在音频中，无论直播与点播，AAC都是目前最常用的一种音频编码格式，例如RTMP直播，HLS直播，RTSP直播，FLV直播，FLV点播，MP4点播等文件中都是常见的AAC音频。

与MP3相比，AAC是一种编码效果更高，编码音质更好的音频编码格式，常见的使用AAC编码后的文件存储格式为m4a,如果在iphone或者m4a。

FFMpeg可以组成AAC的三种编码器具体如下：

* aac: FFmpeg本身的AAC编码实现
* libaac:第三方的AAC编码器
* libfdk\_aac:第三方的AAC编码器

后面两种编码器为非GPL协议，所以需要注意，在预编译时需要注意采用nonfree的支持。

一个音频帧的播放时间的计算方法：   
一个音频帧的播放时间==一个AAC帧对应的采样样本的个数/采样频率(单位为s)   
(一个AAC原始帧包含一段时间内1024个采样及相关数据 )   
例如：采样率 为44.1KHz的音频帧的播放时间计算方法   
当前AAC一帧的播放时间是= 1024\*1000/44100= 22.32ms(单位为ms)

在众多的音频格式当中，MP3和AAC可以说是使用率最高的两种格式了，尤其是现在主流的流媒体音乐平台几乎都在使用这两种格式。那么问题来了，MP3和AAC格式到底有什么区别？

MP3格式大家都比较熟悉了，它是当今应用最广的数字音乐格式，几乎半数以上的数字音乐下载和流媒体音乐文件均以MP3文件的形式进行存储和传输。当然，MP3本身为有损压缩格式，它舍弃了一些人耳不太敏感的音频信息，确保了更小的体积，更易于下载和存储。

MP3文件为了压缩体积（便于存储），一般在20KHz以上就直接砍掉

MP3格式能够适应各种网络条件和音频质量的需求，但是它的流行却导致了音乐文件在互联网上被广泛“共享”，这是因为MP3一直都没有版权保护技术。在盗版音乐下载的鼎盛时期，MP3招致了各大唱片公司的抗议，司法机构最终也给出了法律的制裁。今天来看，效果还是显著的，而随之孕育出来的就是如今的流媒体音乐，盗版音乐下载已经逐渐成为历史。

AAC的全称是“Advanced Audio Coding”，于2000年问世。AAC与MP3格式相比能够提供更好的音质和更高的编解码效率，同时也是当年苹果iTunes Store生态和传奇数字音乐播放器iPod主要使用的音乐文件格式。

AAC和iPod将CD唱片和CD机赶下了神坛，是的，AAC格式因苹果而发扬光大，也为后来iPod的快速崛起立下了汗马功劳。更重要的是，正是AAC和iPod将CD唱片和CD机赶下了神坛，从此开启了全新的数字音乐时代。如今，苹果的流媒体音乐服务Apple Music使用的依然是标准AAC格式。

在技术上，AAC所采用的运算法则与MP3不同，它的压缩比远超MP3，同时支持更多的采样率及更高的解码容错率。因此，AAC可以在比MP3文件缩小约30%的情况下提供更好的音质，这也使得AAC成为目前最好的有损压缩格式之一。

AAC的音频文件格式有ADIF ＆ ADTS：

* ADIF：Audio Data Interchange Format 音频数据交换格式。

这种格式的特征是可以确定的找到这个音频数据的开始，不需进行在音频数据流中间开始的解码，即它的解码必须在明确定义的开始处进行。故这种格式常用在磁盘文件中。

* ADTS的全称是Audio Data Transport Stream，是AAC音频的传输流格式。

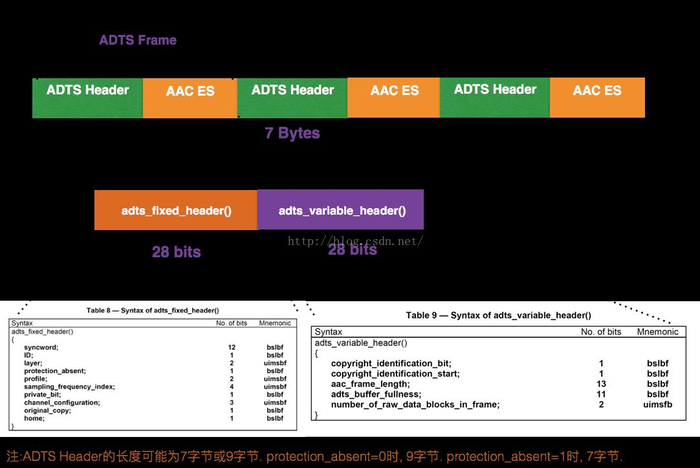
AAC音频格式在MPEG-2（ISO-13318-7 2003）中有定义。AAC后来又被采用到MPEG-4标准中。这种格式的特征是它是一个有同步字的比特流，解码可以在这个流中任何位置开始。它的特征类似于mp3数据流格式。

简单说，ADTS可以在任意帧解码，也就是说它每一帧都有头信息。ADIF只有一个统一的头，所以必须得到所有的数据后解码。且这两种的header的格式也是不同的，目前一般编码后的和抽取出的都是ADTS格式的音频流。

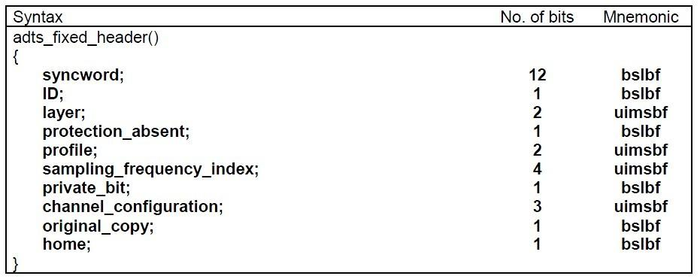
有的时候当你编码AAC裸流的时候，会遇到写出来的AAC文件并不能在PC和手机上播放，很大的可能就是AAC文件的每一帧里缺少了ADTS头信息文件的包装拼接。

只需要加入头文件ADTS即可。一个AAC原始数据块长度是可变的，对原始帧加上ADTS头进行ADTS的封装，就形成了ADTS帧。

AAC音频文件的每一帧由ADTS Header和AAC Audio Data组成。结构体如下：



每一帧的ADTS的头文件都包含了音频的采样率，声道，帧长度等信息，这样解码器才能解析读取。一般情况下ADTS的头信息都是7个字节，分为2部分：  
adts\_fixed\_header();  
adts\_variable\_header();  
其一为固定头信息，紧接着是可变头信息。固定头信息中的数据每一帧都相同，而可变头信息则在帧与帧之间可变。



syncword：同步头总是0xFFF, all bits must be 1，代表着一个ADTS帧的开始

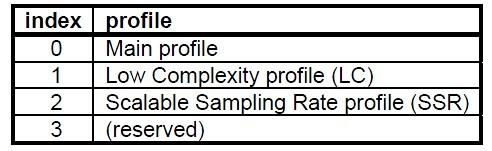
ID：MPEG标识符，0标识MPEG-4，1标识MPEG-2

Layer：always: '00'

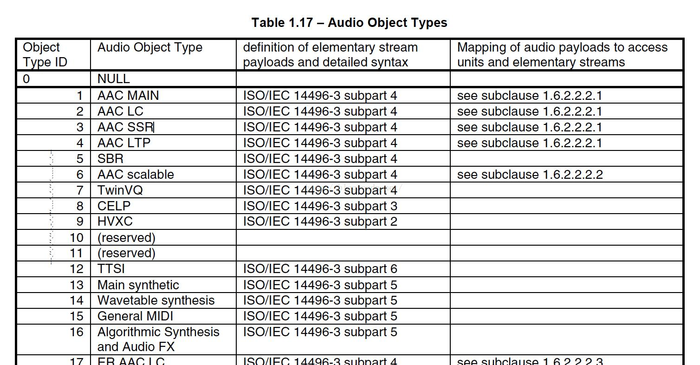
protection\_absent：表示是否误码校验。Warning, set to 1 if there is no CRC and 0 if there is CRC

profile：表示使用哪个级别的AAC，如01 Low Complexity(LC)--- AAC LC。有些芯片只支持AAC LC 。

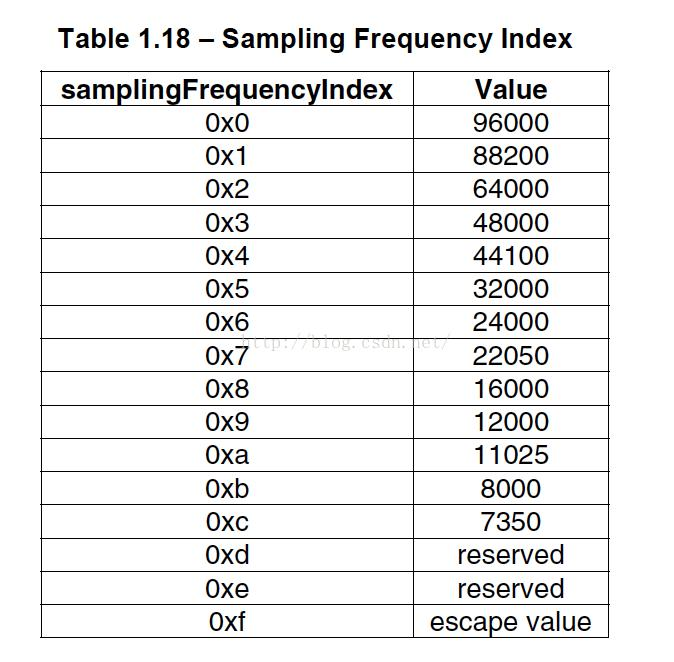
在MPEG-2 AAC中定义了3种：

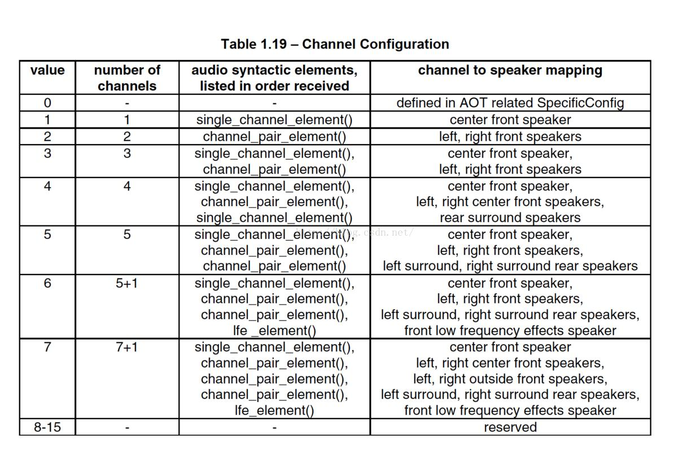


profile的值等于 Audio Object Type的值减1  
profile = MPEG-4 Audio Object Type – 1

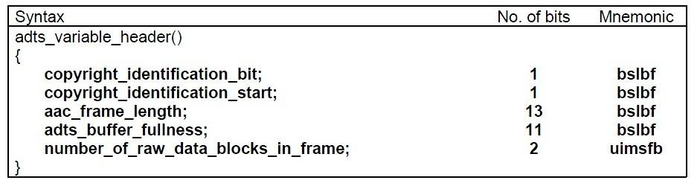


sampling\_frequency\_index：表示使用的采样率下标，通过这个下标在 Sampling Frequencies[ ]数组中查找得知采样率的值。

channel\_configuration: 表示声道数，比如2表示立体声双声道



接下来看下adts\_variable\_header();



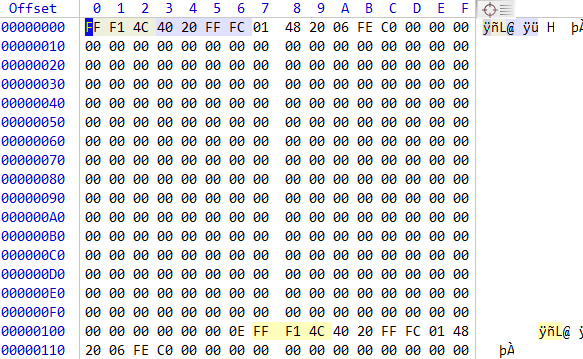
frame\_length: 一个ADTS帧的长度包括ADTS头和AAC原始流.

frame length, this value must include 7 or 9 bytes of header length:  
aac\_frame\_length = (protection\_absent == 1 ? 7 : 9) + size(AACFrame)  
protection\_absent=0时, header length=9bytes  
protection\_absent=1时, header length=7bytes

adts\_buffer\_fullness：0x7FF 说明是码率可变的码流。  
number\_of\_raw\_data\_blocks\_in\_frame：表示ADTS帧中有

number\_of\_raw\_data\_blocks\_in\_frame + 1个AAC原始帧。  
所以说number\_of\_raw\_data\_blocks\_in\_frame == 0 表示说ADTS帧中有一个AAC数据块。

下面是ADTS的AAC文件部分：



第一帧的帧头7个字节为：0xFF 0xF1 0x4C 0x40 0x20 0xFF 0xFC

分析各个关键数值：

111111111111  
0  
00  
1  
01  
0011  
0  
001  
0  
0  
0  
0  
0000100000111(帧长度)  
11111111111  
00

计算帧长度：将二进制 0000100000111 转换成十进制为263。观察第一帧的长度确实为263个字节。

计算方法:（帧长度为13位，使用unsigned int来存储帧长数值）

unsigned int getFrameLength(unsigned char\* str)

{

　　if ( !str )

　　{

　　　　return 0;

　　}

　　unsigned int len = 0;

　　int f\_bit = str[3];

　　int m\_bit = str[4];

　　int b\_bit = str[5];

　　len += (b\_bit>>5);

　　len += (m\_bit<<3);

　　len += ((f\_bit&3)<<11);

　　return len;

}