

# IDO-EVB3568-V1 - Debian 系统使用说明

---

1. 系统用户名及密码
2. 调试口使用
  - 2.1 串口调试
  - 2.2 ADB调试
3. 串口测试
4. CAN测试
5. WIFI使用
6. 蓝牙使用
7. 以太网使用
  - 7.1 静态IP设置
  - 7.2 DNS设置
8. 4G使用
9. 摄像头使用
10. U盘\TF挂载
11. 设置开机启动程序
  - 11.1 Debian10 系统
13. 按键
13. ADC
14. 时间设置
  - 14.1 RTC时间读取和同步
14. RTC
  - 方法一
    - 14.1 获取RTC时间
    - 14.2 设置RTC时间
  - 方法二
    - 14.3 时区修改
15. 音视频
  - 15.1 播放音频的工具

15.2 音量调节

5.10音频

16. 显示屏

16.1 显示屏接口说明

16.2 显示设置

16.2.1 屏幕背光亮度设置

17. 桌面shell终端

18. 浏览器

19. GPU测试



# IDO-EVB3568-V1

## Debian系统使用说明

深圳触觉智能科技有限公司

[www.industio.cn](http://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	luffy	FuYingz he	2022/03/1 7

## 1. 系统用户名及密码

文件系统版本	账户	密码
Debian10	root	未设置密码
	linaro	linaro

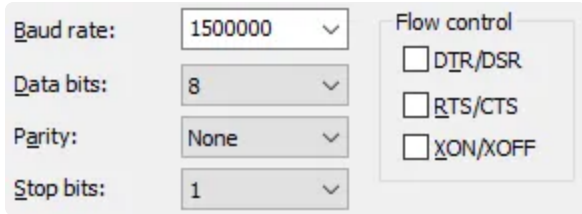
## 2. 调试口使用

### 2.1 串口调试



调试串口为 TTL 电平，开发板接口为 MX1.25 接线端子，使用 USB 转串口模块连接 PC 调试终端。

串口参数：波特率 1500000、数据位 8bit、无校验位、停止位 1bit。



## 2.2 ADB调试

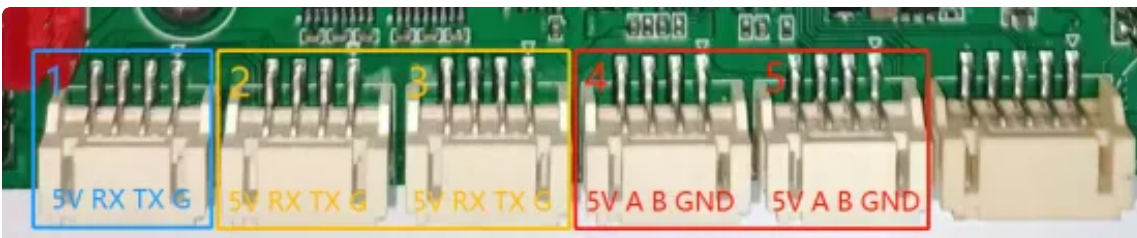


上图红色框内的USB接口为支持OTG模式切换，在系统上电前，使用双公头 USB 数据线连接开发板和 PC 端的USB接口，在PC终端识别到 ADB 设备，即可使用 adb shell 调试。

```
>adb.exe devices
List of devices attached
a4b21b7e423198ac    device

>adb.exe shell
root@linaro-alip:/# ls
bin  etc  lost+found  mnt  rockchip_test  sbin  system  var
boot  home  md5sum.txt  opt  root           srv   tmp     vendor
dev  lib  media      proc  run            sys   usr
```

## 3. 串口测试



串口接口位置及引脚定义如上图所示，设备节点列表如下：

序号	功能	设备节点
----	----	------

1	TTL	/dev/ttyS0
2	RS232	/dev/ttyS3
3	RS232	/dev/ttyS4
4	RS485	/dev/ttyS5
5	RS485	/dev/ttyS7

- Debian10 系统串口测试

系统已安装 microcom 工具，可使用此工具给串口节点设置串口参数，例如 /dev/ttyS0 设置波特率为 115200，命令执行后在终端中输入任意字符（输入的字符不会显示），与 /dev/ttyS0 接口相连的上位机软件将会接收到终端输入的字符。

```

1 # microcom -p /dev/ttyS0 -s 115200
2 connected to /dev/ttyS0
3 Escape character: Ctrl-\
4 Type the escape character to get to the prompt.

```

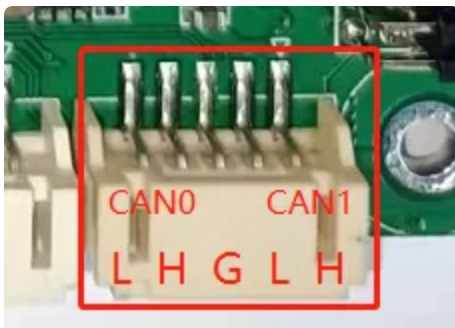
输入 Ctrl-\ 进入 microcom 的命令终端，再输入quit 即可退出microcom 软件。

```

1 Enter command. Try 'help' for a list of builtin commands
2 -> quit
3 exiting

```

## 4. CAN测试



开发板共有两路CAN接口，支持 CANFD 协议，CAN接口测试方法如下：

```
▼ | Bash
1 #关闭can0设备
2 ip link set can0 down
3
4 #设置仲裁段1M波特率, 数据段3M波特率
5 ip link set can0 type can bitrate 1000000 dbitrate 3000000 fd on
6
7 #打印can0信息
8 ip -details link show can0
9
10 #启动can0
11 ip link set can0 up
12
13 #执行candump, 阻塞等待can0接收
14 candump can0
15
16 #canfd格式发送
17 cansend can0 123##1DEADBEEF
18
19 #can格式发送
20 cansend can0 123#1122334455667788
```

## 5. WIFI使用

在使用 WIFI 时需要在开发板的 WIFI 天线座上连接好外接天线。

WIFI 设备节点为: wlan0

```
▼ | Bash
1 # ifconfig wlan0
2 wlan0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
3     ether 90:e8:68:b3:9b:53 txqueuelen 1000 (Ethernet)
4     RX packets 0 bytes 0 (0.0 B)
5     RX errors 0 dropped 0 overruns 0 frame 0
6     TX packets 0 bytes 0 (0.0 B)
7     TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

- Debian10桌面系统WIFI设置

点击桌面右下角的电脑图标, 大弹出的界面中, 选择路由并填写密码上网。

## 6. 蓝牙使用

设备节点: hci0

Debian系统下的蓝牙管理工具开机自动启动，并将蓝牙设备开启，可以点击桌面的右下蓝牙图标，连接蓝牙设备。

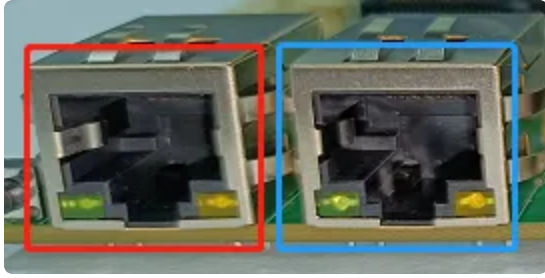
命令行查看蓝牙节点状态如下：

```
Bash |
1 # hciconfig -a
2 hci0:  Type: Primary  Bus: UART
3       BD Address: 67:DF:2D:A3:09:10  ACL MTU: 1021:8  SCO MTU: 64:1
4       UP RUNNING
5       RX bytes:1130 acl:0 sco:0 events:63 errors:0
6       TX bytes:4968 acl:0 sco:0 commands:63 errors:0
7       Features: 0xbf 0xfe 0xcf 0xfe 0xdb 0xff 0x7b 0x87
8       Packet type: DM1 DM3 DM5 DH1 DH3 DH5 HV1 HV2 HV3
9       Link policy: RSWITCH SNIFF
10      Link mode: SLAVE ACCEPT
11      Name: 'linaro-alip'
12      Class: 0x1c0000
13      Service Classes: Rendering, Capturing, Object Transfer
14      Device Class: Miscellaneous,
15      HCI Version: 4.2 (0x8)  Revision: 0x0
16      LMP Version: 4.2 (0x8)  Subversion: 0x2209
17      Manufacturer: Broadcom Corporation (15)
```

扫描周边的BLE蓝牙

```
Bash |
1 # hcitool scan
2 Scanning ...
3       7C:03:AB:A2:DF:BE          n/a
4       94:87:E0:9D:14:12         seeyou
5       B4:20:5B:5A:2F:50         OPPO Reno6 5G
6       04:E5:98:75:D4:D3         Redmi 9
7       7C:2A:DB:01:48:F9         KOOKV
8       D0:97:FE:61:46:0C         GT Neo
9       A4:50:46:9C:DC:23         小米手机8
```

## 7. 以太网使用



开发板两路千兆以太网接口，上图红色框内接口设备节点为 eth0，蓝色框内接口设备节点为 eth1。

以太网接口默认IP获取方式为 dhcp，只需要将以太网接口连接路由器即可为开发板动态分配 IP 地址。

## 7.1 静态IP设置

以eth0设置静态IP地址为例，修改/etc/network/interfaces，在文件中添加如下内容

```
Bash |
1  auto lo
2  iface lo inet loopback
3
4  auto eth0
5  iface eth0 inet static
6  address 192.168.0.234
7  netmask 255.255.255.0
8  gateway 192.168.0.234
9  dns-nameservers 114.114.114.114
```

## 7.2 DNS设置

为开发添加114.114.114.114的DNS，修改/etc/network/interfaces

```
Bash |
1  dns-nameservers 114.114.114.114
```

## 8. 4G使用

开发板带有PCIe座子，可连接 EC20 4G模块。默认系统支持移远拨号管理软件 quectel-CM系统默认适配了 EC20 miniPCIe 4G模块，支持自动拨号上网。使用 4G 功能前，需要先在板载 mini PCIe 接口插入 EC20 模块，在卡槽插入 SIM 卡，并连接好 4G 天线以保证信号的稳定。

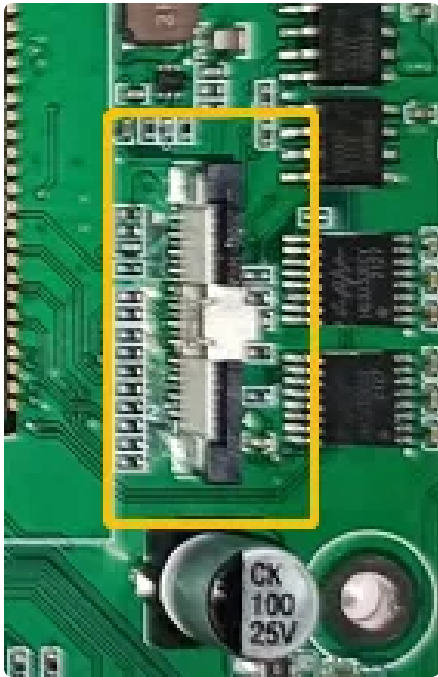


系统已添加 EC20 网络连接管理工具 quectel-CM 用于拨号服务，并自动识别模块插入状态启动服务程序，手动开关服务的方法如下：

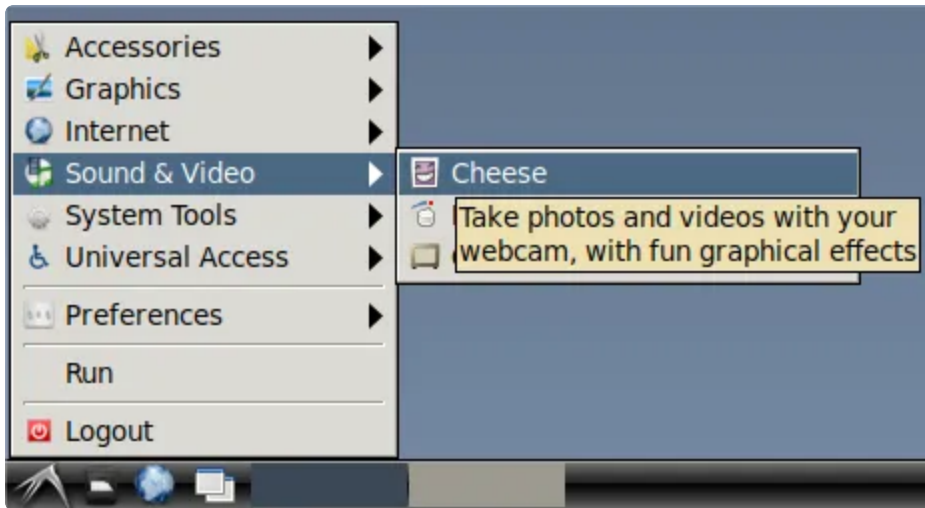
```
▼ Bash |  
1 #停止拨号服务  
2 systemctl stop ec20-dialup.service  
3 #开启拨号服务  
4 systemctl start ec20-dialup.service
```

正常拨号成功后，wwan0将会分配到ip地址，此时可以测试是否能够正常ping通外网。

## 9. 摄像头使用



节点：/dev/video0可用系统自带软件Cheese进行测试，如下图所示：



## 10. U盘\TF挂载



红色框内的接口默认为OTG模式， device和host模式的切换方法如下：

```
1 host:
2 echo host > /sys/devices/platform/fe8a0000.usb2-phy/otg_mode
3 device:
4 echo peripheral > /sys/devices/platform/fe8a0000.usb2-phy/otg_mode
```

- Debian10 系统  
支持FAT32格式U盘/TF卡的自动挂载，默认挂载目录为 /media/linaro/

## 11. 设置开机启动程序

### 11.1 Debian10 系统

- 非界面程序自启动  
在 /etc/rc.local 脚本中添加需要执行的应用程序命令。
- 带界面应用程序自启动  
在 /home/linaro/.config/lxsession/LXDE/autostart 文件中添加需要执行的程序。

## 13. 按键



开发板有一个Recovery按键，在系统正常启动后，可作为普通按键使用

设备节点：/dev/input/event3

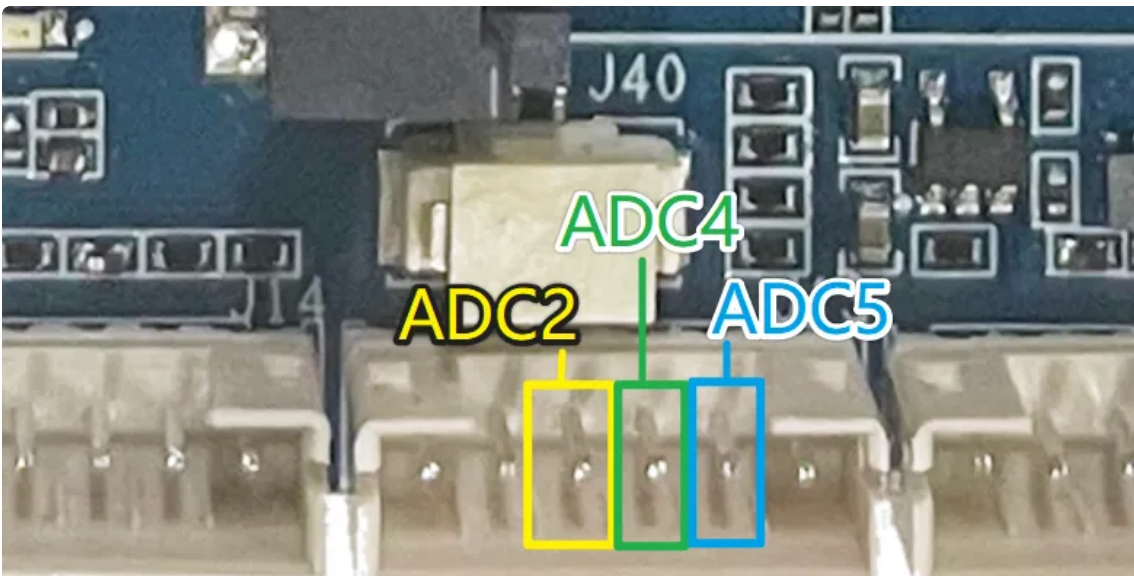
键值：KEY\_VOLUMEUP

在可使用 hexdump 命令确认按键是否有效

```
1 hexdump /dev/input/event3
```

## 13. ADC

开发板引出了3路 ADC 接口，位置如下图所示：



支持10位的SAR-ADC，标准电压位1.8V，ADC接口列表如下所示

接口	设备节点
ADC2	/sys/bus/iio/devices/iio\:device0/in_voltage2_raw
ADC4	/sys/bus/iio/devices/iio\:device0/in_voltage4_raw
ADC5	/sys/bus/iio/devices/iio\:device0/in_voltage5_raw

ADC值读取

```

▼ Bash |
1 cat /sys/bus/iio/devices/iio\:device0/in_voltage2_raw

```

## 14. 时间设置

### 14.1 RTC时间读取和同步

- 系统时间读取和设置

```

▼ Bash |
1 # date
2 Fri Mar 18 12:00:22 CST 2022
3 # date -s "2022-03-18 12:01:00"

```

- rtc时间设置

```

▼ Bash |
1 # hwclock -r
2 2022-03-18 12:01:06.991425+08:00
3 # hwclock -w

```

## 14、RTC

主板包含2个RTC，其中/dev/rtc1为外部RTC（HYM8563），/dev/rtc0为CPU内部的RTC（RK808）。系统默认使用rtc0的时间。所以这里有两种解决方法，如果rtc-hym8563是rtc0，则直接设置即可

```
▼ Bash |
1 root@rk3568-buildroot:/# dmesg | grep rtc
2 [ 1.583028] rk808-rtc rk808-rtc: registered as rtc0
3 [ 1.584797] rk808-rtc rk808-rtc: setting system clock to 2017-08-04T09:0
0:03 UTC (1501837203)
4 [ 1.601112] rtc-hym8563 5-0051: registered as rtc1
5
```

## 方法一

### 14.1 获取RTC时间

```
▼ Bash |
1 root@rk3568-buildroot:/# hwclock
2 Fri Aug 4 09:00:53 2017 0.000000 seconds
```

### 14.2 设置RTC时间

```
▼ Bash |
1 root@rk3568-buildroot:/# date -s '2000-01-30 1:1:1'
2 Sun Jan 30 01:01:01 UTC 2000
3 root@rk3568-buildroot:/# hwclock -w -f /dev/rtc1
4 root@rk3568-buildroot:/# hwclock -r -f /dev/rtc1
5 Sun Jan 30 01:01:11 2000 0.000000 seconds
```

断电重新上电，我们可以看到时间又被复原，我们直接

```
▼ Bash |
1 root@rk3568-buildroot:/# date
2 Fri Aug 4 09:00:17 UTC 2017
3
4 //写入系统时间
5 root@rk3568-buildroot:/# hwclock --hctosys --rtc=/dev/rtc1
6 root@rk3568-buildroot:/# date
7 Sun Jan 30 01:03:58 UTC 2000
8 root@rk3568-buildroot:/# hwclock -r -f /dev/rtc1
9 Sun Jan 30 01:04:17 2000 0.000000 seconds
10
```

## 方法二

如果不想这么麻烦的话，在内核中找到CONFIG\_RTC\_DRV\_RK808把他关掉就行

```
▼ Bash |
1 #CONFIG_RTC_DRV_RK808
```

这时，外部rtc的节点就是外部RTC（HYM8563），也是系统默认使用的rtc，我们常规设置就可以

```
▼ Bash |
1 root@rk3568-buildroot:/# hwclock
2 Fri Aug  4 09:07:26 2017  0.000000 seconds
3
4 //设置系统时间
5 root@rk3568-buildroot:/# date -s '2023-12-19 16:15:20'
6 Tue Dec 19 16:15:20 UTC 2023
7
8 //把系统时间写入rtc
9 root@rk3568-buildroot:/# hwclock -w
10 root@rk3568-buildroot:/# hwclock -r
11 Tue Dec 19 16:15:25 2023  0.000000 seconds
12
13 //断电重启后，直接hwclock就会把rtc时间写入系统
14 root@rk3568-buildroot:/# hwclock
15 Tue Dec 19 16:16:55 2023  0.000000 seconds
16
17
```

## 14.3 时区修改

系统默认时区为 Asia/Shanghai

可以在命令执行 tzselect 命令修改系统时区

## 15. 音视频

### 15.1 播放音频的工具

- aplay 工具

使用 aplay -l 命令可查看当前开发板的声卡信息，如下图所示：

```
▼ | Bash |
1 $ aplay -l
2 List of PLAYBACK Hardware Devices
3 ▼ card 0: rockchiprk809co [rockchip,rk809-codec], device 0: fe410000.i2s-rk81
  7-hifi rk817-hifi-0 [fe410000.i2s-rk817-hifi rk817-hifi-0]
4 Subdevices: 1/1
5 Subdevice #0: subdevice #0
```

播放wav音频文件

```
▼ | Bash |
1 $ aplay -Dhw:0,0 demo.wav
```

- Debian10 系统播放音视频  
界面上可以使用 MPV 播放器播放音视频文件。

## 15.2 音量调节

- Debian10系统音量调节  
点击系统桌面左下角图标，选择 [Sound Video]->[PulseAudio Volume Control]  
在软件界面中调整音量的大小。

-Debian系统播放视频

## 5.10 音频

进入音频设置的图形界面确保以下配置打开

```
▼ | Bash |
1 alsamixer
```

```
AlsaMixer v1.2.7
Card: rockchip,rk809-codec
Chip:
View: F3:[Playback] F4: Capture F5: All
Item: Capture MIC Path [Main Mic]
F1: Help
F2: System information
F6: Select sound card
Esc: Exit

Disabled          SPK_HP          Main Mic          OFF
I2STDM Digital Loo Playback Path < Capture MIC Path > Resume Path
```

播放到HDMI:

```
Bash |
1 aplay -D plug:spk_c0 /usr/share/sounds/alsa/Rear_Center.wav
```

播放到Lineout:

```
Bash |
1 aplay -D plug:spk_c1 /usr/share/sounds/alsa/Rear_Center.wav
```

播放到耳机（需要插入耳机）:

```
Bash |
1 aplay -D plug:spk_c1 /usr/share/sounds/alsa/Rear_Center.wav
```

注意：这里是根据你的声卡选择，如果是接的其他屏幕，如mipi，那么只有一个声卡的情况下，喇叭选择的应该是



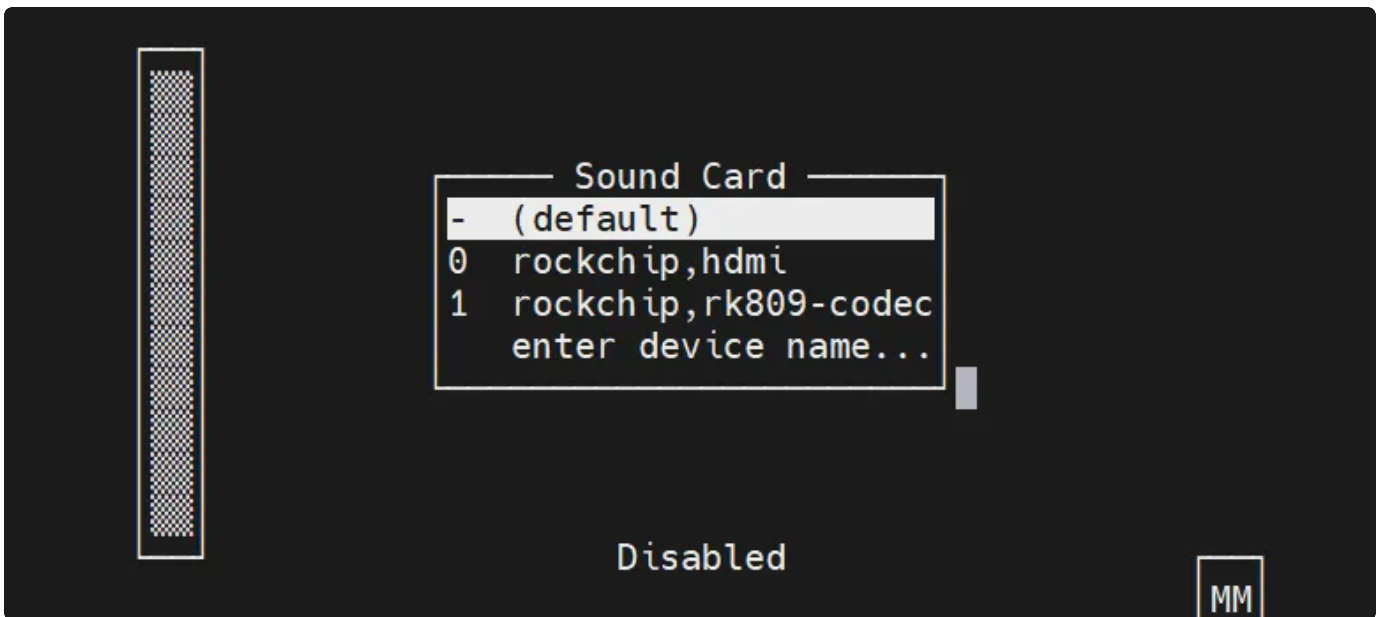
```
1 aplay -D plug:spk_c0 /usr/share/sounds/alsa/Rear_Center.wav
```

音量调节:

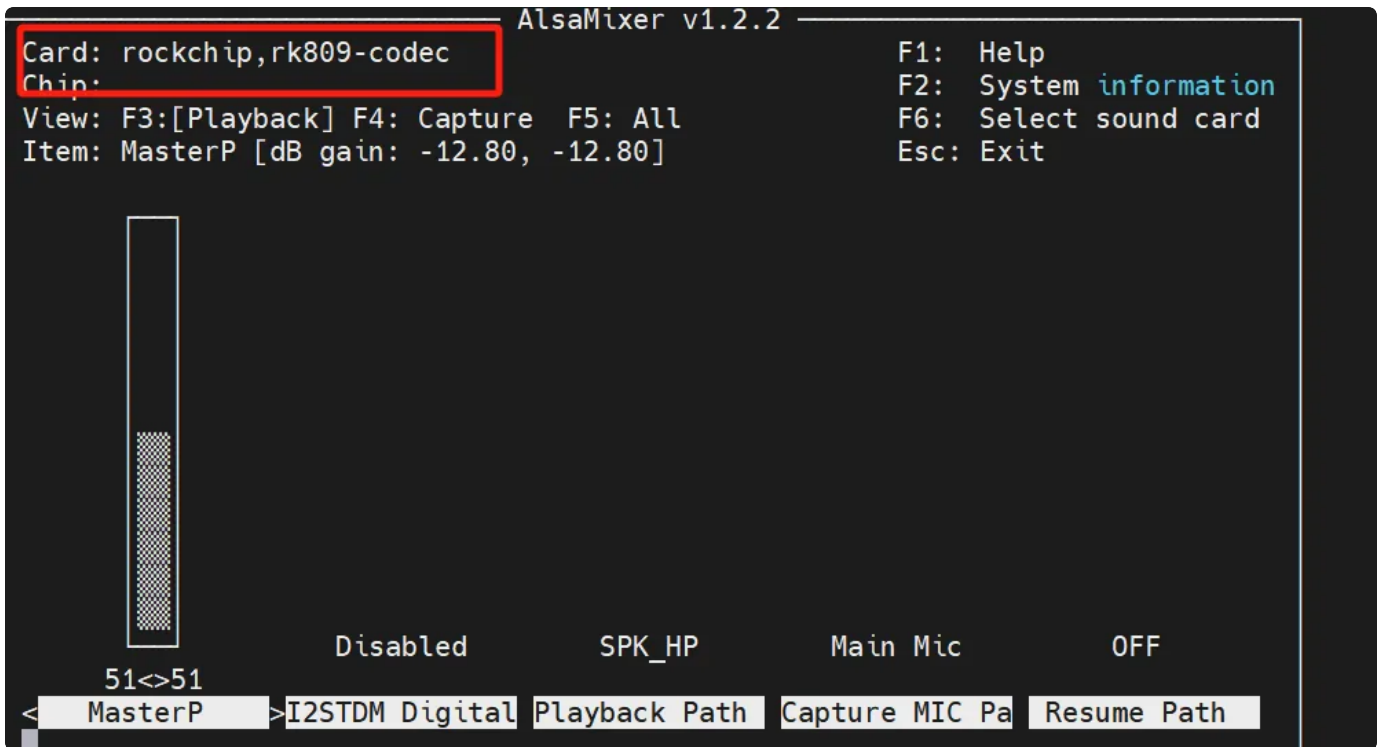
执行alsamixer进入图形界面进行调试

```
1 alsamixer
```

进入图形界面，按s键，选择声卡，如果是喇叭或者耳机则选择为1，如果是hdmi音频则选择为0

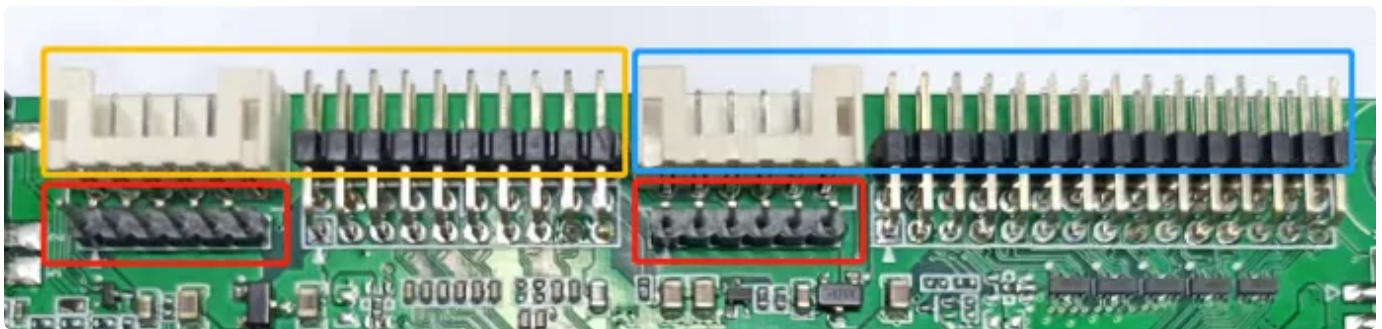


这里可以调节音量为51



## 16. 显示屏

### 16.1 显示屏接口说明



黄色框是eDP屏接口，蓝色框是duallvds屏接口。

红色框分别是两者的供电口，提供12/5/3.3V。



开发板背面，其中红色框是mipi屏接口，蓝色框I2C触摸屏接口。

## 16.2 显示设置

### 16.2.1 屏幕背光亮度设置

- eDP/MIPI屏背光控制

设备节点: /sys/class/backlight/backlight/brightness

设置方法: (支持调节范围 0-255)

```
▼ Bash |  
1 #关闭  
2 echo 0 > /sys/class/backlight/backlight/brightness  
3 #最亮  
4 echo 255 > /sys/class/backlight/backlight/brightness
```

- Dual LVDS屏幕背光控制

设备节点: /sys/class/backlight/backlight1/brightness

设置方法: (支持调节范围 0-255)

```
▼ Bash |  
1 #关闭  
2 echo 0 > /sys/class/backlight/backlight1/brightness  
3 #最亮  
4 echo 255 > /sys/class/backlight/backlight1/brightness
```

## 17. 桌面shell终端

- Debian10系统

点击系统桌面左下角图标, 选择[System Tools]->[LXTerminal]

## 18. 浏览器

- Debian10系统自带Chromium浏览器

点击系统桌面左下角图标, 选择 [Internet]->[Chromium Browser]

## 19. GPU测试

命令行执行 test\_glmark2.sh 脚本测试