

<<数字音频规范与程序设计>>

图书基本信息

书名：<<数字音频规范与程序设计>>

13位ISBN编号：9787508498195

10位ISBN编号：7508498194

出版时间：2012-6-1

出版时间：中国水利水电出版社

作者：曹强

页数：458

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<数字音频规范与程序设计>>

内容概要

《数字音频规范与程序设计：基于Visual C++开发》是作者根据多年的工作经验而总结出来的有关Windows平台下数字音频规范与程序设计的一手资料，由浅入深地介绍了当今多数主流音频相关的技术规范与编程实现，适合于对数字音频感兴趣、准备或正在从事数字音频相关工作和开发的用户。

本书共分11章，涵盖了Windows平台下数字音频相关规范与编程的许多方面，详细介绍了Windows系统下高层与底层的多媒体音频接口（API）；WAVE音频的全面解析、播放与录音；MIDI音乐的全面解析、播放与录制；USB MIDI驱动设计；MP3的格式与编解码；乐音的基频检测；音频混音器原理与编程；低延迟ASIO音频驱动开发；普通音频插件与VST（i）插件的设计；游戏开发中经常使用的DirectSound播放与录制等。

《数字音频规范与程序设计：基于Visual C++开发》最大的特色是不但针对多媒体音频应用程序的开发做了较全面的介绍，可以满足大部分常规音频编程用户的需求，而且对音频驱动的开发也做了较多的切入。

其中一部分是作者近几年独自研究的成果，对于有这方面需求的读者来说都是很宝贵的参考资料。

本书结构清晰，逻辑严密，内容具体且涉及面广泛，不但是从事多媒体（音频）开发与应用的广大开发人员的技术指导书，同时也可作为各高等院校相关专业、非相关专业师生重要的参考读物。

本书中每章都提供了一个或多个经典的编程实例程序，附带在光盘中赠送，光盘中提供了所有章节中实例程序的Visual C++源代码。

<<数字音频规范与程序设计>>

书籍目录

推荐序一 推荐序二 前言 第1章Windows音频体系与VisualC++开发基础 导读 1.1认识声音与音频
1.2Windows音频体系层次结构 1.2.1用户模式与内核模式 1.2.2关于声音的延迟问题及解决方法
1.3VisualC++开发基础 1.3.1区分C、C++和VisualC++三者的关系 1.3.2事件驱动与消息处理 1.3.3开发环境概述 1.3.4MFC概述 1.3.5利用向导创建一个简单的对话框程序 第2章MCI与MMAPI多媒体编程接口
2.1 MCI简介 2.1.1 MCI逻辑结构 2.1.2 MCI接口、函数与命令 2.1.3 MCI消息与宏定义 2.2 MCI编程步骤
2.3 MCI编程实例 2.3.1播放CD 2.3.2 WAVE播放与录音 2.3.3基于MCIWnd的多媒体播放器 2.4 MMAPIS简介
2.4.1四种多媒体文件I/O 2.4.2波形音频处理 2.4.3 MIDI的流缓冲区与非流缓冲区处理 2.4.4多媒体定时器
第3章WAVE格式及播放与录制 3.1认识WAVE 3.2 WAV文件结构 3.3采样点与采样帧的区别 3.4 WAVE中常用的一些块
3.4.1格式块 3.4.2数据块 3.4.3提示块 3.4.4播放列表块 3.4.5关联数据块 3.4.6采样块
3.5多媒体波形音频函数 3.6波形音频处理的回调机制 3.7 WAVE播放 3.7.1 WAVE播放基本流程 3.7.2双缓冲播放机制
3.7.3关于死锁的问题 3.8 WAVE录音 3.9编程实例 3.9.1 WAV文件解析及分离与合并 3.9.2 WAVE播放器
3.9.3 WAVE录音机 第4章MIDI相关标准与程序设计 第5章数字音乐合成 第6章音频混音器编程
第7章音频插件设计 第8章基音的时频域检测 第9章ASIO音频驱动设计 第10章MP3格式与编解码
第11章DirectSound开发与应用 附录 参考文献

<<数字音频规范与程序设计>>

章节摘录

版权页：插图：（1）MIDI解码。

音源驱动根据不同的MIDI信息类型进入对应的处理模块。

比如MIDI发音信息，需要解析其通道、音高与力度，然后进入声音处理单元。

对于控制信息，需要解析其控制号，然后进入相应的控制单元。

（2）声音预处理。

波表合成不是简单的样本回放，它还需要一些重要音色参数的控制，包括音高、音量、包络处理、滤波处理等。

音源驱动得到开音相关信息后，将根据当前的音色号、音高和力度到音色库中查找其对应的样本（一个或多个），找到每个样本对应的音色参数，然后再根据这些参数进行一定的运算，并将部分运算结果传递给DSP做进一步处理。

（3）DSP处理。

DSP根据音源驱动提供的样本地址取出样本数据，并根据音源驱动计算出来的部分音色参数调整波形数据的回放速率和放大系数等。

（4）数模转换。

从DSP出来的数字信息就是最终的波形数据，经过数模（D/A）转换器将数字信号转换为模拟信号后通过扬声器，于是我们最终听到了MIDI音符的发声。

从图5.9中可以看出，波表合成主要由如下几个功能模块组成：逻辑控制模块；发音控制模块；发音处理模块。

逻辑控制模块主要是指音源驱动的MIDI解码与样本选择（寻址）。

发音控制模块主要是指音源驱动对音色参数的解析与运算（比如包络运算、LFO运算、滤波运算等），并将部分计算结果传递给DSP进行处理。

发音处理模块主要是根据音源驱动提供的参数（如样本存储地址、Loop点信息、包络信息等）进行波形数据的回放，当然，回放不是简单的播放，回放时需要进行回放速率的控制、去除噪声、数字放大等处理。

与FM相比，波表（Wave Table）合成确实使电子音乐的发展取得了历史性的进步，虽然它的核心思想并不复杂，但通过对真实乐器的数字化录音与回放，使MIDI音乐听起来更接近自然、更具真实感。

当然，波表合成也有它的缺点，虽然通过音色参数的一些调整可以使原声发生一些变化，但相比于FM波表合成较难产生新的音色。

5.4.2波表合成的关键技术 首先我们来看两个问题，第一个问题是，是不是每个音色都应该像前面所举的例子（见图5.8）一样，每个音高（键位）都需要一个对应的采样样本呢？

答案是否定的，因为使用频率变换技术可以用已有的样本模拟出其相邻音高的音。

如图5.10所示，从c2到B6共有4个八度，但只用了5个采样样本（a.wav为G2采样样本，b.wav为A3采样样本，c.wav为G4采样样本，d.wav为D5采样样本，e.wav为E6采样样本），其他键位的音高都是根据这5个样本变换出来的，比如C2的发音是通过a.wav变换出来的，C4的发音是通过b.wav变换出来的，86的发音是通过e.wav变换出来的。

也就是说，一个或多个样本将整个音高划分了N个区域，每个区域内的其他键位都通过该样本变换发声。

假设当前制作的音色为原声钢琴，那么只需要录制真实钢琴5个键位的样本就可以了，所以，在波表合成器中，并不是每个音色都需要为每个音高采集样本。

事实上，很少有音色为每个键位都分配一个样本，因为这样的话，一个音色占用的样本容量就太大了，当然，整个音高区域只使用一个样本也是不好的，因为频率变换也有精度的问题，关于频率变换的相关信息请参考5.4.3节。

<<数字音频规范与程序设计>>

编辑推荐

《数字音频规范与程序设计:基于Visual C++开发》结构清晰,逻辑严密,内容具体且涉及面广泛,不但是从事多媒体(音频)开发与应用的广大开发人员的技术指导书,同时也可作为各高等院校相关专业、非相关专业师生重要的参考读物。

<<数字音频规范与程序设计>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>