



IAC-IMX6UL-KIT Testing Manual

Version: 1.0
2016.10

QIYANG INTELLIGENTTECHNOLOGY Co., Ltd

Copyright Reserved

Catalogue

Catalogue.....	2
Preface	5
I. Preparation	6
. Board Testing	8
2.1 Buzzer Test.....	8
2.1.1 Overview	8
2.1.2 Test Steps.....	8
2.1.3 FAQ	10
2.2 RTC Test	10
2.2.1 Overview	10
2.2.2 Test Steps	10
2.2.3 FAQ.....	13
2.3 Watchdog Test.....	14
2.3.1 Overview	14
2.3.2 Test steps	14
2.3.3 FAQ	15
2.4 Serial Port Test.....	15
2.4.1 Overview	15
2.4.2 Test steps	16
2.4.3 FAQ	19
2.5 USB Testing	20
2.5.1 Overview	20
2.5.2 Test steps	20
2.5.3 FAQ	21
2.6 TF card testing	22
2.6.1 Overview	22
2.6.2 Test steps	22
2.6.3 FAQ	23
2.7 GPIO testing	24

2.7.1 Overview	24
2.7.2 Test steps	24
Remark: as the step4, set the every GPIO to low level, after pressing the enter, set all the GPIO to high level, press enter again, show testing is ok.....	27
2.7.3 FAQ	27
2.8 Ethernet test	27
2.8.1 Overview	27
2.8.2 Test steps	27
2.8.3 FAQ	28
2.9 3G/4G Module Test.....	28
2.9.1 Overview	28
2.9.2 Test steps	29
2.9.3 FAQ	32
2.10 Wifi Test.....	33
2.10.1 Overview	33
2.10.2 Test steps:	33
2.10.3 FAQ	37
2.11 AD Test.....	37
2.11.1 Overview	37
2.11.2 Test Steps.....	38
2.11.3 FAQ	38
2.12 Button Test.....	39
2.12.1 Overview	39
2.12.2 Test steps	39
2.12.3 FAQ	39
2.13 CAN Test	39
2.13.1 Overview	39
2.13.2 Test steps	40
2.13.3 FAQ	41
2.14 Audio Test	41
2.14.1 Overview	41

2.14.2 Test Steps.....	42
2.14.3 FAQ	43
2.15 LCD Display and VGA Test	43
2.15.1 Overview	43
2.15.2 Test steps	44
2.15.3 FAQ	44
2.16 Touch Panel Test	45
2.16.1 Overview	45
2.16.2 Test Steps.....	45
2.16.3 FAQ	47
2.17 QT Test.....	47
2.17.1 Overview	47
2.17.2 Test steps	48
Remark: If use touch, please use [ts_calibrate] before running QT application. Calibrate the touch panel, then execute the [Imx6_qt_test] program.....	48
2.17.3 FAQ	49

Preface

Welcome to use IAC-IMX6UL-Kit from Zhejiang Qiyang Intelligent Technology Co., Ltd. This Manual mainly introduces different interface functions and testing methods.

Please read carefully before using IAC-IMX6UL board:

IAC-IMX6UL-Kit Hardware manual.pdf and ***IAC-IMX6UL-Kit User Manual.pdf***

Target User

This manual applies to the following engineers:

Test Engineer

Technical Support engineer

Software Development engineer

Version Illustration:

Model Name	Product version
IAC-IMX6UL-Kit	V1.0

Revision record:

The revision history accumulates a description of each document update. The latest version of the document contains updates to all previous versions of the document.

Revision Date	Version	Revision note
2016/10/20	V1.0	Release

I. Preparation

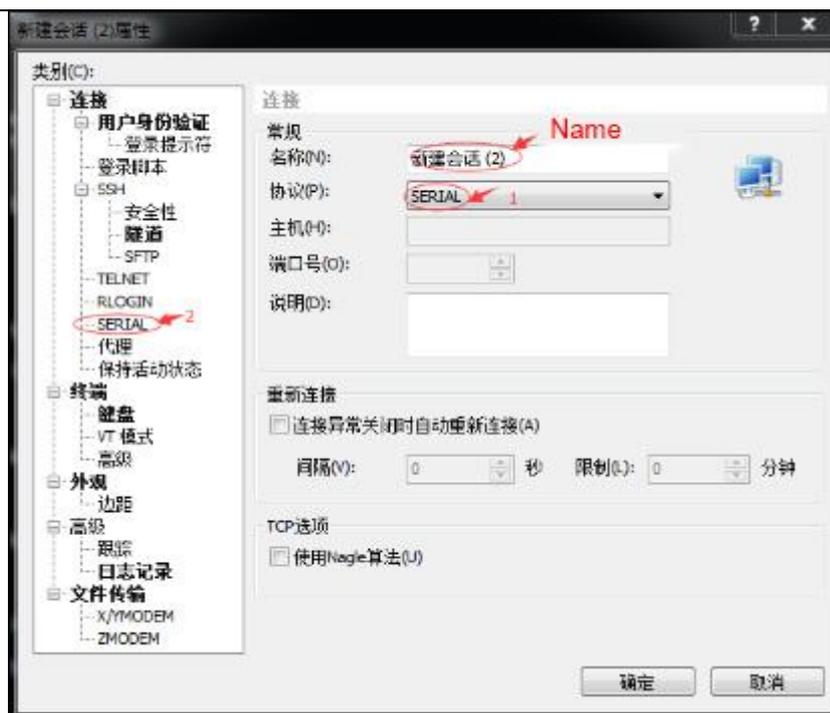
- Connect the core board IAC-IMX6UL-CM-Beta-V1.01 and the base board IAC-IMX6UL-MB-Beta-V2.00
- Connect the 7 inch LCD module(with resistive touch) QY-AT070TN83-V1 to the J8 connector of IMX6UL base board.
- Serial connection: connect the J14 of base board with the PC by our matching serial port cable.

Networking: connect Ethernet port J2 of base board with the PC by the network cable.

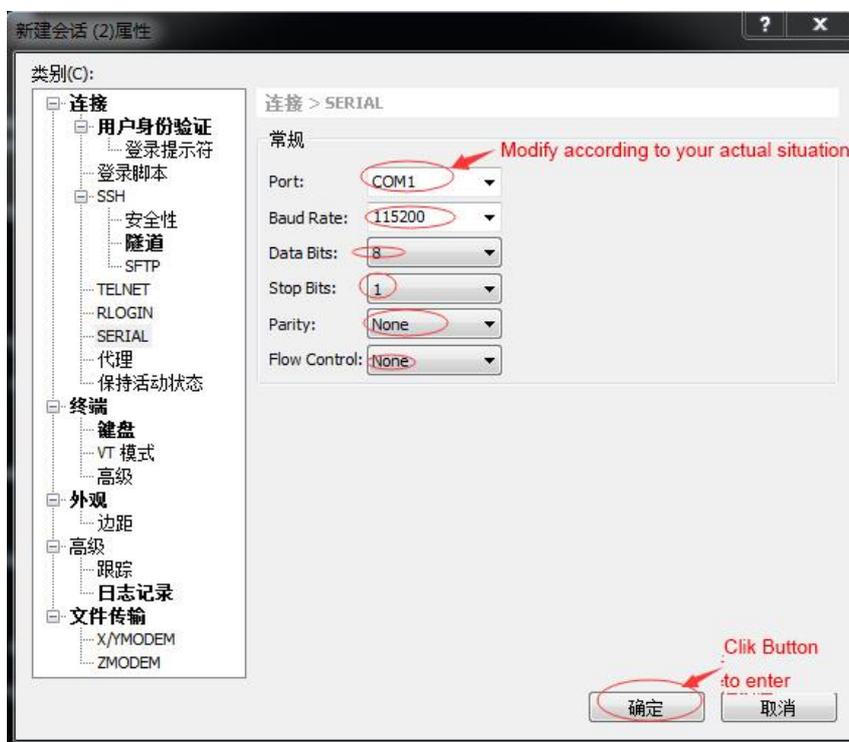
- Serial port setting: Open the terminal communication software---Xshell(mini terminal or hyper terminal from Windows) , Choose the serial port and set as follow: Baud rate (115200) 、 data bits (8 bits) 、 stop bit (1 bit) 、 check bit (No) 、 data flow control (No) , specific operation as follow picture 1-1, 1-2:



Picture 1-1



Picture 1-2



Picture 1-3

- The testing program is in the content /usr/ test, please switch to the content, Command and reference diagram as follow table, the all testing operations are completed in this directory.

Command	# cd /usr/test/
Description	Enter /usr/ test
Test Phenomenon	After executing the command, we can see the application from the print information, please refer the following picture.
Reference	
<pre> root@imx6ulevk:~# cd /usr/test/ root@imx6ulevk:/usr/test# ls 8723bu.ko backlight_test gpio_test rtc_test watchdog_feed_test QiYang_Imx6S_Qt_test buzzer_test keybutton serial_test watchdog_notfeed_test ad_test.sh can_test rs485_test shinian.mp3 </pre>	

. Board Testing

2.1 Buzzer Test

2.1.1 Overview

IAC-IMX6UL-KIT board use [GPIO3_4] to control the buzzer. When set [GPIO3_4] to low level, buzzer doesn't work; when set to high level, buzzer will buzzing.

2.1.2 Test Steps

Step 1 :

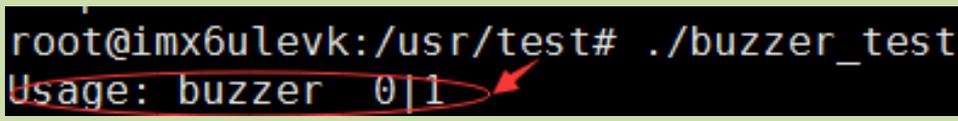
Command	# ls /dev
Description	Check whether there are available device nodes, if there is no buzzer device node, please check the manual 2.1.3
Test Phenomenon	After executing the command, we can see the application from the print information, please refer the following picture.

Reference

```

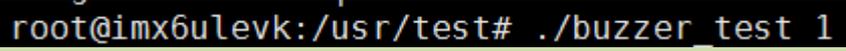
root@imx6ulevk:/usr/test# ls /dev
audio          loop5          ram1           tty11          tty36          tty60
autofs         loop6          ram10          tty12          tty37          tty61
block          loop7          ram11          tty13          tty38          tty62
bus            mem            ram12          tty14          tty39          tty63
char           memory_bandwidth ram13          tty15          tty4           tty7
console        mmcblk1        ram14          tty16          tty40          tty8
cpu_dma_latency mmcblk1boot0  ram15          tty17          tty41          tty9
disk           mmcblk1boot1  ram2           tty18          tty42          tty60
dri            mmcblk1p1     ram3           tty19          tty43          tty61
dsp            mmcblk1rpmb   ram4           tty2           tty44          tty62
fb0            mtab           ram5           tty20          tty45          tty63
fd             mxc_asrc      ram6           tty21          tty46          tty64
full           network_latency ram7           tty22          tty47          ubi_ctrl
fuse           network_throughput ram8           tty23          tty48          urandom
hwrng          null           ram9           tty24          tty49          v4l
i2c-0          ppp            random         tty25          tty5           vcs
i2c-1          pps0           rtc            tty26          tty50          vcs1
initctl        pps1           rtc0           tty27          tty51          vcsa
input          ptmx           shm            tty28          tty52          vcsa1
kmsg           ptp0           snd            tty29          tty53          video0
log            ptp1           stderr         tty3           tty54          watchdog
loop-control   pts            stdin          tty30          tty55          watchdog0
loop0          pxp_device     stdout         tty31          tty56          zero
loop1          qiyang_buzzer tty             tty32          tty57
loop2          qiyang_imx6_gpio tty0           tty33          tty58
loop3          qy_watchdog   tty1           tty34          tty59
loop4          ram0           tty10          tty35          tty6
  
```

Step 2 :

Command	# ./buzzer_test
Description	Run the testing program, do the next step according to the hint
Test Phenomenon	After executing the command, we can see the application from the print information, please refer the following picture.
Reference	

Remark: the red circle is the hint information, we need to add 0 or 1 after the buzzer to change the state.

Step 3 :

Command	# ./ buzzer_test 1
Description	Output high level to control the buzzer to ring
Test Phenomenon	After executing the command, the buzzer ring”beep” until close the Buzzer
Reference	

Step 4 :

Command	# ./ buzzer_test 0
Description	Output low level to control Buzzer to stop ringing
Test Phenomenon	After executing the command, the buzzer stop ringing
Reference	

2.1.3 FAQ

- [/dev] directory don't have [qiyang_buzzer] device node

solution:

- 1.Check whether add the drive program of Buzzer.
- 2.Check the kernel configuration(make menuconfig), whether it is same as the description of *IAC-IMX6UL-Kit BSP development instruction* .

2.2 RTC Test

2.2.1 Overview

QY-IMX6-KIT mainboard connects DS3231 Chip on the back plane by I2C connector as the external hardware clock. Set system time through [date] system command, and then write system time into hardware clock through [hwclock]command. Read hardware clock and print information through [rtc_test]test program.

Power off , then restart the board to check whether the time is accurate.

Please confirm you have already installed the battery on backplane (BT1)before testing RTC.

2.2.2 Test Steps

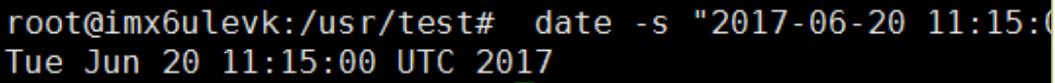
Step 1 :

Command	# ls /dev
Description	Check whether there are available device nodes, if there is no RTC device node, please check the manual 2.2.3 .
Test Phenomenon	After executing the command, we can see the RTC device node from the print information, please refer to the following picture.
Reference	<pre> root@imx6ulevk:/usr/test# ls /dev audio loop5 ram1 tty11 tty36 tty60 autofs loop6 ram10 tty12 tty37 tty61 block loop7 ram11 tty13 tty38 tty62 bus mem ram12 tty14 tty39 tty63 char memory_bandwidth ram13 tty15 tty4 tty7 console mmcblk1 ram14 tty16 tty40 tty8 cpu_dma_latency mmcblk1boot0 ram15 tty17 tty41 tty9 disk mmcblk1boot1 ram2 tty18 tty42 ttymxco dri mmcblk1p1 ram3 tty19 tty43 ttymxco1 dsp mmcblk1rpmb ram4 tty2 tty44 ttymxco2 fb0 mtab ram5 tty20 tty45 ttymxco3 fd mxc_asrc ram6 tty21 tty46 ttymxco4 full network_latency ram7 tty22 tty47 ubi_ctrl fuse network_throughput ram8 tty23 tty48 urandom hwrng null ram9 tty24 tty49 v4l i2c-0 ppp random tty25 tty5 vcs i2c-1 pps0 rtc tty26 tty50 vcs1 initctl pps1 rtc0 tty27 tty51 vcsa input ptmx shm tty28 tty52 vcsa1 kmsg ptp0 snd tty29 tty53 video0 log ptp1 stderr tty3 tty54 watchdog loop-control pts stdin tty30 tty55 watchdog loop0 pxp_device stdout tty31 tty56 zero loop1 qiyang_buzzer tty tty32 tty57 loop2 qiyang_imx6_gpio tty0 tty33 tty58 loop3 qy_watchdog tty1 tty34 tty59 loop4 ram0 tty10 tty35 tty6 </pre> <p><i>RTC设备节点</i> (RTC device node) is indicated by a red arrow pointing to the rtc and rtc0 entries in the terminal output.</p>

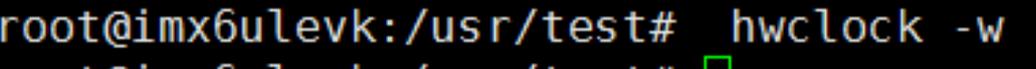
Step 2 :

Command	# date
Description	Execute the [date] command, we can check the current system clock.
Test Phenomenon	After executing the command, we can see the current system clock from the print information, please refer to the picture.
Reference	<pre> root@imx6ulevk:/usr/test# date Tue Jun 20 11:09:54 UTC 2017 </pre>

Step 3 :

Command	# date -s "2017-06-20 10:47:00"
Description	Execute the [date] command to set RTC, such as set time as the PC time
Test Phenomenon	After executing the command, we can see the time from the print information, please refer the following picture.
Reference	

Step 4 :

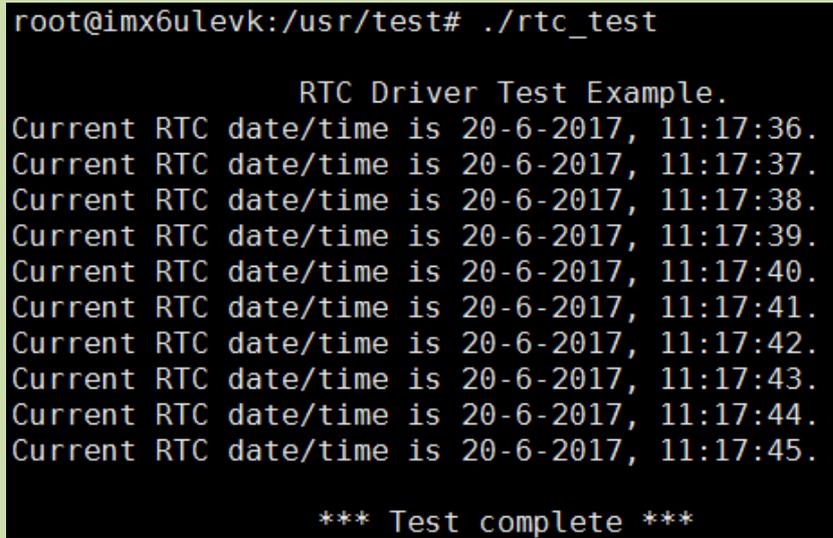
Command	# hwclock -w
Description	Write system time into hardware clock through [hwclock]command
Test Phenomenon	None
Reference	

Step 5 :

Command	# date # hwclock
Description	Respectively use[date] command and [hwclock] command to check the system clock and hardware clock.
Test Phenomenon	After executing the command, we can see the system and hardware time from the print information, please refer the following picture.

Reference	<pre>root@imx6ulevk:/usr/test# date Tue Jun 20 11:16:35 UTC 2017 root@imx6ulevk:/usr/test# hwclock Tue Jun 20 11:16:37 2017 0.000000 seconds</pre>
-----------	--

Step 6 :

Command	#./rtc_test
Description	After setting successful, execute the [rtc_test] program
Test Phenomenon	After executing the command, we can see the RTC from the print information, it walks accurately and no loss on second. please refer the following picture.
Reference	 <pre>root@imx6ulevk:/usr/test# ./rtc_test RTC Driver Test Example. Current RTC date/time is 20-6-2017, 11:17:36. Current RTC date/time is 20-6-2017, 11:17:37. Current RTC date/time is 20-6-2017, 11:17:38. Current RTC date/time is 20-6-2017, 11:17:39. Current RTC date/time is 20-6-2017, 11:17:40. Current RTC date/time is 20-6-2017, 11:17:41. Current RTC date/time is 20-6-2017, 11:17:42. Current RTC date/time is 20-6-2017, 11:17:43. Current RTC date/time is 20-6-2017, 11:17:44. Current RTC date/time is 20-6-2017, 11:17:45. *** Test complete ***</pre>

Step 7 :

After power off, then restart it to check whether the time is saved and walk accurately.

2.2.3 FAQ

- [/dev] directory don't have RTC device node.

Solution:

- 1.Check whether added the RTC drive program.
- 2.Check the kernel configuration [make menuconfig], whether it is same as the description

of ***IAC-IMX6UL-Kit BSP development instruction.***

- Time can not be saved, time travel error is huge, can not check hardware clock.

Solution:

1. Whether installed the battery on BT1.
2. Whether the battery has the electricity.

2.3 Watchdog Test

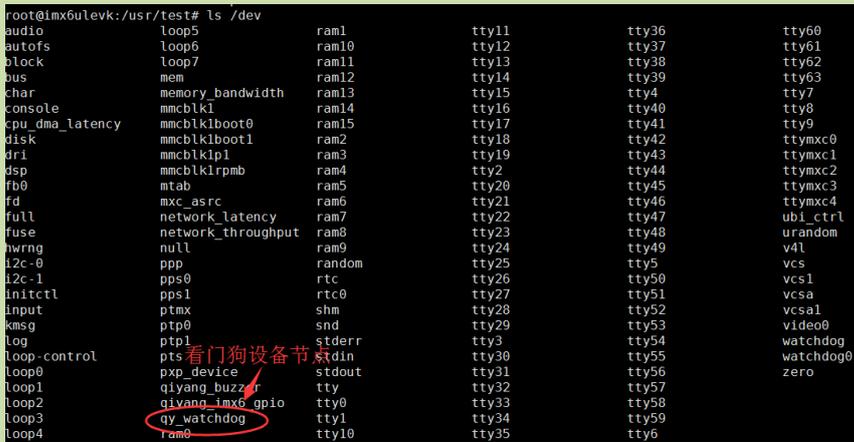
2.3.1 Overview

The mainboard IAC-IMX6UL-KIT has one Watchdog chip, which can restart the crashing system.

In testing process, enable Watchdog, to execute feeding or not feeding dog program, to check whether the system can reset normally.

2.3.2 Test steps

Step 1 :

Command	# ls /dev
Description	Check whether there are available device nodes, if there is no Watchdog device node, please check the manual 2.3.3
Test Phenomenon	After executing the command, we can see the Watchdog device node from the print information, please refer the following picture.
Reference	 <pre> root@imx6ulevk:/usr/test# ls /dev audio loop5 ram1 tty11 tty36 tty60 autofs loop6 ram10 tty12 tty37 tty61 block loop7 ram11 tty13 tty38 tty62 bus mem ram12 tty14 tty39 tty63 char memory_bandwidth ram13 tty15 tty4 tty7 console mmcblk1 ram14 tty16 tty40 tty8 cpu_dma_latency mmcblk1boot0 ram15 tty17 tty41 tty9 disk mmcblk1boot1 ram2 tty18 tty42 ttymx0 dri mmcblk1p1 ram3 tty19 tty43 ttymx1 dsp mmcblk1rpb ram4 tty2 tty44 ttymx2 fb0 mtab ram5 tty20 tty45 ttymx3 fd mxc_asrc ram6 tty21 tty46 ttymx4 full network_latency ram7 tty22 tty47 ubi_ctrl fuse network_throughput ram8 tty23 tty48 urandom hwrng null ram9 tty24 tty49 v4l i2c-0 ppp random tty25 tty5 vcs i2c-1 pps0 rtc tty26 tty50 vcs1 initctl pps1 rtc0 tty27 tty51 vcsa input ptmx shm tty28 tty52 vcsa1 kmsg ptp0 snd tty29 tty53 video0 log ptp1 stderr tty3 tty54 watchdog loop-control pts 看门狗设备节点 tty30 tty55 watchdog0 loop0 pxp_device stdout tty31 tty56 zero loop1 qiyang_buzz tty tty32 tty57 tty57 loop2 qiyang_imx6_gpio tty0 tty33 tty58 tty58 loop3 qiyang_watchdog tty1 tty34 tty59 tty59 loop4 ram0 tty10 tty35 tty6 tty6 </pre>

Step 2 :

Command	# ./watchdog_feed_test
Description	Open the Watchdog and feed.
Test Phenomenon	After executing the command, we can see the Enable Watchdog from the print information, please refer the following picture.
Reference	<pre>root@imx6ulevk:/usr/test# ./watchdog_feed_test watchdog: enable watchdog</pre>

Step 3 :

Command	# ./watchdog_nofeed_test
Description	Open Watchdog, not feed
Test Phenomenon	After Executing the command, the system reboots .
Reference	None

2.3.3 FAQ

- [/dev] directory don't have RTC device node.

Solution:

- 1.Check whether add the Watchdog Drive program.
- 2.Check the kernel configuration [make menuconfig], whether it is same as the description of *IAC-IMX6UL-Kit BSP development instruction*.

2.4 Serial Port Test

2.4.1 Overview

There are 5-ch serial ports :J14 is as the debug UART. The other 4-ch could be used as the common RS232 serial port, including:

COM2(J17), COM3(J16) and COM5(J15) are 5- wire serial port.

COM4(J12) is RS485.

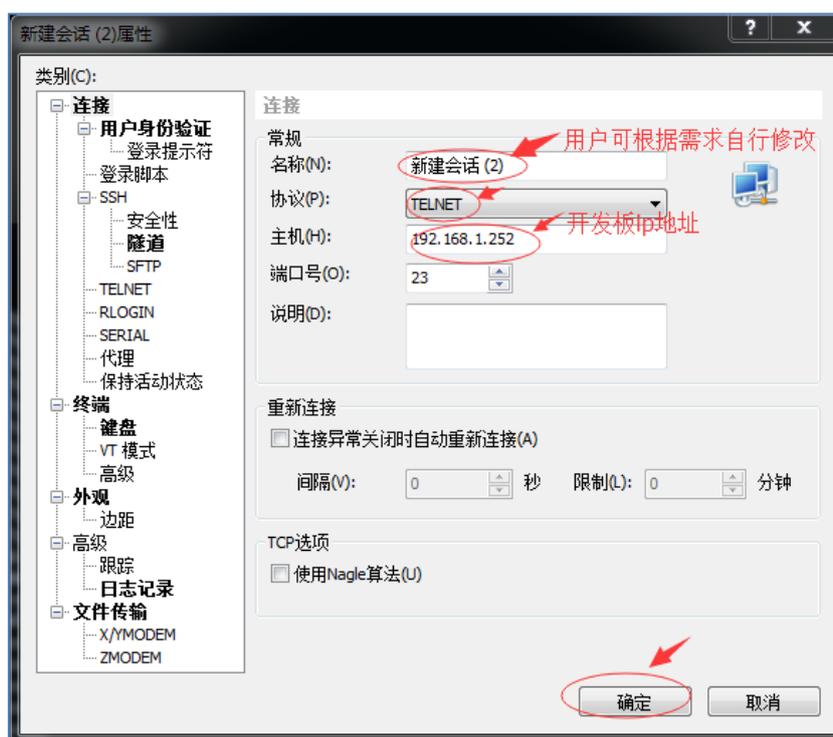
This test program only aims at testing of common RS232.

When do serial ports testing, the PC needs 2-ch serial ports.

1. One connect to debugging port for interaction
2. One connect to under test ports to transmit and receive data

If there is only 1-ch serial port, connect to under test port and connect to development board by network cable. Log in development system as debugging ports through hyper terminal xshell telnet.

- **Log in the development board through PC as follows:**

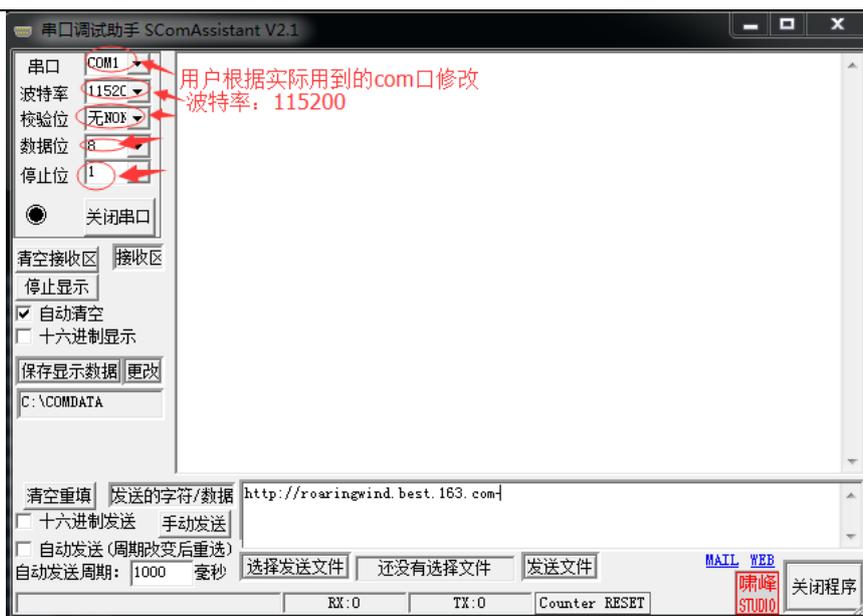


Testing the program, one serial port will send the character data [Comx RS232 test string!] on every second, X is the serial number of the actual test, it could block the serial data and prints through multithreading.

2.4.2 Test steps

Here is introduction on the COM2(J17) as a sample, other serial ports testing is the same.

Step 1: Open the serial debugging assistant and set



Step 2(From now on, these steps are operated under the telnet of super terminal)

Command	# ls /dev
Description	Check whether there are available device nodes, if there is no Serial ports device node, please check the manual 2.4.3
Test Phenomenon	After executing the command, we can see the serial ports device node from the print information, please refer the following picture.
Reference	<pre> root@imx6ulevk:/usr/test# ls /dev/ audio loop1 ptp0 random tty21 tty42 tty63 autofs loop2 ptp1 rtc tty22 tty43 tty7 block loop3 pts rtc0 tty23 tty44 tty8 bus loop4 pxp_device shm tty24 tty45 tty9 char loop5 qiyang_buzzer snd tty25 tty46 ttyimxc0 console loop6 qiyang_imx6_gpio stderr tty26 tty47 ttyimxc1 cpu_dma_latency loop7 qy_watchdog stdin tty27 tty48 ttyimxc2 disk mem ram0 stdout tty28 tty49 ttyimxc3 dri memory_bandwidth ram1 tty tty29 tty5 ttyimxc4 dsp mmcblk1 ram10 tty0 tty3 tty50 ubi_ctrl fb0 mmcblk1boot0 ram11 tty1 tty30 tty51 urandom fd mmcblk1boot1 ram12 tty10 tty31 tty52 v4l full mmcblk1p1 ram13 tty11 tty32 tty53 vcs fuse mmcblk1rpmb ram14 tty12 tty33 tty54 vcs1 hwrng mtab ram15 tty13 tty34 tty55 vcsa i2c-0 mxc_asrc ram2 tty14 tty35 tty56 vcsa1 i2c-1 network_latency ram3 tty15 tty36 tty57 video0 initctl network_throughput ram4 tty16 tty37 tty58 watchdog input null ram5 tty17 tty38 tty59 watchdog0 kmsg ppp ram6 tty18 tty39 tty6 zero log pps0 ram7 tty19 tty4 tty60 loop-control pps1 ram8 tty2 tty40 tty61 loop0 ptmx ram9 tty20 tty41 tty62 </pre>

Step 3 :

Command	# ./serial_test
Description	Run the testing program of serial port

Test Phenomenon	After executing the command, we can see the serial ports hint information from the printed, please refer the following picture.
Reference	<pre> root@imx6ulevk:/usr/test# ./serial_test Invalid arguments! Usage: ./serial_test [0123] 0 --/dev/ttymx1 com2 test, used as rs232. 1 --/dev/ttymx2 com3 test, used as rs232. 2 --/dev/ttymx3 com4 test, used as rs232. 3 --/dev/ttymx4 com5 test, used as rs232. </pre>

Remark:

Com2 -> J17 -> /dev/ttymx1

Com3 -> J16 -> /dev/ttymx2

Com4 -> J12 -> /dev/ttymx3

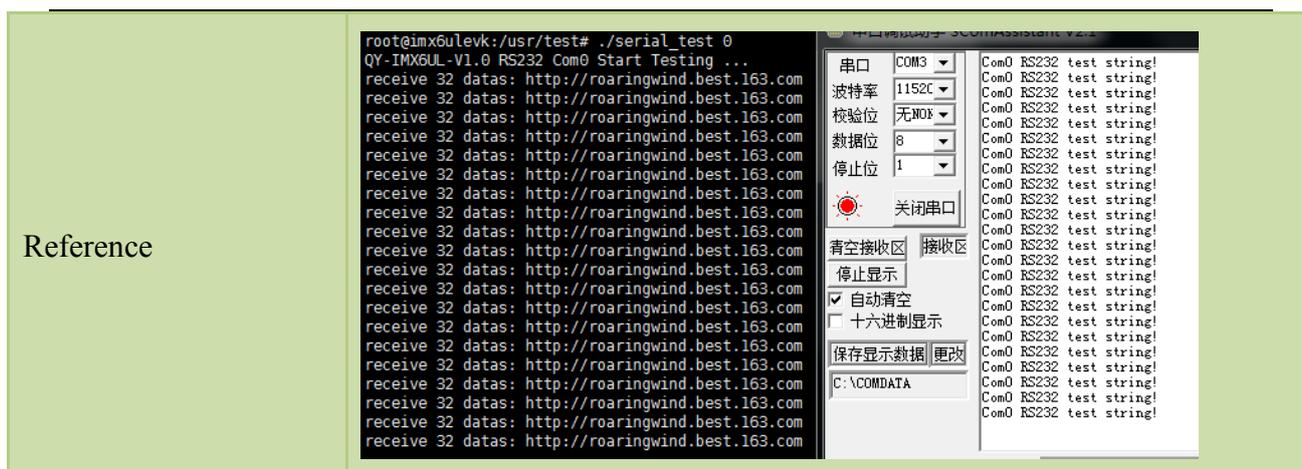
Com5 -> J15-> /dev/ttymx4

Step 3 :

Command	# ./serial_test 0 ^①
Description	Run the serial port application program
Test Phenomenon	After executing the command, the telnet terminal receive and print the information from the serial debugging terminal, and the serial debugging assistant receive and print the information from the serial ports. Please refer the following picture.

Remark L

① /serial_test 0 means test com2.



Reference

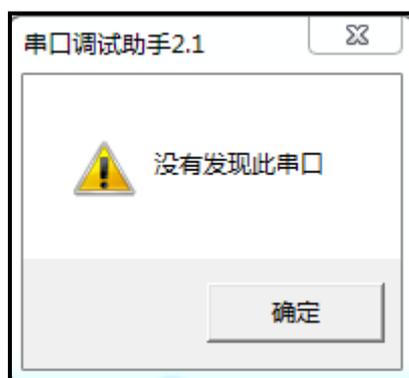
Telnet and Serial Debugging Assistant both can receive data and the data shows no error, it means ports function work normally.

After testing COM2, [Ctrl+C] to quit from the program, and then continue other ports' testing. The testing method is same.

t

2.4.3 FAQ

- Open the serial debugging assistant, it shows a window as below:



Solution:

1. Check whether the serial cable is connected normally or not.
 2. Check whether the serial of hyper terminal has disconnected.
- The communication of serial ports is abnormal or can not be communicated.

Solution:

1. Check the serial cable is normal or not.
2. The debugging software configuration is correct or not.

2.5 USB Testing

2.5.1 Overview

There are 4-ch USB ports on the IAC-IMX6UL-KIT mainboard.

Ch-1 and Ch-4 (J6) are regarded as device port, it is used for downloading firmware program by USB. Ch-2 and Ch-3 are used as USB- Host, including ,1-ch is to multiplex with WIFI port, 1-ch is to multiplex with miniPCIE interface. This test aims at testing of host port.

Development board USB HOST support hot plug, plug USB flash Disk, system will automatically recognize and print USB Flash Disk information. After recognizing, it generates the device node [/dev/sda] and partition node [/dev/sda1] in [/dev] directory.(If there are several partitions, then number parts will be increased step by step) Finally, system will automatically mount all partition to [/media/] directory, judge whether the interface is normal or not through reading and writing corresponded file of directory.

2.5.2 Test steps

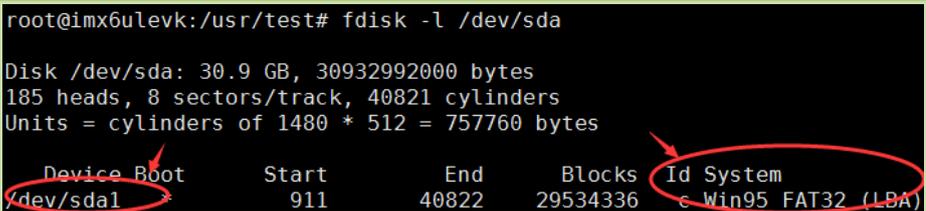
Test the USB Flash Disk with only 1 partition as a sample.

Step 2: Plug normal USB Flash Disk to USB host, debug port prints the following information:

```

root@imx6ulevk:/usr/test# usb 1-1.2: new high-speed USB device number 5 using ci_hdrc
usb-storage 1-1.2:1.0: USB Mass Storage device detected
scsi host1: usb-storage 1-1.2:1.0
scsi 1:0:0:0: Direct-Access    SMI        USB DISK        1100 PQ: 0 ANSI: 6
sd 1:0:0:0: [sda] 60416000 512-byte logical blocks: (30.9 GB/28.8 GiB)
sd 1:0:0:0: [sda] Write Protect is off
sd 1:0:0:0: [sda] Write cache: enabled, read cache: enabled, doesn't support DPO or FUA
sda: sda1
sd 1:0:0:0: [sda] Attached SCSI removable disk
    
```

Step 2:

Command	# fdisk -l /dev/sda
Description	Check the USB Flash Disk by [fdisk] command
Test Phenomenon	After executing the command, we can see the USB Flash Disk device node from the printed, please refer to the following picture.
Reference	 <pre> root@imx6ulevk:/usr/test# fdisk -l /dev/sda Disk /dev/sda: 30.9 GB, 30932992000 bytes 185 heads, 8 sectors/track, 40821 cylinders Units = cylinders of 1480 * 512 = 757760 bytes Device Boot Start End Blocks Id System /dev/sda1 911 40822 29534336 c Win95 FAT32 (LBA) </pre>

Step 3 :

Command	#df -h
Description	Check whether the USB Flash Disk be automatically mounted.
Test Phenomenon	After executing the command, we can see the mounting information from the printed information.
Reference	<pre> root@imx6ulevk:/usr/test# df -h Filesystem Size Used Available Use% Mounted on /dev/root 3.6G 145.0M 3.2G 4% / devtmpfs 247.6M 0 247.6M 0% /dev tmpfs 247.8M 156.0K 247.6M 0% /run tmpfs 247.8M 56.0K 247.6M 0% /var/lib/containers/mountfs-tmpfs /dev/mmcblk1p1 3.6G 145.0M 3.2G 4% /media/mmcblk1p1 /dev/sda1 28.2G 17.9G 10.3G 64% /media/sda1 </pre>

Step 4:

Command	# ls /media/sda1
Description	Check the contents in USB Flash Disk
Test Phenomenon	After executing the command, we can see the contents of TF card from the mounted directory of the printed information , please refer to the following picture.
Reference	<pre> root@imx6ulevk:/usr/test# ls /media/sda1 111 imx.c 1111 imx6ul 1?? imx6ul-sim7100xx?wifi ?? 20170214 imx6ul??? 485_test include ?? kernel.tar.gz ?? lib.c ?? lib.h ??(W&EVDO)????Linux????? V1.1.pdf linux-2.6.30 ??-IMX6?? Linux ?????????v1.15 -2017.1.20.doc linux-3.2.0-psp04.06.00.11.yesq.tar.gz ??_docx linux-imx-4.1.15-r0_lpw ?? linux??? ??-IMX6UL?? Linux ?????????v1.0 -2017.1.23.doc main </pre>

2.5.3 FAQ

- After inserting USB Flash Disk, without any printed information , it can recognize but shows reading and writing error,

Solution:

1.Check the kernel configuration [make menuconfig], whether it is same as the description

of *IAC-IMX6UL-Kit BSP development instruction*.

2.Maybe the USB Flash Disk is damaged, we can formatting it in the PC, then try again.

3.Change another USB Flash Disk to test.

2.6 TF card testing

2.6.1 Overview

IAC-IMX6UL-KIT provides 1-ch TF Card interface (J11) for users to use.

On board TF card interface support hot plug, after plugging in TF card, system will recognize TF card and print the relative information of TF Card.

Generate device node and partition node in [/dev] directory. Then system will automatically mount all partition to [/media/] directory. Through read and write corresponded file of directory to judge whether the interface is normal or not.

2.6.2 Test steps

The following test process will take example of TF card with only one partition. If several partitions, please use the same test method.

Step 1 :

Plug in a 64G TF card and generate device node [/dev/mmcblk0]. (Partition [N] corresponds to partition device node [/dev/mmcblk0pN]) , it prints information as follow:

```
root@imx6ulevk:/usr/test# mmc0: host does not support reading read-only switch, assuming write-enable
mmc0: new high speed SDXC card at address 0007
mmcblk0: mmc0:0007 SD64G 58.2 GiB
mmcblk0: p1
```

Step 2 :

Command	# fdisk -l /dev/mmcblk0
Description	Check the TF information by [fdisk] command and mount.
Test Phenomenon	After executing the command, we can see the contents of TF card information from the printed information, please refer to the following

	picture.
Reference	<pre> root@imx6ulevk:/usr/test# fdisk -l /dev/mmcblk0 Disk /dev/mmcblk0: 62.5 GB, 62537072640 bytes 255 heads, 63 sectors/track, 7603 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Device Boot Start End Blocks Id System /dev/mmcblk0p1 1 7603 61071066 7 HPFS/NTFS </pre>

Step 3:

Command	# df -h
Description	Check the specific mount directory
Test Phenomenon	After executing the command, we can see the mount directory of TF card from the printed information, please refer to the following picture.
Reference	<pre> root@imx6ulevk:/usr/test# df -h Filesystem Size Used Available Use% Mounted on /dev/root 3.6G 145.0M 3.2G 4% / devtmpfs 247.6M 0 247.6M 0% /dev tmpfs 247.8M 152.0K 247.7M 0% /run tmpfs 247.8M 52.0K 247.8M 0% /var/volatile /dev/mmcblk1p1 3.6G 145.0M 3.2G 4% /media/mmcblk1p1 /dev/mmcblk0p1 58.2G 297.1M 58.0G 0% /media/mmcblk0p1 </pre>

Step 4:

Command	# ls /media/mmcblk0p1
Description	Check the contents of TF card
Test Phenomenon	After executing the command, we can see the contents of TF card from the printed information, please refer the following picture.
Reference	<pre> root@imx6ulevk:/usr/test# ls /media/mmcblk0p1 ntfs: (device mmcblk0p1): ntfs_ucstons(): Unicode name contains characters that cannot be u might want to try to use the mount option nls=utf8. ntfs: (device mmcblk0p1): ntfs_filldir(): Skipping unrepresentable inode 0x31. Android README.txt.txt rootfs.yaffs2 uEnvz.txt DCIM Records u-boot uImage LOST.DIR System Volume Information u-boot-spl.bin MLO log u-boot.bin MLO.spi proudlink u-boot.img </pre>

2.6.3 FAQ

- After plugging in SD Card, without any print information, it can recognize but shows reading

and writing error.

Solution:

- 1.Maybe the USB Flash Disk is damaged, we can formatting it in the PC, then try again.
- 2.Change another USB Flash Disk to test.
- 3.Check the kernel configuration [make menuconfig],whether it is same as the description of *IAC-IMX6UL-Kit BSP development instruction*.

2.7 GPIO testing

2.7.1 Overview

This test aims at the following GPIO PIN Definition:

J20[IMX_GPIO4_23, IMX_GPIO4_24, IMX_GPIO4_25 IMX_GPIO4_26, IMX_GPIO4_27, IMX_GPIO4_28, IMX_GPIO1_18]

[Gpio_test 0] test the situation when gpio do not have external connection.set all pin to low level, then set them to high level, through external measure [gpio] actual level to confirm whether gpio is normal or not.

[Gpio_test 1] will read external level signal directly, user can compare read level data and connected level data to confirm whether [gpio] is normal or not.

2.7.2 Test steps

Step 1 :

Command	# ls /dev
Description	Check whether there are available device nodes, if there is no GPIO device node, please check the manual 2.7.3
Test Phenomenon	After executing the command, we can see the GPIO device node from the printed information, please refer the following picture.

Reference	<pre> root@imx6ulevk:/usr/test# ls /dev audio loop1 ptp0 random tty21 tty42 tty63 autofs loop2 ptp1 rtc tty22 tty43 tty7 block loop3 pts gpio tty23 tty44 tty8 bus loop4 ppx_device shm tty24 tty45 tty9 char loop5 qiyang_huzzel tty25 tty46 tty46 tty63 console loop6 qiyang_imx6_gpio tty26 tty47 tty47 tty63 cpu_dma_latency loop7 qy_watchdog stdin tty27 tty48 tty63 disk mem ram0 stdout tty28 tty49 tty63 dri memory_bandwidth ram1 tty tty29 tty50 tty63 dsp mmcblk1 ram0 tty0 tty30 tty51 tty63 fb0 mmcblk1boot0 ram11 tty1 tty31 tty52 tty63 fd mmcblk1boot1 ram12 tty10 tty32 tty53 tty63 full mmcblk1pl ram13 tty11 tty33 tty54 tty63 fuse mmcblk1rmb ram14 tty12 tty34 tty55 tty63 hwrng mtab ram15 tty13 tty35 tty56 tty63 i2c-0 mxs_asrc ram2 tty14 tty36 tty57 tty63 i2c-1 network_latency ram3 tty15 tty37 tty58 tty63 initctl network_throughput ram4 tty16 tty38 tty59 tty63 input null ram5 tty17 tty39 tty60 tty63 kmsg ppp ram6 tty18 tty40 tty61 tty63 log pps0 ram7 tty19 tty41 tty62 tty63 loop-control pps1 ram8 tty20 tty42 tty63 tty63 loop0 ptmx ram9 tty21 tty43 tty63 tty63 </pre>
-----------	--

Step2 :

Command	<code># ./gpio_test</code>
Description	Run the test program [gpio_test].
Test Phenomenon	After executing the command, we can see the hint of how to set and get the GPIO information from the printed information. please refer the following picture
Reference	<pre> root@imx6ulevk:/usr/test# ./gpio_test Invalid arguments! Usage: ./gpio_test [0,1] 0 -- set gpio level. 1 -- get gpio level. </pre>

Remark:

Refer to step2, /gpio_test 0 is to set the gpio level, /gpio_test 1 is to get the gpio level.

Step3 :

Command	<code># ./gpio_test 1</code>
Description	Get the GPIO level on the information from Step1
Test Phenomenon	After executing the command, we can see the GPIO status from the printed information, please refer to the following pictur.

Reference	<pre> root@imx6ulevk:/usr/test# ./gpio_test 1 QY-IMX6UL-v1.x Gpio Start Testing#####pin = 119, level = 0 ... get gpio 'IMX_GPI04_23' level '0'#####pin = 120, level = 0 get gpio 'IMX_GPI04_24' level '0'#####pin = 121, level = 0 get gpio 'IMX_GPI04_25' level '0'#####pin = 122, level = 0 get gpio 'IMX_GPI04_26' level '0'#####pin = 123, level = 0 get gpio 'IMX_GPI04_27' level '0'#####pin = 124, level = 0 get gpio 'IMX_GPI04_28' level '0'#####pin = 18, level = 0 get gpio 'IMX_GPI01_18' level '0' </pre>
-----------	---

Step4 :

Command	# ./gpio_test 0
Description	Set the GPIO level according to the hinted information from Step1
Test Phenomenon	After executing the command, we can see the GPIO setting information from the printed information. please refer to the following picture.
Reference	<pre> root@imx6ulevk:/usr/test# ./gpio_test 0 QY-IMX6UL-v1.x Gpio Start Testing ... set gpio 'IMX_GPI04_23' level '0' set gpio 'IMX_GPI04_24' level '0' set gpio 'IMX_GPI04_25' level '0' set gpio 'IMX_GPI04_26' level '0' set gpio 'IMX_GPI04_27' level '0' set gpio 'IMX_GPI04_28' level '0' set gpio 'IMX_GPI01_18' level '0' Gpios is output low level, now you can measure each pin! Press the ENTER after measure each pins! set gpio 'IMX_GPI04_23' level '1' set gpio 'IMX_GPI04_24' level '1' set gpio 'IMX_GPI04_25' level '1' set gpio 'IMX_GPI04_26' level '1' set gpio 'IMX_GPI04_27' level '1' set gpio 'IMX_GPI04_28' level '1' set gpio 'IMX_GPI01_18' level '1' Gpios is output high level, now you can measure each pin! Press the ENTER after measure each pins! Gpio test OK! </pre>

Remark: as the step4, set the every GPIO to low level, after pressing the enter, set all the GPIO to high level, press enter again, show testing is ok.

2.7.3 FAQ

- [/dev] direcotry don't have the [qiyang_imx6_gpio] nod.

Solution:

- 1、 Check whether add the GPIO drive program.
- 2、 Check the kernel configuration [make menuconfig], whether it is same as the description of *IAC-IMX6UL-Kit BSP development instruction* .

2.8 Ethernet test

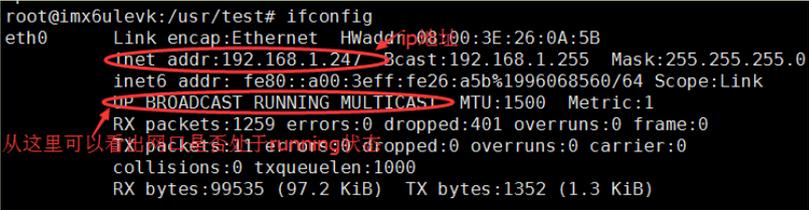
2.8.1 Overview

IAC-IMX6UL-KIT development board has 2-ch 100Mbps Ethernet network onboard(J2, J3). Using [ping] command to test whether it could communicate with network.

2.8.2 Test steps

Here is testing the eth0 as a sample. Others testing is the same.

Step1 : Connect network cable to [eth0](J2)

Command	# ifconfig
Description	Check whether the eth0 is in RUNNING status
Test Phenomenon	After executing the command, we can see the eth0 status from the printed information. please refer to the following picture.
Reference	

Step 2 :

Command	# ping -I eth0 www.baidu.com
---------	-------------------------------------

Description	Communicate with external network
Test Phenomenon	After executing the command, we can see the communication information with outer network from the printed information. please refer to the following picture.
Reference	<pre> root@imx6ulevk:/usr/test# ping -I eth0 www.baidu.com PING www.baidu.com (119.75.216.20): 56 data bytes 64 bytes from 119.75.216.20: seq=0 ttl=53 time=60.302 ms 64 bytes from 119.75.216.20: seq=1 ttl=53 time=60.900 ms 64 bytes from 119.75.216.20: seq=2 ttl=53 time=60.308 ms 64 bytes from 119.75.216.20: seq=3 ttl=53 time=60.258 ms 64 bytes from 119.75.216.20: seq=4 ttl=53 time=60.197 ms 64 bytes from 119.75.216.20: seq=5 ttl=53 time=60.165 ms 64 bytes from 119.75.216.20: seq=6 ttl=53 time=60.607 ms 64 bytes from 119.75.216.20: seq=7 ttl=53 time=60.514 ms 64 bytes from 119.75.216.20: seq=8 ttl=53 time=60.110 ms 64 bytes from 119.75.216.20: seq=9 ttl=53 time=60.629 ms 64 bytes from 119.75.216.20: seq=10 ttl=53 time=60.936 ms --- www.baidu.com ping statistics --- 11 packets transmitted, 11 packets received, 0% packet loss round-trip min/avg/max = 60.110/60.447/60.936 ms </pre>

Remark: Take a reference of step2, it means the Eth 1 works normally. Packet loss means lost data/transmitting data.

2.8.3 FAQ

- After connecting the network LAN, there is no information of eth0 or eth1 become ready

Solution:

- 1、Check the kernel configuration [make menuconfig], whether it is same as the description of *IAC-IMX6UL-Kit BSP development instruction*.
- 2、Check *IAC-IMX6UL-Kit BSP development instruction* whether has been modified into the actual situation.

2.9 3G/4G Module Test

2.9.1 Overview

PCIE interface (J7) of IAC-IMX6UL-KIT could connect to 3G/4G module

By dial-up networking, check whether it could communicate with external network.

2.9.2 Test steps

Step1 :

Connect the 3G/4Gmodule to the J7. The printed information are as follows:

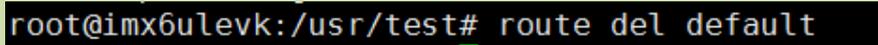
```
usb 1-1.3: GSM modem (1-port) converter now attached to ttyUSB0
usb-storage 1-1.3:1.1: USB Mass Storage device detected
option 1-1.3:1.1: GSM modem (1-port) converter detected
usb 1-1.3: GSM modem (1-port) converter now attached to ttyUSB1
usb-storage 1-1.3:1.2: USB Mass Storage device detected
option 1-1.3:1.2: GSM modem (1-port) converter detected
usb 1-1.3: GSM modem (1-port) converter now attached to ttyUSB2
usb-storage 1-1.3:1.3: USB Mass Storage device detected
option 1-1.3:1.3: GSM modem (1-port) converter detected
usb 1-1.3: GSM modem (1-port) converter now attached to ttyUSB3
usb-storage 1-1.3:1.4: USB Mass Storage device detected
option 1-1.3:1.4: GSM modem (1-port) converter detected
usb 1-1.3: GSM modem (1-port) converter now attached to ttyUSB4
```

Step 2 :

Insert the according mobile phone card to slot(J30) on back of development board. (Here use telecom card as a sample)

Note: the testing way is same for 3G or 4G. Before testing, Please confirm the model of the 3G/4G module, such as Unicom, CMCC or Telecom. Then insert the SIM card to slot(J30), after preparation, we start to connect.

Step3 :

Command	# route del default
Description	Delete the default routing
Test Phenomenon	After executing the command, we can see the hinted information of deleting routing from the printed information. Please refer to the following picture
Reference	

Remark: If appear route:SIOCDELRT: No such process, Ignore it, and do next step.

Step 4 :

Command	# ln -sf /etc/ppp/resolv.conf /etc/resolv.conf
---------	---

Description	Let /etc/resolv.conf soft link to /etc/ppp/resolv.conf
Test Phenomenon	None
Reference	<code>root@imx6ulevk:/usr/test# ln -sf /etc/ppp/resolv.conf /etc/resolv.conf</code>

Step 5 :

Command	# pppd call evdo &
Description	Dial-up, show the following picture, then it is successful.
Test Phenomenon	After executing the command, we can see the Dial-up process and 3G information from the printed information. Please refer to the following picture.
Reference	<pre>Serial connection established. using channel 1 using interface ppp0 Connect: ppp0 <-> /dev/ttyUSB2 rcvd [LCP ConfReq id=0x1 <mru 1448> <asynmap 0x0> <auth chap MD5> <magic 0x1dc8e496> <pcomp> <accomp>] sent [LCP ConfReq id=0x1 <asynmap 0x0> <magic 0x90708ea6> <pcomp> <accomp>] sent [LCP ConfAck id=0x1 <mru 1448> <asynmap 0x0> <auth chap MD5> <magic 0x1dc8e496> <pcomp> <accomp>] rcvd [LCP ConfAck id=0x1 <asynmap 0x0> <magic 0x90708ea6> <pcomp> <accomp>] rcvd [CHAP Challenge id=0x2 <04ed866a5478752caf22756e98e2370>, name = "ericsson.com"] sent [CHAP Response id=0x2 <2c7990b04153596ba9d635fca3351fda>, name = "ctnet@mycdma.cn"] rcvd [CHAP Challenge id=0x3 <f5f1a571dd07a1404b6916add9cb304>, name = "ericsson.com"] sent [CHAP Response id=0x3 <22b06e59086f7573960e8b7d64605846>, name = "ctnet@mycdma.cn"] rcvd [CHAP Success id=0x3 <"000">] CHAP authentication succeeded: CHAP authentication succeeded sent [CCP ConfReq id=0x1 <deflate 15> <deflate(oid#) 15> <bsd v1 15>] sent [IPCP ConfReq id=0x1 <compress VJ 0f 01> <addr 0.0.0.0> <ms-dns1 0.0.0.0> <ms-dns3 0.0.0.0>] rcvd [LCP ProtReq id=0x0 00 fd 01 01 00 0f 1a 04 78 00 18 04 78 00 15 03 2f] Protocol-Reject for 'Compression Control Protocol' (0x80fd) received sent [IPCP ConfReq id=0x4 <compress VJ 07 00> <addr 115.168.75.246>] rcvd [IPCP ConfAck id=0x4 <compress VJ 07 00> <addr 115.168.75.246>] rcvd [IPCP ConfReq id=0x1 <compress VJ 07 00> <addr 10.26.11.109> <ms-dns1 202.101.172.37> <ms-dns3 202.101.173.157>] sent [IPCP ConfReq id=0x2 <compress VJ 07 00> <addr 10.26.11.109> <ms-dns1 202.101.172.37> <ms-dns3 202.101.173.157>] rcvd [IPCP ConfAck id=0x2 <compress VJ 07 00> <addr 10.26.11.109> <ms-dns1 202.101.172.37> <ms-dns3 202.101.173.157>] not replacing existing default route via 192.168.1.1 local IP address 10.26.11.109 remote IP address 115.168.75.246 primary DNS address 202.101.172.37 secondary DNS address 202.101.173.157 Script /etc/ppp/ip-up started (pid 374) Script /etc/ppp/ip-up finished (pid 374), status = 0x0</pre>

Remark: the way of Dial-up is different:

Unicom:# pppd call wcdma &

CMCC: # pppd call tdsdma &

Telecom:# pppd call evdo &

Step 6 :

Command	# ifconfig
Description	Check the ppp0 network node
Test Phenomenon	After executing the command, we can see the information of ppp0 protocol from the printed information. Please refer to the following

	picture.
Reference	<pre> ppp0 Link encap:Point-to-Point Protocol inet addr:10.26.11.109 P-t-P:115.168.75.246 Mask:255.255.255.255 UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1448 Metric:1 RX packets:3 errors:1 dropped:0 overruns:0 frame:0 TX packets:4 errors:0 dropped:0 overruns:0 carrier:0 collisions:0 txqueuelen:3 RX bytes:72 (72.0 B) TX bytes:87 (87.0 B) </pre>

Step 7 :

Command	# route add default gw 10.26.11.109^①
Description	Add the default gateway
Test Phenomenon	None
Reference	<pre>root@imx6ulevk:~# route add default gw 10.26.11.109</pre>

Step8 :

Command	# ping -I ppp0 www.baidu.com
Description	Connect with external network
Test Phenomenon	After executing the command, we can see the information of external network communication from the printed information. Please refer to the following picture.

Remark: ① This IP address is same as the printed ip address of the 3G/4G which uses [ifconfig] command from Step 6 of this Charter.

Reference	<pre> ^Croot@imx6ulevk:~# ping -I ppp0 www.baidu.com PING www.baidu.com (115.239.211.112): 56 data bytes 64 bytes from 115.239.211.112: seq=0 ttl=51 time=62.872 ms 64 bytes from 115.239.211.112: seq=1 ttl=51 time=60.964 ms 64 bytes from 115.239.211.112: seq=2 ttl=51 time=59.313 ms 64 bytes from 115.239.211.112: seq=3 ttl=51 time=58.727 ms 64 bytes from 115.239.211.112: seq=4 ttl=51 time=78.255 ms 64 bytes from 115.239.211.112: seq=5 ttl=51 time=65.461 ms 64 bytes from 115.239.211.112: seq=6 ttl=51 time=63.704 ms 64 bytes from 115.239.211.112: seq=7 ttl=51 time=63.159 ms 64 bytes from 115.239.211.112: seq=8 ttl=51 time=62.638 ms 64 bytes from 115.239.211.112: seq=9 ttl=51 time=61.951 ms 64 bytes from 115.239.211.112: seq=10 ttl=51 time=61.551 ms --- www.baidu.com ping statistics --- 11 packets transmitted, 11 packets received, 0% packet loss round-trip min/avg/max = 58.727/63.508/78.255 ms </pre>
-----------	---

2.9.3 FAQ

- When 3G/4G module is connected to PCIE interface, there is no reaction.

Solution

- 1、 Check the kernel configuration[make menuconfig], whether it is same as the description of *IAC-IMX6UL-Kit BSP development instruction*.
- 2、 Check whether add the node data of 3G/4G module in kernel directory [drivers/usb/serial/option.c].

- When dial-up, the serial port interrupt and appears below information

```

root@QY-IMX65:~# pppd call evdo
timeout set to 15 seconds
abort on (DELAYED)
abort on (BUSY)
abort on (ERROR)
abort on (NO DIALTONE)
abort on (NO CARRIER)
timeout set to 15 seconds
send (AT^M)
expect (OK)

```

Picture 9-1

```

send (ATDT#777^M)
expect (CONNECT)
^M
^M
NO CARRIER
-- failed
Failed (NO CARRIER)
Connect script failed

```

Picture 9-2

Solution:

- 1.Modify [vi /etc/ppp/peers/evdo] as follow:

```

debug
nodetach
lock
/dev/ttyUSB0
115200
user "card"
password "card"
crtscts
show-password
usepeerdns
noauth
noipdefault
novj
novjccomp
noccp
ipcp-accept-local
ipcp-accept-remote
lcp-echo-interval 10
lcp-echo-failure 4
persist
connect 'chat -s -v -f /etc/ppp/peers/evdo-chat-connect'
    
```

Picture 9-3

Modify the red circle on Picture 9-3 into `ttyUSB0~ttyUSB4`, then redial until find the correct node.

2.10 Wifi Test

2.10.1 Overview

IAC-IMX6UL-KIT supports RTL8723BU(USB wifi module.) Communicate with network through the Wifi module.

2.10.2 Test steps:

Step1:

Command	#insmod 8723bu.ko
Description	Load 8723bu, add wifi module to support
Test Phenomenon	After executing the command, we can see the process and information of loading RTL8723BU USB wifi module from the printed information. (If not successful, please check the manual 2.10.3) Please refer to the following picture.

Reference	<pre> RTL871X: init_channel_set ChannelPlan ID 20 Chan num:13 RTL871X: rtw_alloc_macid(eth%d) if1, hwaddr:ff:ff:ff:ff:ff:ff macid:1 RTL871X: Init_ODM_ComInfo_8723b(): fab_ver=0 cut_ver=0 RTL871X: can't get autopm: RTL871X: rtw_macaddr_cfg MAC Address = e0:b9:4d:f9:78:4c RTL871X: bDriverStopped:1, bSurpriseRemoved:0, bup:0, hw_init_completed RTL871X: rtw_ndev_init(wlan0) wifi模块加载成功 RTL871X: _rtw_drv_register_netdev, MAC Address (if1) = e0:b9:4d:f9:78:4c usbcore: registered new interface driver rtl8723bu RTL871X: module init ret=0 </pre>
-----------	---

Step 2 :

Command	#wpa_supplicant -B -i wlan0 -D wext -c /etc/wpa_supplicant.conf
Description	Connect WIFI
Test Phenomenon	After executing the command, we can see whether successfully connect WIFI from the printed information. Please refer to the following picture.
Reference	<pre> IPV6: ADDRCONF(NETDEV_CHANGE): wlan0: link becomes ready RTL871X: HW_VAR_BASIC_RATE: 0x15f -> 0x15T -> 0x15T RTL871X: WMM(0): 0, a42b RTL871X: WMM(1): 0, a44f RTL871X: WMM(2): 0, 5e4322 RTL871X: WMM(3): 0, 2f3222 RTL871X: wmm_para_seq(0): 0 RTL871X: wmm_para_seq(1): 1 RTL871X: wmm_para_seq(2): 2 RTL871X: wmm_para_seq(3): 3 RTL871X: HTOnAssocRsp RTL871X: ==>phy_SwChnl8723B: Channel = 9 RTL871X: UpdateHaIRAMask8723B(): mac_id=0 rssi_level=0 RTL871X: UpdateHaIRAMask8723B => mac_id:0, networkType:0x0b, mask:0x000fffff => rssi_level:0, rate_bitmap:0x000ff015 RTL871X: rtl8723b_set_FwMacIdConfig_cmd(): mac_id=0 raid=0x1 bw=1 mask=0xff015 RTL871X: UpdateHaIRAMask8723B(): mac_id=0 raid=0x1 bw=1 mask=0xff015 init_rate=0x13 RTL871X: rtl8723b_set_FwMediaStatusRpt_cmd(): mstatus = 1 macid=0 RTL871X: +rtl8723b_download_rsvd_page(wlan0): iface_type=0 mstatus(1) RTL871X: rtl8723b_set_FwRsvdPagePkt: Set Rsvd page location to Fw ,TotalPacketLen(842), TotalPageNum(6) RTL871X: 8723BRsvdPageLoc: ProbeRsp=0 PsPoll=2 Null=3 QoSNull=4 BTNull=5 RTL871X: wlan0: 1 DL Rsvd page success! DLBcnCount:1, poll:1 RTL871X: =>mLmeext_joinbss_event_callback RTL871X: send eapol packet RTL871X: rtl8723b_fill_default_txdesc(wlan0): SP Packet(0x888E) rate=0x0 RTL871X: send eapol packet RTL871X: rtl8723b_fill_default_txdesc(wlan0): SP Packet(0x888E) rate=0x0 RTL871X: ~~~~set sta key:unicastkey RTL871X: set pairwise key camid:4, addr:ec:26:ca:5e:e7:b8, kid:0, type:AES RTL871X: ~~~~set sta key:groupkey RTL871X: ==> rtw_set_key_algorithm(4),keyid(1),key_mask(0) RTL871X: set group key camid:5, addr:ec:26:ca:5e:e7:b8, kid:1, type:AES RTL871X: SetHwReg8723B, 6438, RCR= 700078ff RTL871X: UpdateHaIRAMask8723B(): mac_id=0 rssi_level=2 RTL871X: UpdateHaIRAMask8723B => mac_id:0, networkType:0x0b, mask:0x000fffff => rssi_level:2, rate_bitmap:0x000ff000 RTL871X: rtl8723b_set_FwMacIdConfig_cmd(): mac_id=0 raid=0x1 bw=1 mask=0xff000 RTL871X: UpdateHaIRAMask8723B(): mac_id=0 raid=0x1 bw=1 mask=0xff000 init_rate=0x13 </pre> <p style="text-align: right; color: red;">说明wifi连接成功</p>

Step3 :

Command	# ifconfig
---------	-------------------

Description	up Check whether the wlan0 is up.
Test Phenomenon	After executing the command, we can see whether wlan0 is running from the printed information. Please refer to the following picture.
Reference	<pre>wlan0 Link encap:Ethernet HWaddr E0:B9:4D:F9:78:4C inet6 addr: fe80::e2b9:4dff:fe9:784c%1995912912/64 Scope:Link UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1 RX packets:46 errors:0 dropped:7 overruns:0 frame:0 TX packets:9 errors:0 dropped:1 overruns:0 carrier:0 collisions:0 txqueuelen:1000 RX bytes:3822 (3.7 KiB) TX bytes:1082 (1.0 KiB)</pre>

Step 4 :

Command	# vi /etc/wpa_supplicant.conf
Description	Modify the WIFI configuration files
Test Phenomenon	After executing the command, we can see the related information of WIFI from the printed information. Please refer to the following picture.
Reference	<pre># WPA-PSK/TKIP ctrl_interface=/var/run/wpa_supplicant network={ ssid="QY-ZSH" scan_ssid=1 key_mgmt=WPA-EAP WPA-PSK IEEE8021X NONE pairwise=CCMP TKIP group=CCMP TKIP WEP104 WEP40 psk="qiyangtech" }</pre> <p>根据实际情况修改wif的名字</p> <p>根据实际情况修改wif密码</p>

Step 5 :

Command	# ifconfig wlan0 192.168.0.135 # ifconfig
Description	Configure wlan IP, and check whether successful
Test Phenomenon	After executing the command, we can see the configuration IP address from the printed information. Please refer to the following

	picture.
Reference	<pre> root@imx6ulevk:/usr/test# ifconfig wlan0 192.168.0.135 root@imx6ulevk:/usr/test# ifconfig wlan0 Link encap:Ethernet HWaddr E0:B9:4D:F9:78:4C inet addr:192.168.0.135 Bcast:192.168.0.255 Mask:255.255.255.0 inet6 addr: fe80::e2d9:4dff:fe9:784c%1995560656/64 Scope:Link UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1 RX packets:69 errors:0 dropped:7 overruns:0 frame:0 TX packets:9 errors:0 dropped:1 overruns:0 carrier:0 collisions:0 txqueuelen:1000 RX bytes:6647 (6.4 KiB) TX bytes:1082 (1.0 KiB) </pre>

Step 6 :

Command	# route add default gw 192.168.0.1
Description	Set the gateway
Test Phenomenon	None
Reference	<pre> root@imx6ulevk:/usr/test# route add default gw 192.168.0.1 </pre>

Step 7 :

Command	# ping 192.168.0.1
Description	Ping gateway
Test Phenomenon	After executing the command, we can see the communication information with outer network from the printed information. Please refer to the following picture.

Reference

```

root@imx6ulevk:/usr/test# ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1): 5RTL871X: rtl8723b_fill_default_txdesc(wlan0): SP Packet(0x0806)
6 data bytes
64 bytes from 192.168.0.1: seq=0 ttl=64 time=85.156 ms
RTL871X: OnAction_back
RTL871X: OnAction_back, action=0
RTL871X: issue_action_BA, category=3, action=1, status=0, rxbuf_sz=64
64 bytes from 192.168.0.1: seq=1 ttl=64 time=32.904 ms
64 bytes from 192.168.0.1: seq=2 ttl=64 time=42.271 ms
64 bytes from 192.168.0.1: seq=3 ttl=64 time=5.871 ms
64 bytes from 192.168.0.1: seq=4 ttl=64 time=8.122 ms
64 bytes from 192.168.0.1: seq=5 ttl=64 time=8.566 ms
RTL871X: rtl8723b_fill_default_txdesc(wlan0): SP Packet(0x0806) rate=0x0
64 bytes from 192.168.0.1: seq=6 ttl=64 time=2.660 ms
64 bytes from 192.168.0.1: seq=7 ttl=64 time=11.403 ms
64 bytes from 192.168.0.1: seq=8 ttl=64 time=6.302 ms
64 bytes from 192.168.0.1: seq=9 ttl=64 time=4.340 ms
64 bytes from 192.168.0.1: seq=10 ttl=64 time=7.254 ms
64 bytes from 192.168.0.1: seq=11 ttl=64 time=3.858 ms
^C
--- 192.168.0.1 ping statistics ---
12 packets transmitted, 12 packets received, 0% packet loss
round-trip min/avg/max = 2.660/18.225/85.156 ms

```

2.10.3 FAQ

- Load 8723bu, but not successful.

Solution

1.Check the kernel configuration[make menuconfig], whether it is same as the description of *IAC-IMX6UL-Kit BSP development instruction*.

- After inserting the WIFI module, shows: link is not ready

Solution:

1.Plug and insert the WIFI module, or restart the development board and then insert the WIFI module.

- When # ping **www.baidu.com**, show bad address.

Solution:

1.Fail to parse the DNS domain name, modify the file /etc/resolv.conf, modify nameserver IP to gateway IP 192.168.0.1.



```

# Generated by Connection Manager
nameserver 192.168.0.1
nameserver ::1
~

```

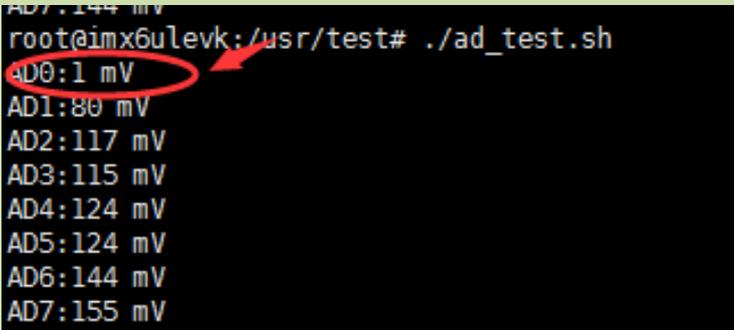
2.11 AD Test

2.11.1 Overview

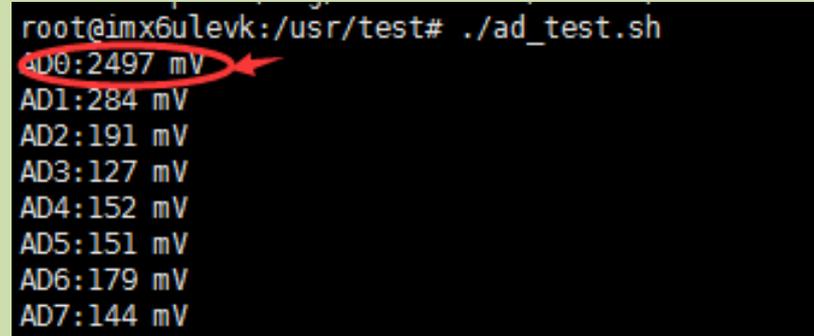
IAC-IMX6UL-KIT has 8-ch AD input [J13], 12 bit precision, the range of input voltage is 0-2.5V

2.11.2 Test Steps

Step 1: AD1 connect the ground, other hung up.

Command	# ./ad_test.sh
Description	Run AD application program
Test Phenomenon	After executing the command, we can see the voltage values of AD1 channel from the printed information. Please refer to the following picture.
Reference	 <pre> AD7:144 mV root@imx6ulevk:/usr/test# ./ad_test.sh AD0:1 mV AD1:80 mV AD2:117 mV AD3:115 mV AD4:124 mV AD5:124 mV AD6:144 mV AD7:155 mV </pre>

Step 2: AD1 channel connect the 2.5V, others hung up.

Command	# ./ad_test.sh
Description	Run AD application program
Test Phenomenon	After executing the command, we can see the voltage values of AD1 channel from the printed information. Please refer to the following picture.
Reference	 <pre> root@imx6ulevk:/usr/test# ./ad_test.sh AD0:2497 mV AD1:284 mV AD2:191 mV AD3:127 mV AD4:152 mV AD5:151 mV AD6:179 mV AD7:144 mV </pre>

2.11.3 FAQ

- Run ad_test.sh program, but with error.

Solution:

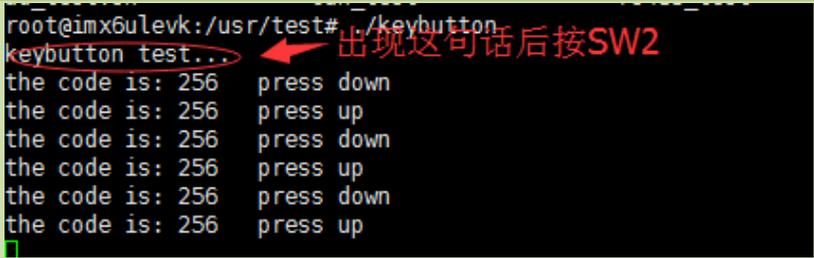
1.Check the kernel configuration [make menuconfig], whether it is same as the description of *IAC-IMX6UL-Kit BSP development instruction*.

2.12 Button Test

2.12.1 Overview

[SW2] is used as key button on the *IAC-IMX6UL-KIT development board*.

2.12.2 Test steps

Command	# ./keybutton
Description	Run AD application program
Test Phenomenon	After executing the command, we can see the information of keybutton test from the printed information. Please refer to the following picture.
Reference	

2.12.3 FAQ

- Run the keybutton program, but with Error.

Solution:

1.Check the kernel configuration[make menuconfig], whether it is same as the description of *IAC-IMX6UL-Kit BSP development instruction*.

2.13 CAN Test

2.13.1 Overview

IAC-IMX6UL-KIT mainboard brings out 2-ch CAN(J18 J19) to test the CAN0 and CAN1. We need to open two terminals, one is telnet terminal(for opening method of telnet terminal

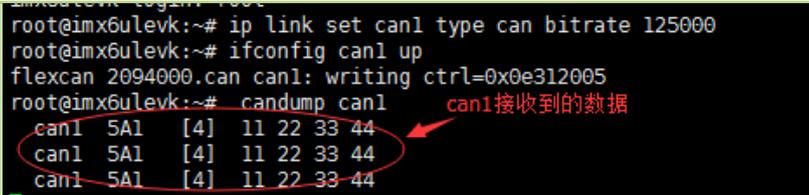
could refer to above chapter 2.4.1), the other is serial terminal(for opening method of serial terminal could refer to above chapter1. preparation). Test the CAN0 and CAN1, check whether the transmitting and receiving is normal.

2.13.2 Test steps

Step 1 :

Connect [1] pin of [J18] with [1] pin of [J19], connect [2] of [J18] with [2] pin of [J19]. Then, one CAN actions as sending end and the other CAN actions as receiving end, then switch to each other to test again.

Step 2: Terminal 1:

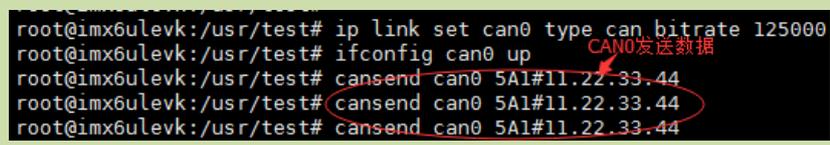
Command	<pre># ip link set can1 type can bitrate 125000 # ifconfig can1 up # candump can1^①</pre>
Description	Configure the related parameters of CAN1
Test Phenomenon	After executing the command, we can see the data information sent from CAN1 from the printed information. Please refer to the following picture.
Reference	 <pre>root@imx6ulevk:~# ip link set can1 type can bitrate 125000 root@imx6ulevk:~# ifconfig can1 up flexcan 2094000.can can1: writing ctrl=0x0e312005 root@imx6ulevk:~# candump can1 can1 5A1 [4] 11 22 33 44 can1 5A1 [4] 11 22 33 44 can1 5A1 [4] 11 22 33 44</pre>

Terminal 2:

Command	<pre># ip link set can0 type can bitrate 125000 # ifconfig can0 up # cansend can0 5A1#11.22.33.44</pre>
---------	---

Remark: ① If there is no printed information after command executing, please check whether can0 is configured well, whether CAN0 has transmitted the data.

。

Description	Configure the related parameters of CAN0
Test Phenomenon	None
Reference	 <pre> root@imx6ulevk:/usr/test# ip link set can0 type can bitrate 125000 root@imx6ulevk:/usr/test# ifconfig can0 up root@imx6ulevk:/usr/test# cansend can0 5A1#11.22.33.44 root@imx6ulevk:/usr/test# cansend can0 5A1#11.22.33.44 root@imx6ulevk:/usr/test# cansend can0 5A1#11.22.33.44 </pre>

Here we could see terminal 1 could receive the data which transmitted from terminal 2. Then switch to each other, that is, CAN0 receive the data which is transmitted from CAN1.

2.13.3 FAQ

- CAN cannot receive and transmit normally

Solution:

- 1、 Use [ifconfig] to check whether [can0] is [up]
- 2、 Check whether the 2* CAN hardware connection is right.

- If using [ifconfig can0 up], but failed.

Solution

- 1、 Please check whether kernel configuration option enables [can0] function.

- If transmit successfully, but the opposite terminal cannot receive.

Solution

- 1、 Please check whether [bitrate] two CAN are set consistently.

2.14 Audio Test

2.14.1 Overview

IAC-IMX6UL-KIT mainboard supports audio function by software decoding. The file system provides [GPLAY] tool to support audio play.

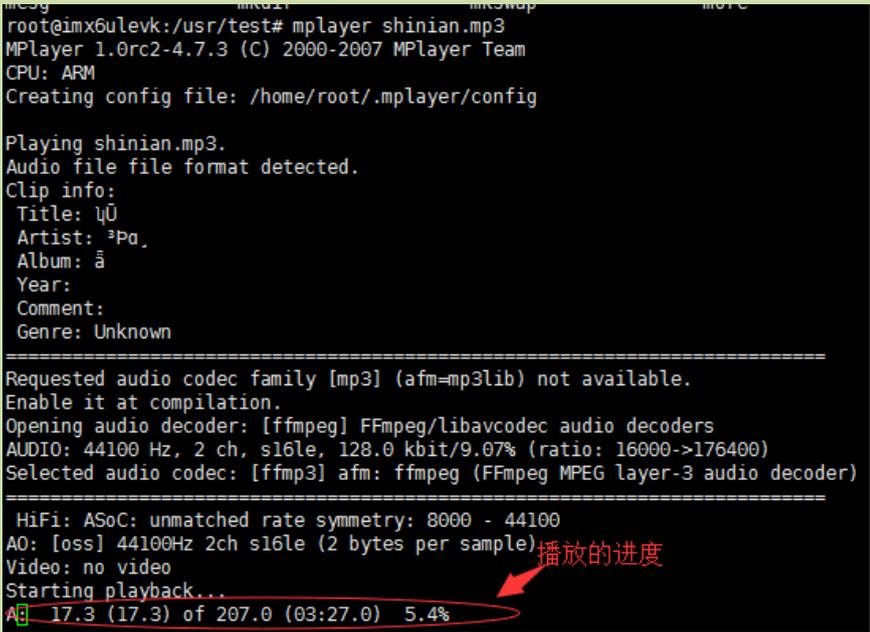
Play the audio by [gplay-1.0] command, please connect external earphone or sound equipment to [J4]

2.14.2 Test Steps

We provide you an audio test file[shinian.mp3], It is in [/usr/test] directory. It could be directly played to make test.

```
root@qy_mx6ul:/usr/test# ls
8723bu.ko          buzzer_test      rtc_test          watchdog_notfeed_test
QiYang_Imx6S_Qt_test  can_test        serial_test
ad_test.sh         gpio_test       shinian.mp3
backlight_test     keybutton       watchdog_feed_test
```

Step 1:

Command	# mplay shinian.mp3
Description	Play the Audio file
Test Phenomenon	After executing the command, we can see the related information of Audio play from the printed information. Please refer to the following picture.
Reference	

You will hear music from MP3 through audio input interface.

Step 2 :

Command	# arecord test.wav
---------	---------------------------

Description	After command, insert the Microphone in J5 and speak to it, Then use [ctrl+c] to end
Test Phenomenon	Please refer to the picture
Reference	<pre>root@imx6ulevk:/usr/test# arecord /usr/test/test.wav Recording WAVE '/usr/test/test.wav' : Unsigned 8 bit, Rate 8000 Hz, Mono</pre>

Step 3 :

Command	# aplay test.wav
Description	Insert headphone in J4, check whether it has sound.
Test Phenomenon	Please refer to the picture.
Reference	<pre>root@imx6ulevk:/usr/test# aplay test.wav Playing WAVE 'test.wav' : Unsigned 8 bit, Rate 8000 Hz, Mono</pre>

2.14.3 FAQ

- Can't play the Audio

Solution

- 1、 Check the kernel configuration(make menuconfig), whether it is same as the description of IAC-IMX6UL-Kit BSP development instruction.

2.15 LCD Display and VGA Test

2.15.1 Overview

IAC-IMX6UL-KIT mainboard provides 1-ch LCD TFT display interface(J8), 1-ch VGA interface(J9).

After system boots, LCD Display or VGA will show its own penguin picture from kernel.

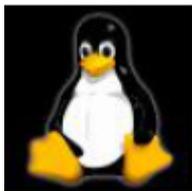
2.15.2 Test steps

Step 1:

Connect LCD or VGA according to hardware manual.

Step 2:

Power on to mainboard, after system boots. LCD or VGA will display the following picture on the left up side corner:



We can check whether the logo picture is distortion or jitter to confirm whether display normal or not, and can also run qt program to test LCD and VGA according to this chapter 2.17.

Note: Please ensure that the resolution of kernel configuration is same as the current resolution of connected LCD or VGA.

Remark: Please ensure that the configured kernel resolution is same as the current connected LCD and VGA.

2.15.3 FAQ

- Display image position and size don't match the lcd monitor

Solution

- 1、 Please confirm whether current kernel resolution is the same as current using LCD monitor data. The default resolution is 800*480.

- LCD Display has no image

Solution:

1、 Check the kernel configuration [make menuconfig], whether it is same as the description of ***IAC-IMX6UL-Kit BSP development instruction*** .

- 1、 Please check the LCD module is normal.
- 2、 Please check the connect cable is normal.

2.16 Touch Panel Test

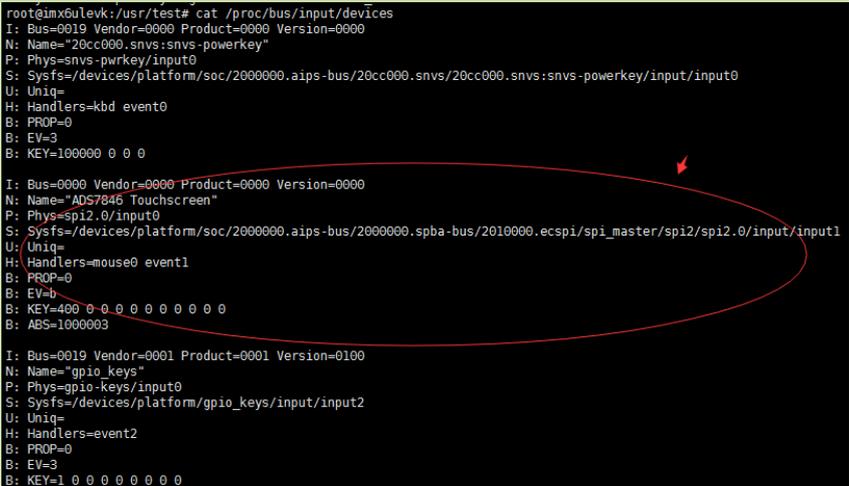
2.16.1 Overview

Connect Touch panel well, used LCD screen resolution must be the same as uboot configured LCD resolution. Our common use is 7 inch LCD touch panel, resolution is 800*480.

Use [Tslib] touch panel test tool to calibrate touch panel, after calibrating to use test tool to drag and draw line. You can find the cursor will show current touch point nearby. And it could be moved as touch panel moved.

2.16.2 Test Steps

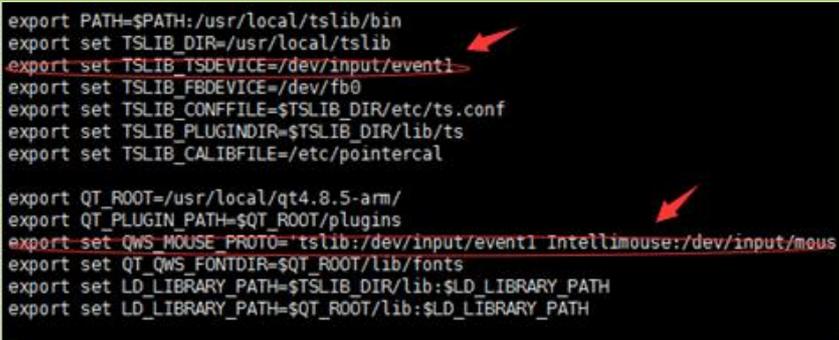
Step 1 :

Command	# cat /proc/bus/input/devices
Description	Check the configuration of touch Panel
Test Phenomenon	After executing the command, we can see the configuration information of touch panel from the printed information. Please refer to the following picture.
Reference	 <pre> root@imx6ulevk:/usr/test# cat /proc/bus/input/devices I: Bus=0019 Vendor=0000 Product=0000 Version=0000 N: Name="20cc000.snvs:snvs-powerkey" P: Phys=snvs-pwrkey/input0 S: Sysfs=/devices/platform/soc/20000000.aips-bus/20cc000.snvs/20cc000.snvs:snvs-powerkey/input/input0 U: Uniq= H: Handlers=kbd event0 B: PROP=0 B: EV=3 B: KEY=100000 0 0 0 I: Bus=0000 Vendor=0000 Product=0000 Version=0000 N: Name="ADS7846 Touchscreen" P: Phys=spi2.0/input0 S: Sysfs=/devices/platform/soc/20000000.aips-bus/20000000.sdba-bus/20100000.ecspi/spi_master/spi2/spi2.0/input/input1 U: Uniq= H: Handlers=mouse0 event1 B: PROP=0 B: EV=b B: KEY=400 0 0 0 0 0 0 0 0 0 0 B: ABS=1000003 I: Bus=0019 Vendor=0001 Product=0001 Version=0100 N: Name="gpio_keys" P: Phys=gpio-keys/input0 S: Sysfs=/devices/platform/gpio_keys/input/input2 U: Uniq= H: Handlers=event2 B: PROP=0 B: EV=3 B: KEY=1 0 0 0 0 0 0 0 0 </pre>

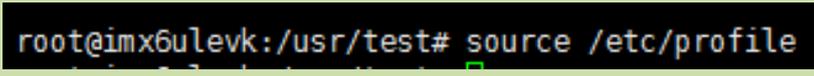
Remark: As the picture of Step 1, the device node of 7 inch resistive touch panel is [/dev/input/event1].

Modify [/etc/]profile file base on your touch panel.

Step 2 :

Command	# vi /etc/profile
Description	<p>Modify /etc/profile file, modify the following sentence according to the actual event node</p> <pre>export set TSLIB_TSDEVICE=/dev/input/event1</pre> <pre>export set</pre> <pre>QWS_MOUSE_PROTO='tslib:/dev/input/event1Intellimouse: /dev/input/mouse1'</pre>
Test Phenomenon	After executing the command, we can see the contents of profile from the printed information. Please refer to the following picture.
Reference	 <pre>export PATH=\$PATH:/usr/local/tslib/bin export set TSLIB_DIR=/usr/local/tslib export set TSLIB_TSDEVICE=/dev/input/event1 export set TSLIB_FBDEVICE=/dev/fb0 export set TSLIB_CONFFILE=\$TSLIB_DIR/etc/ts.conf export set TSLIB_PLUGINDIR=\$TSLIB_DIR/lib/ts export set TSLIB_CALIBFILE=/etc/poointercal export QT_ROOT=/usr/local/qt4.8.5-arm/ export QT_PLUGIN_PATH=\$QT_ROOT/plugins export set QWS_MOUSE_PROTO='tslib:/dev/input/event1 Intellimouse:/dev/input/mous export set QT_QWS FONDIR=\$QT_ROOT/lib/fonts export set LD_LIBRARY_PATH=\$TSLIB_DIR/lib:\$LD_LIBRARY_PATH export set LD_LIBRARY_PATH=\$QT_ROOT/lib:\$LD_LIBRARY_PATH</pre>

Step 3 :

Command	# source /etc/profile
Description	Save environment variables
Test Phenomenon	None
Reference	 <pre>root@imx6ulevk:/usr/test# source /etc/profile</pre>

Step 4 :

Command	# ts_calibrate
Description	Perform the touch screen calibration program.

Test Phenomenon	After executing the command, we can see the hinted information from touch panel.
Reference	None

Remark: After executing this command, finish the calibration according to the display

Step 5 :

Command	# ts_test
Description	Test the precision of touch, we can click on the drag button or line drawing button to test. The Mouse or line will move along the touch point trajectory.
Test Phenomenon	After executing the command, we can see the hinted information from touch panel.
Reference	None

2.16.3 FAQ

- There is no any image and information when testing.

Solution:

- 1、 Check the kernel configuration [make menuconfig], whether it is same as the description of IAC-IMX6UL-Kit BSP development instruction.
- 2、 Please check the touch panel is normal or not.
- 3、 。 Please check the connection between touch panel and development board.

2.17 QT Test

2.17.1 Overview

IAC-IMX6UL-KIT mainboard configuration file system owns 4.8.5 QT library and our company's demo program, this program will bring you into QT world. It loads acquiescently LVDS When using LVDS to start, this QT test program aims at 1024x600 resolution ratio display.

Execute [QiYang_Imx6_qt_test] program, it shows QT interface on LCD screen. It could do interaction with QT by moving cursor through touch panel or mouse.

```
root@imx6ulevk:/usr/test# ls
8723ku_ko  ad_test.sh  buzzer_test  gpio_test  rs485_test  serial_test  watchdog_feed_test
QiYang_Imx6S_Qt_test  backlight_test  can_test  keybutton  rtc_test  shinian.mp3  watchdog_notfeed_test
```

2.17.2 Test steps

Step 1 :

Command	# vi /etc/profile
Description	Modify [/etc/profile]file, please refer to the Step 2 of 2.16.3
Test Phenomenon	Please refer to the Step 2 of 2.16.2
Reference	None

Step 2 :

Command	# ./QiYang_IMX6UL_Qt_test -qws
Description	Execute [QiYang_IMX6UL_Qt_test] program
Test Phenomenon	After executing the command, we can see the following picture from touch panel.
Reference	<p>The screenshot shows a Qt application window titled '杭州启扬智能科技有限公司'. The interface includes a navigation menu on the left with items like '公司简介', '产品介绍', '工控板', '开发板', '核心板', '液晶屏', '联系我们', '售后服务', and '开发板测试'. The main content area displays the company logo, name, and a detailed description of their services in Chinese, including their focus on embedded systems, industrial control, and various IoT applications.</p>

Remark: If use touch, please use [ts_calibrate] before running QT application. Calibrate the touch panel, then execute the [Imx6_qt_test] program.

2.17.3 FAQ

Please refer to chapter 2.15.3 of this manual.