**CODEC 开发文档**

|  |  |  |
| --- | --- | --- |
| **文件状态： [√] 草稿 [ ] 正在修改**  **[ ] 正式发布** | **作者** | **zkh.kaka** |
| **版本** | 0.1 |
|  |  |
| **完成时间** |  |
| **审核** |  |
| **审核时间** |  |
| **密级状态：绝密( ) 秘密( ) 内部资料(√) 公开( )** | | |

**修改记录:**

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 修订者 | 时间 | 说明 |
| 0.1 | Zkh | 2018 / 5 / 12 | Init draft |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**目录**

[文档说明 4](#_Toc11527)

[声音数字化 5](#_Toc27442)

[声音信号的类型 5](#_Toc16433)

[声音数字化过程 5](#_Toc22179)

[声音数字化过程示意图 6](#_Toc18223)

[声音数字化三要素 6](#_Toc15700)

[声卡 6](#_Toc6456)

[声卡的主要功能 6](#_Toc18820)

[芯片类型 7](#_Toc12266)

[数字音频压缩标准 8](#_Toc3399)

[音频压缩方法概述 8](#_Toc8242)

[音频信号压缩编码的分类 8](#_Toc4188)

[音频压缩技术标准 8](#_Toc1868)

[I2S 总线规范简介 9](#_Toc17821)

[I2S的3个主要信号 9](#_Toc29943)

[I2S协议时序 9](#_Toc27892)

[数据格式 9](#_Toc25615)

[ALSA体系架构 11](#_Toc19965)

[ALSA架构简介 11](#_Toc7462)

[一. 概述 11](#_Toc22719)

文档说明**:**

该记录文档主要用于CODEC IC开发。

# 声音数字化

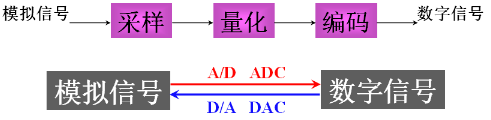
声音是通过一定介质传播的连续的波。

## 声音信号的类型

模拟信号(自然界、物理)

数字信号(计算机)

## 声音数字化过程



**采样**

就是每隔一段时间间隔读一次声音的幅度。单位时间内采样的次数称为采样频率。显然采样频率越高，所得到的离散幅值的数据点就越逼近于连续的模拟音频信号曲线，同时采样的数据量也越大。为了保证数字化的音频能够准确(可逆)地还原成模拟音频进行输出，采样定理要求：采样频率必须大于等于模拟信号频谱中的最高频率的2倍。

常用的音频采样率有：8kHz、11.025kHz、22.05kHz、16kHz、37.8kHz、44.1kHz、48kHz。

例如:话音信号频率在0.3～3.4kHz范围内，用8kHz的抽样频率（fs），就可获得能取代原来连续话音信号的抽样信号，而一般CD采集采样频率为44.1kHz。

**量化**

就是把采样得到的声音信号幅度转换成数字值，用于表示信号强度。

量化精度：用多少个二进位来表示每一个采样值，也称为量化位数。声音信号的量化位数一般是 4,6,8,12或16 bits 。由采样频率和量化精度可以知道，相对自然界的信号，音频编码最多只能做到无限接近，在计算机应用中，能够达到最高保真水平的就是PCM编码，通常PCM约定俗成了无损编码。

**编码**

一个采样率为44.1kHz，量化精度为16bit，双声道的PCM编码输出，它的数据速率则为 44.1K×16×2 =1411.2 Kbps，存储一秒钟需要176.4KB的空间，1分钟则约为10.34M，因此，为了降低传输或存储的费用，就必须对数字音频信号进行编码压缩。

到目前为止，音频信号经压缩后的数码率降低到32至256kbit/s，语音可以低至8kbit/s以下。

对数字音频信息的编码进行压缩的目的是在不影响人们使用的情况下使数字音频信息的数据量最少。通常用如下6个属性来衡量：

—比特率；

—信号的带宽；

—主观/客观的语音质量；

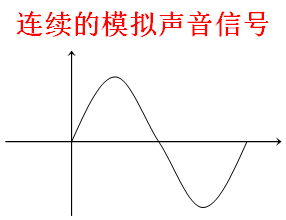
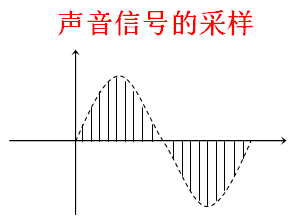
—延迟；

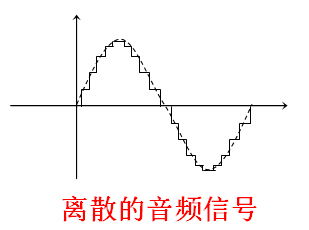
—计算复杂度和对存储器的要求；

—对于通道误码的灵敏度；

为使编码后的音频信息可以被广泛地使用，在进行音频信息编码时需要采用标准的算法。传统会议电视设备主要采用ITU-T推荐的G.711、G.722、G.728和AAC\_LD等音频标准。

## 声音数字化过程示意图



## 声音数字化三要素

|  |  |  |
| --- | --- | --- |
| 采样频率 | 量化位数 | 声道数 |
| 每秒钟抽取声波幅度样本的次数 | 每个采样点用多少二进制位表示数据范围 | 使用声音通道的个数 |
| 采样频率越高  声音质量越好  数据量也越大 | 量化位数越多  音质越好  数据量也越大 | 立体声比单声道的表现力丰富，但数据量翻倍 |
| 11.025kHz  22.05 kHz  44.1 kHz | 8位＝256  16位＝65536 | 单声道  立体声 |

# 声卡

## 声卡的主要功能

声卡是负责录音、播音和声音合成的一种多媒体板卡。其功能包括：

①录制、编辑和回放数字音频文件

②控制和混合各声源的音量

③记录和回放时进行压缩和解压缩

④语音合成技术(朗读文本)

⑤具有MIDI接口(乐器数字接口)

## 芯片类型

CODEC芯片(依赖CPU，价格便宜)

数字信号处理器DSP(不依赖CPU)

# 数字音频压缩标准

## 音频压缩方法概述



压缩编码技术是指用某种方法使数字化信息的编码率降低的技术

音频信号能压缩的基本依据：

①声音信号中存在大量的冗余度；

②人的听觉具有强音能抑制同时存在的弱音现象。

音频信号压缩编码的分类：

①无损压缩(熵编码)

霍夫曼编码、算术编码、行程编码

②有损压缩

波形编码--PCM、DPCM、ADPCM 、子带编码、矢量量化

参数编码--LPC

混合编码--MPLPC、CELP

音频压缩技术标准

|  |  |  |
| --- | --- | --- |
| ****分类**** | ****标准**** | ****说明**** |
| ****电话语****  ****音质量**** | ****G.711**** | ****采样8kHz，量化8bit，码率64kbps**** |
| ****G.721**** | ****采用ADPCM编码，码率32kbps**** |
| ****G.723**** | ****采用ADPCM有损压缩，码率24kbps**** |
| ****G.728**** | ****采用LD-CELP压缩技术，码率16kbps**** |
| ****调幅广****  ****播质量**** | ****G.722**** | ****采样16kHz，量化14bit，码率224(64)kbps**** |
| ****高保真****  ****立体声**** | ****MPEG****  ****音频**** | ****采样44.1kHz，量化16bit，码率705kbps (MPEG三个压缩层次，384-64kbps)**** |

# I2S ****总线规范简介****

I2S(Inter—IC Sound)总线是飞利浦公司为数字音频设备之间的音频数据传输而制定的一种总线标准，该总线专责于音频设备之间的数据传输，广泛应用于各种多媒体系统。

I2S的3个主要信号：

**串行时钟 SCLK**：也叫做位时钟BCLK，即对应数字音频的每一位数据，SCLK的频率=2×采样频率×采样位数 。I2S一般是传输立体声，有两个声道channel，采样频率指得是采样数率，多久去采集一个点，每个点是几个bit组成。在数据传输过程中，I2S总线的发送器和接收器都可以由作为系统的主机来提供系统的时钟频率。

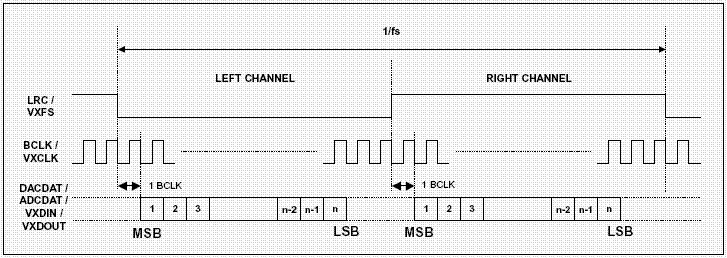
**帧时钟LRCK**：也叫WS(即命令“声道”选择)，用于切换左右声道的数据，LRCK为“0”表示正在传输的是左声道的数据，为“1”表示正在传输的是右声道的数据。LRCLK = FS，就是频率等于采样频率，一般由系统主机提供。

**串行数据SDATA**：就是用二进制补码表示的音频数据，有时为了使系统间能够更好的同步，还需要另外传输一个信号MCLK，称为主时钟，也叫系统时钟（System Clock），是采样频率的256或384倍 。

I2S协议时序：

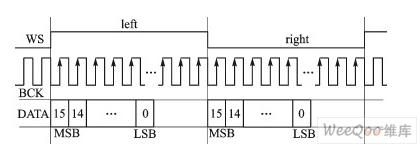
I2S格式的信号无论有多少位数据，数据的最高位总是出现在LRCK变化（也就是一帧开始）后的第2个BCLK脉冲处。这就使得接收端与发送端的有效位数可以不同。如果接收端能处理的有效位数少于发送端，可以放弃数据帧中多余的低位数据；如果接收端能处理的有效位数多于发送端，可以自行补足剩余的位。这种同步机制使得数字音频设备的互连更加方便，而且不会造成数据错。

1个典型的I2S信号时序如下图所示：

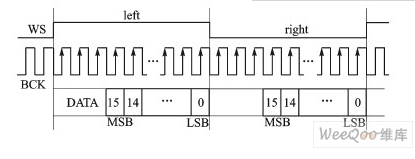


## 数据格式

随着技术的发展，在统一的I2S接口下出现了多种不同的数据格式。根据SD信号相对于SCK和WS位置的不同，分为左对齐和右对齐2种格式，如下图，16位有效位数的左对齐格式：



16位有效位数的右对齐格式：



在上面2种数据传输格式中，当WS为“1”时传输的是左声道的数据，当WS为“0”时传输的是右声道的数据。为了保证数字音频信号的正确传输，发送端和接收端必须采用相同的数据格式。

# ALSA体系架构

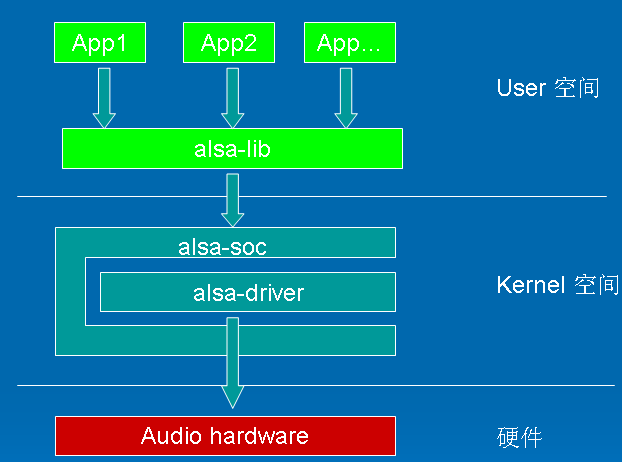
## ALSA架构简介

### 一. 概述

ALSA是Advanced Linux Sound Architecture 的缩写，目前已经成为了linux的主流音频体系结构。

在内核设备驱动层，ALSA提供了alsa-driver，同时在应用层，ALSA为我们提供了alsa-lib，应用程序只要调用alsa-lib提供的API，即可以完成对底层音频硬件的控制。

ALSA软件架构：



在ALSA架构中，用户空间的alsa-lib对应用程序提供统一的API接口，以隐藏了驱动层的实现细节，简化应用程序的实现难度。内核空间中，alsa-soc对alsa-driver的进一步封装，针对嵌入式设备提供了一些列增强的功能。