

EMA40i Linux Reference User Manual

V2.0



Boardcon Embedded Design

www.boardcon.com

Revision History

Ver	Description	Author	Date
V1.0	Initial version	Wei Xianya	2019-05-27
V2.0	Modify testing	Zhou Lijun	2019-10-18

1. Introduction

1.1. About this Manual

This manual is intended to provide the user with an overview of the board and benefits, complete features specifications, and set up procedures. It contains important safety information as well.

1.2. Feedback and Update to this Manual

To help our customers make the most of our products, we are continually making additional and updated resources available on the Boardcon website (www.boardcon.com , www.armdesigner.com).

These include manuals, application notes, programming examples, and updated software and hardware. Check in periodically to see what's new!

When we are prioritizing work on these updated resources, feedback from customers is the number one influence, If you have questions, comments, or concerns about your product or project, please no hesitate to contact us at support@armdesigner.com.

1.3. Limited Warranty

Boardcon warrants this product to be free of defects in material and workmanship for a period of one year from date of buy. During this warranty period Boardcon will repair or replace the defective unit in accordance with the following process:

A copy of the original invoice must be included when returning the defective unit to Boardcon. This limited warranty does not cover damages resulting from lighting or other power surges, misuse, abuse, abnormal conditions of operation, or attempts to alter or modify the function of the product.

This warranty is limited to the repair or replacement of the defective unit. In no event shall Boardcon be liable or responsible for any loss or damages, including but not limited to any lost profits, incidental or consequential damages, loss of business, or anticipatory profits arising from the use or inability to use this product.

Repairs make after the expiration of the warranty period are subject to a repair charge and the cost of return shipping. Please contact Boardcon to arrange for any repair service and to obtain repair charge information.

Content

1 EMA40i Introduction.....	4
1.1 Summary	4
1.2 Processor Features	4
1.3 EMA40i specifications	5
2. Compiler Environment.....	6
2.1 Vmware10.0+ubuntu16.04	6
2.2 Install Tools	6
3. Compile the Source.....	7
3.1 Unzip & Configuration.....	7
3.2 Compile Uboot.....	8
3.3 Compile Kernel	8
3.4 Compile Buildroot	9
3.5 Compile QT	9
3.5.1 Compile QT Library	9
3.5.2 Compile QT app demo	10
3.6 Compile sdk_lib Library.....	10
3.7 Pack Firmware	10
3.8 Automatic Compilation	11
4. Burning Guide.....	13
4.1 Install Firmware Tool.....	13
4.2 Burn image to eMMC via USB	13
5 EMA40i Application Guidance	16
5.1 Serial Terminal	16
5.2 HDMI Display	17
5.3 LVDS and Touch	17
5.4 SD Card.....	17
5.5 USB Host	17
5.6 SSD (SATA 2.0)	18
5.7 EC20(4G)	18
5.8 GPS.....	19
5.9 RTC(PCF8563)	20
5.10 Video Player.....	20
5.10.1 Play Video.....	20
5.10.2 xplayerdemo.....	23
5.11 Audio Player.....	23
5.12 Recording	25
5.13 Ethernet	26
5.14 WiFi	27
5.15 Bluetooth	29
5.16 3D.....	32
5.17 RS485.....	33
5.18 RS232&UART (UART5 / UART6 / UART4 / UART7)	34

1 EMA40i Introduction

1.1 Summary

EMA40i is a single board computer featuring an Allwinner A40i Processor, comes with 1GB DDR3 RAM, 4GB eMMC, and other rich interfaces specifically designed for intelligent industrial control applications such as industrial control, communications and measurement.

EMA40i provides a standard Gigabit Ethernet port and a 10-pin Extra GbE port. Supports HDMI 1.4, RGB, MIPI SDI and dual-LVDS output. The SBC is equipped with a microSD slot and a M.2 slot with NAME SSD support. A mini-PCIe slot supports 4G module and is accompanied by a Nano-SIM slot. Other features include USB 2.0 host ports, as well as a USB2.0 OTG. There's also a RS232 DB9 port, as well as 4x 4-pin headers for RS232, UART(TTL) and USB host.

EMA40i is equipped with WiFi 802.11ac and Bluetooth 4.0, as well as optional support for GPS, 4G LTE connectivity.

1.2 Processor Features

- **CPU**

Quad-core ARM Cortex-A7 CPU Architecture, the most power efficient CPU core ARM's ever developed.

- **GPU**

Mali400 MP2

- **Video Engine**

Supports mainstream high-definition video decoding including H.264, H.263, MPEG1/2/4, xvid, Sorenson Spark, VP6/8, AVS/AVS+, WMV7, WMV8 by 1080p@60fps. In the aspect of video encoding, the A40i supports 1080p@45fps H.264 encoding ability.

- **Camera**

Supports dual COMS sensor parallel interfaces and 4-channel TVIN, which can easily finish multi channel video recording.

- **Display**

Content can be display on 4-lane MIPI DSI display, or RGB panel, or LVDS panel. TV-out on HDMI V1.4 is also supported.

- **Audio**

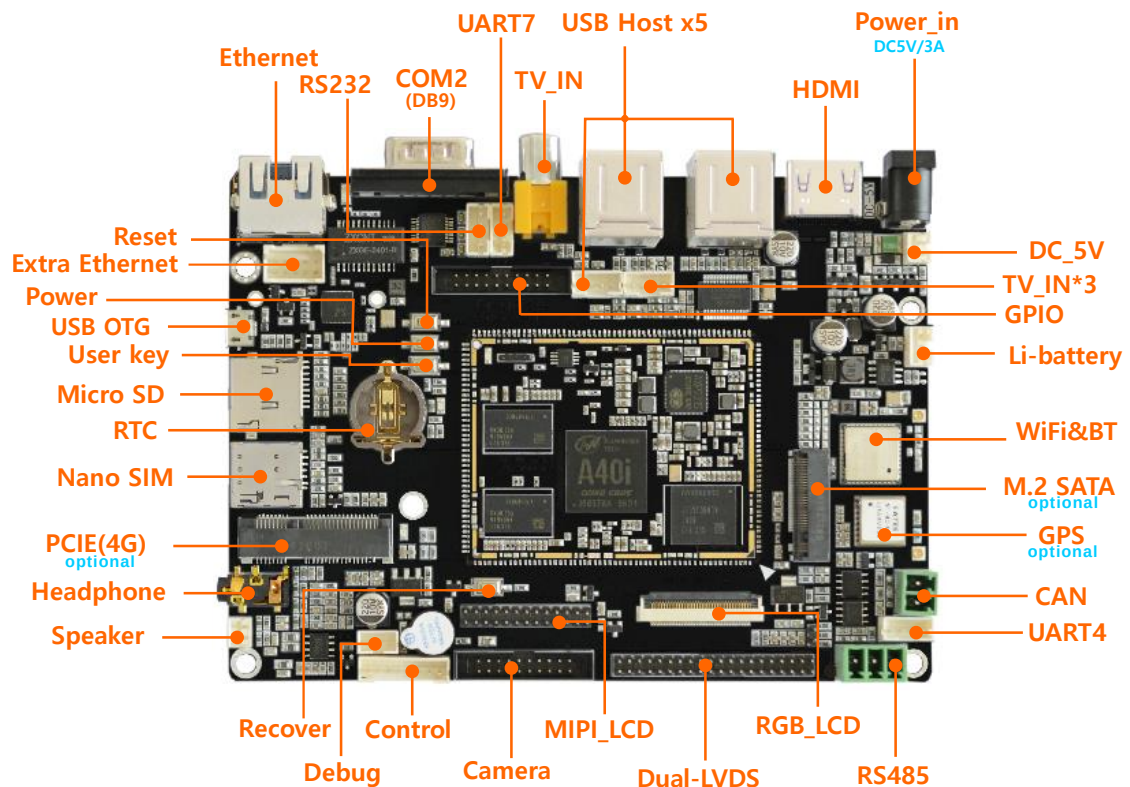
Integrated audio codec with 24 bit/192kHz DAC playback, and supports I2S/PCM interface for connecting to an external audio codec. I2C/PCM interface includes eight channels of TDM with sampling precision up to 32 bit/192kHz.

- **Memory**

Supports external memory interface to NAND Flash, SD/EMMC, Nor Flash and SDRAM port. SDRAM port can be configured to support LPDDR2, LPDDR3, DDR2, DDR3, and DDR3L.

1.3 EMA40i specifications

- * **Processor** – Allwinner A40i Quad-core ARM Cortex-A7 MPCore Processor @ 1.2 GHz
- * **RAM** – 1GB DDR3
- * **eMMC Flash** – 4GB
- * **Interfaces** – Ethernet, 5x UART, TV-IN, 3x USB Host, USB OTG, HDMI, M.2 SATA, PCI-E, CAN, RS485, LCD(RGB, LVDS, MIPI), Camera, GPIO, Audio I/O, SD, SIM, etc.
- * **Operating system**: Linux3.10, Android7.0.1
- * **Application** – Industrial control, communications and measurement, etc.
- * **Dimension** – 140mm x 100mm



2. Compiler Environment

2.1 Vmware10.0+ubuntu16.04

Install Vmware10.0 in windows OS, and then install ubuntu16.04 in VMware to compile. There is no longer describes how to install Ubuntu system, if the user has not understood, please visit the official website of Ubuntu, the operating system is also available for free download at the official website <http://www.ubuntu.com/>

Note: Linux should be compiled by ubuntu 64-bit OS.

2.2 Install Tools

```
# sudo apt-get install build-essential
# sudo apt-get install zlib1g-dev
# sudo apt-get install flex
# sudo apt-get install libx11-dev
# sudo apt-get install gperf
# sudo apt-get install libncurses5-dev
# sudo apt-get install bison
# sudo apt-get install lsb-core
# sudo apt-get install lib32z1-dev
# sudo apt-get install g++-multilib
# sudo apt-get install lib32ncurses5-dev
# sudo apt-get install uboot-mkimage
# sudo apt-get install g++-4.4-multilib
```

If the compilation encountered error, please according to the error message, the installation of the corresponding software package

3. Compile the Source

3.1 Unzip & Configuration

Step 1, unzip the source.

```
$tar -xvf A40i-linux-v1.3.tar.bz2
```

```
$cd a40i-linux-v1.3
```

Step 2, configuration environment

```
$ ./build.sh config
```

Welcome to mkscript setup progress

All available chips:

0. sun8iw11p1

Choice: 0

All available platforms:

0. android

1. dragonboard

2. linux

3. camdroid

Choice: 2

All available kernel:

0. linux-3.10

Choice: 0

All available boards:

0. a40i-boardcon

1. a40i-p3

Choice: 0

All available buildroot:

0. buildroot

1. buildroot-201611

Choice: 1

select buildroot-201611

All available float:

0. gnueabi

1. gnueabihf

Choice: 1

select gnueabihf

All available qt version:

0. 5.9.0

Choice: 0

select 5.9.0

create misc_config gnueabihf



```

weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$
weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$
weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$ ./build.sh config

Welcome to mkscript setup progress
All available chips:
  0. sun8iw11p1
Choice: 0
All available platforms:
  0. android
  1. dragonboard
  2. linux
  3. camdroid
Choice: 2
All available kernel:
  0. linux-3.10
Choice: 0
All available boards:
  0. a40i-boardcon
  1. a40i-p3
Choice: 0
All available buildroot:
  0. buildroot
  1. buildroot-201611
Choice: 1
select buildroot-201611
All available float:
  0. gnuabi
  1. gnuabihf
Choice: 1
select gnuabihf
All available qt version:
  0. 5.9.0
Choice: 0
select 5.9.0
create misc_config gnuabihf
weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$
weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$
weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$

```

Step 3, compile

`$ ./build.sh`

3.2 Compile Uboot

Method 1 compile uboot

`$ cd brandy`

`$ ./build.sh -p sun8iw11p1`

Method 2 Separately compile each part of the boot bin file

`$ cd brandy/u-boot-2014.07`

`$ make distclean`

`$ make sun8iw11p1_config`

`$ make -j32`

After compiling, **u-boot-sun8iw11p1.bin** are generated in the directory:

`lichee\tools\pack\chips\sun8iw11p1\bin\u-boot-sun8iw11p1.bin`

NOTE: After compiling separately, you need to repackage.

Back to **a40i-linux-v1.3** directory, execute `./build.sh`, then `./build.sh pack`, the updated **uboot** is compiled into the new package.

3.3 Compile Kernel

`$cd lichee/linux-3.10`

`$make menuconfig ARCH=arm`

```
$make ARCH=arm CROSS_COMPILE=arm-linux-gnueabi-
```

If it is a soft floating-point toolchain, the last sentence is changed to bold **-gnueabi-**

3.4 Compile Buildroot

If add some packages, do the following (take add **mplayer** as an example)

```
$ cd buildroot-201611/
```

```
$ cp -f configs/sun8iw11p1_defconfig ./
```

```
$ make
```

Select **Audio and video applications** ---> **mplayer**, than **exit**----save, will get the new .config

```
$cp -f .config configs/sun8iw11p1_defconfig
```

```
$ cd ..
```

```
$ rm -f out/sun8iw11p1/linux/common/buildroot/.config
```

```
$ rm -f out/sun8iw11p1/linux/common/buildroot/..config.tmp
```

```
$/build.sh
```

3.5 Compile QT

3.5.1 Compile QT Library

Before compiling the qt library, you need to execute `./build.sh config` and `./build.sh` to ensure that libraries other than Qt are up to the latest.

```
$ ./build.sh config
```

```
$ ./build.sh
```

```
$ cd buildroot-201611/dl
```

```
$ tar zxvf qt-everywhere-opensource-src-5.9.0.tgz
```

```
$ cp -f setenvs.sh qt-everywhere-opensource-src-5.9.0/
```

```
$ cd qt-everywhere-opensource-src-5.9.0/
```

```
$ source ./setenvs.sh
```

```
$makeconfig
```

```
$makeall
```

```
$makeinstall
```

```
$cd ../../../../
```

```
$/build.sh
```

QT5.9.0 are generated in the following directory:

```
lichee/out/sun8iw11p1/linux/common/buildroot/target/usr/local/Qt-5.9.0
```

or

```
lichee/buildroot-201611/dl/qt-everywhere-opensource-src-5.9.0/QT5.9.0
```

If there is something in the QT5.9.0 directory, the Qt library has been successfully.

Compiled: ~/lichee/buildroot-201611/dl/qt-everywhere-opensource-src-5.9.0/Qt-5.9.0\$ **ls**
bin doc include lib mkspecs plugins qml translations

Note:

If do not modify the QT library later, or do not clear the above directory, it does not need to QT library directory to compile again, just needs to execute `./build.sh` and `./build.sh pack` to package the firmware.

3.5.2 Compile QT app demo

➤ **Compile all QT app demo**

```
$cd buildroot-201611/target/user_rootfs_misc/qt_demo/  
$build.sh
```

or

```
$rebuildAll.sh
```

Rebuildall.sh will `make distclean` before compilation.

➤ **Compile qt application demo separately**

Enter a single demo directory and execute the following commands:

```
$cd CameraUI/  
$./makeCameraUI
```

```
$cd Launcher/  
$./makeLauncher
```

```
$cd MediaUI/  
$./makeMediaUI
```

3.6 Compile sdk_lib Library

The compilation of `sdk_lib` is already integrated in `./build.sh`, and for development purposes, this section only explains how to compile separately.

```
$cd buildroot/target/user_rootfs_misc/sdk_lib  
$make cleanall  
$make
```

3.7 Pack Firmware

After the compilation is completed, pack

```
# ./build.sh pack
```



```

weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$
weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$
weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$ ./build.sh pack
build buidroot-201611...
xgnewabi1hf
sun8iw11p1_hf_defconfig
INFO: packing firmware ...
copying tools file
copying configs file
./out/aultls32.fex
./out/aultools.fex
./out/boot_package.cfg
./out/boot_package.fex
./out/boot_package_nor.cfg
./out/cardscrip.fex
./out/cardscrip_secure.fex
./out/cardtool.fex
./out/diskfs.fex
./out/env.cfg
./out/env_bak.cfg
./out/env_burn.cfg
./out/image.cfg
./out/image_linux.cfg' -> './out/image.cfg'
./out/split_xxxx.fex
./out/sunxi.fex
./out/sys_config-MIPI.fex
./out/sys_config.fex
./out/sys_config_3d.fex
./out/sys_config_PCI0.fex
./out/sys_config_RV1.fex
./out/sys_config_bak.fex
./out/sys_config_hdmi.fex
./out/sys_partition_dump.fex
./out/sys_partition_linux.fex' -> './out/sys_partition.fex'
./out/sys_partition_linux_128M.fex
./out/sys_partition_linux_256M.fex
./out/sys_partition_linux_mmc.fex
./out/sys_partition_private.fex

FilePath: rootfs.fex
FileLength=14ec8800sys_config.fex Len: 0x125df
config.fex Len: 0xc8ac
split_xxxx.fex Len: 0x200
sys_partition.fex Len: 0xaba
sunxi.fex Len: 0x259d0
boot0_nand.fex Len: 0x8000
boot0_sdcard.fex Len: 0x8000
u-boot.fex Len: 0x108000
toc1.fex Len: 0x8
toc0.fex Len: 0x8
fes1.fex Len: 0x3e00
boot_package.fex Len: 0x10c000
full_img.fex Len: 0x16
usbtool.fex Len: 0x23000
aultools.fex Len: 0x27c29
aultls32.fex Len: 0x2455d
cardtool.fex Len: 0x11c00
cardscrip.fex Len: 0x779
sunxi_mbr.fex Len: 0x10000
dlinfo.fex Len: 0x4000
arisc.fex Len: 0x6
boot-resource.fex Len: 0x5d9000
Vboot-resource.fex Len: 0x4
env.fex Len: 0x20000
Venv.fex Len: 0x4
boot.fex Len: 0xc05800
Vboot.fex Len: 0x4
rootfs.fex Len: 0x14ec8800
Vrootfs.fex Len: 0x4
BuildImg 0
Dragon execute image cfg SUCCESS 1
-----image is at-----
/home/weixianya/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3/tools/pack/sun8iw11p1_linux_a40i-boardcon_uart0.img
pack finish
weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$

```

The final burnable image is **sun8iw11p1_linux_a40i-boardcon_uart0.img**, which is in the directory of **a40i-linux-v1.3\tools\pack**.

Note: A40i login: **root**

3.8 Automatic Compilation

The steps for automatic compilation are as follows.

\$cd a40i-linux-v1.3

./build.sh config



```
weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$  
weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$  
weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$ ./build.sh config  
  
Welcome to mkscript setup progress  
All available chips:  
0. sun8iw11p1  
Choice: 0  
All available platforms:  
0. android  
1. dragonboard  
2. linux  
3. camdroid  
Choice: 2  
All available kernel:  
0. linux-3.10  
Choice: 0  
All available boards:  
0. a40i-boardcon  
1. a40i-p3  
Choice: 0  
All available buildroot:  
0. buildroot  
1. buildroot-201611  
Choice: 1  
select buildroot-201611  
All available float:  
0. gnuabi  
1. gnuabihf  
Choice: 1  
select gnuabihf  
All available qt version:  
0. 5.9.0  
Choice: 0  
select 5.9.0  
create misc_config gnuabihf  
weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$  
weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$  
weixianya@boardcon:~/opt/A40i/A40i-linux-v1.3/a40i-linux-v1.3$
```

`./build.sh`

`./comp_qtLib-590_only.sh`

`./build.sh pack`

`comp_qtLib-590_only.sh` is The compilation of QT library, you can see in the directory of `a40i-linux-v1.3\comp_qtLib-590_only.sh`.

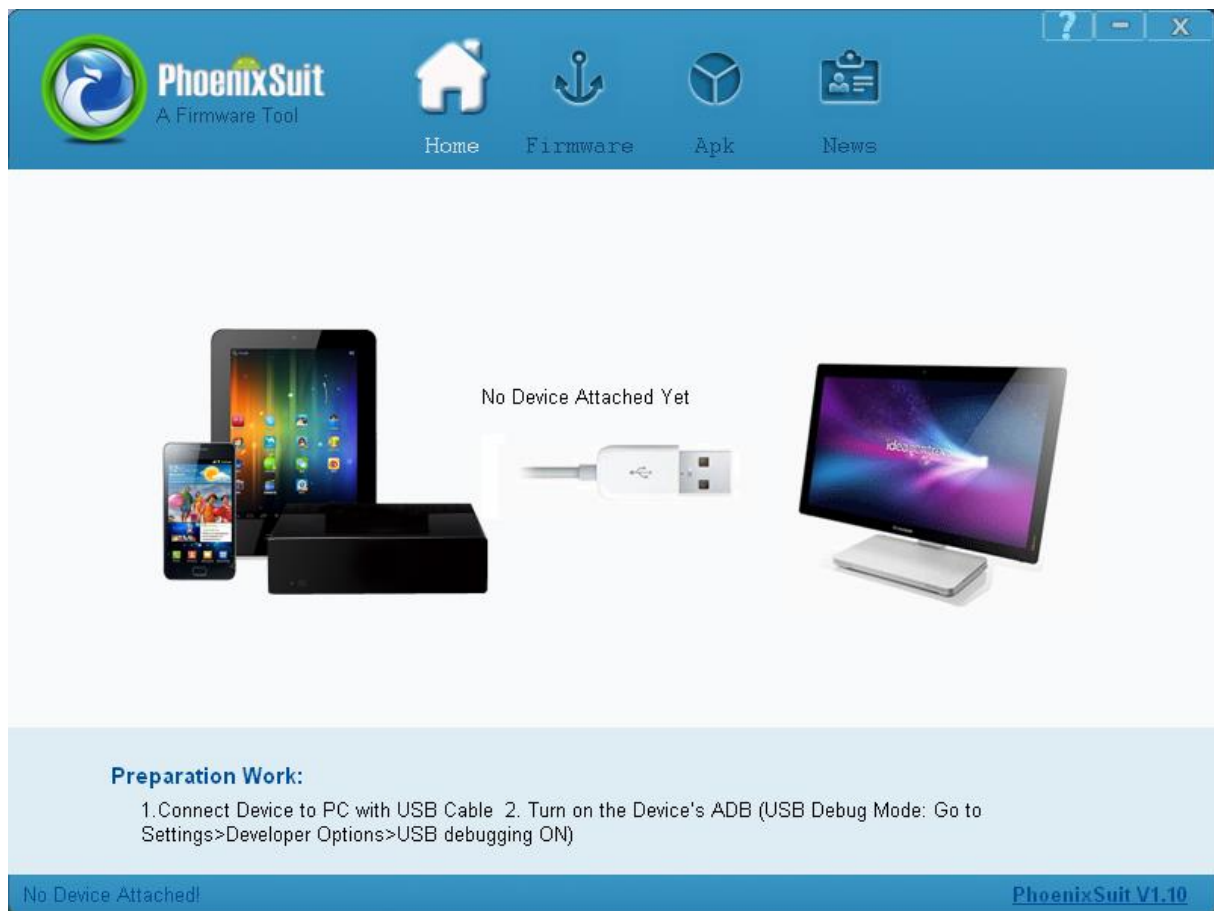
4. Burning Guide

4.1 Install Firmware Tool

Please install **PhoenixSuit1.0.6** (*Path: CD\Tools\PhoenixSuit1.0.6*) according to the PC system.

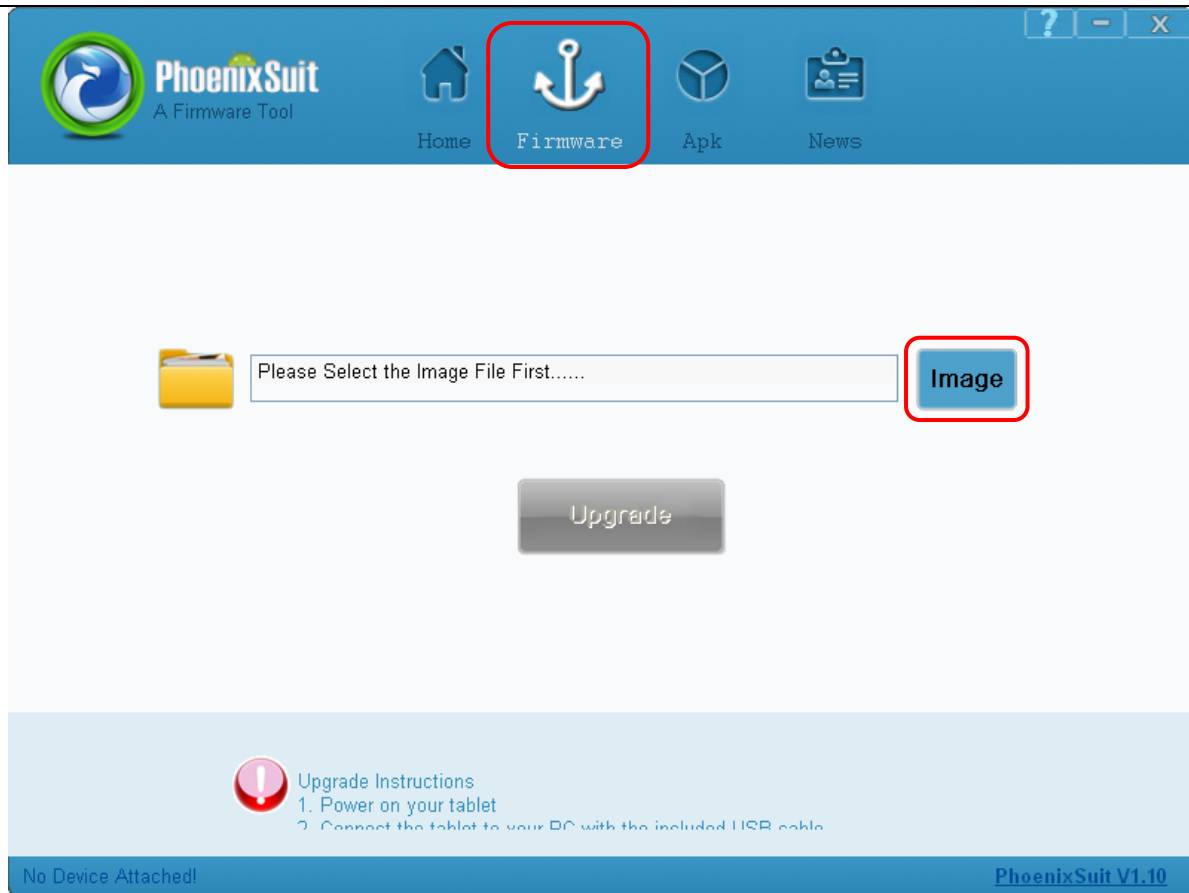
4.2 Burn image to eMMC via USB

Open the tool **PhoenixSuit**

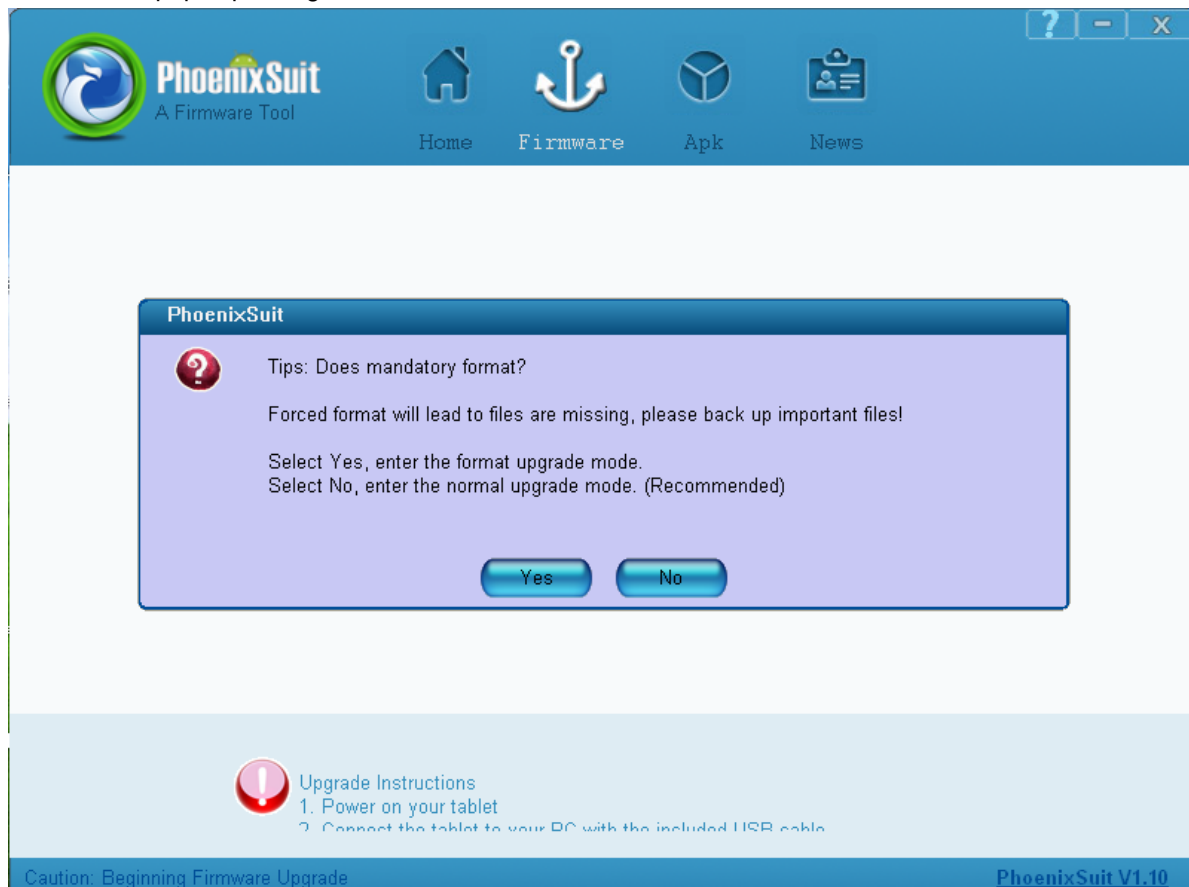


Select "Firmware"

Click "Image" and select the image (*Path: CD\image*)

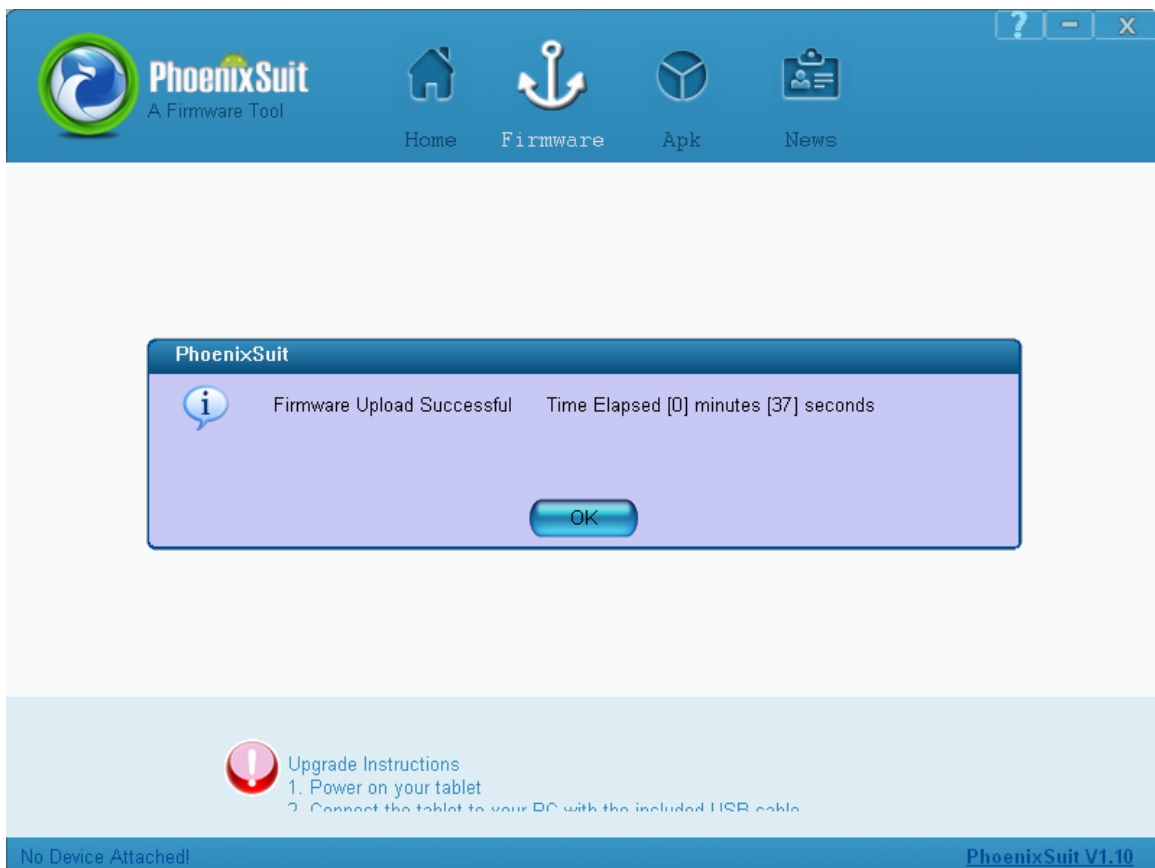
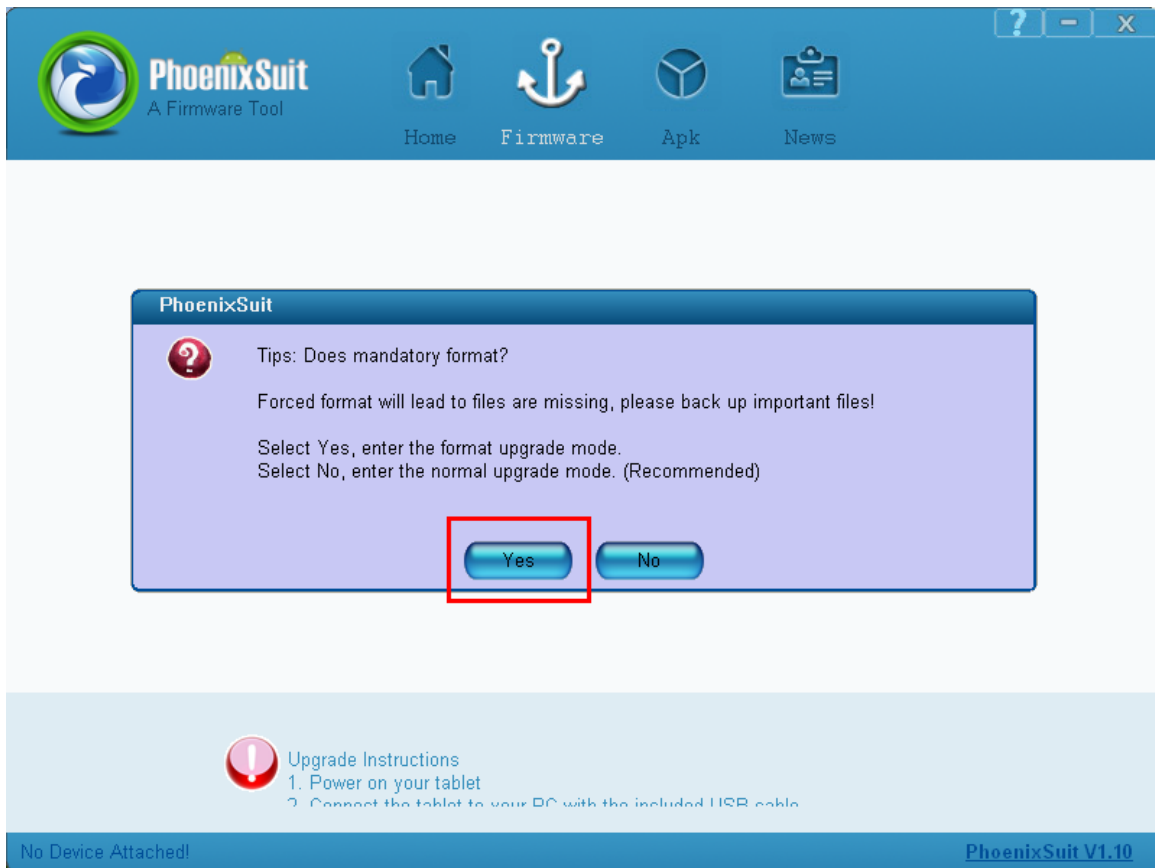


connect PC and board with Micro USB cable, keep pressing the **Recover Key** and power on until the windows PC pops up dialogue box.





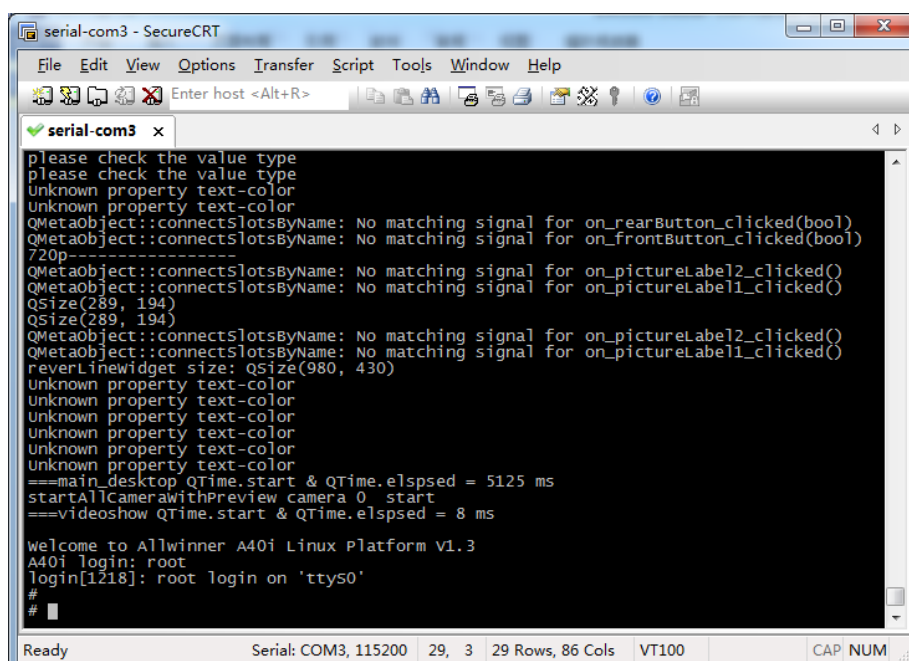
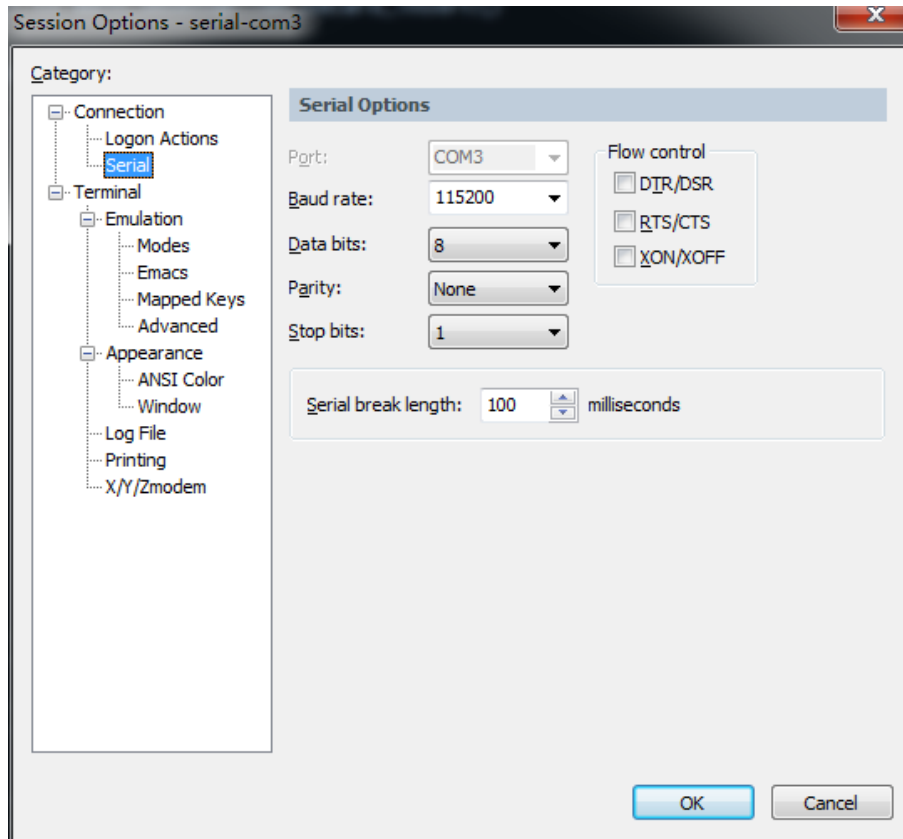
Select "Yes" to enter the format upgrade mode. After the system update is completed, it will restart automatically.



5 EMA40i Application Guidance

5.1 Serial Terminal

Connect the board and PC with USB Serial cable, then power on, the terminal will output startup information.



5.2 HDMI Display

Connect the board and HDMI monitor with a HDMI cable, then power on. After the board boot into Linux Qt, HDMI monitor will show QT UI.

5.3 LVDS and Touch

Connect LVDS to the board, then power on the board. After the board boot into Linux Qt, The LVDS will auto display QT UI.

EMA40i supports HDMI & LVDS display sync.

5.4 SD Card

The system will automatically mount the SD card.

ls /mnt/sdcard/mmcblk1p1/

```
#
#
# ls /mnt/sdcard/mmcblk1p1/test_file/
103056ok1popuc000j1g5u.jpg
Adobe_Flash_Player_10.3.185.24.apk
Hejie-jingguo.mp3
Taylor Swift - Sparks Fly.mp3
Track02.MP3
Zhangzhenyue-zaijian.mp3
[DLK00.com]B1A4 - OK (HD-1080P).mp4
big_buck_bunny_480p_h264.mov
com.mxtech.videoplayer.ad_2417.apk
endless love(480P).avi
endless love(480P).mp4
girls generation MV(720P).mp4
gpstestplus_22.apk
rk_sdmmc.c
test_3G.txt
????????? ?????????? - ??????????????????.flac
????????? - ???????[????????1080P].mp4
?????????-???????[????????1080P].mp4
#
#
```

5.5 USB Host

Insert USB device (e.g. U-disk) to USB Host, the system will automatically read the contents.

ls /mnt/usb/sda4/



```
#  
#  
# ls /mnt/usb/sda4/  
A18_PRO_v2  
A401  
A40i_v1.3_test.txt  
BCM43430B0.hcd  
RK3288_A17_test(debian).txt  
Sounds  
apkTest  
bcmhdh.ko  
brcm_patchram_plus  
com  
hciattach  
hciconfig  
hcidump  
hcidump  
hcidump  
ip  
jellyfish-250-mbps-4k-uhd-hevc-10bit.mkv  
mfgtools.zip  
rk3288-cd  
rk3288_A17  
rk3288_A18  
testFiles  
testRecords  
test_file  
zh1.jpg  
??????IP?????.txt  
#  
#  
#
```

5.6 SSD (SATA 2.0)

Note: SSD does not support hot plugging, please insert before power on.

Insert SSD mass storage device to SATA interface, then power the board when system booting it can auto mount.

```
# ls /mnt/usb/sda1/
```

5.7 EC20(4G)

Execute the follow commands to modify /etc/resolv.conf file:

```
#vi /etc/resolv.conf
```

Add:

```
nameserver 8.8.8.8
```

```
nameserver 192.168.0.2
```

```
# pppd call wcdma&
```

```
# ping www.boardcon.com
```



```
# pppd call wcdma&
[1] 1161
# abort on (NO CARRIER)
abort on (ERROR)
abort on (NO DIALTONE)
abort on (BUSY)
abort on (NO ANSWER)
send (AT^M)
expect (OK)
AT^M^M
OK
-- got it

send (ATZ^M)
expect (OK)
^M
ATZ^M^M
OK
-- got it

send (AT+CGDCONT=1,\"IP\", \"3gnet\", ,0,0^M)
expect (OK)
^M
AT+CGDCONT=1,\"IP\", \"3gnet\", ,0,0^M^M
OK
-- got it

send (AT+CFUN=1^M)
expect (OK)
^M
AT+CFUN=1^M^M
OK
-- got it

send (ATDT*99#^M)
expect (CONNECT)
^M
ATDT*99#^M^M
CONNECT
-- got it

send (^M)
Serial connection established.
Using interface ppp0
Connect: ppp0 <--> /dev/ttyUSB3
Could not determine remote IP address: defaulting to 10.64.64.64
local IP address 10.39.134.99
remote IP address 10.64.64.64
primary DNS address 120.80.80.80
secondary DNS address 221.5.88.88

# ping www.boardcon.com
PING www.boardcon.com (67.222.54.196): 56 data bytes
64 bytes from 67.222.54.196: seq=0 ttl=49 time=567.319 ms
64 bytes from 67.222.54.196: seq=1 ttl=49 time=475.986 ms
```

5.8 GPS

Execute the follow command to test GPS.

```
# cat /dev/ttyS4
```



```
#
#
# cat /dev/ttyS4
$GPRMC,,V,,,,,,,,,N*53
$GPVTG,,,,,,,,,N*30
$GPGGA,,,,,0,00,99.99,,,,,*48
$GPGSA,A,1,,,,,,,,,99.99,99.99,99.99*30
$GPGLL,,,,,V,N*64
$GPTXT,01,01,01,NMEA unknown msg*58
$GPTXT,01,01,01,NMEA unknown msg*58
$GPTXT,01,01,01,NMEA unknown msg*58
$GPTXT,01,01,01,NMEA unknown msg*58
$GPTXT,01,01,01,NMEA unknown msg*58
$GPTXT,01,01,01,NMEA unknown msg*58
$GPTXT,01,01,01,NMEA unknown msg*58
$GPTXT,01,01,01,NMEA unknown msg*58
$GPTXT,01,01,01,NMEA unknown msg*58
$GPTXT,01,01,01,NMEA unknown msg*58
$GPTXT,01,01,01,NMEA unknown msg*58
$GPTXT,01,01,01,NMEA unknown msg*58
$GPTXT,01,01,01,NMEA unknown msg*58
```

5.9 RTC(PCF8563)

#date -s "2019-05-28 15:20:00" (set the system time)

#hwclock -w

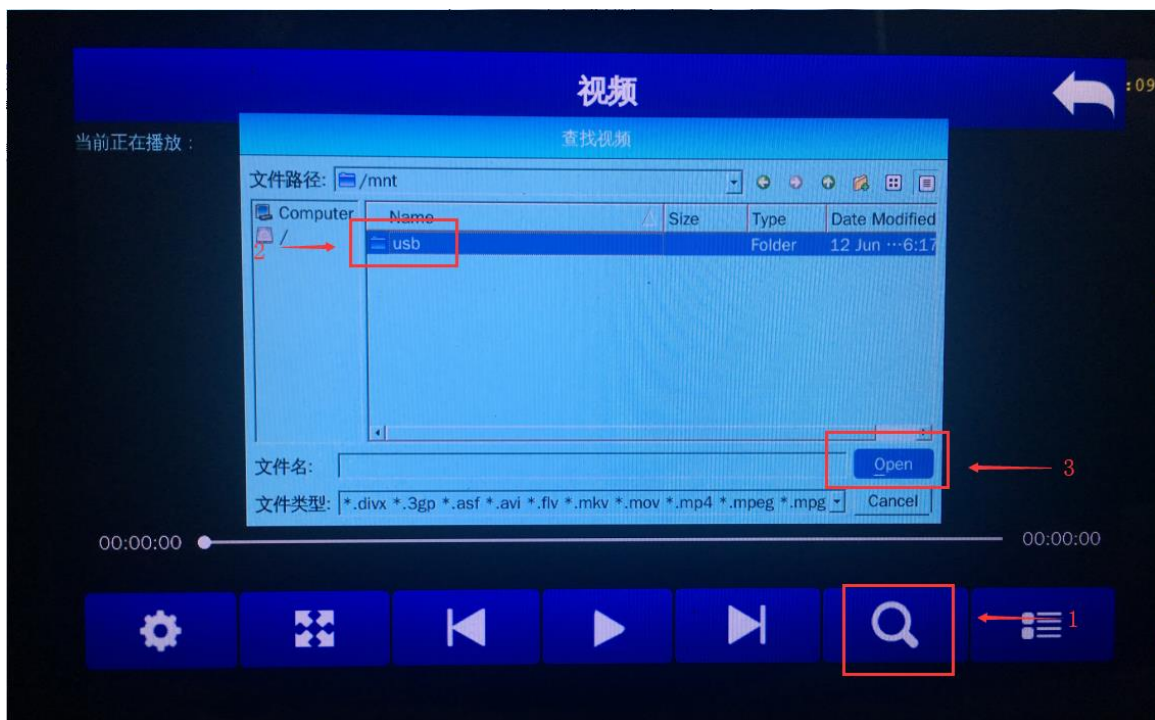
#hwclock

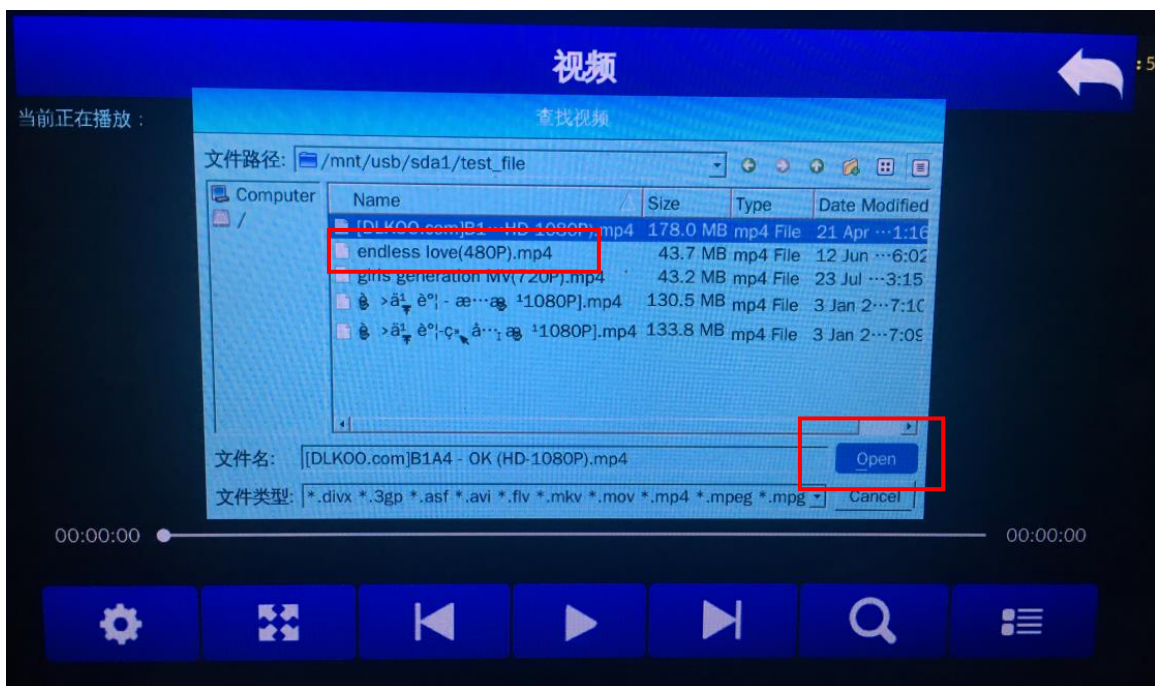
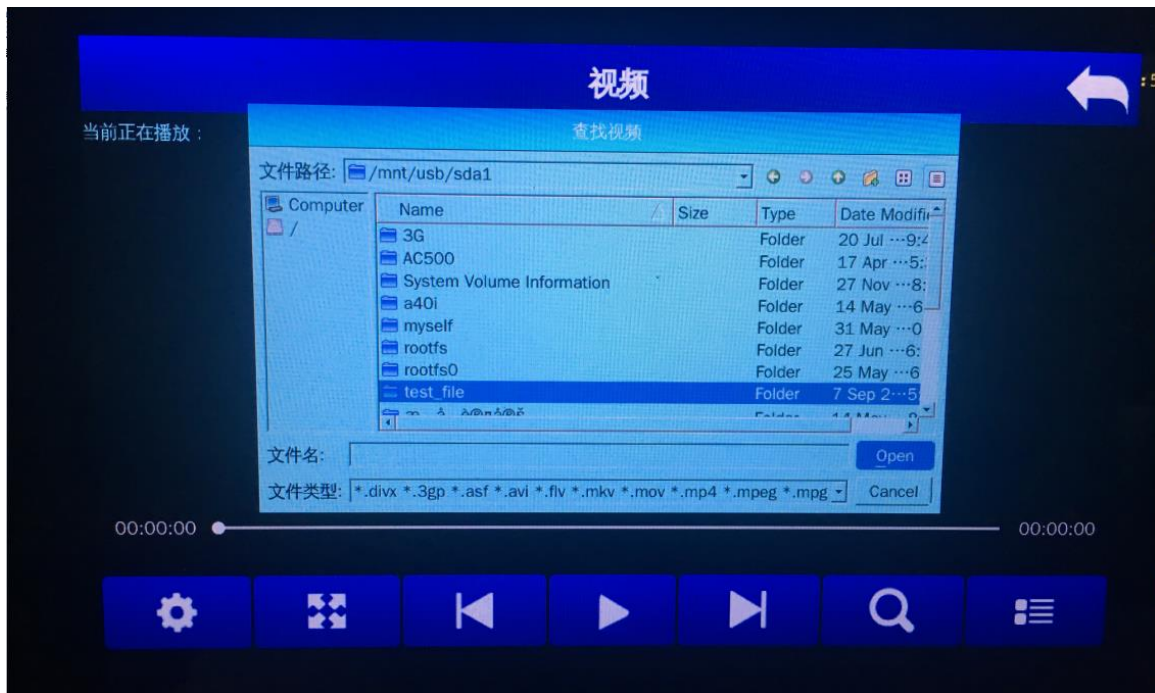
```
#
# date -s "2019-05-28 15:20:00"
Tue May 28 15:20:00 UTC 2019
# hwclock -w
# hwclock
Tue May 28 15:20:04 2019  0.000000 seconds
# hwclock
Tue May 28 15:20:10 2019  0.000000 seconds
# hwclock
Tue May 28 15:20:21 2019  0.000000 seconds
#
```

5.10 Video Player

5.10.1 Play Video

Put MP4 files to the SD card/U-disk and power on. Click the video player icon to play:







5.10.2 xplayerdemo

Cancel `/etc/init.d/runqt` in the `/etc/init.d/rcS` and reboot.

```
# cd /
```

```
# vi /etc/init.d/rcS //find out runqt, than cancel runqt.
```

```
# reboot
```

Restart and enter the root directory

```
# cd /
```

```
# chmod 755 /xplayerdemo
```

```
# /xplayerdemo
```

```
# set url: /mnt/usb/sda4/test_file/Xueziqian-1080P.mp4 (the directory of video)
```

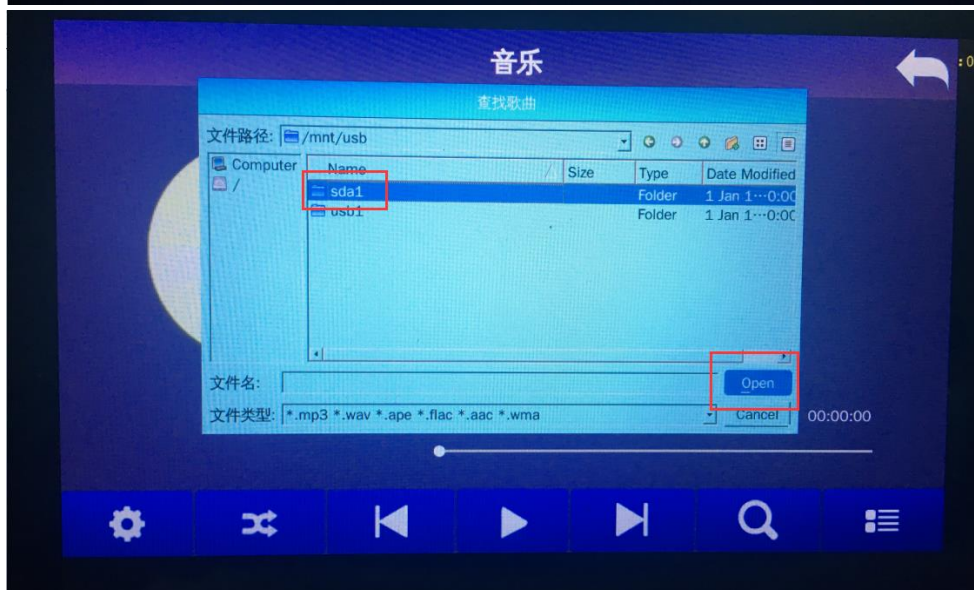
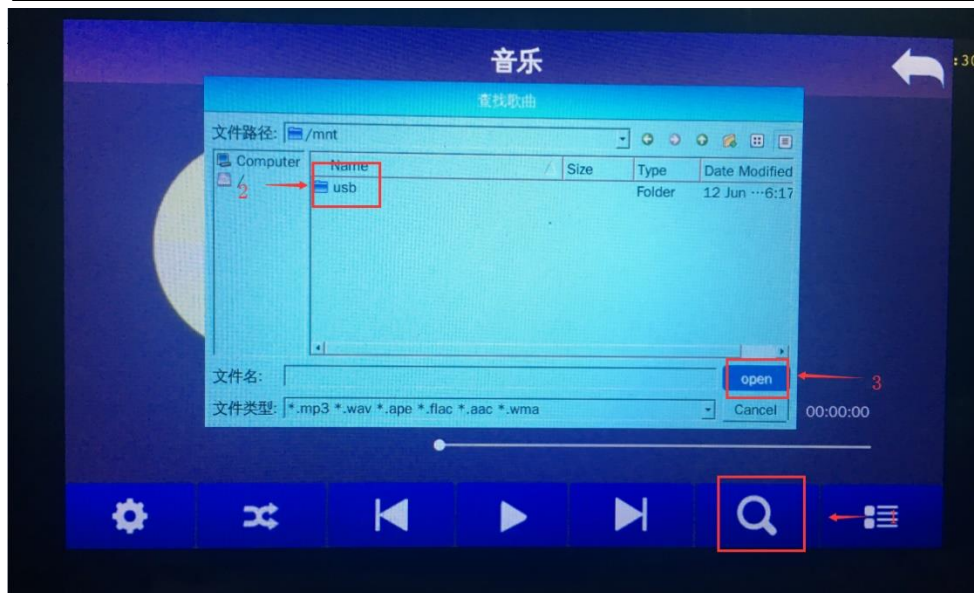
```
# show media info
```

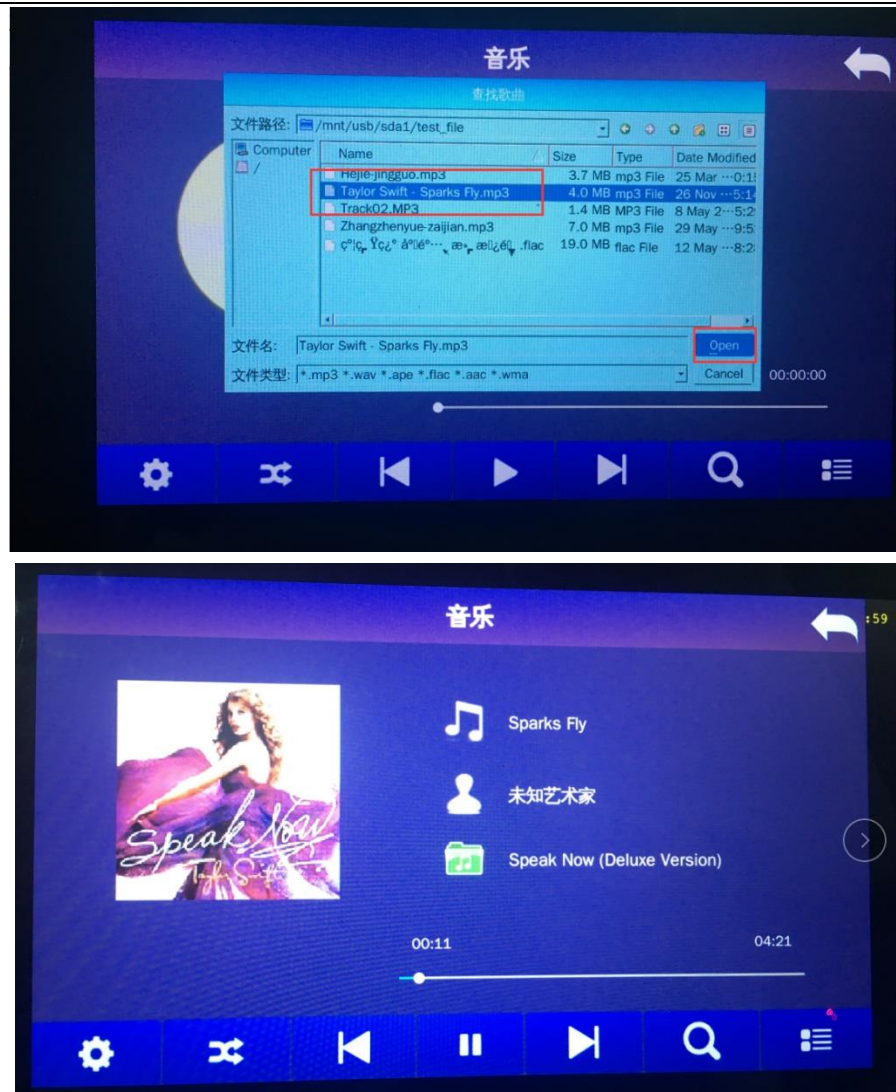
```
# play
```

```
# pause
```

5.11 Audio Player

Put MP3 files to the SD card/U-disk and power on. Click the Audio Player icon:





5.12 Recording

arecord -l

```
#
# arecord -l
**** List of CAPTURE Hardware Devices ****
card 0: audiocodec [audiocodec], device 0: SUNXI-CODEC sun8iw11codec-0 []
Subdevices: 0/1
Subdevice #0: subdevice #0
#
```

Cancel `/etc/init.d/runqt` in the `/etc/init.d/rcS` and reboot.

cd /

vi `/etc/init.d/rcS` //find out runqt, than cancel runqt. In front of the `/etc/init.d/runqt` add

reboot

Restart and enter the root directory

amixer cset numid=32 50

amixer cset numid=25 50

Record:

tinycap 123.wav -D "hw:0,0" -r 48000 -c 2 -f "S16_LE"

or

```
# arecord -D "hw:0,0" -r 48000 -c 2 -f "S16_LE" 123.wav
```

Play:

```
# aplay -D "hw:0,0" -f "S16_LE" 123.wav
```

or

```
# tinyplay 123.wav
```

```
#
#
#
# amixer cset numid=32 50
numid=32,iface=MIXER,name='Left Input Mixer MIC1 Boost Switch'
; type=BOOLEAN,access=rw-----,values=1
: values=on
#
# amixer cset numid=25 50
numid=25,iface=MIXER,name='Right Input Mixer MIC1 Boost Switch'
; type=BOOLEAN,access=rw-----,values=1
: values=on
#
# tinycap 123.wav -D "hw:0,0" -r 48000 -c 2 -f "S16_LE"
Capturing sample: 2 ch, 48000 hz, 16 bit
^CCaptured 778240 frames
#
# aplay -D "hw:0,0" -f "S16_LE" 123.wav
Playing WAVE '123.wav' : Signed 16 bit Little Endian, Rate 48000 Hz, Stereo
#
# tinyplay 123.wav
Playing sample: 2 ch, 48000 hz, 16 bit
#
#
```

5.13 Ethernet

Plug in an Ethernet cable (RJ45).

```
#ifconfig eth0 up
```

```
#udhcpc -i eth0 -R
```

```
#ping www.boardcon.com
```

```
#
# ifconfig eth0 up
[ 2082.620086] **** wyb drivers/net/ethernet/allwinner/sunxi-gmac.c-323-geth_power_on
[ 2082.800130] **** wyb drivers/net/ethernet/allwinner/sunxi-gmac.c-330-geth_power_on
[ 2082.822819] libphy: 1c50000.eth: probed
[ 2082.860365] sunxi-gmac 1c50000.eth eth0: eth0: Type(6) PHY ID 001cc915 at 1 IRQ poll (1c50000.eth-0:01)
# [ 2086.860383] libphy: 1c50000.eth-0:01 - Link is Up - 1000/Full

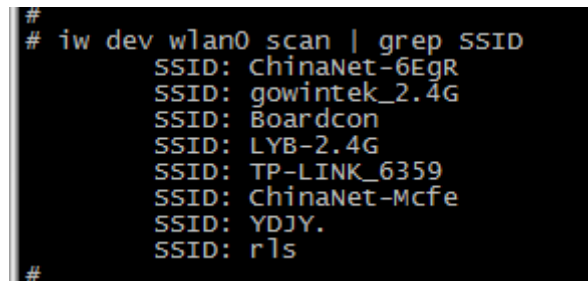
#
# udhcpc -i eth0 -R
udhcpc: started, v1.25.1
udhcpc: sending discover
udhcpc: sending select for 192.168.0.151
udhcpc: lease of 192.168.0.151 obtained, lease time 86400
deleting routers
adding dns 192.168.0.2
adding dns 0.0.0.0
#
# ping www.boardcon.com
PING www.boardcon.com (67.222.54.196): 56 data bytes
64 bytes from 67.222.54.196: seq=0 ttl=50 time=178.174 ms
64 bytes from 67.222.54.196: seq=1 ttl=50 time=180.551 ms
64 bytes from 67.222.54.196: seq=2 ttl=50 time=178.233 ms
64 bytes from 67.222.54.196: seq=3 ttl=50 time=179.809 ms
64 bytes from 67.222.54.196: seq=4 ttl=50 time=177.815 ms
64 bytes from 67.222.54.196: seq=5 ttl=50 time=179.892 ms
64 bytes from 67.222.54.196: seq=6 ttl=50 time=177.282 ms
64 bytes from 67.222.54.196: seq=7 ttl=50 time=179.488 ms
^C
--- www.boardcon.com ping statistics ---
8 packets transmitted, 8 packets received, 0% packet loss
round-trip min/avg/max = 177.282/178.905/180.551 ms
#
```

5.14 WiFi

Step 1, Execute the follow commands to search SSID.

```
# ifconfig wlan0 up
```

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



```
#
# iw dev wlan0 scan | grep SSID
    SSID: ChinaNet-6EgR
    SSID: gowintek_2.4G
    SSID: Boardcon
    SSID: LYB-2.4G
    SSID: TP-LINK_6359
    SSID: ChinaNet-Mcfe
    SSID: YDJY.
    SSID: rls
#
```

Step 2, Type the SSID and password from the list of available networks.

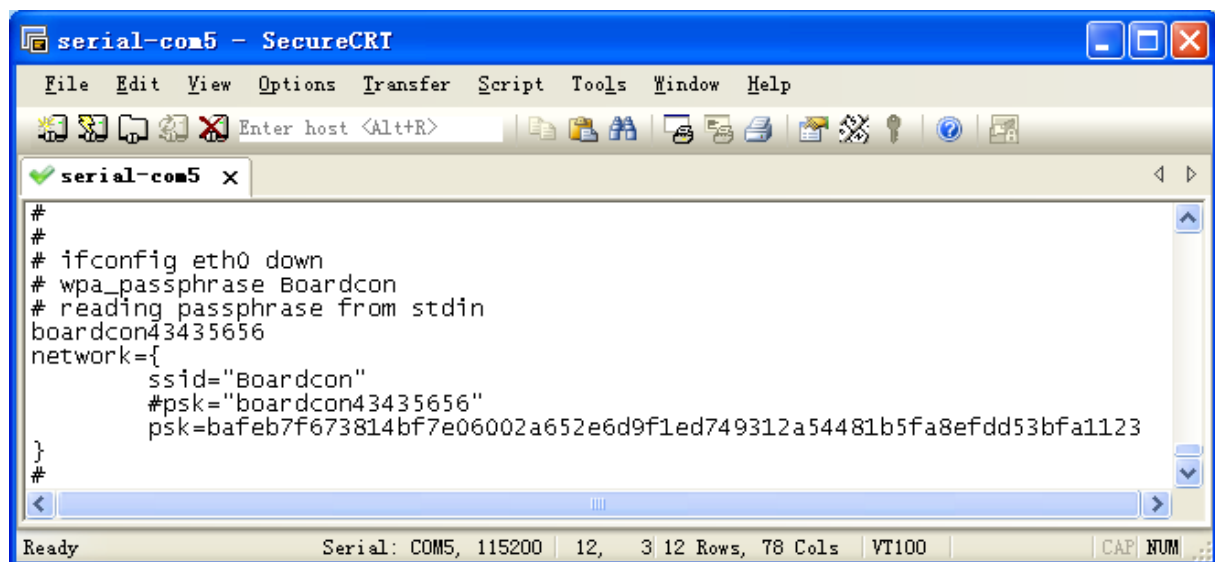
```
# ifconfig eth0 down
```

```
# wpa_passphrase Boardcon      (Boardcon is wifi name)
```

```
# reading passphrase from stdin
```

```
boardcon43435656              (boardcon43435656 is the password)
```

```
network={
    ssid="Boardcon"
    #psk="boardcon43435656"
    psk=bafeb7f673814bf7e06002a652e6d9f1ed749312a54481b5fa8efdd53bfa1123
}
```



Copy **ssid**(e.g. Boardcon) and **#psk**(e.g. boardcon43435656) replace ssid and psk in the directory of **/etc/wpa_supplicant.conf**.

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



```

#
#
#
# vi /etc/wpa_supplicant.conf
ctrl_interface=/var/run/wpa_supplicant
ctrl_interface_group=0
ap_scan=1
update_config=1
network={
    ssid="Boardcon"
    key_mgmt=WPA-PSK
    psk="boardcon43435656"
    priority=2
}
~/etc/wpa_supplicant.conf" 10L, 199C

```

Step 3, add DNS server.

#vi /etc/resolv.conf

In the resolv.conf file add follow command:

nameserver 8.8.8.8

nameserver 192.168.0.2 (gate way)

```

#
#
# cat /etc/resolv.conf
nameserver 8.8.8.8
nameserver 192.168.0.2
#
#

```

Step 4, Connect WiFi

Method 1

set the gateway

ifconfig wlan0 192.168.0.165 up

route add default gw 192.168.0.2 dev wlan0 (Must be add your Router gateway)

Execute the follow command to connect the **wpa_supplicant** tool.

wpa_supplicant -i wlan0 -Dnl80211 -c /etc/wpa_supplicant.conf -B

Method 2

Auto obtain wifi IP address:

Execute the follow commands to connect the **wpa_supplicant** tool.

wpa_supplicant -i wlan0 -Dnl80211 -c /etc/wpa_supplicant.conf -B

udhcpc -i wlan0

busybox apt-get install scrot

Step 5, ping

ping www.boardcon.com



```
# ifconfig wlan0 192.168.0.165 up
# route add default gw 192.168.0.2 dev wlan0
# wpa_supplicant -i wlan0 -Dnl80211 -c /etc/wpa_supplicant.conf -B
Successfully initialized wpa_supplicant
nl80211: could not re-add multicast membership for vendor events: -2 (No such file or directory)
[ 132.539137] P2P interface registered
[ 133.139611] WLC_E_IF: NO_IF set, event Ignored
[ 133.146242] P2P interface started
nl80211: could not re-add multicast membership for vendor events: -2 (No such file or directory)
# [ 134.326191] Connecting with 00:22:aa:4c:a0:d8 channel (6) ssid "Boardcon", len (8)
[ 134.326191]
[ 134.418212] wl_iw_event: Link up with BSSID=00:22:00:00:A0:D8
[ 134.424872] wl_bss_connect_done succeeded with 00:22:aa:4c:a0:d8
[ 134.432303] CFG80211-ERROR: wl_cfg80211_scan_abort: scan abort failed
[ 134.485260] wl_bss_connect_done succeeded with 00:22:aa:4c:a0:d8
[ 135.207383] nf_conntrack: automatic helper assignment is deprecated and it will be removed soon. Use the iptables CT target to attach helpers instead.
#
# ping www.boardcon.com
PING www.boardcon.com (67.222.54.196): 56 data bytes
64 bytes from 67.222.54.196: seq=0 ttl=50 time=237.582 ms
64 bytes from 67.222.54.196: seq=1 ttl=50 time=222.152 ms
64 bytes from 67.222.54.196: seq=2 ttl=50 time=189.069 ms
64 bytes from 67.222.54.196: seq=3 ttl=50 time=181.723 ms
64 bytes from 67.222.54.196: seq=4 ttl=50 time=197.851 ms
64 bytes from 67.222.54.196: seq=5 ttl=50 time=191.311 ms
64 bytes from 67.222.54.196: seq=6 ttl=50 time=183.388 ms
--C
--- www.boardcon.com ping statistics ---
7 packets transmitted, 7 packets received, 0% packet loss
round-trip min/avg/max = 181.723/200.439/237.582 ms
#
```

5.15 Bluetooth

```
# cd /
# source /qt5.9.sh
# openbt
# hciconfig hci0 up
# hciconfig hci0 piscan
# /usr/libexec/bluetooth/bluetoothd -C &
# /usr/libexec/bluetooth/obexd -a -d -n &
```

```
# hciconfig hci0 up
# hciconfig hci0 piscan
# hciconfig hci0 -a
hci0: Type: BR/EDR Bus: UART
BD Address: 11:22:33:44:55:66 ACL MTU: 1021:7 SCO MTU: 64:1
UP RUNNING PSCAN ISCAN
RX bytes:1139 acl:0 sco:0 events:57 errors:0
TX bytes:838 acl:0 sco:0 commands:57 errors:0
Features: 0xbf 0xfe 0xcf 0xfe 0xdb 0xff 0x7b 0x87
Packet type: DM1 DM3 DM5 DH1 DH3 DH5 HV1 HV2 HV3
Link policy: RSWITCH SNIFF
Link mode: SLAVE ACCEPT
Name: 'BCM43436 26MHz Class 1.5 Ampak AP6236'
Class: 0x000000
Service classes: Unspecified
Device class: Miscellaneous
HCI Version: 4.2 (0x8) Revision: 0x4d
LMP Version: 4.2 (0x8) Subversion: 0x410c
Manufacturer: Broadcom Corporation (15)

# /usr/libexec/bluetooth/bluetoothd -C &
[ ] 4635
# /usr/libexec/bluetooth/obexd -a -d -n &
[ ] 5606
# obexd[5696]: OBEX daemon 5.39
obexd[5696]: obexd/src/main.c:main() Entering main loop
obexd[5696]: obexd/src/manager.c:manager_init()
obexd[5696]: obexd/src/plugin.c:plugin_init() Loading builtin plugins
obexd[5696]: obexd/src/mimetype.c:obex_mime_type_driver_register() driver 0x6eb34 mimetype x-obex/folder-listing registered
obexd[5696]: obexd/src/mimetype.c:obex_mime_type_driver_register() driver 0x6eb70 mimetype x-obex/capability registered
obexd[5696]: obexd/src/mimetype.c:obex_mime_type_driver_register() driver 0x6ebac mimetype x-obex/folder-listing registered
obexd[5696]: obexd/src/mimetype.c:obex_mime_type_driver_register() driver 0x6ebf4 mimetype (null) registered
obexd[5696]: obexd/src/plugin.c:add_plugin() Plugin filesystem loaded
obexd[5696]: obexd/src/transport.c:obex_transport_driver_register() driver 0x6ec30 transport bluetooth registered
obexd[5696]: obexd/src/plugin.c:add_plugin() Plugin bluetooth loaded
obexd[5696]: obexd/src/service.c:obex_service_driver_register() driver 0x6ec90 service Nokia OBEX PC Suite Services registered
obexd[5696]: obexd/src/mimetype.c:obex_mime_type_driver_register() driver 0x6ec54 mimetype application/vnd.nokia-backup registered
obexd[5696]: obexd/src/plugin.c:add_plugin() Plugin pcsuite loaded
obexd[5696]: obexd/src/service.c:obex_service_driver_register() driver 0x6ece4 service Object Push server registered
obexd[5696]: obexd/src/plugin.c:add_plugin() Plugin opp loaded
obexd[5696]: obexd/src/service.c:obex_service_driver_register() driver 0x6ed44 service File Transfer server registered
obexd[5696]: obexd/src/plugin.c:add_plugin() Plugin ftp loaded
obexd[5696]: obexd/plugins/irmc.c:irmc_init()
obexd[5696]: obexd/src/mimetype.c:obex_mime_type_driver_register() driver 0x6edd4 mimetype (null) registered
obexd[5696]: obexd/src/service.c:obex_service_driver_register() driver 0x6ed8c service IRMC Sync server registered
obexd[5696]: obexd/src/plugin.c:add_plugin() Plugin irmc loaded
obexd[5696]: obexd/src/mimetype.c:obex_mime_type_driver_register() driver 0x6ee64 mimetype x-bt/phonebook registered
```

```
# hcitool scan
# bt-adaptor -d (search BT device, press CTRL+C to exit after the Bluetooth device appears)
```



```
# hcitool scan
Scanning ...
        CC:4B:73:04:D4:ED      linaro-alip
        04:B1:67:5B:F0:B7      llfyj0
#
#
# bt-adapter -d
Searching...
[A4:DA:32:D8:C2:3C]
  Name: (null)
  Alias: A4-DA-32-D8-C2-3C
  Address: A4:DA:32:D8:C2:3C
  Icon: (null)
  Class: 0x0
  LegacyPairing: 0
  Paired: 0
  RSSI: -100

[80:6F:B0:27:9C:C4]
  Name: (null)
  Alias: 80-6F-B0-27-9C-C4
  Address: 80:6F:B0:27:9C:C4
  Icon: (null)
  Class: 0x0
  LegacyPairing: 0
  Paired: 0
  RSSI: -103

[04:B1:67:5B:F0:B7]
  Name: llfyj0
  Alias: llfyj0
  Address: 04:B1:67:5B:F0:B7
  Icon: phone
  Class: 0x5a020c
  LegacyPairing: 0
  Paired: 0
  RSSI: -88

[CC:4B:73:04:D4:ED]
  Name: linaro-alip
  Alias: linaro-alip
  Address: CC:4B:73:04:D4:ED
  Icon: (null)
  Class: 0x1c0000
  LegacyPairing: 0
  Paired: 0
  RSSI: -70

^C
#
```

bt-device -c CC:4B:73:04:D4:ED (Connect Bluetooth. CC:4B:73:04:D4:ED is the address of the connecting Bluetooth device)

bt-obex -p CC:4B:73:04:D4:ED /opt/test.jpg (send file. /opt/test.jpg is the sent file)



```

bt-device -c CC:4B:73:04:D4:ED
Connect to CC:4B:73:04:D4:ED
Device: linaro-alip (cc:4b:73:04:d4:ed)
Confirm passkey: 871545 (yes/no)? yes
port
# bt-obex -p CC:4B:73:04:D4:ED /opt/test.jpg
obexd[13956]: obexd/client/session.c:obc_session_ref() 0x1b0f30: ref=2
obexd[13956]: obexd/client/bluetooth.c:bluetooth_connect() src 11:22:33:44:55:66 dest cc:4b:73:04:d4:ed service 00001105-0000-1000-8000-00805f9b34fb port 0
obexd[13956]: obexd/client/bluetooth.c:session_connect() session 0x1b1a98
obexd[13956]: obexd/client/bluetooth.c:service_connect()
obexd[13956]: obexd/client/session.c:obc_session_create() session 0x1b0f30 transport Bluetooth driver OPP
obexd[13956]: obexd/client/bluetooth.c:transport_connect() port 4105
obexd[13956]: obexd/client/bluetooth.c:transport_callback()
obexd[13956]: obexd/client/session.c:transport_func()
obexd[13956]: obexd/client/bluetooth.c:bluetooth_getpacketopt()
obexd[13956]: obexd/client/opp.c:opp_probe() /org/bluez/obex/client/session0
obexd[13956]: obexd/client/session.c:obc_session_ref() 0x1b0f30: ref=3
obexd[13956]: obexd/client/session.c:obc_session_register() session(0x1b0f30) registered /org/bluez/obex/client/session0
obexd[13956]: obexd/client/session.c:obc_session_unref() 0x1b0f30: ref=2
obexd[13956]: obexd/client/transfer.c:obc_transfer_register() 0x1b2708 registered /org/bluez/obex/client/session0/transfer0
obexd[13956]: obexd/client/session.c:obc_session_ref() 0x1b0f30: ref=3
obexd[13956]: obexd/client/session.c:obc_session_ref() 0x1b0f30: ref=4
obexd[13956]: obexd/client/session.c:session_process_transfer() Transfer(0x1b2708) started
obexd[13956]: obexd/client/session.c:obc_session_unref() 0x1b0f30: ref=3
[Transfer#test.jpg] waiting
[Transfer#test.jpg] Progress: 8%obexd[13956]: obexd/client/session.c:session_notify_complete() Transfer(0x1b2708) complete
obexd[13956]: obexd/client/session.c:obc_session_ref() 0x1b0f30: ref=4
obexd[13956]: obexd/client/transfer.c:obc_transfer_unregister() 0x1b2708 unregistered /org/bluez/obex/client/session0/transfer0
obexd[13956]: obexd/client/session.c:obc_session_unref() 0x1b0f30: ref=3
obexd[13956]: obexd/client/session.c:obc_session_unref() 0x1b0f30: ref=2
100%
[Transfer#test.jpg] completed
(bt-obex:21971): GLib-CRITICAL **: g_variant_unref: assertion 'value->ref_count > 0' failed
obexd[13956]: obexd/client/session.c:owner_disconnected()
obexd[13956]: obexd/client/session.c:obc_session_shutdown() 0x1b0f30
obexd[13956]: obexd/client/session.c:obc_session_ref() 0x1b0f30: ref=3
obexd[13956]: obexd/client/opp.c:opp_remove() /org/bluez/obex/client/session0
obexd[13956]: obexd/client/session.c:obc_session_unref() 0x1b0f30: ref=2
obexd[13956]: obexd/client/session.c:obc_session_unref() 0x1b0f30: ref=1
obexd[13956]: obexd/client/session.c:session_unregister() Session(0x1b0f30) unregistered /org/bluez/obex/client/session0
obexd[13956]: obexd/client/session.c:obc_session_unref() 0x1b0f30: ref=0
obexd[13956]: obexd/client/session.c:disconnect_complete()
obexd[13956]: obexd/client/bluetooth.c:bluetooth_disconnect()
obexd[13956]: obexd/client/bluetooth.c:session_destroy() 0x1b1a98
obexd[13956]: obexd/client/session.c:session_free() 0x1b0f30
#
#

```

bt-obex -s /opt/ (Receive file. Default received path: ~root/.cache/obexd/)

```

# bt-obex -s /opt/
obexd[13956]: obexd/src/manager.c:register_agent() Agent registered
obexd[13956]: obexd/plugins/bluetooth.c:profile_new_connection() device /org/bluez/hci0/dev_CC_4B_73_04_D4_ED
obexd[13956]: obexd/src/obex.c:obex_session_start()
obexd[13956]: obexd/src/obex.c:cmd_connect()
obexd[13956]: CONNECT(0x0), (null)(0xffffffff)
obexd[13956]: obexd/src/obex.c:cmd_connect() Selected driver: Object Push server
obexd[13956]: CONNECT(0x0), (null)(0x0)
[OBEX Server] OBEX session opened
[OBEX Server] Transfer started
obexd[13956]: obexd/src/obex.c:cmd_put()
obexd[13956]: PUT(0x2), (null)(0xffffffff)
obexd[13956]: obexd/src/obex.c:parse_name() NAME: 00.jpg
obexd[13956]: obexd/src/obex.c:parse_length() LENGTH: 101392
obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=637
obexd[13956]: obexd/src/obex.c:driver_write() 637 written
obexd[13956]: PUT(0x2), CONTINUE(0x10)
[OBEXTransfer] Progress: 0%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
1%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
1%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
2%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
3%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
3%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
4%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
5%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written

```




```
90%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
91%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
91%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
92%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
93%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
93%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
94%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
95%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
95%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
96%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
97%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
97%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
98%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
99%obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=666
obexd[13956]: obexd/src/obex.c:driver_write() 666 written
obexd[13956]: obexd/src/obex.c:recv_data() name=00.jpg type=(null) file=0x7 size=189
obexd[13956]: obexd/src/obex.c:driver_write() 189 written
obexd[13956]: obexd/src/obex.c:transfer_complete()
0%[OBEX Server] Transfer succeeded
obexd[13956]: obexd/src/obex.c:cmd_disconnect() session 0x1b1160
obexd[13956]: DISCONNECT(0x1), (null)(0xffffffff)
obexd[13956]: DISCONNECT(0x1), SUCCESS(0x20)
obexd[13956]: disconnected: Transport got disconnected
obexd[13956]: obexd/src/obex.c:obex_session_destroy()
[OBEX Server] OBEX transfer closed
[OBEX Server] OBEX session closed
AC** Message: SIGINT received
obexd[13956]: obexd/src/manager.c:unregister_agent() Agent unregistered
#
# ls -l ~/root/.cache/obexd/
#
#
#
```

5.16 3D

Execute the follow command to run 3D:

```
# glmark2-es2-fbdev
```

Return to the desktop after execute the command

```
# debug /Launcher
```

CTRL+C to exit.

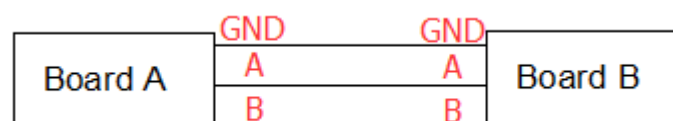


```
# glmark2-es2-fbdev
=====
glmark2 2014.03
=====
OpenGL Information
GL_VENDOR:    ARM
GL_RENDERER:  Mali-400 MP
GL_VERSION:   OpenGL ES 2.0
=====
[build] use-vbo=false: FPS: 63 FrameTime: 15.873 ms
[build] use-vbo=true: FPS: 63 FrameTime: 15.873 ms
[texture] texture-filter=nearest: FPS: 63 FrameTime: 15.873 ms
[texture] texture-filter=linear: FPS: 63 FrameTime: 15.873 ms
[texture] texture-filter=mipmap: FPS: 63 FrameTime: 15.873 ms
[shading] shading-gouraud: FPS: 63 FrameTime: 15.873 ms
[shading] shading-blinn-phong-inf: FPS: 63 FrameTime: 15.873 ms
[shading] shading-phong: FPS: 63 FrameTime: 15.873 ms
[shading] shading-cel: FPS: 63 FrameTime: 15.873 ms
[bump] bump-render=high-poly: FPS: 63 FrameTime: 15.873 ms
[bump] bump-render=normals: FPS: 63 FrameTime: 15.873 ms
[bump] bump-render=height: FPS: 63 FrameTime: 15.873 ms
libpng warning: icCP: known incorrect sRGB profile
[effect2d] kernel=0,1,0;1,-4,1;0,1,0: FPS: 60 FrameTime: 16.667 ms
libpng warning: icCP: known incorrect sRGB profile
[effect2d] kernel=1,1,1,1;1,1,1,1;1,1,1,1: FPS: 26 FrameTime: 38.462 ms
[pulsar] light=false:quads=5:texture=false: FPS: 62 FrameTime: 16.129 ms
libpng warning: icCP: known incorrect sRGB profile
[desktop] blur-radius=5:effect=blur:passes=1:separable=true:windows=4: FPS: 23 FrameTime: 43.478 ms
libpng warning: icCP: known incorrect sRGB profile
[desktop] effect-shadowwindows=4: FPS: 63 FrameTime: 15.873 ms
[buffer] columns=200:interleave=false:update-dispersion=0.9:update-fraction=0.5:update-method-map: FPS: 24 FrameTime: 41.667 ms
[buffer] columns=200:interleave=true:update-dispersion=0.9:update-fraction=0.5:update-method-subdata: FPS: 51 FrameTime: 19.608 ms
[ideas] speed=duration: FPS: 63 FrameTime: 15.873 ms
[jellyfish] <default>: FPS: 63 FrameTime: 15.873 ms
Error: SceneTerrain requires Vertex Texture Fetch support, but GL_MAX_VERTEX_TEXTURE_IMAGE_UNITS is 0
[terrain] <default>: unsupported
[shadow] <default>: FPS: 63 FrameTime: 15.873 ms
[refract] <default>: FPS: 20 FrameTime: 50.000 ms
[conditionals] fragment-steps=0:vertex-steps=0: FPS: 63 FrameTime: 15.873 ms
[conditionals] fragment-steps=5:vertex-steps=0: FPS: 63 FrameTime: 15.873 ms
[conditionals] fragment-steps=0:vertex-steps=5: FPS: 63 FrameTime: 15.873 ms
[function] fragment-complexity=low:fragment-steps=5: FPS: 63 FrameTime: 15.873 ms
[function] fragment-complexity=medium:fragment-steps=5: FPS: 63 FrameTime: 15.873 ms
[loop] fragment-loop=false:fragment-steps=5:vertex-steps=5: FPS: 63 FrameTime: 15.873 ms
[loop] fragment-steps=5:fragment-uniform=false:vertex-steps=5: FPS: 63 FrameTime: 15.873 ms
[loop] fragment-steps=5:fragment-uniform=true:vertex-steps=5: FPS: 63 FrameTime: 15.873 ms
=====
glmark2 Score: 56
=====
#
```



5.17 RS485

Connect the RS485 ports of Board A and B with the test line.



For Board A, execute the follow commands to set RS485 as Receiver.

```
# echo 49 > /sys/class/gpio/export
# echo out > /sys/class/gpio/gpio49/direction
# echo 1 > /sys/class/gpio/gpio49/value    (Set high for receiving.)
# com /dev/ttyS7 115200 8 0 1
```

For Board B, execute the follow commands to set RS485 as Transmitter.

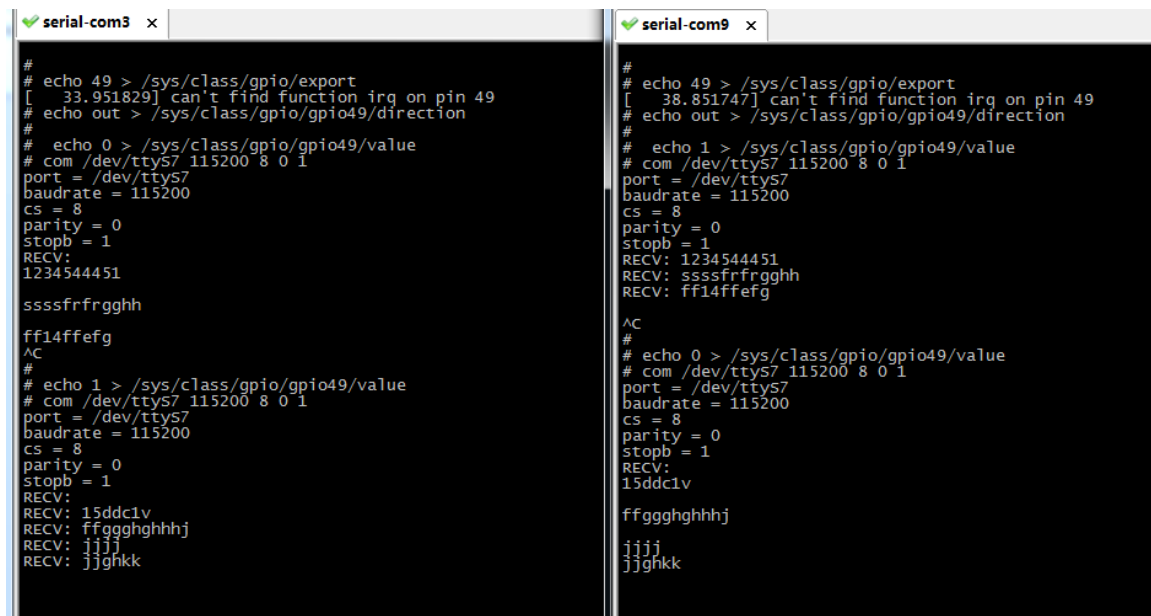
```
# echo 49 > /sys/class/gpio/export
# echo out > /sys/class/gpio/gpio49/direction
# echo 0 > /sys/class/gpio/gpio49/value    (Set low for sending.)
# com /dev/ttyS7 115200 8 0 1
```

The Transmitter and receiver can be converted by execute the command

```
# echo 1 > /sys/class/gpio/gpio49/value    (Set high for receiving.)
# com /dev/ttyS7 115200 8 0 1
```

Or

```
# echo 0 > /sys/class/gpio/gpio49/value    (Set low for sending.)
# com /dev/ttyS7 115200 8 0 1
```



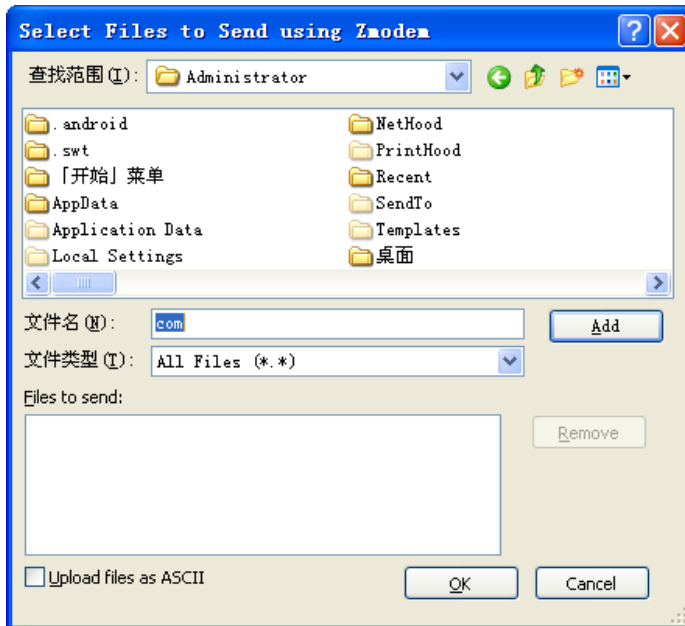
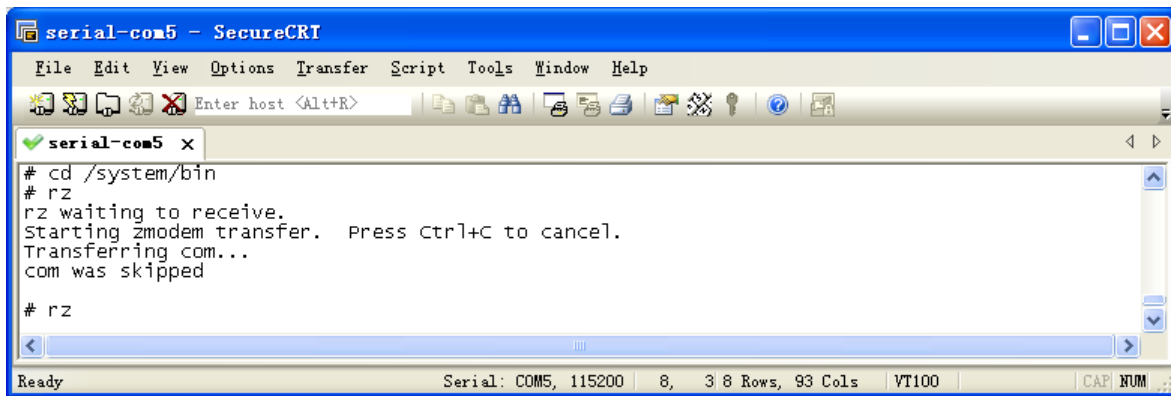
```
serial-com3 x
#
# echo 49 > /sys/class/gpio/export
[ 33.951829] can't find function irq on pin 49
# echo out > /sys/class/gpio/gpio49/direction
#
# echo 0 > /sys/class/gpio/gpio49/value
# com /dev/ttyS7 115200 8 0 1
port = /dev/ttyS7
baudrate = 115200
cs = 8
parity = 0
stopb = 1
RECV:
1234544451
ssssfrfrgghh
ff14ffefg
^C
#
# echo 1 > /sys/class/gpio/gpio49/value
# com /dev/ttyS7 115200 8 0 1
port = /dev/ttyS7
baudrate = 115200
cs = 8
parity = 0
stopb = 1
RECV:
15ddc1v
RECV: ffggghghhj
RECV: jjjj
RECV: jjghkk

serial-com9 x
#
# echo 49 > /sys/class/gpio/export
[ 38.851747] can't find function irq on pin 49
# echo out > /sys/class/gpio/gpio49/direction
#
# echo 1 > /sys/class/gpio/gpio49/value
# com /dev/ttyS7 115200 8 0 1
port = /dev/ttyS7
baudrate = 115200
cs = 8
parity = 0
stopb = 1
RECV: 1234544451
RECV: sssfrfrgghh
RECV: ff14ffefg
^C
#
# echo 0 > /sys/class/gpio/gpio49/value
# com /dev/ttyS7 115200 8 0 1
port = /dev/ttyS7
baudrate = 115200
cs = 8
parity = 0
stopb = 1
RECV:
15ddc1v
ffggghghhj
jjjj
jjghkk
```

5.18 RS232&UART (UART5 / UART6 / UART4 / UART7)

Execute the commands as follow:

```
#cd /system/bin
#rz          (send "com" file)
#chmod 777 com
#cd /
```



Connect RX&TX, then execute the commands to run the test program at serial terminal

```
# com /dev/ttyS5 115200 8 0 1 (Connect COM2 PIN2&3 )
# com /dev/ttyS6 115200 8 0 1 (Connect RS232 PIN1&2 )
# com /dev/ttyS4 115200 8 0 1 (Connect UART4 PIN1&2 )
# com /dev/ttyS7 115200 8 0 1 (Connect UART7 PIN1&2 )
```

```
#
# com /dev/ttys5 115200 8 0 1
port = /dev/ttys5
baudrate = 115200
cs = 8
parity = 0
stopb = 1
dddghhh
RECV: dddghhh
jjjkkrrr
RECV: jjjkkrrr
uuu66666
RECV: uuu66666
^C
#
# com /dev/ttys6 115200 8 0 1
port = /dev/ttys6
baudrate = 115200
cs = 8
parity = 0
stopb = 1
1234567890
RECV: 1234567890
ccccccghj
RECV: cccccghj
ccccccdsdy
RECV: cccccdsdy
^C
#
```