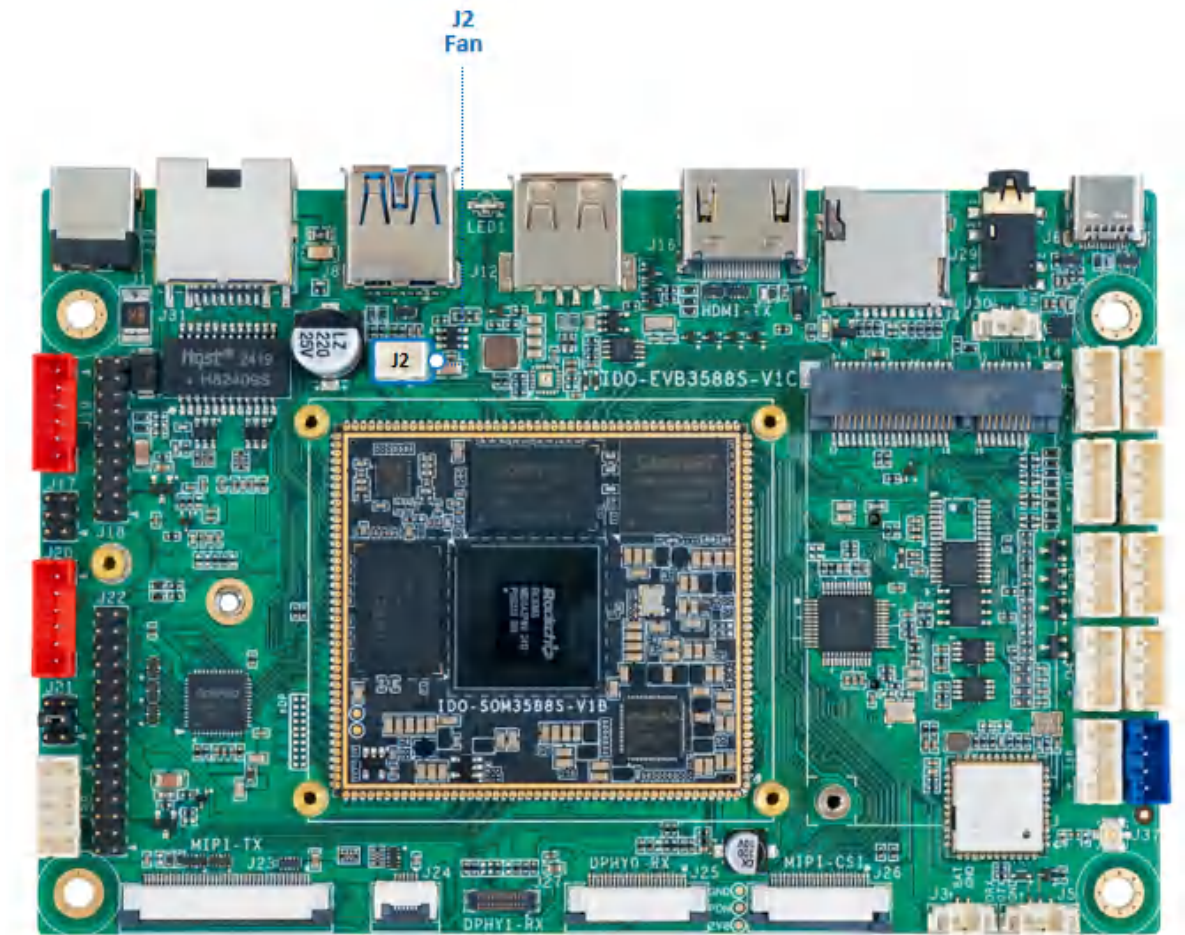
```
1  industio@ido:~$ gst-launch-1.0 v4l2src device=/dev/video51 ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! videoconvert ! autovideosink sync=false
```

3.1.15.3.2. 抓图

```
1 industio@ido:~$ v4l2-ctl --verbose -d /dev/video51 --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
```

3.1.16. FAN 风扇

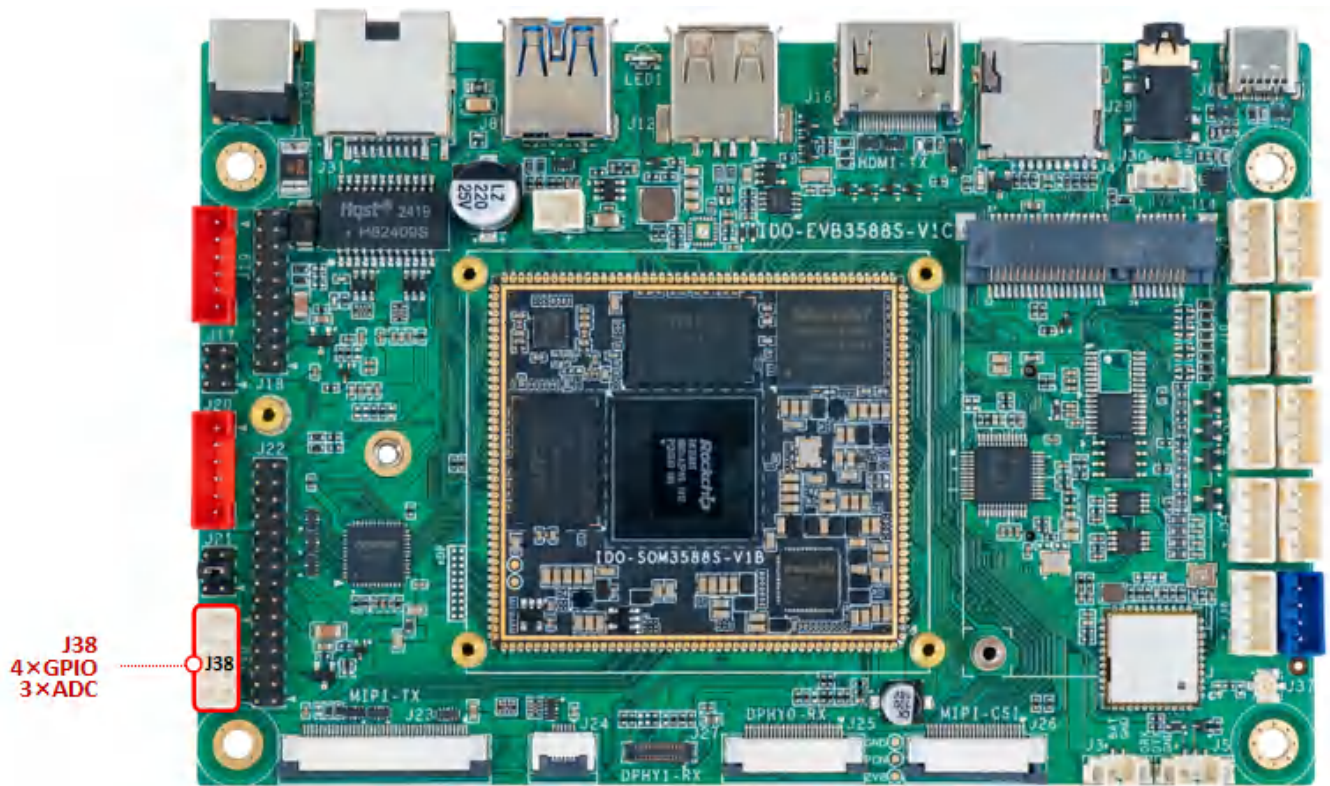
PH2 5V FAN风扇接口J2，如下图所示：



- 在CPU高于 55℃ 风扇自动开启，低于 50℃ 风扇自动关闭

3.1.17. GPIO

GPIO接口，如下图所示：



序号	管脚标号	GPIO 标号	GPIO 序号
1	CS0	GPIO1_A3	35
3	CLK	GPIO1_A2	34
5	MOSI	GPIO1_A1 (被占用，默认为 RS485)	33
7	MISO	GPIO1_A0 (被占用，默认为 RS485)	32
9	GND	地	/
2	AD1	recovery 键	/
4	AD2	ADC2	/
6	AD4	ADC4	/
8	PON	Power 键	/
10	VCC	3.3V 供电	/

```
1  输出:
2  # gpio1_A3 拉高
3  gpioset 1 3=1
4
5  # gpio1_A3 拉低
6  gpioset 1 3=0
7
8  # gpio2_A2 拉高
9  gpioset 1 2=1
10
11 # gpio2_A2 拉低
12 gpioset 1 2=0
13
14 输入:
15 # gpio2_A3 输入获取电平状态
16 gpioget 1 3
17
18 # gpio2_A2 输入获取电平状态
19 gpioget 1 2
```

3.2. GPU

3.2.1. glmark2

```

1  industio@ido:~$ glmark2
2  =====
3  glmark2 2021.02
4  =====
5  OpenGL Information
6  GL_VENDOR:      Mesa
7  GL_RENDERER:    Mali-G610 (Panfrost)
8  GL_VERSION:     3.1 Mesa 25.0.7 (git-de240b8d61)
9  =====
10 [build] use-vbo=false: FPS: 1214 FrameTime: 0.824 ms
11 [build] use-vbo=true: FPS: 1214 FrameTime: 0.824 ms
12 [texture] texture-filter=nearest: FPS: 1393 FrameTime: 0.718 ms
13 [texture] texture-filter=linear: FPS: 1208 FrameTime: 0.828 ms
14 [texture] texture-filter=mipmap: FPS: 1209 FrameTime: 0.827 ms
15 [shading] shading=gouraud: FPS: 1097 FrameTime: 0.912 ms
16 [shading] shading=blinn-phong-inf: FPS: 1234 FrameTime: 0.810 ms
17 [shading] shading=phong: FPS: 1046 FrameTime: 0.956 ms
18 [shading] shading=cel: FPS: 1188 FrameTime: 0.842 ms
19 [bump] bump-render=high-poly: FPS: 896 FrameTime: 1.116 ms
20 [bump] bump-render=normals: FPS: 1486 FrameTime: 0.673 ms
21 [bump] bump-render=height: FPS: 1353 FrameTime: 0.739 ms
22 [effect2d] kernel=0,1,0;1,-4,1;0,1,0;: FPS: 996 FrameTime: 1.004 ms
23 [effect2d] kernel=1,1,1,1,1;1,1,1,1,1;1,1,1,1,1;: FPS: 659 FrameTime: 1.51
    7 ms
24 [pulsar] light=false:quads=5:texture=false: FPS: 1224 FrameTime: 0.817 ms
25 [desktop] blur-radius=5:effect=blur:passes=1:separable=true:windows=4: FPS
    : 330 FrameTime: 3.030 ms
26 [desktop] effect=shadow:windows=4: FPS: 1071 FrameTime: 0.934 ms
27 [buffer] columns=200:interleave=false:update-dispersion=0.9:update-fractio
    n=0.5:update-method=map: FPS: 452 FrameTime: 2.212 ms
28 [buffer] columns=200:interleave=false:update-dispersion=0.9:update-fractio
    n=0.5:update-method=subdata: FPS: 474 FrameTime: 2.110 ms
29 [buffer] columns=200:interleave=true:update-dispersion=0.9:update-fraction
    =0.5:update-method=map: FPS: 624 FrameTime: 1.603 ms
30 [ideas] speed=duration: FPS: 1058 FrameTime: 0.945 ms
31 [jellyfish] <default>: FPS: 1088 FrameTime: 0.919 ms
32 [terrain] <default>: FPS: 100 FrameTime: 10.000 ms
33 [shadow] <default>: FPS: 925 FrameTime: 1.081 ms
34 [refract] <default>: FPS: 304 FrameTime: 3.289 ms
35 [conditionals] fragment-steps=0:vertex-steps=0: FPS: 1350 FrameTime: 0.741
    ms
36 [conditionals] fragment-steps=5:vertex-steps=0: FPS: 1250 FrameTime: 0.800
    ms
37 [conditionals] fragment-steps=0:vertex-steps=5: FPS: 1346 FrameTime: 0.743
    ms

```

```

38 [function] fragment-complexity=low:fragment-steps=5: FPS: 1173 FrameTime:
39 0.853 ms
40 [function] fragment-complexity=medium:fragment-steps=5: FPS: 1301 FrameTim
41 e: 0.769 ms
42 [loop] fragment-loop=false:fragment-steps=5:vertex-steps=5: FPS: 1158 Fram
43 eTime: 0.864 ms
44 [loop] fragment-steps=5:fragment-uniform=false:vertex-steps=5: FPS: 1340 F
45 rameTime: 0.746 ms
[loop] fragment-steps=5:fragment-uniform=true:vertex-steps=5: FPS: 1299 Fr
ameTime: 0.770 ms
=====
                                glmark2 Score: 1032
=====

```

3.2.2. glmark2-es2

```

1  industio@ido:~$ glmark2-es2
2  =====
3  glmark2 2021.02
4  =====
5  OpenGL Information
6  GL_VENDOR:      Mesa
7  GL_RENDERER:    Mali-G610 (Panfrost)
8  GL_VERSION:     OpenGL ES 3.1 Mesa 25.0.7 (git-de240b8d61)
9  =====
10 ▾ [build] use-vbo=false: FPS: 1152 FrameTime: 0.868 ms
11 ▾ [build] use-vbo=true: FPS: 1241 FrameTime: 0.806 ms
12 ▾ [texture] texture-filter=nearest: FPS: 1481 FrameTime: 0.675 ms
13 ▾ [texture] texture-filter=linear: FPS: 1500 FrameTime: 0.667 ms
14 ▾ [texture] texture-filter=mipmap: FPS: 1506 FrameTime: 0.664 ms
15 ▾ [shading] shading=gouraud: FPS: 1127 FrameTime: 0.887 ms
16 ▾ [shading] shading=blinn-phong-inf: FPS: 1203 FrameTime: 0.831 ms
17 ▾ [shading] shading=phong: FPS: 1249 FrameTime: 0.801 ms
18 ▾ [shading] shading=cel: FPS: 1110 FrameTime: 0.901 ms
19 ▾ [bump] bump-render=high-poly: FPS: 951 FrameTime: 1.052 ms
20 ▾ [bump] bump-render=normals: FPS: 1569 FrameTime: 0.637 ms
21 ▾ [bump] bump-render=height: FPS: 1581 FrameTime: 0.633 ms
22 ▾ [effect2d] kernel=0,1,0;1,-4,1;0,1,0;: FPS: 1284 FrameTime: 0.779 ms
23 ▾ [effect2d] kernel=1,1,1,1,1;1,1,1,1,1;1,1,1,1,1;: FPS: 980 FrameTime: 1.02
    0 ms
24 ▾ [pulsar] light=false:quads=5:texture=false: FPS: 1487 FrameTime: 0.672 ms
25 ▾ [desktop] blur-radius=5:effect=blur:passes=1:separable=true:windows=4: FPS
    : 273 FrameTime: 3.663 ms
26 ▾ [desktop] effect=shadow:windows=4: FPS: 1091 FrameTime: 0.917 ms
27 ▾ [buffer] columns=200:interleave=false:update-dispersion=0.9:update-fractio
    n=0.5:update-method=map: FPS: 527 FrameTime: 1.898 ms
28 ▾ [buffer] columns=200:interleave=false:update-dispersion=0.9:update-fractio
    n=0.5:update-method=subdata: FPS: 536 FrameTime: 1.866 ms
29 ▾ [buffer] columns=200:interleave=true:update-dispersion=0.9:update-fraction
    =0.5:update-method=map: FPS: 592 FrameTime: 1.689 ms
30 ▾ [ideas] speed=duration: FPS: 825 FrameTime: 1.212 ms
31 ▾ [jellyfish] <default>: FPS: 1107 FrameTime: 0.903 ms
32 ▾ [terrain] <default>: FPS: 88 FrameTime: 11.364 ms
33 ▾ [shadow] <default>: FPS: 869 FrameTime: 1.151 ms
34 ▾ [refract] <default>: FPS: 331 FrameTime: 3.021 ms
35 ▾ [conditionals] fragment-steps=0:vertex-steps=0: FPS: 1357 FrameTime: 0.737
    ms
36 ▾ [conditionals] fragment-steps=5:vertex-steps=0: FPS: 1382 FrameTime: 0.724
    ms
37 ▾ [conditionals] fragment-steps=0:vertex-steps=5: FPS: 1459 FrameTime: 0.685
    ms

```

```

38 [function] fragment-complexity=low:fragment-steps=5: FPS: 1448 FrameTime:
39 0.691 ms
40 [function] fragment-complexity=medium:fragment-steps=5: FPS: 1218 FrameTim
41 e: 0.821 ms
42 [loop] fragment-loop=false:fragment-steps=5:vertex-steps=5: FPS: 1362 Fram
43 eTime: 0.734 ms
44 [loop] fragment-steps=5:fragment-uniform=false:vertex-steps=5: FPS: 1450 F
45 rameTime: 0.690 ms
[loop] fragment-steps=5:fragment-uniform=true:vertex-steps=5: FPS: 1303 Fr
ameTime: 0.767 ms
=====
                                glmark2 Score: 1110
=====

```

3.3. NPU

将rknn-toolkit2-master.tar.gz压缩包拷贝板子上，使用yolov5进行rknn模型转换测试：

3.3.1. 编译rknn_yolov5_demo

```

1 industio@ido:~$ export GCC_COMPILER=/usr/bin/aarch64-linux-gnu
2 industio@ido:~$ tar -xvf rknn-toolkit2-master.tar.gz
3 industio@ido:~$ cd rknn-toolkit2-master/rknpu2/examples/rknn_yolov5_demo
4 industio@ido:~/rknn-toolkit2-master/rknpu2/examples/rknn_yolov5_demo$ chmod
a+x build-linux_RK3588.sh
5 industio@ido:~/rknn-toolkit2-master/rknpu2/examples/rknn_yolov5_demo$ ./bui
ld-linux.sh -t rk3588 -a aarch64 -b Release

```

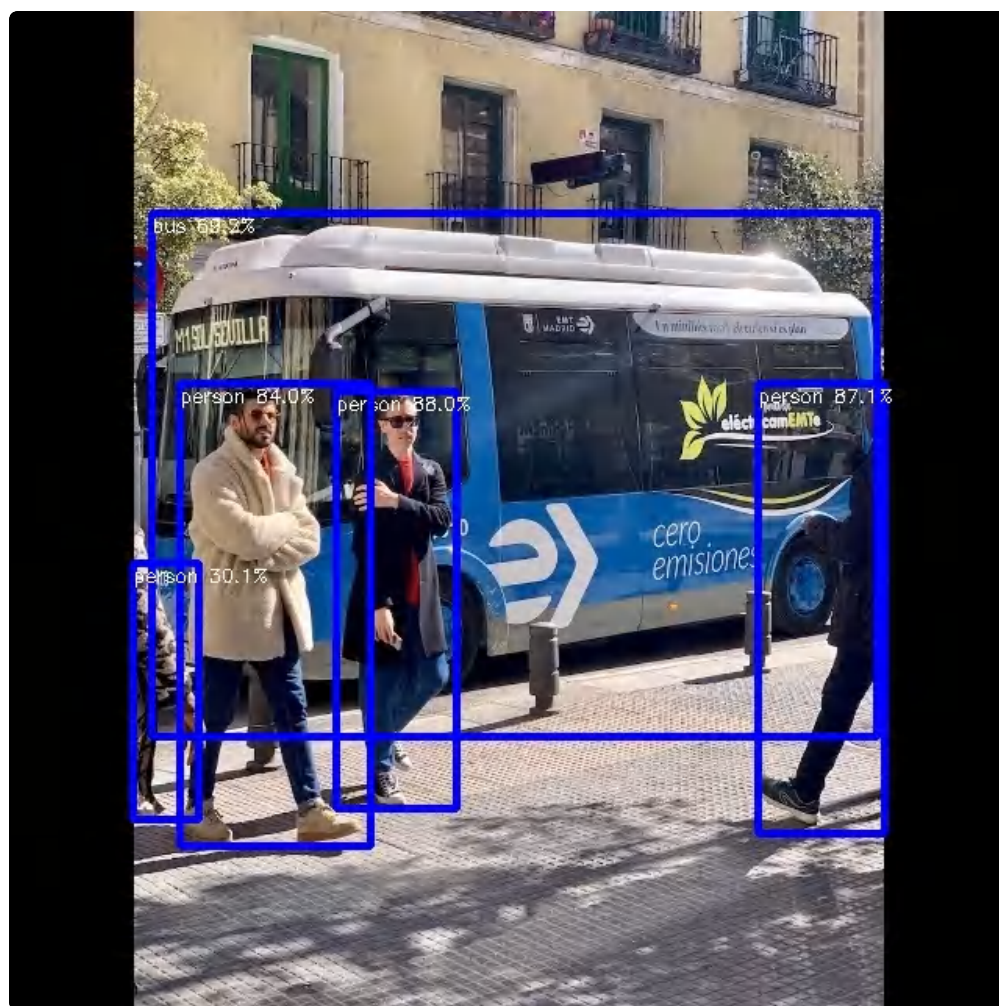
3.3.2. 测试

```

1 industio@ido:~/rknn-toolkit2-master/rknpu2/examples/rknn_yolov5_demo$ cd in
stall/rknn_yolov5_demo_Linux/
2 industio@ido:~/rknn-toolkit2-master/rknpu2/examples/rknn_yolov5_demo/instal
l/rknn_yolov5_demo_Linux$ ./rknn_yolov5_demo ./model/RK3588/yolov5s-640-64
0.rknn ./model/bus.jpg
3

```

执行完后会得到转换后的模型图片，如下所示：

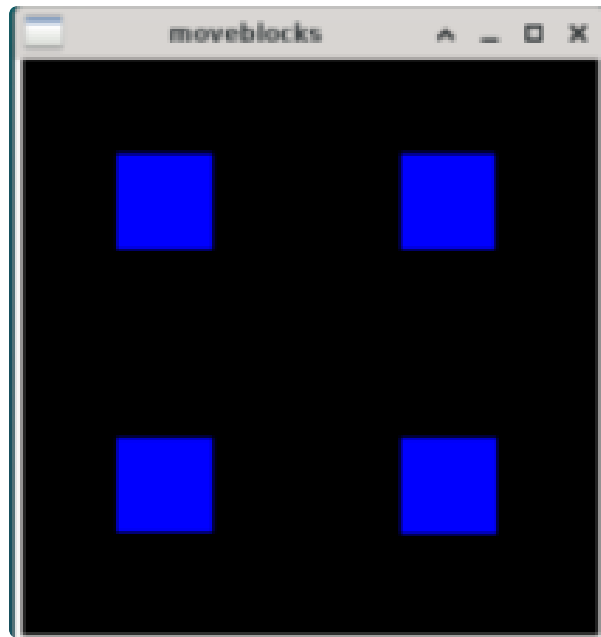


3.4. QT5

支持qt5.15，测试方法如下：

```
industio@ido:~$ chmod a+x ./moveblocks
industio@ido:~$ ./moveblocks
```

结果如下所示：

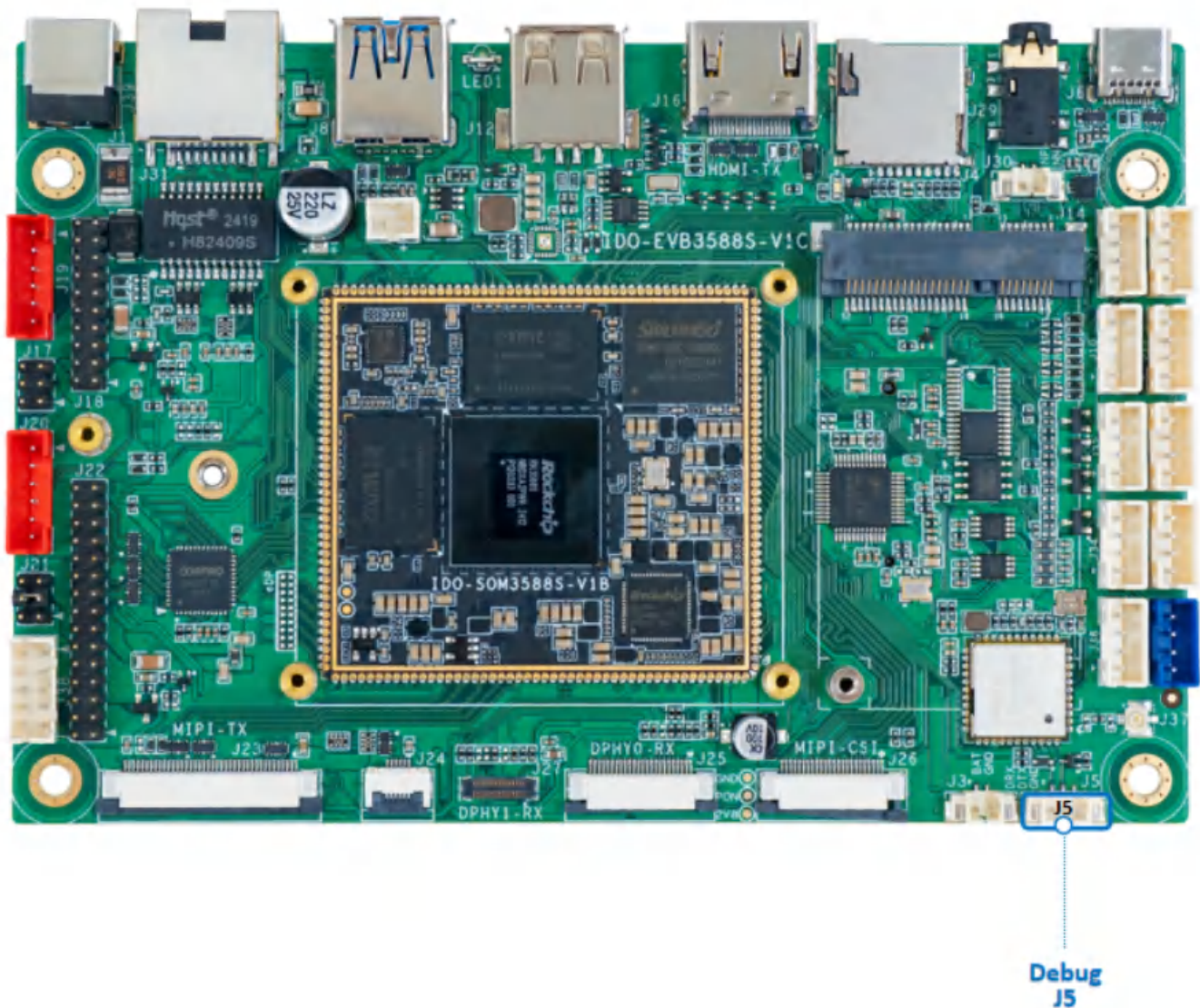


4. IDO-EVB3588S-V1C-Buildroot系统

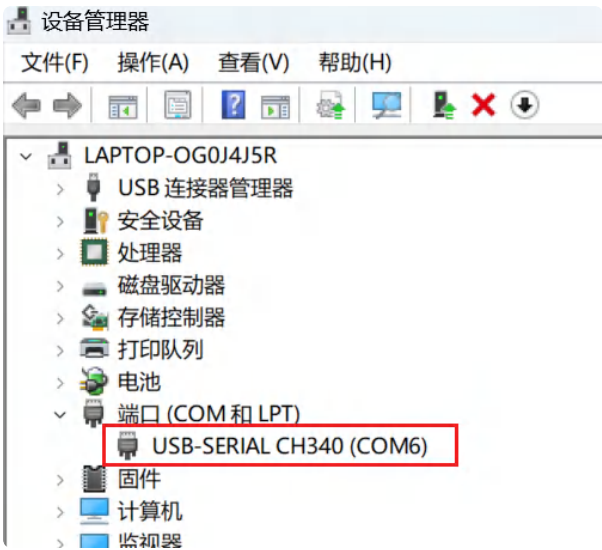
4.1. 功能及接口使用方法

4.1.1. DEBUG UART

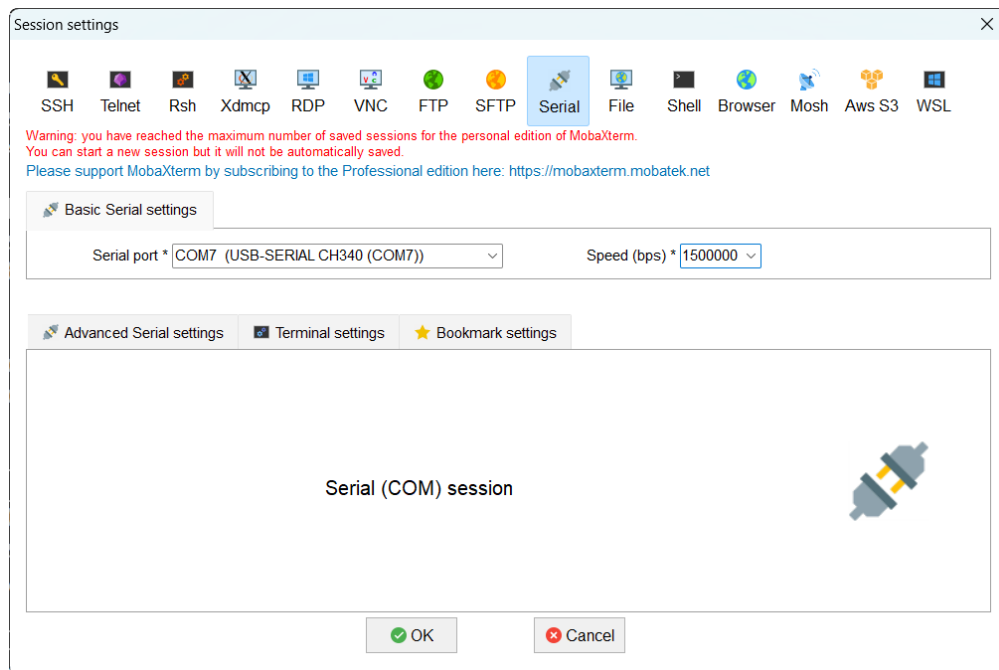
DEBUG UART位置J5，如下图所示：



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。



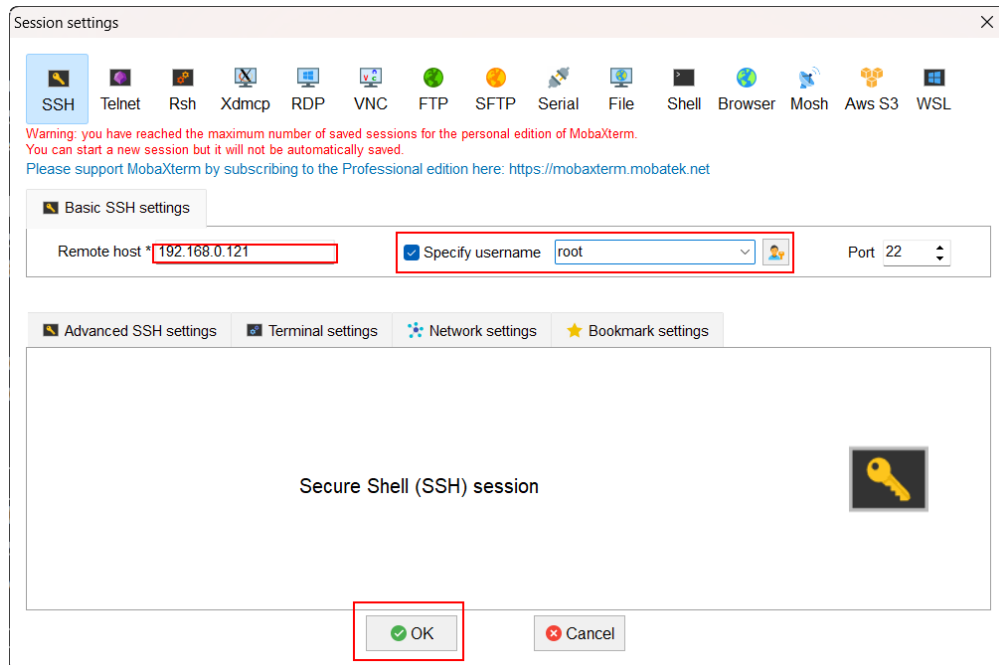
使用调试软件（MobaXterm、putty）等，以MobaXterm为例子，设置参数如下：



4.1.2. SSH登录

在知道IP的前提下，默认可以通过SSH登录进系统。

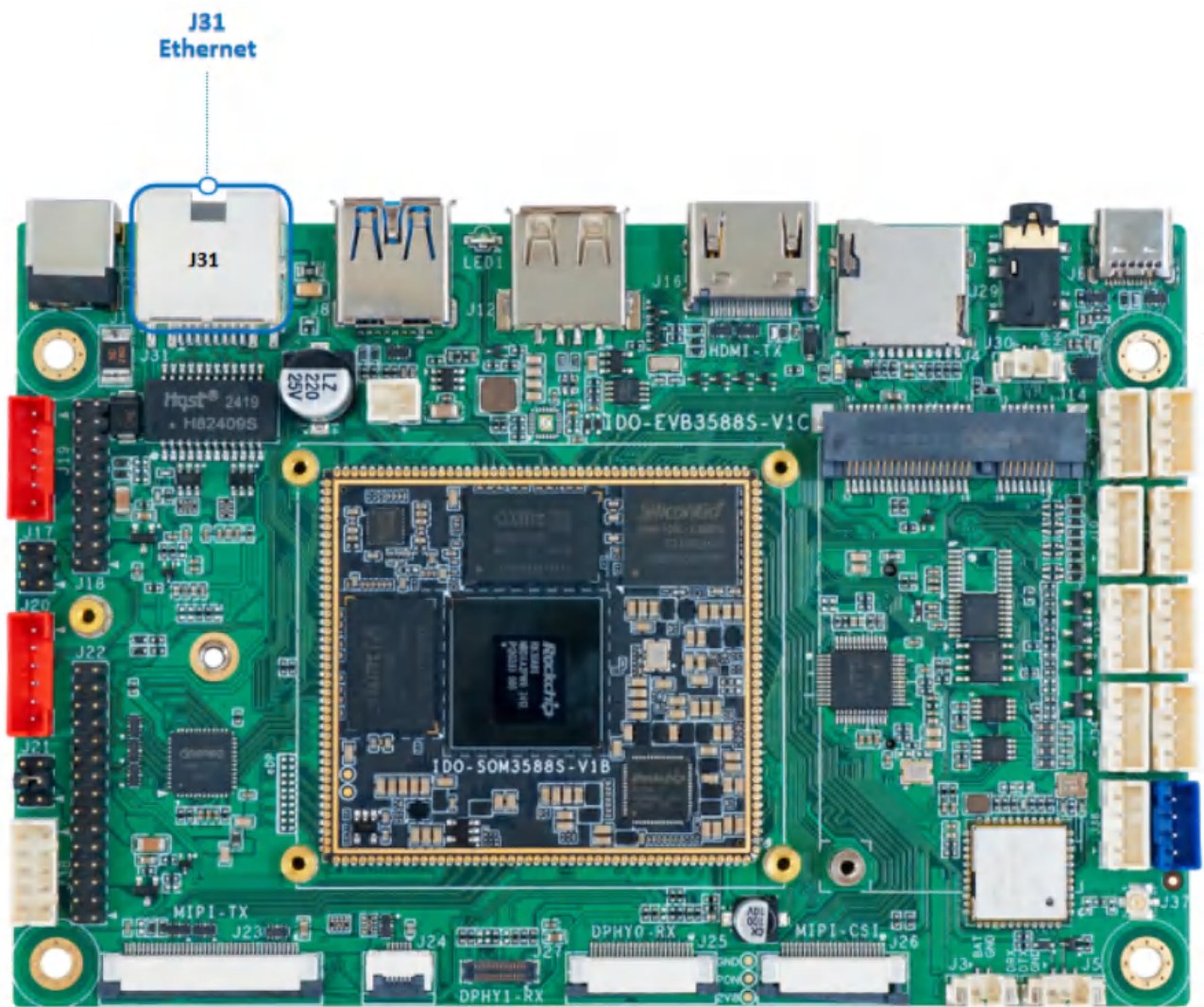
SSH登录账号密码：root/123456



4.1.3. 网络

4.1.3.1. 以太网

主板有一路千兆以太网接口位置J31，如下图所示：



设备节点分别为eth0，以太网接口默认支持DHCP，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。网络正常连接图标，如下图所示：

4.1.3.2. Ethernet动态IP设置

```

1 #eth0
2 root@rk3588-buildroot:/# ifconfig eth0
3 eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
4     inet 192.168.0.90 netmask 255.255.255.0 broadcast 192.168.0.255
5     inet6 fe80::1b83:a156:cb16:c6b0 prefixlen 64 scopeid 0x20<link>
6     ether fa:c6:ba:c7:0c:60 txqueuelen 1000 (Ethernet)
7     RX packets 141 bytes 15129 (14.7 KiB)
8     RX errors 0 dropped 4 overruns 0 frame 0
9     TX packets 89 bytes 9306 (9.0 KiB)
10    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
11    device interrupt 121
12

```

4.1.3.3. Ethernet静态IP设置

```

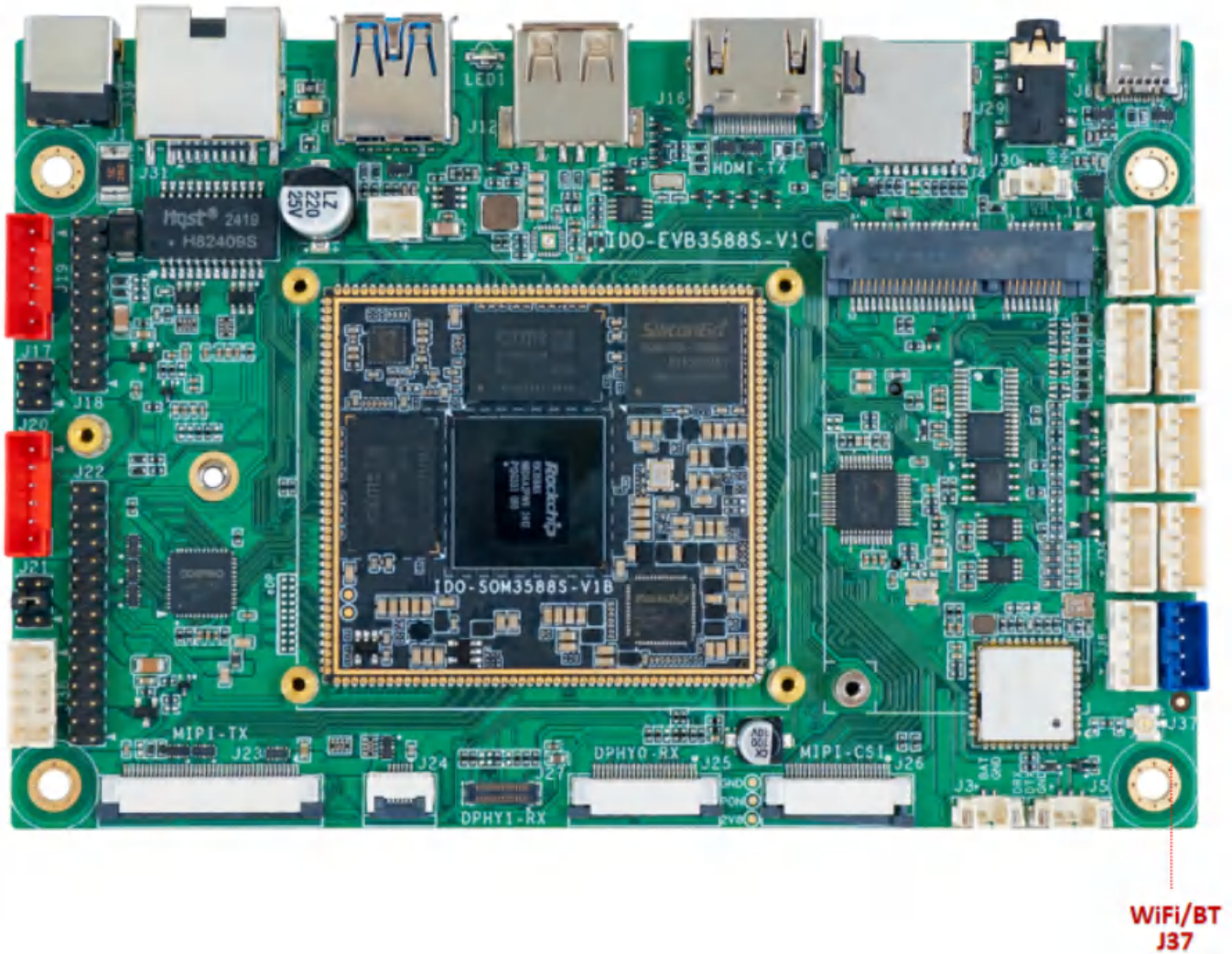
1 root@rk3588-buildroot:/# vim /etc/network/interfaces
2 # interface file auto-generated by buildroot
3
4 auto lo
5 iface lo inet loopback
6
7 auto eth0
8     iface eth0 inet static
9     address 192.168.1.123
10    netmask 255.255.255.0
11    gateway 192.168.1.1
12    nameserver 192.168.1.1
13
14 source-directory /etc/network/interfaces.d
15

```

设备断电重启，此静态IP设置仍然生效。

4.1.4. 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J37，如下图所示：



添加WiFi账号密码:

```
▼ Shell |
1 root@rk3588-buildroot:/# vim /etc/wpa_supplicant.conf
2 ctrl_interface=/var/run/wpa_supplicant
3 ap_scan=1
4
5 network={
6     ssid="SSID"
7     psk="PASSWORD"
8     key_mgmt=WPA-PSK
9 }
10
11
```

注意:

1. ssid: 账号

2. psk: 密码

扫描附近WiFi:

```
1 root@rk3588-buildroot:/# iw dev wlan0 scan | grep SSID
```

WiFi连接:

```
1 root@rk3588-buildroot:/# wpa_supplicant -Dnl80211 -c /etc/wpa_supplicant.conf -i wlan0 &
```

查看wlan0的IP地址, 确认连接成功:

```
1 root@rk3588-buildroot:/# ifconfig wlan0
2 wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
3     inet 192.168.0.118 netmask 255.255.255.0 broadcast 192.168.0.255
4     inet6 fe80::1036:80b4:5082:7440 prefixlen 64 scopeid 0x20<link>
5     ether c0:f5:35:12:ad:a2 txqueuelen 1000 (Ethernet)
6     RX packets 170 bytes 22149 (21.6 KiB)
7     RX errors 0 dropped 5 overruns 0 frame 0
8     TX packets 24 bytes 2683 (2.6 KiB)
9     TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
10
11 root@rk3588-buildroot:/# ping www.baidu.com -I wlan0
12 PING www.wshifen.com (103.235.47.188) from 192.168.0.118 wlan0: 56(84) bytes of data.
13 64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=1 ttl=45 time=274 ms
14 64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=2 ttl=45 time=299 ms
15 64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=3 ttl=45 time=320 ms
16 64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=4 ttl=45 time=241 ms
```

4.1.5. Bluetooth

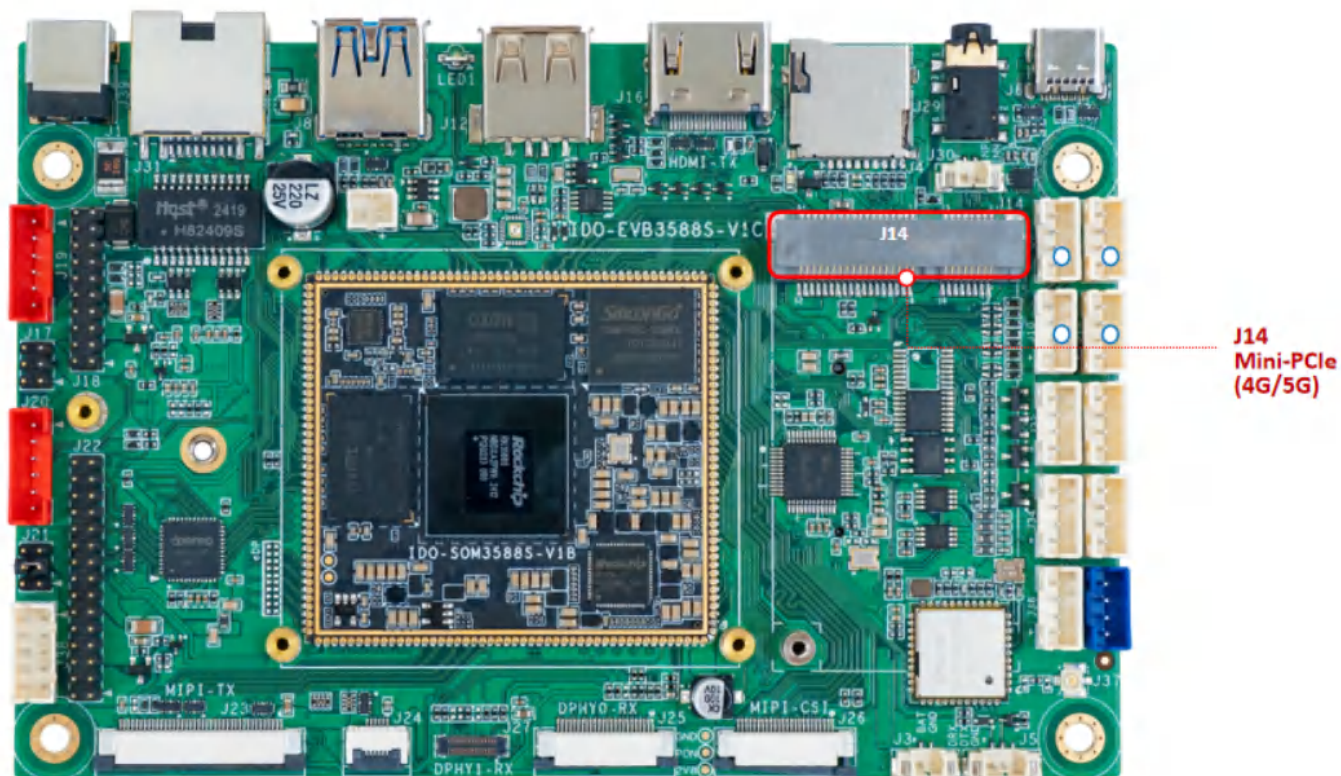
4.1.5.1. 连接方法

▼ Shell

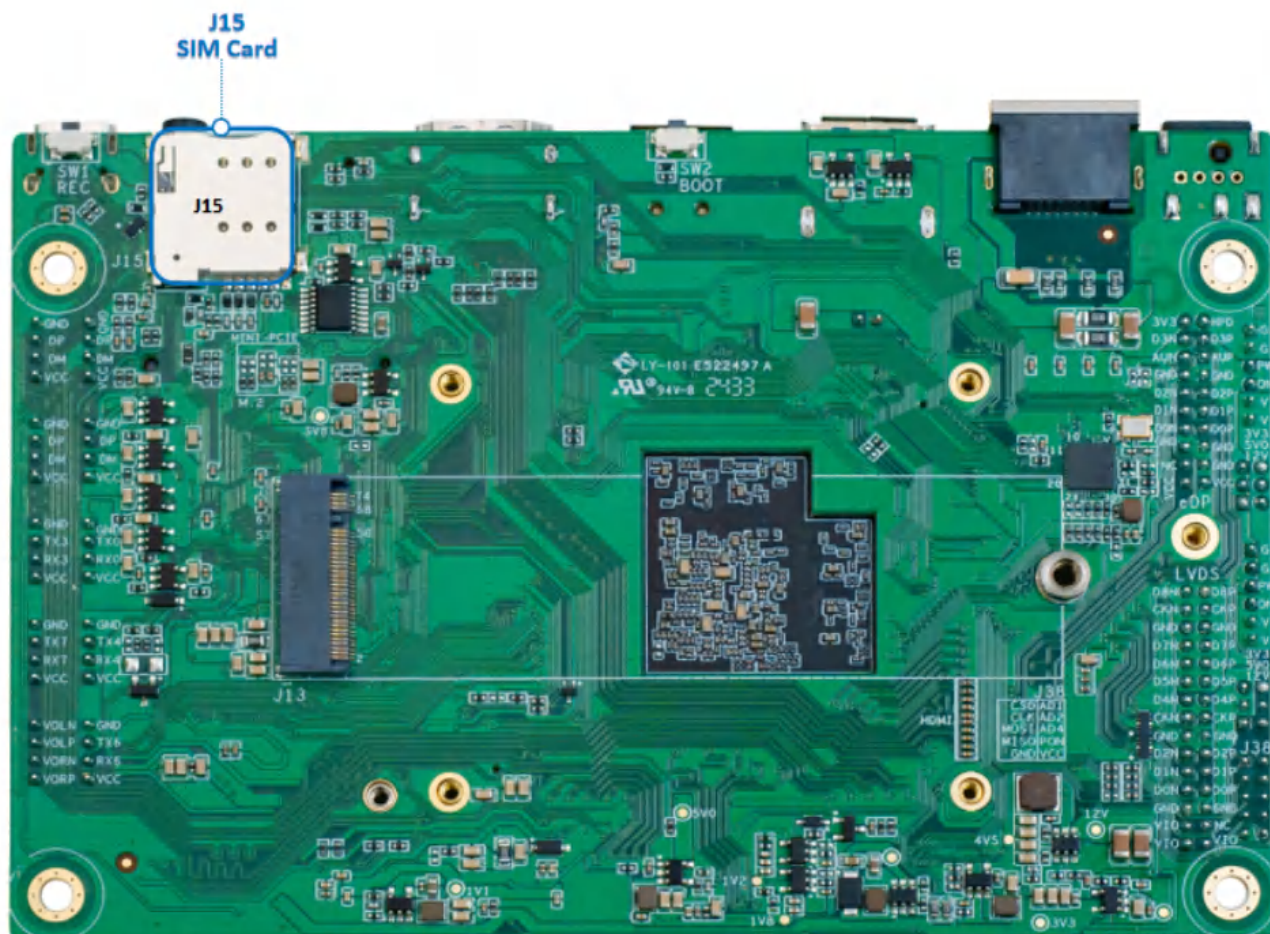
```
1 root@rk3588-buildroot:/# bluetoothctl
2 Agent registered
3 ▼ [CHG] Controller 9C:30:90:07:21:F2 Pairable: yes
4 ▼ [bluetooth]#
5
6 #打开蓝牙
7 ▼ [bluetooth]# power on
8 #扫描蓝牙
9 ▼ [bluetooth]# scan on
10 #查看蓝牙设备
11 ▼ [bluetooth]# devices
12 #信任蓝牙设备
13 ▼ [bluetooth]# trust 98:EE:94:82:D5:71
14 #蓝牙配对
15 ▼ [bluetooth]# pair 98:EE:94:82:D5:71
16 #连接蓝牙
17 ▼ [bluetooth]# connect 98:EE:94:82:D5:71
```

4.1.6. 4G/5G

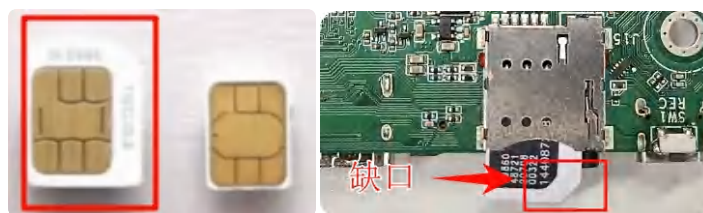
主板内置Mini PCIe 座子扩展 4G/5G模块，4G通信模块适配移远EC20、EC25等通用模组。5G通信模块适配移远RG200U-CN。模块安装位J14，如下图所示：



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J15位于主板背面，如下图所示：



使用Micro尺寸SIM卡，如下图所示：



4G使用 `ifconfig wwan0` 命令可查看到相关网络信息：

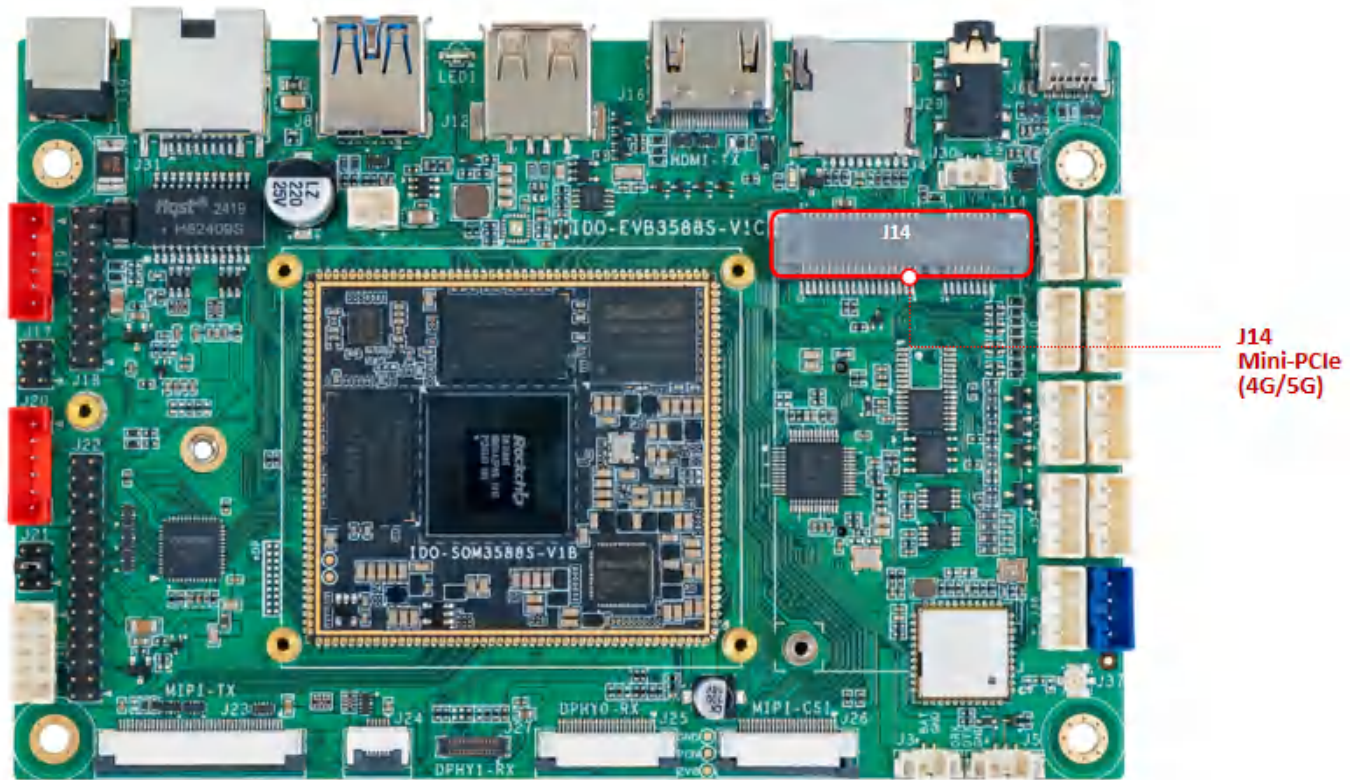
```
1 #拨号
2 root@rk3588-buildroot:/# quectel-CM &
3
4 #查看网络
5 root@rk3588-buildroot:/# ifconfig wwan0
6 wwan0: flags=193<UP,RUNNING,NOARP> mtu 1500
7     inet 10.101.61.51 netmask 255.255.255.248
8     inet6 fe80::fc:f6ff:fe8d:bab6 prefixlen 64 scopeid 0x20<link>
9     ether 02:fc:f6:8d:ba:b6 txqueuelen 1000 (Ethernet)
10    RX packets 42 bytes 7013 (6.8 KiB)
11    RX errors 0 dropped 0 overruns 0 frame 0
12    TX packets 57 bytes 4608 (4.5 KiB)
13    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
14
15 #测试通信
16 root@rk3588-buildroot:/# ping www.baidu.com -I wwan0
```

5G使用 `ifconfig usb0` 命令可查看到相关网络信息:

```
1 #拨号
2 root@rk3588-buildroot:/# quectel-CM &
3
4 #查看网络
5 root@rk3588-buildroot:/# ifconfig usb0
6 usb0: flags=193<UP,RUNNING,NOARP> mtu 1500
7     inet 10.101.61.51 netmask 255.255.255.248
8     inet6 fe80::fc:f6ff:fe8d:bab6 prefixlen 64 scopeid 0x20<link>
9     ether 02:fc:f6:8d:ba:b6 txqueuelen 1000 (Ethernet)
10    RX packets 42 bytes 7013 (6.8 KiB)
11    RX errors 0 dropped 0 overruns 0 frame 0
12    TX packets 57 bytes 4608 (4.5 KiB)
13    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
14
15 #测试通信
16 root@rk3588-buildroot:/# ping www.baidu.com -I usb0
```

4.1.7. mSATA

mSATA与4G/5G复用接口，如下图所示：

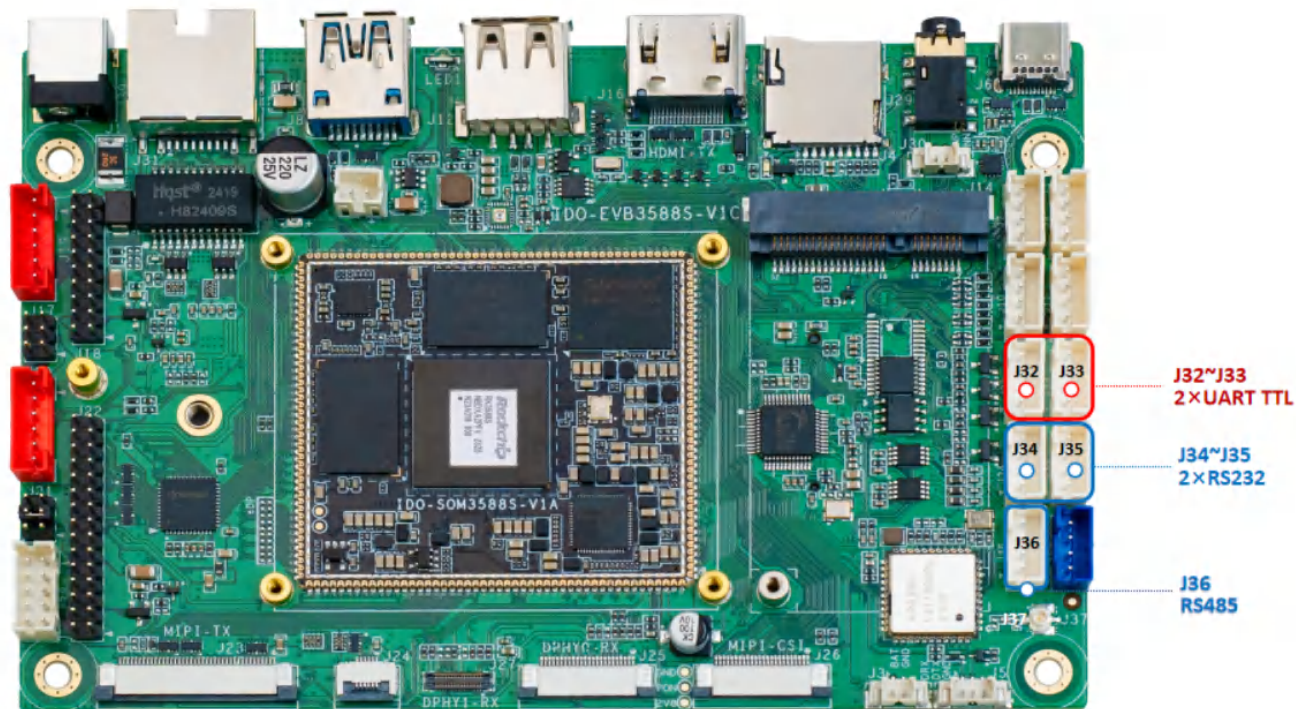


注意：

1. 默认配置为 4G，支持 mSATA 需要修改软硬件

4.1.8. UART

主板一共引出5路串口（不含调试串口），如下图所示：



UART设备节点列表如下：

序号	功能	设备节点
0 (J32)	TTL	/dev/ttyS0
3 (J33)	TTL	/dev/ttyS3
4 (J34)	RS232	/dev/ttyS4
7 (J35)	RS232	/dev/ttyS7
6 (J36)	RS485	/dev/ttyS6

```

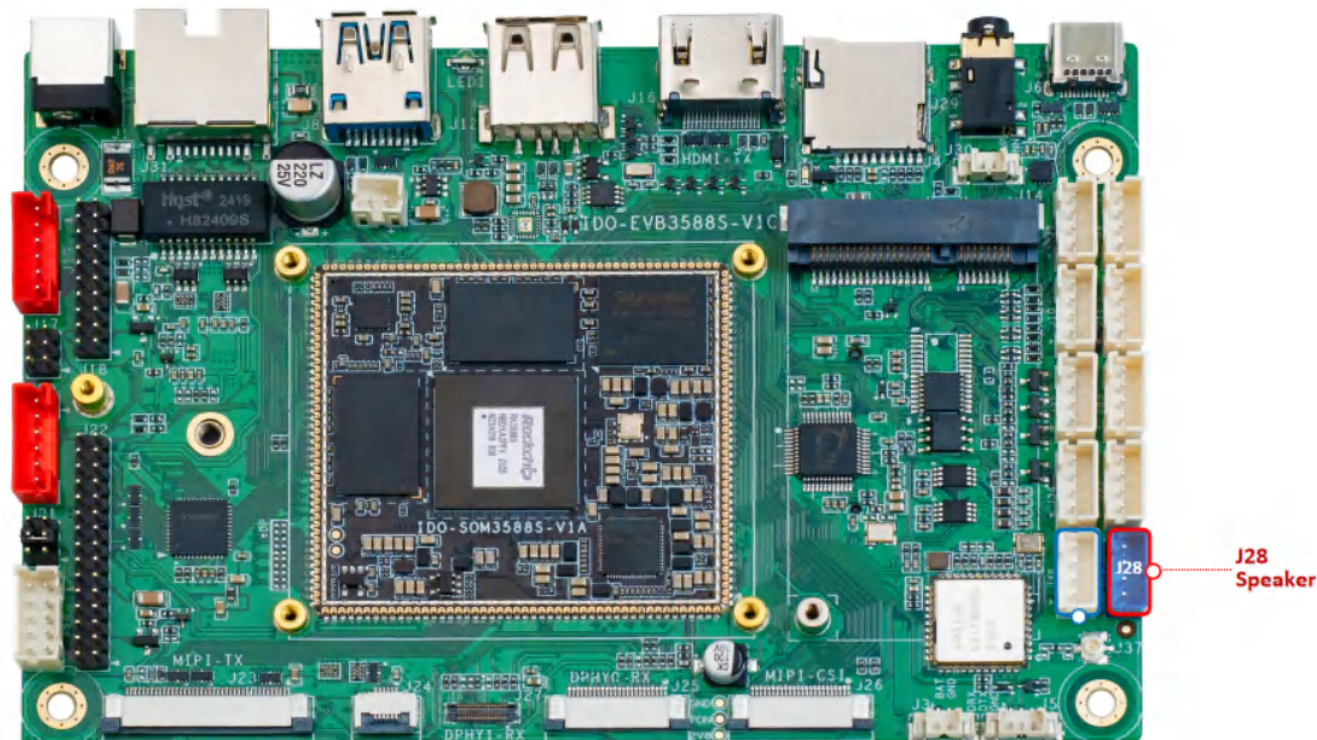
1 root@rk3588-buildroot:/# microcom /dev/ttyS4 -s 115200

```

4.1.9. 声音

4.1.9.1. 喇叭

喇叭接口为PH2.0-4P连接器J28，如下图所示：



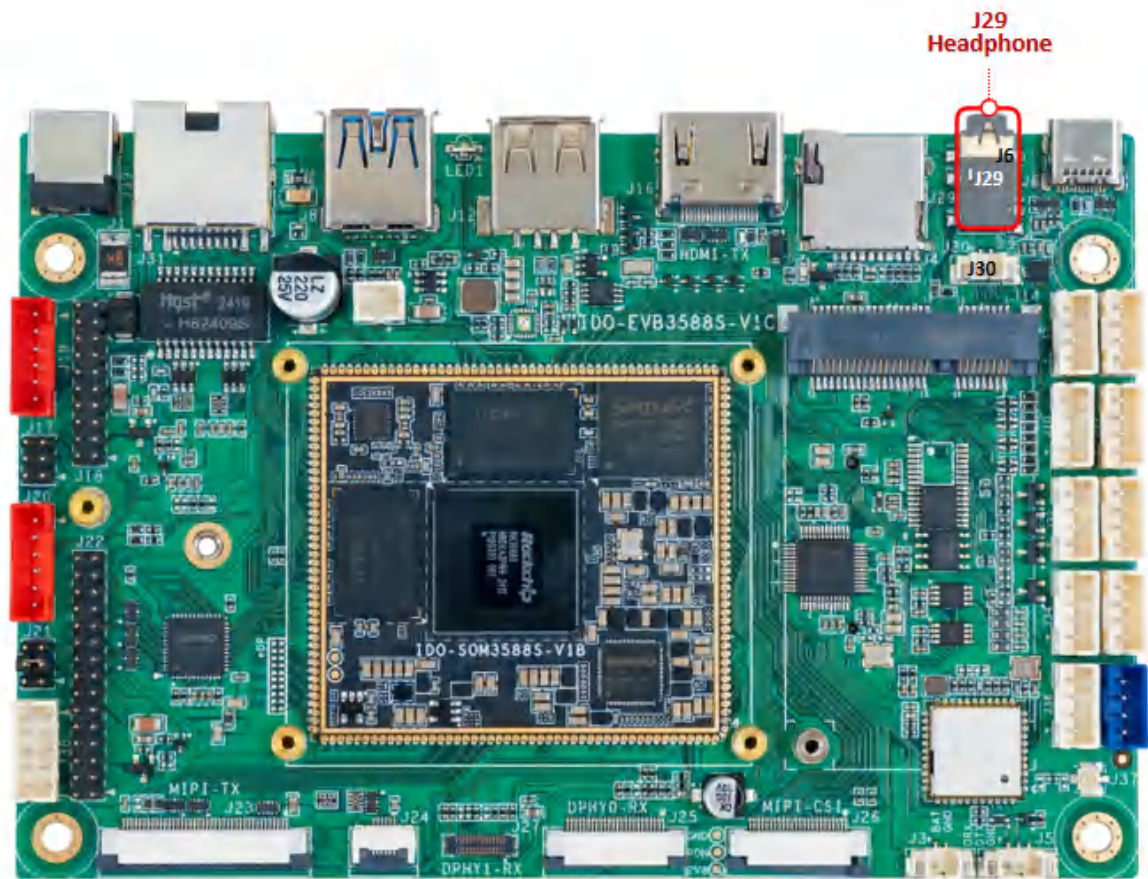
- 支持双声道，每个声道支持4Ω 3W输出

Plain Text

```
1 #查看声卡
2 root@rk3588-buildroot:/# aplay -l
3
4 #音频播放到DP
5 root@rk3588-buildroot:/# aplay -D hw:1,0 xihuangni.wav
6 Playing WAVE 'xihuangni.wav' : Signed 16 bit Little Endian, Rate 44100 H
z, Stereo
7
8 #音频播放到HDMI
9 root@rk3588-buildroot:/# aplay -D hw:2,0 xihuangni.wav
10 Playing WAVE 'xihuangni.wav' : Signed 16 bit Little Endian, Rate 44100 H
z, Stereo
11
12 #音频播放到声卡
13 root@rk3588-buildroot:/# aplay -D hw:0,0 xihuangni.wav
14 Playing WAVE 'xihuangni.wav' : Signed 16 bit Little Endian, Rate 44100 H
z, Stereo
```

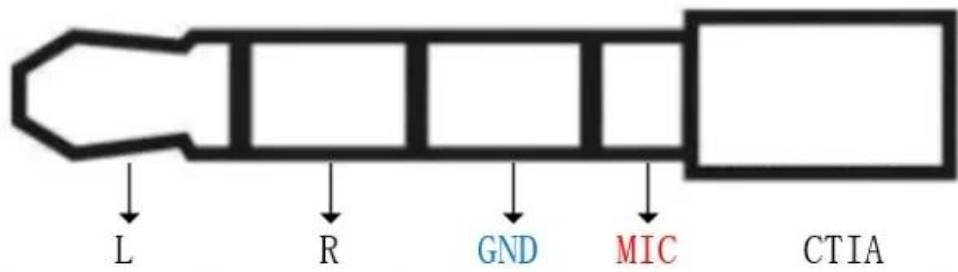
4.1.9.2. 耳机

主板支持1路3.5mm四节耳机座J29，如下图所示：



- 支持耳机检测
- 支持耳机录音

CTIA标准四段式耳机，定义如下：

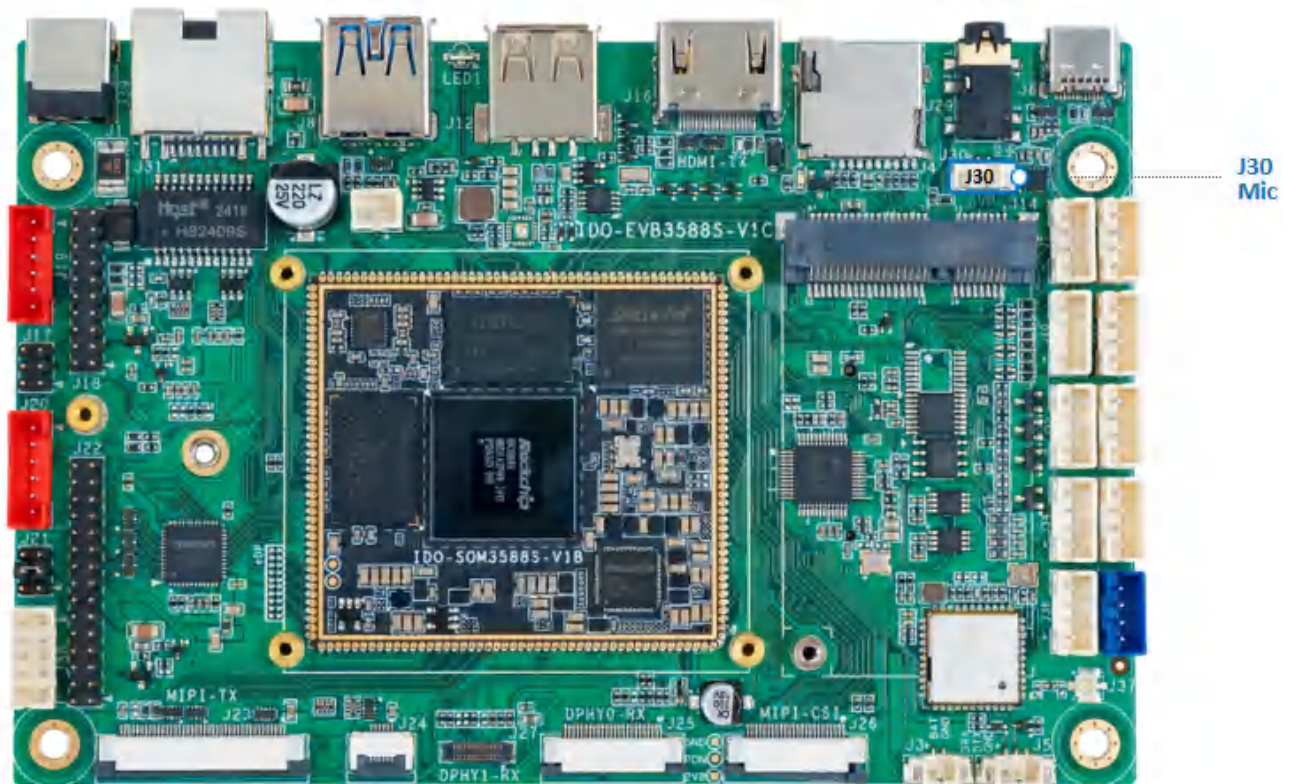


耳机录音：

```
1 #耳机录音
2 root@rk3588-buildroot:/# amixer -c 0 cset numid=43 on
3 root@rk3588-buildroot:/# echo -n "mic1" > /sys/class/es8388_mic/mic_channel
4
5 root@rk3588-buildroot:/# arecord -D hw:0,0 -f cd -r 44100 -c 2 test.wav
6 Recording WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

4.1.9.3. Mic

MX1.25T-2P麦克风接口J30，如下图所示：

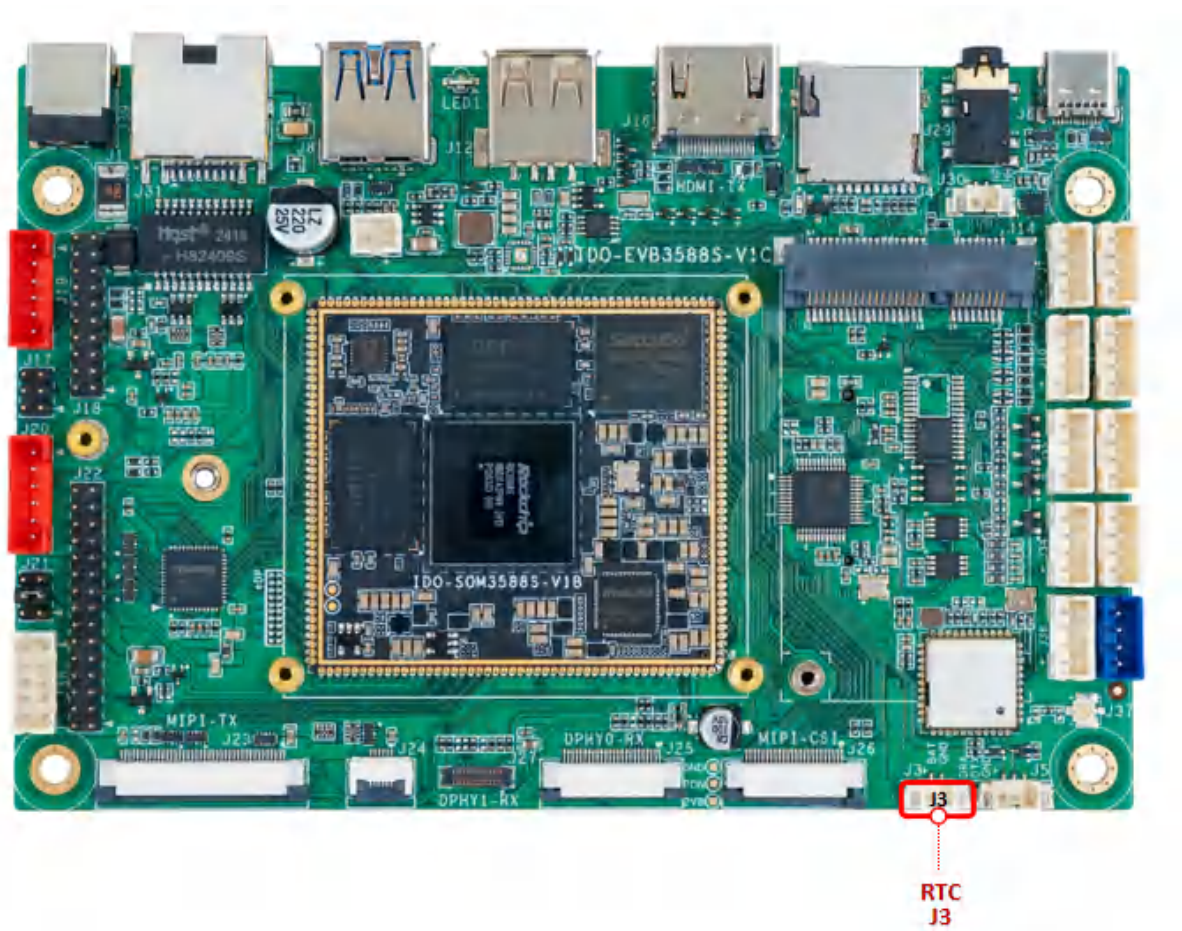


Mic录音：

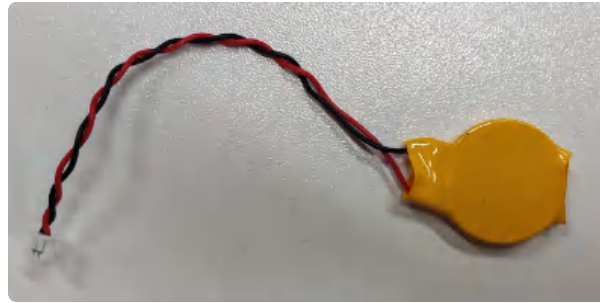
```
1 #Mic录音
2 root@rk3588-buildroot:/# amixer -c 0 cset numid=43 on
3 root@rk3588-buildroot:/# echo -n "mic2" > /sys/class/es8388_mic/mic_channel
4 root@rk3588-buildroot:/# arecord -D hw:0,0 -f S16_LE -r 44100 -c 2 test.wav
5 Recording WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

4.1.10. RTC

MX1.25T-2P RTC电池座J3，如下图所示：



连接3V 纽扣电池，RTC电池参考如下



rtc时间设置方法：

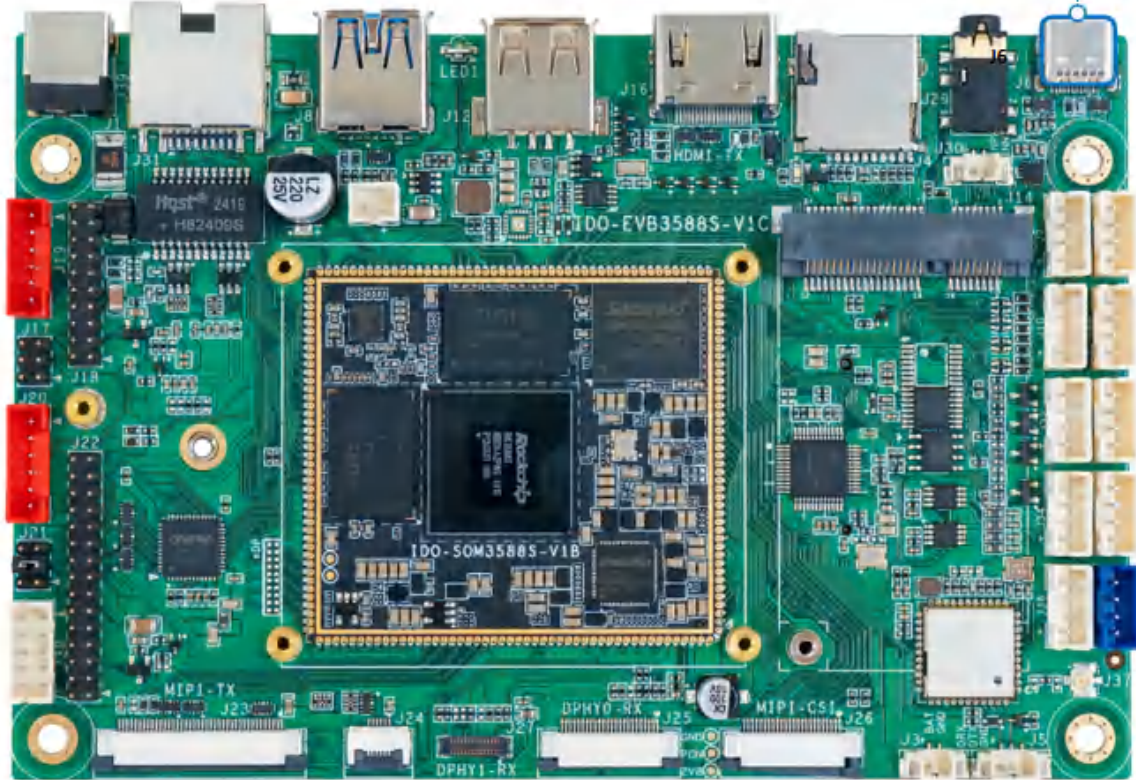
```
1  #设置时间
2  root@rk3588-buildroot:/# date -s "2025-10-10 10:10:10"
3
4  #将rtc时钟调整为与目前的系统时钟一致
5  root@rk3588-buildroot:/# hwclock -w
6
7  #获取硬件rtc当前时间（断电重启读取时间没有太大偏差）
8  root@rk3588-buildroot:/# hwclock -r
9  2025-10-10 10:11:15.381196+08:00
```

4.1.11. USB接口

主板支持1个TypeC接口，支持2个USB-A，4个USB-PH2.0接口，USB对外总供电应小于3A。

4.1.11.1. TypeC 接口

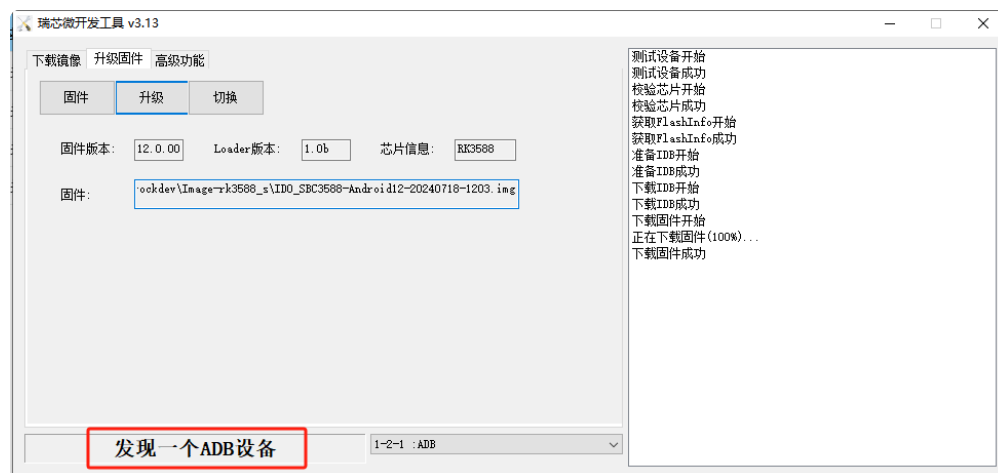
TypeC接口J6，如下图所示：



- 支持Host、Device模式自动切换
- 支持DP显示输出

4.1.11.1.1. 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：



- 可使用ADB相关开发工具对盒子进行功能调试

4.1.11.1.2. 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：



- 可识别U盘、键盘、鼠标并正常使用

4.1.11.1.3. DP模式

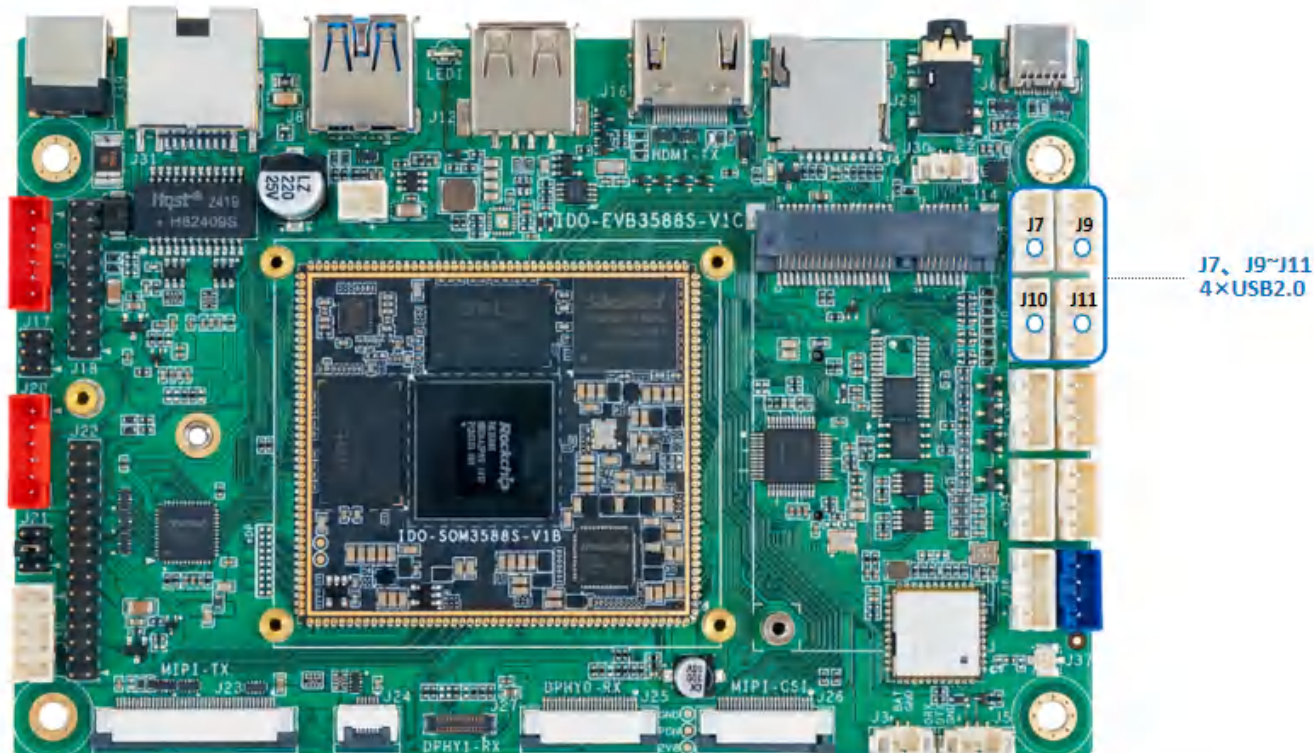
通过TypeC全功能数据线接入DP显示器，或通过TYPE-C to HDMI数据线连接HDMI显示器



- 主板画面通过TYPE-C DP输出正常，DP声音输出正常

4.1.11.2. USB PH2.0

支持四个USB PH2.0，如下图所示：



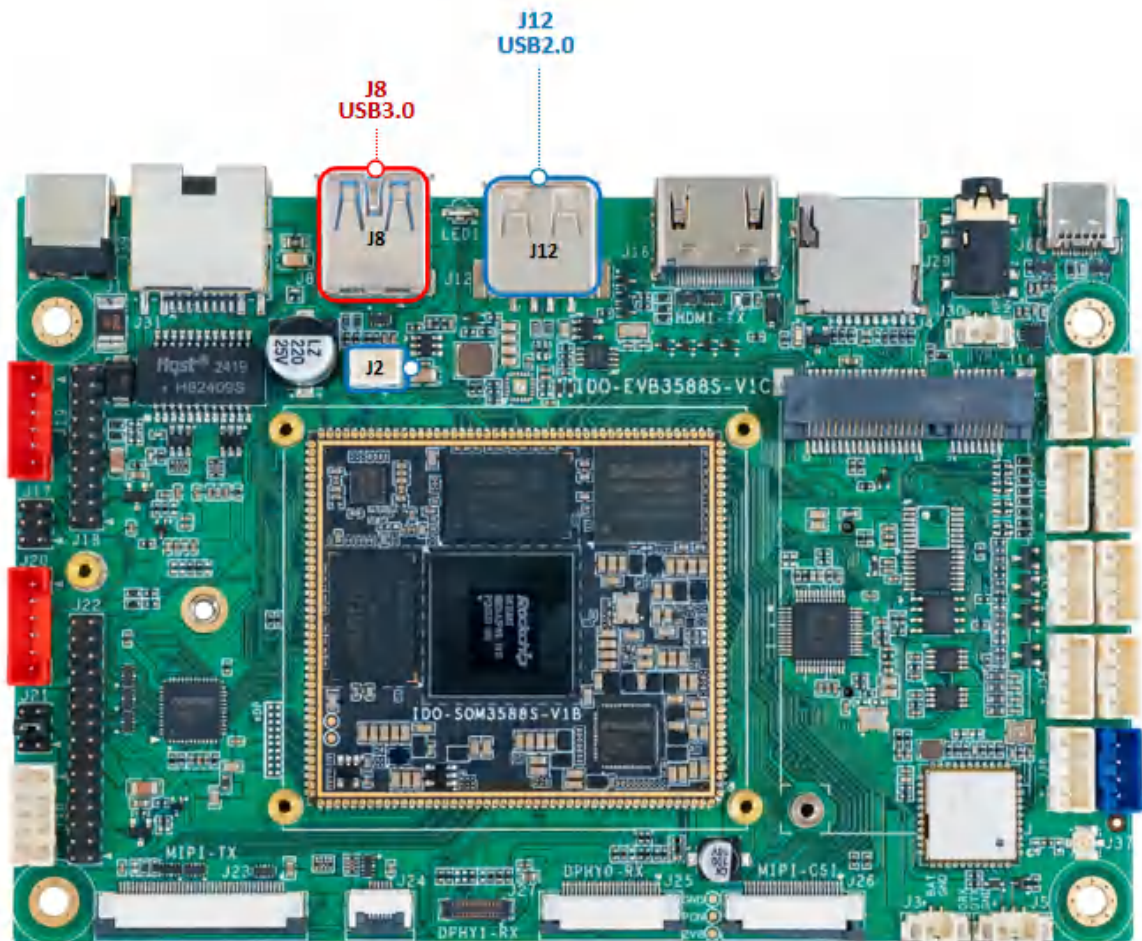
USB电源控制，如下表所示：

USB端口	默认状态	动作	命令
J7	开启	关闭电源	echo 0 > /sys/class/leds/usb_j7/brightness
		开启电源	echo 1 > /sys/class/leds/usb_j7/brightness
J9	开启	关闭电源	echo 0 > /sys/class/leds/usb_j9/brightness
		开启电源	echo 1 > /sys/class/leds/usb_j9/brightness
J10	开启	关闭电源	echo 0 > /sys/class/leds/usb_j10/brightness
		开启电源	echo 1 > /sys/class/leds/usb_j10/brightness
J11	开启	关闭电源	echo 0 > /sys/class/leds/usb_j11/brightness
		开启电源	echo 1 > /sys/class/leds/usb_j11/brightness

供电控制说明：设备节点写"0"关闭电源，写"1"开启电源。

4.1.11.3. USB接口

USB-A接口，如下图所示：



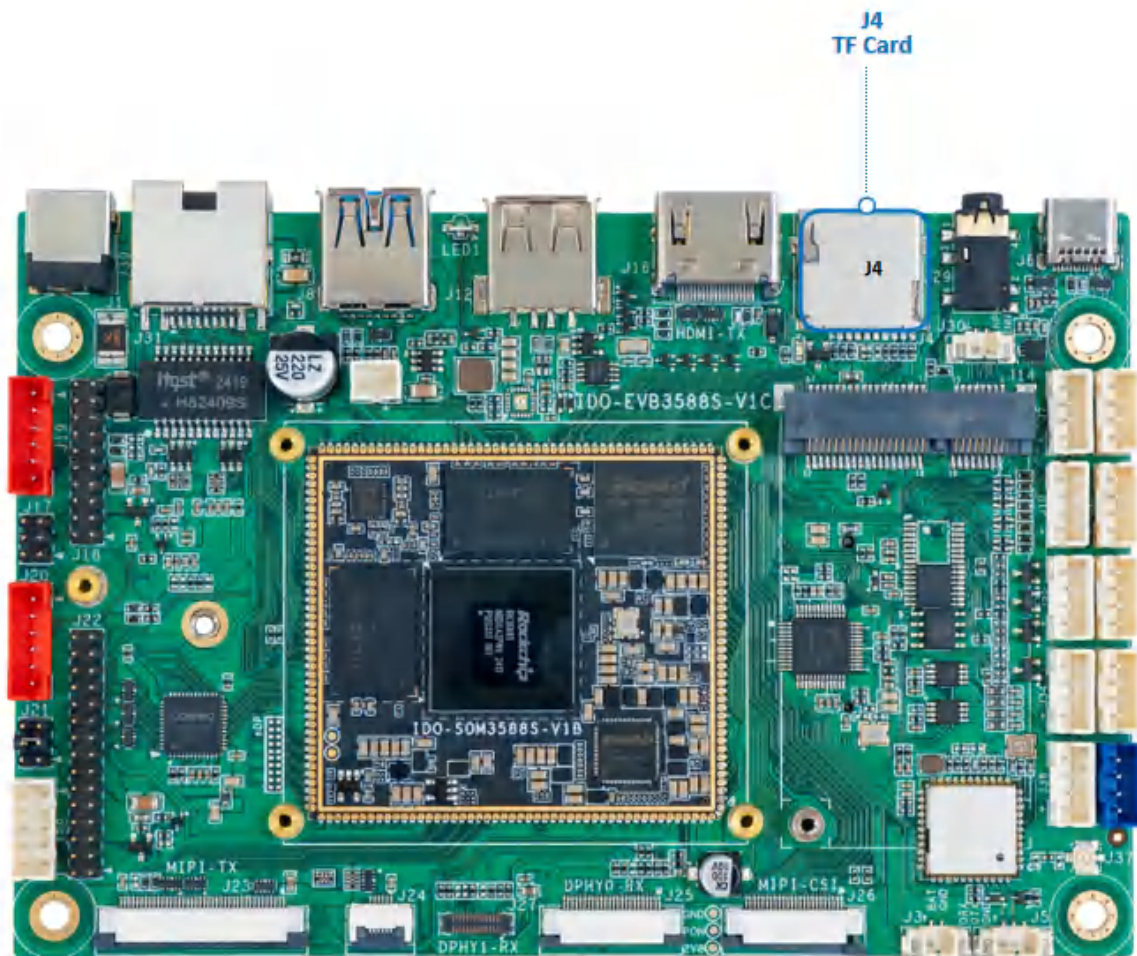
USB电源控制，如下表所示：

USB端口	默认状态	动作	命令
J8	开启	关闭电源	<code>echo 0 > /sys/class/leds/usb_j8/brightness</code>
		开启电源	<code>echo 1 > /sys/class/leds/usb_j8/brightness</code>
J12	开启	关闭电源	<code>echo 0 > /sys/class/leds/usb_j12/brightness</code>
		开启电源	<code>echo 1 > /sys/class/leds/usb_j12/brightness</code>

供电控制说明：设备节点写"0"关闭电源，写"1"开启电源。

4.1.12. TF Card

TF卡座接口位于J4，如下图所示：



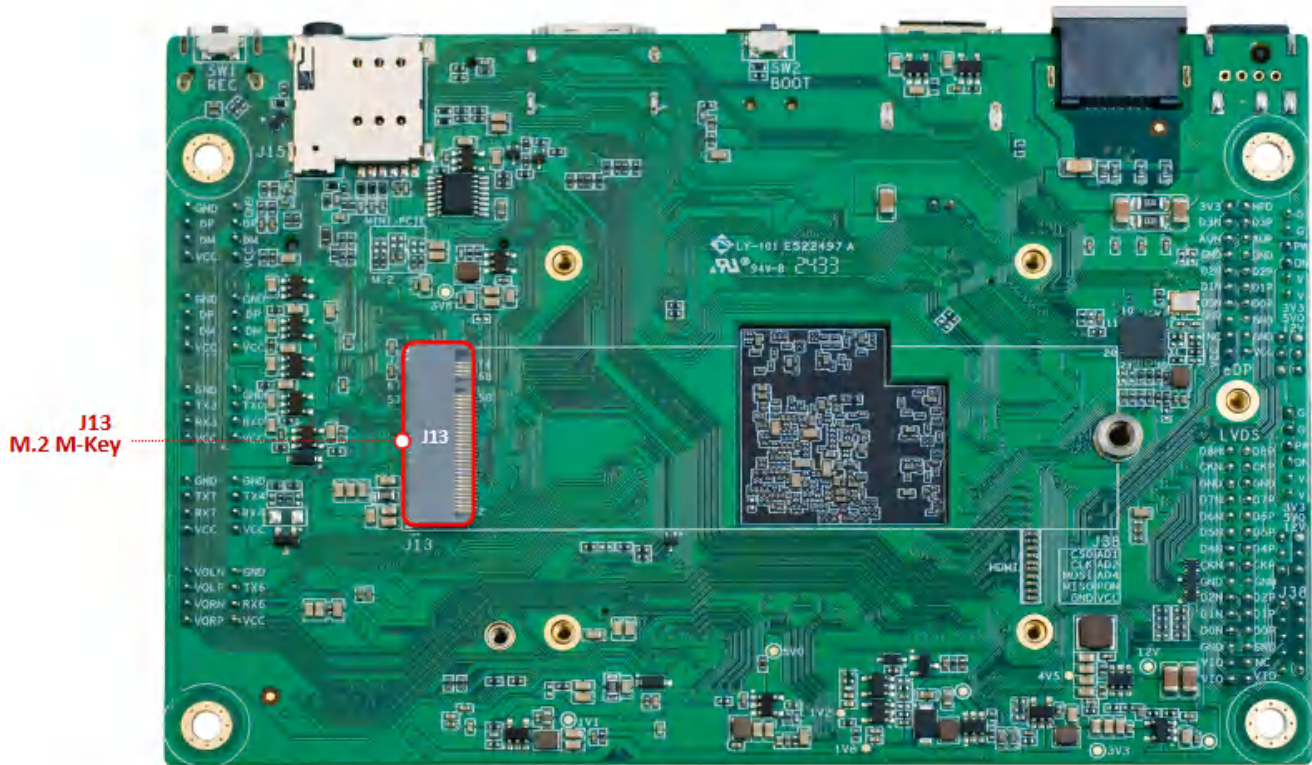
```

1  root@rk3588-buildroot:/# fdisk -l
2  Disk /dev/ram0: 8 MiB, 8388608 bytes, 16384 sectors
3  Units: sectors of 1 * 512 = 512 bytes
4  Sector size (logical/physical): 512 bytes / 4096 bytes
5  I/O size (minimum/optimal): 4096 bytes / 4096 bytes
6  .....
7
8  Device  Boot StartCHS      EndCHS          StartLBA      EndLBA      Sectors  Siz
   Id Type
9  /dev/sda1    0,0,33      1023,254,63          32    30535679    30535648  14.5
   G  c Win95 FAT32 (LBA)
10
11 root@rk3588-buildroot:/# df -h
12 Filesystem      Size  Used Avail Use% Mounted on
13 /dev/root        111G   1.1G  105G   1% /
14 devtmpfs         3.9G   8.0K   3.9G   1% /dev
15 tmpfs            3.9G  328K   3.9G   1% /tmp
16 tmpfs            3.9G  636K   3.9G   1% /run
17 .....
18 /dev/sda1        15G   32K   15G   1% /mnt/udisk/6C39-1D85

```

4.1.13. M.2接口

主板上使用标准M.2-M-key M.2接口连接座J13，如下图所示：



支持PCIe2.1通信，适用2280尺寸NVME固态硬盘。

▼

Shell

```

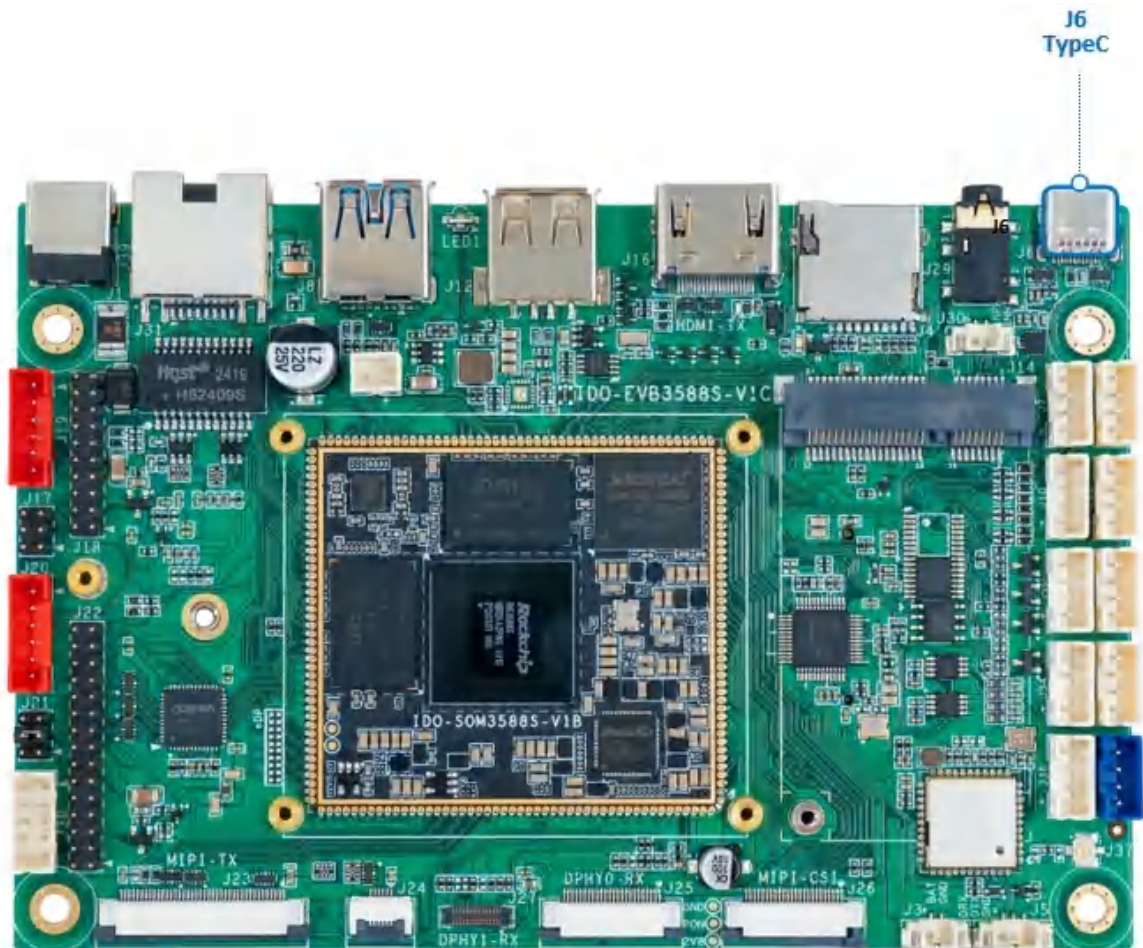
1  root@rk3588-buildroot:/# fdisk -l
2  Disk /dev/nvme0n1: 233 GB, 250059350016 bytes, 488397168 sectors
3  30401 cylinders, 255 heads, 63 sectors/track
4  Units: sectors of 1 * 512 = 512 bytes
5
6  Device          Boot StartCHS      EndCHS          StartLBA        EndLBA        Sector
7  s  Size Id Type
8  /dev/nvme0n1p1    0,32,33      1023,254,63      2048    488397167    488395120
9  232G  7 HPFS/NTFS
10
11 root@rk3588-buildroot:/# df -h
12 文件系统      大小  已用  可用  已用% 挂载点
13  Filesystem    Size  Used Avail Use% Mounted on
14  /dev/root      111G  1.1G  105G   1% /
15  .....
16  /dev/nvme0n1p1 233G   15G  219G   7% /mnt/m2

```

4.1.14. LCD显示

4.1.14.1. DP

DP接口（TYPE-C）J6，如下图所示：



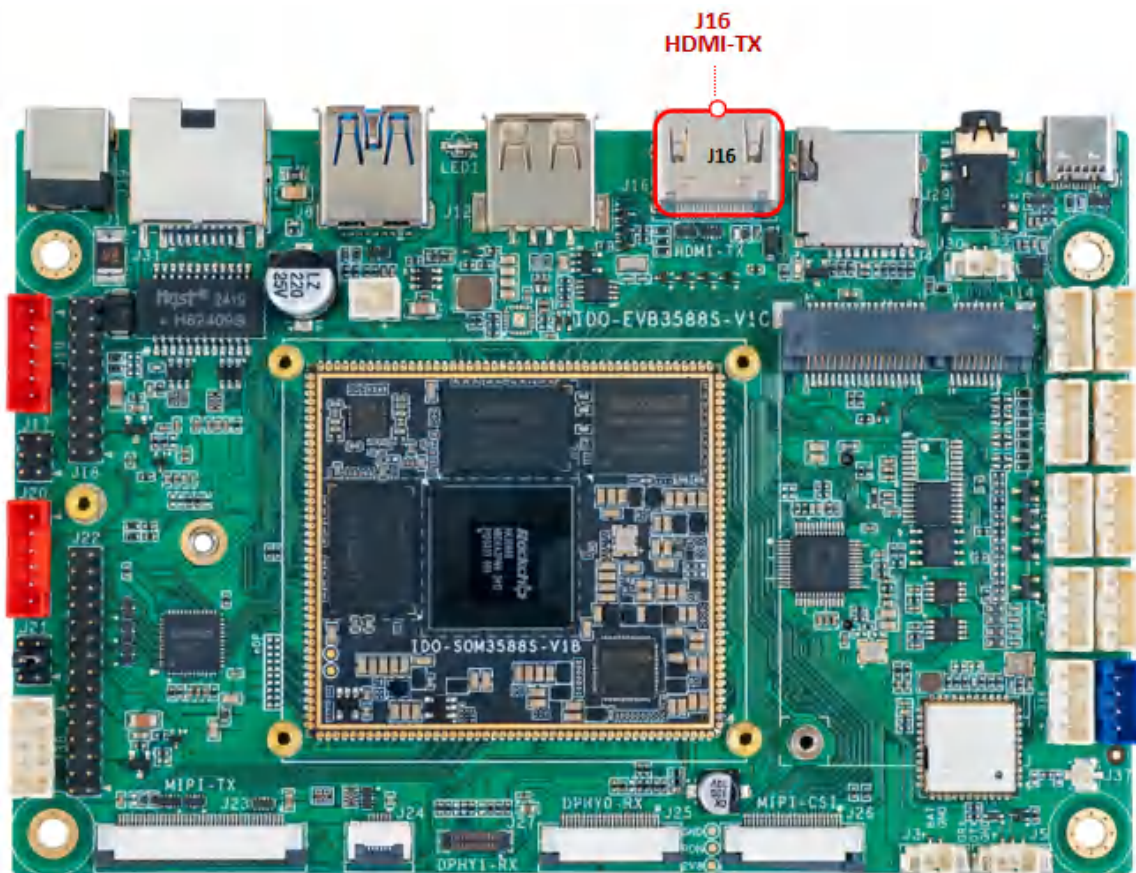
可以使用 USB-C 转 HDMI 高清线连接 HDMI 显示器输出画面，如下图所示：



- 最高支持8K@30fps输出

4.1.14.2. HDMI-TX

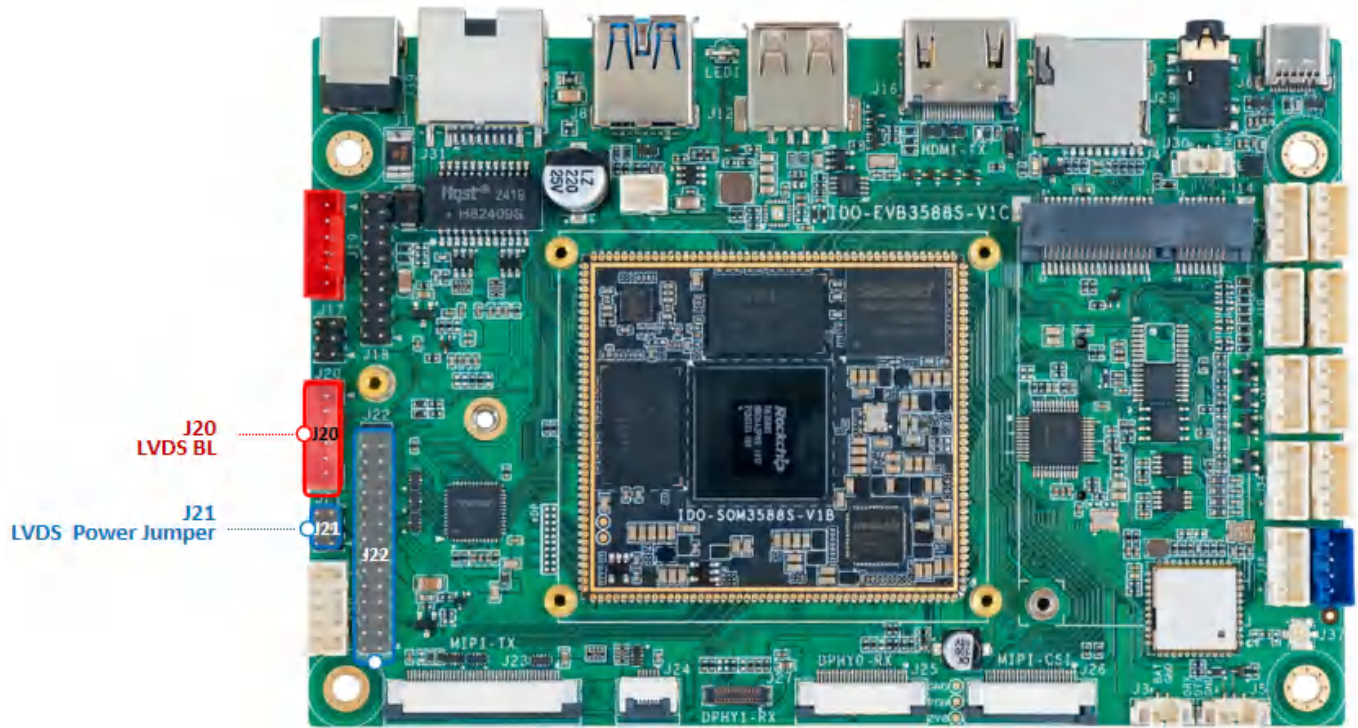
主板标准HDMI-A接口，如下图所示：



- 最高支持 HDMI2.1 8K@60fps 输出

4.1.14.3. Dual LVDS屏

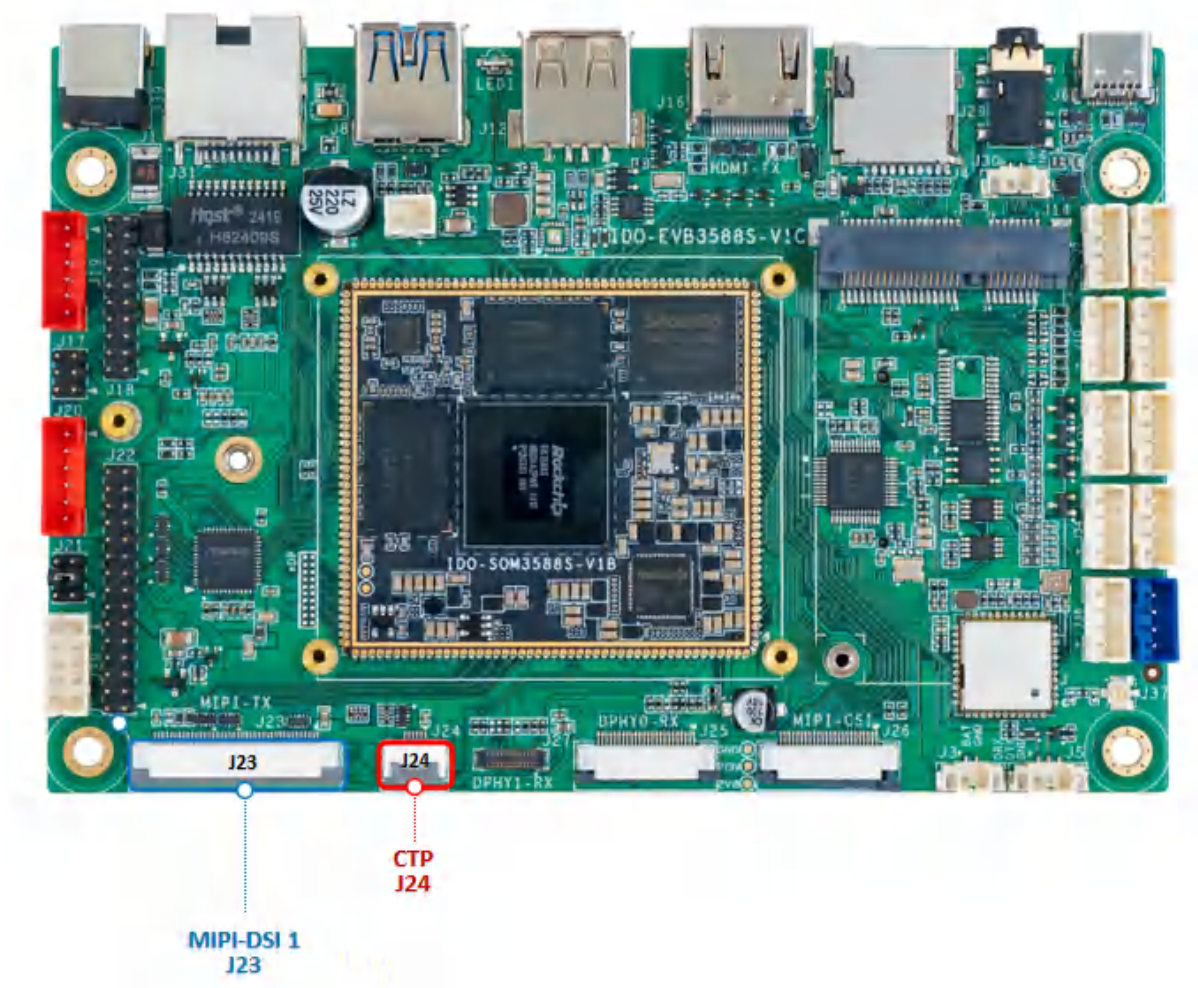
Dual LVDS接口J22，背光接 J20，屏幕供电电压选择接口J21，如下图所示：



J21屏幕供电电压选择接口，接屏前，需要根据具体的屏幕规格书来确认供电跳线帽接到3.3V、5V或12V，默认3.3V。

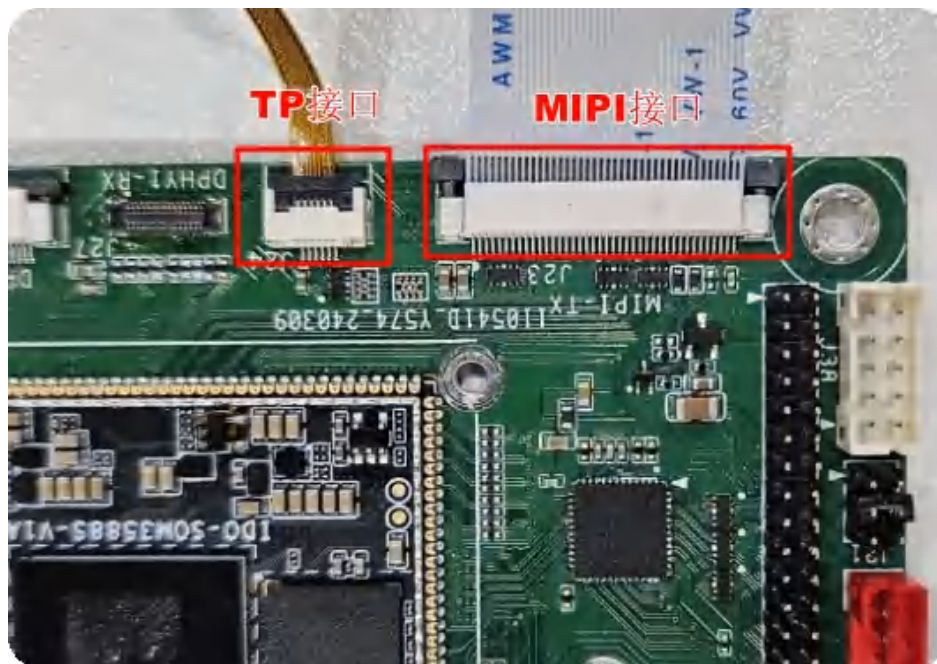
4.1.14.4. MIPI

MIPI接口J23，如下图所示：



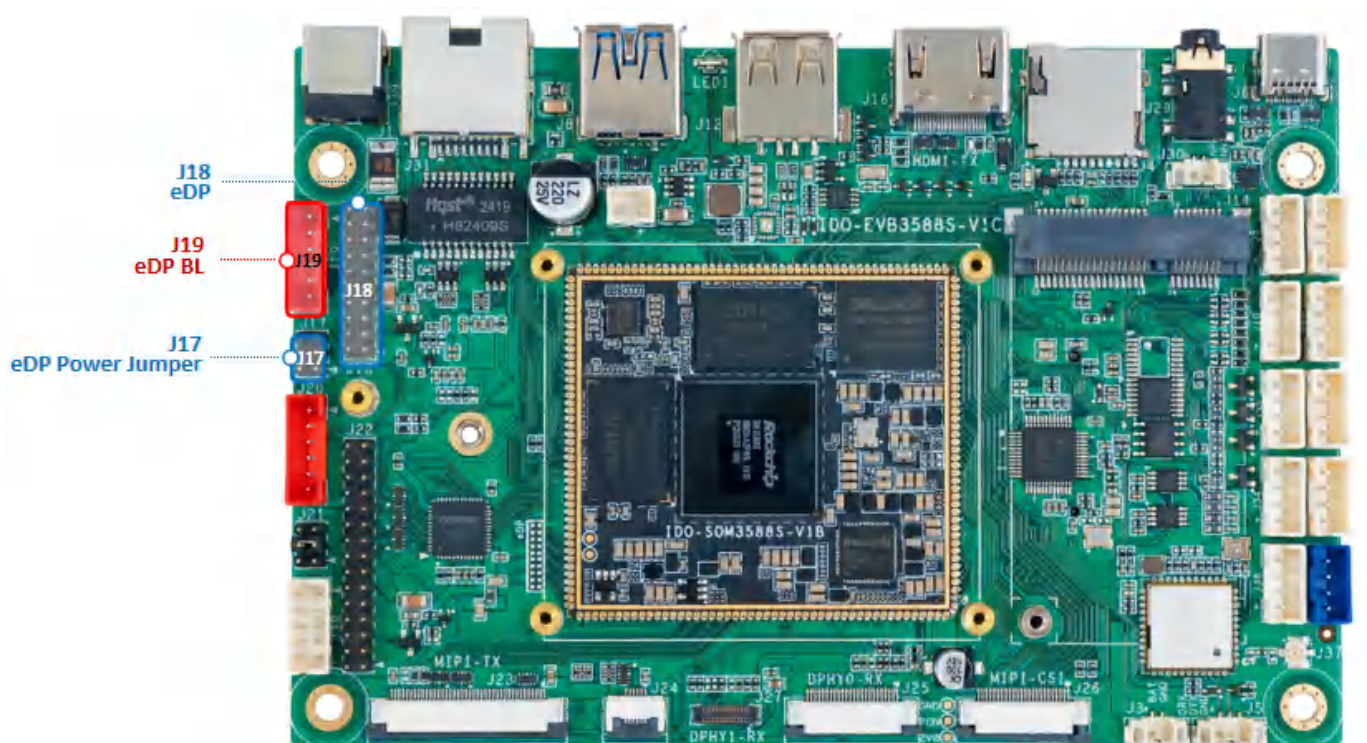
- MIPI供电为3.3V
- MIPI屏LCD排线接线：
- 触摸 TP 接口，下接
- 主板 LCD 排线接线（上接），40Pin FPC 屏座子

接法如下：



4.1.14.5. EDP

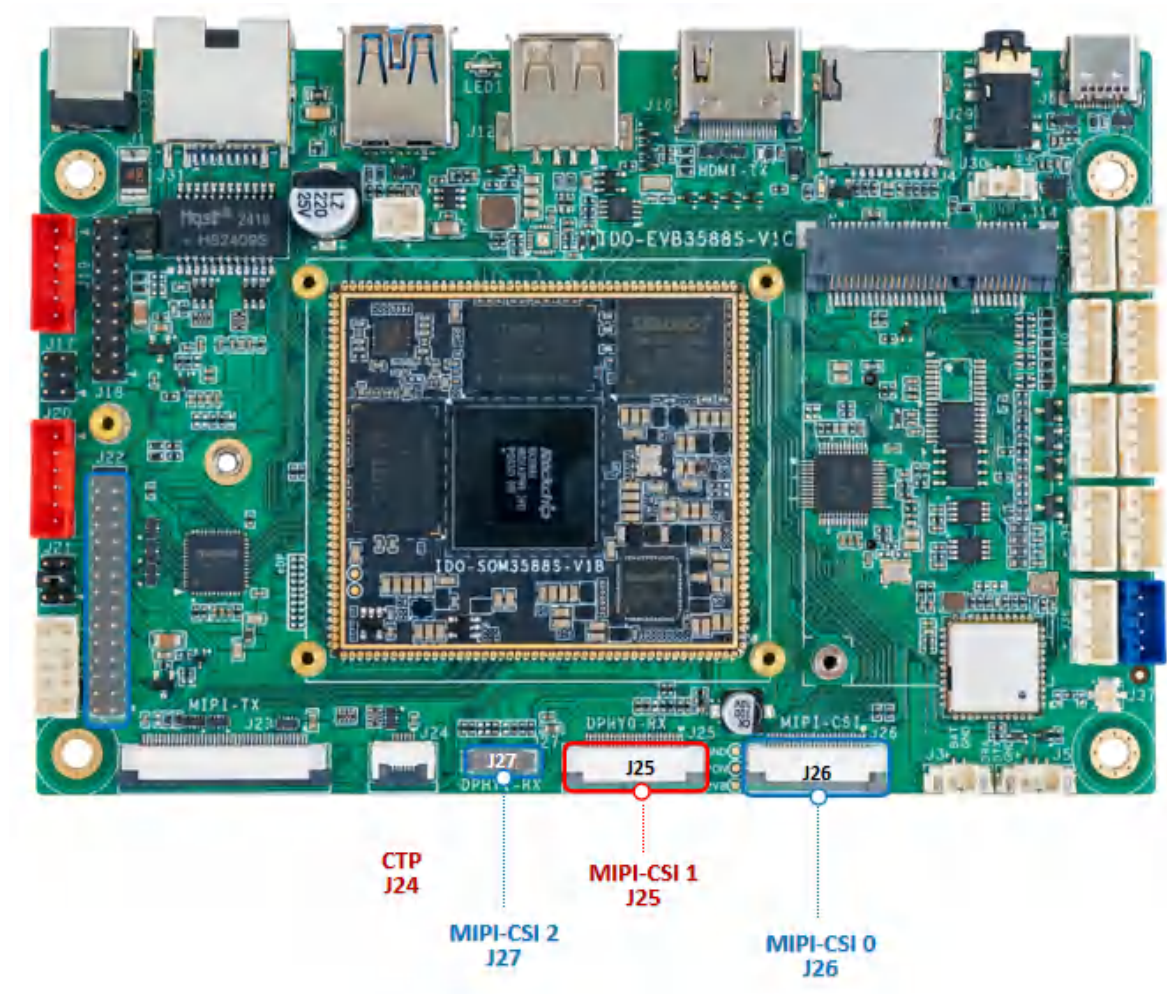
eDP接口，如下图所示：



注意：eDP与HDMI互斥，默认配置为 HDMI 功能，使用 eDP 需要更改硬件和软件

4.1.15. MIPI Camera

MIPI Camer接口J25和J26和J27，如下图所示：



MIPI摄像头接线，如下图所示：



4.1.15.1. MIPI-CSI1

4.1.15.1.1. 预览

▼ Plain Text |

```
1  gst-launch-1.0 v4l2src device=/dev/video33 ! video/x-raw, format=NV12, width=1920, height=1080, framerate=30/1 ! videoconvert ! autovideosink sync=false
```

4.1.15.1.2. 抓图

```
1 root@rk3588-buildroot:/# v4l2-ctl --verbose -d /dev/video33 --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
```

4.1.15.2. MIPI-CSI0

4.1.15.2.1. 预览

```
1 gst-launch-1.0 v4l2src device=/dev/video42 ! video/x-raw, format=NV12, width
h=1920, height=1080, framerate=30/1 ! videoconvert ! autovideosink sync=fal
se
```

4.1.15.2.2. 抓图

```
1 root@rk3588-buildroot:/# v4l2-ctl --verbose -d /dev/video42 --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
```

4.1.15.3. MIPI-CSI2

4.1.15.3.1. 预览

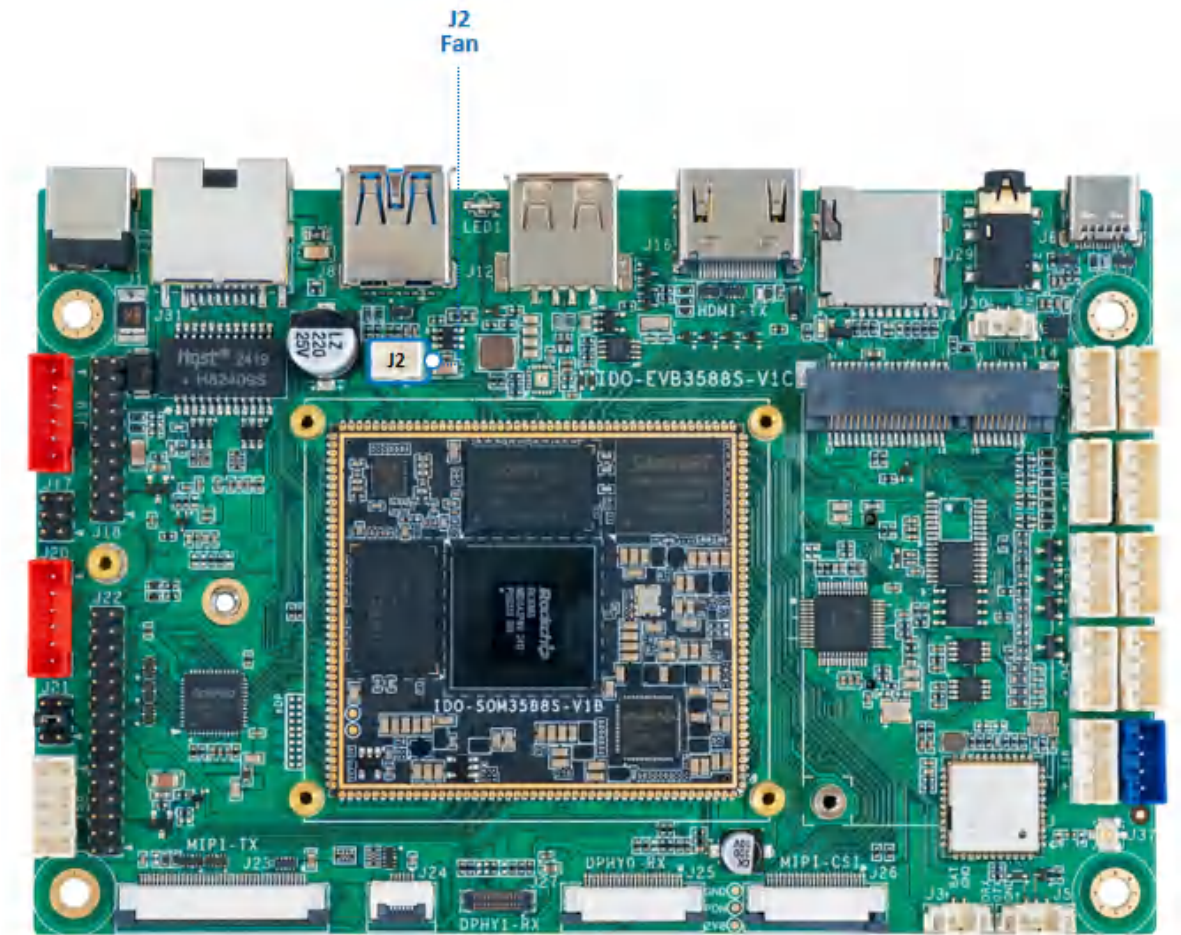
```
1 gst-launch-1.0 v4l2src device=/dev/video51 ! video/x-raw, format=NV12, width
h=1920, height=1080, framerate=30/1 ! videoconvert ! autovideosink sync=fal
se
```

4.1.15.3.2. 抓图

```
1 root@rk3588-buildroot:/# v4l2-ctl --verbose -d /dev/video51 --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
```

4.1.16. FAN 风扇

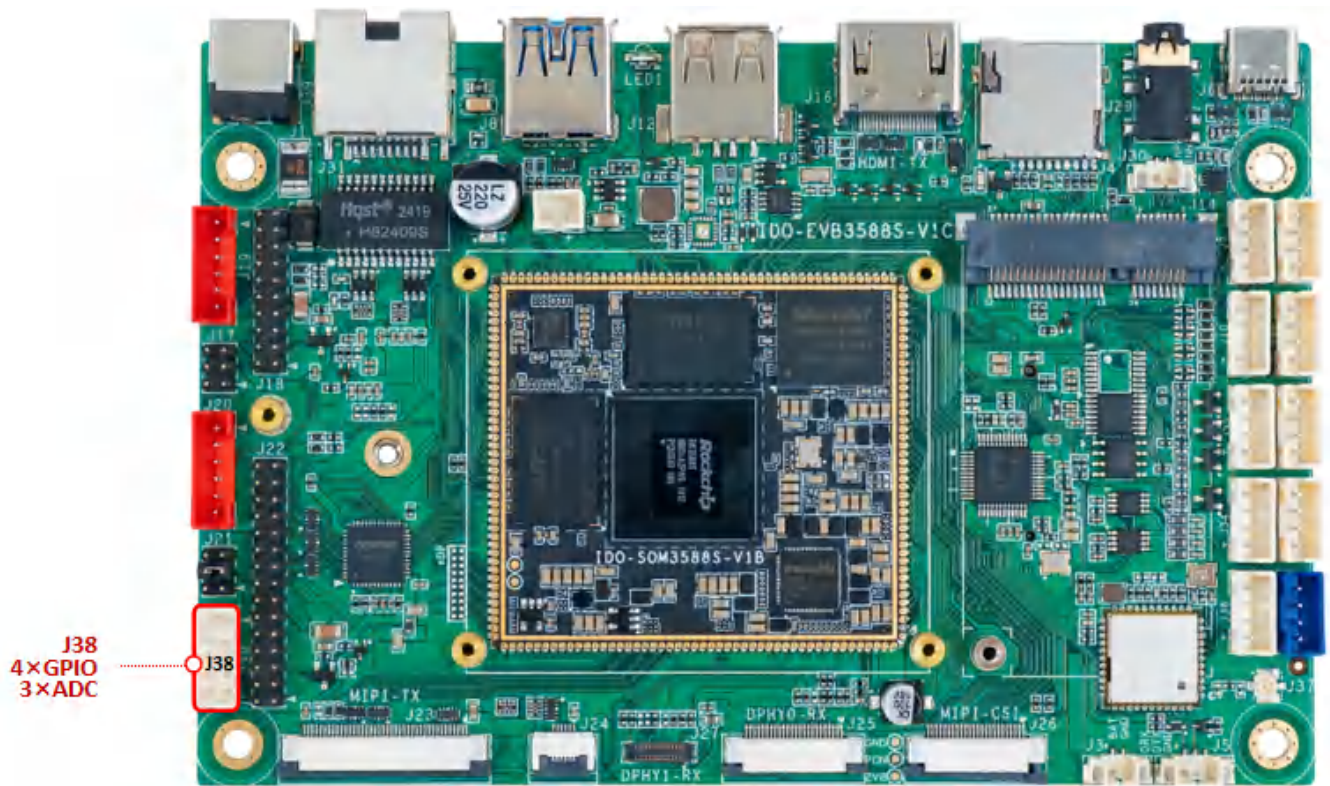
PH2 5V FAN风扇接口J2，如下图所示：



- 在CPU高于 55℃ 风扇自动开启，低于 50℃ 风扇自动关闭

4.1.17. GPIO

GPIO接口，如下图所示：



序号	管脚标号	GPIO 标号	GPIO 序号
1	CS0	GPIO1_A3	35
3	CLK	GPIO1_A2	34
5	MOSI	GPIO1_A1 (被占用，默认为 RS485)	33
7	MISO	GPIO1_A0 (被占用，默认为 RS485)	32
9	GND	地	/
2	AD1	recovery 键	/
4	AD2	ADC2	/
6	AD4	ADC4	/
8	PON	Power 键	/
10	VCC	3.3V 供电	/

GPIO 控制方式

```
▼ Plain Text |
1  输出:
2  # gpio1_A3 拉高
3  gpioset 1 3=1
4
5  # gpio1_A3 拉低
6  gpioset 1 3=0
7
8  # gpio2_A2 拉高
9  gpioset 1 2=1
10
11 # gpio2_A2 拉低
12 gpioset 1 2=0
13
14 输入:
15 # gpio2_A3 输入获取电平状态
16 gpioget 1 3
17
18 # gpio2_A2 输入获取电平状态
19 gpioget 1 2
```

4.2. GPU

执行glmark2-es2-wayland 开启gpu测试窗口:

```

1 root@rk3588-buildroot:/# glmark2-es2-wayland
2 arm_release_ver: g24p0-00eac0, rk_so_ver: 6
3 =====
4     glmark2 2023.01
5     =====
6     OpenGL Information
7     GL_VENDOR:      ARM
8     GL_RENDERER:    Mali-G610
9     GL_VERSION:     OpenGL ES 3.2 v1.g24p0-00eac0.6b39a7384565dd0742f84d5a
10    b1cdf9ef
11    Surface Config:  buf=32 r=8 g=8 b=8 a=8 depth=24 stencil=0 samples=0
12    Surface Size:    800x600 windowed
13    =====
14 [build] use-vbo=false: FPS: 1039 FrameTime: 0.963 ms
15 [build] use-vbo=true: FPS: 1149 FrameTime: 0.871 ms
16 [texture] texture-filter=nearest: FPS: 1179 FrameTime: 0.848 ms
17 [texture] texture-filter=linear: FPS: 1149 FrameTime: 0.871 ms
18 [texture] texture-filter=mipmap: FPS: 1173 FrameTime: 0.853 ms
19 [shading] shading=gouraud: FPS: 1060 FrameTime: 0.944 ms
20 [shading] shading=blinn-phong-inf: FPS: 1057 FrameTime: 0.946 ms
21 [shading] shading=phong: FPS: 1032 FrameTime: 0.970 ms
22 [shading] shading=cel: FPS: 1054 FrameTime: 0.949 ms
23 [bump] bump-render=high-poly: FPS: 863 FrameTime: 1.160 ms
24 [bump] bump-render=normals: FPS: 1205 FrameTime: 0.830 ms
25 [bump] bump-render=height: FPS: 1177 FrameTime: 0.850 ms
26 [effect2d] kernel=0,1,0;1,-4,1;0,1,0;: FPS: 1206 FrameTime: 0.830 ms
27 [effect2d] kernel=1,1,1,1,1;1,1,1,1,1;1,1,1,1,1;: FPS: 964 FrameTime: 1.03
28 8 ms
29 [pulsar] light=false:quads=5:texture=false: FPS: 1133 FrameTime: 0.883 ms
30 [desktop] blur-radius=5:effect=blur:passes=1:separable=true:windows=4: FPS
31 : 779 FrameTime: 1.285 ms
32 [desktop] effect=shadow:windows=4: FPS: 970 FrameTime: 1.032 ms
33 [buffer] columns=200:interleave=false:update-dispersion=0.9:update-fractio
34 n=0.5:update-method=map: FPS: 467 FrameTime: 2.142 ms
35 [buffer] columns=200:interleave=false:update-dispersion=0.9:update-fractio
36 n=0.5:update-method=subdata: FPS: 448 FrameTime: 2.236 ms
37 [buffer] columns=200:interleave=true:update-dispersion=0.9:update-fraction
38 =0.5:update-method=map: FPS: 531 FrameTime: 1.885 ms
39 [ideas] speed=duration: FPS: 964 FrameTime: 1.038 ms
40 [jellyfish] <default>: FPS: 1011 FrameTime: 0.990 ms
41 [terrain] <default>: FPS: 263 FrameTime: 3.812 ms
42 [shadow] <default>: FPS: 1012 FrameTime: 0.988 ms
43 [refract] <default>: FPS: 390 FrameTime: 2.567 ms
44 [conditionals] fragment-steps=0:vertex-steps=0: FPS: 1198 FrameTime: 0.835
45 ms

```

```

39 [conditionals] fragment-steps=5:vertex-steps=0: FPS: 1179 FrameTime: 0.849
40 ms
41 [conditionals] fragment-steps=0:vertex-steps=5: FPS: 1144 FrameTime: 0.875
42 ms
43 [function] fragment-complexity=low:fragment-steps=5: FPS: 1126 FrameTime:
44 0.888 ms
45 [function] fragment-complexity=medium:fragment-steps=5: FPS: 1136 FrameTim
46 e: 0.881 ms
47 [loop] fragment-loop=false:fragment-steps=5:vertex-steps=5: FPS: 1173 Fram
48 eTime: 0.853 ms
[loop] fragment-steps=5:fragment-uniform=false:vertex-steps=5: FPS: 1122 F
rameTime: 0.892 ms
[loop] fragment-steps=5:fragment-uniform=true:vertex-steps=5: FPS: 1120 Fr
ameTime: 0.893 ms
=====
                                glmark2 Score: 983
=====

```

4.3. NPU

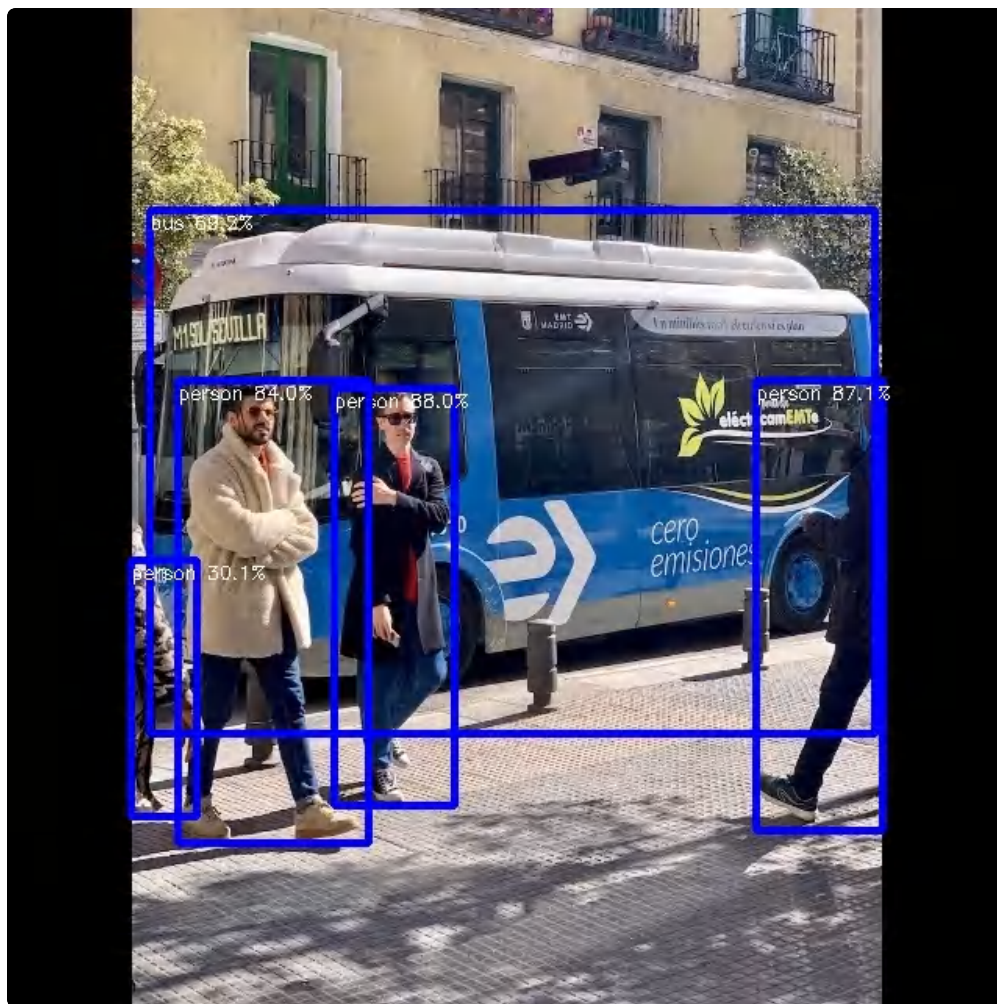
使用yolov5进行rknn模型转换测试，如下所示：

```

1 root@rk3588-buildroot:/# tar -xvf install.tar.gz
2 root@rk3588-buildroot:/# cd install
3 root@rk3588-buildroot:/# chmod a+x rknn_yolov5_demo
4 root@rk3588-buildroot:/# ./rknn_yolov5_demo ./model/RK3588/yolov5s-640-64
0.rknn ./model/bus.jpg
5 post process config: box_conf_threshold = 0.25, nms_threshold = 0.45
6 Loading mode...
7 sdk version: 1.5.2 (c6b7b351a@2023-08-23T15:28:22) driver version: 0.9.8
8 model input num: 1, output num: 3
9   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=INT
8, qnt_type=AFFINE, zp=-128, scale=0.003922
10  index=0, name=output, n_dims=4, dims=[1, 255, 80, 80], n_elems=1632000,
size=1632000, w_stride = 0, size_with_stride=1638400, fmt=NCHW, type=INT
8, qnt_type=AFFINE, zp=-128, scale=0.003860
11  index=1, name=283, 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_t
ype=AFFINE, zp=-128, scale=0.003922
12  index=2, name=285, 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_t
ype=AFFINE, zp=-128, scale=0.003915
13 model is NHWC input fmt
14 model input height=640, width=640, channel=3
15 Read ./model/bus.jpg ...
16 img width = 640, img height = 640
17 once run use 61.330000 ms
18 loadLabelName ./model/coco_80_labels_list.txt
19 person @ (208 244 286 506) 0.884139
20 person @ (478 238 559 526) 0.867678
21 person @ (110 238 230 534) 0.824685
22 bus @ (91 129 554 467) 0.701223
23 person @ (79 353 122 516) 0.334380
24 save detect result to ./out.jpg
25 loop count = 10 , average run 56.958000 ms

```

执行完后会得到转换后的模型图片，如下所示：



4.4. QT5

支持qt5.15，测试方法如下：

```
1 root@rk3588-buildroot:/# chmod a+x ./moveblocks
2 root@rk3588-buildroot:/# ./moveblocks
```

结果如下所示：

