# EMA40i Linux Reference User Manual

V2.1.20220517

Boardcon Embedded Design

www.boardcon.com

# Colophon

| Ver | Description | Author | Date |
|---|---|---|---|
| V1.0 | Initial version | Wei Xianya | 2019-05-27 |
| V2.0 | Modify testing | Zhou Lijun | 2019-09-06 |
| V2.1 | Modify testing | Zhou Lijun | 2022-05-17 |

## 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....4
2. Compiler Environment....5
2.1 Vmware10.0+ubuntu16.04....5
2.2 Install Tools....5
3. Compile the Source....6
3.1 Unzip & Configuration....6
3.2 Compile Uboot....7
3.3 Compile Kernel....7
3.4 Compile Buildroot....7
3.5 Compile QT....8
3.5.1 Compile QT Library....8
3.5.2 Compile QT app demo....8
3.6 Compile sdk_lib Library....9
3.7 Pack Firmware....9
3.8 Automatic Compilation....10
4. Install Tools....10
4.1 Install CP2102 Driver....10
4.2 Install Serial Terminal Tool....11
5. Burning Guide....13
5.1 Burning Introduction....13
5.2 Burn image to eMMC via USB....13
6 EMA40i Application Guidance....16
6.1 Serial Terminal....16
6.2 HDMI Display....17
6.3 LVDS and Touch....17
6.4 SD Card....17
6.5 USB Host....17
6.6 SSD（SATA 2.0）....18
6.7 EC20(4G)....18
6.8 GPS....19
6.9 RTC(PCF8563)....20
6.10 Video Player....20
6.10.1 Play Video....20
6.10.2 xplayerdemo....23
6.11 Audio Player....23
6.12 Recording....26
6.13 Ethernet....27
6.14 WiFi....27
6.15 Bluetooth....30
6.16 3D....33
6.17 RS485....34
6.18 RS232&UART (UART5 / UART6 / UART4 / UART7)....35

# 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**

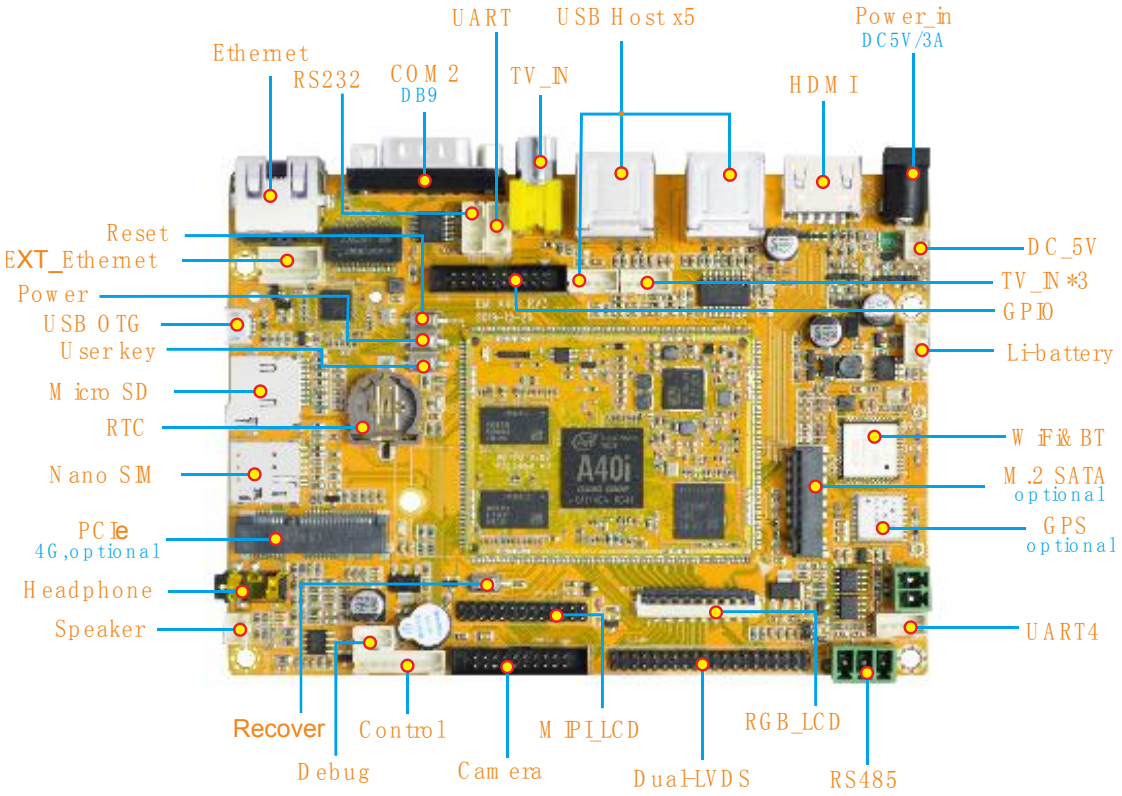
Integrated audio codec with 24 bit/192kHz DAC playback, and supports I2S/PCM interface for connecting to an external audio codec. I2C/PCM interface includes eight channels of TDM with sampling precision up to 32 bit/192kHz.

- **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, 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 complied 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
```

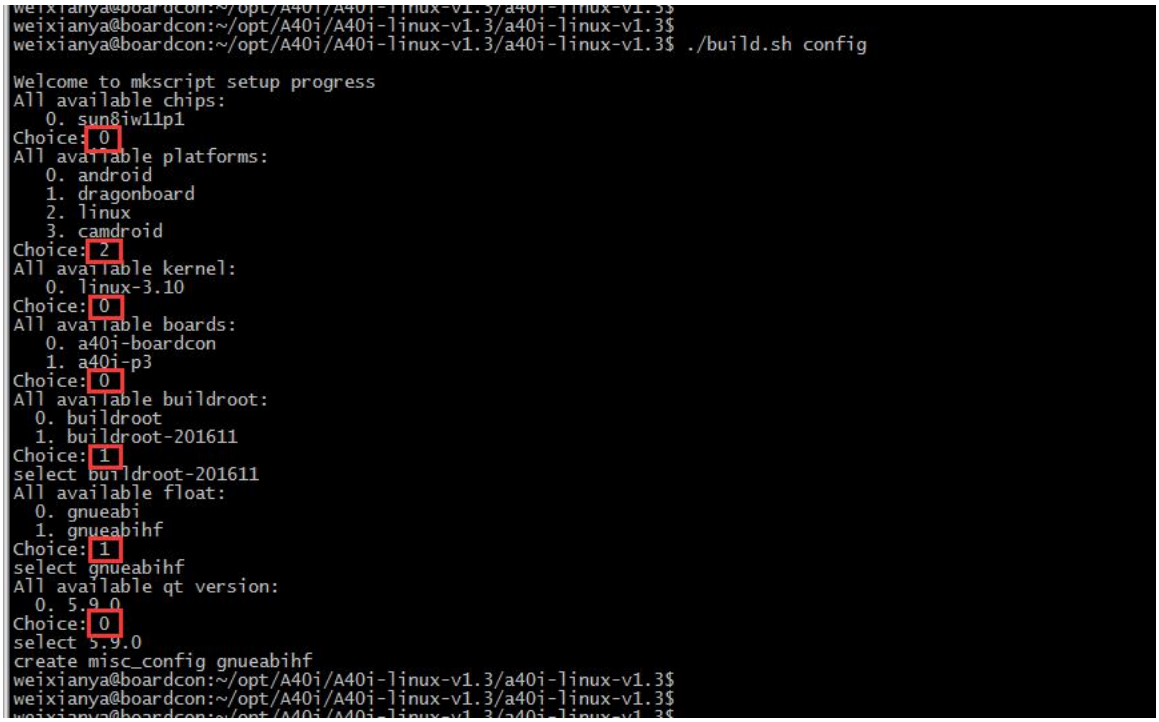
If the compilation encountered error, please according to the error message, the installation of the corresponding software package

# 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




***Step 3***, compile
$ ./build.sh

## 3.2 Compile Uboot

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

## 3.4 Compile Buildroot

If you need to 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 configh 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 disclean before compilation.

- **Compile qt application demo separately**

Enter a single demo directory and execute the following commands:
$cd CameraUI/
$./makeCameraUI

$cd Launcher/
$./makeLauncher

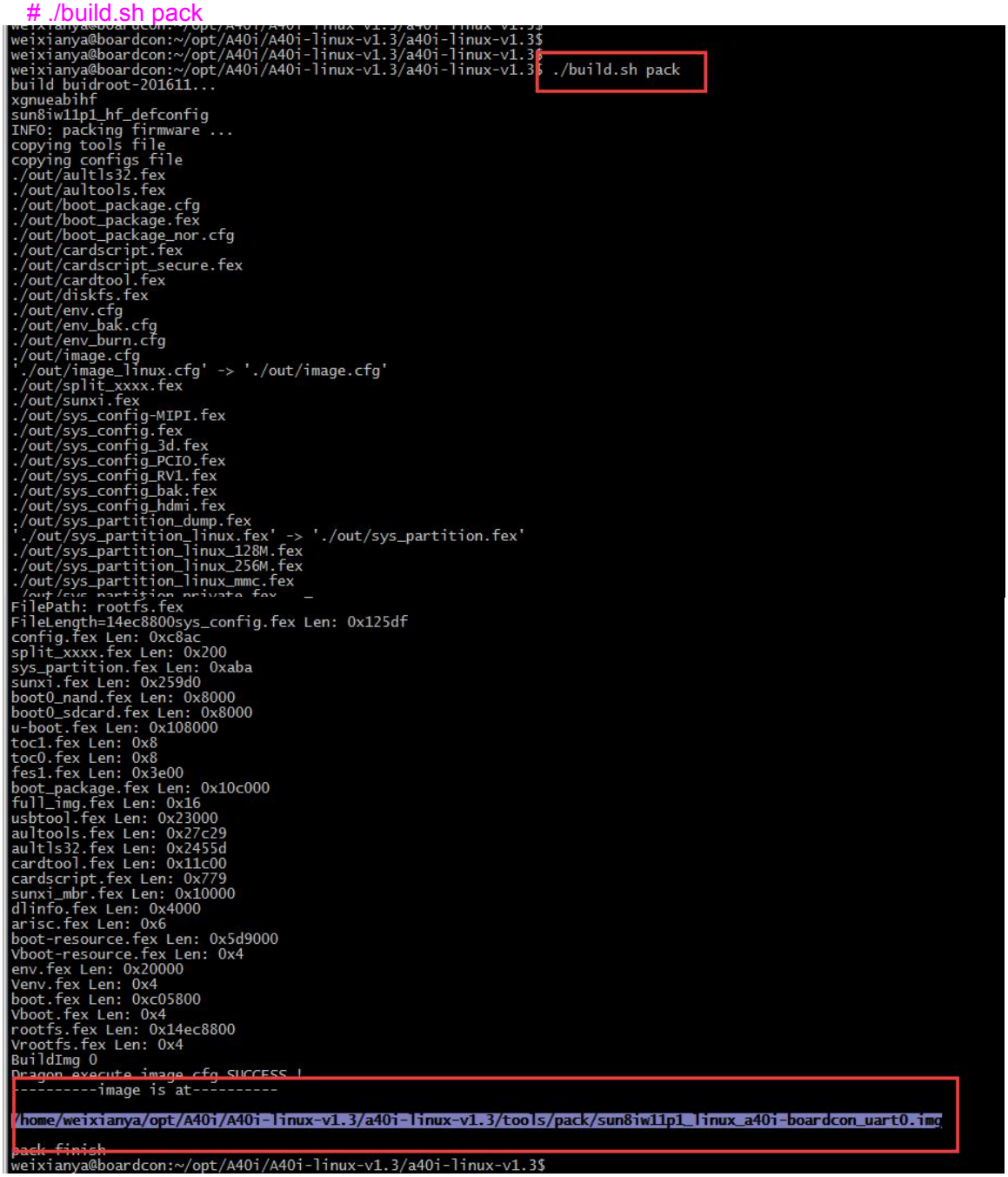
$cd MediaUI/
$./makeMediaUI

## 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




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$ ./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$
```

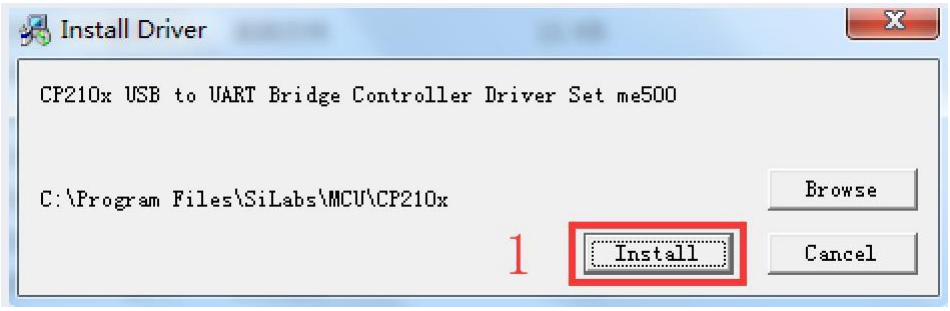
$./build.sh
$./comp_qtLib-590_only.sh
$./build.sh pack

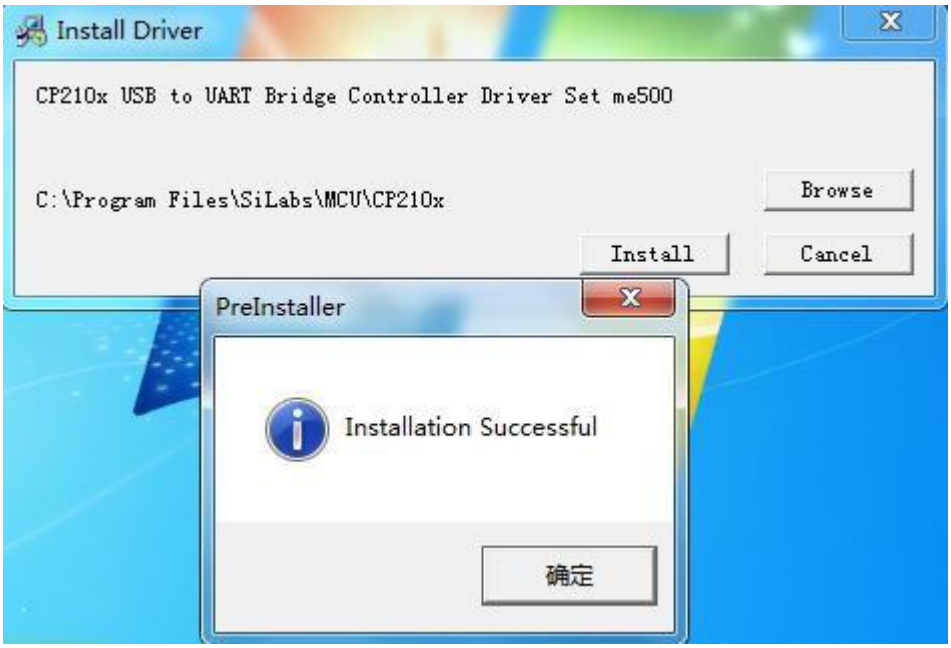
comp_qtLib-590_only.sh is The compilation of QT library，you can see in the directory of **a40i-linux-v1.3\comp_qtLib-590_only.sh**.

# 4. Install Tools

## 4.1 Install CP2102 Driver

Plug the **USB-to-UART cable CP2102** to the PC, unzip **CP2102WIN7.rar** on Windows, then click **preInstaller.exe** to install



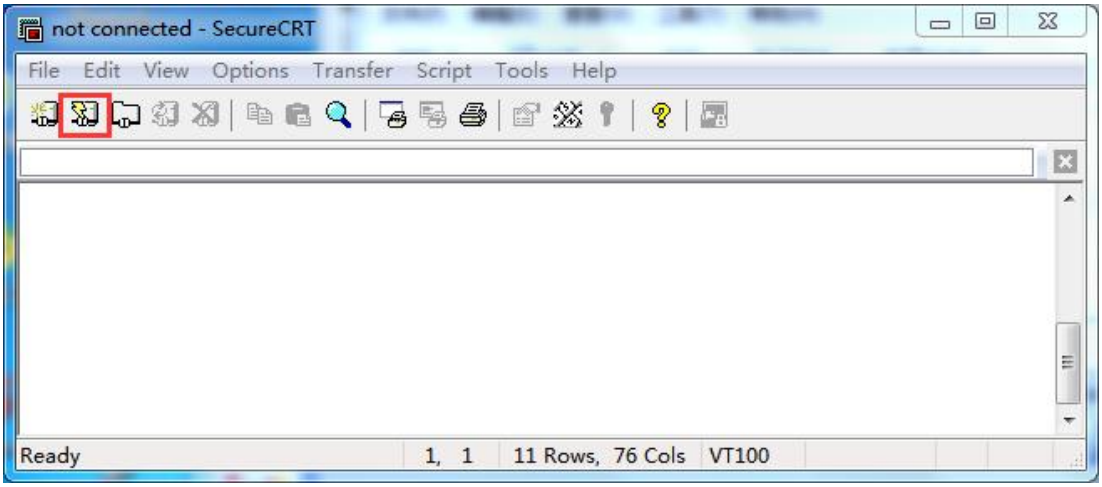



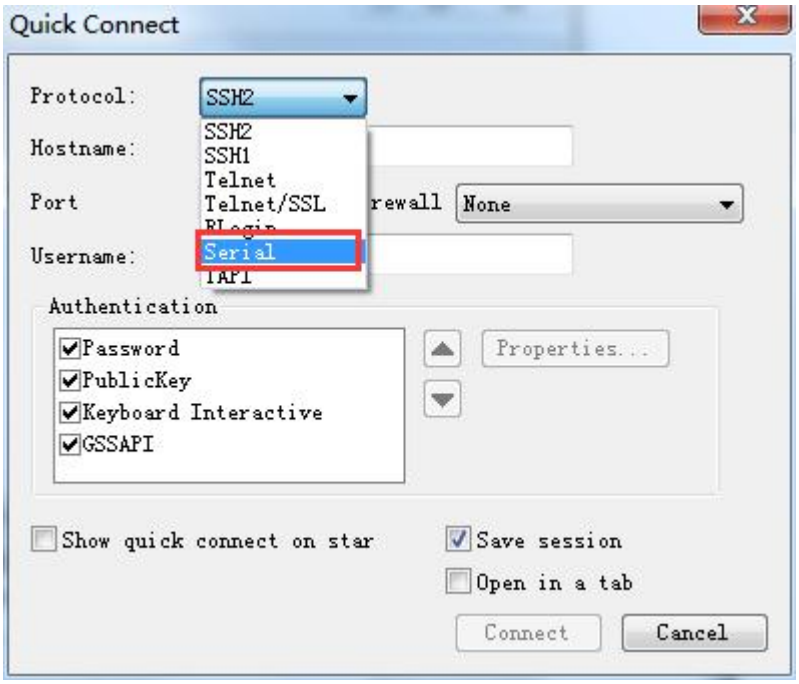

Now the device will be listed under **Device Manager -> PORTS** with unique serial port assigned.

## 4.2 Install Serial Terminal Tool

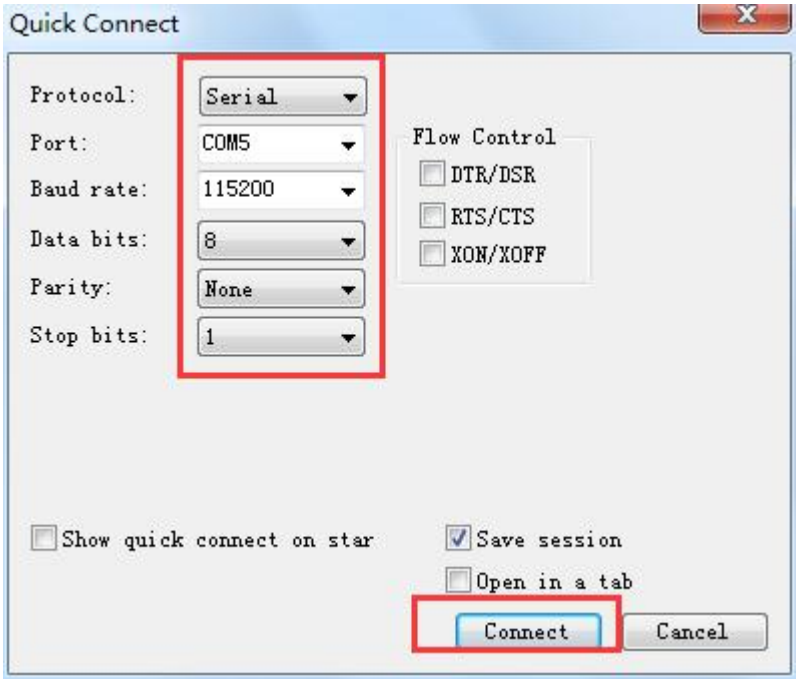
The serial terminal SecureCRT is used for debugging. It can be used directly after decompression.

Open SecureCRT.exe after copy to PC (path: tools\windows\SecureCRT.exe), then click the icon **Quick Connect** to config.






Set the parameters as follow:
**Protocol**: Serial
**Port**: To be specified by user PC
**Baud rate**: 115200
Please check XON/XOFF and RTS/CTS
Check Save session




After all, click **connect**
Illusion: If open more than one serial terminal tools, and they use the same serial port, there will be reported **the port is busy**.
**Solution**: Turn off the serial tool that unnecessary.

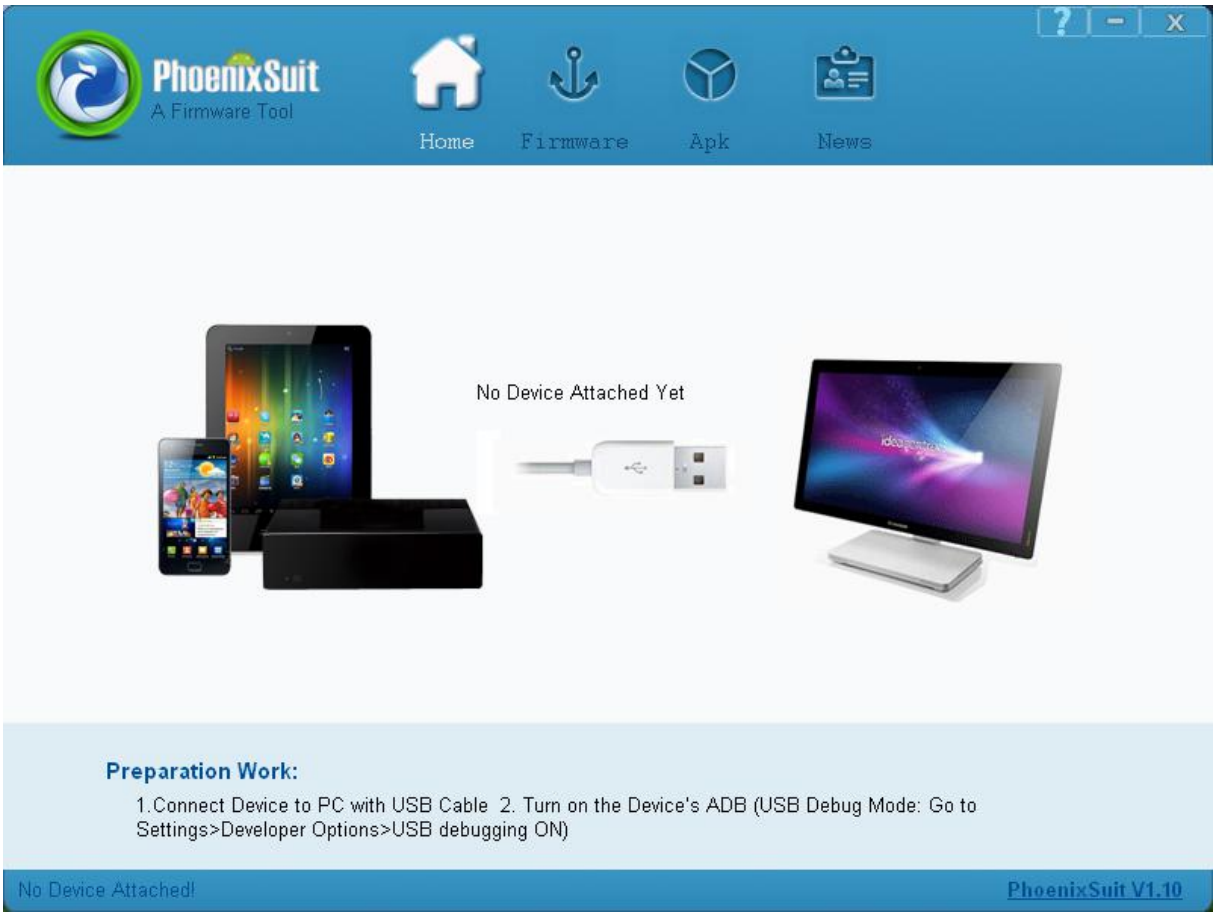
# 5. Burning Guide

## 5.1 Burning Introduction

Before burning, please install the corresponding tools:
Burn image to eMMC via USB: CD\Tools\PhoenixSuit1.0.6. Please installation according to the user system.

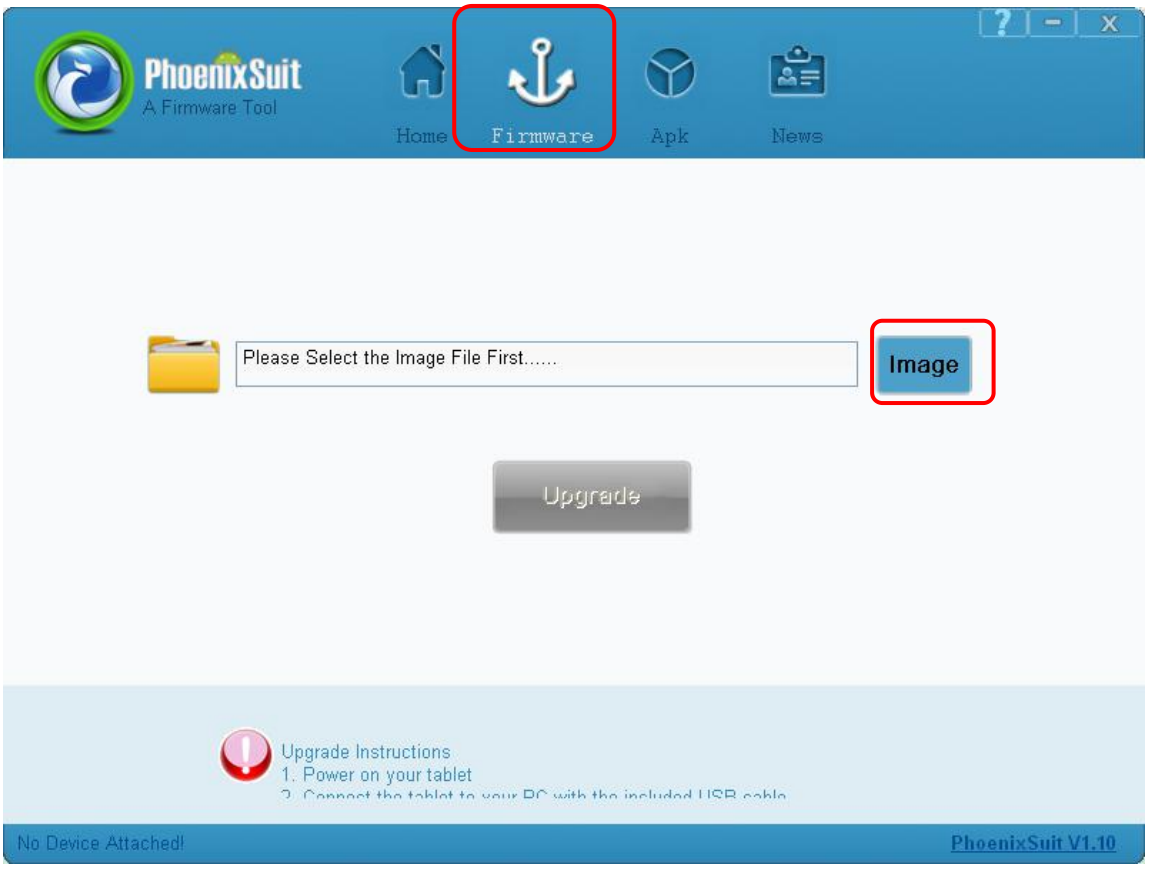
## 5.2 Burn image to eMMC via USB

Open the tool **PhoenixSuit**

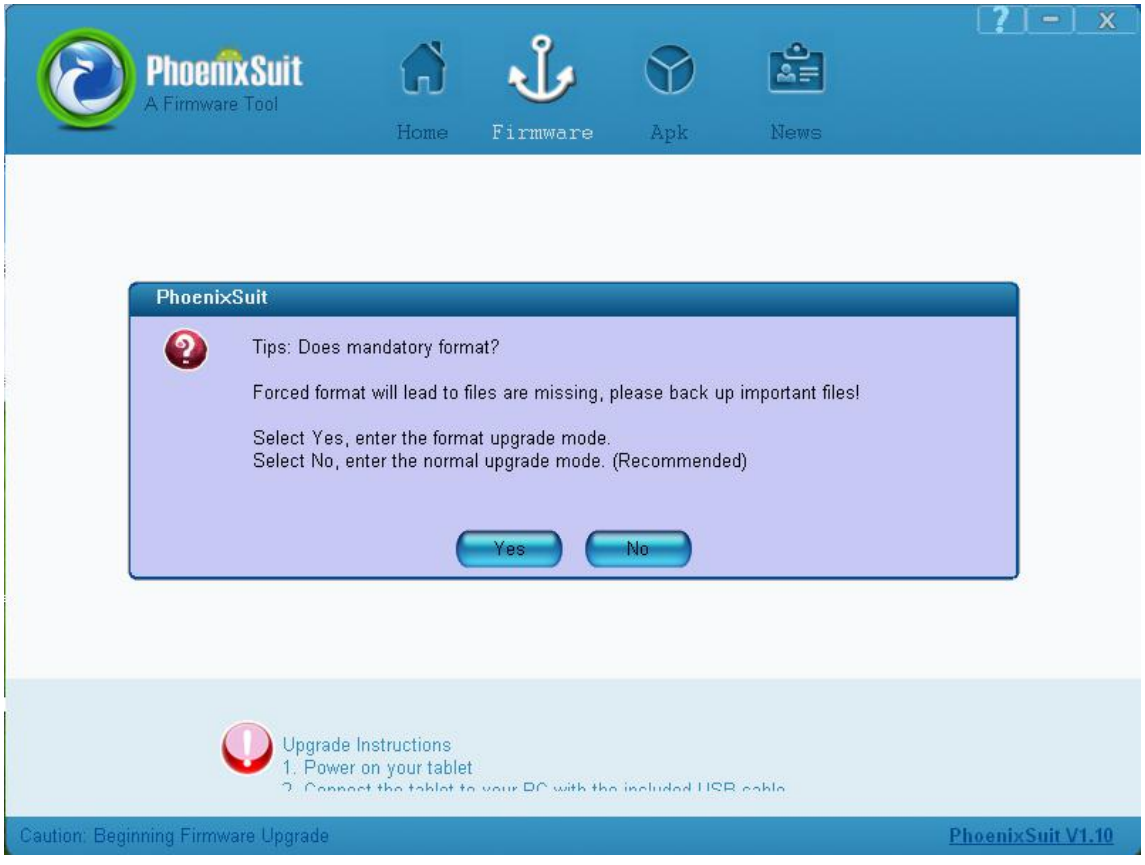



Select “Firmware”

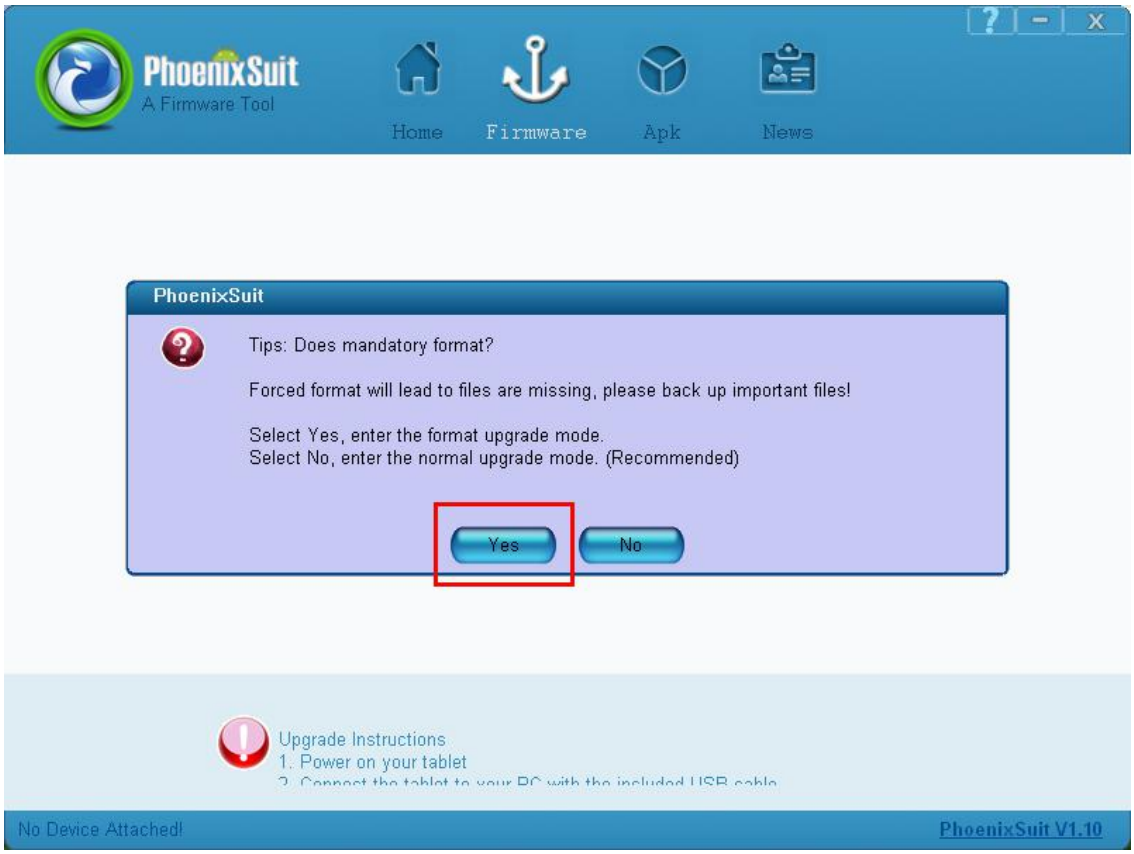
Click “Image” and select the image (***Path: CD\image***)

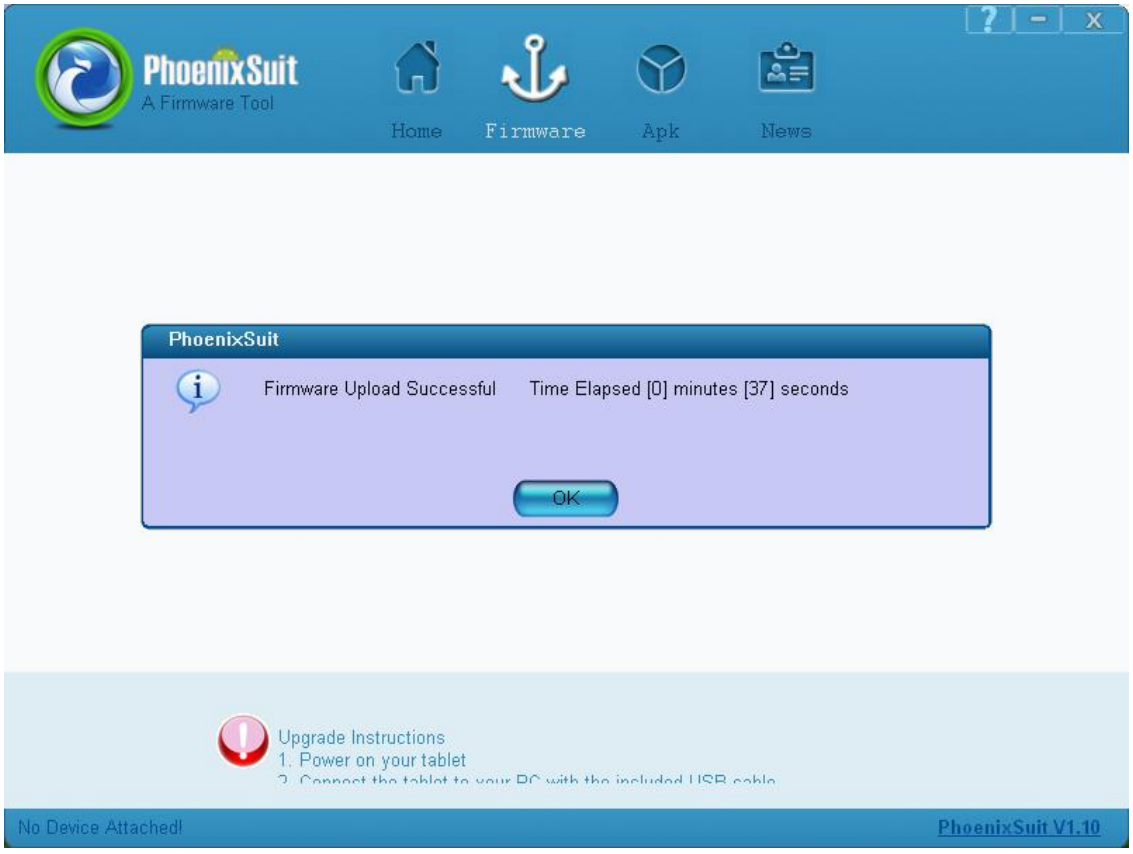



Press “Recover” key and hold then connect the board to PC via USB OTG until pop up the following window.



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

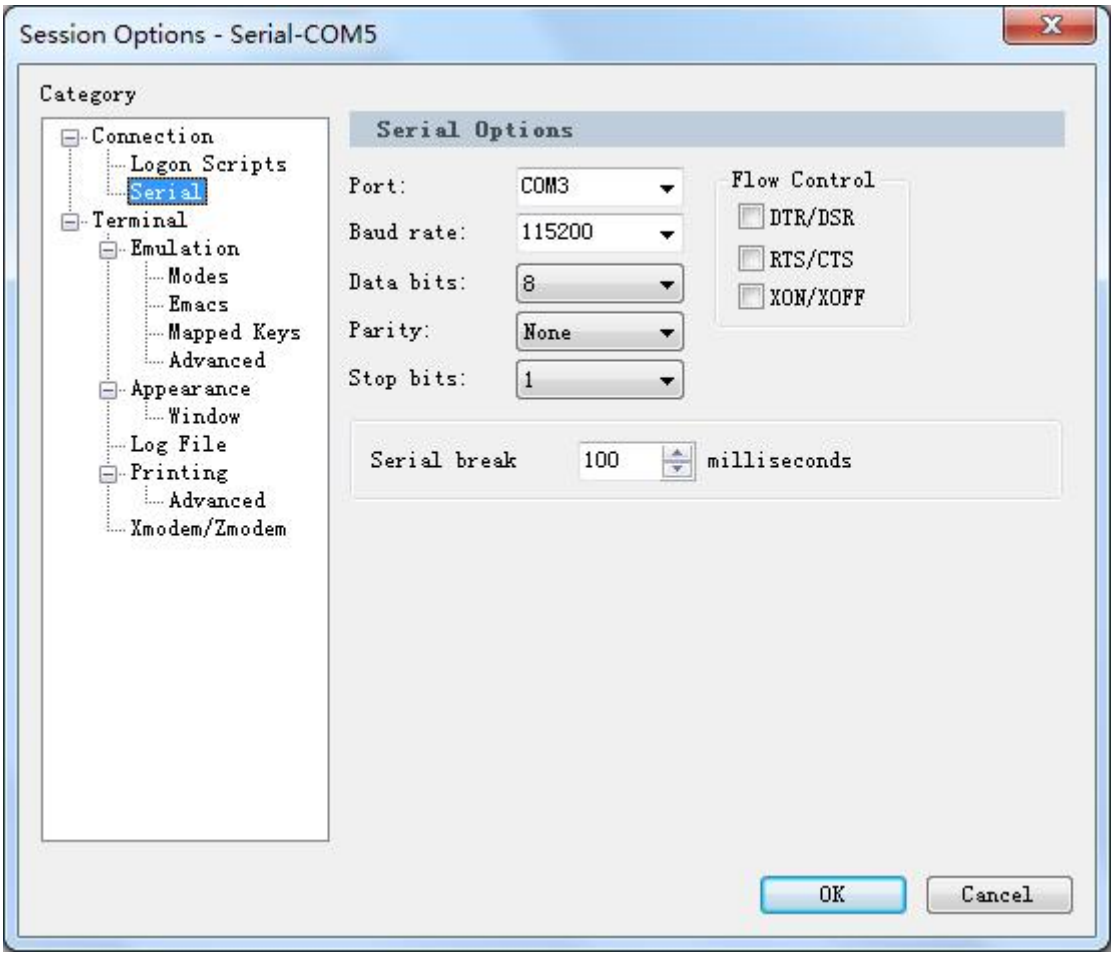


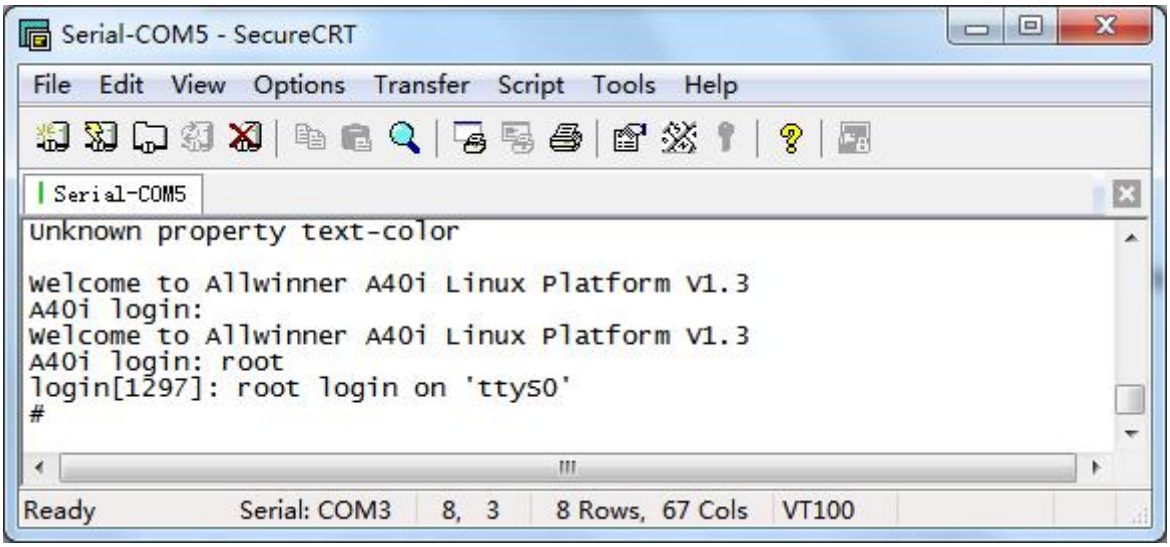



# 6 EMA40i Application Guidance

## 6.1 Serial Terminal

Connect the Serial port to the development board debug port, it will enter linux Serial terminal automatically as below.



## 6.2 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.

## 6.3 LVDS and Touch

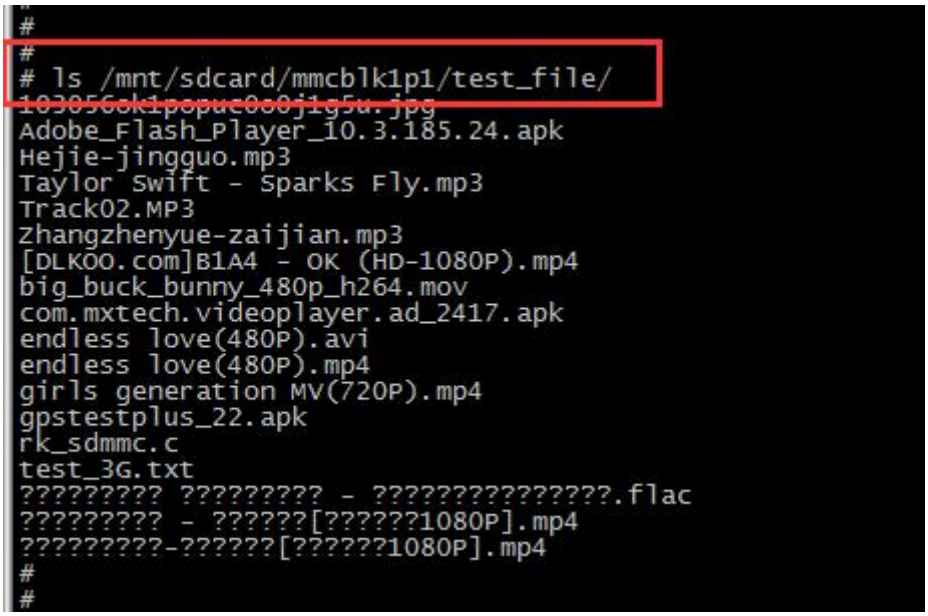
Connect LCDS to the board, then power on the board. After the board boot into Linux Qt, The LVDS will auto display QT UI.

Note: HDMI and LVDS display sync.

## 6.4 SD Card

The system will automatically mount the SD card.

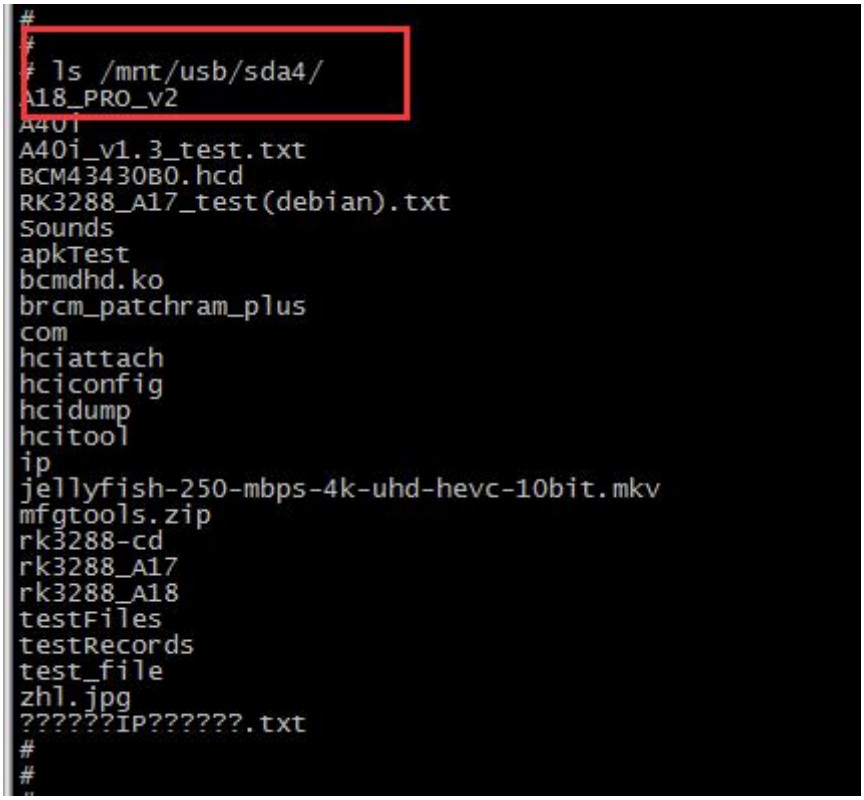
# ls /mnt/sdcard/mmcblk1p1/




## 6.5 USB Host

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

# ls /mnt/usb/sda4/




## 6.6 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/

## 6.7 EC20(4G)

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

#vi /etc/resolv.conf

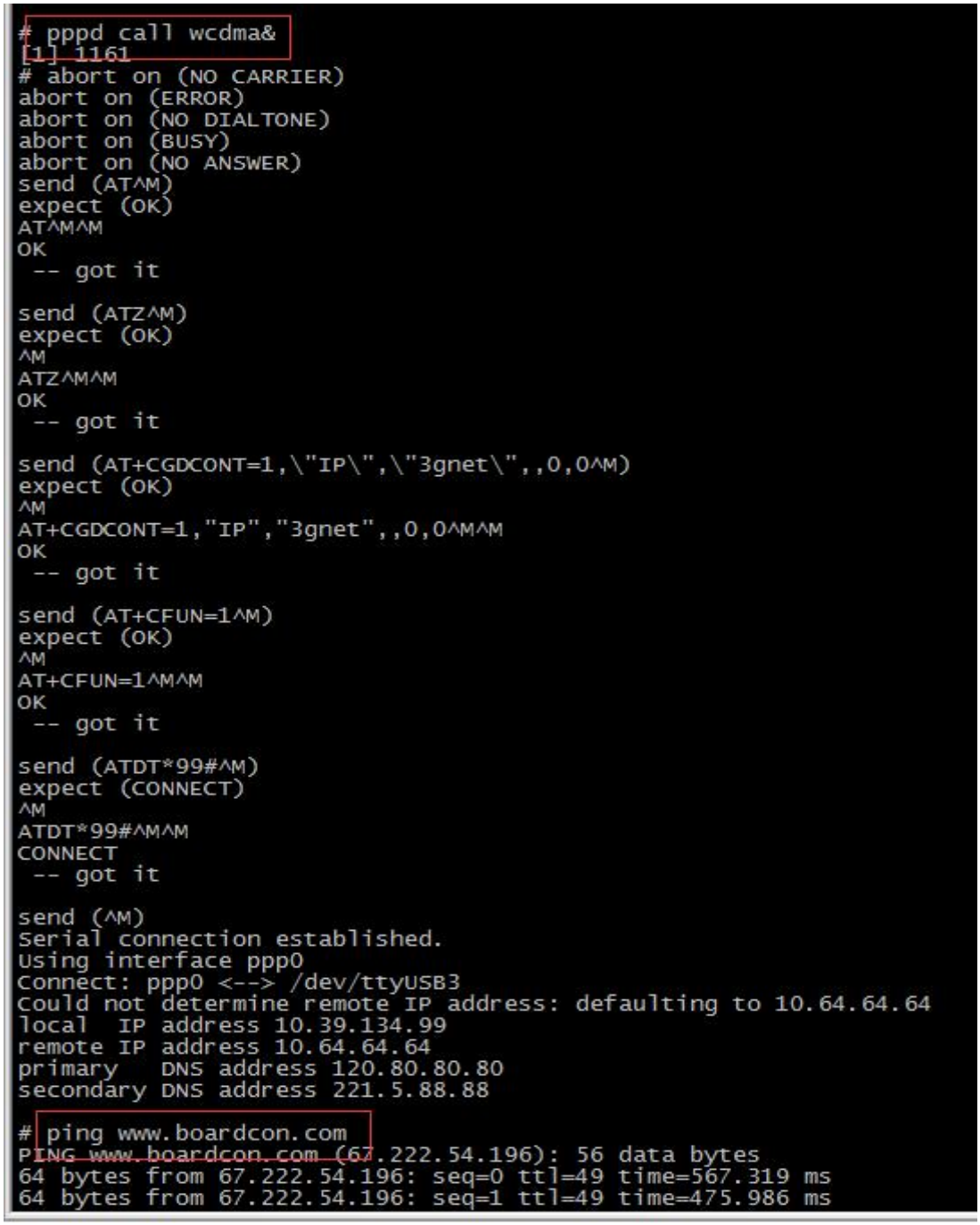
Add:

nameserver 8.8.8.8

nameserver 192.168.0.2

# pppd call wcdma&

# ping www.boardcon.com




## 6.8 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
```

## 6.9 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
#
```

## 6.10 Video Player

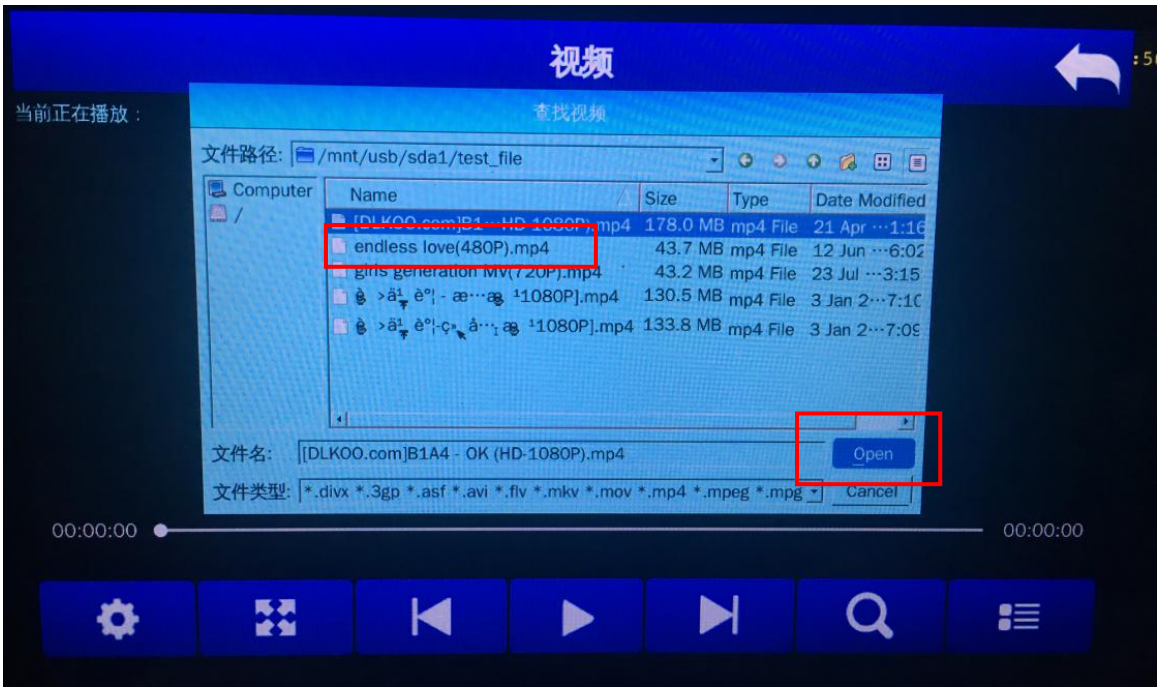
### 6.10.1 Play Video

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

媒体

视频
当前正在播放：
查找视频
文件路径: /mnt
Computer
/
Name
usb
Size
Type
Folder
Date Modified
12 Jun ···6:17
文件名:
Open
3
文件类型: *.divx *.3gp *.asf *.avi *.flv *.mkv *.mov *.mp4 *.mpeg *.mpg
Cancel
00:00:00
00:00:00
1

视频
当前正在播放：
查找视频
文件路径: /mnt/usb/sda1
Computer
/
Name
Size
Type
Date Modifi
3G
AC500
System Volume Information
a40i
myself
rootfs
rootfs0
test_file
Folder
文件名:
Open
文件类型: *.divx *.3gp *.asf *.avi *.flv *.mkv *.mov *.mp4 *.mpeg *.mpg
Cancel
00:00:00
00:00:00


视频
当前正在播放：
查找视频
文件路径: /mnt/usb/sda1/test_file
Computer
/
Name
Size
Type
Date Modified
178.0 MB mp4 File 21 Apr
endless love(480P).mp4
43.7 MB mp4 File 12 Jun
43.2 MB mp4 File 23 Jul
130.5 MB mp4 File 3 Jan
133.8 MB mp4 File 3 Jan
文件名: [DLKOO.com]B1A4 - OK (HD-1080P).mp4
Open
文件类型: *.divx *.3gp *.asf *.avi *.flv *.mkv *.mov *.mp4 *.mpeg *.mpg
Cancel
00:00:00
00:00:00

### 6.10.2 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

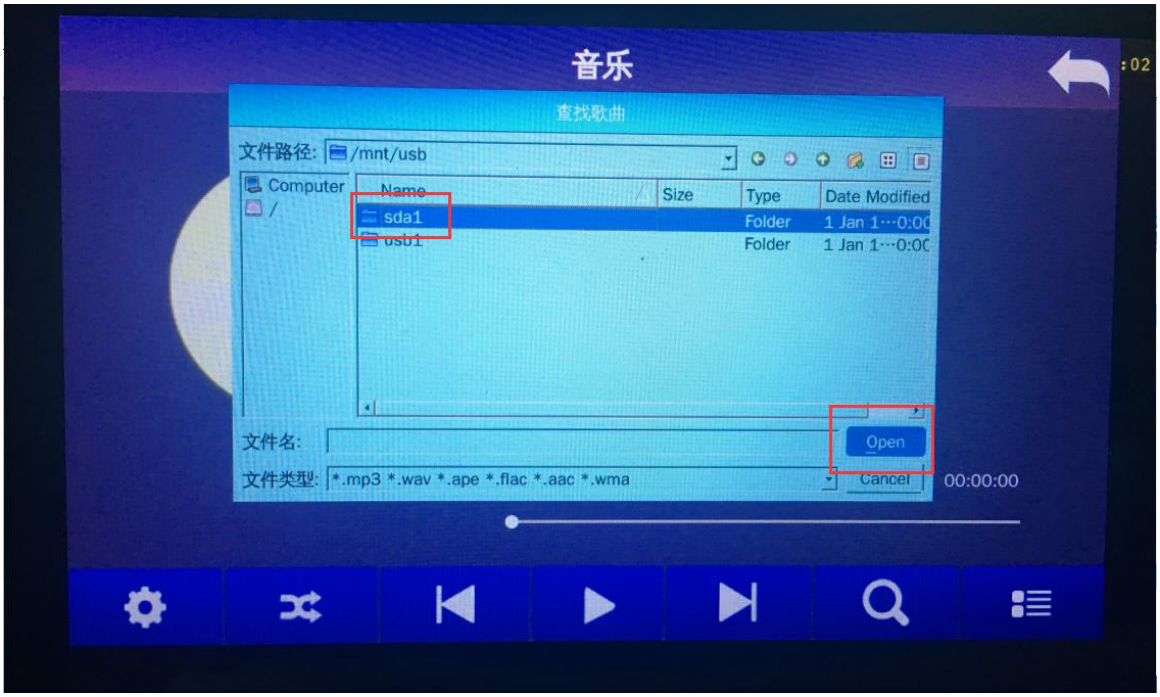
```
# cd /
# chmod 755 /xplayerdemo
# /xplayerdemo
# set url: /mnt/usb/sda4/test_file/Xueziqian-1080P.mp4      (the directory of video)
# show media info
# play
# pause
```

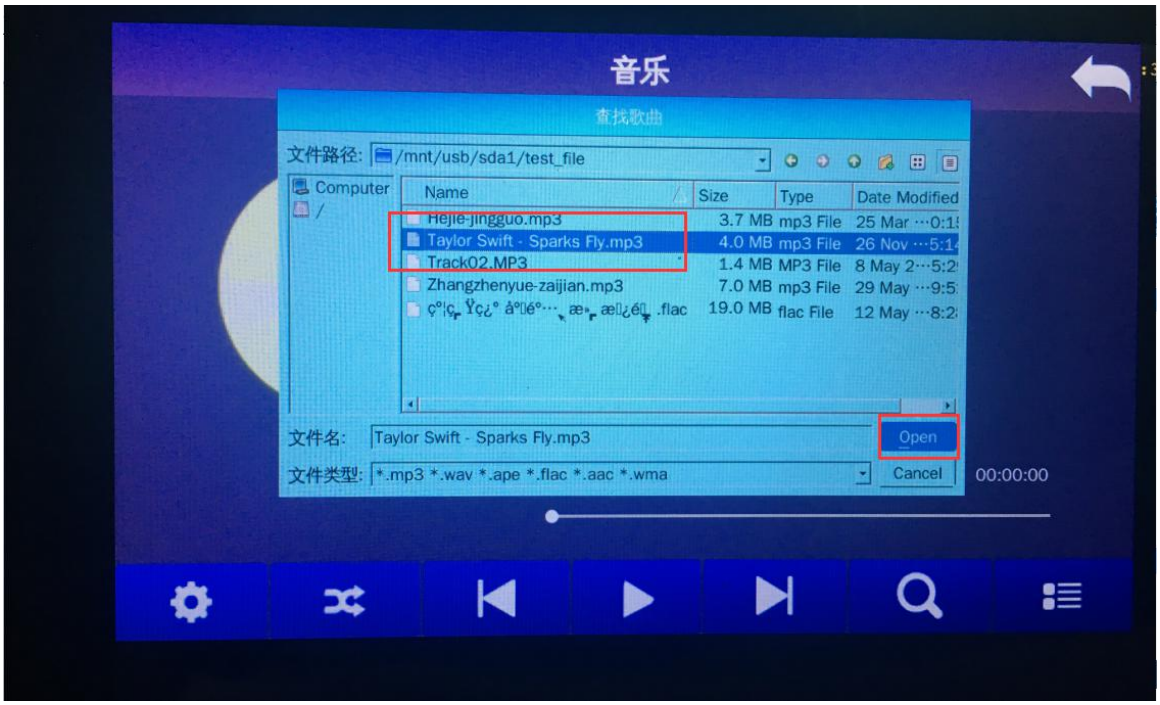
## 6.11 Audio Player

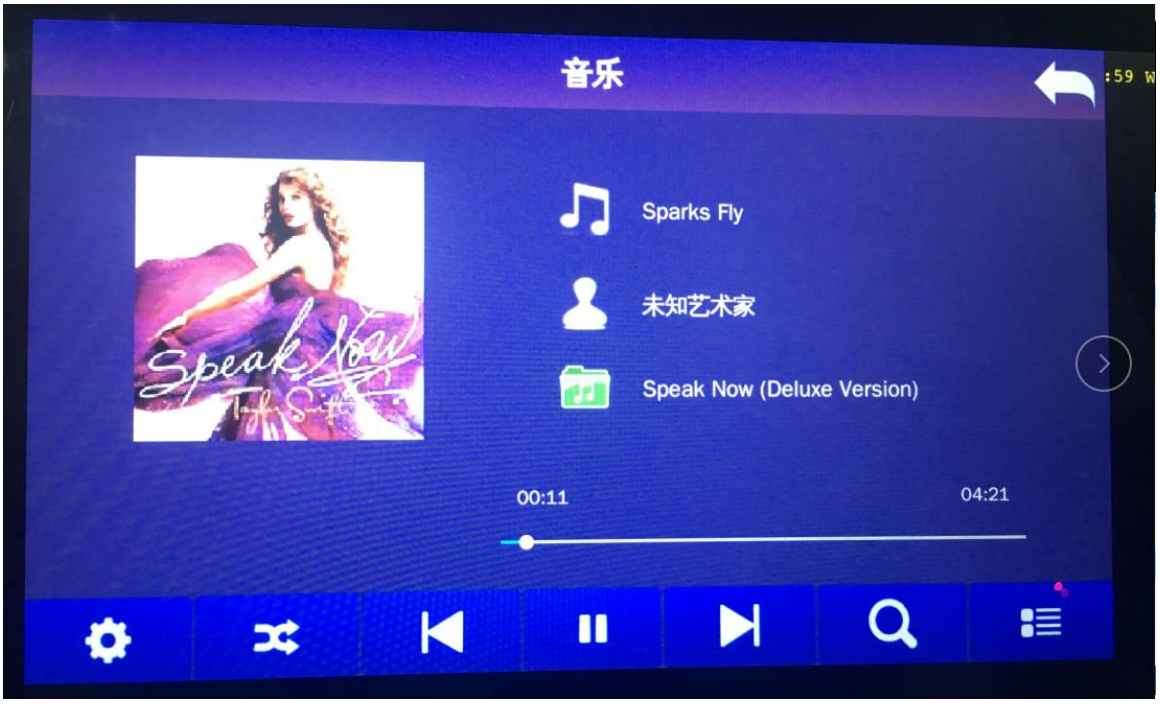
Put MP3 files to the SD card/U-disk and power on. Click the Audio Player icon:

媒体

音乐
查找歌曲
文件路径:
/mnt
Computer
/
usb
Name
Size
Type
Date Modified
Folder
12 Jun ···6:17
2
文件名:
open
3
文件类型:
*.mp3 *.wav *.ape *.flac *.aac *.wma
Cancel
00:00:00


音乐
查找歌曲
文件路径: /mnt/usb
Computer
/
Name
Size
Type
Date Modified
sda1
Folder
usb1
Folder
文件名:
Open
文件类型: *.mp3 *.wav *.ape *.flac *.aac *.wma
Cancel
00:00:00


音乐
查找歌曲
文件路径: /mnt/usb/sda1/test_file
Computer
/
Name
Size
Type
Date Modified
Hejie-jingguo.mp3
3.7 MB mp3 File
25 Mar
Taylor Swift - Sparks Fly.mp3
4.0 MB mp3 File
26 Nov
Track02.MP3
1.4 MB MP3 File
8 May
Zhangzhenyue-zaijian.mp3
7.0 MB mp3 File
29 May
19.0 MB flac File
12 May
.flac
文件名: Taylor Swift - Sparks Fly.mp3
Open
文件类型: *.mp3 *.wav *.ape *.flac *.aac *.wma
Cancel
00:00:00




## 6.12 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

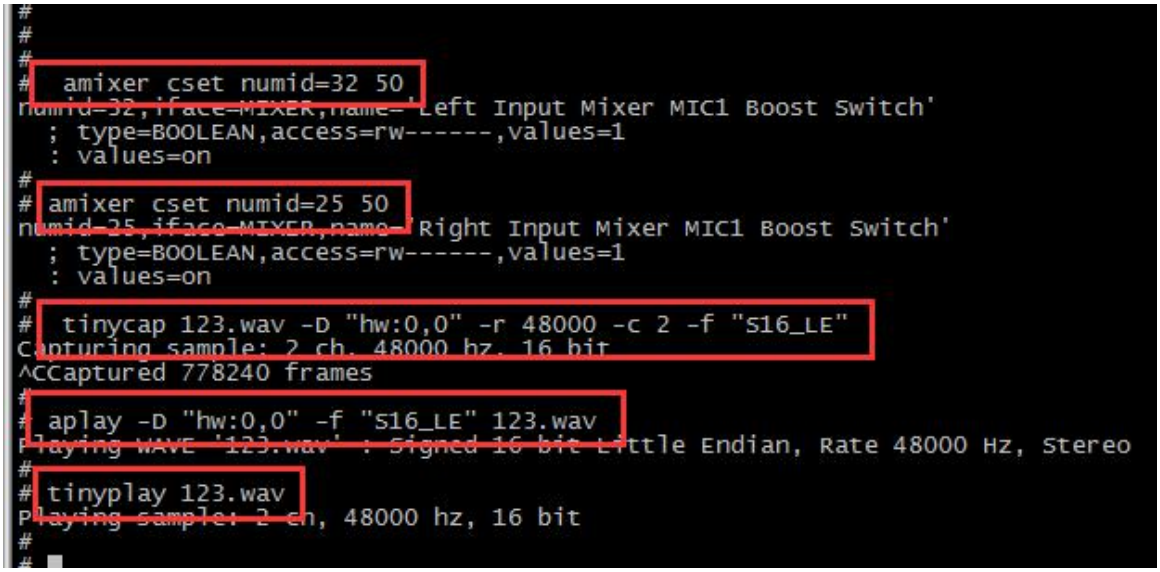
# arecord -D "hw:0,0" -r 48000 -c 2 -f "S16_LE" 123.wav

Play:

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

or

# tinyplay 123.wav



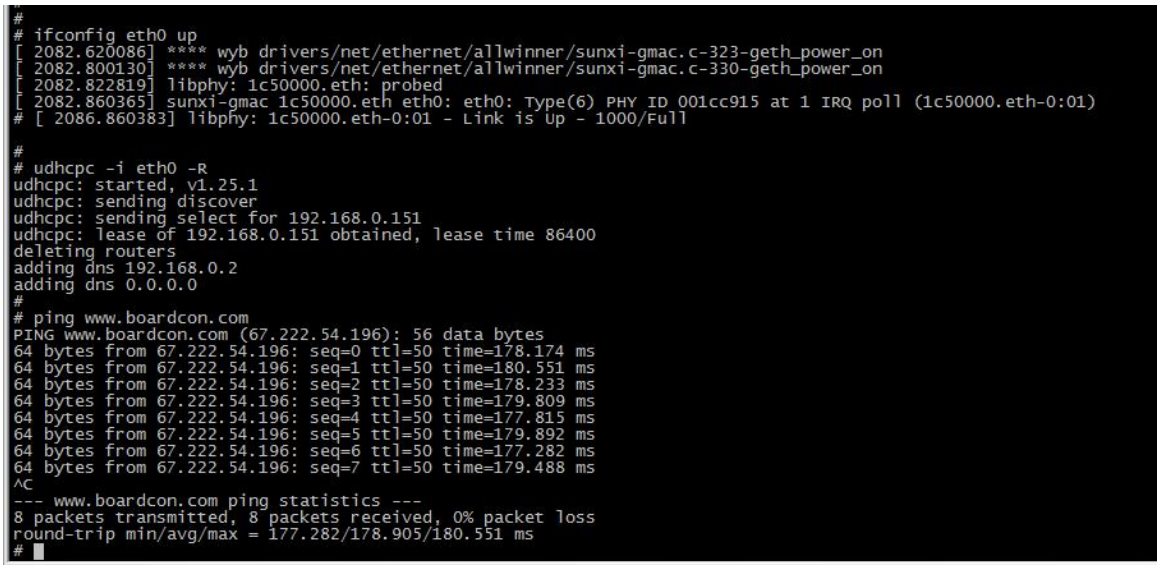

## 6.13 Ethernet

Plug in an Ethernet cable (RJ45).

#ifconfig eth0 up

#udhcpc -i eth0 -R

#ping www.boardcon.com




## 6.14 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)

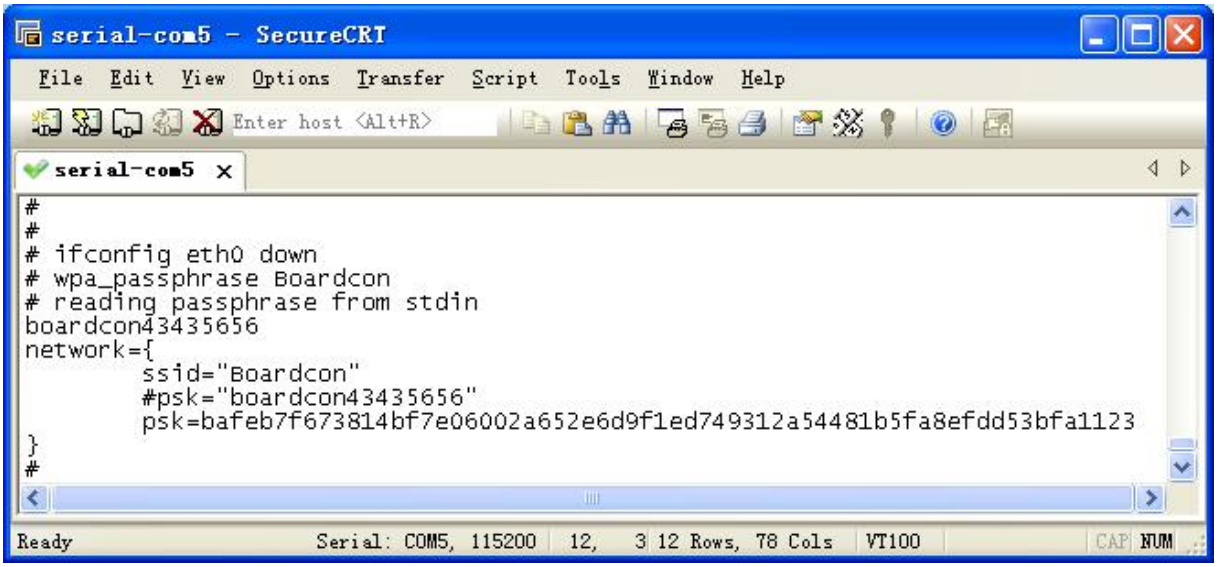
network={

ssid="Boardcon"

#psk="boardcon43435656"

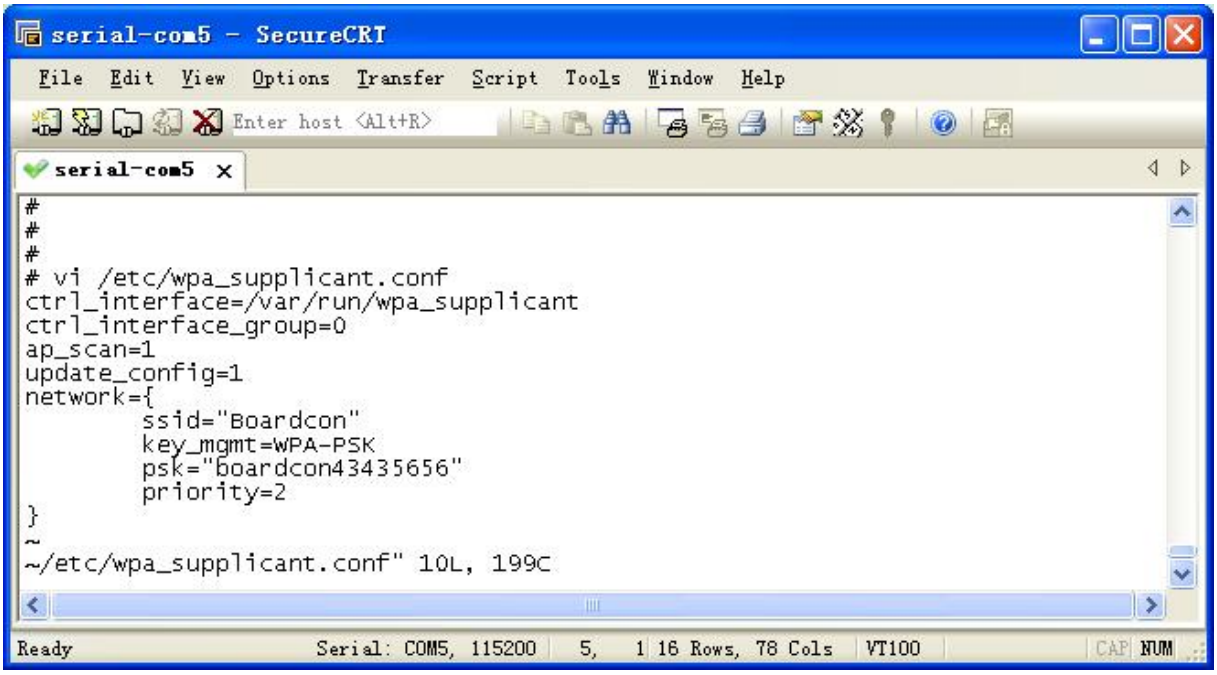
psk=bafeb7f673814bf7e06002a652e6d9f1ed749312a54481b5fa8efdd53bfa1123

}




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



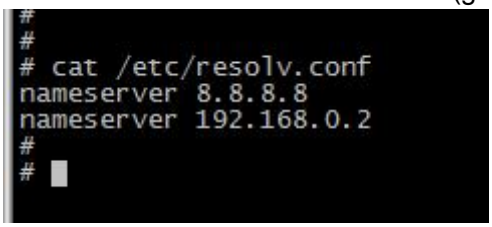

***Step 3***, add DNS server.

#vi /etc/resolv.conf

In the resolv.conf file add follow command:

nameserver 8.8.8.8

nameserver 192.168.0.2 (gate way)




***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] Connectting 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
#
```

# 6.15 Bluetooth

# cd /

# source /qt5.9.sh

# openbt

# hciconfig hci0 up

# hciconfig hci0 piscan

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

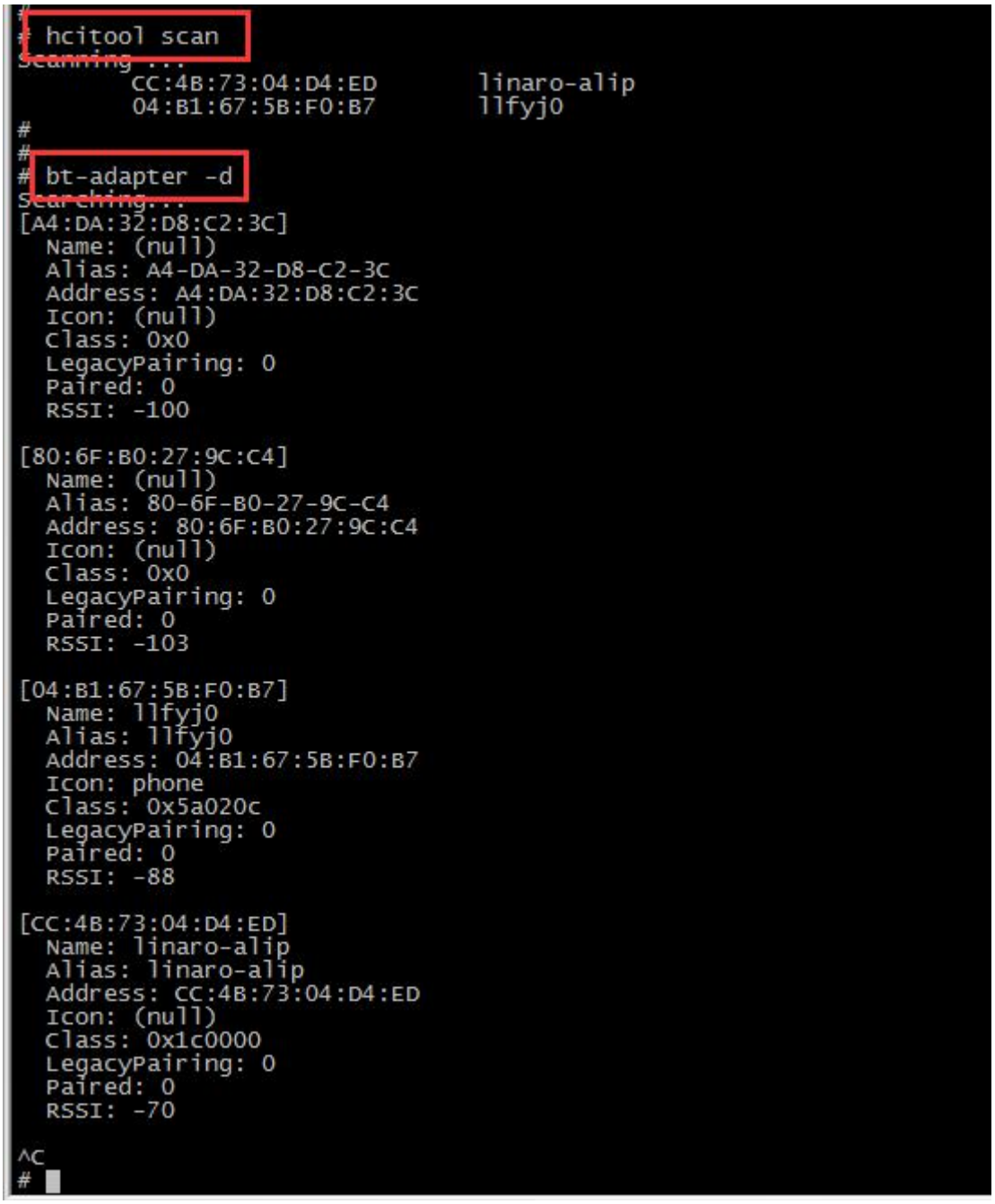
# /usr/libexec/bluetooth/obexd -a -d -n &

```
#
#
# 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:1189 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 &
[1] 4655
#
# /usr/libexec/bluetooth/obexd -a -d -n &
[2] 5696
# 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
```

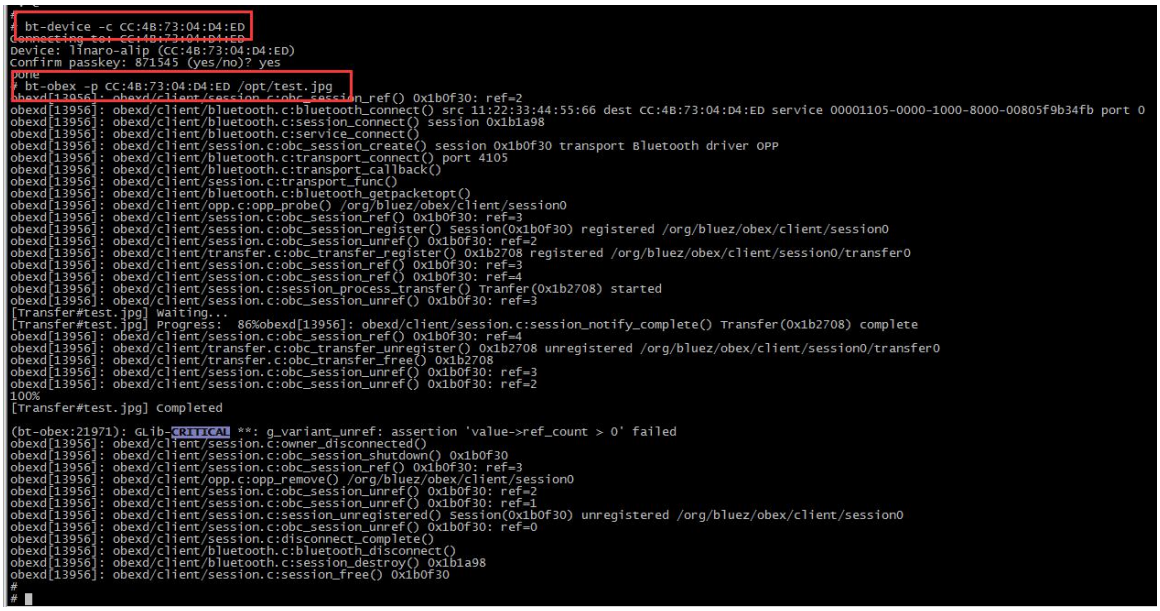
# hcitool scan

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

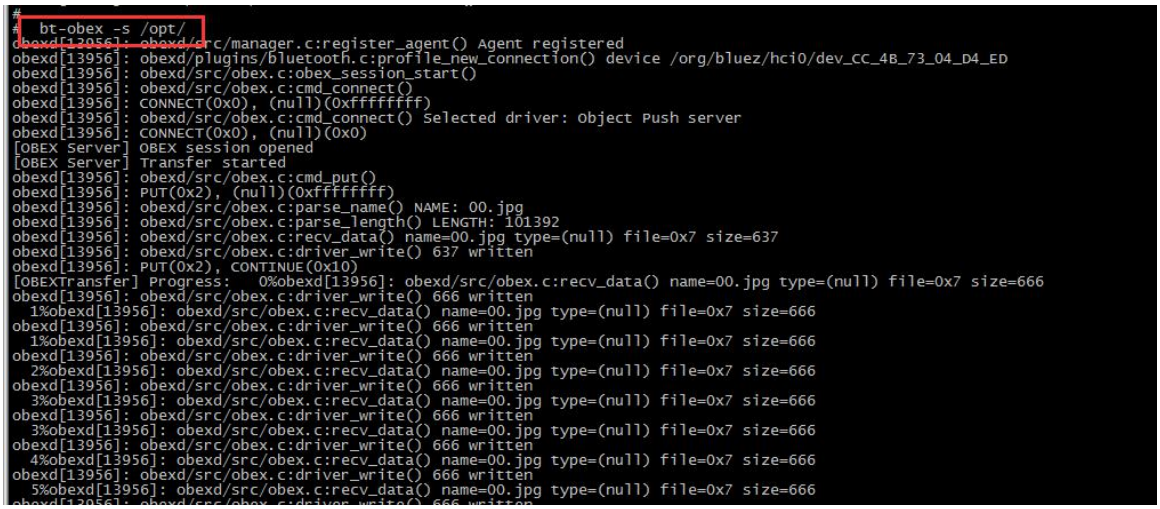


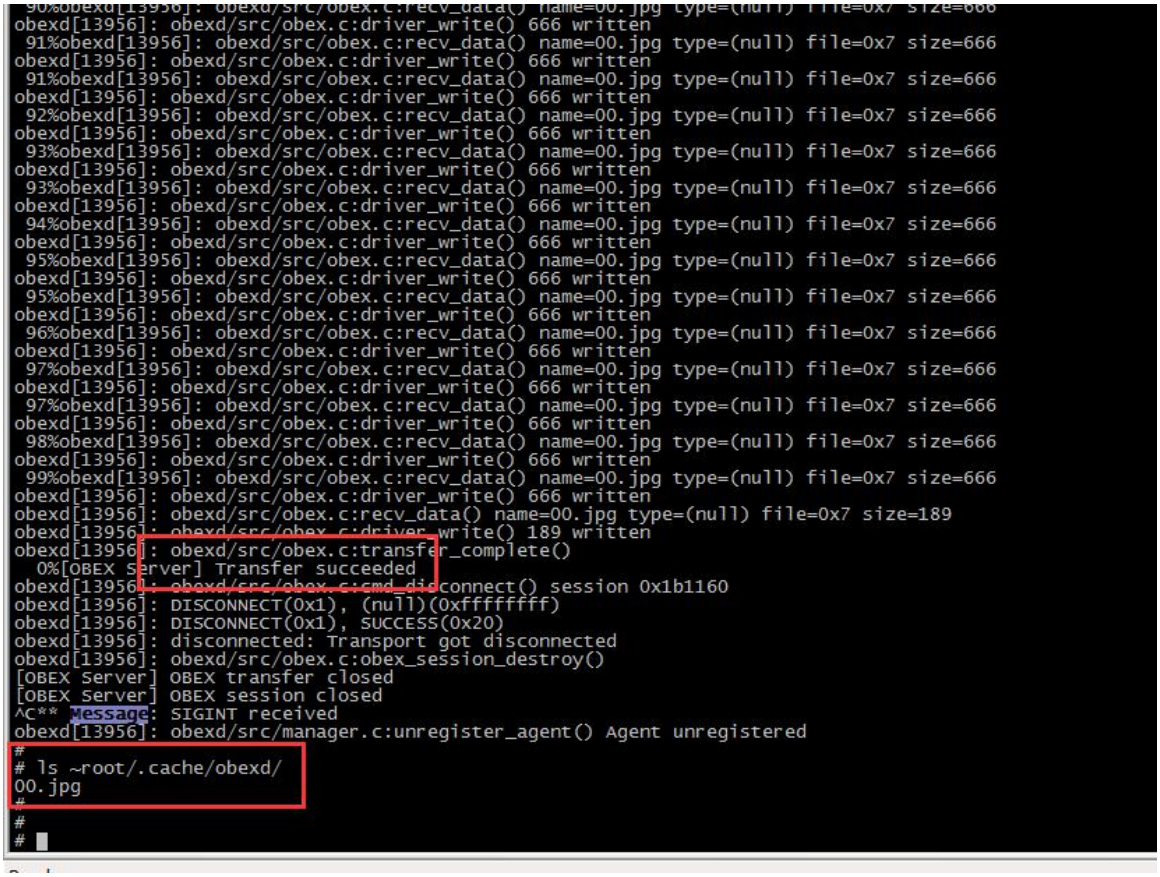

# bt-device -c CC:4B:73:04:D4:ED (Connect Bluetooth. CC:4B:73:04:D4:ED is the address of the connecting Bluetooth device)

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




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






## 6.16 3D

Execute the follow command to run 3D:

# glmark2-es2-fbdev

Return to the desktop after execute the command

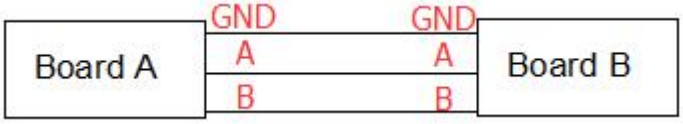
# debug /Launcher

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;1,1,1,1,1;: 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=shadow:windows=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=false:update-dispersion=0.9:update-fraction=0.5:update-method=subdata: FPS: 51 FrameTime: 19.608 ms
[buffer] columns=200:interleave=true:update-dispersion=0.9:update-fraction=0.5:update-method=map: FPS: 28 FrameTime: 35.714 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
=======================================================
#
```

## 6.17 RS485

Connect the two RS485 between EMA40i Board A and Board 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.

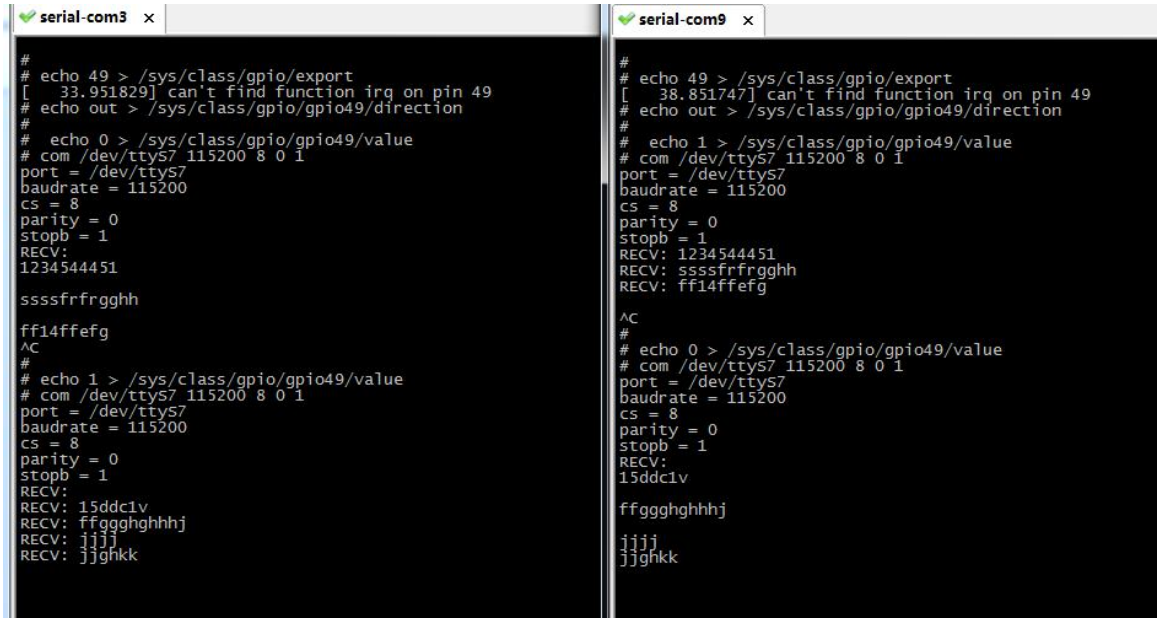
```
# echo 49 > /sys/class/gpio/export
# echo out > /sys/class/gpio/gpio49/direction
# echo 0 > /sys/class/gpio/gpio49/value    (Set low for sending.)
# com /dev/ttyS7 115200 8 0 1
```

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

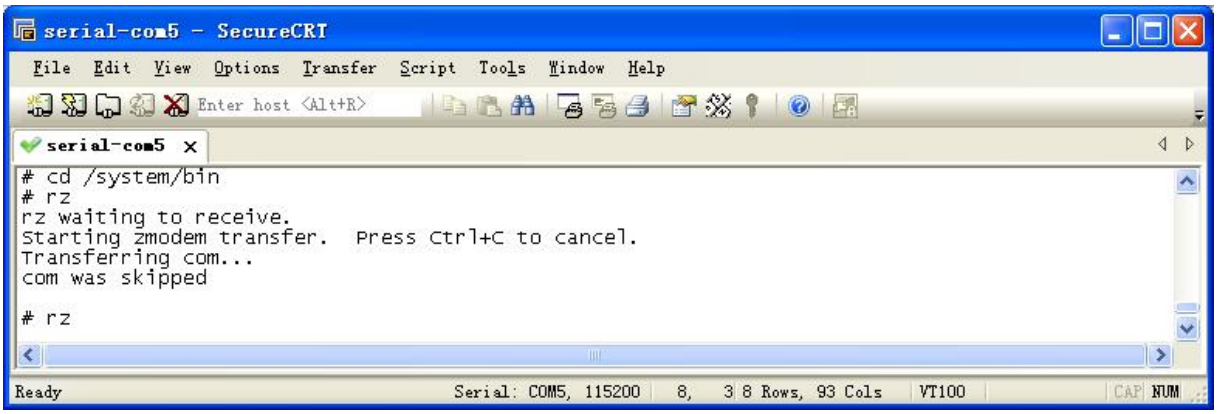
```
# echo 0 > /sys/class/gpio/gpio49/value    (Set low for sending.)
# com /dev/ttyS7 115200 8 0 1
```




## 6.18 RS232&UART (UART5 / UART6 / UART4 / UART7)

Execute the commands as follow:

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



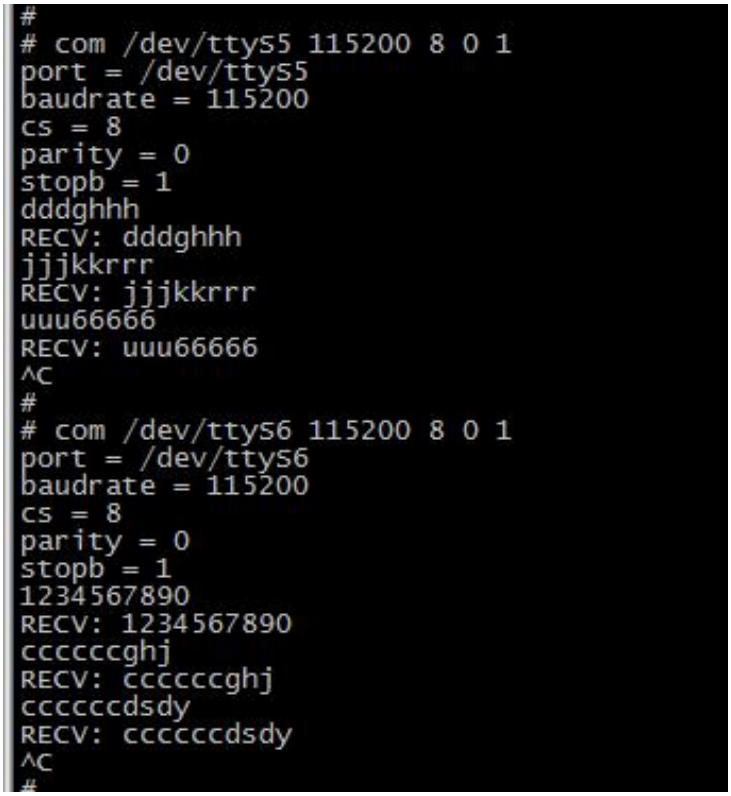
Connect RX&TX then execute the commands to run the test program at serial terminal

# 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: ccccccghj
ccccccdsdy
RECV: ccccccdsdy
^C
#