

# ***Rockchip***

## *Linux Audio Developer Guide*

**Version:** 0.01

**Date:** 2018.06

## **Warranty Disclaimer**

This document is provided by "status" and Fuzhou Rockchip Electronics Co., Ltd ("our company ", the same below) does not provide any express or implied statement or warranty of any accuracy, reliability, integrity, merchantability, specific purpose and non-infringement of any statement, information and content of this document. This document is intended as a guide only for use.

Due to product version upgrades or other reasons, this document may be updated or modified from time to time without notice.

## **Trademarks**

Rockchip and “瑞芯微”、“瑞芯” are trademarks of Fuzhou Rockchip Electronics Co., Ltd. and are exclusively owned by Fuzhou Rockchip Electronics Co., Ltd.

References to other companies and their products use trademarks owned by the respective companies and are for reference purpose only.

## **Copyright © 2018 Fuzhou Rockchip Electronics Co., Ltd.**

Exceeding the scope of reasonable use, No part of this publication may be copied or reproduced without the written permission of our company, and may not be transmitted in any form.

Fuzhou Rockchip Electronics Co., Ltd

Address: No.18 Building, A District, No.89 Software Boulevard, FuZhou, FuJian, PRC

Website: [www.rock-chips.com](http://www.rock-chips.com)

Tel: +86-591-83991906

Fax: +86-591-83951833

Mail: [service@rock-chips.com](mailto:service@rock-chips.com)

# **Preface**

## **Overview**

This document mainly introduces Rockchip audio related development and notes on Linux platform.

## **Chipset model**

Chip Name	Kernel Version
RK3308	4.4.x

## **Applicable object**

This document is mainly suitable for the following engineers:

- Field application engineers
- Software development engineers

## **Revision history**

Revision Date	Version No.	Author	Revision Description
2018.06.27	0.1	Xing Zheng	Initial version of Audio introduction

## **Table of Content**

Chapter 1. Audio on Linux .....	1-1
1.1 Overview .....	1-1
1.2 Preparations of Audio Hardware Environment .....	1-1
Chapter 2. Audio Development.....	2-1
2.1 Common ALSA Tools .....	2-1
2.2 Frequently asked questions .....	2-1

# Chapter 1. Audio on Linux

## 1.1 Overview

At present, audio framework of Linux platform are mainly ALSA (Advanced Linux Sound Architecture). ALSA is a kernel component that provides sound device drivers in Linux and used to replace the original Open Sound System(Open Sound System, OSSv3). In addition to sound device drivers, ALSA also includes a library of user spaces that developers can use with these advanced APIs without interacting directly with kernel driver.

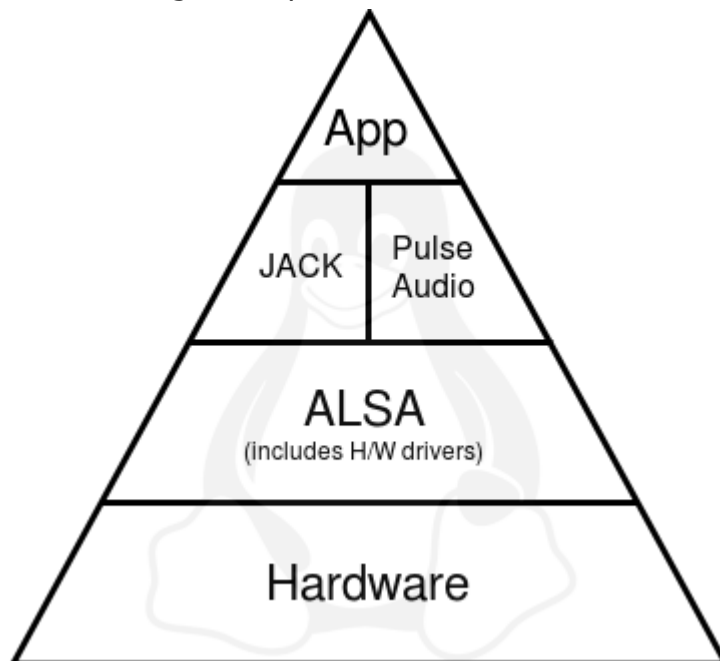


Figure 1 - The Linux Sound Stack 2017

## 1.2 Preparations of Audio Hardware Environment

From the figure 1 we can see that preparations of Hardware are necessary to playback audio. We mainly talk about Embedded Linux environment here and simply introduce the interface and peripheral about audio:

In Embedded Linux platform, SoC usually connect to external Codec device through I2S/PCM interfaces to collect or play audio.

Take I2S interface for instance, connection of SoC and Codec maybe like this:

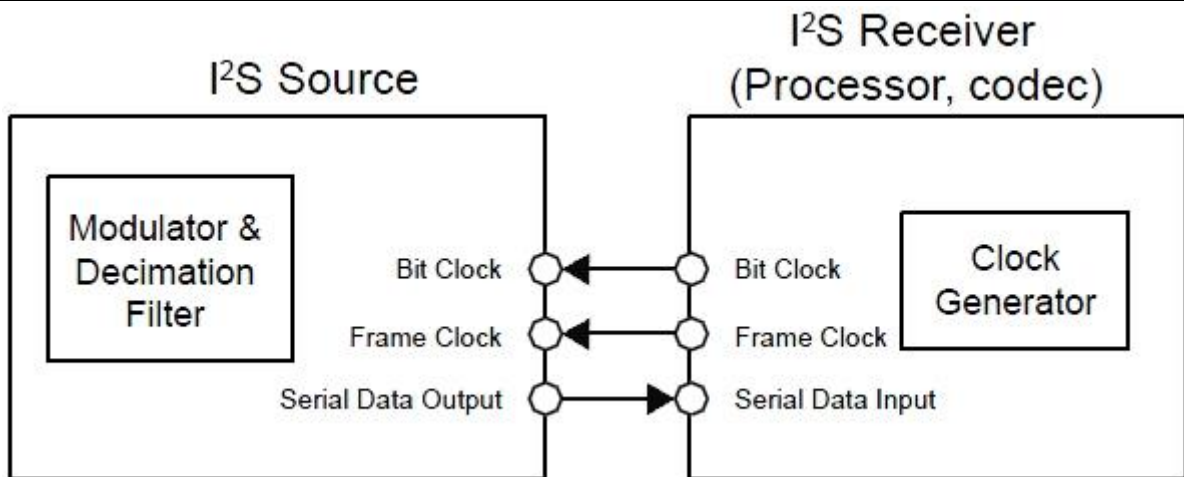


Figure 2 - I2S Bus connection diagram

You can google it if you want more protocols and information about I2S:

<https://en.wikipedia.org/wiki/I%C2%B2S>

If the Audio Hardware environment is ready, you can see the message of codec success binding in your platform start up log.

```
[ 7.401583] ALSA device list:
[ 7.401722] #0: rockchip,rk3308-acodec
```

Figure 3 - ALSA Codec is ready

## Chapter 2. Audio Development

### 2.1 Common ALSA Tools

The alsa-utils provides common ALSA development tools of Linux platform.

- aplay

Audio playback tool which users can use to play audio of WAV format, for example:

```
# aplay sine_440hz_48k_03s.wav
Playing WAVE 'sine_440hz_48k_03s.wav' : Signed 16 bit Little Endian, Rate 48000 Hz, Stereo
```

- arecord

Audio recording tool which users can use to record the sound acquired by microphone, for example recording the audio files with double-channel, 48KHZ sampling rate and 16bit depth and save the record.wav in the /tmp directory:

```
# arecord -r 48000 -c 2 -f S16_LE /tmp/record.wav
Recording WAVE '/tmp/record.wav' : Signed 16 bit Little Endian, Rate 48000 Hz, Stereo
```

- amixer

Audio configuration tool which users can use to configure audio parameters. There are lot of details about using this tool and descriptions in this document are also very detailed, including the previous aplay and arecord, you can refer to

<https://blog.csdn.net/orz415678659/article/details/8866944>

You can use help to view parameters to check more command details.

### 2.2 Frequently asked questions

- To specify the sound card

First you can use the command of aplay -L to check the sound card was loaded normally or not.

```
/ # aplay -L
null
Discard all samples (playback) or generate zero samples (capture)
sysdefault:CARD=rockchiprk3308a
rockchip,rk3308-acodec,
Default Audio Device
sysdefault:CARD=rockchiprk3308p
rockchip,rk3308-pcm,
Default Audio Device
```

As the figure above shows there are two sound cards in the system, the first sound card is needed to play audio, that is card0, so we can use -D to specify the sound card.

```
# aplay -Dhw:0,0 sine_440hz_48k_03s.wav
Playing WAVE 'sine_440hz_48k_03s.wav' : Signed 16 bit Little Endian, Rate 48000 Hz, Stereo
```

The hw:0, 0 means card0:device0. If we need to push audio to the second sound card, change it to hw:1,0

```
# aplay -Dhw:1,0 sine_440hz_48k_03s.wav
Playing WAVE 'sine_440hz_48k_03s.wav' : Signed 16 bit Little Endian, Rate 48000 Hz, Stereo
```

In the same way, when recordings, we can also use arecord -Dhw:0, 0 specify the sound card

for recording.

```
# arecord -D hw:0,0 -r 48000 -c 2 -f S16_LE /tmp/record.wav  
Recording WAVE '/tmp/record.wav' : Signed 16 bit Little Endian, Rate 48000 Hz, Stereo
```

- Preparation of asound.conf file

The aplay and arecord will assigned default device to open when they don't use "-Dhw" parameter, detailed description documents usually in /etc/asound.conf. Users could achieve more complex audio configuration by it, such as description of analog audio devices, sound track mapping, audio path routing and so on. You can refer to the introduction documents on official website for details:

<https://www.alsa-project.org/main/index.php/Asoundrc>