Version 2021 05 04 v0.2 by slash.linux.c@gmail.com

[1] Ubuntu software package manager tool

User Help
$ apt-get --help or $ man apt-get

Update apt database
$ sudo apt-get update

Search package
apt-cach search package-name
$ sudo apt-cache search git-cola

Install package
apt-get install package-name
$ sudo apt-get install git-cola

Remove package
apt-get remove package-name
$ sudo apt-get remove git-cola

Version 2021 05 04 v0.2 by slash.linux.c@gmail.com

[2] Tracking code command

Find Function Name in U-boot source code **grep -r -n “function name” “dir_name”**
$ cd ~/nanopim4-slash/bootload/u-boot-rockchip
$ grep -r “gpio_rockchip_ops” drivers/gpio/

Fine special file include “String” **find -name “*.c” | xarge grep -n “String”**
$ cd ~/nanopim4-slash/bootload/u-boot-rockchip
$ find ./ -name "*.c" | xargs grep -r -n gpio_rockchip_ops

[3] Patch File Exercise

1. Create patch file
$ cd ~/nanopim4-slash/sample/patch_exerise/exercise_1/
$ diff **-Naur** hello_1.c hello_2.c > hello.patch

2. Add patch file
$ cd ~/nanopim4-slash/sample/patch_exerise/exercise_2/
$ cp ../exercise_1/hello.patch ./
$ patch -p0 < hello.patch

3. Remove patch file
$ cd ~/nanopim4-slash/sample/patch_exerise/exercise_2/
$ patch -R -p0 < hello.patch

Version 2021 05 04 v0.2 by slash.linux.c@gmail.com

[4] Git

1. Create an empty Git repository

```
$ cd ~/nanopim4-slash/sample/git_exerise/
$ git init
```

Check the .git directory

```
$ ls -la
$ tree .git
```

2. Show the working tree status

```
$ git status
```

3. Add file contents to the index

```
$ git add .  [add all file]  OR
$ git add ./Makefile
$ git add ./hello.c
```

4. Record changes to the repository

```
$ git commit
```

Write your log

5. Check your git log

```
$ git log
```

5. Modify the source code

```
$ geany [gedit, vim] ./hello.c
```

6. Show changes between commits, commit and working tree

```
$ git diff
```

7. Add file contents to the index

```
$ git add ./hello.c
```

8. Check status

```
$ git status
```

9. Do git commit and use short comment

```
$ git commit -m “add slash string”
```

10. Show the new git patch

```
$ git show
```

11. Create a new branch

```
$ git branch branch-test
$ git branch
```

12. Check to New branch

```
$ git checkout branch-test
$ git branch
```

13. Modify hello.c in New Branch

```
$ Modify hello.c
```

13. Commit a new change

Ref Step 7 - 9

14. Create a formation patch

```
$ git format -1
```

15. Checkout To master branch then use the patch file

```
$ git checkout master
$ git am ././0001-PATCH-NAME.patch
```

Version 2021 05 04 v0.2 by slash.linux.c@gmail.com

[5] Cross Compile Toolchain

Enter toolchain directory

```
$ cd ~/nanopim4-slash/toolchain/
$ tar -xvjf  gcc-linaro-6.3.1-2017.05-x86_64_aarch64-linux-gnu.tar.bz2
```

Modify set_toolchain.sh

```
$ cd ~/nanopim4-slash/toolchain/
```

set_toolchain.sh contents

```
export PATH=$PATH:/home/cadtc/nanopim4-slash/toolchain/gcc-linaro-6.3.1-2017.05-x86_64_aarch64-linux-gnu/bin
export ARCH=arm64
export CROSS_COMPILE=aarch64-linux-gnu-
export KERNELDIR=/home/cadtc/project/rk3399/kernel/rockchip-rk3399-nanopi-m4
```

Initial compile environment

```
$ cd ~/nanopim4-slash/toolchain/
$ source ./set_toolchain.sh
```

check environment

```
$ echo $ CROSS_COMPILE
$ aarch64-linux-gnu-gcc -v
```

[6] Media Tool For Gstreamer

[For Video]

Play a Video

```
<Target Board>

$ gst-launch-1.0 playbin uri=file:///oem/SampleVideo_1280x720_5mb.mp4
```

Play a Video Test Patten

```
<Target Board>

$ gst-launch-1.0 videotestsrc ! video/x-raw, width=1280, height=720 ! kmssink
```

Play a H.264 Video

```
<Target Board>

$ gst-launch-1.0 filesrc location=/oem/200frames_count.h264 ! \
decodebin name=dec ! \
videoconvert ! \
kmssink
```

[For Audio]

Play a Audio Test

```
<Target Board>

$ gst-launch-1.0 audiotestsrc ! audioconvert ! alsasink device-name=realtekrt5651co
```

Play a MP3 File

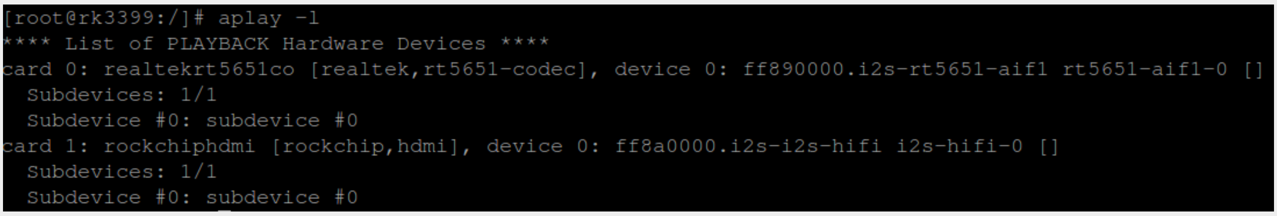
```
<Target Board>

$ gst-launch-1.0 filesrc location="oem/piano2-CoolEdit.mp3" ! \
mpegaudioparse ! \
avdec_mp3 ! \
audioconvert ! \
alsasink device=hw:0
```

[7] Media Tool For ALSA-Tool

<command-line sound player and record for ALSA soundcard driver>

list all soundcards and digital audio devices <Target Board>

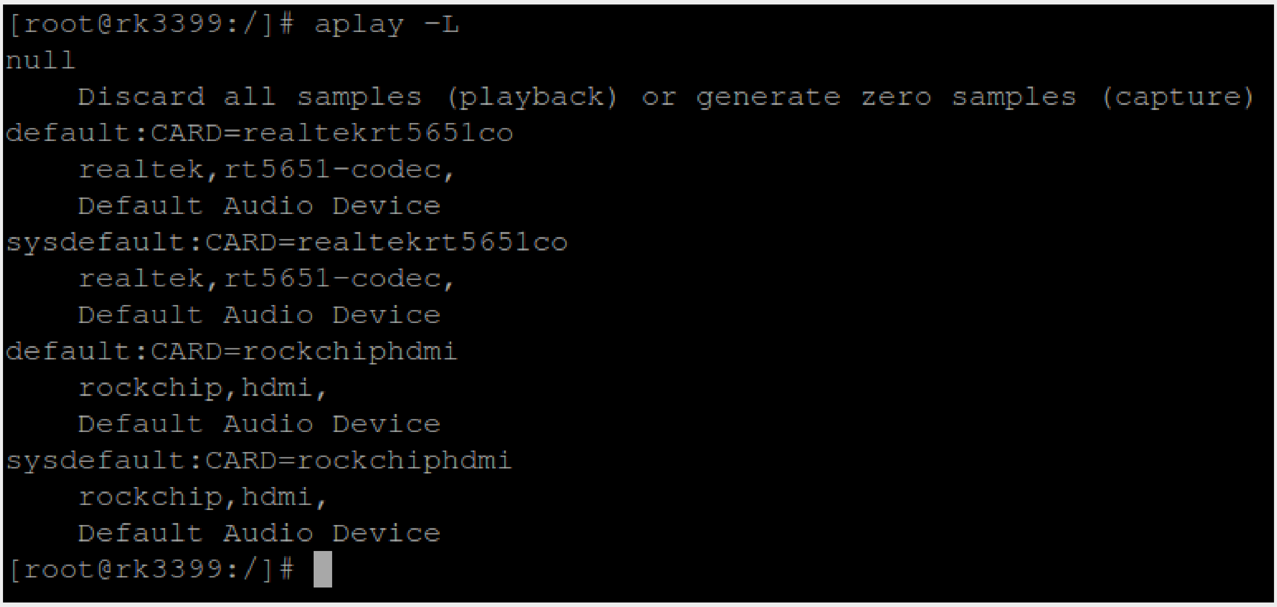
$ aplay -l



```
[root@rk3399:/]# aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: realtekrt5651co [realtek,rt5651-codec], device 0: ff890000.i2s-rt5651-aif1 rt5651-aif1-0 []
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 1: rockchiphdmi [rockchip,hdmi], device 0: ff8a0000.i2s-i2s-hifi i2s-hifi-0 []
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

list device names

$ aplay -L



```
[root@rk3399:/]# aplay -L
null
    Discard all samples (playback) or generate zero samples (capture)
default:CARD=realtekrt5651co
    realtek,rt5651-codec,
    Default Audio Device
sysdefault:CARD=realtekrt5651co
    realtek,rt5651-codec,
    Default Audio Device
default:CARD=rockchiphdmi
    rockchip,hdmi,
    Default Audio Device
sysdefault:CARD=rockchiphdmi
    rockchip,hdmi,
    Default Audio Device
[root@rk3399:/]# 
```

Play a WAV file for ALSA soundcard

aplay -D${PCM_DEVICE} $WAV_FILE

$ aplay -Dsysdefault:CARD=realtekrt5651co /home/slash/Ensoniq-ZR

$ aplay -Ddefault:CARD=rockchiphdmi /home/slash/Ensoniq-ZR-76-01

Version 2021 05 04 v0.2 by slash.linux.c@gmail.com

Record a WAV file from ALSA soundcard

```
arecord -D${PCM_DEVICE}  $WAV_FILE
```

```
$ arecord -Dhw:0,0 -r 44100 -t wav -f CD -d 5 /tmp/test.wav
```

Graphical mixer program

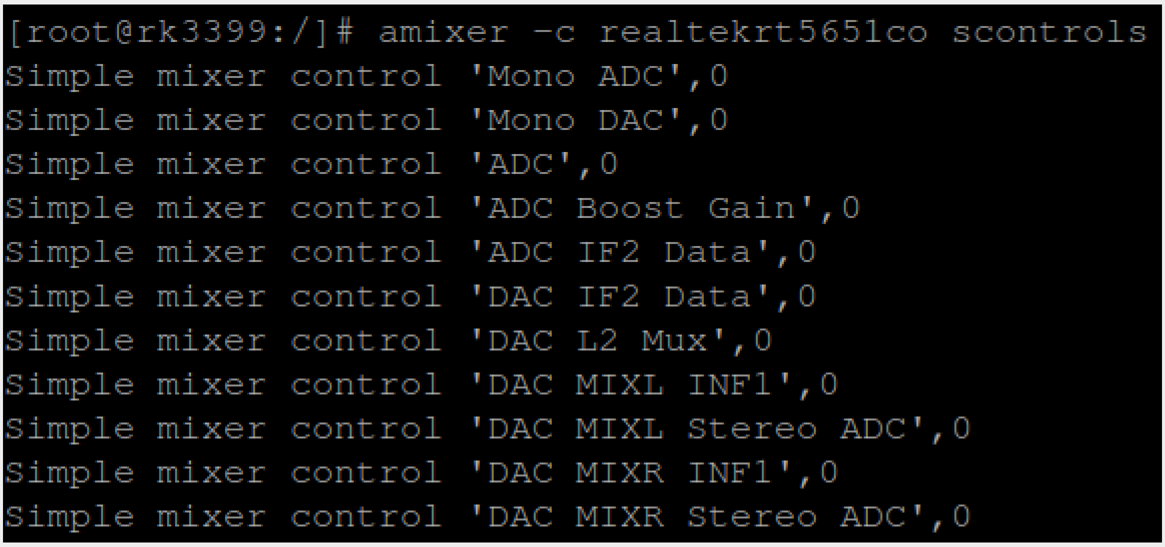
```
$ alsamixer
```

command-line mixer for ALSA soundcard driver

```
$ amixer
```

show all mixer simple controls

```
$ amixer -c realtekrt5651co scontrols
```

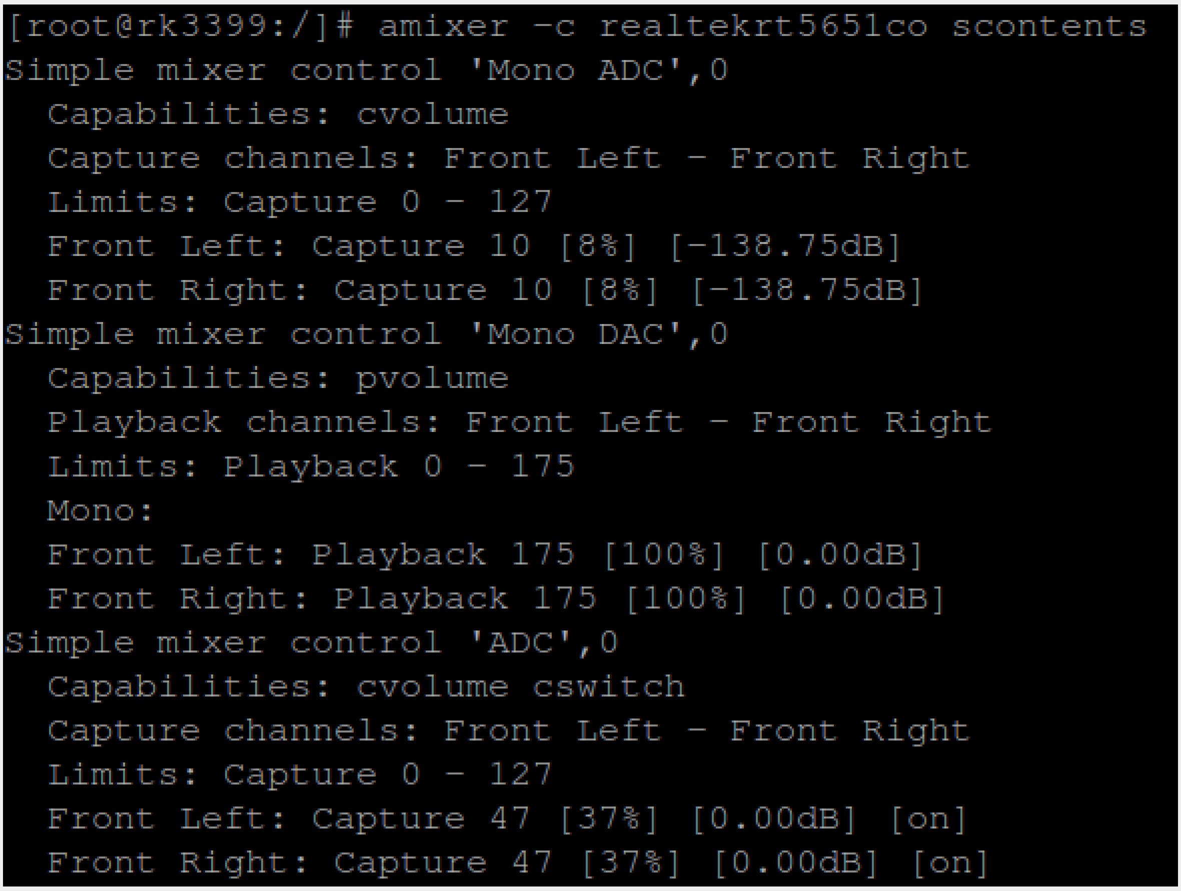


```
[root@rk3399:/]# amixer -c realtekrt5651co scontrols
Simple mixer control 'Mono ADC',0
Simple mixer control 'Mono DAC',0
Simple mixer control 'ADC',0
Simple mixer control 'ADC Boost Gain',0
Simple mixer control 'ADC IF2 Data',0
Simple mixer control 'DAC IF2 Data',0
Simple mixer control 'DAC L2 Mux',0
Simple mixer control 'DAC MIXL INF1',0
Simple mixer control 'DAC MIXL Stereo ADC',0
Simple mixer control 'DAC MIXR INF1',0
Simple mixer control 'DAC MIXR Stereo ADC',0
```

Version 2021 05 04 v0.2 by slash.linux.c@gmail.com

show contents of all mixer simple controls

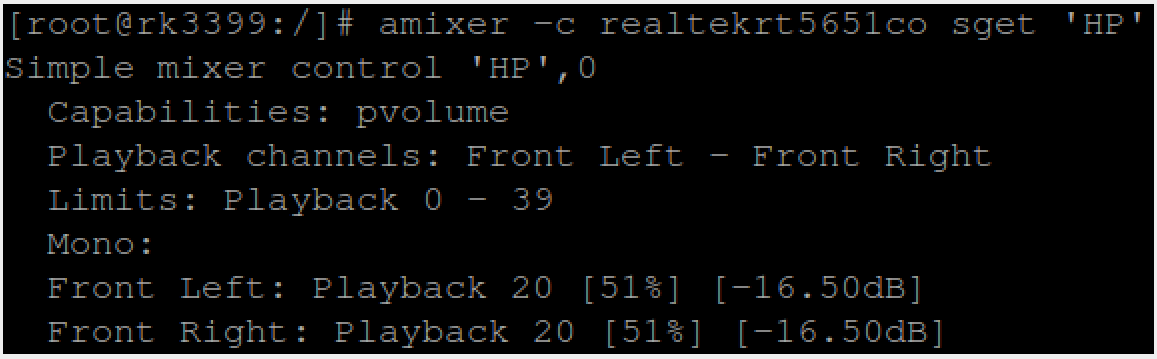
$ amixer -c realtekrt5651co scontrols



```
[root@rk3399:/]# amixer -c realtekrt5651co scontents
Simple mixer control 'Mono ADC',0
  Capabilities: cvolume
  Capture channels: Front Left - Front Right
  Limits: Capture 0 - 127
  Front Left: Capture 10 [8%] [-138.75dB]
  Front Right: Capture 10 [8%] [-138.75dB]
Simple mixer control 'Mono DAC',0
  Capabilities: pvolume
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 175
  Mono:
  Front Left: Playback 175 [100%] [0.00dB]
  Front Right: Playback 175 [100%] [0.00dB]
Simple mixer control 'ADC',0
  Capabilities: cvolume cswitch
  Capture channels: Front Left - Front Right
  Limits: Capture 0 - 127
  Front Left: Capture 47 [37%] [0.00dB] [on]
  Front Right: Capture 47 [37%] [0.00dB] [on]
```

Set contents for one mixer simple control

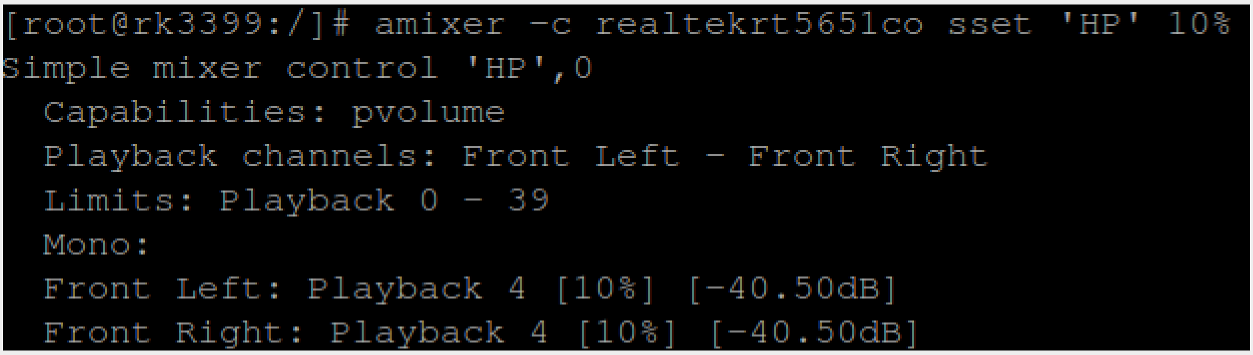
$ amixer -c realtekrt5651co sget 'HP'



```
[root@rk3399:/]# amixer -c realtekrt5651co sget 'HP'
Simple mixer control 'HP',0
  Capabilities: pvolume
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 39
  Mono:
  Front Left: Playback 20 [51%] [-16.50dB]
  Front Right: Playback 20 [51%] [-16.50dB]
```

Set contents for one mixer simple control

```
$ amixer -c realtekrt5651co sset 'HP'  10%
```

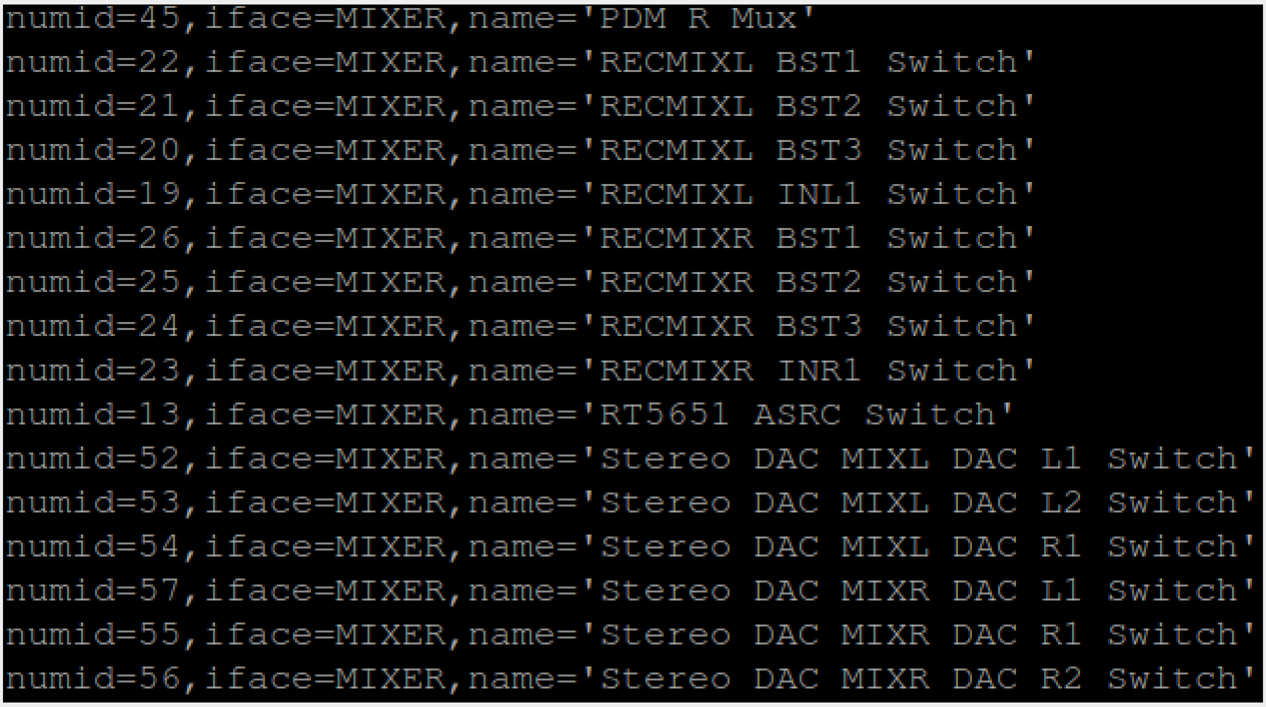


```
[root@rk3399:/]# amixer -c realtekrt5651co sset 'HP' 10%
Simple mixer control 'HP',0
  Capabilities: pvolume
  Playback channels: Front Left - Front Right
  Limits: Playback 0 - 39
  Mono:
  Front Left: Playback 4 [10%] [-40.50dB]
  Front Right: Playback 4 [10%] [-40.50dB]
```

Version 2021 05 04 v0.2 by slash.linux.c@gmail.com

Show all controls for given card

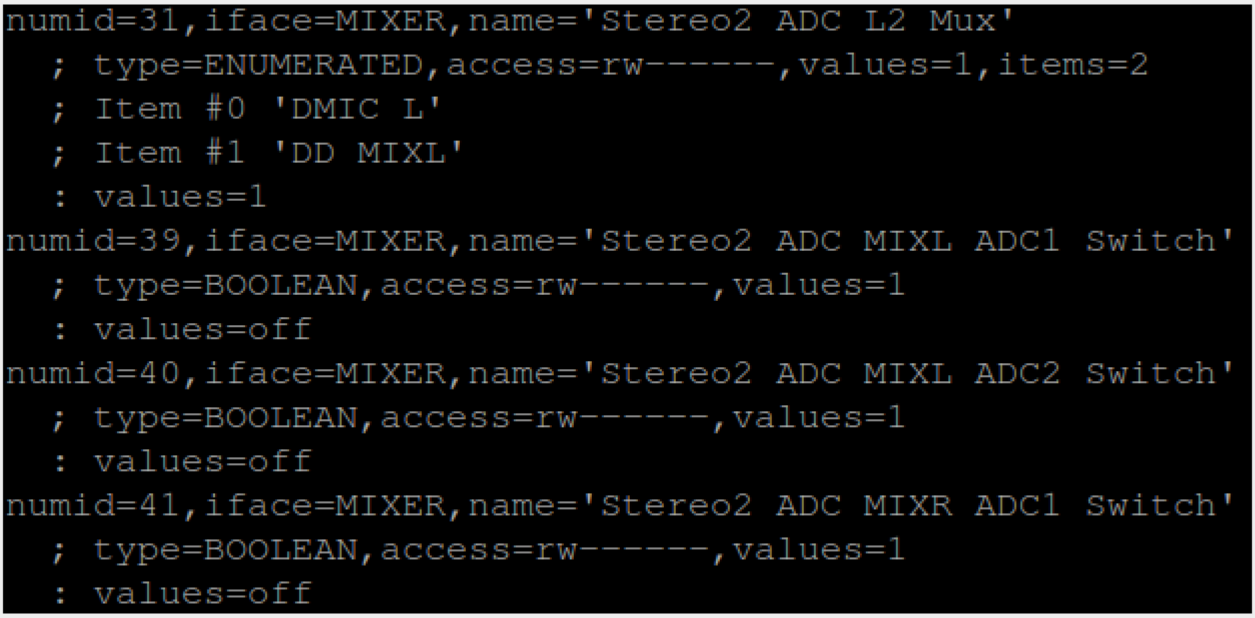
$ amixer -c realtekrt5651co controls



```
numid=45,iface=MIXER,name='PDM R Mux'
numid=22,iface=MIXER,name='RECMIXL BST1 Switch'
numid=21,iface=MIXER,name='RECMIXL BST2 Switch'
numid=20,iface=MIXER,name='RECMIXL BST3 Switch'
numid=19,iface=MIXER,name='RECMIXL INL1 Switch'
numid=26,iface=MIXER,name='RECMIXR BST1 Switch'
numid=25,iface=MIXER,name='RECMIXR BST2 Switch'
numid=24,iface=MIXER,name='RECMIXR BST3 Switch'
numid=23,iface=MIXER,name='RECMIXR INR1 Switch'
numid=13,iface=MIXER,name='RT5651 ASRC Switch'
numid=52,iface=MIXER,name='Stereo DAC MIXL DAC L1 Switch'
numid=53,iface=MIXER,name='Stereo DAC MIXL DAC L2 Switch'
numid=54,iface=MIXER,name='Stereo DAC MIXL DAC R1 Switch'
numid=57,iface=MIXER,name='Stereo DAC MIXR DAC L1 Switch'
numid=55,iface=MIXER,name='Stereo DAC MIXR DAC R1 Switch'
numid=56,iface=MIXER,name='Stereo DAC MIXR DAC R2 Switch'
```

Show contents of all controls for given card

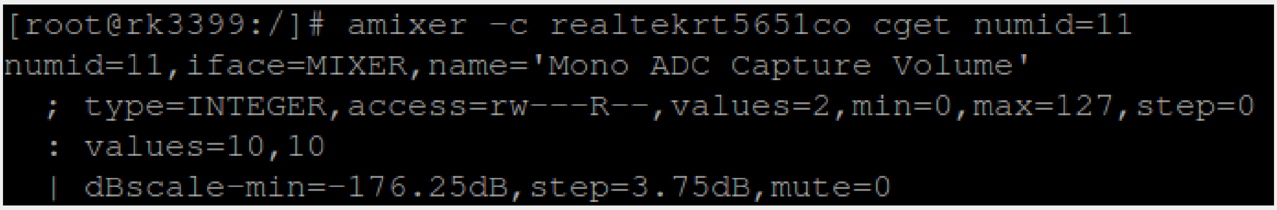
$ amixer -c realtekrt5651co contents



```
numid=31,iface=MIXER,name='Stereo2 ADC L2 Mux'
  ; type=ENUMERATED,access=rw------,values=1,items=2
  ; Item #0 'DMIC L'
  ; Item #1 'DD MIXL'
  : values=1
numid=39,iface=MIXER,name='Stereo2 ADC MIXL ADC1 Switch'
  ; type=BOOLEAN,access=rw------,values=1
  : values=off
numid=40,iface=MIXER,name='Stereo2 ADC MIXL ADC2 Switch'
  ; type=BOOLEAN,access=rw------,values=1
  : values=off
numid=41,iface=MIXER,name='Stereo2 ADC MIXR ADC1 Switch'
  ; type=BOOLEAN,access=rw------,values=1
  : values=off
```

cget cID → get control contents for one contro

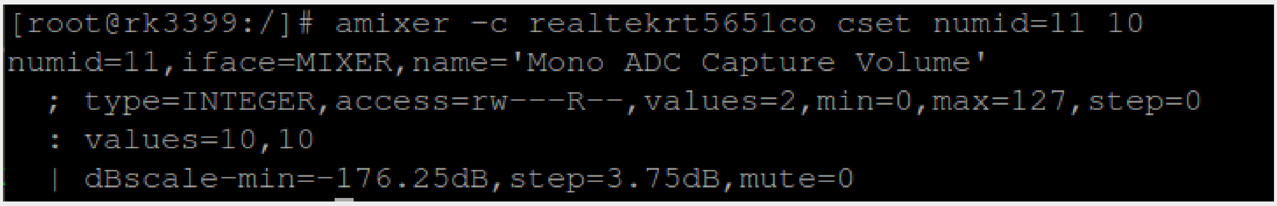
$ amixer -c realtekrt5651co cget numid=11



```
[root@rk3399:/]# amixer -c realtekrt5651co cget numid=11
numid=11,iface=MIXER,name='Mono ADC Capture Volume'
  ; type=INTEGER,access=rw---R--,values=2,min=0,max=127,step=0
  : values=10,10
  | dBscale-min=-176.25dB,step=3.75dB,mute=0
```

cset cID → set control contents for one control

$ amixer



```
[root@rk3399:/]# amixer -c realtekrt5651co cset numid=11 10
numid=11,iface=MIXER,name='Mono ADC Capture Volume'
  ; type=INTEGER,access=rw---R--,values=2,min=0,max=127,step=0
  : values=10,10
  | dBscale-min=-176.25dB,step=3.75dB,mute=0
```

[8] Network

Check network IP

```
ifconfig $ETHERNET
```

```
$ ifconfig eth0
```

Change MAC address

```
$ ifconfig $ETHERNET down
$ ifconfig $ETHERNET hw ether $MAC_ADDR
$ ifconfig $ETHERNET up
```

Get Dynamic Network IP

```
$ udhcpc -i $ETHERNET
```

Send ICMP ECHO_REQUEST to network hosts

```
ping  $Host_IP_Address

$ ping 8.8.8.8
```

Perform network throughput tests

Step 1.
[Host Side]

```
$ iperf3 -s
```

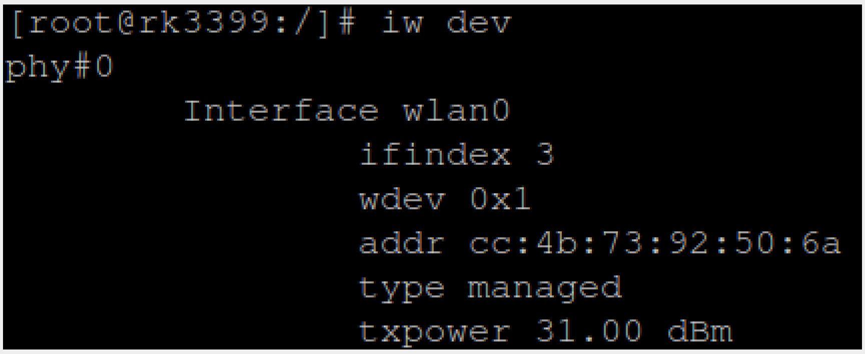
Step 2.
[Target Board Side]

```
$ iperf3 -c $Host_IP_Addr
```

[9] Wireless Network

Show / manipulate wireless devices and their configuration

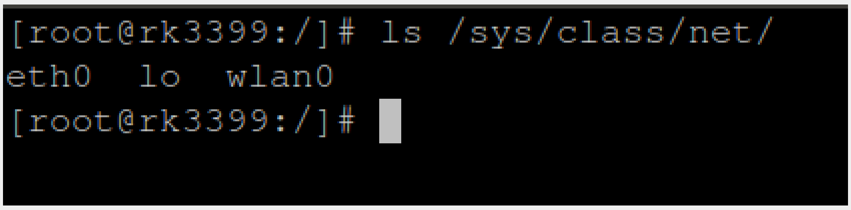
$ iw dev



```
[root@rk3399:/]# iw dev
phy#0
        Interface wlan0
                ifindex 3
                wdev 0x1
                addr cc:4b:73:92:50:6a
                type managed
                txpower 31.00 dBm
```

Check Network Device

$ ls /sys/class/net

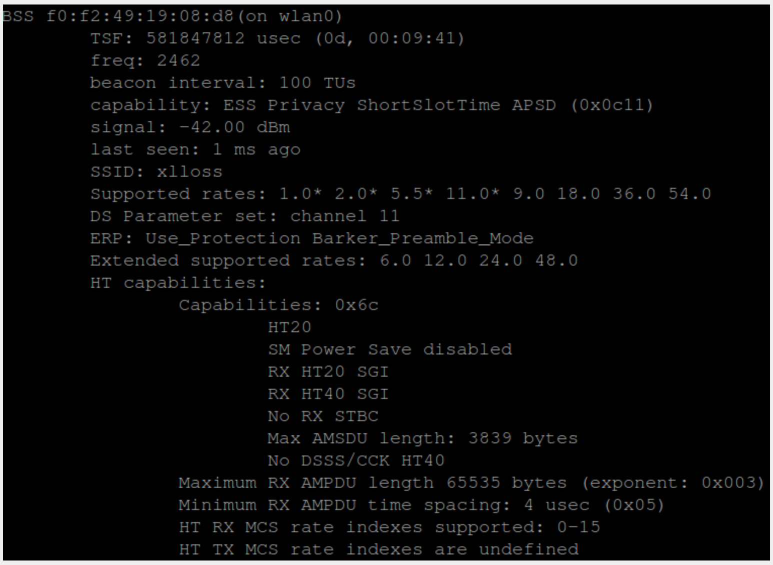


```
[root@rk3399:/]# ls /sys/class/net/
eth0   lo   wlan0
[root@rk3399:/]#
```

Version 2021 05 04 v0.2 by slash.linux.c@gmail.com

Scan AP Router SSID

$ iw wlan0 scan

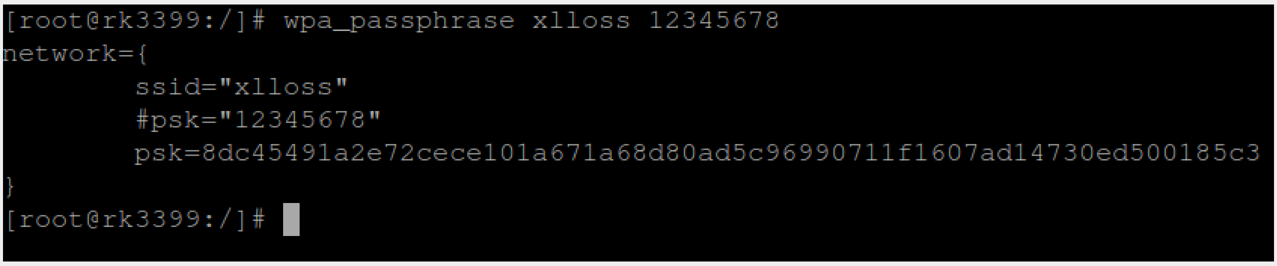


```
BSS f0:f2:49:19:08:d8(on wlan0)
        TSF: 581847812 usec (0d, 00:09:41)
        freq: 2462
        beacon interval: 100 TUs
        capability: ESS Privacy ShortSlotTime APSD (0x0c11)
        signal: -42.00 dBm
        last seen: 1 ms ago
        SSID: xlloss
        Supported rates: 1.0* 2.0* 5.5* 11.0* 9.0 18.0 36.0 54.0
        DS Parameter set: channel 11
        ERP: Use_Protection Barker_Preamble_Mode
        Extended supported rates: 6.0 12.0 24.0 48.0
        HT capabilities:
                Capabilities: 0x6c
                        HT20
                        SM Power Save disabled
                        RX HT20 SGI
                        RX HT40 SGI
                        No RX STBC
                        Max AMSDU length: 3839 bytes
                        No DSSS/CCK HT40
                Maximum RX AMPDU length 65535 bytes (exponent: 0x003)
                Minimum RX AMPDU time spacing: 4 usec (0x05)
                HT RX MCS rate indexes supported: 0-15
                HT TX MCS rate indexes are undefined
```

Generate a WPA PSK from an ASCII passphrase for a SSID

wpa_passphrase ${SSID} ${PASSWORD}

$ wpa_passphrase xlloss 12345



```
[root@rk3399:/]# wpa_passphrase xlloss 12345678
network={
        ssid="xlloss"
        #psk="12345678"
        psk=8dc45491a2e72cece101a671a68d80ad5c96990711f1607ad14730ed500185c3
}
[root@rk3399:/]#
```

$ wpa_passphrase xlloss 12345 > /etc/wpa_supplicant.conf

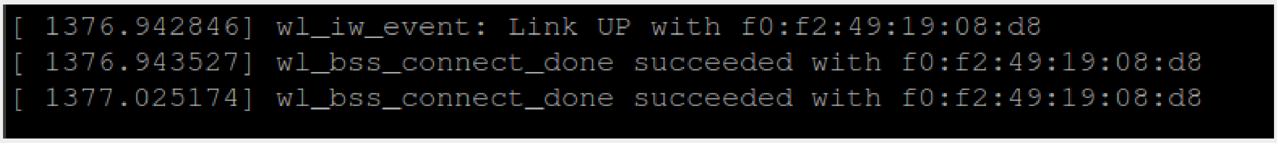
$ vim /etc/wpa_supplicant.conf

Version 2021 05 04 v0.2 by slash.linux.c@gmail.com

Connect the Wireless AP Route

wpa_supplicant -B -D wext -i ${WLAN_DEV} -c /etc/wpa_supplicant.conf

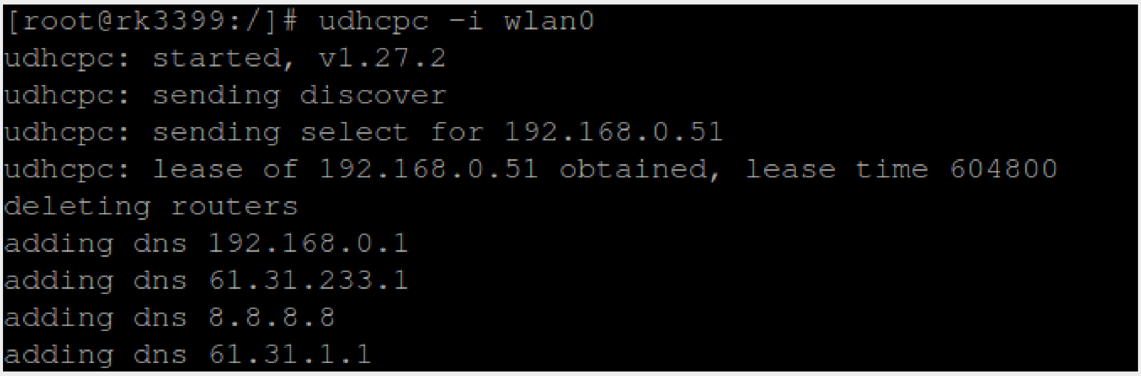
$ wpa_supplicant -B -D wext -i wlan0 -c /etc/wpa_supplicant.conf



```
[ 1376.942846] wl_iw_event: Link UP with f0:f2:49:19:08:d8
[ 1376.943527] wl_bss_connect_done succeeded with f0:f2:49:19:08:d8
[ 1377.025174] wl_bss_connect_done succeeded with f0:f2:49:19:08:d8
```

Get IP From DHCP Server

udhcpc -i ${WLAN_DEV}
$ udhcpc -i wlan0



```
[root@rk3399:/]# udhcpc -i wlan0
udhcpc: started, v1.27.2
udhcpc: sending discover
udhcpc: sending select for 192.168.0.51
udhcpc: lease of 192.168.0.51 obtained, lease time 604800
deleting routers
adding dns 192.168.0.1
adding dns 61.31.233.1
adding dns 8.8.8.8
adding dns 61.31.1.1
```

Check WLAN Device – IP Address

Check IP

$ ifconfig wlan0

Check WLAN Device – Ping

Ping a Network IP

$ ping -I wlan0 8.8.8.8

[10] SSH

Transmit a File to Remote

[Host]

```
scp  {FILE} ${Remote_Account}@${Remote_Netowrk_IP}:${Remote_Folder}
```

```
$ scp ./test_file root@192.168.1.100:/tmp/test_dir
```

Mount a Remote Folder to Target board - Step 1

[Target Board]

```
$ mkdir /tmp/test_dir

$ mkdir /tmp/test_dir/testme
```

Mount a Remote Folder to Target board – Step 2

[Host]

```
sshfs  ${Remote_Account}@${Remote_Netowrk_IP}:${Remote_Folder}  ${Local_Folder}
```

```
$ sshfs root@192.168.0.27:/tmp/slash ./test/
```

[11] I2C Tool

Get a I2C Client Device Register

[Target]

```
i2cget [-f] [-y] I2CBUS CHIP-ADDRESS [DATA-ADDRESS [MODE]]
```

```
$ i2cget -f -y 2 0x50 0x13
```

Set a Data to I2C Client Device Register

[Target]

```
i2cset [-f] [-y] [-m MASK] [-r] I2CBUS CHIP-ADDRESS DATA-ADDRESS [VALUE]
```

```
$ i2cset -f -y 2 0x50 0x13 0x01
```

Dump Data From I2C Client Device Register

[Target]

```
i2cdump [-f] [-y] [-r first-last] I2CBUS ADDRESS
```

```
$ i2cdump -f -y 2 0x50
```