ATK-DLT113IS

Quick test manual

V1.0

ALIENTEK

1. Shopping:

TMALL: https://zhengdianyuanzi.tmall.com

TAOBAO: https://openedv.taobao.com

2. Download

Address: http://www.openedv.com/docs/index.html

3. FAE

Website : www.alientek.com

Forum : http://www.openedv.com/forum.php

Videos : www.yuanzige.com

Fax : +86 - 20 - 36773971

Phone : +86 - 20 - 38271790

Disclaimer

The product specifications and instructions mentioned in this document are for reference only and subject to update without prior notice; Unless otherwise agreed, this document is intended as a product guide only, and none of the representations made herein constitutes a warranty of any kind. The copyright of this document belongs to Guangzhou Xingyi Electronic Technology Co., LTD. Without the written permission of the company, any unit or individual shall not be used for profit-making purposes in any way of dissemination.

In order to get the latest version of product information, please regularly visit the download center or contact the customer service of Taobao ALIENTEK flagship store. Thank you for your tolerance and support.

Revision History:

Version	Version Update Notes	Responsible person	Proofreading	Date
V1.0	release officially	ALIENTEK	ALIENTEK	2023.03.01

Catalogue

Chapter 1. Introduces ATK-DLT113IS hardware resources 1
Chapter 2. Preparation of ATK-DLT113IS development board before use 2
2.1 Install the driver and serial debugging terminal software 2
2.1.1 Install the CH340 driver....2
2.1.2 How to use serial port to debug terminal software....2
Chapter 3. ATK-DLT113IS burning system 6
3.1 Burn the system to EMMC/NAND....6
3.1.1 Use of the PhoenixSuit software 6
3.1.2 Connect the development board....8
3.1.3 One-button brushing function 8
3.1.4 Complete erase upgrade 11
3.2 Burn system to TF card....15
3.2.1 Use of the PhoenixCard software....15
3.2.2 Burn the system to the TF card 16
3.2.3 The TF card Launching tests....19
3.3 Make batch burn cards 20
3.3.1 Use of the PhoenixCard software....20
3.3.2 Make a burn card....20
3.3.3 Update the development board system using the burning card 23
Chapter 4. Functional test 25
4.1 LED Testing....25
4.2 KEY test....25
4.3 LCD Testing....26
4.3.1 RGB interface test....26
4.3.2 LVDS interface testing....27
4.3.3 LVDS interface testing....28
4.3.4 Touch test....29
4.3.5 Backlight test....29
4.4 Serial port test 30
4.5 USB Testing....31
4.5.1 USB HOST Test....32
4.5.2 USB OTG testing 32
4.6 Network testing 34
4.7 CAN Test 35
4.8 RTC clock testing....37
4.9 Audio Testing....38
4.9.1 Playing the test 38
4.9.2 Recording test 38
4.10 Onboard wifi test....39
4.10.1 Station mode 39
4.10.2 Hotspot (SoftAP) mode....41

4.11 Onboard Bluetooth test 43
4.11.1 Bluetooth pairing....44
4.11.2 Bluetooth rfcomm 46
4.11.3 Bluetooth communication test....46
4.12 4G module EC20 test 47
4.12.1 ppp dial-up Internet access....50
4.12.2 GPS test....51
4.13 4G module ME3630-W test 52
4.13.1 pppd dial-up Internet access....54
4.13.2 ECM access to the Internet 55
4.14 TF(SD) card test....56
4.15 GPADC testing 57
4.16 CVBS IN interface test....58
4.17 CPU temperature....60
4.18 CPU frequency....60
4.19 QT interface 61
4.19.1 QT is briefly closed....61
4.19.2 QT is permanently closed....61
4.20 Problems with the serial port....61

Chapter 1. Introduces ATK-DLT113IS hardware resources

The development board adopts the design of combining the bottom board and the core board, and uses the form of stamp holes for connection. The core board has a small footprint and excellent stability. The development board is powered by a 12V/1A power adapter. In terms of realistic interfaces, the development board has three screen interfaces, namely RGB, LVDS and MIPI DSI, and at the same time elicits three audio input interfaces and two audio output interfaces. In addition, a gigabit network port, two 485 and two CAN interfaces are also onboard, which is very suitable for use in industrial scenarios.

The bottom board resource diagram of the ATK-DLT113IS development board is shown in the figure:

Figure 2.1-1 ATK-DLT113IS development board resource diagram

Chapter 2. Preparation of ATK-DLT113IS development board before use

A brief introduction to this section is as follows:

- Install the serial debugging driver of the development board, and how to use the serial port to debug the host computer.
- Log in to the development board.

2.1 Install the driver and serial debugging terminal software

2.1.1 Install the CH340 driver

The serial port of the development board needs to install the CH340 driver to communicate with the computer. T113-i development board development board CD A disk - Basic information→3_softwares→CH340 driver (USB serial port driver)_XP_WIN7 to find the SETUP.EXE and double-click to install. A successful pre-installation or a successful installation is successful. If the installation of Win10 fails, please download the driver wizard, first use the USB Type-C data cable provided in the delivery to connect the USB_TTL of the development board bottom board, connect the USB interface end to the computer, the driver wizard scans the hardware driver, and install the corresponding serial port driver.

Figure 2.1.1 Double-click SETUP.EXE to install

Figure 2.1.2 Click Start to install the CH340 driver

2.1.2 How to use serial port to debug terminal software

There are many terminals that can use serial ports such as Xshell, SecureCRT, MobaXterm, and even Ubuntu terminals that can connect to the serial port on the development board. In this paper, MobaXterm installation as an example, in many terminal software is relatively friendly to the novice, because the interface is good-looking, but there are also shortcomings, can not automatically reconnect the serial port.

T113-i development board→Development board CD A disk - Basic Information→3_softwares MobaXterm_Installer_v12.3.zip Double click to unzip and install. The installation process is very simple, the next step is the default installation. We won't be writing the installation here.

Install on desktop open MobaXterm icon, software interface as shown below.

Figure 2.1-1 MobaXterm software interface

Connect the hair board with USB Typec data cable to USB_TTL, connect to the computer, plug in the power supply of the development board, and dial the code switch has been set when shipping. Configure MobaXterm as follows: Click Session→Select Serial type→Select com port corresponding to the development board device→Select Baud rate of 115200 (the default baud rate of the development board is 115200) →Advanced Settings → Set Flow Control as None→Click OK.

Figure 2.1-2 MobaXterm operation steps

When the development board is powered on, the information printed by the serial port terminal is as follows.

```
[212]HELLO! BOOT0 is starting!
[215]BOOT0 commit : 069ed30b88
[218]set pll start
[220]fix vccio detect value:0xc0
[227]periph0 has been enabled
[230]set pll end
[232][pmu]: bus read error
[234]board init ok
[236]enable_jtag
[238]get_pmu_exist() = -1
[241]ddr_efuse_type: 0x0
[243]trefi:7.8ms
[245][AUTO DEBUG] single rank and full DQ!
[250]ddr_efuse_type: 0x0
[252]trefi:7.8ms
[254][AUTO DEBUG] rank 0 row = 14
[258][AUTO DEBUG] rank 0 bank = 8
[261][AUTO DEBUG] rank 0 page size = 2 KB
[265]DRAM BOOT DRIVE INFO: V0.34
[268]DRAM CLK = 792 MHz
[270]DRAM Type = 3 (2:DDR2,3:DDR3)
[274]DRAMC ZQ value: 0x7b7bfb
[276]DRAM ODT value: 0x42.
[279]ddr_efuse_type: 0x0
[282]DRAM SIZE = 256 MB
[289]DRAM simple test OK.
[291]rtc standby flag is 0x0, super standby flag is 0x0
[297]dram size =256
[299]card no is 0
[301]sdcard 0 line count 4
[304][mmc]: mmc driver ver 2021-05-21 14:47
[313][mmc]: Wrong media type 0x0
[316][mmc]: ***Try SD card 0***
[324][mmc]: HSSDR52/SDR25 4 bit
[327][mmc]: 50000000 Hz
[329][mmc]: 30250 MB
[331][mmc]: ***SD/MMC 0 init OK!!!***
[429]Loading boot-pkg Succeed(index=0).
[432]Entry_name        = u-boot
[440]Entry_name        = optee
[444]Entry_name        = dtb
[447]mmc not para
```

Figure 2.1-3 Start printing information

After normal startup, you will see the terminal print the following message again:

```
su: unknown user postgres
done
Starting postgresql: su: unknown user postgres
OK
/etc/init.d/S50powerkey_display: line 12: powerkey_display: command not found
/etc/init.d/S50powerkey_suspend: line 12: powerkey_suspend: command not found
[   10.488723] random: crng init done
ssh-keygen: generating new host keys: RSA DSA ECDSA ED25519
Starting sshd: Privilege separation user sshd does not exist
OK
[   14.558373] [EXFAT] trying to mount...
formating  /dev/by-name/UDISK to vfat...
mkfs.fat 4.1 (2017-01-24)
Starting dnsmasq: OK
Trying to connect to SWUpdate...
swu_param: ####
swu_software: ####
swu_mode: ####
no swupdate_cmd to run, wait for next swupdate
root@ATK-DLT113IS:/$
```

Figure 2.1-4 Startup completion information

Chapter 3. ATK-DLT113IS burning system

A brief introduction to this section is as follows:

- Burn the system to EMMC/NAND using PhoenixSuit software
- Using PhoenixCard software, burn system to TF card

3.1 Burn the system to EMMC/NAND

3.1.1 Use of the PhoenixSuit software

Allwinner offers a PhoenixSuit desktop burning tool, which runs on Windows and needs to be installed before running. The path to the file is: Development board CD disk A - Basic Information→5_tools→3_T113-i System Burn Tools→PhoenixSuit V1.19.zip, extract the file. Open the unzipped file. Double-click to open PhoenixSuit.exe. Here the author demonstrates the steps to burn a factory system. For more information about PhoenixSuit, see the PhoenixSuit_ User Guide in the PhoenixSuit sibling directory.

Figure 3.1-1 PhoenixSuit

The schematic diagram of the PhoenixSuit software is as follows:

Figure 3.1-2 PhoenixSuit interface

To install the otg driver, go to PhoenixSuit V1.19\Drivers\AW_Driver and double-click InstallUSBDrv.exe to install the otg driver.

Figure 3.1-3 InstallUSBDrv

After the drivers were installed, the PhoenixSuit software could recognize the development board when it was connected to the computer.

3.1.2 Connect the development board

Before burning, it is necessary to connect the USB_TTL and USB_OTG interfaces of the development board, as shown in the following figure:

Figure 3.1-4 The development board is connected to the OTG interface

USB_TTL: SERIAL PORT printing terminal, can see the serial port information when burning, burning can also be not connected, here the author will connect it.

USB_OTG: Burn interface, you can connect to the system burn.

3.1.3 One-button brushing function

After the software has successfully connected to the device, the user can perform the burning operation. Note here that after the OTG interface of the development board is connected to the computer, the burning tool does not show a successful connection. Here, the software home page prompts "no device connection!!" You can ignore it.

Click "One-click brush" to enter the brush interface, as shown below:

Figure 3.1-5 PhoenixSuit steps

Click the "Browse" button and select the firmware file with the suffix "*.img" in the pop-up window. The system image is in the development board CD A disk - basic data \8_system_image \EMMC path, as shown in the following figure: (The development board has two versions of EMMC/NAND, and supports a variety of screen interfaces RGB, LVDS, MIPI DSI, so we can see here that we provide a variety of images for convenient burning test, here we demonstrate burning EMMC version of the image, If you need to burn NAND version, go to the development board CD A disk - Basic information \8_system_image \NAND and select the corresponding image.

After selecting the firmware, select single or multi-partition download, select all partitions. In the future, if you only need to update the contents of a partition, you can check the corresponding partition separately.

Figure 3.1-6 Select burn image

Don't press the "Upgrade Now" button after you've checked it! Next, hold down the UPDATE button of the development board, and then press the RESET button to burn.

Figure 3.1-7 Development board button

The burning process is shown in the following figure:

Figure 3.1-8 PhoenixSuit burning

Just wait for the burning to complete. Restart the development board after successful burning, and the serial terminal can see the system successfully started and print information:

```
[    7.930582] RTW: _netdev_vir_if_open(wlan1) (bup=1) exit
[    7.936536] RTW: -871x_drv - drv_open, bup=1
[    7.941518] RTW: cfg80211_rtw_set_power_mgmt(wlan0) enabled:1, timeout:-1
[    7.949169] RTW: rtw_ndev_notifier_call(wlan0) state:1
[    7.955194] RTW: rtw_ndev_notifier_call(wlan1) state:14
OK
[    7.961133] RTW: _netdev_vir_if_open(wlan1) , bup=1
[    7.966984] RTW: _netdev_vir_if_open(wlan1) (bup=1) exit
[    7.973208] RTW: cfg80211_rtw_set_power_mgmt(wlan1) enabled:1, timeout:-1
[    7.973223] RTW: rtw_ndev_notifier_call(wlan1) state:1
Initializing postgresql data base...
su: unknown user postgres
done
Starting postgresql: su: unknown user postgres
OK
/etc/init.d/S50powerkey_display: line 12: powerkey_display: command not found
/etc/init.d/S50powerkey_suspend: line 12: powerkey_suspend: command not found
[   10.467544] random: crng init done
Starting sshd: Privilege separation user sshd does not exist
OK
[   10.656697] FAT-fs (mmcblk0p7): Volume was not properly unmounted. Some data may be corrupt. Please run fsck.
/dev/by-name/UDISK already format
[   10.719603] FAT-fs (mmcblk0p7): Volume was not properly unmounted. Some data may be corrupt. Please run fsck.
Starting dnsmasq: OK
Trying to connect to SWUpdate...
swu_param: ####
swu_software: ####
swu_mode: ####
no swupdate_cmd to run, wait for next swupdate
root@ATK-DLT113IS:/$
```

Figure 3.1-9 The development board starts and finishes printing information

3.1.4 Complete erase upgrade

Here is another way to burn, a full disk erase upgrade, in which you can skip the steps in Section 3.1.3.

After the software has successfully connected to the device, the user can perform the burning operation. Note here that after the OTG interface of the development board is connected to the computer, the burning tool does not show a successful connection. Here, the software home page prompts "no device connection!!" You can ignore it.

Click "One-click brush" to enter the brush interface, as shown below:

Figure 3.1-10 PhoenixSuit steps

Click the "Browse" button and select the firmware file with the suffix "*.img" in the pop-up window. The system image is in the development board CD A disk - basic data \8_system_image \EMMC path, as shown in the following figure: (The development board has two versions of EMMC/NAND, and supports a variety of screen interfaces RGB, LVDS, MIPI DSI, so we can see here that we provide a variety of images for convenient burning test, here we demonstrate burning EMMC version of the image, If you need to burn NAND version, go to the development board CD A disk - Basic information \8_system_image \NAND and select the corresponding image.

After selecting the firmware, select the full erase upgrade.

Figure 3.1-11 Mirror selection

Don't press the "Upgrade Now" button when you're done! Next, hold down the UPDATE button of the development board, and then press the RESET button to burn.

Figure 3.1-12 Development board button

The burning process is shown in the following figure:

Figure 3.1-13 PhoenixSuit burning

Just wait for the burning to complete. Restart the development board after successful burning, and the serial terminal can see the system successfully started and print information:

```
[    7.930582] RTW: _netdev_vir_if_open(wlan1) (bup=1) exit
[    7.936536] RTW: -871x_drv - drv_open, bup=1
[    7.941518] RTW: cfg80211_rtw_set_power_mgmt(wlan0) enabled:1, timeout:-1
[    7.949169] RTW: rtw_ndev_notifier_call(wlan0) state:1
[    7.955194] RTW: rtw_ndev_notifier_call(wlan1) state:14
OK
[    7.961133] RTW: _netdev_vir_if_open(wlan1) , bup=1
[    7.966984] RTW: _netdev_vir_if_open(wlan1) (bup=1) exit
[    7.973208] RTW: cfg80211_rtw_set_power_mgmt(wlan1) enabled:1, timeout:-1
[    7.973223] RTW: rtw_ndev_notifier_call(wlan1) state:1
Initializing postgresql data base...
su: unknown user postgres
done
Starting postgresql: su: unknown user postgres
OK
/etc/init.d/S50powerkey_display: line 12: powerkey_display: command not found
/etc/init.d/S50powerkey_suspend: line 12: powerkey_suspend: command not found
[   10.467544] random: crng init done
Starting sshd: Privilege separation user sshd does not exist
OK
[   10.656697] FAT-fs (mmcblk0p7): Volume was not properly unmounted. Some data may be corrupt. Please run fsck.
/dev/by-name/UDISK already format
[   10.719603] FAT-fs (mmcblk0p7): Volume was not properly unmounted. Some data may be corrupt. Please run fsck.
Starting dnsmasq: OK
Trying to connect to SWUpdate...
swu_param: ####
swu_software: ####
swu_mode: ####
no swupdate_cmd to run, wait for next swupdate
root@ATK-DLT113IS:/$
```

Figure 3.1-14 Start to finish printing information

3.2 Burn system to TF card

3.2.1 Use of the PhoenixCard software

Allwinner offers a PhoenixCard host burning tool that runs on Windows and needs to be installed before running. The file path is: development board CD disk A - Basic Data \5_tools \ 113-i System Burning Tool \PhoenixCard V4.3.1.zip, extract the file. Open the unzipped file. Double-click to open PhoenixCard.exe. Here the author demonstrates the steps to burn a factory system. For more information about PhoenixCard, see the PhoenixCard_ User Guide in the PhoenixCard sibling directory.

st › PhoenixCard V4.3.1

名称	修改日期	类型	大小
ImageOps.dll	2021/2/5 11:16	应用程序扩展	241 KB
ImgDecode32.dll	2021/2/5 11:16	应用程序扩展	52 KB
ImgDecode64.dll	2021/2/5 11:16	应用程序扩展	233 KB
ini_fun.lhs	2021/2/5 11:16	LHS 文件	4 KB
IniParasPlg.dll	2021/2/5 11:16	应用程序扩展	44 KB
Langplg.dll	2021/2/5 11:16	应用程序扩展	56 KB
LangPlgex.dll	2021/2/5 11:16	应用程序扩展	24 KB
LangPlgex_1.dll	2021/2/5 11:16	应用程序扩展	24 KB
log.ini	2023/11/6 13:40	配置设置	3 KB
loghelper.dll	2023/11/1 13:46	应用程序扩展	127 KB
lua5.1.dll	2021/2/5 11:16	应用程序扩展	617 KB
luaBase.dll	2021/2/5 11:16	应用程序扩展	136 KB
luasocket.dll	2021/2/5 11:16	应用程序扩展	24 KB
Mbr2Gpt.dll	2023/9/4 9:31	应用程序扩展	11 KB
option.cfg	2023/9/7 13:59	Configuration 源...	1 KB
ParserManager.dll	2023/8/23 10:35	应用程序扩展	81 KB
PhoenixCard.exe	2023/11/6 15:35	应用程序	1,802 KB
PhoenixCard.lan	2023/9/7 17:24	LAN 文件	8 KB
PhoenixCard_ChangeLog.pdf	2023/11/6 16:07	WPS PDF 文档	80 KB
PhoenixCard_User_Guide.pdf	2021/4/25 8:58	WPS PDF 文档	515 KB
PhoenixCard_使用指南.pdf	2023/10/25 19:27	WPS PDF 文档	755 KB
PhoenixCardFun.dll	2023/11/6 15:25	应用程序扩展	301 KB
PlgVector.dll	2021/2/5 11:16	应用程序扩展	70 KB
regbasefun.lhs	2021/2/5 11:16	LHS 文件	1 KB
regsocketfun.lhs	2021/2/5 11:16	LHS 文件	1 KB
reguifun.lhs	2021/2/5 11:16	LHS 文件	1 KB
thisdata.fex	2021/2/5 11:16	FEX 文件	14,799 KB
updatelist.cfg	2021/2/5 11:16	Configuration 源...	2 KB

Figure 3.2-1 PhoenixCard software

PhoenixCard software diagram is as follows:

Figure 3.2-2 PhoenixCard interface

3.2.2 Burn the system to the TF card

First, connect the TF card to the computer and wait for the software to identify the TF card. After the software is connected to the TF card, the interface is as follows:

Figure 3.2-3 PhoenixCard burning process

Next click the "Firmware" button in the upper left corner of the interface and select the firmware with the suffix "*.img", here the author selected "atk_dlt113is_buildroot_"rgb-7-600p_uart0.img ". The factory system firmware is in the development board disc A - Basic data \8_system_image \EMMC path, and the firmware burned to the SD card can use the EMMC version of the firmware.

Figure 3.2-4 Mirror selection

After selecting the firmware, select the type of card to make as "Boot card", click "Burn card", then you can burn. After burning, the interface is as follows:

Figure 3.2-5 PhoenixCard is done

3.2.3 The TF card Launching tests

Insert the made TF card into the TF card slot of the development board, power on and start, and the development board will start the factory system. The serial port printing information is as follows:

```
Initializing postgresql data base...
su: unknown user postgres
done
Starting postgresql: su: unknown user postgres
OK
/etc/init.d/S50powerkey_display: line 12: powerkey_display: command not found
/etc/init.d/S50powerkey_suspend: line 12: powerkey_suspend: command not found
[   10.465371] random: crng init done
ssh-keygen: generating new host keys: RSA DSA ECDSA ED25519
Starting sshd: Privilege separation user sshd does not exist
OK
[   16.814322] [EXFAT] trying to mount...
formating  /dev/by-name/UDISK to vfat...
mkfs.fat 4.1 (2017-01-24)
Starting dnsmasq: OK
Trying to connect to SWUpdate...
swu_param: ####
swu_software: ####
swu_mode: ####
no swupdate_cmd to run, wait for next swupdate
root@ATK-DLT113IS:/$
```

Figure 3.2-6 Print the information when the startup is complete

3.3 Make batch burn cards

3.3.1 Use of the PhoenixCard software

Allwinner offers a PhoenixCard host burning tool that runs on Windows and needs to be installed before running. The path to the file is: Development board CD disk A - Basic Information→5_tools→3_T113-i System Burning Tool→PhoenixCard V4.3.1.zip. Unzip the file. Open the unzipped file. Double-click to open PhoenixCard.exe. Here the author demonstrates the steps to make a batch card. For more information on the capabilities of the PhoenixCard software, see the PhoenixCard_ User Guide in the PhoenixCard Software sibling directory.

Figure 3.3-1 PhoenixCard

3.3.2 Make a burn card

First, connect the TF card to the computer and wait for the software to identify the TF card. After the software is connected to the TF card, the interface is as follows:

Figure 3.3-2 PhoenixCard interface

Next click the "Firmware" button in the upper left corner of the interface and select the firmware with the suffix "*.img", here the author selected "atk_dlt113is_buildroot_"rgb-7-600p_uart0.img ". The factory system firmware is in the development board disc A - Basic data \8_system_image \EMMC path, and the firmware burned to the SD card can use the EMMC version of the firmware.

Figure 3.3-3 Mirror selection

After the firmware is selected, select the type of card to be made as "mass production card", click "Burn card", and then you can burn. After burning, the interface is as follows:

Figure 3.3-4 Making Card Steps

3.3.3 Update the development board system using the burning card

After making the burning card, insert the burning card into the TF card slot of the development board, power on and start, and wait for the serial port terminal to have the following printed information indicating the completion of system burning. At this time, pull out the burning card and restart the development board to enter the system after burning.

```
chunk 6422(6429)
chunk 6423(6429)
chunk 6424(6429)
chunk 6425(6429)
chunk 6426(6429)
chunk 6427(6429)
chunk 6428(6429)
[52.064]successed in writting part rootfs
origin_verify value = 602dcc99, active_verify value = 602dcc99
[52.074]successed in verify part rootfs
[52.077]successed in download part rootfs
[52.082]successed in downloading part
uboot size = 0x158000
storage type = 2
sunxi_sprite_deal_uboot ok
[52.217]successed in downloading uboot
[52.224][mmc]: write mmc 2 info ok
dram para[0] = 318
dram para[1] = 3
dram para[2] = 7b7bfb
dram para[3] = 1
dram para[4] = 10e2
dram para[5] = 1000000
dram para[6] = 1c70
dram para[7] = 42
dram para[8] = 18
dram para[9] = 0
dram para[10] = 4a2195
dram para[11] = 2423190
dram para[12] = 8b061
dram para[13] = b4787896
dram para[14] = 0
dram para[15] = 48484848
dram para[16] = 48
dram para[17] = 1620121e
dram para[18] = 0
dram para[19] = 0
dram para[20] = 0
dram para[21] = 770000
dram para[22] = 2
dram para[23] = b4056103
dram para[24] = 0
dram para[25] = 0
dram para[26] = 0
dram para[27] = 0
dram para[28] = 0
dram para[29] = 0
dram para[30] = 0
dram para[31] = 0
storage type = 2
[52.293]successed in downloading boot0
CARD OK
[52.297]sprite success
sprite_next_work=3
next work 3
SUNXI_UPDATE_NEXT_ACTION_SHUTDOWN
[55.305][mmc]: mmc exit start
[55.322][mmc]: mmc 2 exit ok
```

Figure 3.3-5 Process of burning

Chapter 4. Functional test

4.1 LED Testing

The ATK-DLT113IS development board is equipped with two LED lights, a blue system power light and a yellow-green user LED light.

Input the following instructions in the factory system to control the user LED light (DS1). See how the DS1 lamp of the current system is triggered.

```
cat /sys/class/leds/heartbeat/trigger        // See the current and supported triggers for DS1
```

The result is shown in the following screenshot:

```
root@ATK-DLT113IS:/$ cat /sys/class/leds/heartbeat/trigger
none rc-feedback rfkill-any rfkill-none kbd-scrolllock kbd-numlock kbd-capslock kbd-kanalock kbd-shiftlock kbd-altgrlo
ck kbd-ctrllock kbd-altlock kbd-shiftllock kbd-shiftrlock kbd-ctrlllock kbd-ctrlrlock mmc0 [heartbeat] rfkill0 rfkill1
```

Figure 4.1-1 View trigger mode

It can be seen that the trigger mode of DS1 at this time is [heartbeat], which means that the current trigger condition is the system heartbeat light. The trigger mode can be modified by modifying the string of trigger. The command will look as follows:

```
echo none > /sys/class/leds/heartbeat/trigger // Change how DS1 is triggered, setting it to none
```

The following command is used to control the DS1 lamp.

```
echo 1 > /sys/class/leds/heartbeat/brightness //Light up DS1
echo 0 > /sys/class/leds/heartbeat/brightness //Put out DS1
```

4.2 KEY test

The ATK-DLT113IS development board is equipped with seven keys, one of which is a reset button, one is a burn button, and the other five are ADC keys derived from the same LRADC pin.

Input the following instructions in the factory system to control the ADC keys. The command is as follows:

```
evtest
```

```
root@ATK-DLT113IS:/$ evtest
No device specified, trying to scan all of /dev/input/event*
Available devices:
/dev/input/event0:      sunxi-keyboard
/dev/input/event1:      sunxi-ts
/dev/input/event2:      sunxi-gpadc0
/dev/input/event3:      sunxi-gpadc1
/dev/input/event4:      audiocodec sunxi Audio Jack
Select the device event number [0-4]: 0
```

Figure 4.2-1 evtest interface

Figure 4.2.1 tells us that "/dev/input/event0" is the interrupt event of gpio-keyboard, just enter 0 (if there are multiple input devices, the number may change, you need to enter the number according to the actual situation)

```
Select the device event number [0-4]: 0
Input driver version is 1.0.1
Input device ID: bus 0x19 vendor 0x1 product 0x1 version 0x100
Input device name: "sunxi-keyboard"
Supported events:
  Event type 0 (EV_SYN)
  Event type 1 (EV_KEY)
    Event code 28 (KEY_ENTER)
    Event code 114 (KEY_VOLUMEDOWN)
    Event code 115 (KEY_VOLUMEUP)
    Event code 139 (KEY_MENU)
    Event code 172 (KEY_HOMEPAGE)
Properties:
Testing ... (interrupt to exit)
```

Figure 4.2-2 Keypress value

In Figure 4.2.2, the red box represents the key value reported by the key. Printing information from the kernel appears at the press of a key. As shown below:

```
Properties:
Testing ... (interrupt to exit)
Event: time 1733423917.366421, type 1 (EV_KEY), code 28 (KEY_ENTER), value 1
Event: time 1733423917.366421, -------------- SYN_REPORT ------------
Event: time 1733423917.561731, type 1 (EV_KEY), code 28 (KEY_ENTER), value 0
Event: time 1733423917.561731, -------------- SYN_REPORT ------------
Event: time 1733423918.624310, type 1 (EV_KEY), code 172 (KEY_HOMEPAGE), value 1
Event: time 1733423918.624310, -------------- SYN_REPORT ------------
Event: time 1733423918.803995, type 1 (EV_KEY), code 172 (KEY_HOMEPAGE), value 0
Event: time 1733423918.803995, -------------- SYN_REPORT ------------
Event: time 1733423920.679124, type 1 (EV_KEY), code 139 (KEY_MENU), value 1
Event: time 1733423920.679124, -------------- SYN_REPORT ------------
Event: time 1733423920.862718, type 1 (EV_KEY), code 139 (KEY_MENU), value 0
Event: time 1733423920.862718, -------------- SYN_REPORT ------------
Event: time 1733423922.323758, type 1 (EV_KEY), code 114 (KEY_VOLUMEDOWN), value 1
Event: time 1733423922.323758, -------------- SYN_REPORT ------------
Event: time 1733423922.499543, type 1 (EV_KEY), code 114 (KEY_VOLUMEDOWN), value 0
Event: time 1733423922.499543, -------------- SYN_REPORT ------------
Event: time 1733423924.296529, type 1 (EV_KEY), code 115 (KEY_VOLUMEUP), value 1
Event: time 1733423924.296529, -------------- SYN_REPORT ------------
Event: time 1733423924.507478, type 1 (EV_KEY), code 115 (KEY_VOLUMEUP), value 0
Event: time 1733423924.507478, -------------- SYN_REPORT ------------
```

Figure 4.2-3 Key press test

4.3 LCD Testing

ATK-DLT113IS development board has three hardware display interfaces RGB (RGB666), LVDS and MIPI DSI.

4.3.1 RGB interface test

RGB565 can be used directly with RGB screens. The resolution of the different screens is identified due to the default high and low state of the ALIENTEK screen at the highest bit R7\G7\B7 of the R, G and B data lines. There are five different RGB screens for ALIENTEK, as shown in the following table:

Screen size	Touch chip	Screen ID identification code
4.3 inch screen (480x272)	gt9xx	0x00
4.3 inch screen (800x480)	gt9xx	0x04
7 inch screen (800x480)	gt9xx	0x01
7 inch screen (1024x600)	gt9xx	0x02
10 inch screen (1280x800)	gt9xx	0x05

The image burned by the factory system of EMMC version development board is atk_dlt113is_buildroot_rgb-7-600p_uart0.img. NAND version factory system burn image atk_dlt113is_nand_buildroot_rgb-7-600p_uart0.img. Therefore, the factory system of the development board connected to the 7 inch RGB screen (1024x600) can light up the screen. The hardware solution is shown in the following figure: (The brown buckle is pulled up to loosen the buckle to take out the cable, and the brown buckle is pressed down to fix the cable)

Figure 4.3-1 RGB interface

4.3.2 LVDS interface testing

Before testing the LVDS screen, we need to burn the firmware corresponding to the screen. The firmware burned by the EMMC version development board is the development board disc A - Basic information \8_system image \EMM\atk_dlt113is_buildroot_lvds_uart0.img, The firmware burned by the NAND version development board is the development board CD A disk - Basic information \8_system image \NAND\atk_dlt113is_nand_buildroot_lvds_uart0.img. Refer to Chapter III for the burning steps. After burning the system, connect the screen and restart the development board to light up the screen. The hardware connection diagram is as follows: (Push the brown button up to loosen the card to take out the cable, press the button down to fix the cable)

Figure 4.3-2 LVDS interface

4.3.3 LVDS interface testing

Before testing the LVDS screen, we need to burn the firmware corresponding to the screen. The firmware burned by the EMMC version development board is the development board disc A - Basic information \8_system image \EMM\atk_dlt113is_buildroot_lvds_uart0.img, The firmware burned by the NAND version development board is the development board CD A disk - Basic information \8_system image \NAND\atk_dlt113is_nand_buildroot_lvds_uart0.img. Refer to Chapter III for the burning steps. After burning the system, connect the screen and restart the development board to light up the screen. The hardware connection diagram is as follows: (Push the brown button up to loosen the card to take out the cable, press the button down to fix the cable)

Board / Screen	EMMC version	NAND version
5.5 inch 720p MIPIscreen	atk_dlt113is_buildroot_mipi-5p5-720p_uart0.img	atk_dlt113is_nand_buildroot_mipi-5p5-720p_uart0.img
5.5 1080p MIPI screen	atk_dlt113is_buildroot_mipi-5p5-1080p_uart0.img	atk_dlt113is_nand_buildroot_mipi-5p5-1080p_uart0.img
10.1 800p MIPI screen	atk_dlt113is_buildroot_mipi-10p1-800p_uart0.img	atk_dlt113is_nand_buildroot_mipi-10p1-800p_uart0.img

Refer to Chapter III for the burning steps. After burning the system, connect the screen and restart the development board to light up the screen. The hardware connection diagram is as follows: (Push the brown button up to loosen the card to take out the cable, press the button down to fix the cable)

Figure 4.3-3 MIPI DSI interface

4.3.4 Touch test

The touch test is as simple as the button test using the evtest command.In the debug terminal of the development board, type the following command:

```
evtest
```

```
root@ATK-DLT113IS:/$ evtest
No device specified, trying to scan all of /dev/input/event*
Available devices:
/dev/input/event0:      sunxi-keyboard
/dev/input/event1:      sunxi-ts
/dev/input/event2:      sunxi-gpadc0
/dev/input/event3:      sunxi-gpadc1
/dev/input/event4:      audiocodec sunxi Audio Jack
/dev/input/event5:      Goodix Capacitive TouchScreen
Select the device event number [0-5]:
```

Figure 4.3-4 evtest tool

/dev/input/event5 is the RGB screen touch test device. Test RGB screen touch direct input 5.

4.3.5 Backlight test

In the default factory system, RGB, LVDS and MIPI screen backlight support 256 levels of PWM adjustment, brightness level 0 to 255, the default 255. When set to 0, the backlight is turned off. The higher the number, the brighter the screen. The RGB screen test code is as follows:

```
cat /sys/class/backlight/backlight/max_brightness // Check the LCD maximum brightness level
cat /sys/class/backlight/backlight/brightness     // Check the current brightness level
echo 100 > /sys/class/backlight/backlight/brightness
// Modify the current brightness level to observe the brightness change of the screen
cat /sys/class/backlight/backlight/brightness     // Check the current brightness level
```

4.4 Serial port test

ATK-DLT113IS development board has two RS485 (uart3 and uart4) on board. Here, we need to use the ALIENTEK USB converter module to test. If the user has 485 related test tools to test by themselves, it is not necessary to use this module. Figure below.

Figure 4.4-1 ALIENTEK multi-in-one USB converter module

Connect the A of RS485 of the base board to the A of the ALIENTEK USB converter module with dupont wire, and connect B to B. Using the ALIENTEK USB converter module, you will see a serial port in the PC(computer side). The hardware connection is shown in the following figure:

Figure 4.4-2 Connection mode between development board and serial port tool

Configure the serial port with the following commands

```
stty -F /dev/ttyAS4 ispeed 115200 ospeed 115200 cs8
```

The stty instruction is explained:

(1) -F(--file) : Opens the specified device and uses it as input instead of standard input

(2) ispeed N: Sets the input rate to N

(3) ospeed N: This sets the output rate to N

(4) csn: Sets the character length to N

See stty-help for more information on how to use this directive

Input the following instructions to wait for the data from the host computer, be careful not to send Chinese, otherwise it may be garbled. Press CTRL-C to end the receive.

```
cat /dev/ttyAS4
```

Figure 4.4-3 The serial port receives data

Receive the cat program and send the data to the host computer using the following command:

```
echo www.openedv.com > /dev/ttyAS4
```

Figure 4.4-4 Serial port to send data

4.5 USB Testing

The T113i chip has two usb ports: USB HOST and USB OTG. ATK-DLT113IS development board uses CH334R to transfer USB HOST out of 4 USB HOST interfaces (two as ordinary USB HOST, one as 4G module, one as usb WiFi). USB OTG is directly exported as the OTG interface of the development board.

Figure 4.5-1 USB interface

4.5.1 USB HOST Test

Test requirements: Get a USB disk (FAT32 format). The development board is normally powered on and connected to the debugging serial port. When we connect the USB disk to the USB HOST interface, the following message is printed:

```
root@ATK-DLT113IS:/# [  132.895406] usb 1-1.2: new high-speed USB device number 5 using sunxi-ehci
[  133.047986] usb-storage 1-1.2:1.0: USB Mass Storage device detected
[  133.055925] scsi host0: usb-storage 1-1.2:1.0
[  134.126263] scsi 0:0:0:0: Direct-Access     Mass      Storage Device   1.00 PQ: 0 ANSI: 0 CCS
[  134.267371] sd 0:0:0:0: [sda] 60579840 512-byte logical blocks: (31.0 GB/28.9 GiB)
[  134.276601] sd 0:0:0:0: [sda] Write Protect is off
[  134.282065] sd 0:0:0:0: [sda] Mode Sense: 03 00 00 00
[  134.288607] sd 0:0:0:0: [sda] No Caching mode page found
[  134.294598] sd 0:0:0:0: [sda] Assuming drive cache: write through
[  134.344614]  sda: sda1
[  134.357825] sd 0:0:0:0: [sda] Attached SCSI removable disk
create /dev/sda
create /dev/sda1
[  134.444134] FAT-fs (sda1): Volume was not properly unmounted. Some data may be corrupt. Please run fsck.
```

The capacity of the U disk

sda denotes the node of the USB drive, and sda1 denotes that the USB drive has only one partition, which is sda1

Figure 4.5-2 USB initialization information

See the above printed information, indicating that the U disk identification is successful. Here the author's U disk format for FAT32 so, can be automatically mounted. To see if the mount was successful, run the following command:

```
df -h
```

```
root@ATK-DLT113IS:/# df -h
Filesystem          Size  Used Avail Use% Mounted on
/dev/root           992M  383M  594M  40% /
devtmpfs            231M     0  231M   0% /dev
tmpfs               240M  3.0M  237M   2% /dev/shm
tmpfs               240M   72K  240M   1% /tmp
tmpfs               240M  164K  240M   1% /run
/dev/by-name/UDISK  6.2G  4.0K  6.2G   1% /mnt/UDISK
/dev/sda1            29G   32K   29G   1% /media/usb0
root@ATK-DLT113IS:/#
```

The sda1 partition of the USB drive is mounted in the /media/usb0 directory of the system

Figure 4.5-3 USB device mount information

4.5.2 USB OTG testing

- OTG HOST test

See Section 4.5.1 for the test method. A typeC to typeA master line is required. The following image:

Figure 4.5-4 typeC to typeA master line

- OTG slave test

OTG slave machine default factory system as ADB function, Allwinner provided this version of ADB can not be used under Windows system, do not know why. For this test, run the following command on Ubuntu to install adb.

```
sudo apt install adb
```

Using TYPC interface to connect to the computer, there is a "Google Configfsffs gadget" device in the VMware mobile device. Connecting this interface to the virtual machine, the success is as follows:

Figure 4.5-5 Connecting to a virtual machine

ADB commands are commonly used in three: shell, push and pull, other advanced commands please do your own research.

1. shell

```
adb shell
exit        // Exit terminal
```

Enter the terminal of the development board, and the terminal at this time is relative to the debugging serial port. As shown below:

Figure 4.5-6 adb shell

1. push

```
adb push xxxx /
```

push means to copy files to the board. Here "xxxx" means the file to copy, and "/" means copy to the root directory.

2. pull

```
adb pull xxxx ./
```

pull means to copy the file of the development board to the Ubuntu system, "xxxx" means to copy the file, "./ "means to copy to the current directory of the Ubuntu system.

4.6 Network testing

ATK-DLT113IS development board equipped with a gigabit network chip, adaptive 10/100/1000M. Use the following command to test the network in the filesystem, and plug the network cable into the network port to ensure that the network cable can access the Internet. Check the IP obtained by the network.

```
ifconfig
```

```
root@ATK-DLT113IS:/$ ifconfig
eth0      Link encap:Ethernet  HWaddr 4A:13:E4:F9:79:75
          inet addr:192.168.6.231  Bcast:192.168.6.255  Mask:255.255.255.0
          inet6 addr: fe80::6946:6d11:9d3a:dbfe/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:221 errors:10 dropped:0 overruns:0 frame:0
          TX packets:14 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:14253 (13.9 KiB)  TX bytes:1608 (1.5 KiB)
          Interrupt:42
```

Figure 4.6-1 Look at the enabled network node eth0

Check whether the network can access the Internet, if there is a data reply, it means that the network function is normal. Press Ctrl+C to terminate the command.

```
ping -I eth0 www.baidu.com                    // Use eth0 network port ping Baidu
```

```
root@ATK-DLT113IS:/$ ping -I eth0 www.baidu.com
PING www.baidu.com (183.2.172.42): 56 data bytes
64 bytes from 183.2.172.42: seq=0 ttl=52 time=4.308 ms
64 bytes from 183.2.172.42: seq=1 ttl=52 time=3.973 ms
64 bytes from 183.2.172.42: seq=2 ttl=52 time=4.207 ms
^C
--- www.baidu.com ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 3.973/4.162/4.308 ms
```

Figure 4.6-2 ping Baidu

Test whether it is Gigabit network, pay attention to the use of gigabit network cable, gigabit router or gigabit switch, PC (computer) network card must be gigabit network card, or the speed of the test or recognition may be 100 megabit! In the figure below, you can see that the software identifies 1Gbps/Full as a gigabit network.

```
root@ATK-DLT113IS:/$
root@ATK-DLT113IS:/$ [ 5078.199221] sunxi-gmac 4500000.gmac0 eth0: Link is Up - 1Gbps/Full - flow control off
```

Figure 4.6-3 Identified as a gigabit network

Gigabit network connection speed can be tested using the iperf3 command.

Set up Ubuntu as the server.

```
iperf3 -s
```

ATK-DLT113IS the client, through the iperf3 command, enter the ip of the server.

```
iperf3 -B 192.168.6.231 -c 192.168.6.51 -i 1
```

Explanation:

- -B 192.168.6.104: IP address of the client, the development board has two IP addresses, one address needs to be specified. Or default to the eth0 network port.
- -c 192.168.6.84: The IP address of the Ubuntu server.
- -i 1: Trigger period in seconds.

```
root@ATK-DLT113IS:/$ iperf3 -B 192.168.6.231 -c 192.168.6.51 -i 1
Connecting to host 192.168.6.51, port 5201
[  5] local 192.168.6.231 port 43815 connected to 192.168.6.51 port 5201
[ ID] Interval           Transfer     Bitrate         Retr  Cwnd
[  5]   0.00-1.01   sec  86.0 MBytes   712 Mbits/sec    0    301 KBytes
[  5]   1.01-2.01   sec  85.5 MBytes   720 Mbits/sec    0    348 KBytes
[  5]   2.01-3.01   sec  86.2 MBytes   721 Mbits/sec    0    368 KBytes
[  5]   3.01-4.00   sec  85.0 MBytes   721 Mbits/sec    0    368 KBytes
[  5]   4.00-5.01   sec  86.2 MBytes   719 Mbits/sec    0    390 KBytes
[  5]   5.01-6.01   sec  86.2 MBytes   720 Mbits/sec    0    390 KBytes
[  5]   6.01-7.01   sec  85.0 MBytes   718 Mbits/sec    0    409 KBytes
[  5]   7.01-8.01   sec  86.2 MBytes   718 Mbits/sec    0    409 KBytes
^C[  5]   8.01-8.38   sec  30.9 MBytes   715 Mbits/sec    0    409 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         Retr
[  5]   0.00-8.38   sec   717 MBytes   719 Mbits/sec    0             sender
[  5]   0.00-8.38   sec  0.00 Bytes  0.00 bits/sec                  receiver
```

Figure 4.6-4 Gigabit network testing

If the gigabit network speed is not measured, please check whether the gigabit network cable is used, whether the network card of the PC is gigabit network card, whether the router is gigabit, etc., can not have a 100 megabit device, otherwise the test is 100 megabit network.

Note: The actual test can not run the theoretical value, please refer to the actual test speed.

4.7 CAN Test

There are two channels of CAN on the ATK-DLT113IS development board. To test the CAN communication function, users need to have an instrument to test CAN (can also use two channels of CAN docking communication), such as Zhou Ligong's CAN analyzer, Chuangxin Technology's CAN analyzer and Guangcheng Technology's CAN analyzer. For the use of CAN instrument and CAN host computer, please refer to the manufacturer's manual. If you CAN't use it, please consult the CAN manufacturer's technical support.

The main CAN features of ATK-DLT113IS development board are as follows:

- Supports the industry standard AMBA Peripheral bus (APB) and is fully compliant with the AMBA specification.
- Support APB 32-bit bus width operation
- Support CAN 2.0A and 2.0B protocol specifications
- Before the test, please use the CAN analyzer or CAN test equipment to connect the CAN interface of ATK-DLT113IS development board, CANH connected to the CANH of the instrument, CANL connected to the CANL of the CAN instrument.
- The USBCANFD-200U of Zhou Ligong is used in this test. The connection is shown in the following figure:

Figure 4.7-1 USBCANFD-200U

- Set the can device communication baud rate of can1 to 1000000, that is, the communication baud rate of 1MBit/s.

```
ip link set awlink1 type can bitrate 1000000 loopback on
```

- Turn on the can1 device

```
ip link set awlink1 up
```

```
root@ATK-DLT113IS:/$ ip link set awlink1 type can bitrate 1000000 loopback on
root@ATK-DLT113IS:/$  ip link set awlink1 up
[   94.474597] IPv6: ADDRCONF(NETDEV_CHANGE): awlink1: link becomes ready
root@ATK-DLT113IS:/$
```

Figure 4.7-2 Turn on the CAN1 device

- Send data:
- CAN send data, input the following commands:

```
cansend awlink1 123#1122334455667788
```

Figure 4.7-3 CAN communication

- Accept data:

For receiving, you can use the candump command to receive data:

```
candump -x -d awlink1 &
```

Figure 4.7-4 CAN communication

4.8 RTC clock testing

ATK-DLT113IS development board has RTC clock chip PCF8563, which belongs to the external RTC clock of the chip. If you need to improve the accuracy of the clock, you need to use high precision crystal.

Please check whether the RTC button battery is installed on the development board bottom plate. You can also use a multimeter to check whether the RTC battery has power, and it is normal to measure about 3.3v. Prevent the RTC battery power can not save time. (Note that the battery is a consumable, if no power, please replace the button battery)

The Linux system is divided into two clocks, one is the system time (software clock), the other is the hardware clock (hardware clock). The date and hwclock commands can be used to view and set the system and hardware time, respectively. The system clock will disappear when the power is off, and the RTC clock will run for a long time when there is a battery. The system clock is synchronized with the RTC clock when the system is restarted.

To check the system clock, use the date command. (As long as the development board can access the Internet, the process of the system will automatically update the time)

```
date
```

Figure 4.8-1 Check the system time

If there is no automatic update time (if the board has an Internet connection), you can run the following command:

```
ntpdate -u cn.pool.ntp.org
```

Figure 4.8-2 Update time

You can set the current system time by running the following command without a cable.

```
date -s "2024-12-10 10:00:00"
```

Set RTC hardware time (write system time into hardware time)

```
hwclock -w
```

Look at the hardware clock to see if it is the one set by hwclock-w above.

```
hwclock
```

```
root@ATK-DLT113IS:/$ hwclock
Tue Dec 10 09:00:15 2024  0.000000 seconds
root@ATK-DLT113IS:/$
```

Figure 4.8-3 Hardware time

Note that the time zone of system time and RTC time is UTC. China time is 8 hours faster than UTC, so the program needs to add 8 hours.

4.9 Audio Testing

4.9.1 Playing the test

Audio test script is pre-installed in the factory system, which is used to quickly run audio test in the terminal to verify the working state of audio equipment. To run the test script, you need to go through a series of steps, first navigate to the specific script directory (/opt/shell/audio), and then execute the script file. The command will look as follows:

```
cd /opt/shell/
./alientek_audio.sh
```

```
root@ATK-DLT113IS:/opt/shell# ./alientek_audio.sh
请选择要运行的测试:
1. 播放测试
2. MIC1录音测试
3. MIC2录音测试
4. 耳机录音测试
5. FMIN录音测试
6. LINEIN录音测试
请输入你的选择 (1-6):
```

Figure 4.9-1 alientek audio.sh script

Enter 1 to start playing the test.

```
root@ATK-DLT113IS:/opt/shell# ./alientek_audio.sh
请选择要运行的测试:
1. 播放测试
2. MIC1录音测试
3. MIC2录音测试
4. 耳机录音测试
5. FMIN录音测试
6. LINEIN录音测试
请输入你的选择 (1-6): 1
运行播放测试...
Press `k` to see a list of keyboard shortcuts.
Now playing /opt/ui/src/apps/resource/audio/陈楚生 - 有没有人告诉你.mp3
[ 6930.060871] [SNDCODEC][sunxi_card_hw_params][635]:stream_flag: 0
Redistribute latency...
[ 6931.466876] [SNDCODEC][sunxi_check_hs_detect_status][192]:plugin --> switch:3
[ 6932.866832] [SNDCODEC][sunxi_check_hs_detect_status][230]:plugout --> switch:0
@:00:04.7 / 0:05:44.0
```

Figure 4.9-2 Play test

4.9.2 Recording test

Recording is the same as playing, and you can use a script to record.First, navigate to the specific script directory (/opt/shell/audio), and the command is as follows:

```
cd /opt/shell/
./alientek_audio.sh
```

```
root@ATK-DLT113IS:/opt/shell# ./alientek_audio.sh
请选择要运行的测试:
1. 播放测试
2. MIC1录音测试
3. MIC2录音测试
4. 耳机录音测试
5. FMIN录音测试
6. LINEIN录音测试
请输入你的选择 (1-6):
```

Figure 4.9-3 Choose the appropriate configuration

Input 2 starts the on-board MIC1 recording test, input 3 starts the on-board MIC2 recording test, input 4 starts the headphone recording test, input 5 starts the FMIN recording test, and input 6 starts the LINEIN recording test. The recording starts for 10 seconds after the selection, and the recorded audio is played automatically after the recording.

```
root@ATK-DLT113IS:/opt/shell# ./alientek_audio.sh
请选择要运行的测试:
1. 播放测试
2. MIC1录音测试
3. MIC2录音测试
4. 耳机录音测试
5. FMIN录音测试
6. LINEIN录音测试
请输入你的选择 (1-6): 2
运行MIC1录音测试...
Recording WAVE 'mic.wav' : [ 6983.000329] [SNDCODEC][sunxi_card_hw_params][635]:stream_flag: 1
Signed 16 bit Little Endian, Rate 48000 Hz, Mono
```

Figure 4.9-4 Recording test

4.10 Onboard wifi test

4.10.1 Station mode

Scan WIFI hotspots, and use the iw instruction provided by the file system to scan the name of the hotspot. To see the full list of hotspots, start with wlan0 and use the iw wlan0 scan command, which filters out some of the information using grep.

```
ifconfig wlan0 up
iw wlan0 scan | grep SSID
```

The following figure scans the hot spot name "ALIENTEK-linux", where "x00\x00\..." Because the Chinese characters can not be displayed. So the user's hot spots had better be composed of English characters.

```
root@ATK-DLT113IS:/$ iw wlan0 scan | grep SSID
[   95.283625] RTW: cfg80211_rtw_scan(wlan0)
[   95.291196] RTW: [HW_VAR_CHECK_TXBUF] Empty in 10 ms
[   95.297040] RTW: wlan0 sleep m0=0x00000002, ori reg_0x4d4=0x00000000
[   99.879926] RTW: wlan0 wakeup m0=0x00000002, ori reg_0x4d4=0x00000002
[   99.887300] RTW: survey done event(38) band:0 for wlan0
[   99.893621] RTW: rtw_indicate_scan_done(wlan0)
        SSID: ALIENTEK-linux
        SSID: qm1506
                * UTF-8 SSID
        SSID:
        SSID:
```

Figure 4.10-1 Scanning hot spots

Edit the hotspot information to connect to, change /etc/wpa_supplicant.conf to the following, and type :wq save and exit. Please change the ssid and psk of personal hotspot.

Explanation:

- ssid is the wireless network name
- psk is the wireless network password

```
vi /etc/wpa_supplicant.conf
```

Enter the following:

```
ctrl_interface=/var/run/wpa_supplicant
ap_scan=1

network={
    ssid="ALIENTEK-linux"
    psk="1590202****"
}
```

If the hotspot is passwordless, change /etc/wpa_supplicant.conf to the following

```
ctrl_interface=/var/run/wpa_supplicant
ap_scan=1

network={
    ssid="ALIENTEK-linux"
    key_mgmt=NONE
}
```

Execute the following commands to connect to the hotspot and get the ip address.

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

```
[  622.322405] RTW: ==> rtw_set_key algorithm(4),keyid(1),key_mask(0)
[  622.331604] RTW: rtw_join_done_chk_ch(wlan0) union:40,1,2
[  622.337345] RTW: rtw_cfg80211_set_encryption, ret=0
[  622.357896] IPv6: ADDRCONF(NETDEV_CHANGE): wlan0: link becomes ready
[  622.358046] RTW: rtw_ndev_notifier_call(wlan0) state:4
[  622.376818] RTW: =>mlmeext_joinbss_event_callback - End to Connection without 4-way
[  622.386277] RTW: rtw_ndev_notifier_call(wlan0) state:4
[  622.395524] RTW: set pairwise key camid:0, addr:02:d0:29:4f:50:c7, kid:0, type:AES
[  622.403180] RTW: cfg80211_rtw_change_station(wlan0) mac:02:d0:29:4f:50:c7
[  622.412588] RTW: set group key camid:1, addr:02:d0:29:4f:50:c7, kid:1, type:AES

root@ATK-DLT113IS:/$ udhcpc -i wlan0
udhcpc: started, v1.33.2
udhcpc: sending discover
udhcpc: sending select for 192.168.3.77
udhcpc: lease of 192.168.3.77 obtained, lease time 86400
deleting routers
adding dns 192.168.3.1
root@ATK-DLT113IS:/$
```

Figure 4.10-2 Connect to a WIFI hotspot

If you see the above result, you have connected to the WIFI hotspot and obtained the ip information. We can also use the ifconfig directive to see the ip address obtained by our wlan0.

```
root@ATK-DLT113IS:/$ ifconfig
eth0      Link encap:Ethernet  HWaddr 4A:13:E4:F9:79:75
          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:42

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: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)

wlan0     Link encap:Ethernet  HWaddr 48:8F:4C:6F:E8:B4
          inet addr:192.168.3.77  Bcast:192.168.3.255  Mask:255.255.255.0
          inet6 addr: fe80::7caa:c14a:f589:e495/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:42 errors:0 dropped:0 overruns:0 frame:0
          TX packets:22 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6394 (6.2 KiB)  TX bytes:3002 (2.9 KiB)
```

Figure 4.10-3 Retrieve the ip address

ping Baidu to test connectivity, or ping gateway to test WIFI connectivity. Note: If the ping fails to Baidu, please restart the development board, do not plug in the network cable, and reconnect to the WIFI! By default, only one network device is allowed to connect to the public network.

```
ping www.baidu.com -I wlan0
ping 192.168.3.1 -I wlan0
```

```
root@ATK-DLT113IS:/$ ping www.baidu.com -I wlan0
[  743.512596] RTW: OnAction_back
[  743.516038] RTW: OnAction_back, action=0
[  743.521353] RTW: issue_addba_rsp_wait_ack(wlan0) ra=02:d0:29:4f:50:c7 status:=0 tid=0 size:64, acked, 1/3 in 10 ms
PING www.baidu.com (183.2.172.42): 56 data bytes
64 bytes from 183.2.172.42: seq=0 ttl=51 time=10.217 ms
64 bytes from 183.2.172.42: seq=1 ttl=51 time=8.399 ms
64 bytes from 183.2.172.42: seq=2 ttl=51 time=11.726 ms
64 bytes from 183.2.172.42: seq=3 ttl=51 time=13.991 ms
^C
--- www.baidu.com ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 8.399/11.083/13.991 ms
root@ATK-DLT113IS:/$
```

Figure 4.10-4 ping Baidu test

4.10.2 Hotspot (SoftAP) mode

We can turn on a WIFI hotspot on the board and let other phones (or other boards) connect to it. In this way, the mobile phone (or other boards) can be connected with the development board to form a local area network, which can communicate with the ATK-DLT113IS board through TCP, UDP, etc. If you do not know what TCP and UDP communication is, it is recommended to look for information on your own.

Directly from the directory /opt/shell/wifi from the previous step, execute the following command

```
source ./wifi_apmode.sh -m softap -d wlan0
```

```
root@ATK-DLT113IS:/opt/shell/wifi# source ./wifi_apmode.sh -m softap -d wlan0
您的WIFI配置信息是:
mode   : softap
device: wlan0
sh: udhcpd: command not found
Configuration file: //opt/shell/wifi/wifi.conf
Using interface wlan0 with hwaddr f0:a8:82:e4:9d:b9 and ssid "alientek_softap"
[   75.620385] IPv6: ADDRCONF(NETDEV_CHANGE): wlan0: link becomes ready
wlan0: interface state UNINITIALIZED->ENABLED
wlan0: AP-ENABLED
[1]+  Done(127)                udhcpd -fS /etc/udhcpd.conf
WIFI设置softap模式完成!
root@ATK-DLT113IS:/opt/shell/wifi#
```

Figure 4.10-5 Script configuration

Enter the following command to view the configured network card information

```
iw dev
```

```
root@ATK-DLT113IS:/opt/shell/wifi# iw dev
phy#0
        Interface wlan1
                ifindex 8
                wdev 0x2
                addr f2:a8:82:e4:9d:b9
                type managed
                channel 6 (2437 MHz), width: 20 MHz (no HT), center1: 2437 MHz
                txpower 42949572.96 dBm
        Interface wlan0
                ifindex 7
                wdev 0x1
                addr f0:a8:82:e4:9d:b9
                ssid alientek_softap
                type AP
                channel 6 (2437 MHz), width: 20 MHz (no HT), center1: 2437 MHz
                txpower 20.00 dBm
root@ATK-DLT113IS:/opt/shell/wifi#
```

Figure 4.10-6 View the configured network card information

As you can see, the name of the open hotspot is "alientek_softap" and the default password is 12345678. To change the hotspot name and password, simply open the wifi_apmode.sh script.

After the phone enters the password to connect, it can be connected, and the development board will automatically assign IP to the phone (please update to the file system above v1.2 version to experience). After successful connection, some mobile phones will judge that the network is unable to access the Internet, and will ask you whether you want to maintain the network connection, we can click to maintain the connection!

The connection is successful as shown below. After that, you can use the factory system Qt UI, TCP and UDP to send messages to each other with the mobile phone! Mobile terminal please download TCP communication related software.

Figure 4.10-7 Mobile connection hotspot

4.11 Onboard Bluetooth test

You can see from the boot print that the Bluetooth driver is loaded.

```
[    4.985043] rtk_btusb: HCI reset.
[    4.992392] RTW: WARN Invalid rate 0x0 in MRateToHwRate
[    4.997697] rtk_btusb: read_ver_rsp->lmp_subver = 0x6fec
[    4.998247] RTW: LocNullData: 5
[    5.004212] rtk_btusb: read_ver_rsp->hci_rev = 0xd8c7
[    5.004216] rtk_btusb: patch_entry->lmp_sub = 0x8723
[    5.004243] rtk_btusb: Rtk patch end 0
[    5.004247] rtk_btusb: chip type value: 0x76
[    5.004252] rtk_btusb: btusb_open set HCI UP RUNNING
[    5.004478] rtk_btcoex: Open BTCOEX
[    5.008513] RTW: WARN Invalid rate 0x0 in MRateToHwRate
[    5.013666] rtk_btusb: btusb_open end
[    5.019456] RTW: LocQosNull: 6
[    5.036717] rtk_btcoex: BTCOEX hci_rev 0xd8c7
[    5.037708] RTW: WARN Invalid rate 0x0 in MRateToHwRate
[    5.043615] rtk_btcoex: BTCOEX lmp_subver 0x6fec
[    5.048000] RTW: LocBTQosNull: 7
[    5.051334] rtk_btcoex: rtk_vendor_cmd_to_fw: opcode 0xfc1b
[    5.056192] RTW: WARN Invalid rate 0x0 in MRateToHwRate
[    5.079716] rtk_btusb: btusb_notify: hci0 evt 3
```

Figure 4.11-1 Bluetooth initialization information

To see if Bluetooth is enabled, run the following code:

```
hciconfig -a
```

```
root@ATK-DLT113IS:/$ hciconfig -a
hci0:   Type: Primary  Bus: USB
        BD Address: 48:8F:4C:6F:E8:B5  ACL MTU: 1021:6  SCO MTU: 255:12
        DOWN
        RX bytes:650 acl:0 sco:0 events:38 errors:0
        TX bytes:401 acl:0 sco:0 commands:38 errors:0
        Features: 0xff 0xff 0xff 0xfa 0xdb 0xbf 0x7b 0x87
        Packet type: DM1 DM3 DM5 DH1 DH3 DH5 HV1 HV2 HV3
        Link policy: RSWITCH HOLD SNIFF PARK
        Link mode: SLAVE ACCEPT
```

Figure 4.11-2 View Bluetooth capabilities

4.11.1 Bluetooth pairing

Bluetooth pairing requires the bluetoothd service, which runs in the background as a system service and is responsible for managing the Bluetooth adapter and handling the connection, pairing, and communication of Bluetooth devices. It is one of the core components related to Bluetooth in Linux system. bluetoothd is stored in /usr/libexec/bluetoothing, and we can start it in the background by running the following command:

```
/usr/libexec/bluetooth/bluetoothd -C &
```

Enter the following command to enter the bluetoothctl interactive mode.

```
bluetoothctl
```

Execute the following commands in turn:

```
power on            # Turn on the power
agent on            # Enable proxies
discoverable on     # The opening can be detected
scan on             # Enable scanning device
```

```
[bluetooth]# pair E0:40:07:B0:07:32
Attempting to pair with E0:40:07:B0:07:32
Failed to pair: org.bluez.Error.AlreadyExists
[bluetooth]# disconnect E0:40:07:B0:07:32
Attempting to disconnect from E0:40:07:B0:07:32
Successful disconnected
[bluetooth]# untrust E0:40:07:B0:07:32
Changing E0:40:07:B0:07:32 untrust succeeded
[bluetooth]# remove E0:40:07:B0:07:32
[DEL] Device E0:40:07:B0:07:32 Alientek
Device has been removed
[bluetooth]# scan on
[bluetooth]# [  562.046180] rtk_btcoex: hci (periodic)inq start
Discovery started
[CHG] Controller 48:8F:4C:6F:E8:B5 Discovering: yes
[CHG] Device 7D:41:02:86:86:81 RSSI: -74
[CHG] Device 7D:41:02:86:86:81 TxPower: 8
[CHG] Device 7D:41:02:86:86:81 ManufacturerData Key: 0x004c
[CHG] Device 7D:41:02:86:86:81 ManufacturerData Value:
  10 07 34 1b 7f 6e 24 d5 28                       ..4..n$.(
[CHG] Device A6:D7:3C:42:4A:D3 RSSI: -70
[CHG] Device 6C:06:D6:A3:FC:AB RSSI: -88
[CHG] Device 72:F2:A7:76:03:23 RSSI: -74
[CHG] Device 72:F2:A7:76:03:23 TxPower: 12
[CHG] Device 72:F2:A7:76:03:23 ManufacturerData Key: 0x004c
[CHG] Device 72:F2:A7:76:03:23 ManufacturerData Value:
  01 20 05 00 00 01 00 08 08 e0 00 00 24 00 10 00  . ..........$...
  00                                               .
[CHG] Device 6F:AC:44:3A:87:F4 RSSI: -76
[CHG] Device B0:CC:FE:4A:CC:28 RSSI: -76
```

Figure 4.11-3 bluetoothctl interaction mode

Wait until you scan your Bluetooth device and enter scan off to stop scanning.

```
scan off
```

Scan to their own Bluetooth device, after confirmation can be paired. Enter the following to pair.

```
pair E0:40:07:B0:07:32          # Fill in your Bluetooth MAC
```

```
[bluetooth]# pair E0:40:07:B0:07:32
Attempting to pair with E0:40:07[  837.202174] rtk_btusb: btusb_notify: hci0 evt 1
:B0:07:32
[  837.211235] rtk_btcoex: hci create connection, start paging
[bluetooth]# [  839.882201] rtk_btcoex: connected, handle 0004, status 0x00
[  839.888664] rtk_btcoex: Page success
[  839.967179] rtk_btcoex: io capability request
[CHG] Device E0:40:07:B0:07:32 Connected: yes
Request confirmation
[agent] Confirm passkey 358594 (yes/no): yes
[Alientek]# [  846.326179] rtk_btcoex: link key notify
[  846.378027] rtk_btcoex: l2cap op 2, len 16, out 0
[  846.383453] rtk_btcoex: RX l2cap conn req, hndl 0x0004, PSM 0x0001, scid 0x0044
[  846.383566] rtk_btcoex: l2cap op 3, len 20, out 1
[  846.392142] rtk_btcoex: PSM(0x0001) do not need parse
[  846.402907] rtk_btcoex: TX l2cap conn rsp, hndl 0x0004, dcid 0x0041, scid 0x0044, result 0x0001
[  846.421029] rtk_btcoex: l2cap op 2, len 16, out 0
[  846.426388] rtk_btcoex: l2cap op 3, len 20, out 1
[  846.426404] rtk_btcoex: RX l2cap conn req, hndl 0x0004, PSM 0x0001, scid 0x0044
[  846.431739] rtk_btcoex: l2cap op 3, len 20, out 1
[  846.440264] rtk_btcoex: PSM(0x0001) do not need parse
[  846.445281] rtk_btcoex: l2cap op 2, len 16, out 1
[  846.451603] rtk_btcoex: TX l2cap conn rsp, hndl 0x0004, dcid 0x0000, scid 0x0044, result 0x0007
[  846.466126] rtk_btcoex: l2cap op 2, len 16, out 0
[  846.471518] rtk_btcoex: l2cap op 3, len 20, out 0
[  846.471559] rtk_btcoex: l2cap op 3, len 20, out 1
[  846.476908] rtk_btcoex: TX l2cap conn rsp, hndl 0x0004, dcid 0x0041, scid 0x0044, result 0x0000
```

Enter yes

Figure 4.11-4 Bluetooth pairing

The phone also pops up the pairing code.

Figure 4.11-5 Cell phone pairing code

Then type exit to exit interactive mode.

```
exit
```

If the Bluetooth connection has no effect or fails, you can clean up the Bluetooth information in bluetoothctl and scan on the connection again:

```
devices                          # View the matched Bluetooth
disconnect E0:40:07:B0:07:32
```

```
# The command is used to disconnect from a specific Bluetooth device
untrust    E0:40:07:B0:07:32        # Removes a Bluetooth device from the trust list
remove E0:40:07:B0:07:32          # Remove a Bluetooth device from the pairing list
```

4.11.2 Bluetooth rfcomm

Bluetooth RFCOMM is a Bluetooth protocol that provides a means of serial data transfer that enables data transfer between Bluetooth devices. RFCOMM can convert serial data into Bluetooth packets for wireless transmission.

RFCOMM provides the concept of a virtual serial port so that data transmission between Bluetooth devices can be carried out like a serial port. This method is widely used in Bluetooth headset, Bluetooth printer and other devices. When using RFCOMM for data transmission, a Bluetooth connection needs to be established and the RFCOMM channel specified. Then, in the devices at both ends of the connection, data can be sent and received via RFCOMM. This method is suitable for application scenarios that require serial communication, such as wireless controllers, wireless sensors, etc.

The author write the open bluetooth rfcomm build rfcomm channel script, located in the/opt/shell/bluetooth/rfcomm_init sh. The script reads as follows:

```
if [[ "$(pidof rfcomm)" != "" ]]
then
kill -9 $(pidof rfcomm)
fi
hciconfig hci0 piscan
hciconfig hci0 noauth
sleep 1
sdptool add SP
sleep 1
rfcomm watch hci0 &
echo "rfcomm init finished! please use bluetoothctl cmd to pair your device and then connect it!"
```

Executing the rfcomm_init.sh script shows that channel 1 has been created and is waiting for a connection.

```
rfcomm_init.sh
```

```
root@ATK-DLT113IS:/opt/shell/bluetooth$ ./rfcomm_init.sh
Serial Port service registered
 init finished! please use !
Waiting for connection on channel 1
root@ATK-DLT113IS:/opt/shell/bluetooth$
```

Figure 4.11-6 Establish the rfcomm channel

At this time, we need to perform Bluetooth pairing, and then use the Android mobile phone "Bluetooth Debug treasure" App to connect. Apple doesn't have a similar app.

4.11.3 Bluetooth communication test

Install Bluetooth Debugger on Android phone. After opening the software, the paired and unpaired Bluetooth will be displayed in the list (if not searched, you may need to open it again to be scanned). You need to execute discoverable on or exit the interactive terminal and then execute hciconfig hci0

piscan to open again. Bluetooth debugger, after opening the software, the matching Bluetooth is shown as follows.

Figure 4.11-7 Paired Bluetooth display

Board receive data can use the cat instruction to receive data

```
cat /dev/rfcomm0
```

```
root@ATK-DLT113IS:/opt/shell/bluetooth$ cat /dev/rfcomm0
zhengdianyuanzi[  883.171269] rtk_btcoex: count_pan_packet_timeout: pan_packet_count 18
[  883.178496] rtk_btcoex: timeout_handler: pan idle->busy!
```

Figure 4.11-8 The development board receives the data

Similarly, we can send data to the phone and execute the following command.

```
echo "正点原子" > /dev/rfcomm0
```

Figure 4.11-9 Development board occurrence data

Note that the author found that the mobile phone without the brand may not be connected to the Bluetooth debugger, if such a problem occurs, you can try to use other testing methods.

4.12 4G module EC20 test

The Quectel EC20 4G module is a high performance, high reliability wireless communication module designed for embedded systems. It supports 4G LTE network, has excellent data transmission

speed and stability, and can be widely used in the Internet of things, Internet of vehicles, industrial automation and other fields.

LED flashing corresponding to LED_WWAN by default:

Pin working state	The indicated network status
Slow flash (200ms high /1800ms low)	Finding network status
Slow flash (1800ms high /200ms low)	Standby mode
Flash (125ms high /125ms low)	Data transfer mode
High level	On the phone

Preparation before the experiment: EC20 4G module, 4G network card and antenna (for amplifying signal).

4G module interface is reserved on the base board of ATK-DLT113IS. ME3630-W, EC20 and other 4G modules are installed. Prepare the EC20 module, pay attention to the need to buy the antenna, the module alone will not work properly! (Note: there are many types of EC20 modules, EC20-CE modules have been tested at present, including EC20-CE series and a variety of modules, different module functions are not the same, such as the support of the operator is not the same, please consult the seller for details), other EC20 series, please test by yourself, theoretically driven the same, there is a need to find far technical support.) . Insert the EC20 4G module into the 4G module interface and screw on the screw. Ensure that the 4G module is connected to the seat interface. Please use the original antenna and connect the antenna to the MAIN interface of the 4G module.

- Correctly insert the 4G card and plug in the module, the WWAN LED on the bottom board will light green after the development board is started. If the green light of the WWAN LED does not light up, please check whether the module is correctly connected and inserted, whether the 4G card is inserted, and whether the antenna is connected properly. The development board must be connected with the 12V power supply provided with it, and cannot only use the serial port USB_TTL for power supply.
- Before the 4G module test, insert the 4G card into the SIM card slot of the bottom board, and then plug in the EC20 4G module. At the same time, plug in the antenna, and the antenna is connected to the MAIN of the module. After inserting the 4G card and antenna correctly, the WWAN LED on the bottom board will light up green light after the development board is started. If the light does not light up, please check whether the 4G card is inserted in the right position and whether the antenna is connected correctly, and then try to reinsert the module. Must plug in the development board used power! Otherwise, the power supply is insufficient and the module cannot work properly. The module installation looks like the following:

Figure 4.12-1 EC20 installation diagram

We can print information on the serial port and see the following print information:

```
root@ATK-DLT113IS:/$ [   14.390023] usb 1-1.1: new high-speed USB device number 5 using sunxi-ehci
[   14.564423] idVendor=2c7c, idProduct=125, bInterfaceNumber =0
[   14.571073] option 1-1.1:1.0: GSM modem (1-port) converter detected
[   14.578877] usb 1-1.1: GSM modem (1-port) converter now attached to ttyUSB0
[   14.588319] idVendor=2c7c, idProduct=125, bInterfaceNumber =1
[   14.599778] option 1-1.1:1.1: GSM modem (1-port) converter detected
[   14.608449] usb 1-1.1: GSM modem (1-port) converter now attached to ttyUSB1
[   14.617552] idVendor=2c7c, idProduct=125, bInterfaceNumber =2
[   14.624395] option 1-1.1:1.2: GSM modem (1-port) converter detected
[   14.633236] usb 1-1.1: GSM modem (1-port) converter now attached to ttyUSB2
[   14.642306] idVendor=2c7c, idProduct=125, bInterfaceNumber =3
[   14.648897] option 1-1.1:1.3: GSM modem (1-port) converter detected
[   14.657509] usb 1-1.1: GSM modem (1-port) converter now attached to ttyUSB3
[   14.668413] qmi_wwan_q 1-1.1:1.4: cdc-wdm0: USB WDM device
[   14.676079] qmi_wwan_q 1-1.1:1.4: Quectel EC20-CE-FILG work on RawIP mode
[   14.689052] qmi_wwan_q 1-1.1:1.4: rx_urb_size = 1520
[   14.695901] qmi_wwan_q 1-1.1:1.4 wwan0: register 'qmi_wwan_q' at usb-sunxi-ehci-1.1, WWAN/QMI Raw IP device, a2:34:bb:5f:58:07
```

Figure 4.12-2 USB information printed by EC20

Then check to see if /dev/ttyusb0-3 nodes are generated

```
ls /dev/ttyUSB*
```

```
root@ATK-DLT113IS:/$ ls /dev/ttyUSB*
/dev/ttyUSB0  /dev/ttyUSB1  /dev/ttyUSB2  /dev/ttyUSB3
```

Figure 4.12-3 Look at the USB generated node

The functions of the four ttyUSB are shown in the following figure, not all these functions are tested, here we only test the Internet function. Please refer to the EC20 4G module manual for details.

Product			USB Driver	Interface
UC15	VID: 0x05c6	PID: 0x9090	USB Serial	ttyUSB0 → DM
UC20	VID: 0x05c6	PID: 0x9003		
EC25	VID: 0x2c7c	PID: 0x0125		ttyUSB1 → For GPS NMEA message output
EC21	VID: 0x2c7c	PID: 0x0121		
EC20	VID: 0x05c6	PID: 0x9215		ttyUSB2 → For AT commands
EC20 R2.0	VID: 0x2c7c	PID: 0x0125		
EG91	VID: 0x2c7c	PID: 0x0191		ttyUSB3 → For PPP connections or AT commands
EG95	VID: 0x2c7c	PID: 0x0195		
UC20	VID: 0x05c6	PID: 0x9003	GobiNet or QMI WWAN	ethX or wwanX → Interface 4 can be used as USB Network Adapter
EC25	VID: 0x2c7c	PID: 0x0125		
EC21	VID: 0x2c7c	PID: 0x0121		
EC20	VID: 0x05c6	PID: 0x9215		
EC20 R2.0	VID: 0x2c7c	PID: 0x0125		
EG91	VID: 0x2c7c	PID: 0x0191		
EG95	VID: 0x2c7c	PID: 0x0195		
EG06	VID: 0x2c7c	PID: 0x0306		
EP06	VID: 0x2c7c	PID: 0x0306		
EM06	VID: 0x2c7c	PID: 0x0306		
BG96	VID: 0x2c7c	PID: 0x0296		

Figure 4.12-4 Functional diagram of four-way ttyUSB

4.12.1 ppp dial-up Internet access

Enter the directory "/opt/shell/4G", which stores the 4G module script. If you do not see the 4G directory, please update the factory system by yourself.

```
cd /opt/shell/4G
```

The ppp dials are mainly ppp-on-1000, ppp-on-10010, and ppp-on-10086. The three scripts are configured with different APN values for different operators. ppp-on-1000, ppp-on-10010 and ppp-on-10086 are the scripts that need to be executed by the telecom card, the script that needs to be executed by the Unicommon card and the script that needs to be executed by the mobile card, respectively.

For example, if a mobile card is used in this test, the script executed would be ppp-on-10086.

```
./ppp-on-10086 &
```

```
Connect: ppp0 <--> /dev/ttyUSB2
sent [LCP ConfReq id=0x1 <asyncmap 0x0> <magic 0x589a14cd> <pcomp> <accomp>]
rcvd [LCP ConfReq id=0x0 <asyncmap 0x0> <auth chap MD5> <magic 0xf9d4d2fc> <pcomp> <accomp>]
sent [LCP ConfAck id=0x0 <asyncmap 0x0> <auth chap MD5> <magic 0xf9d4d2fc> <pcomp> <accomp>]
rcvd [LCP ConfAck id=0x1 <asyncmap 0x0> <magic 0x589a14cd> <pcomp> <accomp>]
rcvd [LCP DiscReq id=0x1 magic=0xf9d4d2fc]
rcvd [CHAP Challenge id=0x1 <c3578775156505b984a38be6245a0190>, name = "UMTS_CHAP_SRVR"]
sent [CHAP Response id=0x1 <a2e8f8f6325ba6f4cdcb754dc0c8cd1e>, name = "Anyname"]
rcvd [CHAP Success id=0x1 ""]
CHAP authentication succeeded
CHAP authentication succeeded
sent [IPCP ConfReq id=0x1 <addr 0.0.0.0> <ms-dns1 0.0.0.0> <ms-dns2 0.0.0.0>]
rcvd [IPCP ConfReq id=0x0]
sent [IPCP ConfNak id=0x0 <addr 0.0.0.0>]
rcvd [IPCP ConfNak id=0x1 <addr 10.78.18.240> <ms-dns1 120.196.165.7> <ms-dns2 221.179.38.7>]
sent [IPCP ConfReq id=0x2 <addr 10.78.18.240> <ms-dns1 120.196.165.7> <ms-dns2 221.179.38.7>]
rcvd [IPCP ConfReq id=0x1]
sent [IPCP ConfAck id=0x1]
rcvd [IPCP ConfAck id=0x2 <addr 10.78.18.240> <ms-dns1 120.196.165.7> <ms-dns2 221.179.38.7>]
Could not determine remote IP address: defaulting to 10.64.64.64
local  IP address 10.78.18.240
remote IP address 10.64.64.64
primary   DNS address 120.196.165.7
secondary DNS address 221.179.38.7
```

Figure 4.12-5 Successful IP acquisition

Use the ifconfig directive to see the ip address obtained.

```
root@ATK-DLT113IS:/opt/shell/4G$ ifconfig
eth0      Link encap:Ethernet  HWaddr 4A:13:E4:F9:79:75
          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:42

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: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)

ppp0      Link encap:Point-to-Point Protocol
          inet addr:10.78.18.240  P-t-P:10.64.64.64  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:4 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3
          RX bytes:52 (52.0 B)  TX bytes:58 (58.0 B)
```

Figure 4.12-6 EC20 PPP0 gets the IP address

Test your Internet connection by ping www.baidu.com. The -I argument specifies ppp0(4G network), and press Ctrl +c to end the ping. The result below indicates that you have Internet access.

```
ping www.baidu.com -I ppp0
```

```
root@ATK-DLT113IS:/opt/shell/4G$ ping www.baidu.com -I ppp0
PING www.baidu.com (183.240.98.161): 56 data bytes
64 bytes from 183.240.98.161: seq=0 ttl=52 time=89.113 ms
64 bytes from 183.240.98.161: seq=1 ttl=52 time=30.111 ms
64 bytes from 183.240.98.161: seq=2 ttl=53 time=32.553 ms
64 bytes from 183.240.98.161: seq=3 ttl=53 time=28.861 ms
64 bytes from 183.240.98.161: seq=4 ttl=53 time=28.566 ms
64 bytes from 183.240.98.161: seq=5 ttl=53 time=28.371 ms
64 bytes from 183.240.98.161: seq=6 ttl=53 time=28.198 ms
64 bytes from 183.240.98.161: seq=7 ttl=53 time=27.881 ms
```

Figure 4.12-7 PPP0 ping Baidu

4.12.2 GPS test

EC20 also has a GPS function, which can be set and turned on through the AT command. This section only teaches how to turn on and view the original GPS data, and other AT commands move company information.

Note: There is a GPS antenna interface in Figure 4.12.1 EC20 installation. EC20 with GPS antenna can be purchased. The antenna needs to be connected to the GPS interface.

The 4G module is normally initialized with the following printing information:

```
root@ATK-DLT113IS:/$ [   14.390023] usb 1-1.1: new high-speed USB device number 5 using sunxi-ehci
[   14.564423] idVendor=2c7c, idProduct=125, bInterfaceNumber =0
[   14.571073] option 1-1.1:1.0: GSM modem (1-port) converter detected
[   14.578877] usb 1-1.1: GSM modem (1-port) converter now attached to ttyUSB0
[   14.588319] idVendor=2c7c, idProduct=125, bInterfaceNumber =1
[   14.599778] option 1-1.1:1.1: GSM modem (1-port) converter detected
[   14.608449] usb 1-1.1: GSM modem (1-port) converter now attached to ttyUSB1
[   14.617552] idVendor=2c7c, idProduct=125, bInterfaceNumber =2
[   14.624395] option 1-1.1:1.2: GSM modem (1-port) converter detected
[   14.633236] usb 1-1.1: GSM modem (1-port) converter now attached to ttyUSB2
[   14.642306] idVendor=2c7c, idProduct=125, bInterfaceNumber =3
[   14.648897] option 1-1.1:1.3: GSM modem (1-port) converter detected
[   14.657509] usb 1-1.1: GSM modem (1-port) converter now attached to ttyUSB3
[   14.668413] qmi_wwan_q 1-1.1:1.4: cdc-wdm0: USB WDM device
[   14.676079] qmi_wwan_q 1-1.1:1.4: Quectel EC20-CE-FILG work on RawIP mode
[   14.689052] qmi_wwan_q 1-1.1:1.4: rx_urb_size = 1520
[   14.695901] qmi_wwan_q 1-1.1:1.4 wwan0: register 'qmi_wwan_q' at usb-sunxi-ehci-1.1, WWAN/QMI Raw IP device, a2:34:bb:5f:58:07
```

Figure 4.12-8The 4G module is initialized normally

You can see above that you have initialized 4 serial ports. Run the following command to use GPS function.

```
echo -e "AT+QGPS=1\r\n" >/dev/ttyUSB2
// Send the AT+QGPS=1 data to USB2 by echo command
cat /dev/ttyUSB1
```

```
root@ATK-DLT113IS:/$ echo -e "AT+QGPS=1\r\n" >/dev/ttyUSB2
root@ATK-DLT113IS:/$ cat /dev/ttyUSB1
$GPVTG,,T,,M,,N,,K,N*2C

$GPGSA,A,1,,,,,,,,,,,,,,,*32

$GPGGA,,,,,,0,,,,,,,,,*66

$GPRMC,,V,,,,,,,,,,,N*53

$GPVTG,,T,,M,,N,,K,N*2C

$GPGSA,A,1,,,,,,,,,,,,,,,*32

$GPGGA,,,,,,0,,,,,,,,,*66
```

Figure 4.12-9 GPS raw data

Disabling GPS allows you to run the following code: You can terminate the cat command by pressing Ctrl + c.

```
echo -e "AT+QGPSEND\r\n" >/dev/ttyUSB2
cat /dev/ttyUSB1
```

At this time we can see that ttyUSB1 serial port no longer prints the original GPS data.

4.13 4G module ME3630-W test

4G module interface is reserved on ATK-DLT113IS, ME3630-W, EC20 and other 4G modules are installed. Insert the ME3630-W module into the interface of the 4G module and screw it on. Ensure that the 4G module is connected to the seat interface. Please use the original antenna and connect the antenna to the MAIN interface of the 4G module. (Warm tip: after the 4G module is unsealed, the gold finger is easy to oxidiz.After a long time, there will be a layer of oxide film on it. If many experiments are not successful, please check whether the 4G module is inserted or whether there is an oxide film. Please save the 4G module after use!)

Insert the 4G card correctly (support mobile, China Unicong, Telecom 4G card) and insert the module, and the WWAN LED on the bottom board will light green after the development board is started. If the green light of the WWAN LED does not light up, please check whether the module is properly connected and inserted, and whether the 4G card is inserted.

Figure 4.13-1 Installation diagram of ME3630

After connecting the 4G module normally, there is the following printed information.

```
[    8.998456] usb 1-1.1: new high-speed USB device number 5 using sunxi-ehci
[    9.162631] idVendor=19d2, idProduct=1476, bInterfaceNumber =0
[    9.169391] option 1-1.1:1.0: GSM modem (1-port) converter detected
[    9.177151] usb 1-1.1: GSM modem (1-port) converter now attached to ttyUSB0
[    9.185983] idVendor=19d2, idProduct=1476, bInterfaceNumber =1
[    9.197102] option 1-1.1:1.1: GSM modem (1-port) converter detected
[    9.204810] usb 1-1.1: GSM modem (1-port) converter now attached to ttyUSB1
[    9.213705] idVendor=19d2, idProduct=1476, bInterfaceNumber =2
[    9.220558] option 1-1.1:1.2: GSM modem (1-port) converter detected
[    9.228552] usb 1-1.1: GSM modem (1-port) converter now attached to ttyUSB2
[    9.242130] cdc_ether 1-1.1:1.3 usb0: register 'cdc_ether' at usb-sunxi-ehci-1.1, ZTE CDC Ethernet Device, e2:74:c4:ca:31:e2
[   10.248488] random: crng init done
```

Figure 4.13-2 ME3630-W Initialization information

After the 4G module is loaded normally, you can see 3 ttyUSB* generated in /dev/. The 4G module communicates with /dev/ttyUSB2 via the AT command.

```
ls /dev/ttyUSB*
```

```
root@ATK-DLT113IS:/$ ls /dev/ttyUSB*
/dev/ttyUSB0  /dev/ttyUSB1  /dev/ttyUSB2
root@ATK-DLT113IS:/$
```

Figure 4.13-3 Look at the ttyUSB node

4.13.1 pppd dial-up Internet access

The factory file system /opt/shell/4G directory has been stored in the test 4G module related scripts. Create a new directory that will be used to generate DNS client configuration files when the following script is run.

```
mkdir /etc/ppp/          # If it already exists, you don't need to duplicate it
```

Go to /opt/shell/4G.

```
cd /opt/shell/4G
```

Use the ls command to see that there are multiple script files in this path. If there are no such files, please update your system in time.

Execute the following ppp-on script with & running in the background.

```
./ppp-on-10086 &
```

```
Using interface ppp0
Connect: ppp0 <--> /dev/ttyUSB2
sent [LCP ConfReq id=0x1 <asyncmap 0x0> <magic 0xe59eeacd> <pcomp> <accomp>]
rcvd [LCP ConfReq id=0x1 <asyncmap 0x0> <magic 0x6bc65d8> <pcomp> <accomp>]
sent [LCP ConfAck id=0x1 <asyncmap 0x0> <magic 0x6bc65d8> <pcomp> <accomp>]
rcvd [LCP ConfAck id=0x1 <asyncmap 0x0> <magic 0xe59eeacd> <pcomp> <accomp>]
sent [IPCP ConfReq id=0x1 <addr 0.0.0.0> <ms-dns1 0.0.0.0> <ms-dns2 0.0.0.0>]
rcvd [IPCP ConfReq id=0x1 <addr 10.15.153.165>]
sent [IPCP ConfAck id=0x1 <addr 10.15.153.165>]
rcvd [IPV6CP ConfReq id=0x1 <addr fe80::1dd9:ba99:8b10:789a>]
Unsupported protocol 'IPv6 Control Protocol' (0x8057) received
sent [LCP ProtRej id=0x2 80 57 01 01 00 0e 01 0a 1d d9 ba 99 8b 10 78 9a]
rcvd [IPCP ConfNak id=0x1 <addr 10.15.153.164> <ms-dns1 120.196.165.7> <ms-dns2 120.196.165.7>]
sent [IPCP ConfReq id=0x2 <addr 10.15.153.164> <ms-dns1 120.196.165.7> <ms-dns2 120.196.165.7>]
rcvd [IPCP ConfAck id=0x2 <addr 10.15.153.164> <ms-dns1 120.196.165.7> <ms-dns2 120.196.165.7>]
local  IP address 10.15.153.164
remote IP address 10.15.153.165
primary   DNS address 120.196.165.7
secondary DNS address 120.196.165.7
```

Figure 4.13-4 PPP dial-up Internet access to obtain IP addresses

Use ifconfig to see the network name of the 4G module.

```
root@ATK-DLT113IS:/opt/shell/4G$ ifconfig
eth0      Link encap:Ethernet  HWaddr 4A:13:E4:F9:79:75
          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:42

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: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)

ppp0      Link encap:Point-to-Point Protocol
          inet addr:10.15.153.164  P-t-P:10.15.153.165  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:4 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3
          RX bytes:68 (68.0 B)  TX bytes:54 (54.0 B)

usb0      Link encap:Ethernet  HWaddr E2:74:C4:CA:31:E2
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:5 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:428 (428.0 B)  TX bytes:658 (658.0 B)
```

Figure 4.13-5 ppp0 network card

ping Baidu test connectivity, -I is the specified network card. If you see the following result, a reply with data means it's connected: Press Ctrl +c to terminate the command.

```
ping www.baidu.com -I ppp0
```

```
root@ATK-DLT113IS:/opt/shell/4G$ ping www.baidu.com -I ppp0
PING www.baidu.com (183.240.98.198): 56 data bytes
64 bytes from 183.240.98.198: seq=0 ttl=51 time=64.537 ms
64 bytes from 183.240.98.198: seq=1 ttl=51 time=38.316 ms
64 bytes from 183.240.98.198: seq=2 ttl=51 time=38.128 ms
64 bytes from 183.240.98.198: seq=3 ttl=51 time=36.425 ms
64 bytes from 183.240.98.198: seq=4 ttl=51 time=36.125 ms
64 bytes from 183.240.98.198: seq=5 ttl=51 time=36.562 ms
64 bytes from 183.240.98.198: seq=6 ttl=51 time=36.382 ms
```

Figure 4.13-6 ME3630-W Internet test

4.13.2 ECM access to the Internet

In the view of the operating system, the CDC ECM device is a virtual Ethernet card that contains the MAC and IP addresses required by standard network cards.

ECM_DEMO_AUTO and ECM_DEMO are high emerging ECM online programs (we don't need to know the source code, because it's just a tool). The instructions are as follows:

```
./ECM_DEMO -t up              // Enable ecm Internet access
./ECM_DEMO -t down            // Turn off the ecm connection
./ECM_DEMO -t up -p /dev/ttyUSB1 -a 3gnet
// When enabled, specify the appropriate apn and dial-in port
./ECM_DEMO -t down -p /dev/ttyUSB1        // Turn off the ecm connection
```

The parameters can be viewed in the source code or by running ECM_DEMO-h

ECM_DEMO_AUTO and ECM_DEMO parameters are the same, the difference is that ECM_DEMO executes the process will exit, ECM_DEMO_AUTO will run and every once in a while will check whether the network is disconnected, the network will automatically reconnect

ECM_DEMO_AUTO executes the dialing process automatically by default, so it is suitable for booting programs.

To configure the ECM connection, if you have pppd running, run the disconnect script to disconnect the pppd dial-up connection, and then run the following command to configure the ECM connection.

```
./disconnect
./ECM_DEMO -t up
```

```
root@ATK-DLT113IS:/opt/shell/4G$ ./ECM_DEMO -t up
[12_12_07:27:16] ECM_DEMO success
```

Figure 4.13-7 ecm-on Internet access

Use the ifconfig command to check the ip address obtained. If the ip address is not obtained, use udhcpc -i usb0 to obtain it.

```
ifconfig
```

```
root@ATK-DLT113IS:/opt/shell/4G$ ifconfig
eth0      Link encap:Ethernet  HWaddr 4A:13:E4:F9:79:75
          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:42

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: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)

usb0      Link encap:Ethernet  HWaddr E2:74:C4:CA:31:E2
          inet addr:10.15.153.164  Bcast:10.15.153.167  Mask:255.255.255.248
          inet6 addr: fe80::ed43:c72:a883:7ff9/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:20 errors:0 dropped:0 overruns:0 frame:0
          TX packets:21 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2192 (2.1 KiB)  TX bytes:2775 (2.7 KiB)
```

Figure 4.13-8 usb0 nodes

We can instruct the usb0 node to ping Baidu.

```
ping www.baidu.com -I usb0
```

4.14 TF(SD) card test

Instruction Prompt:

The time command is often used to measure the running time of a command, dd is used to copy, read from the if(input file) file and write to the of(output file) specified file, bs is the block size of each write, and count is the number of read and write blocks. if=/dev/zero" does not produce IO, that is, it can continuously output data, so it can be used to test pure write speed.

After the development board is started, insert the TF card into the slot at the bottom of the development board. Development board will automatically mount TF card. (Note that you need to use

FAT32 format TF card, not NTFS format). Use the df -h command to determine the partition on which SD is mounted. Here is the result:

```
root@ATK-DLT113IS:/$ df -h
Filesystem          Size  Used Avail Use% Mounted on
/dev/root           992M  394M  583M  41% /
devtmpfs            105M     0  105M   0% /dev
tmpfs               115M     0  115M   0% /dev/shm
tmpfs               115M   80K  114M   1% /tmp
tmpfs               115M  152K  114M   1% /run
/dev/by-name/UDISK  2.5G  4.0K  2.5G   1% /mnt/UDISK
/dev/mmcblk1p1       29G  180M   29G   1% /mnt/sdcard/mmcblk1p1
root@ATK-DLT113IS:/$
```

mmcblk1p1 is the SD card partition device, mounted in /mnt/sdcard/mmcblk1p1 directory

Figure 4.14-1 df-h to view the partition

Read speed test

Often used to test the hard disk read speed instructions have hdparm and dd, this only demonstrates hdparm read TF card speed. (The reading speed depends on the type of TF card you are using)

```
hdparm -t /dev/mmcblk1p1
```

```
root@ATK-DLT113IS:/$ hdparm -t /dev/mmcblk1p1

/dev/mmcblk1p1:
Timing buffered disk reads:   89 MB in 3.00 seconds = 30301 kB/s
root@ATK-DLT113IS:/$
```

Figure 4.14-2 Read test

It can be seen that the read speed is 30301KB/s.

Write speed test

```
time dd if=/dev/zero of=/mnt/sdcard/test bs=1024k count=50 conv=fdatasync
```

```
root@ATK-DLT113IS:/mnt/sdcard$ ls
mmcblk1p1  test
 count=50 conv=fdatasyncdcard$ time dd if=/dev/zero of=/mnt/sdcard/test bs=1024k
50+0 records in
50+0 records out
52428800 bytes (52 MB, 50 MiB) copied, 1.51987 s, 34.5 MB/s

real    0m1.529s
user    0m0.000s
sys     0m0.461s
```

Figure 4.14-3 Writing tests

Delete the test file in the directory "/mnt/sdcard".

4.15 GPADC testing

Note: The maximum absolute value of ADC acquisition voltage is 1.8V, please do not exceed 1.8V, otherwise it may cause damage to the chip. ATK-DLT113IS has an ADC at RP1 to facilitate user testing. The potentiometer is connected to GPADC1 interface, as shown below:

Figure 4.15-1 Potentiometer diagram

In the factory system, there is a tool to test the voltage value of GPADC. You can execute the following command to see the voltage value of GPADC:

```
gpadc_test
```

```
root@ATK-DLT113IS:/$ gpadc_test
Available devices:
/dev/input/event2:      sunxi-gpadc0
/dev/input/event3:      sunxi-gpadc1
Select the device event number: 3
sunxi-gpadc1
[  192.421239] Enable gpadc1
adc vol: 0.95 V
```

Figure 4.15-2 Check the GPADC voltage value

4.16 CVBS IN interface test

To test the CVBS IN camera interface, it is necessary to prepare an analog camera that supports NTSC standard.

The camera module is connected to the TV_IN1 interface of the development board, and the connection with the development board is shown as follows: (The screen used in the test is an RGB screen)

Figure 4.16-1 Connect development board

There is tvd_test test tool in the development board, which can be used to test the interface. Execute the following command to display the camera data on the screen.

```
fbinit
tvd_test -d /dev/video4 -c 1 -i 720*480 -I 720*480
```

```
root@ATK-DLT113IS:/$ tvd_test -d /dev/video4 -c 1 -i 720*480 -I 720*480
[    38.080814] VE: enable hw clock
[    38.084378] enable_cedar_hw_clk,563
INFO    : cedarc <VeInitialize:14[    38.090277] VE: VE real_freq=576000000
[    38.090277]
95>: *** ic_version = 0x1301000010210,
INFO    : cedarc <VeInitialize:1495>: *** ic_version = 0x1301000010210,
INFO    : cedarc <log_set_level:71>: Set log level to 5 from /vendor/etc/cedarc.c[    38.113351] [tvd] vidioc_s_fmt_vid_cap:1623
[    38.113351] interface=0
[    38.113351] system=NTSC
[    38.113351] format=0
[    38.113351] output_fmt=YUV420
onf
```

Figure 4.16-2 tvd_test

After successful execution, the camera data will be displayed on the screen.

Figure 4.16-3 Camera picture

4.17 CPU temperature

Use the following instructions to check the temperature of the CPU, the temperature is related to the ambient temperature and CPU running status, such as running some large applications, running video and audio, the temperature will be higher, it is not recommended to touch the CPU surface by hand to prevent the chip from burning out with static electricity.

```
cat /sys/devices/virtual/thermal/thermal_zone*/temp
```

The result is shown in the following screenshot:

```
root@ATK-DLT113IS:/$ cat /sys/devices/virtual/thermal/thermal_zone*/temp
32696
root@ATK-DLT113IS:/$
```

Figure 4.17-1 Checking the CPU temperature

4.18 CPU frequency

ATK-DLT113IS development board processor supports up to 1.4GHz main frequency. Run the following command to see which dynamic modes are supported by the CPU:

```
cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_available_governors
```

The result is shown in the following screenshot:

```
governorsDLT113IS:/$ cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_available_g
conservative ondemand userspace powersave performance schedutil
root@ATK-DLT113IS:/$
```

Figure 4.18-1 Check the working mode of the CPU

As you can see, there are six modes supported: conservative, ondemand, userspace, powersave, performance, and schedutil.

- ondemand mode: (Ondemand mode) In this mode, the CPU frequency is dynamically adjusted according to the load of the system. When the system load is low, the CPU frequency is reduced to save energy. When the system load is high, the CPU frequency is increased to provide better performance.
- userspace mode: (User mode) In this mode, the user can manually control the CPU frequency. The user can set the frequency of the CPU with specific tools or commands without relying on the system to adjust it automatically.
- performance mode: (Performance mode) This is a high performance mode where the CPU is operated at the highest frequency to provide maximum computing power. This pattern is suitable for tasks that require maximum performance, but may consume more energy.
- schedutil mode: (Scheduler mode) This is a newer scheduler mode in the Linux kernel that tries to dynamically select the appropriate CPU frequency based on the workload of the system. Its goal is to minimize energy consumption and heat while providing good performance.

Check the current CPU frequency

```
cat /sys/devices/system/cpu/cpu*/cpufreq/cpuinfo_cur_freq
// Check the current CPU frequency
cat /sys/devices/system/cpu/cpu*/cpufreq/scaling_cur_freq
// Check the last CPU set on the software frequency
cat /sys/devices/system/cpu/cpufreq/policy0/scaling_available_frequencies
// See the supported CPU frequencies
```

```
echo performance > /sys/devices/system/cpu/cpufreq/policy0/scaling_governor
// Enable CPU performance mode, run the highest frequency
echo userspace > /sys/devices/system/cpu/cpufreq/policy0/scaling_governor
// Switch governor to userspace
echo 1008000 > /sys/devices/system/cpu/cpufreq/policy0/scaling_setspeed
// Setting the CPU frequency requires first switching to userspace
```

4.19 QT interface

ATK-DLT113IS development board default is to run QT interface.

4.19.1 QT is briefly closed

To close QT temporarily, run the following code:

```
/etc/init.d/S90weston stop
fbinit
```

The first command temporarily shuts down QT. The second command clears the display of the screen. Run this code and you'll see the screen go black.

4.19.2 QT is permanently closed

To permanently close QT, simply remove the "S90weston" and "S91systemui" files from the /etc/init.d/ directory.

4.20 Problems with the serial port

When you type a long command, you find that you can't wrap a line. You can fix this by running the following code after the terminal is started:

```
resize
export TERM=xterm
```

Once this is done, the terminal is now ready to enter long commands.