

# MPEG简介 + 如何计算CBR和VBR的MP3的播放时间

版本：v1.6

Crifan Li

## 摘要

本文主要介绍了MPEG相关的一些基础知识，MP3中常见的术语，以及XING和VBRI头的简介，然后介绍如何计算CBR和VBR的MP3播放时间，最后总结了关于MP3的一些常见的数据的来源。



## 本文提供多种格式供：

在线阅读	<a href="#">HTML</a> <sup>1</sup>	<a href="#">HTMLs</a> <sup>2</sup>	<a href="#">PDF</a> <sup>3</sup>	<a href="#">CHM</a> <sup>4</sup>	<a href="#">TXT</a> <sup>5</sup>	<a href="#">RTF</a> <sup>6</sup>	<a href="#">WEBHELP</a> <sup>7</sup>
下载（7zip压缩包）	<a href="#">HTML</a> <sup>8</sup>	<a href="#">HTMLs</a> <sup>9</sup>	<a href="#">PDF</a> <sup>10</sup>	<a href="#">CHM</a> <sup>11</sup>	<a href="#">TXT</a> <sup>12</sup>	<a href="#">RTF</a> <sup>13</sup>	<a href="#">WEBHELP</a> <sup>14</sup>

HTML版本的在线地址为：

[http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/html/mpeg\\_vbr.html](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/html/mpeg_vbr.html)

有任何意见，建议，提交bug等，都欢迎去讨论组发帖讨论：

[http://www.crifan.com/bbs/categories/mpeg\\_vbr/](http://www.crifan.com/bbs/categories/mpeg_vbr/)

## 修订历史

修订 1.0	2009-09-19	crl
1. 简介MPEG相关知识 2. 详细介绍如何计算CBR和VBR的MP3的播放时间		
修订 1.4	2011-04-24	crl
1. 修正了VBR播放时间的计算公式（原中文说明部分有误） 2. 添加了一些MP3相关的知识点的解释 3. 调整了排版格式		

<sup>1</sup> [http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/html/mpeg\\_vbr.html](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/html/mpeg_vbr.html)

<sup>2</sup> [http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/htmls/index.html](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/htmls/index.html)

<sup>3</sup> [http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/pdf/mpeg\\_vbr.pdf](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/pdf/mpeg_vbr.pdf)

<sup>4</sup> [http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/chm/mpeg\\_vbr.chm](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/chm/mpeg_vbr.chm)

<sup>5</sup> [http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/txt/mpeg\\_vbr.txt](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/txt/mpeg_vbr.txt)

<sup>6</sup> [http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/rtf/mpeg\\_vbr.rtf](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/rtf/mpeg_vbr.rtf)

<sup>7</sup> [http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/webhelp/index.html](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/webhelp/index.html)

<sup>8</sup> [http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/html/mpeg\\_vbr.html.7z](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/html/mpeg_vbr.html.7z)

<sup>9</sup> [http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/htmls/index.html.7z](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/htmls/index.html.7z)

<sup>10</sup> [http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/pdf/mpeg\\_vbr.pdf.7z](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/pdf/mpeg_vbr.pdf.7z)

<sup>11</sup> [http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/chm/mpeg\\_vbr.chm.7z](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/chm/mpeg_vbr.chm.7z)

<sup>12</sup> [http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/txt/mpeg\\_vbr.txt.7z](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/txt/mpeg_vbr.txt.7z)

<sup>13</sup> [http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/rtf/mpeg\\_vbr.rtf.7z](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/rtf/mpeg_vbr.rtf.7z)

<sup>14</sup> [http://www.crifan.com/files/doc/docbook/mpeg\\_vbr/release/webhelp/mpeg\\_vbr.webhelp.7z](http://www.crifan.com/files/doc/docbook/mpeg_vbr/release/webhelp/mpeg_vbr.webhelp.7z)

---

修订 1.5	2011-07-02	crl
1. 详解两种CBR播放时间的计算公式		
修订 1.6	2012-08-09	crl
1. 通过Docbook发布		

---

---

## MPEG简介 + 如何计算CBR和VBR的MP3的播放时间:

Crifan Li

版本 : v1.6

出版日期 2012-08-09

版权 © 2012 Crifan, <http://crifan.com>

本文章遵从 : [署名-非商业性使用 2.5 中国大陆\(CC BY-NC 2.5\)](http://creativecommons.org/licenses/by-nc/2.5/)<sup>15</sup>

---

<sup>15</sup> [http://www.crifan.com/files/doc/docbook/soft\\_dev\\_basic/release/html/soft\\_dev\\_basic.html#cc\\_by\\_nc](http://www.crifan.com/files/doc/docbook/soft_dev_basic/release/html/soft_dev_basic.html#cc_by_nc)

---

# 目录

正文之前 .....	ix
1. 声明 .....	ix
2. 此文目的 .....	ix
1. MPEG的相关知识 .....	1
1.1. MPEG是啥 .....	1
1.2. 为啥没有了MPEG-3 .....	1
1.3. MPEG2.5又是啥 .....	1
1.4. MPEG中不同帧之间的关系 .....	2
1.5. MPEG-1和MPEG-2音频特性 .....	2
1.5.1. MPEG-1音频 ( ISO/IEC 11172-3 ) .....	2
1.5.2. MPEG-2音频 ( ISO/IEC 13813-3 ) .....	2
1.5.2.1. MPEG-2/LSF的特性 .....	2
1.5.2.2. MPEG-2/多声道 的特性 .....	2
1.5.3. 什么是ISO/IEC 11172-3和ISO/IEC 13818-3 .....	2
2. MP3中常见的术语 .....	3
2.1. 啥叫MP3 .....	3
2.2. 什么是LSF .....	3
2.3. 什么是帧 ( frame ) .....	3
2.4. 什么是帧头 ( Frame Header ) .....	3
2.5. 啥是CBR和VBR .....	3
2.6. 比特率 ( Bitrate ) .....	4
2.7. 边信息 ( Side Information ) .....	4
2.8. MP3的TAG .....	4
3. MPEG的帧Frame .....	5
3.1. MPEG帧头 ( Frame Header ) .....	5
3.1.1. 举例说明MPEG帧头的含义 .....	7
3.2. MPEG音频数据 .....	7
3.3. MPEG帧的大小 .....	8
3.3.1. 常见MPEG帧的音频数据大小是418字节 .....	9
4. MP3的播放时间的计算公式及XING和VBRI头介绍 .....	11
4.1. CBR的MP3的播放时间(duration)计算方法 .....	11
4.1.1. 计算CBR的MP3的播放时间的方法之一 .....	11
4.1.2. 计算CBR的MP3的播放时间的方法之二 .....	11
4.1.3. 计算CBR的MP3的播放时间的两种方法的总结 .....	12
4.1.3.1. 举例说明两种方法计算出结果的不同 .....	13
4.2. VBR的MP3的播放时间(duration)计算方法 .....	14
4.2.1. 平均比特率法 .....	14
4.2.2. 总帧数法 .....	14
4.2.3. VBR的两种Header : XING和VBRI .....	16
4.2.3.1. Xing TAG / Xing头 ( header ) .....	16
4.2.3.2. VBRI头 ( header ) .....	17
5. 计算CBR和VBR的MP3文件的播放时间的步骤 .....	19
5.1. 定位到MPEG的帧头的位置 .....	19
5.2. 解析MPEG帧头, 获取必要信息 .....	19
5.3. 判断是VBR还是CBR, 根据公式计算播放时间 .....	19
5.3.1. 定位出Xing头的位置 .....	19
5.3.2. 如果有Xing头, 则是VBR, 解析XING头, 计算播放时间 .....	19
5.3.3. 如果没XING头, 定位出VBRI头位置, 找VBRI头 .....	20
5.3.4. 如果也没找到VBRI头, 则是CBR, 计算CBR的播放时间 .....	20
6. MP3相关的一些知识点的解释 .....	21
6.1. MP3的文件的内容组织结构 .....	21
6.2. MP3帧的时长是26ms的来历 .....	21
6.3. 怎么算出来MP3的压缩比大概是1:11的 .....	22
6.4. 怎么算出来MP3的压缩比大概是1:11的 .....	23
6.4.1. 原因之一 .....	23

6.4.2. 原因之二 .....	23
7. 后记 .....	24
参考书目 .....	25

---

## 插图清单

3.1. MPEG帧头含义举例 .....	7
-----------------------	---

---

## 表格清单

3.1. MPEG音频的帧头的格式 .....	5
3.2. MPEG比特率索引表 (单位: Kbit/s) .....	6
3.3. MPEG帧的采样率索引表 (单位: Hz) .....	6
3.4. MPEG帧的采样数索引表 (单位: 个/帧) .....	7
4.1. XING 头的格式及含义 .....	16
4.2. VBRI头的格式及含义 .....	17
5.1. MPEG Layer III的边信息 (side information) (单位: 字节) .....	19
6.1. MP3文件的内容组织结构 .....	21

---

## 公式清单

3.1. Frame Data Size .....	8
3.2. 帧数据大小 .....	9
4.1. CBR播放时间 ( CBR Duration ) .....	11
4.2. VBR MP3总的时长 ( VBR Duration ) .....	15
4.3. XING头位置 .....	17



---

# 正文之前

## 1. 声明

本文所写内容，多数整理自互联网，版权归原作者所有

笔者知识有限，文中难免有误，欢迎批评指正，admin (at) crifan.com

觉得此文对你有帮助，想要发邮件来感谢的，也欢迎哈，^\_^

欢迎盗版，盗版不究，但请转载时注明原作者

## 2. 此文目的

了解MPEG相关知识

了解MP3的常见术语的含义

详解VBR MP3的帧头格式及含义

搞懂如何去计算CBR和VBR的MP3文件的播放时间 ( duration )

---

# 第 1 章 MPEG 的相关知识

想要了解如何计算VBR的MP3的播放时间之前，要了解一些和MP3相关的一些基本概念，其中主要是MPEG的相关知识和编解码的一些基础知识。

## 1.1. MPEG是啥

MPEG全名Moving Pictures Experts Group，动态图像专家组，简单说就是一个专家组，专门研究一些音视频规范的，所以才叫专家，不是我们国家的“砖家”哦。

这个专家组是在ISO/IEC (International Standards Organization/International Electrotechnical Commission,国际标准化组织/国际电工委员会) 联合指导下成立的。

这个组，专门去研究出一个数字音视频的压缩相关的规范，所以最后研究出适用于不同应用环境的N多规范。

和事物发展的过程类似，研究出这么多的规范也是，不同时期，不同的版本，针对不同的应用。也是由简到繁。

并且，命名规则都是，按照阿拉伯数字从小到大的：MPEG1，MPEG2，MPEG4，MPEG-7，最新版本，好像都有MPEG-21了。

## 1.2. 为啥没有了MPEG-3

估计有人纳闷了，中间的MPEG3咋没了呢？

是没MPEG3，当然，不是被刘谦变魔术变没了，而是由于当时设计者没有规划好，导致已经设计好的MPEG2，性能太好了，都能干本来打算让MPEG3干的活了，所以后来干脆就不去再设计MPEG3了，原定计划就取消了，也就没了MPEG3。

看来这个MPEG3，待遇貌似不比胎死腹中好多少。对于很多人误解的，以为MP3就是MPEG-3，也就错的离谱了。

关于MP3的名称来历，下面会再解释。

## 1.3. MPEG2.5又是啥

MPEG2.5，简单说就是出身不正，不是官方推出的规范。MPEG 2.5是针对MPEG2的一个非官方的扩展版本，支持更低的采样率。关于其更多解释，网上找到这些：MPEG声音标准提供三个独立的压缩层次：Layer I、Layer II和Layer III。用户具体选哪个Layer，可以根据自己的要求，在权衡复杂性和声音质量之后，做出自己的选择。

1. Layer I的编码器最为简单，编码器的输出数据率为384 kb/s，主要用于小型数字盒式磁带(digital compact cassette，DCC)。
2. Layer II的编码器的复杂程度属中等，编码器的输出数据率为256 kb/s~192 kb/s，其应用包括数字广播声音(digital broadcast audio，DBA)、数字音乐、CD-I(compact disc-interactive)和VCD(video compact disc)等。
3. Layer III的编码器最为复杂，编码器的输出数据率为64 kb/s，主要应用于ISDN上的声音传输。

对于Layer III：

- A. MPEG-1 Layer III支持的采样率为32,44.1,48khz,比特率支持32---320kbps
- B. MPEG-2 Layer III支持的采样率为16,22.05,24khz,比特率支持8---160kbps。Fraunhofer对此又进行扩展，将原来MPEG-2所支持的低采样率再除以2，得到：8, 11.025, 和 12 kHz，比特率跟MPEG-2相同，称为 "MPEG 2.5"。

## 1.4. MPEG中不同帧之间的关系

对于Layer I和Layer II,不同的帧之间,是互相独立的。也就意味着,你可以任意截取MPEG的音频文件,然后找到第一个正确的帧头,后开始解码,对于余下的帧,都是同样的处理,解码后,进行播放,这样都可以正确的播放。

而对于Layer III,所有帧不保证都是互相独立的。此处不保证完全独立,指的是当前的帧和前后临近的那些帧,不一定完全独立,有可能有一定关系。这是由于可能用到“字节蓄水池 (byte reservoir)”的技术,即内部的一个数据缓存,当前帧和其后的一些帧,都是相关的,最差情况下,要连续缓冲保存9个帧,才能对第一帧解码。

## 1.5. MPEG-1和MPEG-2音频特性

### 1.5.1. MPEG-1音频 ( ISO/IEC 11172-3 )

MPEG-1音频 ( ISO/IEC 11172-3 ) 描述了三层音频编码,具有如下特性:

1. 一个或两个音频声道
2. 采样率32KHz、44.1KHz或48KHz
3. 比特率从32Kbps到448Kbps
4. 每一层都有其自己的其他特点。

### 1.5.2. MPEG-2音频 ( ISO/IEC 13813-3 )

MPEG-2音频 ( ISO/IEC 13813-3 ) 包含了对MPEG-1的两种扩展。通常称为MPEG-2/低采样率 ( LSF ) 和MPEG-2/多声道 ( Multichannel ) 。

#### 1.5.2.1. MPEG-2/LSF的特性

1. 一个或两个音频声道
2. 采样率只有MPEG-1的一半
3. 比特率从8Kbps到256Kbps

#### 1.5.2.2. MPEG-2/多声道 的特性

1. 多达5个全范围的音频声道和一个LFE ( Low Frequency Enhancement , 也叫做 重低音 ) 声道
2. 采样率和MPEG-1相同
3. 对于5.1声道,最高的比特率可达1Mbps

### 1.5.3. 什么是ISO/IEC 11172-3和ISO/IEC 13818-3

由于MPEG只是ISO/IEC下面的一个组织,所以,关于MPEG音频部分的规范,也都是出自ISO/IEC之手。

因此,ISO/IEC 11172和ISO/IEC 13818,其实就是MPEG-1和MPEG-2的别名。

另外,由于MPEG-1和MPEG-2,每个都分好几个部分,其中,第3部分是关于音频 ( Audio ) 的。

所以,ISO/IEC 11172-3和ISO/IEC 13818-3,就分别对应着MPEG-1的音频, MPEG-2的音频,也就是我们常常提到的MPEG的音频文件所对应的规范。

---

## 第 2 章 MP3中常见的术语

知道了MPEG的来龙去脉后，在了解MP3的播放时间如何计算之前，还要知道其他一些MP3相关的知识及常见的术语：

### 2.1. 啥叫MP3

注意，这个MP3，不是MPEG-3，但是为何叫MP3，是因为：

MPEG规范中规定了，每一个版本的MPEG，比如MPEG1，MPEG2等，都有三种不同的Layer，不同Layer的序号命令是以罗马数字的，所以叫做Layer I，Layer II，Layer III。其中，MPEG-1或MPEG-2的Layer III，被称为MP3，而其中最常见的是MPEG 1 的Layer III，所以，被大家所熟知的MP3，一般都指的是MPEG-1的Layer III。

即MPEG-1的Layer III被简称为MP3。

根据事物发展由简到繁的规律，我们知道，Layer III，相对Layer I和Layer II，有着更复杂的压缩算法。正是其相对复杂，用了很多算法，比如声学上的掩蔽效应（masking effect），Huffman压缩等，才使得可以实现，在尽可能保持音质的前提下，极大地减少了音频文件大小，也就是说，尽量让你听上去音频声音和音质都没啥变化，但是MP3的文件大小，相对于原先没处理的音频数据或者其他格式的，比如WAV格式等，要小很多。

音质满足大家的要求，文件又小，在互联网时代，就非常方便大家互相交流传播。

这也是MP3如此流行的主要原因之一。

### 2.2. 什么是LSF

MPEG2/2.5 也常被简称为LSF（Low Sampling Frequencies），低采样率。

### 2.3. 什么是帧（frame）

帧，即数据帧，通俗点说，就是一段数据，一块数据，数据块。

对于MPEG音频文件本身，并没有什么文件头，而是由很多数据块所组成，这样的单个的数据块，就叫做一个（数据/音频）帧（frame）。

而MP3文件，就是由很多个帧所组成。

帧，也是其他很多音视频技术中的基本单位。

### 2.4. 什么是帧头（Frame Header）

每一个帧里面，包含了帧头和音频数据。帧头，就是在帧的头部，有一定长度的数据，用于描述改帧音频的一些参数，用于解码器识别相关音频参数，用于对音频帧解码。

对于MPEG的帧头，是固定的32 比特，即4字节。

### 2.5. 啥是CBR和VBR

CBR（Constant BitRate）固定（/不变）比特率，VBR（Variable BitRate），不定（/可变）比特率。此处的固定与可变，指的是MP3的比特率，而不是指采样率。

音频文件可以被编码器编码成CBR或VBR。

VBR意思就是，每一帧的比特率都不一定相同（当然，很有可能临近的一些帧的比特率相同）。由于VBR在编码的时候，根据当前采样的声音数据的复杂度，去判断是采用何种比特率。比如对于很复杂的

声音，那么就用更多的比特位去编码，如果是很简单的声音甚至是无声（silence）的数据，那么就可以用很少的比特位去编码，这样不同情况下，所产生的MP3音频数据的大小，是不同的，但是却可以一直保持同样的音质。

因此，总体来说，对于同样大小的MP3文件，VBR的音质一般要比CBR的好。

很多种音视频文件，都可以按照VBR来压缩，比如MP3、WMA、OGG Vorbis、AAC，MPEG-2的视频等。

VBR的优点是，和CBR相比，用更小的空间，即文件更小，实现更高的音质。缺点是，编码复杂度增加，编码和解码都需要更长时间，而且很早之前，有些硬件编码器可能和VBR不兼容。不过现在一般硬件的音频解码器，都可以很好的兼容VBR了。

## 2.6. 比特率 ( Bitrate )

即每秒包含了多少个比特的数据。比特率常用Kbps ( kilo bits per second , 千比特每秒 ) 表示。注意，此处的千比特=1000 bits，而不是1024。

## 2.7. 边信息 ( Side Information )

在MPEG音频的帧头的后面，有一些解码器会用到的一些信息，用于解码器控制音频流的播放，它就叫做边信息。

## 2.8. MP3的TAG

MP3文件中，会在一些位置（文件最开始或者是最后，或者其他某个固定位置），存放一些TAG，即标签，用于描述文件的相关信息，其中最常见的一些，比如，MP3音乐的歌手名，专辑名，专辑年份，曲风等等。

关于MP3的TAG，现在常见的有ID3v1（ID3的第一版），ID3v2（ID3的第二版），APEv2等。

同一文件中，一般只存在某一种TAG，即，是ID3v2或APEv2或ID2v1或其他。

与此相关的应用方面，我们用千千静听去播放MP3的时候，播放器中点击文件属性，还可以看到有MP3标签的读取优先级方面的设置，默认设置为APEv2 > ID3v2 > ID3v1。

如果MP3文件中存在ID3的TAG话：

ID3v1会放在音频文件的最后，大小128个字节，其中前三个字节是字符“TAG”。

ID3v2一般放在音频文件的开头处，前三个字节是字符“ID3”。

关于ID3的具体格式，请参考附录中的引用文献。

MP3的TAG，只是用于存储歌曲等方面的辅助描述信息，与MP3的解码，没啥关系，故此处不在过多介绍。

# 第 3 章 MPEG的帧Frame

MPEG音频文件，由一个个的帧 ( Frame ) 组成。

每一MPEG帧都有个帧头 ( Frame Header ) ，位于帧的最开始处，接下来的是音频数据(Audio Sample/Audio Data) ，即：

MPEG帧 = MPEG帧头 + MPEG音频数据

下面分别详细介绍MPEG帧头和音频数据的细节内容：

## 3.1. MPEG帧头 ( Frame Header )

MPEG音频的三种Layer的，尽管他们的压缩方法各不相同，但是帧头格式都一样。

先说一下大小，MPEG帧头，共32bit=4字节。

然后再看具体的格式及含义，如下所示：

表 3.1. MPEG音频的帧头的格式

位置 (bit)	长度 (bit)	含义	示例
0	11	用于同步帧，找到此帧头 ( 所有位均置1 )	1111 1111 111
11	2	<b>MPEG音频的版本ID</b>	11
		00 MPEG 2.5 (MPED-2的非官方扩展版本)	
		01 保留	
		10 MPEG 2 (ISO/IEC 13818-3)	
		11 MPEG 1 (ISO/IEC 11172-3)	
13	2	<b>Layer的索引</b>	01
		00 保留	
		01 Layer III	
		10 Layer II	
		11 Layer I	
15	1	<b>保护位</b>	1
		0 用16位的CRC保护下面的帧头	
		1 无CRC	
16	4	详见： <a href="#">表 3.2 “MPEG比特率索引表 ( 单位：Kbit/s ) ”</a>	1001
20	2	详见： <a href="#">表 3.3 “MPEG帧的采样率索引表 ( 单位：Hz ) ”</a>	11
22	1	填充位。如果设置了此位，就会对每帧数据填充一个slot (对于帧大小的计算很重要)	0
23	1	私有位 (仅用于标示性的)	1
24	2	<b>声道的模式</b>	01
		00 立体声	
		01 混合立体声	
		10 <a href="#">双声道</a> (两个单声道)	
		11 一个声道 (单声道)	

位置 (bit)	长度 (bit)	含义	示例
26	2	模式扩展 (仅用于联合立体声) (注: 此处不是本文重点, 故忽略相关的索引表)	00



### 双声道

双声道文件由两个独立的单声道所组成。大多数解码器把双声道输出成立体声, 但是实际上, 不是所有的双声道都是立体声的。

表 3.2. MPEG比特率索引表 (单位: Kbit/s)

比特率索引	MPEG 1			MPEG 2, 2.5 (LSF)	
	Layer I	Layer II	Layer III	Layer I	Layer II & III
0000	空闲				
0001	32	32	32	32	8
0010	64	48	40	48	16
0011	96	56	48	56	24
0100	128	64	56	64	32
0101	160	80	64	80	40
0110	192	96	80	96	48
0111	224	112	96	112	56
1000	256	128	112	128	64
1001	288	160	128	144	80
1010	320	192	160	160	96
1011	352	224	192	176	112
1100	384	256	224	192	128
1101	416	320	256	224	144
1110	448	384	320	256	160
1111	空闲				



### 常见MP3的比特率

**强调部分**: 就是我们此处所关心的MP3 (MPEG-1, Layer III) 的比特率。

**红色部分**: 即128kbps, 192kbps, 320kbps等, 就是我们常见的MP3的比特率。

表 3.3. MPEG帧的采样率索引表 (单位: Hz)

采样率索引	MPEG 1	MPEG 2 (LSF)	MPEG 2.5 (LSF)
00	44100 Hz	22050 Hz	11025 Hz
01	48000 Hz	24000 Hz	12000 Hz
10	32000 Hz	16000 Hz	8000 Hz
11	空闲		



### 常见MP3采样率

**强调部分**: 就是我们最常见的MP3 (MPEG-1) 的采样率44100Hz=44.1K Hz

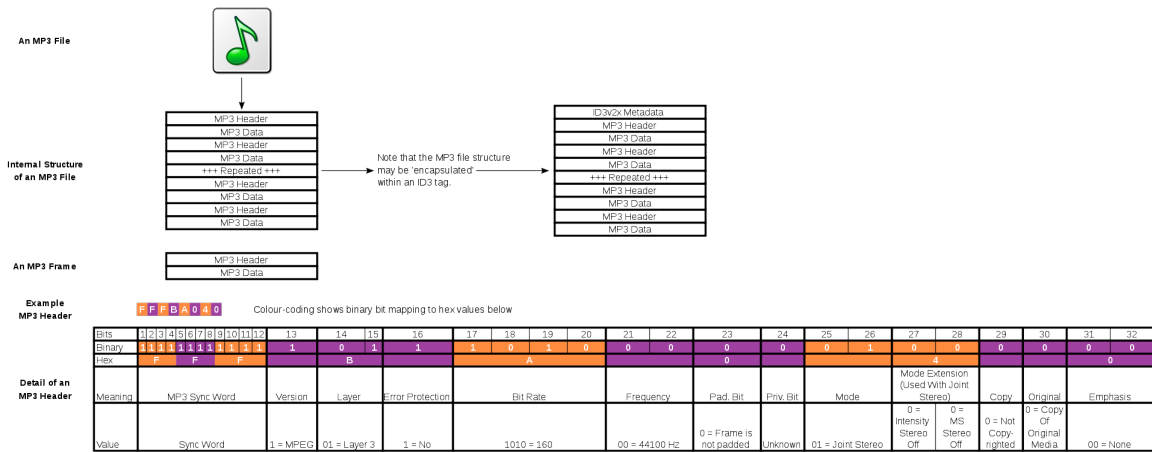
从上面MPEG帧头的格式中可以看出，MPEG的音频，都是有相对固定的比特率，采样率等参数，关于这每一帧的这些参数的具体值，都是找到索引值，然后查对应的索引表，而得知具体的值。

不过，顺便提一点，在MPEG标准中，也描述了一种自由格式（free format），这种自由格式意思为用一个固定比特率对文件进行编码，而此固定的比特率不是我们那些索引表中所预定义好的。对于这类自由格式的MPEG音频，一般的解码器都无法解码。

### 3.1.1. 举例说明MPEG帧头的含义

下图给出了一般MP3的格式及举例说明了MPEG的帧头所对应的信息：

图 3.1. MPEG帧头含义举例



## 3.2. MPEG音频数据

MPEG帧，除了开始部分的MPEG帧头外，余下的就是MPEG的音频数据。

需要注意的，MP3中的帧，是MPEG帧，其中的音频数据部分，是经过MP3的相关算法压缩后的数据，而不是原始采样过来的数据。

MPEG音频数据部分，包含了**固定数目**的音频采样（Audio Sample）。

其中关于采样个数：

MPEG的不同规范（MPEG-1/2/3），以及同一规范中不同的Layer(Layer I/II/III)，每一帧所对应的采样数，都是固定的，其具体的值参见下表：

表 3.4. MPEG帧的采样数索引表（单位：个/帧）

	MPEG 1	MPEG 2 (LSF)	MPEG 2.5 (LSF)
Layer I	384	384	384
Layer II	1152	1152	1152
Layer III	1152	576	576



### 常见MP3每帧采样数

**强调部分：**就是我们此处所关心的，MPEG-1，Layer III，即MP3，不论是CBR还是VBR，文件中的每一帧，其采样的个数/采样数，都是固定的**1152**个。

而后面要介绍如何计算VBR MP3的播放时间，正是基于此前提：



MP3，即MPEG-1，Layer III，不论是CBR，还是VBR，每一帧的采样个数都是固定的1152个。即每一帧，都是固定的1152个采样。

同时，我们还要注意另外一点，那就是对于被某个编码器将原始音频数据编码为MP3数据后，得到的MP3文件，对于此单个文件，其采样率，始终都是一样的。也就是说，如果解析MP3的第一帧MPEG头得到的采样率是44100Hz的话，那么此MP3文件后面的所有的帧的采样率，也肯定都是44100Hz，即这个采样率，对于同一个MP3文件来说，是固定的。

于此相对的是，VBR中的Variable Bitrate，中的Variable，可变的，指的是，变化的比特率，而不是采样率。

总结一下就是：

CBR和VBR中的固定和可变，都是指的是比特率Bitrate，而不是采样率Sample Rate。对于同一MP3文件，不论CBR还是VBR，采样率都是固定的。

因为该采样率对应着此MP3被编码器编码的那一时刻，编码器的采样率也是最开始的时候就设置好，并且之后不会再变化（除了你重新录制另外一个MP3文件）。

关于这个知识点，一定要搞清楚，否则就会出现我最开始遇到的情况，以为VBR的采样率也是变化的，导致别人问我那么是不是意味着对VBR MP3解码，每一帧都要重新设置解码器的采样率，如果回答是，那mplayer等常见解码器的代码实现中，没看到对应设置，只看到了最开始解码MP3时候，只设置一次其采样率，其后解码每一帧，都是没有重新设置采样率的。而实际结果是，VBR变化的只是比特率，采样率是固定的，所以只需要在解码MP3最开始的时候设置一次即可。

### 3.3. MPEG帧的大小

前面已经解释了，MPEG帧= 帧头 + 数据。

下面来看看，MPEG的帧的大小，即帧头的大小，加上帧数据的大小。

解释MPEG帧大小之前，先要介绍个名词：Slot，槽。

MPEG帧，由一个个的Slot（槽）组成。

Layer I中，一个Slot是4个字节；

Layer II和Layer III中，一个Slot是一个字节。

所以，此处可以简单的理解为：

MPEG的Layer III中，帧是有一个个字节所组成。（是不是听起来像句废话，^\_^）

好了，知道了此处MPEG的Layer III的帧的基本单位为字节之后，我们再来看看帧的大小是多少。

首先，帧头，不用多说，都是前面提到的，固定的32比特=4字节。

其次要好好解释一下MPEG帧的音频数据的大小，可用如下公式计算：

#### 公式 3.1. Frame Data Size

Frame\_Data\_Size

= Audio\_Data\_Size + Frame\_Padding\_Size

= Frame\_Time \* Frame\_Bitrate + Frame\_Padding\_Size

= (Sample\_Number \* Time\_per\_Sample) \* Frame\_Bitrate + Frame\_Padding\_Size

= (Sample\_Number \* (1/Sample\_Rate)) \* Frame\_Bitrate + Frame\_Padding\_Size

### 公式 3.2. 帧数据大小

帧数据大小

= 音频数据大小 + 帧的填充大小

= 帧的时长 \* 帧的比特率 + 帧的填充大小

= ( 采样个数 \* 每一采样的时长 ) \* 帧的比特率 + 帧的填充大小

= ( 采样个数 \* ( 1 / 采样率 ) ) \* 帧的比特率 + 帧的填充大小

其中：

1. 帧的填充大小：
 

对于MPEG的Layer III来说，单位就是字节，不够8bit一个字节的的话，添加padding对应的bit，凑够一个字节。所以，具体padding几个bit，要看每一帧的数据的bit是否是8的倍数，如果本身是8的倍数，那么padding就是0，如果不是，根据具体情况决定补齐几个bit。
2. 采样个数：
 

对于MP3，即MPEG-1，Layer III来说，不论CBR还是VBR，每一帧采样个数都是固定的1152个。
3. 采样率：
 

对于MP3，即MPEG-1，Layer III来说，不论CBR还是VBR，对于单个MP3文件来说，也是固定的，每一帧采样率，都是一样的。采样率是多少，通过解析第一帧，即可得知所有帧的采样率。
4. 帧的比特率：
  - A. CBR：
 

每一帧都是一样的。通过解析第一帧即可得知其他所有帧的比特率。
  - B. VBR：
 

每一帧都不同，所以要针对每一帧具体解析帧头，才能得知每一帧的比特率具体是多少。

### 3.3.1. 常见MPEG帧的音频数据大小是418字节



最后来说明，

以常见的采样率为44100 Hz，比特率为128kbps的CBR的MP3来计算：

帧数据大小

= ( 采样个数 \* ( 1 / 采样率 ) ) \* 帧的比特率 + 帧的填充大小

= ( 1152 \* ( 1 / 44100 Hz ) ) \* 128kbps + 填充大小

= 3343.7 比特 + 填充大小

= 417.959字节 + 填充大小

= 418 字节

对应的MPEG帧大小为：

MPEG帧大小

= 帧头 + 帧数据

= 4 + 418

= 422字节

而对于VBR的帧的大小，就不是能这么简单计算出来的了。

因为VBR是每一帧的比特率都是变化的，所以对于每一帧的大小，都先要解析每一帧的帧头，得到每一帧的比特率，然后才可以计算出来每个帧的大小。

另外提及一点，由于舍入误差，官方的计算帧大小的方法和此稍有不同。根据ISO标准，应该以slot为单位进行计算，然后对结果取整，再乘以slot的大小。

不过，我们此处计算的是MPEG的Layer III，本身slot就是一个字节，所以计算方法是对的。

如果计算的是Layer I，一个slot是4字节，就要先以4字节为单位进行计算，然后对结果取整，再乘以slot大小，即再乘以4字节。

# 第 4 章 MP3的播放时间的计算公式及 XING和VBRI头介绍

## 4.1. CBR的MP3的播放时间(duration)计算方法

### 4.1.1. 计算CBR的MP3的播放时间的方法之一

对于计算CBR的MP3的播放时间，其是Constant Bitrate，固定的比特率，每一帧的比特率也都是固定的同样的大小，所以，相对来说，很容易想得到，用文件大小，直接除以比特率，就可以得到文件的播放时间了，即就用如下公式可以计算MP3的播放时间：

#### 公式 4.1. CBR播放时间 ( CBR Duration )

CBR Duration

$$= \text{File Size(Byte)} \times 8 \text{ bit/Byte} \div (\text{Bitrate ( K bit/s)} \times 1000 \text{ bit/Kbit})$$

CBR播放时间

$$= \text{文件大小 ( 字节)} \times 8 \text{ 比特/字节} \div (\text{比特率 千比特/秒} \times 1000 \text{ 比特/千比特})$$

其中：

#### 1. 文件大小：

严格地说，应该是MP3的文件总大小，减去MP3的Tag的大小，即

$$\text{文件大小} = \text{总的MP3文件大小} - \text{MP3的Tag大小}$$

其中，MP3的Tag，往往和MP3文件总大小相比，几乎可以忽略不计，所以，一般也可以直接用总的MP3为文件大小，直接来计算：

$$\text{文件大小} = \text{总的MP3文件大小}$$

#### 2. 比特率：

可以通过解析MP3文件的第一帧的MPEG的帧头，得到比特率的索引值，然后查比特率索引表，即可得到比特率是多少。

所以，可以看出，对于CBR的文件，可以用上面的公式，获得MP3文件大小后，再去解析第一帧的MPEG帧头，得到比特率索引值，查表得到比特率的值，然后就算出整个CBR MP3文件的播放时间。

### 4.1.2. 计算CBR的MP3的播放时间的方法之二

另外，多说一句，我原先以为，还有另外一种计算方法，即：

“总帧数乘以每一帧的时长法”

$$\text{总的时长} = \text{每一帧的时长} * \text{总的帧数}$$

就是先计算每一帧的时间长度，再计算一共有多少帧，然后将两者相乘，即可得到文件总时长。

其中：

$$= \text{每一帧的采样个数} * \text{每一采样的时长}$$

$$= \text{每一帧的采样个数} * (1 / \text{每一帧的采样频率})$$

而

总的帧数

= 文件大小 / 单个帧的大小

= 文件大小 ( 字节 ) \* 8比特/字节 / ( ( 每个帧的时长 \* 比特率 ( 千比特/秒 ) \* 1000比特/千比特 ) )

但是后来发现，上面这个计算总的帧数的方法，其实是不精确的，是把每一帧的4字节的MPEG帧头漏掉了。

因此，才会有之前的想法，认为这两种计算CBR的MP3的播放时间的计算公式是同一种：

此法，其实和上面的是同一个方法，因为上面两个等式相乘之后，即为：

总的时长

= 每一帧的时长 \* 总的帧数

= 每一帧的时长 \* [总的文件大小 ( 字节 ) \* 8比特/字节 / ( ( 每个帧的时长 \* 比特率 ( 千比特/秒 ) \* 1000比特/千比特 ) )]

= [总的文件大小 ( 字节 ) \* 8比特/字节 / [ 比特率 千比特/秒 \* 1000比特/千比特 ]]

还是和上面的方法是，是同一个公式。

而对于上面的总的帧数的计算方法，真正正确的是：

总的帧数

= 文件大小 ÷ 单个帧的大小

= 文件大小 ÷ ( 帧头 + 帧数据 )

= 文件大小 ÷ ( 固定的4字节 + 帧数据 )

= 文件大小 ( 字节 ) × 8比特/字节 ÷ ( 4 + [ 每个帧的时长 × 比特率 ( 千比特/秒 ) × 1000 ( 比特/千比特 ) ] )

方法二，即为：

CBR播放时间

= 每一帧的时长 × 总的帧数

= ( 每一帧的采样个数 × ( 1 / 每一帧的采样频率 ) ) × ( 文件大小 ( 字节 ) × 8比特/字节 ÷ ( 4 + [ 每个帧的时长 × 比特率 ( 千比特/秒 ) × 1000 ( 比特/千比特 ) ] ) )

= 文件大小 ( 字节 ) × ( 4 + ( 每一帧的采样个数 ÷ 每一帧的采样频率 ) × 比特率 ( 千比特/秒 ) × 1000 ( 比特/千比特 ) ÷ 8比特/字节 ) × ( 每一帧的采样个数 ÷ 每一帧的采样频率 )

其中：

文件大小：是总的MP3文件大小，减去MP3的Tag的大小。

### 4.1.3. 计算CBR的MP3的播放时间的两种方法的总结

也因此，最开始的原以为的第二种方法，和第一种，是同一种的想法，其实是错误的。正确的理解是，第二种方法的确是另外一个方法，而且更准确。

因为第一种方法：

**CBR播放时间**

$$= \text{文件大小 (字节)} \times 8 \text{比特/字节} \div (\text{比特率 千比特/秒} \times 1000 \text{ 比特/千比特})$$

中的比特率，只是针对每一帧的音频的数据的，而并不包括4字节的帧头，而（参见上面的解释）：“常见MPEG帧的音频数据大小是418字节”，所以，这样比较起来，至少是，没 $418+4=422$ 的这一个帧，其中只有418个是真正的音频数据，而比特率，指的只是这个帧数据的比特率，并不包括4字节的帧头，如果也包含进去了，那么则产生了误差，误差大概是 $4/(418+4)=0.95\%$ ，即大概是百分之一的误差。

即，方法一，计算出来的CBR的播放时间，大概有1%的误差。

而第二种方法：

**CBR播放时间**

$$= \text{每一帧的时长} \times \text{总的帧数}$$

$$= \text{文件大小 (字节)} \times (4 + (\text{每一帧的采样个数} \div \text{每一帧的采样频率}) \times \text{比特率 (千比特/秒)} \times 1000 (\text{比特/千比特}) \div 8 \text{比特/字节}) \times (\text{每一帧的采样个数} \div \text{每一帧的采样频率})$$

就把每一帧的4字节的帧头，也计算进去了，好处是计算出来的播放时间更加精确，坏处是，计算起来，相对第一种方法来说，比较麻烦。

而第一种方法，虽然有大概1%的误差，但是很容易计算。所以现实中，常用第一种方法。

### 4.1.3.1. 举例说明两种方法计算出结果的不同

我找了个CBR的MP3，用千千静听去查看其播放时间，推断出，其用的是第一种方法计算的。

具体推断过程如下：

（此处为了方便计算，忽略了MP3的Tag的大小，而把整个文件大小看作是MP3的所有帧的总大小）

该MPEG 1- Layer III的MP3：

文件大小=11777537 字节

采样个数：1152

采样率：44100Hz

比特率：96kbps

**【方法一】**

总时长

$$= 11777537 \times 8 \div (96 \times 1000)$$

$$= 981.46$$

**【方法二】**

总时长

$$= 11777537 \div (4 + (1152 \div 44100) \times (96 \times 1000 \div 8)) \times (1152 \div 44100)$$

$$= 11777537 \div (4 + 313.47) \times 0.02612$$

$$\begin{aligned} &\approx 11777537 \div (4 + 314) \times 0.02612 \\ &= 11777537 \div 318 \times 0.02612 \\ &= 967.48 \end{aligned}$$

即，方法二，是精确的MP3的总时长，为967.48，而方法一，计算出来的是有误差的981.46，误差为 $(981.46 - 967.48) / 967.48 = 1.34\%$ ，即大概为1%。

而千千静听查看到此MP3的播放时间为：16:20.454 = 980.454秒，

考虑到我们上面用方法一算出来的MP3的播放时间是981.46秒，是把MP3的Tag也算进去了，所以去掉Tag的大小，就是千千静听上看到的980.454秒了。

### 【总结】

因此，虽然方法二计算出来的播放时间更准确，但是实际上，为了更简便的计算，往往采用了误差大概只有1%，（一般用户也可以接受此误差），的第一种方法：

### CBR播放时间

$$= \text{文件大小 (字节)} \times 8 \text{ 比特/字节} \div (\text{比特率 千比特/秒} \times 1000 \text{ 比特/千比特})$$

## 4.2. VBR的MP3的播放时间(duration)计算方法

而对于VBR，由于每一帧的比特率都是变化的，所以计算起来就相对要复杂一些，下面就来详细介绍。

想要计算VBR的MP3的播放时间，总的来说，有两种方法：

### 4.2.1. 平均比特率法

这个方法，就是和CBR同样的思路，对于VBR的MP3来说，假如也像CBR的MP3一样，也有个类似的每一帧都是固定的某个值的比特率，那么计算整个VBR的播放时间，也就可以用上面CBR一样的公式去计算了。

由此，就有了平均比特率的概念，即，将所有帧的比特率的值相加，得到一个总的比特率的值，然后除以总的帧数，就得到了一个平均比特率，这样，使得理论上，此VBR相当于一个比特率为该平均比特率的CBR了。

不过，可以看出，需要计算平均比特率之前，要先得到每一帧的比特率的值，以及总的帧的数目，然后才可以计算平均比特率的值。

而为了得到每一帧的比特率的值，就要将整个VBR MP3文件都遍历一遍，以此找到所有的帧，并解析每一个帧的帧头，得到比特率索引值，然后查表得到比特率的值。

如此做的话，效率显然很低。因为此处只是为了计算整个VBR MP3的播放时间，却要遍历整个文件，还要解析每一帧的帧头，显得很是得不偿失。

所以，就有了更好的，效率更高的，下面要介绍的另一种方法，来计算VBR MP3的播放时间。

另外，需要提醒的是，对于平均比特率来说，往往和第一帧的比特率相差很大。因为常见的MP3音乐的开头部分，即第一帧或者前几帧，多数是一些无声的数据，或者本身包含信息量很少，比特率很低的数据。因此，其意味着，如果解码器对于VBR文件，误解为CBR文件，按照CBR所有帧的比特率都相同的逻辑，去解析第一帧，得到一个比特率，然后用此比特率来计算整个文件的播放时间的话，那么往往计算出的播放时间和实际的相差很大。这也就是后面引用中一个帖子里面遇到的情况，即，Media Player Classic播放VBR的MP3时的时间问题。

### 4.2.2. 总帧数法

总帧数法，即利用总的帧的数目，来计算VBR的播放时间。

此方法的前提，是我们前面就强调过的：

1. MP3，即MPEG-1，Layer III，不论是CBR，还是VBR，每一帧的采样个数都是固定的1152个。即每一帧，都是固定的1152个采样。
2. CBR和VBR中的固定和可变，都是指的是比特率Bitrate，而不是采样率Sample Rate。对于同一MP3文件，不论CBR还是VBR，采样率都是固定的。

了解了这两个前提后，就可以看出，对于VBR来说，虽然每一帧的比特率不同，但是每一帧的时间都是固定的，因为

每一帧的时间

= 该帧的采样个数 \* 该帧的采样率

= 1152 \* 采样率

其中：

1. 采样个数：  
MPEG-1，Layer III，即MP3，不论是CBR还是VBR，都是固定的1152
2. 采样率：  
对于单个的VBR文件，都是统一的，固定的，常见的是44100Hz。采样率可以通过解析第一帧的帧头得出采样率索引，然后查表得到采样率。

既然知道了每一帧的时间都是固定的，那么很容易就想到，如果知道VBR MP3有一共多少帧，那么就可以用 总的帧数 \* 每一帧的时间 = 总的时间长度了。

所以，剩下的事情，就是去得到VBR MP3的总的帧数。

最简单，但是效率很低的方法就是，像上面方法1一样，遍历整个VBR文件，找出一共有多少帧，对于第一帧，解析第一帧的帧头，得到采样率。

这样有了采样率和总的帧数，就可以用上面的解释的原理来计算了，对应公式就是：

#### 公式 4.2. VBR MP3总的时长 ( VBR Duration )

VBR Duration

= Total\_Frame\_Number \* Time\_Per\_Frame

= Total\_Frame\_Number \* ( Sample\_Number \* Time\_Per\_Sample )

= Total\_Frame\_Number \* ( Sample\_Number \* ( 1 / Frame\_Sample\_Rate ) )

VBR MP3总的时长

= 总的帧数 \* 单个帧的时长

= 总的帧数 \* ( 帧的采样个数 \* 每个帧的时长 )

= 总的帧数 \* ( 帧的采样个数 \* ( 1 / 帧的采样率 ) )

其中：

1. 总的帧数：  
VBR中的总的帧的数目。
2. 帧的采样个数：



对于MP3 ( MPEG1 , Layer III ) 来说, 是固定的1152个采样。

3. 帧的采样率：

通过解析第一帧，即可得知帧采样率索引，查表，即可得此采样率。

但是，可以看到，虽然此遍历整个文件以得到总的帧数的方法，但是还是显得效率不高。此处我们毕竟只是需要知道总的帧数而已，却还是要遍历文件。

对此问题，想象一下，要是有人在VBR的文件头部，单独提供了这个总的帧数，那么不就可以省去了我们再去遍历整个文件了吗？

而实际情况是，你所想到的事情，别人已经帮你实现了。^\_^。

现实中，VBR文件中，就是已经有了对应的头Header，用于存放VBR相关的信息。

这样的头信息，也就是下面将要介绍的XING和VBRI。

### 4.2.3. VBR的两种Header : XING和VBRI

VBR的帧头，记录了和VBR相关的一些信息，至少包含了我们前面介绍的，用于方便我们计算VBR的播放时长的总的帧数。

VBR MP3的帧头，主要有两种类型，XING和VBRI。

此外，VBR的头中，往往还包含了一个用于定位的TOC ( table of content ) 目录表。即用于在快进或快退的时候，通过表中的信息，可以方便地定位到对应的位置。如果没有此TOC表，需要单独去计算出对应的位置，比较麻烦。

关于它们的具体格式和含义，下面就对其进行详细解释。

#### 4.2.3.1. Xing TAG / Xing头 ( header )

此tag由XING公司推出的算法/规范，所以叫做XING。

对于大多数的VBR文件都加了此头，但并不全是。此头位于MPEG音频头后面的某个特定位置（多数是0x24）。包含了此XING头的第一个帧，其后的数据是空的，所以即使解码器没有考虑到此头，也可以正常处理此帧。对于Layer III的文件来说，比如常见的MP3，此VBR放在边信息（side information）之后。

下表是XING头的具体格式及含义：

表 4.1. XING 头的格式及含义

位置 ( 字节 )	长度 ( 字节 )	含义	示例								
0	4	4个ASCII字符的VBR头 ID, 要么是Xing, 要么是Info, 无NULL结尾 ( 普通字符串都以NULL,即\0结尾 )	'Xing'								
4	4	存放一个标志, 用于表示接下来存在哪些域/字段, 各字段逻辑或的结果 : <table border="1" style="margin-left: 20px;"> <tr> <td>0x0001</td> <td>存在总帧数 ( Frames ) 字段</td> </tr> <tr> <td>0x0002</td> <td>存在文件大小 ( Bytes ) 字段</td> </tr> <tr> <td>0x0004</td> <td>存在TOC字段</td> </tr> <tr> <td>0x0008</td> <td>存在音频质量指示字段</td> </tr> </table>	0x0001	存在总帧数 ( Frames ) 字段	0x0002	存在文件大小 ( Bytes ) 字段	0x0004	存在TOC字段	0x0008	存在音频质量指示字段	0x0007 就表示下面存在 :  总帧数 文件大小总字节数 TOC表
0x0001	存在总帧数 ( Frames ) 字段										
0x0002	存在文件大小 ( Bytes ) 字段										
0x0004	存在TOC字段										
0x0008	存在音频质量指示字段										
8	4	总帧数(Frames), 大端[可选]	7344								
8或12	4	文件总大小, 单位字节, 大端[可选]	45000								

位置 (字节)	长度 (字节)	含义	示例
8, 12, 16	100	TOC表, 大端[可选]	
8或12, 16, 108, 112, 116	4	音频质量指示, 最差0, 最好100, 大端[可选]	0

虽然知道了XING头的具体含义, 可以去根据具体的值, 解析出对应的含义了, 但是, 由于其是放在side information之后的, 所以, 要先定位, 找到XING头, 关于其位置, 用如下公式计算:

### 公式 4.3. XING头位置

#### XING头位置

= MPEG头位置 + MPEG帧头大小 + 边信息大小

= MPEG头位置 + 4字节 + 边信息大小

其中:

1. MPEG头位置:  
即通过程序去找到连续的11个bit都是1的位置, 即可同步MPEG的帧, 找到对应的MPEG头的开始处。
2. 边信息大小:  
详细信息, 后面用到此公式时会具体解释。

根据头的格式, Xing头里面必须包含ID和flag这两个段。其他字段都是可选的, 是否包含, 要看flag的值。有时候这个Xing头, CBR里面也有, 此时, 前面的ID的值就是Info, 而不是Xing了。

### 4.2.3.2. VBRI头 ( header )

据了解, 目前此头信息, 只有用Fraunhofer的编码器生成的MPEG音频文件, 才会用到此头。

其和Xing头不一样, 其放在第一个MPEG头的后面, 大小正好是32字节。其位置, 长度和示例, 都是以字节为单位。

下表是VBRI头的具体格式及含义, 单位为字节:

表 4.2. VBRI头的格式及含义

位置 (字节)	长度 (字节)	含义	示例
0	4	4个ASCII字符的VBR头ID: "VBRI" 无NULL结尾	"VBRI"
4	2	版本ID, 大端, 类型: DWORD	1
6	2	延迟, 类型: float	7344
8	2	音频质量指示	75
10	4	文件总大小, 大端, 类型: DWORD	45000
14	4	总的帧数, 大端, 类型: DWORD	7344
18	2	TOC表的表项数目, 大端, 类型: WORD	100
20	2	TOC表项的缩放因子, 大端, 类型: DWORD	1
22	2	单个TOC表项的大小, 单位字节, 最大为4, 大端, 类型: DWORD	2
24	2	帧数/表项, 大端, 类型: WORD	845

MP3的播放时间的计算公式及XING和VBRI头介绍

位置 (字节)	长度 (字节)	含义	示例
26		用于检索的TOC表, 整型值, 可以通过每个表项大小乘以表项个数得到此TOC表的总大小, 大端	

# 第 5 章 计算CBR和VBR的MP3文件的播放时间的步骤

此处只是大概总结一下，具体解析出播放时间，需要哪些步骤，可以参考相关源码：

1. 可以参考 [1]，注册登录后，可以下载源代码，自己看，就知道了。
2. 也可以去参考 [2] 中的Mplayer的VBR的patch，里面写的更加简单，也更容易看明白。

下面就解释一下，关于如何去计算MP3的文件的播放时间，的具体的逻辑和顺序：

## 5.1. 定位到MPEG的帧头的位置

由于在计算MP3播放时间之前，要先找到对应的MPEG的帧头，所以，先要找到MPEG具体在某个位置。

具体方法是，如果文件开始没有ID3 V2的头信息，那么一般MPEG的帧头位置是0，当然，具体还是要根据帧头中的同步位（sync bit），共11位去定位找到帧头。

## 5.2. 解析MPEG帧头，获取必要信息

主要是根据帧头格式，解析出MPEG的版本，MPEG的Layer，以及采样率，比特率等信息，用于后面的解码和计算播放时长。

## 5.3. 判断是VBR还是CBR，根据公式计算播放时间

### 5.3.1. 定位出Xing头的位置

根据 [公式 4.3 “XING头位置”](#) 定位到XING头所在位置。上式中：

1. MPEG头位置：  
如果没有ID3 V2这类的信息的话，那么MPEG头位置就是文件的最开始，即0的位置。
2. MPEG帧头大小：  
固定的32 bit = 4字节，所以加4。
3. 边信息大小：  
对应Layer III，根据MPEG的版本，查下表可得，单位为字节：

表 5.1. MPEG Layer III的边信息（side information）（单位：字节）

	MPEG 1	MPEG 2/2.5 (LSF)
立体声，联合立体声，双声道	32 <sup>①</sup>	17
单声道	17	9

① 我们最常见的，双声道的MP3，不论是VBR还是CBR，对应的边信息，都是32个字节。

### 5.3.2. 如果有Xing头，则是VBR，解析XING头，计算播放时间

如果对应XING头的位置有对应的“Xing”字符，那就说明是Xing头。

那么就可以解析Xing头，找到对应的我们所需要的一些值，尤其是总的帧数。

然后用之前介绍的[公式 4.2 “VBR MP3总的时长 \( VBR Duration \)”](#) 去计算VBR的播放时间。

其中：

1. **总的帧数 ( Number of Frames )**：

可以通过解析XING头，找出里面总帧数 ( Frames )，这个字段，一般都是存在的。

2. **帧的采样个数 ( Samples Per Frame )**：

根据前面解析MPEG，找到MPEG的版本，基于属于哪个Layer，然后根据[表 3.4 “MPEG帧的采样数索引表 \( 单位：个/帧 \)”](#) 查得每帧的采样个数，得到每一帧有多少个采样。

而对于MP3 ( MPEG-1, Layer III ) 此处肯定是对应的1152。

3. **采样率 ( Sampling Rate )**：

根据前面解析MPEG，找到2比特大小的采样率索引，然后根据[表 3.3 “MPEG帧的采样率索引表 \( 单位：Hz \)”](#) 找到对应的采样率。

此处，也就是找到我们前面所说的，总的帧数，加上另外两个参数：帧的采样数和帧的采样率（都是通过解析第一帧的帧头，即可算出对应的值），然后我们就可以算出VBR的MP3文件的总的播放时间长度了。

### 5.3.3. 如果没XING头，定位出VBRI头位置，找VBRI头

计算出VBRI的位置，如果该位置找到“VBRI”字符，那么说明是VBRI头。

然后解析VBRI，找到对应的总的帧数，然后和XING头算法类似，用上面的公式计算出VBR的播放时间即可。

### 5.3.4. 如果也没找到VBRI头，则是CBR，计算CBR的播放时间

如果连VBRI头也没找到，即，Xing和VBRI都没有，那么则是CBR。

然后用[公式 4.1 “CBR播放时间 \( CBR Duration \)”](#) 计算出播放时间长度。

其中：

1. **文件大小 ( File Size )**：

文件大小 = 总文件大小 - 头信息大小

这里的头信息，指的是ID3 V1或ID3 V2之类的头信息。

而总文件大小，这个不用多解释，就是整个MP3的大小。

两者单位都是字节。

2. **比特率**：

通过解析第一帧的MPEG，即可得到比特率的索引值，然后查[表 3.2 “MPEG比特率索引表 \( 单位：Kbit/s \)”](#)，即可得到此MP3的比特率大小。

然后套用上面的公式，即可算出CBR的MP3的播放时间长度了。

# 第 6 章 MP3相关的一些知识点的解释

## 6.1. MP3的文件的内容组织结构

总结起来，一般的MP3文件所包含的内容的结构如下：

表 6.1. MP3文件的内容组织结构

第二帧，格式如下：	
[ID3.....] [APE 头]	ID3 V2的头，大多数最新的MP3，都有这个头 用于APE格式的头，现在也用于MPEG
第一帧，格式如下：	
MPEG音频头 边信息 [Xing 头]	固定的4 字节 9/17/32 字节 8-120字节，如果是VBR，多数都有此Xing头，而且只有第一帧有
MPEG音频数据	
第二帧，格式如下：	
MPEG音频头 边信息 MPEG音频数据	固定的4字节 9/17/32 字节
第三帧，格式如下：	
MPEG音频头 边信息 MPEG音频数据	固定的4字节 9/17/32 字节
.....	
最后一帧，格式如下：	
MPEG音频头 边信息 MPEG音频数据	固定的4字节 9/17/32 字节
[TAG .....] 128字节的ID3 V1信息，如果没有前面的ID3 V2，多数都有这个ID3 V1的头	



### 提示

[]号内的，表示，可选，即如果有的话。

## 6.2. MP3帧的时长是26ms的来历

时长，即时间长度。

MP3，即MPEG-1，Layer III。

MPEG1，Layer III，根据表 3.4 “MPEG帧的采样数索引表（单位：个/帧）”可以查得，每一帧的采样个数，是1152，这个值是固定的。

而MPEG-1所支持的采样率，根据表 3.3 “MPEG帧的采样率索引表（单位：Hz）”可以看出有三种，44.1K，48K和32K Hz。而我们实际常见的CBR或者VBR的MP3所采用的采样率，多数都是44.1KHz=44100Hz。

所以，每一个采样的时长 =  $1/44100$  秒

因此：

每一帧的总时长

= 每一帧的采样个数 \* 每个采样的时长

=  $1152 \text{个采样} * 1/44100 \text{秒/采样}$

=  $1152 / 44100$  秒

= 0.026秒

= 26 ms ( 毫秒 )

这就是所谓的，MP3的每帧的时长，都是26ms这一说法的缘由。

## 6.3. 怎么算出来MP3的压缩比大概是1:11的

正常，所谓CD里面的歌曲，是无损格式，即原始的声音，没有经过压缩的。

相对来说，音质最高，但是文件体积太大。到底有多大，我们可以来算一下：

每秒的原始CD的音频文件的数据量

= 声道数目 \* 采样率 \* 每个采样占用多少bit

=  $2 \text{声道} * 44100 \text{Hz} * 16 \text{位}$

= 1411200 bit

而对应的MP3文件来说，即将原始的音频数据文件，经过MP3的压缩算法压缩后，数据量就小多了。以常见的双声道的，频率为44100Hz，比特率为128Kbps的MP3为例：

每秒的MP3的数据量

= 比特率 \* 1秒

= 128K bits

= 128 000 bit

因此：

MP3的压缩率

= 每秒的原始CD的音频文件的数据量 / 每秒的MP3的数据量

=  $1411200 \text{ bit} / 128 \text{ 000 bit}$

= 11.025

$\approx 11$

这就是大家常说的，MP3的压缩比大概有1:11，即在保持不错的音质前提下，将数据量减小到了原来的1/11。通俗点说，原先CD音质的，无压缩的歌曲是11M的话，那么对应的MP3就只有1M。相对很不错的压缩比，大大减小了歌曲的体积，在互联网时代，此优势得到很好的体现，因此MP3才真正流行了起来。

当然，如果其他参数同上，而MP3比特率是64Kbps，那么对应压缩比有1:22；

同理，如果是256Kbps的MP3，那么压缩比就小了，只有1:5.5左右了。

## 6.4. 怎么算出来MP3的压缩比大概是1:11的

### 6.4.1. 原因之一

实际上，如果清楚了MP3的音频帧的大小，你就会发现，这些帧头很小，对于数据帧本身和MP3文件大小的影响，可以忽略不计。

因为，MPEG的帧头，一共就32bit=4字节，相对于每一帧的音频数据，以常见的采样率为44100 Hz，比特率为128kbps的CBR的MP3来说，是418个字节，只相当于音频数据的1/100左右，所以，相对来说，对于整个文件的影响很小。

所以说，每个帧都加上这4个字节的帧头，对于数据帧本身和MP3文件大小的影响，可以忽略不计。

这只是每一帧都保留一个MPEG帧的帧头的原因之一。

### 6.4.2. 原因之二

虽然对于CBR的MP3，除了第一帧之外的每一帧的MPEG帧头，是可以去掉。但是对于VBR的MP3，每一帧的MPEG帧头，都包含了当前帧的比特率的信息，而VBR的每一帧的比特率的大小，都是不一定相同的，因此每一帧都必须要有对应的MPEG帧的帧头，来提供这些相关信息，供解码器正确解码和播放MP3。



---

## 第 7 章 后记

最后想说的是，其实很无语，对于很多东西，尤其是计算机相关的技术，网上能找到的中文资料，多数都是说的不是太清楚。其中，包括我这里要找的，如何去计算VBR的播放时间，多数情况是，不论是去百度Google一下，还是去Google百度一下，找了半天，还是没看到有几个人能说清楚。让人越发感叹道，国内的计算机方面的技术，和国外，差的的确不是一点两点。人家把规范都定好了，过了N年了，结果咱们到现在，也没几个搞清楚咋回事。。。因此，对于计算机方面的资料，尤其是涉及底层技术的，能详细解释清楚你的问题的，多数都是一些英文原版资料。所以，在此，再次，感谢一下原作者。。。

---

# 参考书目

- [1] [MPEG Audio Frame Header](#) ( 登陆该页面后, 有源码和程序供下载 )<sup>1</sup>
- [2] [MPlayer-1.0rc1 和 MPlayer-1.0rc2 的VBR patch【已修复mad播放VBR MP3播放总时长显示问题】](#)<sup>2</sup>
- [3] [MPEG AUDIO FRAME HEADER \(mp3 format\)](#)<sup>3</sup>
- [4] [MPEG Audio Layer I/II/III frame header](#)<sup>4</sup>
- [5] [MP3 Profi Info](#)<sup>5</sup>
- [6] [MP3 Tech](#)<sup>6</sup>
- [7] [Media Player Classic播放VBR的MP3时的时间问题](#)<sup>7</sup>
- [8] [什么是MP3](#)<sup>8</sup>
- [9] [Variable bitrate](#)<sup>9</sup>
- [10] [MPEG Audio Compression Basics](#)<sup>10</sup>
- [11] [ID3 tag version 2](#)<sup>11</sup>
- [12] [MP3](#)<sup>12</sup>
- [13] [Mp3tag](#)<sup>13</sup>
- [14] [CBR, VBR and bit reservoir](#)<sup>14</sup>
- [15] [Difference Between ABR and VBR](#)<sup>15</sup>

---

<sup>1</sup> <http://www.codeproject.com/KB/audio-video/mpegaudioinfo.aspx>

<sup>2</sup> [http://www.crifan.com/mplayer-10rc1\\_and\\_mplayer-10rc2\\_fixed\\_the\\_vbr\\_patch\\_mad\\_play\\_vbr\\_mp3\\_player\\_displays\\_the\\_total\\_length\\_of\\_the\\_problem/](http://www.crifan.com/mplayer-10rc1_and_mplayer-10rc2_fixed_the_vbr_patch_mad_play_vbr_mp3_player_displays_the_total_length_of_the_problem/)

<sup>3</sup> <http://www.datavoyage.com/mpgscript/mpeghdr.htm>

<sup>4</sup> [http://www.mp3-tech.org/programmer/frame\\_header.html](http://www.mp3-tech.org/programmer/frame_header.html)

<sup>5</sup> [http://www.goat.cz/index.php?path=MP3\\_MP3ProfiInfo](http://www.goat.cz/index.php?path=MP3_MP3ProfiInfo)

<sup>6</sup> <http://www.mp3-tech.org/>

<sup>7</sup> <http://xialulee.spaces.live.com/blog/cns!4ee324c8acfa82db!230.entry?wa=wsignin1.0&sa=835890451>

<sup>8</sup> <http://wenwen.soso.com/z/q140151046.htm>

<sup>9</sup> [http://en.wikipedia.org/wiki/Variable\\_bitrate](http://en.wikipedia.org/wiki/Variable_bitrate)

<sup>10</sup> <http://www.datavoyage.com/mpgscript/mpeghdr.htm>

<sup>11</sup> <http://id3.org/id3v2-00>

<sup>12</sup> <http://en.wikipedia.org/wiki/MP3>

<sup>13</sup> <http://zh.wikipedia.org/zh/Mp3tag>

<sup>14</sup> <http://www.hydrogenaudio.org/forums/lofiversion/index.php/t50345.html>

<sup>15</sup> <http://www.differencebetween.net/technology/difference-between-abr-and-vbr/>