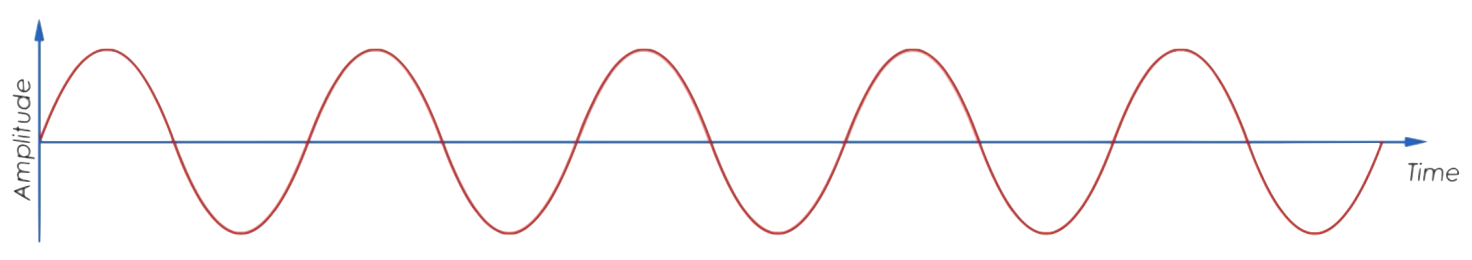
## 一、背景

大家知道人耳能听到声音，是因为声音在音源处产生后，对空气产生了一系列忽强忽弱的压力，这种压力的变化（振动）可以被人耳内部的耳膜感受到，然后经由各种神经细胞，汇总到大脑中形成声音的感觉。

有时候这种压力比想象的要大的多，它不仅可以驱动薄薄的耳膜，当强度足够大时，甚至可以吹动人的头发和衣服。参加过室外音乐节的朋友，如果碰巧在现场比较靠近喇叭的地方，应该有切身的体会。人耳能听到的这种压力振动的范围大概在20Hz到20KHz，Hz即赫兹，即每秒钟振动发生的次数。

除了耳道，身体的其它一些部分也可以感受到声音的振动，例如骨传导耳机就是绕过了耳膜，利用头部骨骼来传递声音的振动。

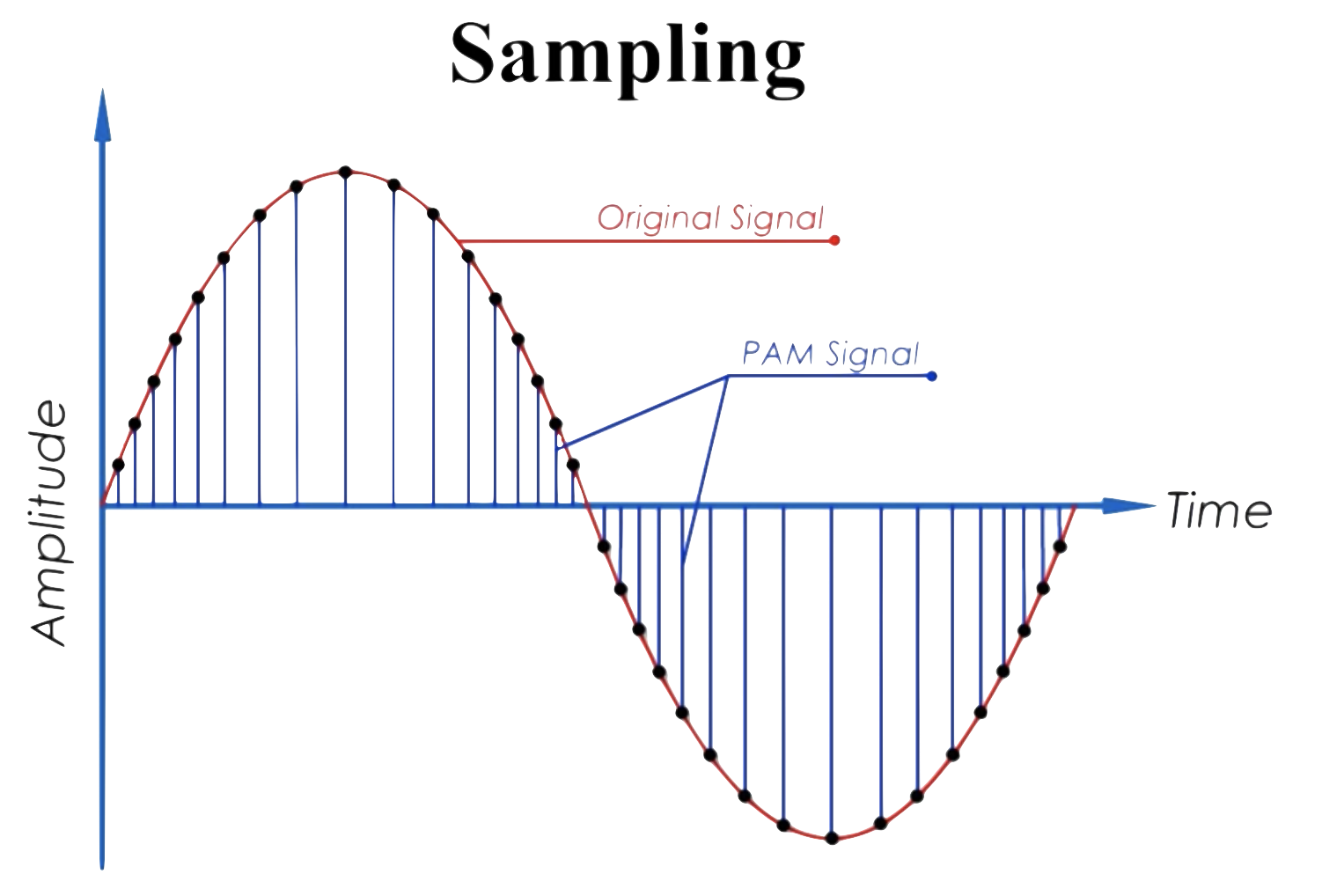
声音是由物体的振动产生的，这种振动引起了周围空气压强的震荡，我们称这种震荡的函数表现形式为波形。

## 声音的三要素：

* 音调，音频的快慢
* 音量，振动的幅度
* 音色，谐波

## 二、定义

现实生活中我们听到的声音是连续的，但是在计算机中只能处理数字信号，不能处理模拟信号，所以我们只能通过采样的方式，把模拟信号转化成数字信号。



假设 44100HZ 16bit stereo: 每秒钟有 44100 次采样, 采样数据用 16 位(2字节)记录, 双声道(立体声)，采集4分钟的数据：44100\*16\*2\*4\*60=338688000bit=40M，这个数据就是没经过压缩的音频数据，我们叫做PCM数据。

PCM(Pulse Code Modulation)，脉冲编码调制。人耳听到的是模拟信号，PCM是把声音从模拟信号转化为数字信号的技术。原理是用一个固定的频率对模拟信号进行采样，采样后的信号在波形上看就像一串连续的幅值不一的脉冲(脉搏似的短暂起伏的电冲击)，把这些脉冲的幅值按一定精度进行量化，这些量化后的数值被连续的输出、传输、处理或记录到存储介质中，所有这些组成了数字音频的产生过程(抽样、量化、编码三个过程)。

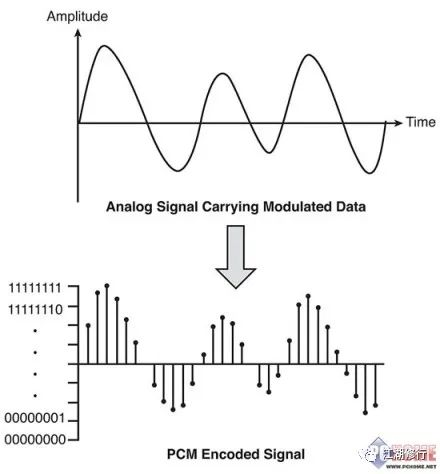
PCM中文称脉冲编码调制(Pulse Code Modulation)，是70年代末发展起来的，记录媒体之一的CD，在80年代初由飞利浦和索尼公司共同推出。

CD中的音频编码标准就是声音通道数为2、采样精度16bit、编码格式为线性PCM、采样率固定是44.1KHz。

要将一段音频模拟信号转换为数字表示，包含如下三个步骤：

1. Sampling(采样)
2. Quantization(量化)
3. Coding(编码)

量化处理过程，就是将时间连续的信号，处理成时间离散的信号，并用实数表示。这些实数将被转换为二进制数用于模拟信号的存储和传输。



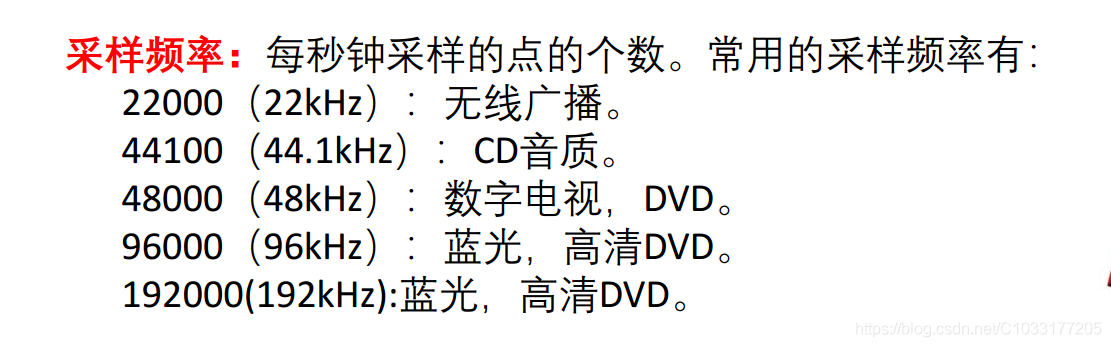
1. 采样率，Sample Rate

采样率表示音频信号每秒的数字快照数。该速率决定了音频文件的频率范围。采样率越高，数字波形的形状越接近原始模拟波形。低采样率会限制可录制的频率范围，这可导致录音表现原始声音的效果不佳。一般数字音频常用的采样率电话频率8kHz、CD频率44.1kHz、DVD频率48kHz。

通常，采样率的单位用Hz表示，例如1Hz表示每秒钟对原始信号采样一次，1KHz表示每秒钟采样1000次。1MHz表示每秒钟采样1百万次。

根据场景的不同，采样率也有所不同，采样率越高，声音的还原程度越高，质量就越好，同时占用空间会变大。

例如：通话时的采样率为8KHz，常用的媒体采样率有44KHz，对于一些蓝光影片采样率高达1MHz。



工科的朋友们可能对奈奎斯特采样定理比较熟悉，就是对声音信号做采样时，采样频率一定要大于声音信号最高频率的两倍。一个通用的声音记录系统，当然需要应对声音的大范围频率，即20KHz声音需要的采样频率至少是40KHz，考虑到后续的各种传输、处理过程带来的损失，采样频率一般会高于40KHz。

常用的采样频率大概分三个系列，分别以8KHz、12KHz、11.025KHz为基准。由此衍生出来的采样频率就有三个系列。

8K、16K、32K、64K、128K

12K、24K、48K、96K、192K

11.025K、22.05K、44.1K、88.2K

1. 采样大小（位深度），Sample Size

位深度决定动态范围。采样声波时，为每个采样指定最接近原始声波振幅的振幅值。较高的位深度可提供更多可能的振幅值，产生更大的动态范围、更低的噪声基准和更高的保真度。普通的CD是16-bit。

1. 声道数，Number of Channels

通道个数。常见的音频有立体声(stereo)和单声道(mono)两种类型，立体声包含左声道和右声道。另外还有环绕立体声等其它不太常用的类型。

1. Sign

表示样本数据是否是有符号位，比如用一字节表示的样本数据，有符号的话表示范围为-128 ~ 127，无符号是0 ~ 255。

1. 字节序，Byte Ordering

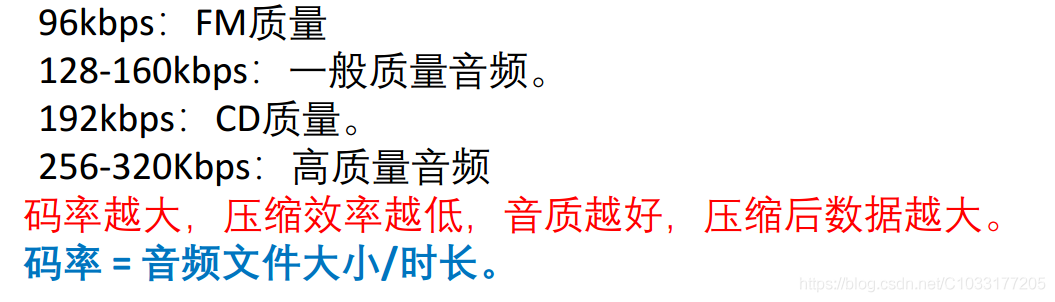
字节序是little-endian还是big-endian。通常均为little-endian。

PCM信号的两个重要指标是采样频率和量化精度，当在播放音乐时，应用程序从存储介质中读取音频数据（MP3、WMA、AAC等），经过解码后，最终送到音频驱动程序中的就是PCM数据，反过来，在录音时，音频驱动不停地把采样所得的PCM数据送回给应用程序，由应用程序完成压缩、存储等任务。

## 三、存储方式

一秒数据大小(码率)=采样大小\*采样率\*声道数 (Kb/s)  
假设采样率为8kHz、声道数2、采样为16bit，时长为1s，则音频数据的大小为：  
1 \*8000 \*16 \*2 = 256000 bit / 8 = 32000 byte / 1024 = 31.25 KB

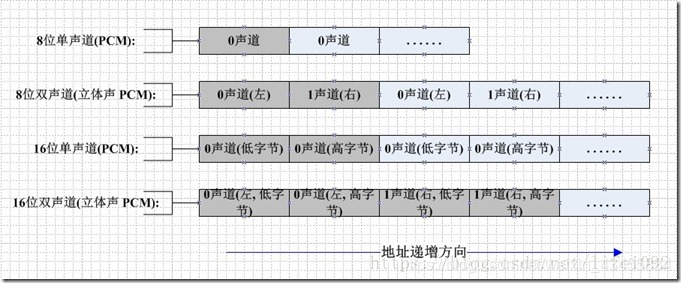
码率，压缩后的音频数据的比特率，常见的码率有：



帧：音频的帧的概念没有视频帧那么清晰，几乎所有视频编码格式都可以简单的认为一帧就是编码后的一张图像。  
帧长：  
可以指每帧采样数播放的时间，mp3 48K，1152个采样点，每帧则为24ms；aac则是每帧是1024个采样点。攒够一帧的数据才送去做编码。  
也可以指压缩后每帧的数据长度，所以需要注意。

每帧持续时间（秒）=每帧采样点数/采样频率（hz）

如果是单声道的音频文件，采样数据按时间的先后顺序依次存入（有的时候也会采用LRLRLR方式存储，只是另一个声道的数据为0），如果是双声道的话就按照LRLRLR的方式存储，存储的时候与字节序有关。



## 四、调节音量、播放速度

### 分离双声道PCM音频数据左右声道的数据

### 按照双声道的LRLRLR的PCM音频数据可以通过将它们交叉的读出来的方式来分离左右声道的数据。

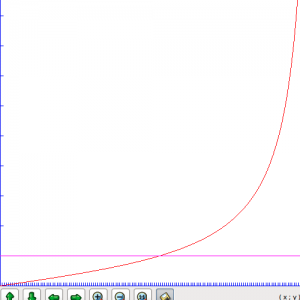
### 调节某个声道的音量

### 一般来说 PCM 数据中的波形幅值越大，代表音量越大，对于 PCM 音频数据而言，它的幅值（即该采样点采样值的大小）代表音量的大小。如果我们需要降低某个声道的音量，可以通过减小某个声道的数据的值来实现降低某个声道的音量。

### 音量控制就是这么简单，但是要注意两点：

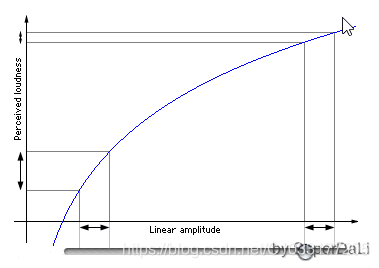
### 若采样点的数据乘以扩大系数之后的值 小于 -32768 或 大于 32768 ，则此处采样的数值只能取 -32768 或 32768

### 我们将采样点的数据乘以2并不代表将声音的音量扩大了两倍，事实上也的确如此。声音音量的增益系数与音量的关系如图：



对数描述：  
平时表示声音强度我们都是用分贝（db）作单位的，声学领域中，分贝的定义是声源功率与基准声功率比值的对数乘以10的数值。根据人耳的心理声学模型，人耳对声音感知程度是对数关系，而不是线性关系。人类的听觉反应是基于声音的相对变化而非绝对的变化。对数标度正好能模仿人类耳朵对声音的反应。所以用分贝作单位描述声音强度更符合人类对声音强度的感知。前面我们直接将声音乘以某个值，也就是线性调节，调节音量时会感觉到刚开始音量变化很快，后面调的话好像都没啥变化，使用对数关系调节音量的话声音听起来就会均匀增大。

如下图所示，横轴表示音量调节滑块，纵坐标表示人耳感知到的音量，图中取了两块横轴变化相同的区域，音量滑块滑动变化一样，但是人耳感觉到的音量变化是不一样的，在左侧也就是较安静的地方，感觉到音量变化大，在右侧声音较大区域人耳感觉到的音量变化较小。



### 调节声音速度，调整采样率

### 若是要改变音频的采样频率，我们只需要对采样点做适当的丢弃或者复制就可以。

### 如10Hz表示每秒钟采样10次，我们只需要将2\*n(n为从0开始的值)处的采样数据舍弃，就可以得到10/2 = 5Hz的采样数据。

### 比如：原始音频为opus编码，单声道，采样率为48kHz，采样点大小为16-bit。如何得到编码为speex，采样率为16kHz，采样大小为16-bit的音频？ 我们需要以下几步：

### 将opus解码为PCM格式数据（叫做PCM1），此时的PCM1的采样率为48kHz

### 将PCM1的数据中第 3\*n（n为从0开始的自然数） 个位置的采样点，丢弃3\*n+1 和3\*n+2位置的采样点。得到PCM2，此时的PCM2采样率为48kHz / 3 = 16kHz

### 将PCM2编码为speex数据

## 五、WAV

WAV即WAVE，是经典的Windows音频数据封装格式，由Microsoft开发。数据本身格式为PCM，也可以支持一些编码格式的数据，比如最近流行的AAC编码。如果是PCM，则为无损格式，文件会比较大，并且大小相对固定，可以使用以下公式计算文件大小。

FileSize = HeadSize + TimeInSecond \* SampleRate \* Channels \* BitsPerSample / 8

其中HeadSize为WAV文件头部长度；

SampleRate，即采样率，可选8000、16000、32000、44100或48000；

Channels表示声道数量，通常为1或2；

BitsPerSample代表单个Sample的位深，可选8、16以及32，其中32位时可以是float类型。

WAV是一种极其简单的文件格式，如果对其结构足够熟悉，完全可以自己通过代码写入WAV文件，从而免去引入一些复杂中间库。特别是在对音频进行调试的时候，能提高效率，降低复杂度。

WAV格式遵循RIFF规范，所有WAV都有一个文件头，记录着音频流的采样和编码信息。数据块的记录方式是小尾端(little-endian)。

以最简单的无损WAV格式文件为例，此时文件的音频数据部分为PCM，比较简单，重点在于WAV头部。一个典型的WAV文件头部长度为44字节，包含了采样率，通道数，位深等信息，如下表所示。



上表为典型的WAV头部格式，从0x00到0x2B总共44字节，从0x2C开始一直到文件末尾都是PCM音频数据。所以如果你已经知道了PCM的采样信息，那么可以直接跳过头部的解析，直接从0x2C开始读取PCM即可，但是对于另一些无损的WAV文件却是不行的。

有一些WAV的头部并不仅仅只有44个字节，比如通过FFmpge编码而来的WAV文件头部信息通常大于44个字节。这是因为根据WAV规范，其头部还支持携带附加信息，所以只按照44个字节的长度去解析WAV头部信息是不一定正确的，还需要考虑附加信息。那么如何知道一个WAV文件头部是否包含附加信息呢？根据"fmt "子块长度来判断即可。

如果fmt SubChunk Size等于0x10(16)，表示头部不包含附加信息，即WAV头部信息长度为44；如果等于0x12(18)，则包含附加信息，此时头部信息长度大于44。