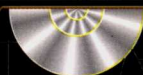


THOMSON

Premier 游戏开发丛书

含CD

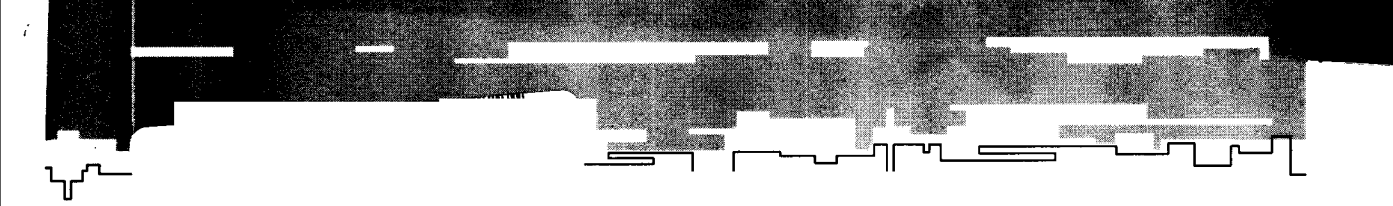


游戏音效 编程

[美] Mason McCuskey 著
朱庆生 主译

cqup
•com
•cn

重庆大学出版社

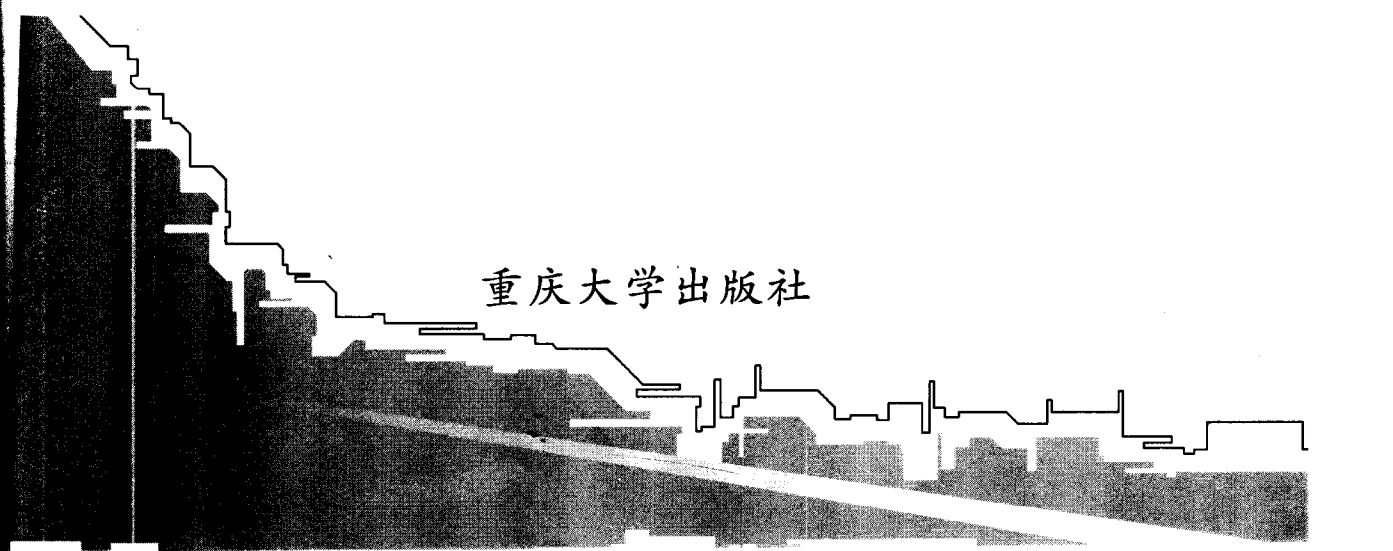


游戏音效编程

原著：[美] Mason McCuskey

主译：朱庆生

译者：王勇 陈波 葛亮



重庆大学出版社

Mason McCuskey

Beginning Game Audio Programming

ISBN: 1-59200-029-0

Copyright © 2003 by Premier Press, a division of Thomson Learning.

Original language published by Thomson Learning. All Rights reserved.

本书原版由汤姆森学习出版集团出版。版权所有,盗印必究。

Chongqing University Press is authorized by Thomson Learning to publish and distribute exclusively this simplified Chinese edition. This edition is authorized for sale in the People's Republic of China only (excluding Hong Kong, Macao SAR and Taiwan). Unauthorized export of this edition is a violation of the Copyright Act. No part of this publication may be reproduced or distributed by any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

本书中文简体字翻译版由汤姆森学习出版集团授权重庆大学出版社独家出版发行。此版本仅限在中华人民共和国境内(不包括中国香港、澳门特别行政区及中国台湾)销售。未经授权的本书出口将被视为违反版权法的行为。未经出版者预先书面许可,不得以任何方式复制或发行本书的任何部分。

981-265-254-x

版贸核渝字(2004)第34号

图书在版编目(CIP)数据

游戏音效编程/(美)麦卡斯基(McCuskey, M.)著;
朱庆生译. —重庆:重庆大学出版社, 2005. 3
(Premier 游戏开发丛书)
ISBN 7-5624-3348-8

I. 游... II. ①麦... ②朱... III. 游戏—声音质量
—程序设计 IV. TP311.5

中国版本图书馆 CIP 数据核字(2005)第 016988 号

游戏音效编程

Youxi Yinxiao Biancheng

[美]Mason McCuskey(麦卡斯基)著 朱庆生 主译

出版者:重庆大学出版社

网 址: <http://www.cqup.com.cn>

电 话: (023)65102378 65105781

出版人:张鸽盛

责任编辑:王 斌 武雨南川

责任校对:李定群

印刷者:自贡新华印刷厂印刷

发 行 者:全国新华书店经销

开 本: 787 × 960 1/16 印张: 20.5 字数: 411 千

版 次: 2005 年 5 月第 1 版 2005 年 5 月第 1 次印刷

书 号: ISBN 7-5624-3348-8

印 数: 1—3 000

定 价: 42.80 元

社 址: 重庆市沙坪坝正街 174 号重庆大学(A 区)内

邮 编: 400030

传 真: (023)65103686 65105565

版式设计: 王 斌

责任印制: 秦 梅

译者序

游戏音效编程在游戏开发中至关重要,它直接影响游戏的整体效果。本书作者从事游戏编程工作多年,在游戏音效编程方面具有丰富的实践经验。在本书中,他从音频基本知识入手,系统地介绍如何在游戏中添加声音或背景音乐等实用技术。全书分为两部分,第一部分“音频引擎基础”,重点介绍如何把 WAV 音频、MIDI 音频、MP3 音频、CD 音频和轨迹音乐等不同格式的音频文件引入游戏软件中,主要以 DirectMusic、DirectSound、DirectAudio 为开发工具,对每一种音频文件的使用方法进行详细描述,其目的是引导读者掌握播放和控制游戏音效的基本知识和技巧;第二部分“高级音频引擎”,重点介绍如何把动态音乐、3D 音乐、音频特效和环境音效等技术引入游戏软件中,主要以 DirectX Audio、OPENAL、DirectPlay 等为软件工具,对多种高级音效技术逐一进行讲述,其目的是引导读者掌握动态音乐的基本术语、音频脚本的基本知识、音频可视化技术和高级游戏音效的实用编程技巧。本书配有大量源程序对所涉及的术语和基础知识进行讲解和支撑,并提供了包含实例源代码的配套光盘。

本书结构清晰,语言简洁,深入浅出,实例丰富,可操作性强,适用于游戏开发人员及广大游戏编程爱好者,也可作为高等院校相关专业师生的参考书籍。如果你正在进行游戏编程或希望将来开发游戏软件,为游戏添加音效是必不可少的,那么,选择《游戏音效编程》必有收获。

本书由朱庆生负责组织翻译和审校,参与翻译工作的人员还有王勇、陈波、葛亮。由于译者经验和水平有限,译文难免有不妥之处,欢迎广大读者批评指正。

译者

2004 年 8 月于重庆

丛书编辑致读者的信

如果读者喜欢我,那么很可能也会喜欢游戏中的音乐和声效。过去在游戏开发过程中,音乐和声效并不是人们关注的要点,通常的做法是先罗列出所需要的声效和音乐段的清单,并确定播放质量标准,如16-bit 立体声、44 kHz 采样频率,然后雇某人去完成。这主要是因为开发者不想花费太多的时间去处理游戏音效。如在游戏中,可能会为飞机使用一些3D音频,为游戏某处添加多普勒频移(doppler frequency shifting),或者为隧道中的声音改变音调等。但现在如果还这样做,就犯了一个大错。音频技术在过去的5~7年中以不可思议的速度发展并成熟。Windows 平台上 DirectAudio 的音频处理能力、硬件供应商在3D声音建模方面的技术进步、丰富的超高频有损或无损音频压缩技术为游戏设计人员创建高质量游戏音效环境提供了极大的支持和方便。

一定要记住“声音比视频更重要”。事实上,声音可以创造紧张、兴奋、愤怒等效果。试着在没有声音的情况下玩你最喜欢的游戏,它还有吸引力吗?这并不神秘,声音对你的大脑具有直接和立即的影响,它极大程度地抚慰或者刺激你的大脑。在任何情况下,都不能忽视声音。3D图形的谜团已经解开,是时间了解声音了——学习并掌握它。

《游戏音效编程》正是为此而编写的一本书。学完本书,将了解和掌握作为一名音频编程专家所应该具备的相关知识。本书并非一本高技术性书籍,它与数字信号处理、傅立叶变换、脉冲反应等知识无关。但对于有兴趣设计音效的人来说,本书是一本具有很好参考价值的基础性入门书籍。

书中将学到怎样使用 DirectAudio 的特性,比如数字文件播放、DirectMusic 及音频脚本。此外,也将学到如何播放 MP3 文件,以及下一代格式音乐 Ogg Vorbis。

本书也将涉及怎样在应用中播放 CD、混音、时间处理等细节。另外,还涉及3D声音建模、OpenAL 使用、音频可视化及如何使用工具创建声音和音乐。

本书最后讨论了内容创建、授权,以及创建完整声效和音乐的策略。作为希望了解更多音频编程知识的游戏音频开发者、或者希望自己书写 MP3 播放程序的业余爱好者,本书都可能成为你的一本重要参考书。我确信你无法找到另一本与此主题相关的、浅显易懂而富含实例的相同书籍。作者 Mason McCuskey 花费了大量的时间和精力,使得本书更有趣、更有价值、更引人入胜,更不用提他所创建的有趣音效了。请记住,不要听我的,听你自己的。

诚挚的

A handwritten signature in black ink, reading "André LaMothe". The signature is fluid and cursive, with a long horizontal stroke at the end.

André LaMothe

Premier 出版社游戏开发丛书编辑

内容简介

开发游戏时,人们常常忽视游戏的音效。开发者往往把主要精力花费在游戏的图像和动画等方面,而忽视了背景音乐和声音效果。当他们意识到这一点时,通常为时已晚。这种做法显然是不正确的,因为好的游戏音效和音乐可以使玩家融入游戏世界,并与其情绪产生共鸣。音效的作用还不仅限于此。如果没有高超的游戏音效的映衬,再好的图像技巧也无法使游戏表现摆脱平庸,对玩家没有足够的吸引力。

以前,关于游戏音效的书很少,因为那时游戏音效设计都很简单。采用 20 世纪 90 年代的声卡,只能让计算机说一段话或播放一段简单的单声道音乐,不言而喻,其实现方法也就比较简单。今天,3D 音效、环境音效、MP3 播放以及 CD 音效等技术为游戏音效的开发提供了方便的音效技术支持(即使是设计简单游戏)。

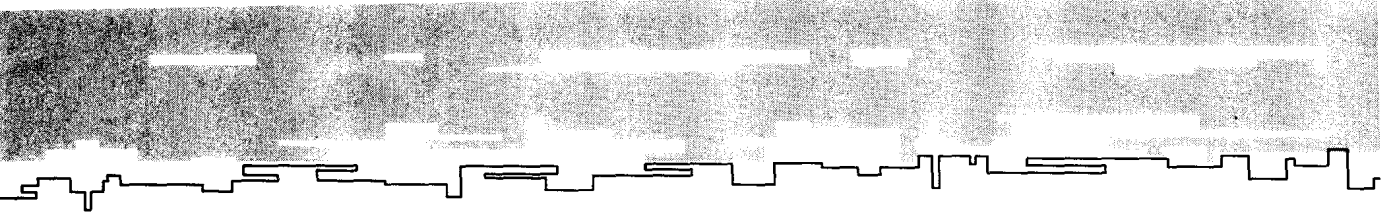
当游戏程序员为游戏设计音效时,会面临许多技术问题。本书的编写目的就是针对这些问题,试图教会读者设计游戏音效的一些基本技能。

全书分为两个部分,第一部分讲述在诸如播放音效、装载并分析 WAV 文件、以及播放不同格式音乐(如 MIDI、MP3、CD)等简单设计任务中,如何克服实施中的技术困难。第二部分讲述一些高级的音效技术。在这一部分中,读者将学习如何合成动态音乐并将其应用于游戏中;也将学习关于 3D 音乐编程、环境音效、音效脚本设计、音乐特效处理及音效可视化方面的基础知识。此外,专门有一章讲述如何使用 DirectPlay 语音软件在游戏中实现实时声音对话。

本书的绝大部分内容都基于 DirectX 及 DirectX Audio 技术,少数章节基于 OpenAL Audio 或普通 Win32 音频 API。所有代码都采用 C++ 编写,并基于标准模板库(STL)的低层数据结构(如向量、映射等)。

为了学习和理解本书的内容,必须了解 C++,虽然并不需要精通 C++;其次,还需要知道 Windows 程序是如何工作的;另外必须知道事件驱动的程序设计以及 Win32 API 的应用,包括如何书写按钮、编辑框、列表框等的 GUI(图形用户接口)代码;也必须掌握 DirectX 的基本知识,熟悉如何获取和释放不同 DirectX 组件的接口;理解 DirectX 的功能和用途以及当你需要时到哪里获得帮助。

如果具备了以上所有知识,请从第一章开始逐章依次学习本书的内容。



致谢

许多人都为本书的编辑和出版做出了贡献。首先,我想感谢丛书编辑 André LaMothe, 以及责任编辑 Jenny Davidson 和技术编辑 Heather Holland, 他们帮助我将本书写得更加一致和连贯。同时,我也要感谢 Jeremy Martin 为光盘中的示例音乐所做的贡献。当然,我还要感谢我的家人和朋友们——谢谢你们!

作者

Mason McCuskey 是一位有 8 年编程经验的软件工程师,他于 1990 年成功创立了自己的游戏公司 Spin Studios。2000 年,在游戏开发者协会举办的第二届游戏程序竞赛(Game Developer's Conference 2nd Annual Independent Games Festival)中,Spin Studios 公司入围前 9 名进入决赛。Mason 也是 Premier 出版的《DirectX 特效游戏编程》(《Special Effects Game Programming with DirectX》)一书的作者,他还为 gamedev.net 和其他 Web 网站写了许多文章。

目 录

第 1 章 音频基础知识	1
1.1 音效原理及术语	2
1.2 游戏音效发展简史	7
1.3 购买或设计音频引擎	13
1.4 本章小结	15

第 1 部分 音频引擎基础

第 2 章 DirectX Audio 介绍	19
2.1 DirectX Audio 基础	20
2.2 如何配置 DirectX Audio	23
2.3 音阶生成器的示例程序	26
2.4 本章小结	37

第 3 章 播放 WAV 文件	39
3.1 用 DirectMusic 处理声效	40
3.2 音频引擎中错误处理	42
3.3 设计音频引擎	46
3.4 播放音频的其他工具	56
3.5 本章小结	57

第 4 章 加载 WAV 文件	59
4.1 剖析 WAV 格式	60

4.2 加载 WAV 的其他方法	67
4.3 本章小结	72
第5章 声效控制技巧	73
5.1 连续播放音频	74
5.2 音频实例程序	75
5.3 循环播放音频	78
5.4 用 DirectX Audio 控制音量	79
5.5 用混音器 API 控制音量	84
5.6 本章小结	88
第6章 播放 MIDI	89
6.1 MIDI 工作原理	90
6.2 类的变更	91
6.3 加载和播放 MIDI	93
6.4 改变音乐节拍	95
6.5 检测音乐节拍	99
6.6 用 Windows MCI 播放 MIDI	104
6.7 本章小结	104
第7章 播放 MP3 和 WMA	105
7.1 播放 MP3	106
7.2 播放 WMA	115
7.3 本章小结	116
第8章 播放 Ogg Vorbis	117
8.1 Ogg Vorbis 类库结构	119
8.2 安装 Vorbis	120
8.3 使用 Vorbis	121
8.4 本章小结	125
第9章 播放 CD 音频	127
9.1 用 MCI 播放 CD 音频	129

9.2 CD 播放器示例程序	142
9.3 本章小结	143

第 10 章 播放轨迹音乐	145
10.1 轨迹音乐基础	147
10.2 用 MikMod 播放轨迹音乐	148
10.3 本章小结	150

第 2 部分 高级音频引擎功能

第 11 章 动态音乐	153
11.1 动态音乐基础	154
11.2 轨迹音乐与 DirectMusic 术语对照	162
11.3 动态音乐编程	166
11.4 本章小结	172

第 12 章 脚本和特效	173
12.1 音频脚本	174
12.2 音频特效	191
12.3 本章小结	202

第 13 章 DirectX Audio 中的 3D 音效	203
13.1 3D 音效的概念	204
13.2 3D 音效编程	210
13.3 外层空间的示例程序	214
13.4 本章小结	218

第 14 章 OpenAL 中的 3D 音效	219
14.1 OpenAL 与 DirectX Audio 中 3D 音效的区别	220
14.2 将 OpenAL 集成到音频引擎	224
14.3 用 OpenAL 的外层空间示例程序	230
14.4 本章小结	231

第 15 章 3D 音效中的高级技术	233
15.1 环境建模	234

15.2 环境音效扩展技术 EAX	243
15.3 杜比数字编码技术	244
15.4 本章小结	244
第 16 章 DirectPlay 语音	245
16.1 DirectPlay 语音基础	246
16.2 DirectPlay 速成	249
16.3 使用向导 CNetConnectWizard	254
16.4 DirectPlay 语音编程	257
16.5 作为 3D 音效缓冲区的语音	261
16.6 本章小结	261
第 17 章 音频可视化	263
17.1 离散傅立叶变换	264
17.2 示例程序	269
17.3 图形渲染	273
17.4 快速傅立叶变换	274
17.5 本章小结	274
结束语	275
附录 A 创建动态音乐	277
A.1 安装 DirectMusic Producer	278
A.2 创建静态音乐	280
A.3 动态音乐	294
A.4 小结	302
附录 B 创建游戏音效的体会和技巧	305
B.1 音乐	306
B.2 声效	307
B.3 语音	309
B.4 小结	310

第 1 章

音频基础知识

这一章是游戏音效的入门知识。为了确保后续章节的学习效果,本章暂时没有具体示例程序,而只有大量的概念和术语。

在这一章中,将学到数字音频的核心知识——计算机如何听和唱;将了解关于视频游戏音频的发展简史;还将知道使用授权软件或免费下载软件来设计游戏引擎的方法。

为了避免读者花了几周时间学习本书之后抱怨:“原来我可以下载一个引擎!”,在本章中将先给出一张游戏引擎表。

1.1 音效原理及术语

所有的游戏声音都是以一个单字节流结束。因此,本书所讨论的内容以及游戏音效本身,都是基于构造大量的一维字节数组的。听起来似乎很简单,事实也确实如此。然而,这个简单的原理有时却被大量的术语搞得含糊不清。考虑到这点,这里先介绍一些基本的音效术语。

1.1.1 采样频率及采样质量

采样频率的概念贯穿音频程序设计的始终。要明白这个概念,需要首先了解数字音频是如何录制并最终存储到计算机硬盘上的(参见图 1.1)。

假设你正在制作一个动作游戏并需要一个令人毛骨悚然的尖叫声音。你找到朋友或同事,说服他坐到麦克风前,然后点击音频软件的“录音”按钮并暗示他开始尖叫。

当你点击“录音”按钮后,音频软件启动声音硬件设备开始捕获声音。通常,硬件系统捕获声音的能力取决于声卡上一块称为模拟-数字转换器的 ADC 芯片的性能。ADC 芯片时刻观察着从麦克风传过来的电流以确定声音有多大,然后,它把这个数字传给声卡设备驱动程序,并最终透过 Windows 到达音频程序,最后音频程序再将这个数字写入内存或硬盘。这个数据片断称作一个采样。(通常整段录音或采样集合也被称作一个采样,所以,必须根据上下文确定其实际所指)。

术语采样频率是指每秒钟采样的次数。采样频率以赫兹(Hz)为单位,这个单位以秒为计量基准。即 8 000 Hz 表示每秒发生 8 000 次。与字节和千字节的关系类似,频率单位也有赫兹和千赫兹(kHz),如 22 kHz 表示每秒发生 22 000 次,8 000 Hz 等于 8 kHz。

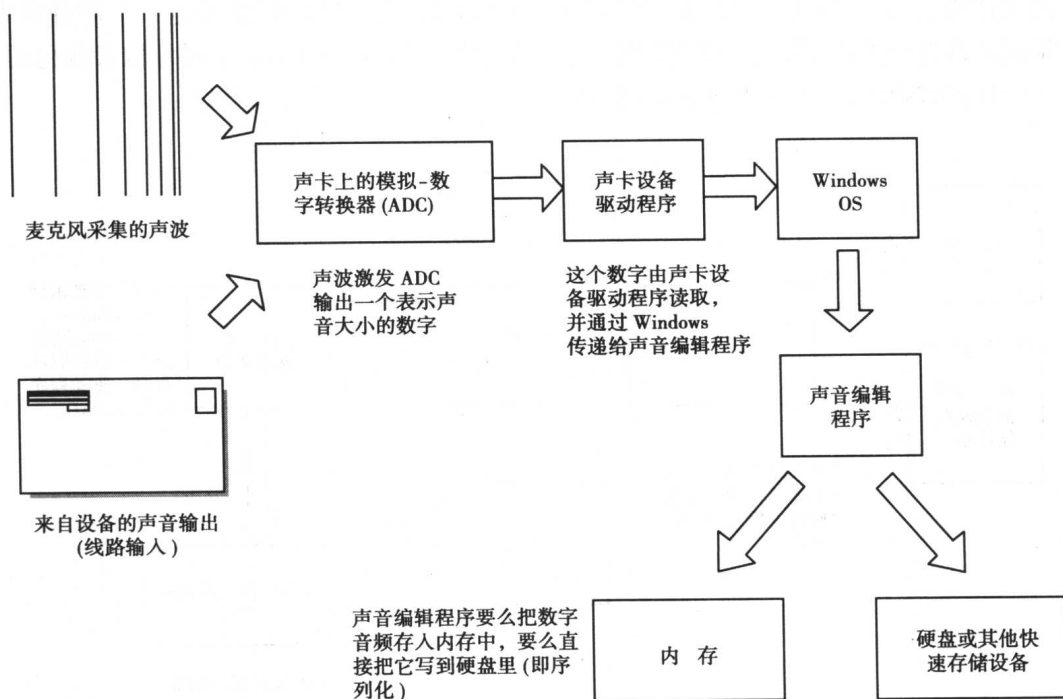


图 1.1 计算机如何录制声音

从某种意义上理解, 采样频率对于音频编程如同分辨率对于图像编程。我们知道: 图像的像素越多, 所表示的视频图像质量就越高; 同样, 声音的采样频率越高, 所表现的声音品质也越好。一般声卡都支持以 11 025 Hz、22 050 Hz 或 44 100 Hz 等标准采样频率来录制声音。CD 音频的采样

频率是 44.1 kHz, 如果要录制一个 CD 音乐节目, 则计算机将以每秒 44 100 次的速度来测量空气中的声音, 这将涉及庞大的数据量。

当播放音乐时, 计算机读取这些数字并回送给声卡 (见图 1.2)。

现在介绍影响音效的另一个术语——采样质量。采样质量 (也称每个样本的比特数——bits per sample, 简称比特 bits) 描绘每个声音采样的精确程度。采样质量通常为 8 比特或 16 比特, 8 比特可以表达 256 种不同程度的音高, 16 比特却可以表达 65 536 种音高。显然, 16 比特比 8 比特声音质量高, 它能够更加细致地表现声音。极端情况下, 如果仅有 1 比

音频剪辑

播放音频剪辑 1.1 和 1.2, 区分 44.1 kHz 与 8 000 Hz 采样频率的差别。

特的采样质量,则只能表达“喧闹”和“安静”两种状态,这无法较好地表达一个声音样本。如果说采样频率类似于图像分辨率,则声音样本的比特数就类似于表示图像像素颜色的位数——比特数越大,所表示的声音就越准确。

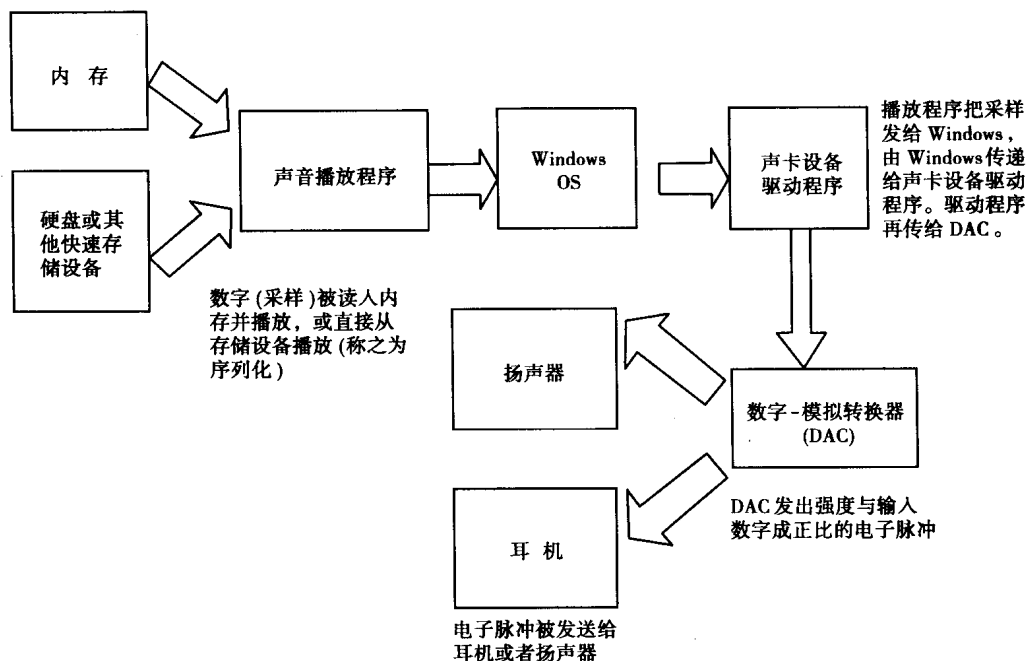


图 1.2 你的计算机如何播放声音

为什么是 44 100 Hz?

为什么 CD 的采样频率选择 44 100 Hz? 为了回答这个问题,我们需回顾音频技术的发展历史。在 CD 出现前,数字音频被存储在视频录像带上——这在当时的一种最好选择。录像带被设计成用于存储和播放每秒 60 帧的视频数据,由于每个视频帧有 245 条扫描线,而每条扫描线上都有红、绿、蓝 3 个样本。所以,总共有 $245 \times 3 \times 60 = 44\ 100$ 个样本。

1.1.2 立体声音

到目前为止,我们只谈到一个声道的声音,也称单声道声音,简称单声道。

为了获取立体声,需要在不同的位置放置两个麦克风(一个在声源的左边,另一个在右边)。但在实际中,很少采用这种方式录制立体声音,而是使用混音器或混音软件

来控制录音。混音器接受单声道的声音样本后,控制从指定声道的扬声器输出声音样本。

音频剪辑

播放音频剪辑 1.3 和 1.4, 区分 8 位采样与 16 位采样的细微差别。

一个关于 CD 的故事

我的岳父(一个音乐家)曾告诉我说,CD 之所以选择存储 74 分钟音乐,是因为人们希望能不间断地欣赏贝多芬的《第九交响曲》。工程师通过计算发现《第九交响曲》的长度为 74 分钟,便定义了可以存储如此容量数据的光盘的物理指标。一个数百年前逝世的古典音乐家却能够影响今天无处不在的音频技术,真是非常有趣。

不论通过什么方式,最终是要获得有两个声道的立体声音样本。CD 音质即是指采样频率为 44 100 Hz 的 16 位立体声。对于 CD 音质的声音,每个声道每秒钟有 44 100 个样本,每个样本用 16 位(2 个字节)来记录,立体声有两个声道,所以,每秒钟的 CD 声音有 $44\ 100 \text{ 样本/声道} \times 2 \text{ 字节/样本} \times 2 \text{ 声道} = 176\ 400 \text{ 字节}$ (约 176 kB)。这也说明了“为什么一张存储容量为 650 MB 的 CD,最多存放 74 分钟 CD 音乐”的原因。

1.1.3 声音格式、压缩及编码

当然,如果我们仅能用每秒 176 kB 的庞大数据量来存储 CD 音乐的话,数字世界无疑将会令人失望。一段 3 分钟长的歌曲将有 31 680 kB(约 30 MB)的信息量!如果真是这样,唱片商会十分高兴,因为他们不用担心人们在线交换音乐。即使是在高速宽带连接下,为了 3 分钟音乐而传输 30 MB 数据量实在令人难以接受。

聪明的人类想出了多种音频压缩算法来处理这个问题。这些算法用更紧凑的方式表达同样的数据,从而可以用更少的字节存储相同的信息。举个简单的例子,假设有一个单声道 WAV 文件,里面存有两秒钟的寂静之声。以前,如果使用 CD 音质,这些声音将会用 88 200 字节的相同的“0”值来表示。然而,可用如下方式压缩它们:“后面 88 200 个字节都是 0”,而不需要重复地把每个“0”都写一遍。这是一种最常用的行程编码(RLE)算法的工作原理。

RLE 是一种非常简单的压缩算法,详见图 1.3。另一种更好的流行音频压缩算法是 MPEG Layer 3,用 MPEG Layer 3 压缩的声音文件以 MP3 为扩展名,其他压缩算法几乎都已