



# DA020R

产品使用手册

2025-02-10

四川万物纵横科技股份有限公司

## 目 录

第一章	配件清单 .....	3
第二章	搭建 Linux 开发环境 .....	4
2.1.	开发环境介绍 .....	4
2.2.	安装 Linux 开发工具 (必须).....	4
第三章	编译盒子系统镜像 .....	6
3.1.	SDK 说明 .....	6
3.2.	SDK 编译 .....	7
第四章	烧写盒子系统镜像 .....	9
4.1.	系统升级说明 .....	9
4.2.	调试工具使用 .....	10
第五章	盒子硬件接口使用说明 .....	13
5.1.	HDMI 使用说明 .....	13
5.2.	USB&TF 卡使用说明 .....	13
5.3.	网口使用说明 .....	14
5.4.	RS232 接口使用说明 .....	16
5.5.	RS485 接口使用说明 .....	17
5.6.	RTC 使用说明 .....	18
5.7.	音频使用说明 .....	18
5.8.	继电器&GPIO 使用说明 (IO 默认为 OUT) .....	21
5.9.	PWR&SYS 指示灯使用说明 .....	24
5.10.	RST 按键使用说明 .....	24
5.11.	4G 模块使用说明 .....	24

## 第一章 配件清单

DA020R 默认配件如下：

配件名称	数量	说明
DA020R盒子	1个	
电源适配器	1个	12V/3A
1.5m国标AC线	1个	
接线端子	2个	插拔式
螺丝钉	4个	

## 第二章 搭建 Linux 开发环境

### 2.1. 开发环境介绍

#### 1) 软件开发环境:

本开发包默认在 ubuntu 18.04 64 位系统上编译, 在 32 位 Linux 系统、或较老版本的 Linux 系统、偏冷门的 Linux 系统上可能存在未知的兼容性问题。

- 主机系统及编译器版本: ubuntu 18.04.
- DA020R 盒子网盘资料( 开发工具/环境、软/硬件资料、扩展资料、规格书、用户手册等全部 资料 )

#### 2) 硬件开发环境:

- DA020R 盒子
- Micro USB Type C 数据线
- 推荐主机系统配置: 双核以上处理器、4GB 以上内存、500GB 以上存储空间 或更好配置

#### 3) 其它开发配件推荐 (需客户自备):

- TF 卡: 8GB 以上
- 显示器: AOC /其它型号

**注意: 此文档的操作基于 DA020R 盒子。**

### 2.2. 安装 Linux 开发工具 (必须)

#### 1) 安装前, 请确保 ubuntu 系统能正常连接网络。

#### 2) 执行以下命令配置并安装所需开发工具。 步骤 1: 安装所需开发包

执行 `sudo apt-get install repo git-core gitk git-gui gcc-arm-linux-gnueabi\ubuntu-boot-tools device-tree-compiler gcc-aarch64-linux-gnu mtools parted libudev-dev\libusb-1.0-0-dev python-linaro-image-tools linaro-image-tools autoconf\autotools-dev`

libsigsegv2 m4 intltool libdrm-dev curl sed make binutils\build-essential gcc g++ bash  
patch gzip gawk bzip2 perl tar cpio python unzip rsync\ file bc wget libncurses5 libqt4-  
dev libglib2.0-dev libgtk2.0-dev libglade2-dev cvs\ git mercurial rsync openssh-client  
subversion asciidoc w3m dlatex graphviz\ python-matplotlib libc6:i386 libssl-dev  
expect fakeroot cmake flex bison liblz4-tool\ libtool keychain

## 第三章 编译盒子系统镜像

### 3.1. SDK 说明

#### 1) 解压 SDK

1、 把 SDK 放到虚拟机中 (当前版本: DA020R2-sdk-v2.2.5\_image-v1.4.tar.bz2)

2、 执行下列命令, 解压并同步 SDK;

```
mkdir DA020R2-sdk-v2.2.5
```

```
tar -xvf DA020R2-sdk-v2.2.5_image-v1.4.tar.bz2 -C DA020R2-sdk-v2.2.5 sync
```

#### 2) SDK 简介

经过上面的步骤, 目录上就会出现了很多文件夹, 大概的说明如下:

- **buildroot**: 定制根文件系统。
- **app**: 存放上层应用程序。
- **external**: 相关库, 包括音频、视频等。
- **kernel**: kernel 代码。
- **device/rockchip**: 存放每个平台的一些编译和打包固件的脚本和预备文件。
- **docs**: 存放开发指导文件、平台支持列表、工具使用文档、Linux 开发指南等。
- **prebuilts**: 存放交叉编译工具链。
- **rkbin**: 存放固件和工具。
- **rockdev**: 存放编译输出固件。
- **tools**: 存放一些常用工具。
- **u-boot**: U-Boot 代码。

## 3.2. SDK 编译

(进入解压后的目录后操作下面)

### 1) 全自动编译

执行编译之前先选择对应平台板级配置:

source envsetup.sh (选择 88——rockchip\_rv1126\_DA020R)

./build.sh lunch (选择 16——BoardConfig-DA020R.mk)

./build.sh all (全编译)

update 完整镜像在解压目录的 IMAGE 或 rockdev 路径下

```
Hash value: 90ab4a28e0a008a4dec70215ef1b6a575686c89247065094f3ece58ac35ec12e
Image 2 (rawdisk)
Description: unavailable
Created: Fri May 12 15:11:36 2023
Type: RAWDisk Image
Compression: uncompressed
Data Size: 6973126 Bytes = 6809.69 KiB = 6.65 MiB
Architecture: ARM
OS: Linux
Load Address: 0xffffffff02
Entry Point: unavailable
Hash algo: sha256
Hash value: 87133b6019f463c80eef5da3c32508ab6bfec10367b50619cc83ec90ca532ab8
Image 3 (resource)
Description: unavailable
Created: Fri May 12 15:11:36 2023
Type: Multi-File Image
Compression: uncompressed
Data Size: 141312 Bytes = 138.00 KiB = 0.13 MiB
Hash algo: sha256
Hash value: b4f40610ef22529127eac3b8d70e69c25e7b343de8971befdb4239496c2f356c
Default Configuration: 'conf'
Configuration 0 (conf)
Description: unavailable
Kernel: kernel
Init Ramdisk: rawdisk
FDT: fdt
done.
you take 0:31.65 to build recovery
Running build_recovery succeeded.
Skipping build_ramboot for missing source: RK_CFG_RAMBOOT.
Running build_all succeeded.
```

### 2) u-boot 编译

./build.sh uboot

### 3) kernel 编译

./build.sh kernel

### 4) recovery 编译

./build.sh recovery

### 5) rootfs 编译

```
./build.sh rootfs
```

## 6) 固件打包

上面 U-boot/kernel 等各个部分编译后，进入工程目录根目录执行以下命令  
自动完成所有固件打包到 rockdev 目录下

```
./mkfirmware.sh
```

```
./build.sh updateimg
```

**\*打包好的 update.img 在 rockdev 下，可以直接拷贝到 windows 下进行  
烧写更新系统。**

## 第四章 烧写盒子系统镜像

DA020R 盒子支持 TF 卡方式烧写升级系统镜像。

### 4.1. 系统升级说明

制卡工具位于网盘资料：00 调试工具 /03 做卡烧写工具 /SDDiskTool\_v1.72.zip。

1) 使用 SDDiskTool\_v1.72 工具制作固件升级 SD 烧写卡 (首先 SD 卡要先格式化为 FAT32)

选择“磁盘设备”，勾选“固件升级”，选择“升级固件”，点击“开始创建”；然后开始做卡，直到做卡成功(大概需等五分钟以上)：



2) 做卡完成后,正常卸载磁盘,然后将 SD 卡插入盒子卡槽重新上电进行烧写,等待几分钟烧写之后拔出 TF 卡。

```
13.802444] mmcblk0: p1 p2 p3 p4 p5 p6 p7 p8 p9 p10
33.759314] vcc1v2_dvdd: disabling
33.759388] vcc2v8_avdd: disabling
33.759414] vcc_otg_vbus: disabling
33.759460] vcc_dovdd: disabling
45.320172] mmcblk0: p1 p2 p3 p4 p5 p6 p7 p8 p9 p10
45.350329] GPT:disk_guids don't match.
45.350373] GPT:partition_entry_array_crc32 values don't match: 0x5c208417 != 0xf96c1294
45.350382] GPT: Use GNU Parted to correct GPT errors.
45.350447] mmcblk0: p1 p2 p3 p4 p5 p6 p7 p8 p9 p10
45.449750] GPT:disk_guids don't match.
45.449792] GPT:partition_entry_array_crc32 values don't match: 0x5c208417 != 0xf96c1294
45.449800] GPT: Use GNU Parted to correct GPT errors.
45.449866] mmcblk0: p1 p2 p3 p4 p5 p6 p7 p8 p9 p10
55.255463] GPT:disk_guids don't match.
55.255499] GPT:partition_entry_array_crc32 values don't match: 0x5c208417 != 0xf96c1294
55.255506] GPT: Use GNU Parted to correct GPT errors.
55.255560] mmcblk0: p1 p2 p3 p4 p5 p6 p7 p8 p9 p10
Please remove SD CARD!!!, wait for reboot.
```

注意：看到 **Please remove SD CARD!!!,wait for reboot.**文字代表升级成功，此时拔掉 SD 卡 等待系统重启完成即可，切记要等待出现 remove SD CARD，再拔出 TF 烧写卡。

**提示：使用闪迪卡，如果其他卡类型不能成功做卡，请更换闪迪卡，最大支持 32GB。**

## 4.2. 调试工具使用

1) 硬件先使用 Type-C 数据线接上盒子 DBG 口，连接到 PC 端

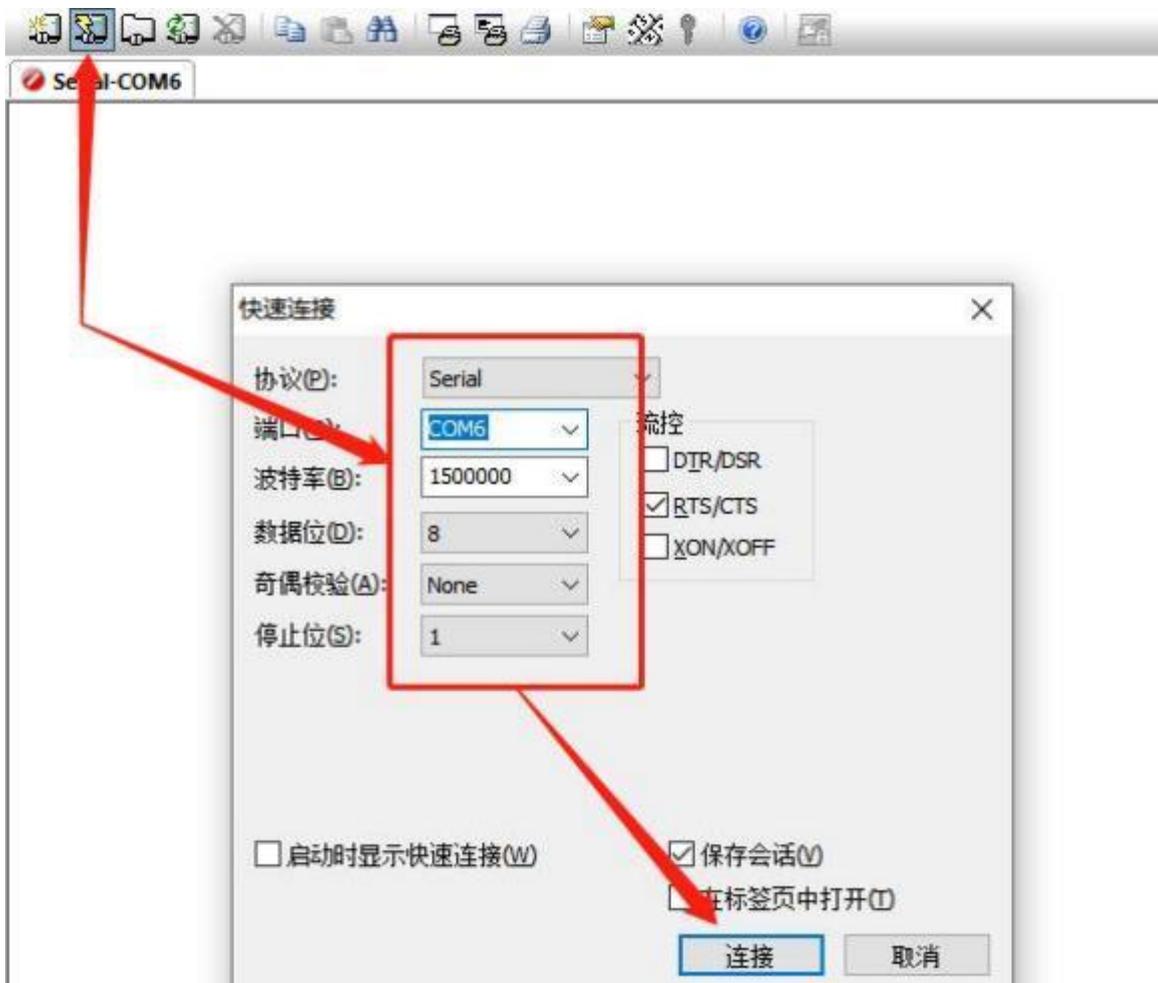


2) SecureCRT 工具 (解压并打开)，使用方法步骤如下：

1、查看设备管理器 (PC 端识别到的 COM 口)



2、查看设备管理器 (PC 端识别到的 COM6 口，串口波特率为 1500000)



3、重新上电，开机正常打印，进入调试终端

```

iCamId": 1 }, { "sMediaPath": "\userdata\photo1", "sThumbPath": "\userdata\photo1\thumb",
lack_list", "sThumbPath": "", "sFormat": "%v%md%H%M%S", "iType": 2, "iCount": 0, "iDuty": 10,
unt": 0, "iDuty": 40, "iUseSize": 0, "iNum": 0, "iCamId": 0 }, { "sMediaPath": "\userdata\whit
us": 1, "iFreeSize": 1024, "iFreeSizeNotice": 512, "iLowFreeStatus": 0 }
db_monitor_get_media_path: type = 2
storage_manager[604]: [manage.c][signal_update_media_path]:storage_manager signal_update_media_
: "VIDEO_%v%md%H%M%S", "iType": 0, "iCount": 0, "iDuty": 0, "iUseSize": 0, "iNum": 0, "iCamId":
: 0, "iDuty": 0, "iUseSize": 0, "iNum": 0, "iCamId": 0 }, { "sMediaPath": "\userdata\video1",
iCamId": 1 }, { "sMediaPath": "\userdata\photo1", "sThumbPath": "\userdata\photo1\thumb",
lack_list", "sThumbPath": "", "sFormat": "%v%md%H%M%S", "iType": 2, "iCount": 0, "iDuty": 10,
unt": 0, "iDuty": 40, "iUseSize": 0, "iNum": 0, "iCamId": 0 }, { "sMediaPath": "\userdata\whit
us": 1, "iFreeSize": 1024, "iFreeSizeNotice": 512, "iLowFreeStatus": 0 }
db_monitor_get_media_path: type = 4
face_config: en 1
    volume 50
    live_det_en 1
    live_det_th 50
    face_det_th 55
    face_rec_th 50
    min_pixel 144
    corner_x 0
    corner_y 0
    det_width 720
    det_height 1280
    nor_width 720
    nor_height 1280
Playing WAVE '/etc/authorize.fail.wav': signed 16 bit Little Endian, Rate 16000 Hz, Stereo
load face feature from /oem/face_data.db
face number is 0
[ 33.761457] vcc1v2_dvdd: disabling
[ 33.761537] vcc2v8_avdd: disabling
[ 33.761565] vcc_otg_vbus: disabling

[root@RV1126_RV1109:~]#
[root@RV1126_RV1109:~]#

```

4、进入系统之后，可以查询到系统版本，串口端输入命令 **brd\_version** (**ema\_version**)，例如结果如图：

```

[root@RV1126_RV1109:~]# brd_version
Chip : RV1126
Hardware_ver : IVP11xx_a2
SDK : rv1126_rv1109_linux_v2.2.5
ID : 051301
Software_Type: buildroot-2018.02-rc3
Software_ver : V1.4
Date : 2022-11-04

```

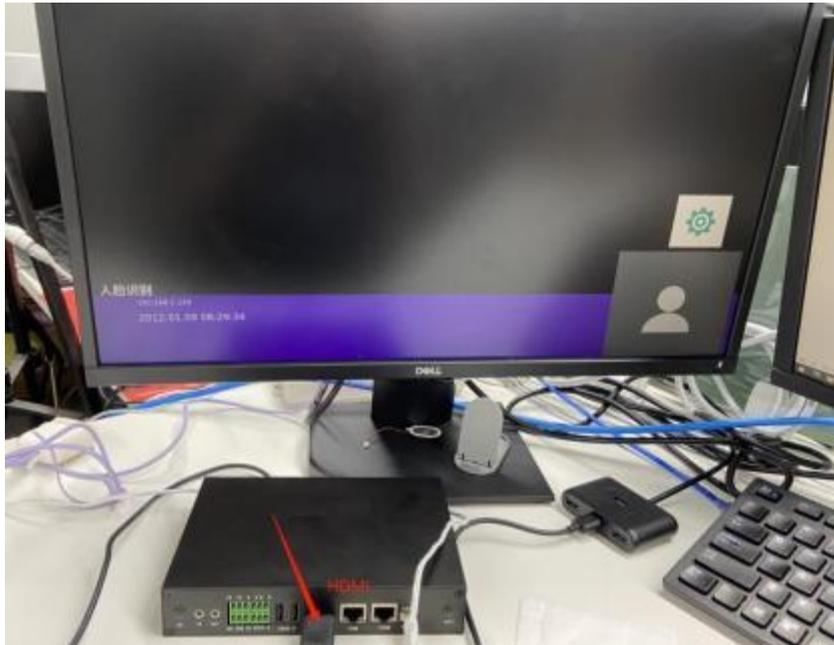
提示：如果进入终端键盘敲击命令无反应，请把设置串口的 **RTS/CTS** 勾选去掉重新进入即可。

## 第五章 盒子硬件接口使用说明

DA020R 盒子接口资源非常多，以下为硬件接口使用说明。

### 5.1. HDMI 使用说明

盒子系统开机自启动 QfacialGate 程序，在 HDMI 显示屏显示界面如下  
(只作为验证 HDMI 显示 功能)：



**注意：**开机显示的自启动程序是官方自带，不作为盒子的功能使用，仅供参考。

### 5.2. USB&TF 卡使用说明

DA020R 盒子上有 2 个 USB2.0 接口，一个 TF 卡槽：

DA020R 接入 U 盘,会自动挂载到/media/usb0 下,但 U 盘需要 FAT32 格式。

如下：

```
[root@RV1126_RV1109:/]# df -hT
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/root       ext4      944M  312M  575M  36% /
devtmpfs       devtmpfs  239M   0    239M   0% /dev
tmpfs          tmpfs     239M   0    239M   0% /dev/shm
tmpfs          tmpfs     239M  372K  239M   1% /tmp
tmpfs          tmpfs     239M  292K  239M   1% /run
/dev/mmcblk0p7  ext2      125M  1.7M  117M   2% /oem
/dev/mmcblk0p8  ext2      6.0G  4.0M  6.0G   1% /userdata
/dev/sda1       vfat      15G   11G  4.1G  73% /media/usb0
```

盒子接入 TF 卡，会自动挂载到/mnt/sdcard 下,如下：

```
[root@RV1126_RV1109:/]# df -hT
Filesystem      Type      Size  Used Avail Use% Mounted on
/dev/root       ext4      944M  312M  575M  36% /
devtmpfs       devtmpfs  239M   0    239M   0% /dev
tmpfs          tmpfs     239M   0    239M   0% /dev/shm
tmpfs          tmpfs     239M  376K  239M   1% /tmp
tmpfs          tmpfs     239M  288K  239M   1% /run
/dev/mmcblk0p7  ext2      125M  1.7M  117M   2% /oem
/dev/mmcblk0p8  ext2      6.0G  4.0M  6.0G   1% /userdata
/dev/mmcblk1p1  vfat      15G   11G  4.1G  73% /mnt/sdcard
```

以上可以通过读写数据测试：

写数据测试：`dd if=/dev/zero of=/media/usb0/test1 bs=4096 count=104857` 读数据测试：`dd if=/media/usb0/test1 of=/dev/null bs=4096 count=104857`

**注意：**测试得出的结果差异与测试的设备也有直接关联

### 5.3. 网口使用说明

1) 使用网线与 DA020R 盒子上的网口连接。



2) 输入 `ifconfig` 指令,可查到 `eth0` 和 `eth1` 的 IP 地址 (`eth0` 的 IP 地址是

默认, eth1 可自动识别) ; 修改为静态 IP (/etc/rc.local) , 直接在文件里修改 IP 地址保存即可, 已存在 eth0 的 IP 地址——192.168.1.150。

```
[root@RV1126_RV1109:/]# cat /etc/rc.local
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.
ifconfig eth0 192.168.1.150 up
ifconfig eth1 10.1.30.134 up

[...]
```

```
[root@RV1126_RV1109:/]# ifconfig
eth0      Link encap:Ethernet  Hwaddr 6A:A7:6D:13:57:78
          inet addr:192.168.1.150  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::68a7:bdff:fe13:5778/64  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3 errors:0 dropped:0 overruns:0 frame:0
          TX packets:32 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:656 (656.0 B)  TX bytes:5662 (5.5 KiB)
          Interrupt:62

eth1      Link encap:Ethernet  Hwaddr 62:A7:2A:7F:72:7A
          inet addr:10.1.30.134  Bcast:10.1.30.255  Mask:255.255.255.0
          inet6 addr: fe80::60a7:2aff:fe7f:727a/64  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:56 errors:0 dropped:0 overruns:0 frame:0
          TX packets:88 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:9848 (9.6 KiB)  TX bytes:14148 (13.8 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128  Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:91 errors:0 dropped:0 overruns:0 frame:0
          TX packets:91 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7560 (7.3 KiB)  TX bytes:7560 (7.3 KiB)
```

### 3) IP 地址设置及 ping 地址

使用 udhcpc 方式重新获取 IP

```
udhcpc -i eth0 up
```

```
[root@RV1126_RV1109:/]# udhcpc -i eth0 up
udhcpc: started, v1.27.2
Failed to kill daemon: No such file or directory
udhcpc: sending discover
udhcpc: sending select for 10.1.30.130
udhcpc: lease of 10.1.30.130 obtained, lease time 86400
Failed to kill daemon: No such file or directory
deleting routers
adding dns 10.1.30.1
```

```
ping -I eth0 www.baidu.com
```

```
[root@RV1126_RV1109:/]# ping -I eth0 www.baidu.com
PING www.baidu.com (14.215.177.38) from 10.1.30.130 eth0: 56(84) bytes of data.
64 bytes from www.baidu.com (14.215.177.38): icmp_seq=1 ttl=56 time=4.13 ms
64 bytes from www.baidu.com (14.215.177.38): icmp_seq=2 ttl=56 time=3.55 ms
64 bytes from www.baidu.com (14.215.177.38): icmp_seq=3 ttl=56 time=3.46 ms
64 bytes from www.baidu.com (14.215.177.38): icmp_seq=4 ttl=56 time=3.69 ms
64 bytes from www.baidu.com (14.215.177.38): icmp_seq=5 ttl=56 time=3.43 ms
64 bytes from www.baidu.com (14.215.177.38): icmp_seq=6 ttl=56 time=3.49 ms
64 bytes from www.baidu.com (14.215.177.38): icmp_seq=7 ttl=56 time=3.80 ms
64 bytes from www.baidu.com (14.215.177.38): icmp_seq=8 ttl=56 time=3.81 ms
64 bytes from www.baidu.com (14.215.177.38): icmp_seq=9 ttl=56 time=3.50 ms
64 bytes from www.baidu.com (14.215.177.38): icmp_seq=10 ttl=56 time=5.01 ms
64 bytes from www.baidu.com (14.215.177.38): icmp_seq=11 ttl=56 time=3.92 ms
64 bytes from www.baidu.com (14.215.177.38): icmp_seq=12 ttl=56 time=3.67 ms
64 bytes from www.baidu.com (14.215.177.38): icmp_seq=13 ttl=56 time=3.54 ms
```

## 5.4. RS232 接口使用说明

- 1) RS232 硬件连接 RX 和 TX 端，使用杜邦线将 RX 与 TX 短接



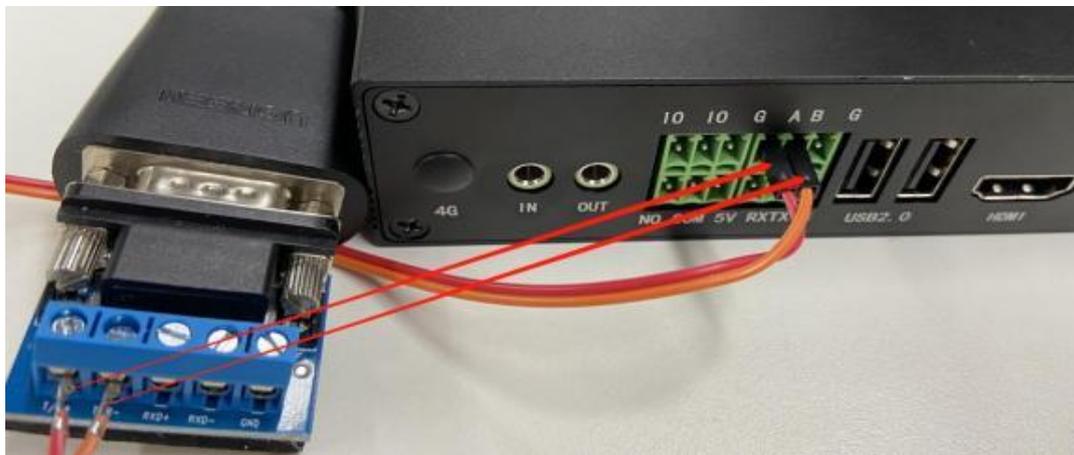
- 2) RS232 对应设备是/dev/ttyS3,默认波特率是 115200, 上传 uart\_test 测试文件, 测试如下:

```
[root@RV1126_RV1109:/usr/local]# ./uart_test -d /dev/ttyS3 -l 10
Read timeout after receiving the first bytes: for 100ms
Read timeout after receiving the first bytes: for 100ms
[ 242.324157] ttys3 set divisor fail, q[uart /dev/ttyS3] setup port unsuccessfully: [baudrate=115200o
databits=8 stopbits=1, parity=ty=0 enablehwtl=0, readTimeout=0 minReadReturnBytes=0]
[uart /dev/ttyS3] open device isuccessfully
[uart /dev/ttyS33] setup port successfully: [baudrate=115200 databits=8 stopbits=1, parity=0 enablehwc
tl=0, readTimeout=1 minReadReturnBytes=252]
= dll,dlh:0
Frame 1: ok
Frame 2: ok
Frame 3: ok
Frame 4: ok
Frame 5: ok
Frame 6: ok
Frame 7: ok
Frame 8: ok
Frame 9: ok
Frame 10: ok
```

uart\_test 程序可以自发自收

## 5.5. RS485 接口使用说明

1) RS485 硬件连接 A 和 B 端,板端 A 与 RS485 转 USB 的 A 相连,板端 B 与 RS485 转 USB 的 B 相连



2) USB 接入 PC 端,打开 PC 端的串口调试工具,选择 RS485 的对应端口使用电脑的 RS485 设备与设备连接并进行收发测试 (板端 RS485 对应 /dev/ttyS1,波特率是 115200)

盒子端设置波特率: `stty -F /dev/ttyS1 ispeed 115200 ospeed 115200 cs8 -parenb -cstopb`

① PC 端发送, 盒子端接收:

盒子端用 microcom 打开对应串口, PC 端在串口工具中发送数据:



② 盒子端发送, PC 端接收:

盒子端用 echo 命令发送数据，PC 端在串口中接收数据：



## 5.6. RTC 使用说明

在串口终端中输入以下命令，查询和设置 date 相关指令：

设置系统时间: `date -s "2022-10-13 14:31:56"`

写入当前时间到 rtc: `hwclock -w`

查看硬件时间: `hwclock`

查看当前系统时间: `date`

```
[root@RV1126_RV1109:/]# date -s "2022-10-13 14:31:55"
Thu Oct 13 14:31:55 +08 2022
[root@RV1126_RV1109:/]# date
Thu Oct 13 14:31:58 +08 2022
[root@RV1126_RV1109:/]# hwclock -w
[root@RV1126_RV1109:/]# hwclock
Thu Oct 13 14:32:03 2022 0.000000 seconds
[root@RV1126_RV1109:/]#
```

## 5.7. 音频使用说明

1) DA020R 盒子具有一组 Line in/out

2) 修改音频文件采样率

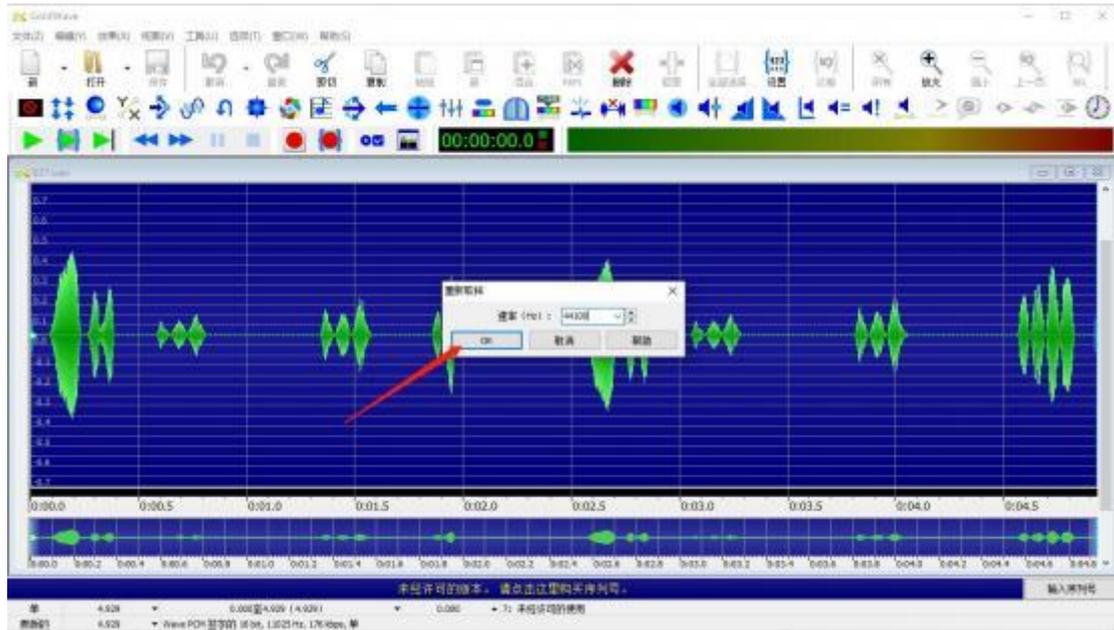
2.1、下载并安装 GoldWave 音频专用软件，然后打开 GoldWave.exe 程序：



2.2、导入音频文件 (文件——打开——选择音频存放路径，软件底部最后一栏可以看到音频一些 基本信息)： (声卡支持采样率为 44100Hz)

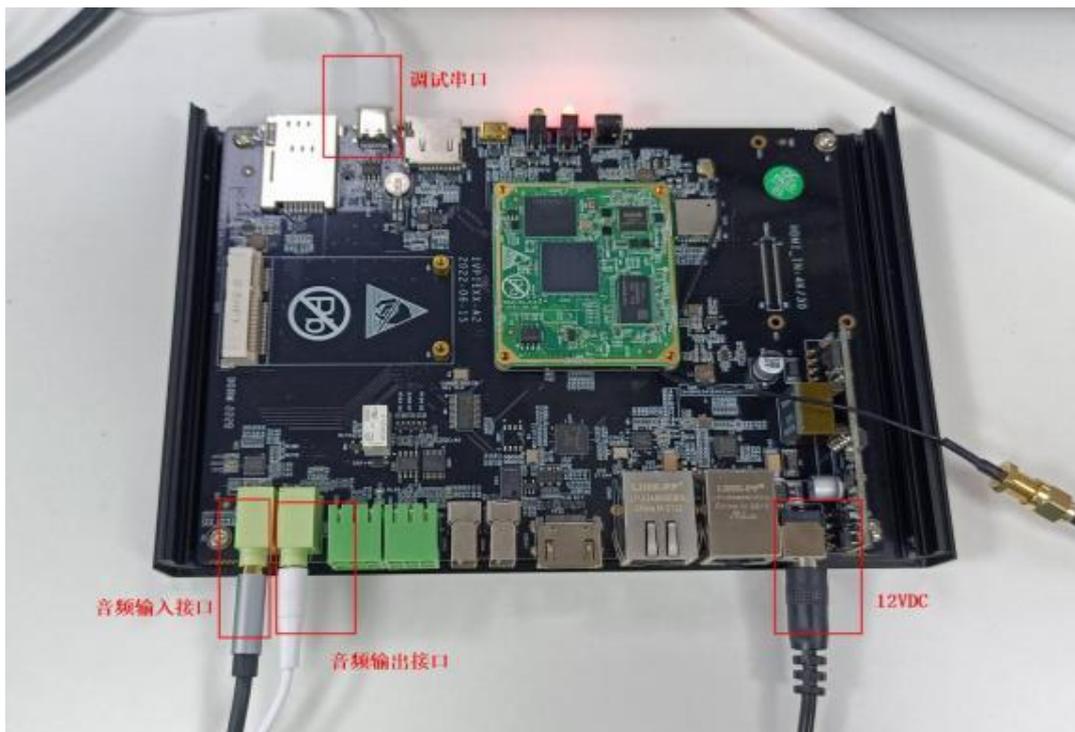


2.3、音频文件重新取样 (效果——重取样——速率改成 44100Hz——OK——保存即可)



### 3) 播放音频文件

3.1、设备接上 DC\_12V 适配器电源、TYPE-C 转 USB 串口线、3.5mm 音频线 (音频线另一端接上 PC 端耳机孔或其他音频输入设备)，音频输出接口需接上耳机或音箱等其他音频输出设备。



### 3.2、设备进入系统终端，进行音频的输入输出测试

第一种：PC 端需播放要录入的音频（随便播放一个音频文件）

盒子端录入音频：arecord -f cd -t wav test.wav

```
[root@RV1126_RV1109:~]# arecord -f cd -t wav test.wav
Recording WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

播放录入的音频：aplay test.wav (耳机可以听到录取的音频声音)

```
[root@RV1126_RV1109:~]# aplay test.wav
Playing WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
[root@RV1126_RV1109:~]# █
```

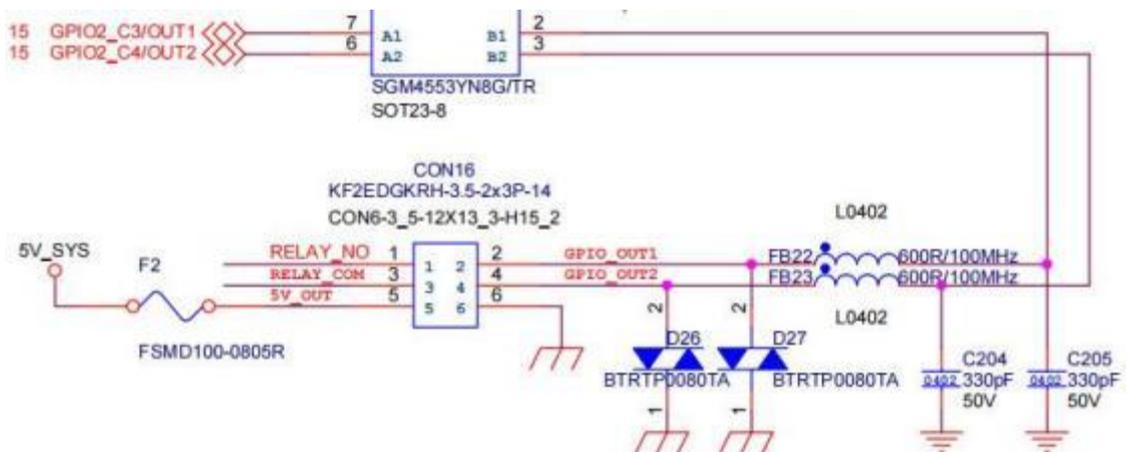
第二种：盒子端上传一个采样率为 44100Hz 的音频文件，直接播放即可，例：

```
[root@RV1126_RV1109:/usr/local]# aplay 827.wav
Playing WAVE '827.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Mono
[root@RV1126_RV1109:/usr/local]# █
```

可听到该音频的声音

### 5.8. 继电器&GPIO 使用说明（IO 默认为 OUT）

1) 控制 led\_relay 的值，使用万用表测量 CON16\_1 和 CON16\_3 通断



\*以上可通过万用表去测量对应的值（0~5V）

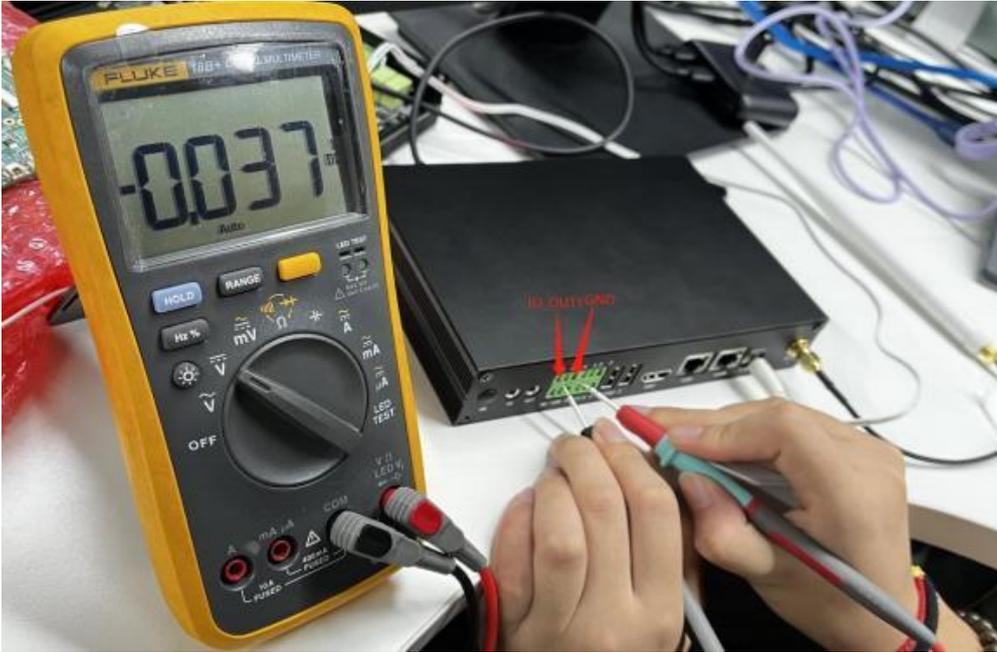
2) CON16\_1 和 CON16\_3 通断测试 (万用表量 NO 和 COM 脚) :

echo 1 >/sys/class/leds/led\_relay/brightness: 通 (有响声)

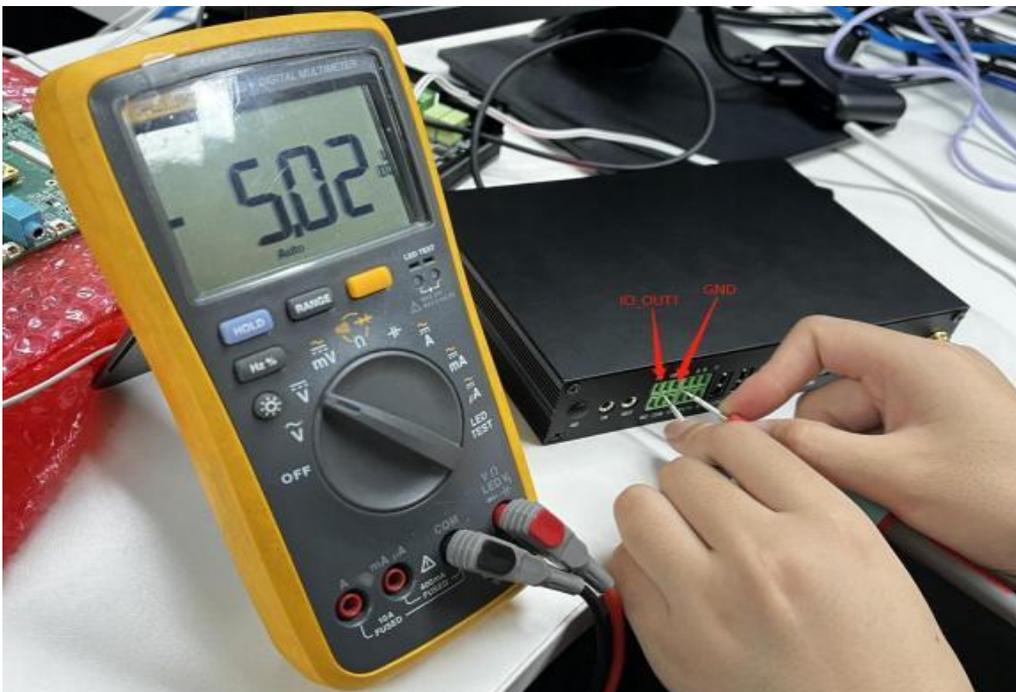
echo 0 >/sys/class/leds/led\_relay/brightness: 断 (无响声)

3) OUT1 测试:

echo 0 >/sys/class/leds/led\_out1/brightness



echo 1 >/sys/class/leds/led\_out1/brightness



4) OUT2 测试:

echo 0 >/sys/class/leds/led\_out2/brightness



echo 1 >/sys/class/leds/led\_out2/brightness



## 5.9. PWR&SYS 指示灯使用说明



1) LED2 作为电源指示灯，通电即可点亮，断电则灭

2) LED3 通过控制系统 led\_g,可控制 led3 的亮灭

```
echo 0 >/sys/class/leds/led_g/brightness:灭
```

```
echo 1 >/sys/class/leds/led_g/brightness:亮
```

## 5.10. RST 按键使用说明

RST: 此按键按下后，系统会重启。

## 5.11. 4G 模块使用说明

**提示：4G 模块是选配，标配里不带有，如有需求，请联系商务，谢谢配合！**

1、首先确认是搭配的 L506 模块还是 NL668 模块

例如用 AT 命令查询模块信息 (以 NL668 为例)：

```
microcom -t 10000 -s 115200 /dev/ttyUSB2
```

回车后敲：ate 然后回车；再敲 ati 回车；再敲 at+cpin? 回车后敲 at+csq  
回车等待退出

```
[root@RV1126_RV1109:/]# microcom -t 10000 -s 115200 /dev/ttyUSB2
ate
OK

Manufacturer: Fibocom Wireless Inc.
Model: NL668-CN
Revision: 19006.1000.00.02.21.11
ESN: +GSN: 0x80353C5D
+GCAP: +CGSM
IMEI: 865804052472155
MEID: A100006F72DC12

OK

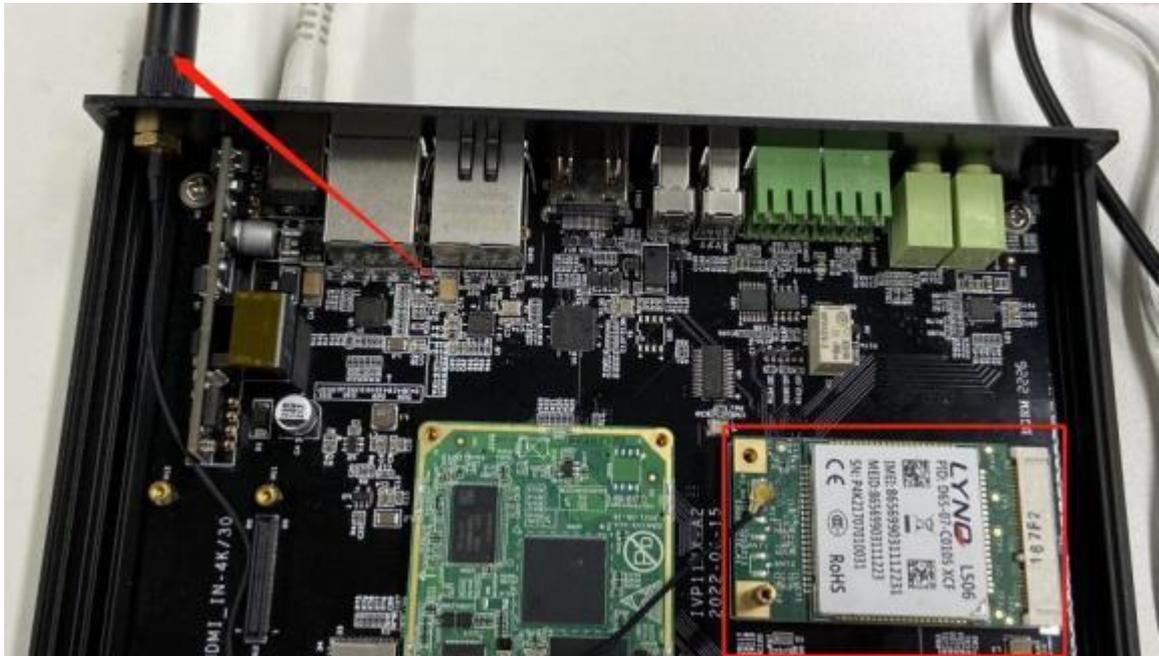
+CPIN: READY

OK

+CSQ: 99,99

OK
[root@RV1126_RV1109:/]#
```

## 2、连接 4G 模块 (L506 模块) 和 4G 天线



1)默认使用 ppp 拨号，上传 ppp 拨号程序到板端任意目录下 (例如：  
/userdata/)

```
[root@RV1126_RV1109:/userdata/ppp]# ls
disconnect gosuncn_options gosuncn_ppp_dialer ppp-on readme.txt
[root@RV1126_RV1109:/userdata/ppp]#
```

2)后台执行 ppp-on，拨号获取 IP 地址

```
[root@RV1126 RV1109:/userdata/ppp]# ./ppp-on &
[root@RV1126 RV1109:/userdata/ppp]#
pppd options in effect:
debug          # (from gosuncn_options)
nodetach       # (from gosuncn_options)
persist        # (from gosuncn_options)
dump           # (from gosuncn_options)
noauth         # (from gosuncn_options)
user Anyname   # (from gosuncn_options)
password ????? # (from gosuncn_options)
/dev/ttyUSB2   # (from gosuncn_options)
115200         # (from gosuncn_options)
lock           # (from gosuncn_options)
connect /usr/sbin/chat -v -f /userdata/ppp/gosuncn_ppp_dialer # (from command line)
crtscts        # (from gosuncn_options)
modem          # (from gosuncn_options)
novj           # (from gosuncn_options)
ipcp-accept-local # (from gosuncn_options)
ipcp-accept-remote # (from gosuncn_options)
noipdefault    # (from gosuncn_options)
defaultroute   # (from gosuncn_options)
usepeerdns     # (from gosuncn_options)
noccps         # (from gosuncn_options)
nobsdcomp      # (from gosuncn_options)
RDY
+CPIN: READY
+CFUN: 1
SMS Ready
Call Ready

OK
ATH
```

### 3)ifconfig 查询 ppp 节点获取到的 IP

```
[root@RV1126 RV1109:/userdata/ppp]# ifconfig
eth0      Link encap:Ethernet  HWaddr 6A:A7:6D:13:57:78
          inet addr:192.168.1.150 Bcast:192.168.1.255 Mask:255.255.255.0
          UP BROADCAST MULTICAST MTU:1500 Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
          Interrupt:62

eth1      Link encap:Ethernet  HWaddr 16:AA:53:05:92:02
          UP BROADCAST MULTICAST MTU:1500 Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:65536 Metric:1
          RX packets:1986 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1986 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:139020 (135.7 KiB) TX bytes:139020 (135.7 KiB)

ppp0      Link encap:Point-to-Point Protocol
          inet addr:10.28.212.154 P-t-P:10.64.64.64 Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
          RX packets:51 errors:0 dropped:0 overruns:0 frame:0
          TX packets:51 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3
          RX bytes:5605 (5.4 KiB) TX bytes:3202 (3.1 KiB)

wlan0     Link encap:Ethernet  HWaddr C4:3C:80:07:CE:58
          UP BROADCAST MULTICAST MTU:1500 Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)
```

### 4)ping 网络: ping -I ppp0 www.baidu.com

```
[root@RV1126_RV1109:/userdata/ppp]# ping -I ppp0 www.baidu.com
PING www.a.shifen.com (14.215.177.38) from 10.28.212.154 ppp0: 56(84) bytes of data.
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=1 ttl=54 time=32.8 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=2 ttl=54 time=70.9 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=3 ttl=54 time=70.4 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=4 ttl=54 time=68.3 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=5 ttl=54 time=166 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=6 ttl=54 time=65.3 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=7 ttl=54 time=83.3 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=8 ttl=54 time=63.1 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=9 ttl=54 time=40.9 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=10 ttl=54 time=59.6 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=11 ttl=54 time=59.5 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=12 ttl=54 time=57.1 ms
```

注意：使用不同运营商的数据卡，需要修改脚本配置

### 3、连接 4G 模块 (NL668 模块) 和 4G 天线



#### 1) 终端输入命令查询模块是否是 18 模式

```
echo -e "ate0\r\n" > /dev/ttyUSB2
```

```
cat /dev/ttyUSB2 &
```

```
echo -e "at+gtusbmode?\r\n" > /dev/ttyUSB2
```

```
[root@RV1126_RV1109:/]# echo -e "ate0\r\n" > /dev/ttyUSB2
[root@RV1126_RV1109:/]# cat /dev/ttyUSB2 &
[root@RV1126_RV1109:/]# echo -e "at+gtusbmode?\r\n" > /dev/ttyUSB2
[root@RV1126_RV1109:/]#
+GTUSBMODE: 18
OK
```

如果查询是 18 模式，直接进行第二步操作；

```
[root@RV1126_RV1109:/]# echo -e "ate0\r\n" > /dev/ttyUSB2
[root@RV1126_RV1109:/]# cat /dev/ttyUSB2 &
[root@RV1126_RV1109:/]# echo -e "at+gtusbmode?\r\n" > /dev/ttyUSB2
[root@RV1126_RV1109:/]#
+GTUSBMODE: 17
OK
```

如果查询是 17 模式，请先用命令设置为 18 模式，然后将模块重新启动，查询是 18 模式之后可进行第二步操作；

echo -e "at+gtusbmode=18\r\n" > /dev/ttyUSB2 (设置模块为 18 模式)

```
[root@RV1126_RV1109:/]# echo -e "at+gtusbmode=18\r\n" > /dev/ttyUSB2
[root@RV1126_RV1109:/]#
OK
设置18模式
```

echo -e "at+cfun=15\r\n" > /dev/ttyUSB2 (重启模块并使新模式生效)

```
[root@RV1126_RV1109:/]# echo -e "at+cfun=15\r\n" > /dev/ttyUSB2
[root@RV1126_RV1109:/]# [ 2302.335525] usb 1-1.4: USB disconnect, device number 5
2302.336428] option1 ttyUSB8: GSM modem (1-port) converter now disconnected from ttyUSB8
2302.336623] option 1-1.4:1.0: device disconnected
2302.339007] option1 ttyUSB1: GSM modem (1-port) converter now disconnected from ttyUSB1
2302.339239] option 1-1.4:1.1: device disconnected
2302.340501] option1 ttyUSB2: GSM modem (1-port) converter now disconnected from ttyUSB2
2302.340639] option 1-1.4:1.2: device disconnected
2302.341756] option1 ttyUSB3: GSM modem (1-port) converter now disconnected from ttyUSB3
2302.341988] option 1-1.4:1.3: device disconnected
2302.342584] qmi_wwan 1-1.4:1.4 wwan0: unregister 'qmi_wwan' usb-ffe80000.usb-1.4, WWAN/QMI device
2313.649668] usb 1-1.4: new high-speed USB device number 6 using ehci-platform
2313.878900] usb 1-1.4: New USB device found, idVendor=1500, idProduct=1001, bcdDevice= 3.18
2313.878959] usb 1-1.4: New USB device strings: Mfr=1, Product=2, SerialNumber=3
2313.878980] usb 1-1.4: Product: Fibocom Modem
2313.879000] usb 1-1.4: Manufacturer: Fibocom Wireless Inc.
2313.879019] usb 1-1.4: SerialNumber: f6db423a
2313.882239] option 1-1.4:1.0: GSM modem (1-port) converter detected
2313.883052] usb 1-1.4: GSM modem (1-port) converter now attached to ttyUSB0
2313.884334] option 1-1.4:1.1: GSM modem (1-port) converter detected
2313.885092] usb 1-1.4: GSM modem (1-port) converter now attached to ttyUSB1
2313.886293] option 1-1.4:1.2: GSM modem (1-port) converter detected
2313.886829] usb 1-1.4: GSM modem (1-port) converter now attached to ttyUSB2
2313.887598] option 1-1.4:1.3: GSM modem (1-port) converter detected
2313.888227] usb 1-1.4: GSM modem (1-port) converter now attached to ttyUSB3
2313.889215] qmi_wwan 1-1.4:1.4: cdc-wdm0: USB WDM device
2313.892440] qmi_wwan 1-1.4:1.4 wwan0: register 'qmi_wwan' at usb-ffe80000.usb-1.4, WWAN/QMI device, be:fa:24:95:81:77
新电重启模块
```

然后查询是否已设置为 18 模式：

```
[root@RV1126_RV1109:/]# echo -e "ate0\r\n" > /dev/ttyUSB2
[root@RV1126_RV1109:/]# cat /dev/ttyUSB2 &
[root@RV1126_RV1109:/]# echo -e "at+gtusbmode?\r\n" > /dev/ttyUSB2
[root@RV1126_RV1109:/]#
+GTUSBMODE: 18
OK
```

**提示：设置 18 模式成功后，会一直处于 18 模式，只要改变模式，必须重启模块**

2) 查看是否拨号成功

查看有无拨号:

echo -e "AT+GTRNDIS?\r\n" > /dev/ttyUSB2 (返回为 0 , 则拨号未成功)

```
[root@RV1126_RV1109:/]# echo -e "AT+GTRNDIS?\r\n" > /dev/ttyUSB2
[root@RV1126_RV1109:/]#
+GTRNDIS: 0
OK
```

手动拨号: echo -e "AT+GTRNDIS=1,1 \r\n" > /dev/ttyUSB2

echo -e "AT+GTRNDIS? \r\n" > /dev/ttyUSB2 1 查看是否拨号成功

```
[root@RV1126_RV1109:/]# echo -e "AT+GTAUTOCONNECT=1 \r\n" > /dev/ttyUSB2
[root@RV1126_RV1109:/]#
OK

[root@RV1126_RV1109:/]# echo -e "AT+GTRNDIS?\r\n" > /dev/ttyUSB2
[root@RV1126_RV1109:/]#
+GTRNDIS: 1,1,"10.9.84.227,240e:47c:a00:222:1d35:9df7:1b45:d48","1.0.0.0,240e:1f:1::1","255.255.255.255,240e:1f:1::33"
OK
```

以上表示拨号成功, 也可自行设置自动拨号:

echo -e "AT+GTAUTOCONNECT=1 \r\n" > /dev/ttyUSB2 (重启系统可自动拨号)

```
[root@RV1126_RV1109:/]# echo -e "AT+GTAUTOCONNECT=1 \r\n" > /dev/ttyUSB2
[root@RV1126_RV1109:/]#
OK 自动拨号
```

3) 不接网线不连 WiFi 的情况下能 ping 通外网

ifconfig 查看 4G 节点, 一般为 usb0

udhcpc -i usb0 获取 4G 模块 IP

ping www.baidu.com -I usb0 若 ping 通则 4G 可以上网, 功能正常, 4G 测试完成

```
[root@RV1126_RV1109:/]# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: can0: <NOARP,ECHO> mtu 16 qdisc noop state DOWN group default qlen 10
    link/can
3: eth0: <BROADCAST,MULTICAST,DYNAMIC,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether fa:0a:78:9c:86:7e brd ff:ff:ff:ff:ff:ff
    inet 10.1.45.2/24 brd 10.1.45.255 scope global eth0
        valid_lft forever preferred_lft forever
4: wlan0: <NO-CARRIER,BROADCAST,MULTICAST,DYNAMIC,UP> mtu 1500 qdisc mq state DOWN group default qlen 1000
    link/ether 0c:cf:89:4a:61:ac brd ff:ff:ff:ff:ff:ff
5: p2p0: <NO-CARRIER,BROADCAST,MULTICAST,DYNAMIC,UP> mtu 1500 qdisc mq state DOWN group default qlen 1000
    link/ether 0e:cf:89:4a:61:ac brd ff:ff:ff:ff:ff:ff
6: usb0: <BROADCAST,MULTICAST,DYNAMIC,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 76:72:8f:f4:9f:a2 brd ff:ff:ff:ff:ff:ff
    inet 192.168.225.2/24 brd 192.168.225.255 scope global usb0
        valid_lft forever preferred_lft forever
[root@RV1126_RV1109:/]# udhcpc -i usb0
udhcpc: started, v1.27.2
Failed to kill daemon: No such file or directory
udhcpc: sending discover
udhcpc: sending select for 192.168.225.2
udhcpc: lease of 192.168.225.2 obtained, lease time 43200
Failed to kill daemon: No such file or directory
deleting routers
adding dns 192.168.225.1
[root@RV1126_RV1109:/]# ping www.baidu.com -I usb0
PING www.baidu.com (14.215.177.38) from 192.168.225.2 usb0: 56(84) bytes of data:
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=1 ttl=53 time=166 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=2 ttl=53 time=79.4 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=3 ttl=53 time=74.9 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=4 ttl=53 time=71.1 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=5 ttl=53 time=43.6 ms
^C
--- www.baidu.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4003ms
rtt min/avg/max/mdev = 43.697/87.171/166.547/41.604 ms
[root@RV1126_RV1109:/]#
```