**Android build steps for open-Q 820 board:**

1.curl <https://storage.googleapis.com/git-repo-downloads/repo> > ~/bin/repo

chmod a+x ~/bin/repo

2.repo init -u git://codeaurora.org/platform/manifest.git -b release -m LA.UM.5.5.r1-01300-8x96.0.xml

3.repo sync -cj4 -q --no-tags

4.Apply APQ8096 Snapdragon 820 Dragonboard patches

5.tar -xzvf ../proprietary.tar.gz -C vendor/qcom/

6.source build/envsetup.sh

7.lunch msm8996-userdebug

8.make -j4

After build the images are available under below path,

akothapally@cicdim:~/android/out/target/product/msm8996$ ls \*.img

boot.img cache.img persist.img ramdisk.img ramdisk-recovery.img recovery.img system.img userdata.img

Then flash these images on board using fast boot.

Smoke test on Board with new build:

1.play a .wav file using music app and record a .wav file using Sound recorder app.

2.Using tinyplay play a .wav file,the following result is observed.

msm8996:/ # tinyplay /sdcard/Music/file\_example\_WAV\_1MG.wav

Playing sample: 2 ch, 8000 hz, 16 bit

Error playing sample

The above log is from /android/external/tinyalsa/tinyplay.c file.

Tinyplay path using the below tinymix commands,

for tinyplay ,tinymix settings:

tinymix 'SLIMBUS\_6\_RX Audio Mixer MultiMedia1' '1'

for headphones

tinymix 'SLIM RX2 MUX' 'AIF4\_PB'

tinymix 'SLIM RX3 MUX' 'AIF4\_PB'

tinymix 'SLIM\_6\_RX Channels' 'Two'

tinymix 'RX INT1\_2 MUX' 'RX2'

tinymix 'RX INT2\_2 MUX' 'RX3'

tinymix 'RX INT1 DEM MUX' 'CLSH\_DSM\_OUT'

tinymix 'RX INT2 DEM MUX' 'CLSH\_DSM\_OUT'

for record a song tinycap settings:

tinymix 'MultiMedia1 Mixer SLIM\_0\_TX' '1'

tinymix 'MultiMedia1 Mixer TERT\_MI2S\_TX' '1'

tinymix 'AIF1\_CAP Mixer SLIM TX7' '1'

tinymix 'SLIM\_0\_TX Channels' 'One'

tinymix 'SLIM TX7 MUX' 'DEC7'

tinymix 'ADC MUX7' 'DMIC'

tinymix 'DMIC MUX7' 'DMIC0'

tinymix 'IIR0 INP0 MUX' 'DEC7'

For **kernel changes** i.e any changes in files and build the module using the below command:

**make -j4 bootimage**

After build, the boot.img is changed and we need to flash it on board using adb commands.

For any files **changes in user space file**s like audio flinger, audio policy and tinyalsa filles then build the module using below command: **mm**

The command is given in the folder contain android.mk file.

After build, the result files is .so or. a file.We need to add this file in the specific folder.

Example:

akothapally@cicdim:~/android$ source build/envsetup.sh

including device/generic/mini-emulator-arm64/vendorsetup.sh

including device/generic/mini-emulator-armv7-a-neon/vendorsetup.sh

including device/generic/mini-emulator-x86\_64/vendorsetup.sh

including device/generic/mini-emulator-x86/vendorsetup.sh

including device/qcom/common/vendorsetup.sh

including vendor/qcom/proprietary/common/vendorsetup.sh

including sdk/bash\_completion/adb.bash

akothapally@cicdim:~/android$ lunch msm8996-userdebug

PLATFORM\_VERSION\_CODENAME=REL

PLATFORM\_VERSION=7.0

TARGET\_PRODUCT=msm8996

TARGET\_BUILD\_VARIANT=userdebug

TARGET\_BUILD\_TYPE=release

TARGET\_BUILD\_APPS=

TARGET\_ARCH=arm64

TARGET\_ARCH\_VARIANT=armv8-a

TARGET\_CPU\_VARIANT=kryo

TARGET\_2ND\_ARCH=arm

TARGET\_2ND\_ARCH\_VARIANT=armv7-a-neon

TARGET\_2ND\_CPU\_VARIANT=cortex-a53

HOST\_ARCH=x86\_64

HOST\_2ND\_ARCH=x86

HOST\_OS=linux

HOST\_OS\_EXTRA=Linux-3.19.0-66-generic-x86\_64-with-Ubuntu-14.04-trusty

HOST\_CROSS\_OS=windows

HOST\_CROSS\_ARCH=x86

HOST\_CROSS\_2ND\_ARCH=x86\_64

HOST\_BUILD\_TYPE=release

BUILD\_ID=OpenQ820\_N\_v3.3

OUT\_DIR=out

akothapally@cicdim:~/android$ cd external/tinyalsa/

akothapally@cicdim:~/android/external/tinyalsa$ mm

PLATFORM\_VERSION\_CODENAME=REL

PLATFORM\_VERSION=7.0

TARGET\_PRODUCT=msm8996

TARGET\_BUILD\_VARIANT=userdebug

TARGET\_BUILD\_TYPE=release

TARGET\_BUILD\_APPS=

TARGET\_ARCH=arm64

TARGET\_ARCH\_VARIANT=armv8-a

TARGET\_CPU\_VARIANT=kryo

TARGET\_2ND\_ARCH=arm

TARGET\_2ND\_ARCH\_VARIANT=armv7-a-neon

TARGET\_2ND\_CPU\_VARIANT=cortex-a53

HOST\_ARCH=x86\_64

HOST\_2ND\_ARCH=x86

HOST\_OS=linux

HOST\_OS\_EXTRA=Linux-3.19.0-66-generic-x86\_64-with-Ubuntu-14.04-trusty

HOST\_CROSS\_OS=windows

HOST\_CROSS\_ARCH=x86

HOST\_CROSS\_2ND\_ARCH=x86\_64

HOST\_BUILD\_TYPE=release

BUILD\_ID=OpenQ820\_N\_v3.3

OUT\_DIR=out

make: Entering directory `/home/people/akothapally/android'

Running kati to generate build-msm8996-mmm-external\_tinyalsa\_Android.mk.ninja...

out/build-msm8996-mmm-external\_tinyalsa\_Android.mk.ninja is missing, regenerating...

PLATFORM\_VERSION\_CODENAME=REL

PLATFORM\_VERSION=7.0

TARGET\_PRODUCT=msm8996

TARGET\_BUILD\_VARIANT=userdebug

TARGET\_BUILD\_TYPE=release

TARGET\_BUILD\_APPS=

TARGET\_ARCH=arm64

TARGET\_ARCH\_VARIANT=armv8-a

TARGET\_CPU\_VARIANT=kryo

TARGET\_2ND\_ARCH=arm

TARGET\_2ND\_ARCH\_VARIANT=armv7-a-neon

TARGET\_2ND\_CPU\_VARIANT=cortex-a53

HOST\_ARCH=x86\_64

HOST\_2ND\_ARCH=x86

HOST\_OS=linux

HOST\_OS\_EXTRA=Linux-3.19.0-66-generic-x86\_64-with-Ubuntu-14.04-trusty

HOST\_CROSS\_OS=windows

HOST\_CROSS\_ARCH=x86

HOST\_CROSS\_2ND\_ARCH=x86\_64

HOST\_BUILD\_TYPE=release

BUILD\_ID=OpenQ820\_N\_v3.3

OUT\_DIR=out

PRODUCT\_COPY\_FILES device/qcom/common/media/media\_profiles.xml:system/etc/media\_profiles.xml ignored.

PRODUCT\_COPY\_FILES device/qcom/common/media/media\_codecs.xml:system/etc/media\_codecs.xml ignored.

No private recovery resources for TARGET\_DEVICE msm8996

Starting build with ninja

ninja: Entering directory `.'

[100% 36/36] host StaticLib: libtinyalsa\_32 (out/host/linux-x86/obj32/STATIC\_LIBRARIES/libtinyalsa\_intermediates/libtinyalsa.a)

make: Leaving directory `/home/people/akothapally/android'

#### make completed successfully (12 seconds) ####

**Enable dynamic logs for Audio:**

echo -n 'file apq8084.c +p' > /sys/kernel/debug/dynamic\_debug/control

echo -n 'file soc-core.c +p' > /sys/kernel/debug/dynamic\_debug/control

echo -n 'file soc-dapm.c +p' > /sys/kernel/debug/dynamic\_debug/control

echo -n 'file msm-pcm-voice-v2.c +p' > /sys/kernel/debug/dynamic\_debug/control

echo -n 'file msm-dai-q6-v2.c +p' > /sys/kernel/debug/dynamic\_debug/control

echo -n 'file msm-pcm-afe-v2.c +p' > /sys/kernel/debug/dynamic\_debug/control

echo -n 'file msm-pcm-routing-v2.c +p' > /sys/kernel/debug/dynamic\_debug/control

echo -n 'file msm-pcm-q6-v2.c +p' > /sys/kernel/debug/dynamic\_debug/control

echo -n 'file q6adm.c +p' > /sys/kernel/debug/dynamic\_debug/control

echo -n 'file q6afe.c +p' > /sys/kernel/debug/dynamic\_debug/control

echo -n 'file q6asm.c +p' > /sys/kernel/debug/dynamic\_debug/control

**ASoC Architecture qualcomm platforms:**

MSM frontend CPU driver − /kernel/sound/soc/msm/msm-dai-fe.c

MSM Hexagon audio platform driver − /kernel/sound/soc/msm/msm-pcm-q6.c

MSM Hexagon routing platform driver − /kernel/sound/soc/msm/msm-pcm-routing.c

MSM Hexagon backend CPU driver − /kernel/sound/soc/msm/msm-dai-q6.c

Q6 voice platform drivers − kernel/sound/soc/msm/msm-pcm-voice.c and msm-pcm-voip.c

ALSA codec drivers − /kernel/sound/soc/codecs/wcd9310.c, wcd9310-tables.c, etc. (for WCD9310 codec)

I2C/SLIMbus drivers − /kernel/drivers/slimbus/slim-msm-ctrl.c, slimbus.c

ASM driver − kernel/sound/soc/msm/qdsp6/q6asm.c

ADM driver − kernel/sound/soc/msm/qdsp6/q6adm.c

AFE driver − kernel/sound/soc/msm/qdsp6/q6afe.c

Voice driver − kernel/sound/soc/msm/qdsp6/q6voice.c

Platform-specific machine driver − msm8960.c (for MSM8960), MDM9x15 (for MDM9x15), etc.

AFE − Audio Frontend

ASM − Audio Stream Manager

ADM − Audio Device Manager

VSM − Voice Stream Manager

The routing map for the DSP defined as intercon in msm-pcm-routing.c describes the various routing paths or devices to which the different types of streams can be routed. Each record in the table is a collection of three entries. Reading from right to left, the first entry is the audio source, the second is the control (if any), and the third is the audio sink for the corresponding physical hardware route it represents, e.g.:

{"SLIMBUS\_6\_RX Audio Mixer", "MultiMedia1","MM\_DL1"},

{"SLIMBUS\_6\_RX Audio Mixer", "MultiMedia2","MM\_DL2"}

{"SLIMBUS\_0\_RX Audio Mixer", "MultiMedia1", "MM\_DL1"}

{"SLIMBUS\_0\_RX", NULL, "SLIMBUS\_0\_RX Audio Mixer"}

MM\_DL1 (Multimedia Downlink 1) is a playback stream that in the first example above is a source that is routed to a mixer in the DSP called SLIMBUS\_0\_RX Audio Mixer. In the next example, SLIMBUS\_0\_RX Audio Mixer is the source whose output is routed to the SLIMBUS\_0\_RX port connected to the hardware codec.

The audio route map describes the various stream routes supported by the DSP firmware.

**adb commands:**

adb reboot bootloader

fastboot flash boot boot.img

fastboot flash cache cache.img

fastboot flash persist persist.img

fastboot flash userdata userdata.img

fastboot flash system system.img

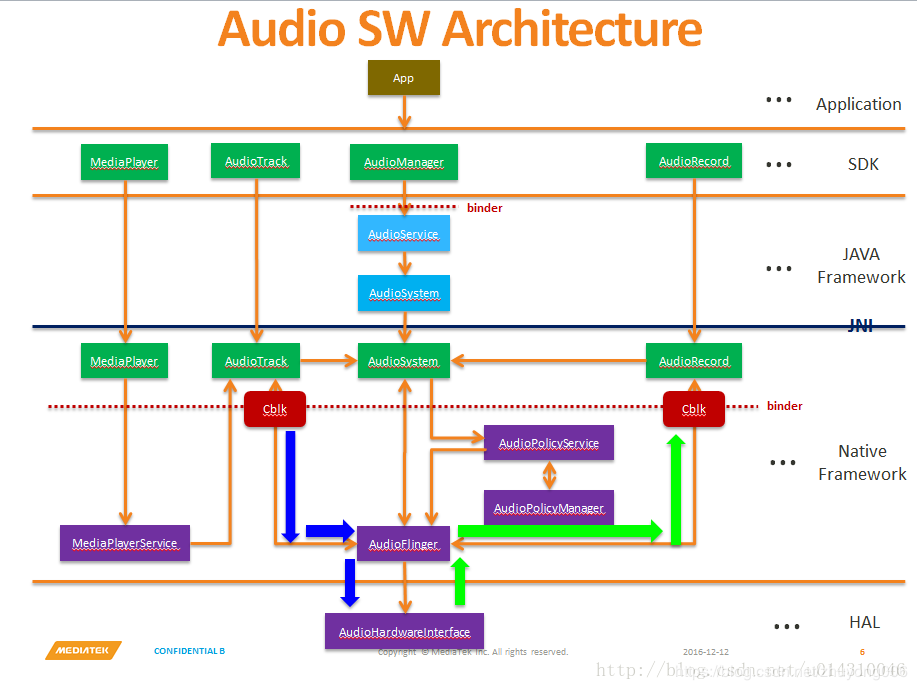
fastboot flash recovery recovery.img

fastboot reboot

adb root

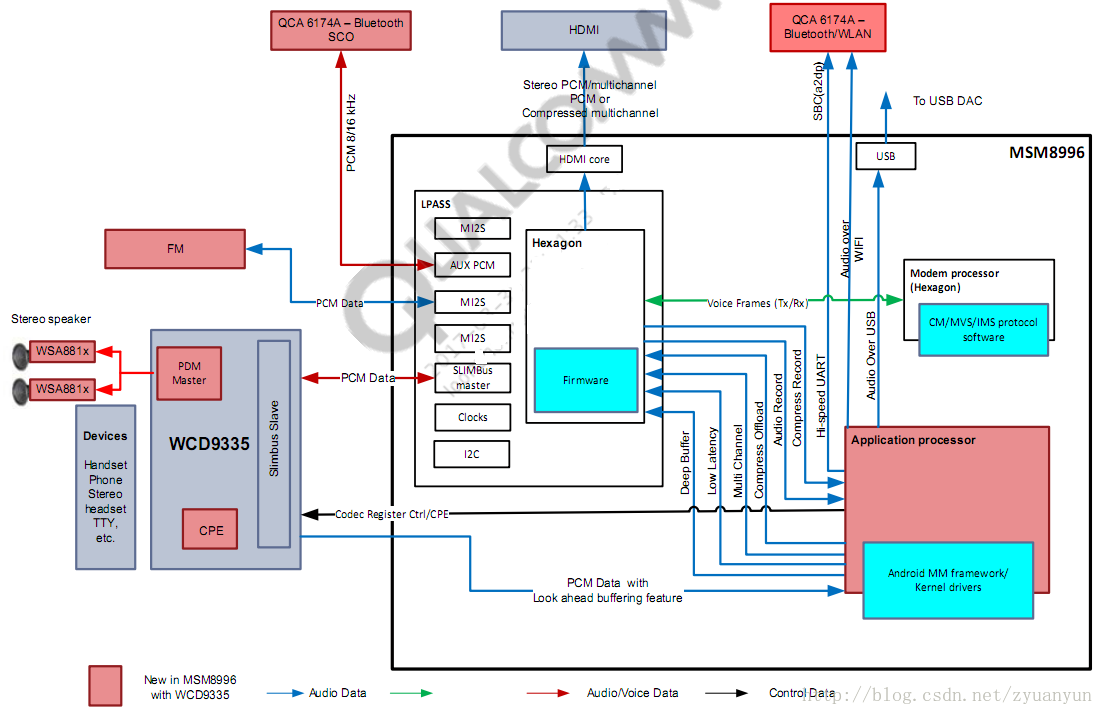
adb remount

adb shell



**Block diagram of audio data stream**

**Qualcomm MSM8996 audio block diagram:**



**Audio Policy Service**: Formulator of audio policy, responsible for audio device switching strategy selection, volume adjustment strategy, etc.  
 **Audio Flinger**: Implementer of audio policy, responsible for management of input and output stream devices and processing and transmission of audio stream data.  
 **Audio HAL**: Audio hardware abstraction layer Responsible for interaction with audio hardware devices, directly called by Audio Flinger.