

# 《嵌入式系统》大作业-part2

## 音乐播放器

### 1. 实验介绍

音频信号是一种连续变化的模拟信号，但计算机只能处理和记录二进制的数字信号，由自然音源得到的音频信号必须经过一定的变换，成为数字音频信号之后，才能送到计算机中作进一步的处理。

数字音频系统通过将声波的波形转换成一系列二进制数据，来实现对原始声音的重现，实现这一步骤的设备常被称为（A/D）。A/D 转换器以每秒钟上万次的速率对声波进行采样，每个采样点都记录下了原始模拟声波在某一时刻的状态，通常称之为样本（sample），而每一秒钟所采样的数目则称为采样频率，通过将一串连续的样本连接起来，就可以在计算机中描述一段声音了。对于采样过程中的每一个样本来说，数字音频系统会分配一定存储位来记录声波的振幅，一般称之为采样分辨率或者采样精度，采样精度越高，声音还原时就会越细腻。

上一次作业中，我们已经学会了如何读取音频文件及操作实验平台，本次实验我们将通过 Linux Alsa 音频框架，编程实现音乐播放功能。

### 2. 实验要求

1. 实现 wav 格式音乐播放（60%）
2. 音量调节【能够放大和关小】（20%）
3. 实验文档（20%）

提交时在网络学堂打包提交：1）源码及可执行文件；2）实验文档；3）实验录屏和录音，程序必须要能够在开发板上执行。

本次实验需要在期末现场演示。

### 3. 音频播放驱动介绍

#### 3.1 Alsa 架构介绍

ALSA 全称是 Advanced Linux Sound Architecture，中文音译是 Linux 高级声音体系。ALSA 是 Linux 内核 2.6 后续版本中支持音频系统的标准接口程序，由 ALSA 库、内核驱动和相关测试开发工具组成，更好的管理 Linux 中音频系统。

ALSA 是 Linux 系统中为声卡提供驱动的内核组件。它提供了专门的库函数

来简化相应应用程序的编写。相较于 OSS 的编程接口, ALSA 的函数库更加便于使用。

对应用程序而言 ALSA 无疑是一个更佳的选择, 因为它具有更加友好的编程接口, 并且完全兼容于 OSS。

ALSA 系统包括 7 个子项目:

驱动包 alsa-driver

开发包 alsa-lib

开发包插件 alsa-libplugins

设置管理工具包 alsa-utils

OSS 接口兼容模拟层工具 alsa-oss

特殊音频固件支持包 alsa-firmware

其他声音相关处理小程序包 alsa-tools

ALSA 声卡驱动与用户空间体系结构交互如下图所示:

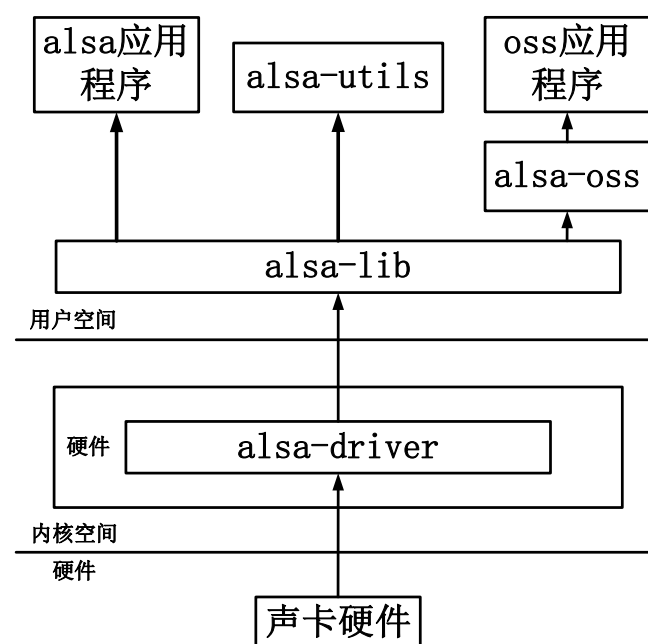


图 1 ALSA 声卡驱动与用户空间交互

### 3.2 Alsa 常用接口说明

从代码角度体现了 alsa-lib 和 alsa-driver 及 hardware 的交互关系。用户层的 alsa-lib 通过操作 alsa-driver 创建的设备文件/dev/snd/pcmC0D0p 等对内核层进行访问。内核层的 alsa-drivier 驱动再经由 sound core 对硬件声卡芯片进行访问。

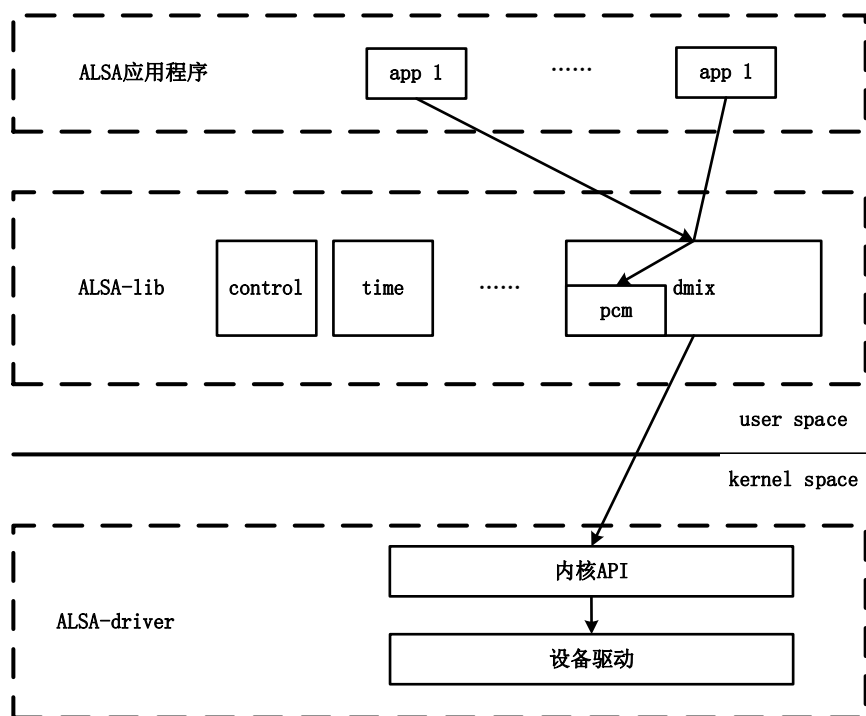


图 2 Alsa 接口访问过程

为了方便操作访问，alsa-lib 中封装了相关接口，通过 pcmCXXDxp/pnCXXDXc 节点 (/dev/snd/pnCXXDXx) 去实现播放、录音功能。

详细 pcm 接口说明可查阅：

<https://www.alsa-project.org/alsa-doc/alsa-lib/pcm.html>

[https://www.alsa-project.org/alsa-doc/alsa-lib/group\\_\\_p\\_c\\_m.html](https://www.alsa-project.org/alsa-doc/alsa-lib/group__p_c_m.html)

这里我们列举几个常用的 API

函数名
snd_pcm_open
snd_pcm_info
snd_pcm_hw_params_any
snd_pcm_hw_params_set_access
snd_pcm_hw_params_set_format
snd_pcm_hw_params_set_channels
snd_pcm_hw_params_set_rate_near
snd_pcm_hw_params_set_buffer_size_near
snd_pcm_hw_params
snd_pcm_sw_params_current
snd_pcm_sw_params
snd_pcm_readi
snd_pcm_writel
snd_pcm_close

## 4. wav 音频播放

## 4.1 开发板音频驱动开发与检测

在开发板执行以下命令：

```
root@myir:/home# aplay -l
```

执行完成后，会输出当前系统的声卡设备，如下图所示：

```
root@myir:/home# aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: WM8960BKRC [WM8960-BKRC], device 0: 4400a004.audio-controller-wm8960-hifi-Playback wm8960-hifi-Play [4400a004.audio-contro
ller-wm8960-hifi-Playback wm8960-hifi-Play]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

图 3 系统声卡设备

检测到系统存在声卡设备进行下面的操作。

## 4.2 链接 ALSA 库

在编译命令后加上 `-lasound`，引入 `alsa` 库。

## 5 框架说明

本次实验提供代码框架。程序入口参数有 `-m`(文件名)，`-f`(采样位)，`-r`(采样频率)。采样位为 `SND_PCM_FORMAT_S16_LE` 表示 16 位小端采样，其他参数自行查阅相关文档资料。

## 6 其他说明

- 本实验需能够完整播放一首 **wav** 格式的音频文件，其他格式音频不予接受，只能播放片段会扣分。
- 可以不使用提供的代码框架，但是需要调用 ALSA 驱动实现音频播放。
- 程序需要对用户的输入有容错机制，不能出现运行过程中出现卡崩的现象。
- 对于音量的调节，需要在音频播放的过程中进行调节，考虑使用多线程结合中断处理实现。