

[19] 中华人民共和国国家知识产权局



[12] 发明专利申请公开说明书

[21] 申请号 200510067376.5

[51] Int. Cl.

H04N 5/91 (2006.01)

H04N 5/917 (2006.01)

G11B 20/12 (2006.01)

G10L 19/00 (2006.01)

[43] 公开日 2006 年 4 月 19 日

[11] 公开号 CN 1761308A

[22] 申请日 2005.4.14

[21] 申请号 200510067376.5

[30] 优先权

[32] 2004.10.14 [33] US [31] 10/966,443

[32] 2004.4.14 [33] US [31] 60/562,671

[32] 2004.6.18 [33] US [31] 60/580,995

[71] 申请人 微软公司

地址 美国华盛顿州

[72] 发明人 S·斯尔维拉 J·D·约翰斯顿

N·苏姆普地 W-G·陈

C·梅瑟 S·斯米尔诺夫

[74] 专利代理机构 上海专利商标事务所有限公司

代理人 沈昭坤

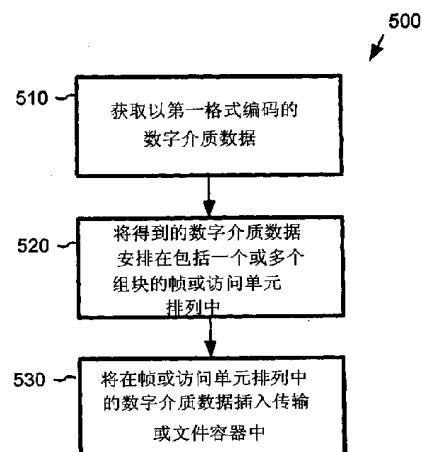
权利要求书 4 页 说明书 18 页 附图 7 页

[54] 发明名称

数字介质通用基本流

[57] 摘要

所述技术和工具包括用于将给定格式的数字介质数据(例如音频、视频、静止图像、和/或文本等等)映射成对在诸如数字视频盘(DVD)的光盘上编码数据有用的传输或文件容器格式的技术和工具。可使用数字介质通用基本流来把数字介质流(例如音频流、视频流或图像)映射成任何任意传输或文件容器(包括光盘格式)和其它传输,诸如传播流、无线传输等等。对流中数字介质的任意给定帧进行解码所需的信息可在每个经编码帧中携带。数字介质通用基本流的实现将介质流的数据安排在帧中,这些帧具有一个或多个组块。



1. 在数字介质系统中，一种将第一格式的数字介质数据映射成传输格式的方法，其特征在于，所述方法包括：

    获取以所述第一格式编码的数字介质数据；

    将所述获得的数字介质数据安排在帧排列中，所述数字介质数据的帧排列具有一尺寸并包括数字介质数据组块和元数据组块，所述帧排列可操作使数字视频盘解码器能访问和解码所述数字介质数据组块；以及

    以所述传输格式将所述数字介质数据的帧排列插入数字介质数据流。

2. 如权利要求 1 所述的方法，其特征在于，所述数字介质数据是音频，且所述传输格式用于在计算机可读数据存储光盘上存储音频数据。

3. 如权利要求 1 所述的方法，其特征在于，所述第一格式是 Windows 介质音频格式而第二格式是 DVD-A 经压缩音频格式。

4. 如权利要求 1 所述的方法，其特征在于，所述第一格式是 Windows 介质音频格式而第二格式是 DVD 音频记录格式。

5. 如权利要求 1 所述的方法，其特征在于，所述元数据组块包括指数元数据尺寸的信息。

6. 如权利要求 5 所述的方法，其特征在于，所述元数据组块包括指示元数据类型的信息。

7. 如权利要求 1 所述的方法，其特征在于，所述帧排列还包括循环冗余码校验组块。

8. 如权利要求 1 所述的方法，其特征在于，所述帧排列还包括同步组块，所述同步组块包括用于验证有效同步模式的长度字段。

9. 如权利要求 1 所述的方法，其特征在于，所述帧排列还包括格式标头组块，所述格式标头组块包括流属性。

10. 如权利要求 1 所述的方法，其特征在于，所述帧排列还包括内容描述符元数据。

11. 如权利要求 1 所述的方法，其特征在于，所述尺寸是固定尺寸。

12. 如权利要求 1 所述的方法，其特征在于，所述尺寸是可变尺寸。

13. 如权利要求 1 所述的方法，其特征在于，所述第一格式是 Windows 介质

音频格式而第二格式是 MPEG-2 程序流格式。

14. 一种具有存储其上的计算机可读指令的计算机可读介质，所述指令用于使数字介质处理器执行权利要求 1 所述的方法。

15. 在一数字信号处理器中，一种把音频数据映射成用于在计算机可读数据存储光盘上存储音频数据的格式的方法，其特征在于，所述方法包括：

    获取音频数据；

    将所述获得的音频数据转换成具有固定尺寸的音频数据访问单元，所述音频数据访问单元包括音频数据组块、同步组块、元数据组块、以及循环冗余码校验组块；以及

    将所述音频数据访问单元以一种格式插入音频数据流，所述格式为用于在计算机可读数据存储光盘上存储音频数据的格式。

16. 在数字介质系统中，一种把音频数据解码成用于在计算机可读数据存储光盘上存储音频数据的格式的方法，其特征在于，所述方法包括：

    获取以用于在计算机可读数据存储光盘上存储音频数据的格式进行编码的数据，所述帧排列中获得的音频数据具有固定尺寸并包括音频数据组块和元数据组块，所述帧排列包括从中间格式换码的音频数据；以及

    解码所述获得的音频数据。

17. 如权利要求 16 所述的方法，其特征在于，所述中间格式是 Windows 介质音频格式，且所述用于在计算机可读数据存储光盘上存储音频数据的格式是 DVD 格式。

18. 在数字介质系统中，有一种把数字介质数据编码为用于映射到传输容器中通用基本流的方法，其特征在于，所述方法包括：

    获取根据选定数字介质编/解码器编码的数字介质流；

    将所述获得的数字介质流安排在具有帧排列的基本流中，其中帧包括多个语法元素，包括至少一元数据元素、一同步模式元素和表示与下一接近帧的同步模式的距离的一长度元素；以及

    将所述基本流插入所述传输容器。

19. 一种对根据权利要求 18 的方法编码的数字介质数据进行解码的方法，其特征在于，所述方法包括：

    将所述基本流从所述传输容器中分离出来；

    解析所述基本流以标识所述同步模式和长度的第一次出现；

---

解析所述基本流以标识所述同步模式在由所述长度标注距离上的第二次出现；以及

从所述同步模式的经标识的出现来标识所述基本流的帧。

20. 如权利要求 18 所述的方法，其特征在于，所述语法元素还包括多个可任选的组块组件，每个组块组件具有表示所述组块组件类型的语法元素，所述同步模式和长度语法元素定义所述帧的范围，而不管是包括还是略去了任何特定类型组块组件的帧。

21. 如权利要求 20 所述的方法，其特征在于，所述组块组件语法元素类型的编码方案包括用于所述基本流定义后来扩展的转义码。

22. 如权利要求 18 所述的方法，其特征在于，所述帧排列中另一帧的语法元素包括替代所述同步块的块结束组块组件，用以表示这种帧的结束。

23. 在数字介质系统中，有一种把数字介质数据编码为用于映射到传输容器中通用基本流的方法，其特征在于，所述方法包括：

获取根据选定数字介质编/解码器经编码的数字介质流；

将所述获得的数字介质流安排在具有帧排列的基本流中，其中帧包括多个语法元素，包括至少表示所述选定数字介质编/解码器的一编/解码器属性组块元素；以及

将所述基本流插入所述传输容器。

24. 如权利要求 23 所述方法，其特征在于，表示所述选定数字介质编/解码器的所述编/解码器属性组块元素包括选定数字介质编/解码器的版本信息。

25. 一种把至少一原始格式的数字介质数据映射成存储、发送、或传递传输容器格式的方法，其特征在于，所述方法包括：

获取所述至少一原始格式的数据，以及扫描、解析、传送、解码或执行所述至少一原始格式所需的任何侧信息、元数据信息、或辅助信息；

把作为组块组件序列的所述数据安排到基本流中，所述组块组件来自在所述组块组件的预定组块类型标头中编码的可任选包括的一组组块类型，其中该安排根据所述数字介质的格式、存储、传输、传递或呈现所需或所要求的来把可任选包括的组块类型的组块组件包括为编码成比特流或从中略去，所述组块序列由至少一个包含原始介质数据的组块组件，和至少一个包含所述侧信息、元数据信息或辅助信息的组块组件组成；以及

将所述基本流的组块组合成数据包的一顺序集合或传输容器格式的序列流，

用于所述数字介质的自包含存储、传输、传递或呈现。

---

## 数字介质通用基本流

### 相关申请

5 本申请声明对以下美国临时专利申请的权利：申请号为 60/562,671 题为“Mapping of Audio Elementary Stream”（“音频基本流的映射”）于 2004 年 4 月 14 日提交的美国临时专利申请，以及申请号为 60/580,995 题为“Digital Media Universal Elementary Stream”（“数字介质通用基本流”）于 2004 年 6 月 18 日提交的美国临时专利申请，两个申请都在此引入作为参考。

10

### 技术领域

本发明一般涉及数字介质（例如音频、射频、和/或静态图像等等）的编码和解码。

15

### 背景技术

引入了光盘、数字视频盘、便携式数字介质播放器、数字无线网络、以及因特网上的音频和视频传送之后，数字音频和视频已变得常见了。工程师使用各种技术以有效处理数字音频和视频而仍保持数字音频或视频的质量。

20

数字音频信息被处理成表示音频信息的一系列数字。例如，单个数字可表示音频采样，它是特定时间上的幅度值（即音量）。若干因素影响音频信息的质量，包括采样深度、采样率、以及信道模式。

25

采样深度（或精度）指示用以表示采样的数字范围。可能用于采样的值越多质量越高，因为数字可捕捉幅度上更多微弱的变化。例如，8-比特采样有 256 个可能值，而 16-比特采样则具有 65,536 个可能值。24-比特采样可非常精细地捕捉正常的音量变化，且也可捕捉特别高的音量。

采样率（通常测量为每秒的采样数）也影响质量。采样率越高质量越高，因为可表示更大的带宽。某些普通的采样率为 8,000、11,025、22,050、32,000、44,100、48,000 和 96,000 采样/秒。

30

单声和立体声是音频的两种普通信道模式。在单声模式中，音频信息在一个信道中展现。在立体声模式中，音频信息通常在标为左右信道的两个信道中展现。

通常也使用诸如 5.1 信道、7.1 信道、或者 9.1 信道环绕声的其它带有多个信道的模式。高质量音频信息的成本是高比特率的。高质量音频信息消耗大量的计算机存储器和传输能力。

许多计算机和计算机网络缺乏用以处理原始数字音频或视频的存储器或资源。  
5 编码（也称为编码技术或比特率压缩）通过把信息转换成较低比特率，降低了存储和传送音频或视频信息的成本。编码可以是无损的（其中质量不受损害）或有损的（其中解析质量受损害-尽管可能感觉音频质量并未受损害-但比特率的降低相比无损编码而言是更引入注目的）。解码（也称为解压缩）从经编码形式中提取原始信息的重建版本。

10 响应于对数字介质数据的有效编码和解码的需求，已开发了许多音频和视频编码器/解码器系统（“codec—多媒体数字信号编解码器”）。例如，参看图 1，音频编码器 100 取输入音频数据 110，并使用一个或多个编码模块将其编码以产生经编码音频输出数据 120。在图 1 中，使用分析模块 130、频率变换器模块 140、质量缩减器(有损编码)模块 150、以及无损编码器模块 160 以产生经编码音频数据 120。

15 控制器 170 协调并控制编码过程。

现有的音频 codec 包括微软公司的 Windows 介质音频（“WMA”）codec。某些其它 codec 系统由运动图象专家组（“MPEG”）、音频层 3（“MP3”）标准、MPEG-2 高级音频编码[“AAC”]标准或由其它诸如 Dolby（提供 AC-2 和 AC-3 标准）的商业供应商提供或指定。

20 不同的编码系统使用特定的基本比特流，用于包括在能够携带一个以上基本比特流的复合流中。这种复合流也称为传输流。通常，传输流在基本流上提出了诸如缓冲器尺寸限制的某些限制，并需要在基本流中包括某些信息以便于解码。通常基本流包括一访问单元以便于基本流的同步和准确解码，并提供在传输流中对不同基本流的标识。

25 例如，AC-3 标准的修订版 A 描述了由同步帧序列组成的基本流。每个同步帧包含同步信息标头、比特流信息标头、六个经编码音频数据块、以及错误校验字段。同步信息标头包含用于在比特流中获取和维持同步的信息。该同步信息包括同步字、循环冗余码校验字、采样率信息以及帧尺寸信息。比特流信息包括编码模式信息（例如信道的数量和类型）、时间码信息、以及其它参数。

30 AAC 标准描述了音频数据传输流（ADTS）帧，该帧包括固定标头、可变标头、任选的错误校验字、以及原始数据块。固定标头包含不随帧变化的信息（例如

同步字、采样率信息、信道配置信息等等)，但仍然每帧重复以允许对比特流的随机访问。可变标头包含随帧变化的数据（例如帧长度信息、缓冲器充实度信息、原始数据块数量等等）。错误校验块包括用于循环冗余码校验的变量 `crc_check`。

现有的传输流包括 MPEG-2 系统或传输流。MPEG-2 传输流可包括多个基本流，  
5 诸如一个或多个 AC-3 流。在 MPEG-2 传输流中，由至少 `stream_type` 变量、  
`stream_id` 变量以及音频描述符来标识 AC-3 基本流。音频描述符包括用于单个 AC-3  
流的信息，诸如比特流、信道数量、采样率、以及描述性文本字段。

对于有关 `codec` 系统的更多信息，参见相应标准或技术出版物。

## 10 发明内容

总而言之，详细说明涉及用于诸如音频流的数字介质编码和解码的各种技术和工具。所述技术和工具包括用于将给定格式的数字介质数据（例如音频、视频、静止图像、和/或文本等等）映射成对在诸如数字视频盘（DVD）的光盘上编码数据有用的传输或文件容器格式的技术和工具。

15 本说明书详述了可由这些技术和工具使用的数字介质通用基本流，以把数字介质流映射成任何任意的传输或文件容器，包括不仅光盘格式而且其它诸如广播流、无线传输等等的传输。所述数字介质通用基本流携带在该流中解码流所需的信息。此外，可在每个经编码帧中携带解码流中数字介质的任意给定帧的信息。

20 数字介质通用基本流包括称为组块的流组件。数字介质通用基本流的实现将介质流的数据安排成帧，而这些帧具有一个或多个组块。组块包括组块标头（包括组块类型标识符）以及组块数据，尽管对于某些组块类型而言并不显现组块数据，诸如组块的所有信息都在组块标头中展现的组块类型（例如块的结束组块）。在某些实现中，组块被定义为组块标头和直到下一组块标头开始的所有随后信息。

25 在一实现中，数字介质通用基本流使用组块来加入有效的编码模式，包括带有同步模式和长度字段的同步组块。某些实现在“肯定签到”基础上使用可选元素来编码流。在一实现中，批组块的结束或者可使用同步模式/长度字段来标记流帧的结束。此外，在某些流的帧中，可略去同步模式/长度组块和块的结束组块。因而，同步模式/长度组块以及块的结束组块也是该流的任选元素。

30 在一实现中，帧可携带定义介质流及其特征的称为流属性组块的信息。相应地，基本流的基本形式可简单地由指定 `codec` 属性的流属性组块的单一实例，以及介质有效载荷组块流组成。该基本形式对于低等待延时或低比特率的应用程序是有

用的，诸如语音或其它实时介质流应用程序。

数字介质通用基本流还包括扩展机制，该机制使流的定义扩展能编码最近定义的 codec 或组块类型，而无需破坏对于现有解码器属性的兼容性。通用基本流定义是可扩展的，因为使用先前不具有语义含义的组块类型码可定义新的组块类型，且包含这种新定义组块类型的通用基本流通过通用基本流的现有的或继承的解码器保持可解析。这些新定义的组块可以是“提供长度的”（其中组块的长度在组块的语法元素中进行编码）或“长度预定义的”（其中长度在组块类型编码中隐含）。然后可由现有继承解码器的解析器“丢弃”或略去新定义的组块，不会丢失比特流解析或扫描。

10

#### 附图说明

图 1 是根据现有技术音频编码器系统的框图。

图 2 是适当计算环境的框图。

图 3 是通用音频编码器系统的框图。

15

图 4 是通用音频解码器系统的框图。

图 5 是显示使用包括一个或多个组块的帧或访问单元排列，来把第一格式的数字机制数据映射成传输或文件容器的技术的流程图。

图 6 是显示用于解码帧或访问单元排列中数字介质数据的技术的流程图，该帧或访问单元排列包括从传输或文件容器中获取的一个或多个组块。

20

图 7 示出了把 WMA Pro 音频基本流映射成 DVD-A CA 格式的示例性映射。

图 8 示出了把 WMA Pro 音频基本流映射成 DVD-AR 格式的示例性映射。

图 9 示出了对用于映射成任意容器的通用基本流的定义。

#### 具体实施方式

所述诸实施例涉及用于数字介质编码和解码的技术和工具，尤其涉及使用可被映射成任意传输或文件容器的数字介质通用基本流的编解码器。所述技术和工具包括这样的技术和工具：用于将给定格式的音频数据映射成对在诸如数字视频盘（DVD）的光盘和其它传输或文件容器上编码音频数据有用的格式。在某些实现中，数字音频数据被安排为适于后来以 DVD 格式翻译和存储的中间格式。该中间格式可以是例如 Windows 介质音频（WMA）格式，更具体地则可以是如下所述作为通用基本流的 WMA 格式表示。DVD 格式可以是例如 DVD 音频录音（DVD-AR）

格式或 DVD 压缩音频 (DVD-A CA) 格式。尽管示出了这些技术对音频流的特定应用，还可以使用这些技术来编码/解码其它形式的数字介质，包括但不限于视频、静止图像、文本、超文本、以及多媒体等等。

可组合或独立地使用各种各样的技术和工具。不同实施例实现一种或多种所述技术 5 和工具。

### I. 计算环境

所述通用基本流和传输映射实施例可在其中执行数字介质和音频信号处理的各种装置的任一种上实现，包括：计算机、数字介质播放机、传输和接收装置、便携式介质播放机、音频会议、Web 介质流应用等等。通用基本流和传输映射可以 10 硬件电路（例如 ASIC、FDGA 等的电路）实现，也可以计算机或其它计算环境中执行的数字介质或音频处理软件（在中央处理单元（CPU）或数字信号处理器、音频卡等等上执行）实现，如图 1 所示。

图 2 示出了其中可实现所述实施例的适当计算环境 200 的一般示例。计算环境 200 并非旨在暗示对本发明使用范围或功能的任何限制，因为本发明可在多种多样的通用或专用计算环境中实现。 15

参照图 2，计算环境 200 包括至少一个处理单元 210 和存储器 220。在图 2 中最基本配置 230 包括在虚线内。处理单元 210 执行计算机可执行指令并可以是真实或虚拟处理器。在多处理系统中，多处理单元执行计算机可执行指令以增加处理功率。存储器 220 可以是易失性存储器（例如寄存器、高速缓存、RAM）、非易失性存储器（例如 ROM、EEPROM、闪存等）、或者是两者的某些组合。存储器 220 存储实现音频编码器或解码器的软件 280。 20

计算环境可具有附加特征。例如，计算环境 200 包括存储器 240、一个或多个输入装置 250、一个或多个输出装置 260、以及一个或多个通信链接 270。诸如总线、控制器或网络的相互连接机制（未示出）将计算环境 200 的组件相互连接起来。 25 通常，操作系统软件（未示出）提供在计算环境 200 中执行的其它软件的操作环境，并协调计算环境 200 的组件的行动。

存储器 240 可以是可移动或不可移动的，并包括磁盘、磁带或磁卡、CD-ROM、CD-RW、DVD、或任何可用于存储信息并可在计算环境 200 中访问的其它介质。存储器存储实现音频编码器或解码器的软件 280 的指令。 30

输入装置 250 可以是诸如键盘、鼠标、笔、或跟踪球、语音输入装置、扫描装置的触摸输入装置，或向计算环境 200 提供输入的另一装置。对于音频，输入装

置 250 可以是接受模拟或数字形式音频输入的声卡或类似装置，或者向计算环境提供音频采样的 CD-ROM 或 CD-RW。输出装置 260 可以是显示器、打印机、扬声器、CD 刻录机、或可从计算机环境 200 提供输出的另一装置。

通信连接 270 使能经通信介质与另一计算实体的通信。通信介质传送诸如计算机可执行指令、压缩音频或视频信息、或数据信号中的其它数据（例如已调制数据信号）的信息。已调制数据信号是具有以这种在信号中编码信息的方式来设置或改变其一个或多个特征的信号。作为示例，而非限制，通信介质包括以电学、光学、RF、红外、声学、和其它载波实现的有线和无线技术。

本发明可在计算机可读介质的一般上下文中进行描述。计算机可读介质是可在计算环境中访问的任何可用介质。作为示例而非限制，对于计算环境 200，计算机可读介质包括存储器 220、存储 240、通信介质、以及以上的任意组合。

本发明可在诸如包括在程序模块中、目标真实或虚拟处理器上计算环境中执行的计算机可执行指令的一般上下文中进行描述。一般而言，程序模块包括执行特定任务或实现特定抽象数据结构的例程、程序、库、对象、类、组件、数据结构等等。在各个实施例中程序模块的功能可在程序模块之间组合或拆分。程序模块的计算机可执行指令可在本地或分布式计算环境中执行。

## II. 通用音频编码器和解码器

在某些实现中，数字视频数据被安排为适于后来映射成传输或文件容器的中间格式。音频数据可通过音频编码器安排成这种中间格式，且随后由音频解码器进行解码。

图 3 是通用音频编码器 300 的框图，而图 4 是通用音频解码器 400 的框图。编码器和解码器中模块之间的所示关系指示编码器和解码器中信息的主要流动；为了简便起见未示出其它关系。取决于实现和所需的压缩类型，编码器或解码器的模块可添加、略去、拆分成多个模块、组合成其它模块、和/或以类似模块进行替换。

### A. 音频编码器

参照图 3，示例性音频编码器 300 包括选择器 308、多信道预处理器 310、分割器/铺砌配置器 320、频率变换器 330、感觉模拟器 340、加权器 342、多信道变换器 340、量化器 360、熵编码器 370、控制器 380、以及比特流多路复用器 [“MUX”]390。

编码器 300 接收在某些采样深度和采样率上脉冲编码调制（PCM）格式的输入音频采样 305 时间序列。编码器 300 压缩音频采样 305 并多路传输由各种编码器

300 模块产生的信息以用诸如微软 Windows 介质音频[“WMA”]格式来输出比特流 395。

选择器 308 选择用于音频采样 305 的编码模式（无损或有损模式）。无损编码模式通常用于高质量（以及高比特率）压缩。有损编码模式包括诸如加权器 342 和量化器 360 的组件，并通常用于可调质量（以及可调比特率）压缩。选择器 308 上的选择判决取决于用户输入或其它标准。

对于多信道音频数据的有损编码，可任选地多信道预处理器 310 重新排列时间域音频采样 305。多信道预处理器 310 可向 MUX 390 发送诸如用于多信道后处理的指令的侧信息。

分割器/铺砌配置器 320 把音频输入采样 305 的帧分割成具有时变尺寸和窗口成形功能的子帧块（即窗口）。子帧块的尺寸和窗口取决于帧中瞬时信号的检测、编码模式以及其他因素。当编码器 300 使用有损编码时，尺寸可变的窗口允许瞬时清晰度可变。分割器/铺砌配置器 320 向频率变换器 330 输出经分割的数据块，并向 MUX 390 输出诸如块尺寸的侧信息。分割器/铺砌配置器 320 可在每个信道基础上分割多信道音频的帧。

频率变换器 330 接收音频采样，并将它们转换成频率领域中的数据。频率变换器 330 向加权器 342 输出频率系数数据块，并向 MUX 390 输出诸如块尺寸的侧信息。频率变换器 330 向感觉模拟器 340 输出频率系数和侧信息。

感觉模拟器 340 模拟人类听觉系统的属性，以改进对一给定比特率重建音频信号的感觉质量。一般而言，感觉模拟器 340 根据一听觉模型处理音频数据，然后向量化基带加权器 342 提供可用以产生音频数据的加权因子的信息。感觉模拟器 340 使用各种听觉模型的任一种，并向加权器 342 传递激励模式信息或其它信息。

加权器 342 基于从感觉模拟器 340 接收的信息产生用于量化矩阵的加权系数，并将该加权系数应用到从频率变换器 330 接收的数据中。量化矩阵的加权系数包括音频数据中多个量化基带的每一个的权重。量化基带加权器 342 向信道加权器 344 输出加权系数数据块，并向 MUX 390 输出诸如加权因子集的侧信息。可压缩加权因子集可用于更有效的表示。

信道加权器 344 基于从感觉模拟器 340 接收的信以及本地重建信号的质量而产生信道的信道特定权重因子（是标量）。信道加权器 344 向多信道变换器 350 输出加权系数数据块，并向 MUX 390 输出诸如信道权重因子集的侧信息。

对于多信道音频数据，由信道加权器 344 所产生噪声频谱成型的频率系数数

据的多个信道常常是互相关联的，因而多信道变换器 355 可应用多信道变换。多信道变换器 350 产生提供给 MUX 390 的侧信息，它指示例如所使用的多信道变换以及多信道变换分割部分。

量化器 360 量化多信道变换器 350 的输出，产生提供给熵编码器 370 的经量化系数数据以及提供给 MUX 390 的包括量化步长大小的侧信息。

熵编码器 370 无损地压缩从量化器 360 接收的经量化系数数据。熵编码器 370 可计算用于编码音频信息的比特数，并将该信息传送给速率/质量控制器 380。

控制器 380 与量化器 360 一起工作以调整编码器 300 输出的比特率和/或质量。控制器 380 接收来自编码器 300 其它模块的信息，并处理收到的信息以确定给定当前条件下的所需量化因子。控制器 380 向量化器 360 输出量化因子，目的是满足质量和/或比特率限制。

MUX 390 多路复用从音频编码器 300 的其它模块接收的侧信息，以及从熵编码器 370 接收的熵经编码数据。MUX 390 可包括存储要由编码器 300 输出的比特流 395 的虚拟缓冲器。缓冲器的当前充实度和其它特征可由控制器 380 使用以调节质量和/或比特率。

#### B. 视频解码器

参照图 4，相应的音频解码器 400 包括比特流多路分解器[“DEMUX”]410、一个或多个熵解码器 420、铺砌配置解码器 430、反向多信道变换器 440、反向量化器/加权器 450、反向频率变换器 460、重叠器/相加器 470、以及多信道后处理器 480。解码器 400 比编码器 300 稍简单一些，因为解码器 400 不包括用于速率/质量控制或感觉模拟的模块。

解码器 400 接收 WMA 格式或另一格式的经压缩音频信息的比特流 405。比特流 405 包括经熵编码的数据以及解码器从中重建音频采样 495 的侧信息。

DEMUX 410 解析比特流 405 中的信息并将信息发送给解码器 400 的模块。DEMUX 410 包括一个或多个缓冲器，以补偿因为音频复杂性的波动、网络不稳定性和/或其它因素引起的比特率上的变化。

一个或多个熵解码器 420 无损地解压缩从 DEMUX 410 接收的熵编码。通常，熵解码器 420 应用在编码器 300 中所使用熵编码技术的相反技术。为了简单，熵解码器模块在图 4 中示出，尽管不同的熵解码器可用于有损和无损的编码模式甚至在其中使用了。还有，为了简便，图 4 未示出模式选择逻辑。当解码以有损编码模式压缩的数据时，熵解码器 420 产生经量化频率系数数据。

铺砌配置解码器 430 接收并在必要时解码信息，该信息指示来自 DEMUX 410 的帧的铺砌模式。然后铺砌配置解码器 430 向解码器 400 的各个其它模块传递铺砌模式信息。

反向多信道变换器 440 接收来自熵解码器 420 的经量化频率系数数据，以及 5 来自分割配置解码器 430 的分割模式信息、来自 DEMUX 410 的指示例如所用多信道变换和经变换铺砌部分的侧信息。使用该信息，反向多信道变换器 440 必要时解压缩该变换矩阵，并有选择性地和灵活地将一个或多个反向多信道变换应用到音频数据中。

反向量化器/加权器 450 接收来自 DEMUX 410 的铺砌和信道量化因子、以及 10 量化矩阵，并接收来自反向多信道变换器 440 的经量化频率系数数据。该反向量化器/加权器 450 必要时解压缩收到的量化因子/矩阵信息，然后执行反向量化和加权。

反向频率变换器 460 接收由反向量化器/加权器 450 产生的频率系数数据输出、以及来自 DEMUX 410 的侧信息、来自分割配置解码器 430 的分割模式信息。反向频率变换器 460 把在编码器中使用的频率变换和输出块的相反应用到重叠器/相加器 15 470 中。

除了接收来自分割配置解码器 430 的分割模式信息，重叠器/相加器 470 还接收来自反向频率变换器 460 的经解码信息。必要时重叠器/相加器 470 叠加并累加音频数据，并使帧或其它用不同模式编码的音频数据序列交错。

多信道后处理器 480 可选地将重叠器/相加器 470 输出的时域音频采样重新排列成矩阵。多信道后处理器选择性地将视频数据重新排列成矩阵，以创建回放的仿真通道、执行诸如扬声器之间信道空间旋转的特定效果、向下折叠信道用于在较少的扬声器上回放、或用于任何其它目的。对于比特流受控后处理，后处理变换矩阵随着时间而变化，并在比特流 405 中发信号或被包括在比特流 405 中。

对于更多有关 WMA 音频编码器和解码器的信息，参见专利申请号为 10/642, 25 550 题为“Multi-channel Audio Encoding and Decoding”（“多信道音频编码和解码”）发表为美国专利申请发表号 2004-0049379 于 2003 年 8 月 15 日提交的美国专利；以及专利申请号为 10/642,551 题为“Quantization and Inverse Quantization for Audio”（“音频的量化和反向量化”）发表为美国专利申请发表号 2004-0044527 于 2003 年 8 月 15 日提交的美国专利，两个专利都在此引入作为参考。

### 30 III. 音频基本流映射中的创新

所述技术和工具包括这样的技术和工具，用于把给定中间格式（诸如如下所

述的通用基本流格式) 的音频基本流映射成适于在光盘(诸如 DVD)上存储和回放的传输或其它文件容器格式。说明书和附图显示并描述了比特流格式和语义, 以及用于在格式之间进行映射的技术。

在此描述的实现中, 数字介质通用基本流使用称为组块的流组件来编码流。  
5 例如, 数字介质通用基本流的实现将介质流的数据安排成帧, 这些帧具有一个或多个类型的一个或多个组块, 诸如同步组块、格式标头/流属性组块、包括经压缩音频数据(例如 WMA Pro 音频数据)的音频数据组块、元数据组块、循环冗余码校验组块、时间标记组块、块结束组块、和/或某些其它类型的现有组块或在将来定义的组块。组块包括组块标头(可包括例如一字节的组块类型语法元素)和组块数据,  
10 尽管对于某些组块类型而言并不显现组块数据, 诸如组块的所有信息都在组块标头中展现的组块类型(例如块的结束组块)。在某些实现中, 组块被定义为组块标头和直到下一组块标头开始的所有信息(例如组块数据)。

例如, 图 5 示出了使用包括一个或多个组块的帧或访问单元排列, 把第一格式的数字机制数据映射成传输或文件容器的技术 500。在 510, 获取以第一格式编码的数字介质数据。在 520, 得到的数字介质数据被安排在包括一个或多个组块的帧或访问单元排列中。然后, 在 530, 将在帧或访问单元排列中的数字介质数据插入传输或文件容器中。  
15

图 6 显示用于解码帧或访问单元排列中数字介质数据的技术 600, 该帧或访问单元排列包括从传输或文件容器中获取的一个或多个组块。在 610, 从传输或文件容器中获取在包括一个或多个组块的帧排列中的音频数据。然后, 在 620, 解码获得的音频数据。  
20

在一实现中, 通用基本流格式被映射成 DVD-AR 格式。在另一实现中, 通用基本流格式被映射成 DVD-CA 区域格式。在又一实现中, 通用基本流格式被映射成任一传输或文件容器。在这样的实现中, 通用基本流格式被视为中间格式, 因为  
25 所述技术和工具可将此格式中的数据转换或映射成随后适于在光盘上存储的格式。

在某些实现中, 通用音频基本流是 Windows 介质音频(WMA)格式的变体。对于有关 WMA 格式的更多信息, 参见申请号为 60/488,508 题为“Lossless Audio Encoding and Decoding Tools and Techniques”(无损音频编码和解码工具和技术)于 2003 年 7 月 18 日提交的美国临时专利, 以及申请号为 60/488,727 题为“Audio  
30 Encoding and Decoding Tools and Techniques”(音频编码和解码工具和技术)于 2003 年 7 月 18 日提交的美国临时专利, 两个专利在此引入作为参考。

一般而言，数字信息可表示为一系列数据对象（诸如访问单元、组块或帧）以便于处理和存储数字信息。例如，数字音频或视频文件可表示为一系列包含数字音频或视频采样的数据对象。

当一系列数据对象表示数字信息时，如果数据对象尺寸相同则处理该系列得以简化。<sup>5</sup> 例如，假设相同尺寸的音频访问单元存储在数据结构中。使用该序列中访问单元的序数并已知序列中访问单元的尺寸，可根据该数据结构开始处的一偏移量来访问特定访问单元。

在某些实现中，<sup>10</sup> 诸如图 3 所示编码器 300 的音频编码器以诸如通用基本流格式的中间格式编码音频数据。然后可使用音频数据映射器或转换器来把中间格式的流映射成适于在光盘上存储的格式（诸如具有固定尺寸访问单元的格式）。然后诸如图 4 所示的解码器 400 的一个或多个音频解码器可解码经编码音频数据。

例如，第一格式的音频数据（例如 WMA 格式）被映射成第二格式（例如 DVD-AR 或 DVD-CA 格式）。首先，获取以第一格式编码的音频数据。<sup>15</sup> 在第一格式中，获得的音频数据被安排在具有固定尺寸或最大容许尺寸（例如当映射成 DVD-AR 格式时为 2011 字节，或某些其它最大尺寸）的帧中。该帧可包括组块，包括同步组块、格式标头/流属性组块、包括经压缩 WMA Pro 音频数据组块、元数据组块、循环冗余码校验组块、块结束组块、和/或某些其它类型的现有组块或在将来定义的组块。该排列使解码器（诸如数字音频/视频解码器）能访问并解码音频数据。<sup>20</sup> 然后将该音频数据排列以第二格式插入音频数据流中。第二格式是用于在计算机可读光学数据存储盘（例如 DVD）上存储音频数据的格式。

同步组块可包括同步模式和用于校验特定同步模式是否有效的长度字段。基本流帧的结束或者可用块结束组块来标记。此外，在基本流的基本形式中可略去诸如在瞬时应用程序中可能有用的同步组块和块结束组块（或可能其它类型组块）。

某些实现中特定组块类型的细节提供如下。

#### <sup>25</sup> IV. 将通用基本流映射成 DVD 音频格式的实现

以下示例详述了 WMA Pro 经编码音频流在 DVD-AR 和 DVD-A CA 区域上的通用基本流格式表示的映射。在此例中，该映射符合在 WMA Pro 已被接受为可选编/解码器时 DVD-CA 区域的要求，也符合在 WMA Pro 被包括为可选编/解码器时 DVD-AR 规范的要求。

<sup>30</sup> 图 7 示出了把 WMA Pro 流映射到 DVD-A CA 区域的映射。图 8 示出了把 WMA Pro 流映射成 DVD-AR 中音频对象（AOB）的映射。在这些图所示的示例中，在

访问单元或 WMA Pro 帧中携带有解码给定 WMA Pro 帧所需的信息。在图 4 和 5 中，包括 10 字节数据的流属性标头，对于给定流而言是固定的。可在例如 WMA Pro 帧或访问单元中携带流属性信息。或者，可在 CA 区域 CA 管理器的流属性标头中、或者 DVD-AR PS 的包标头或所有标头中携带流属性信息。

5 如图 4 和 5 所示的特定比特流元素如下所示：

流属性：定义介质流及其特征。流属性标头包含有大量的对给定流固定的数据。有关流属性的更多细节如下在表格 1 中提供：

比特位置	字段名称	字段描述
0-2	VersNum	WMA 比特流的版本号
3-6	BPS	经解码音频采样的比特深度(Q 索引)
7-10	cChan	音频信道数量
11-15	SampRt	经解码音频的采样率
16-31	CMap	信道映射
32-47	EncOpt	编码器选项结构
48-50	Profile Support	描述该流属于(M1,M2,M3)的编码简介的字段
51-54	Bit-Rate	经编码流的比特率（单位为 Kbps）
55-79	Reserved	保留位-设定为 0

表格 1. 流属性

组块类型：单字节组块标头。在此例中，组块类型字段在每类数据组块之前。  
10 组块类型字段携带了对随后数据组块的描述。

同步模式：在此例中有两个字节的同步模式，使解析器能够找到 WMA Pro 帧的开始。组块类型被嵌入同步模式的第一字节中。

长度字段：在此例中，长度字段指示先前同步编码开始处的偏移。与长度字段相组合的同步模式提供足够唯一的信息组合以防止仿真。当读取器遇到一同步模式时，它向前解析到下一同步模式，并校验在第二同步模式中指定的长度与它已解析的字节长度相对应，以便从第一同步模式抵达第二同步模式。如果这得到验证，解析器遇到了有效同步模式并可开始解码。或者，解码器可通过推理用它发现的第一同步模式开始解码，而不是等待下一同步模式。这样，解码器可在解析和校验下一同步模式之前执行某些采样的回放。

20 元数据：携带有关元数据类型和尺寸的信息。在此例中，元数据组块包括：指示元数据类型的 1 字节；指示组块尺寸字节数 N 的 1 字节 (>256 字节的元数据作为具有相同 ID 的多个组块进行传输)；N 字节组块；以及当没有其它元数据时用于 ID 标记的编码器输出零字节。

内容描述符元数据：在此例中，元数据组块提供用于传送有关音频流内容的

基本描述性信息的低比特率信道。内容描述符元数据为 32 比特长。该字段是可选的，且如果必要可重复（例如每 3 秒 1 次）以节约带宽。更多有关内容描述符元数据的细节如下在表格 2 中提供：

比特位置	字段名称	字段描述
0	Start	当此比特被设定时，它标记元数据的开始。
1-2	Type	本字段标识当前元数据字符串的内容。值为： Bit1 Bit2      字符串描述 0        0      标头 0        1      艺术家 1        0      专辑 1        1      未定义(自由文本)
3-7	Reserved	应设定为 0。
8-15	Byte0	元数据的第一字节
16-23	Byte1	元数据的第二字节
24-31	Byte2	元数据的第三字节

表格 2. 内容描述符元数据

5 真正的内容描述符字符串由接收器从包含在元数据中的字节流进行组装。流中的每个字节表示一个 UTF-8 字符。如果元数据字符串在块结束之前结束，则用 0x00 填充该元数据。字符串的开始和结束由“Type”字段中的转换隐含。因此，发送器在发送内容描述符元数据时循环所有的四个类型 - 即使一个或多个字符串是空的。

10 CRC（循环冗余码校验）:CRC 涵盖了从先前 CRC 开始或包括先前同步模式的一切，取更接近的一个但不包括 CRC 本身。

呈现时间标记：尽管未在图 4 和 5 中示出，呈现时间标记携带了时间标记信息以在必要的任何时候与视频流同步。在此例中，它被指定为 6 个字节以支持 100 纳秒的粒度。例如，为了在 DVD-AR 规范中提供呈现时间标记，携带它的适当位置将是在包标头中。

## 15 V. 另一通用基本流定义

图 9 示出了通用基本流的另一定义，它可被用作在上例中映射成 DVD 音频格式的 WMA 音频流中间格式。更广泛地，在此例中定义的通用基本流可用来把躯体各种各样的数字介质流映射成任意传输或文件容器。

20 在此例中所述的通用基本流中，数字介质被编码成数字介质的离散帧序列（例如 WMA 音频帧）。通用基本流以携带解码来自帧本身的任何给定数字介质帧所需所有信息的方式来编码数字介质流。

以下是对如图 9 所示流帧中标头组件的描述。

组块类型：在此例中，组块类型是在每类数据组块之前的单字节组块标头。组块类型字段携带了对随后数据组块的描述。该基本流定义定义了众多的组块类型，它包括了使基本流定义能用附加的、后来定义的组块类型来补充或扩展的转义机制。新定义的组块可以是“提供长度的”（其中组块的长度在组块的语法元素中进行编码）或“长度预定义的”（其中长度在组块类型码中隐含）。然后可由现有继承解码器的解析器“丢弃”或略去新定义的组块，不会丢失比特流解析或扫描。组块类型后面的逻辑及其使用在下一章节中详述。

同步模型：是两个字节的同步模式，使解析器能够找到基本流帧的开始。组块类型被置于同步模式的第一字节。在此例中使用的确切模式如下详述。

10 长度字段：在此例中，长度字段指示先前同步编码开始处的偏移。与长度字段相组合的同步模式提供足够唯一的信息组合以防止仿真。当读取器遇到一同步模式时，它解析到随后的长度字段、解析到下一接近的同步模式、并验证在第二同步模式中指定的长度与它已解析的从第一同步模式抵达第二同步模式所遇到的字节长度相对应。如果的确如此，解析器遇到了有效同步模式并可开始解码。对于某些帧诸如第比特率情形，可由编码器略去同步模式和长度字段。然而，编码器应当一起略去它们。

15 呈现时间标记：在此例中，呈现时间标记携带了时间标记信息以在必要的任何时候与视频流同步。在所示基本流定义实现中，它被指定为 6 个字节以支持 100 纳秒的粒度。然而，该字段在指定时间标记字段长度的组块尺寸字段之后。

20 在某些实现中，呈现时间标记字段可由文件容器携带，例如微软高级系统格式（ASF）或 MPEG-2 程序流（PS）文件容器。呈现时间标记字段被包括于在此所述的基本流定义实现中，以显示在最基本状态中流可携带解码音频流并使其与视频流同步所需的所有信息。

25 流属性：定义介质流及其特征。有关流属性的更多细节如下在此例中提供。流属性标头仅需在内部数据不随流改变时在文件开始处可用。

在某些实现中，流属性字段由文件容器携带，例如 ASF 或 MPEG-2 PS 文件容器。流属性字段被包括于在此所述的基本流定义实现中，以显示在最基本状态中流可携带解码音频流所需的所有信息。如果它被包括在基本流中，该字段在指定流属性数据长度的组块尺寸字段之后。

30 以上的表格 1 显示了用 WMA Pro 编/解码器进行编码的流的流属性。类似的流属性标头可对每一个编/解码器定义。

音频数据有效载荷：在此例中，音频数据有效载荷携带经压缩数字介质数据，诸如经压缩 Windows 介质音频帧数据。可以用数字介质流而不是经压缩音频的方式使用基本流，在经压缩音频情形中数据有效载荷是这种流的经压缩数字介质数据。

5 元数据：该字段携带有关元数据类型和尺寸的信息。可携带的元数据类型包括内容描述符、折叠、DRC 等等。可如下进行元数据的结构化。

在此例中，每个元数据组块具有：

-指示元数据类型的 1 字节

-指示组块尺寸字节数 N 的 1 字节 (>256 字节的元数据作为具有相同 ID 的多  
10 个组块进行传输)

-N 字节组块

CRC：在此例中，CRC 涵盖了从先前 CRC 后或在该 CRC 处开始并包括先前同步模式的一切，取决于哪一个更接近，直到但不包括 CRC 本身。

15 EOB：在此例中，EOB（块结束）组块被用以标记给定块或帧的结束。如果同步组块显现，不需 EOB 来结束先前的块或帧。类似地，如果 EOB 展现，同步组块不需要定义下一块或帧的开始。对于低速率流，如果不考虑初试和起动则不必携带任一组块。

#### A.组块类型

在此例中，组块 ID（组块类型）辨别在通用基本流中携带的数据类型。它足够灵活能够表示所有不同的编/解码器类型及其相关联编/解码数据，包括流属性及任何元数据，同时允许基本流扩展以携带音频、视频、或其它数据类型。后来添加的组块类型可使用 LENGTH\_PROVIDED 或者 LENGTH\_PREDEFINED 类以指示其长度，它使现有基本流解码器的解析器能跳过这些后来定义的解码器未经编程以解码的组块。  
20

25 在此所述的基本流定义的实现中，使用单字节组块类型字段来表示和辨别所有的编/解码数据。在所示实现中，有三类组块如表格 3 所示。

组块范围	标记类型
0x00 到 0x92	LENGTH_PROVIDED
0x93 到 0xBF	LENGTH_AND_MEANING_PREDEFINED
0xC0 到 0xFF	LENGTH_PREDEFINED
0x3F	转义码 (对于附加编/解码器)
0x7F	转义码 (对于附加流属性)

表格 3. 用于组块类的标记

对于 LENGTH\_PROVIDED 类的标记，数据在显式表达随后数据长度的长度字段后面。尽管数据本身可携带长度标识符，整个语法仍定义了长度字段。

该类中元素的表格如表格 4 所示。

组块类型(十六进制)	数据流	流属性标记(十六进制)
0x00	PCM 流	0x40
0x01	WMA 语音	0x41
0x02	RT 语音	0x42
0x03	WMA Std	0x43
0x04	WMA +	0x44
0x05	WMA Pro	0x45
0x06	WMA 无损	0x46
0x07	PLEAC	0x47
.....	.....	
0x3E	附加编/解码器	0x7E

5

表格 4. LENGTH\_PROVIDED 类的元素

LENGTH\_PROVIDED 类中元数据元素的表格如下表 5 所示。

组块类型(十六进制)	元数据
0x80	内容描述符元数据
0x81	向下折叠
0x82	动态范围控制
0x83	多字节填充元素
0x84	呈现时间标记
....	....
0x92	附加元数据

表格 5. LENGTH\_PROVIDED 类中元数据元素

LENGTH 字段元素遵从标记的 LENGTH\_PROVIDED 类。LENGTH 字段元素的表格如下表 6 所示。

字段的第一比特(MSB)	长度定义
0	一单字节长度字段 (MSB 是比特 7) 7 个 LSB (比特数是 6 到 0) 以字节数指示随后数据字段的尺寸。 这是用于除某些音频有效载荷外所有数据的最通用尺寸字段。
1	一三字节长度字段 (MSB 是比特 23) 比特数 22 到 3 以字节数指示随后字段的尺寸 如果使用长度字段来定义音频有效载荷的尺寸，比特数 2 到 0 指示音频帧的数量
1	如果比特 22 到 3 的值为 “FFFFF”，这表示一转义码，而比特 2 到 0 是不受约束的。它后面跟有 4 字节大小的字段，指示有效组合的附加字节大小。该值 FFFF 被加到附加的 4 字节无符号位上以得到总的数据的字节数长度。

表格 6. LENGTH\_PROVIDED 标记后 LENGTH 字段的元素

对于 LENGTH\_AND\_MEANING\_PREDEFINED 的标记，下表 7 定义了组块类型之后字段的长度。

组块类型 (十六进制)	名称	长度
0x93	同步字	5 字节
0x94	CRC	2 字节
0x95	单字节填充元素	1 字节
0x96	END_OF_BLOCK	1 字节
...	...	...
0xBF	(附加标记定义)	XX

表格 7. 对 LENGTH\_AND\_MEANING\_PREDEFINED 标记的组块类型之后字段长度

对于 LENGTH\_PREDEFINED 标记，组块类型的比特 5 到 3 定义了不理解该组块类型的解码器，或不需要把数据包括在该组块类型的解码器在组块类型之后必须跳过的数据长度，如表 8 所示。组块类型的两个最高位比特（即比特 7 和 6）= 11。

组块类型比特数 5 到 3	要跳过的数据长度 (单位: 字节)
000	1
001	1
010	2
011	4
100	8
101	16
110	32
111	32

表格 8. 对 LENGTH\_PREDEFINED 标记在组块类型之后要跳过的数据长度

对于 2-字节、4-字节、8-字节、16-字节数据，最多有 8 个不同标记是可能的，由组块类型的比特 2 到 0 表示。对于 1-字节和 32-字节数据而言，可能的标记数量加倍为 16，因为 1-字节和 32-字节数据都可以用两种方法表示（例如，1-字节的 000 或 001 和 32-字节的 110 或 111，比特数为 5 到 3，如上图 8 所示）。

## B.元数据字段

向下折叠：该字段包含创作者控制折叠情形中有关折叠矩阵的信息。该字段携带根据所携带的折叠组合其尺寸可变化的折叠矩阵。在最坏情形中，对于从 7.1 (8 信道，包括次低音扬声器) 到 5.1 (6 信道，包括次低音扬声器) 的向下折叠，  
5 尺寸会是 8x6 矩阵。向下折叠字段在每个访问单元中重复以涵盖向下折叠矩阵随时  
间改变的情形。

DRC：该字段包含文件的 DRC (动态范围控制) 信息 (例如 DRC 系数)。

内容描述符元数据：在此例中，元数据组块提供用于传送有关音频流内容的基本描述信息的低比特率信道。内容描述符元数据为 32 字节长。该字段是可选的，  
10 且如果必要可每三秒重复一次以节约带宽。有关内容描述符元数据的更多细节在上面的表格 2 中提供。

真正的内容描述符字符串由接收器从元数据中所包含的字节流进行组合。流中的每个字节表示一个 UTF-8 字符。如果元数据字符串在结束块之前结束了，可用 0x00 填充元数据。字符串的开始和结束由 “Type” 字段中的转换暗示。因此，  
15 在发送内容描述符元数据时，发送器在所有的 4 个类型上循环—即使一个或多个字符串是空的。

已在说明书和附图中描述并说明了本发明的诸原理，可以理解各种实施例可在排列和细节上作更改而不背离这些原理。应当理解在此描述的程序、过程、或方法不相关或不受限于任何特定类型的计算环境，除非另有指出。各类通用或专用计算环境可根据在此所述的教学来使用或执行操作。软件中所示实施例的元素可在硬件中得到实现，反之亦然。  
20

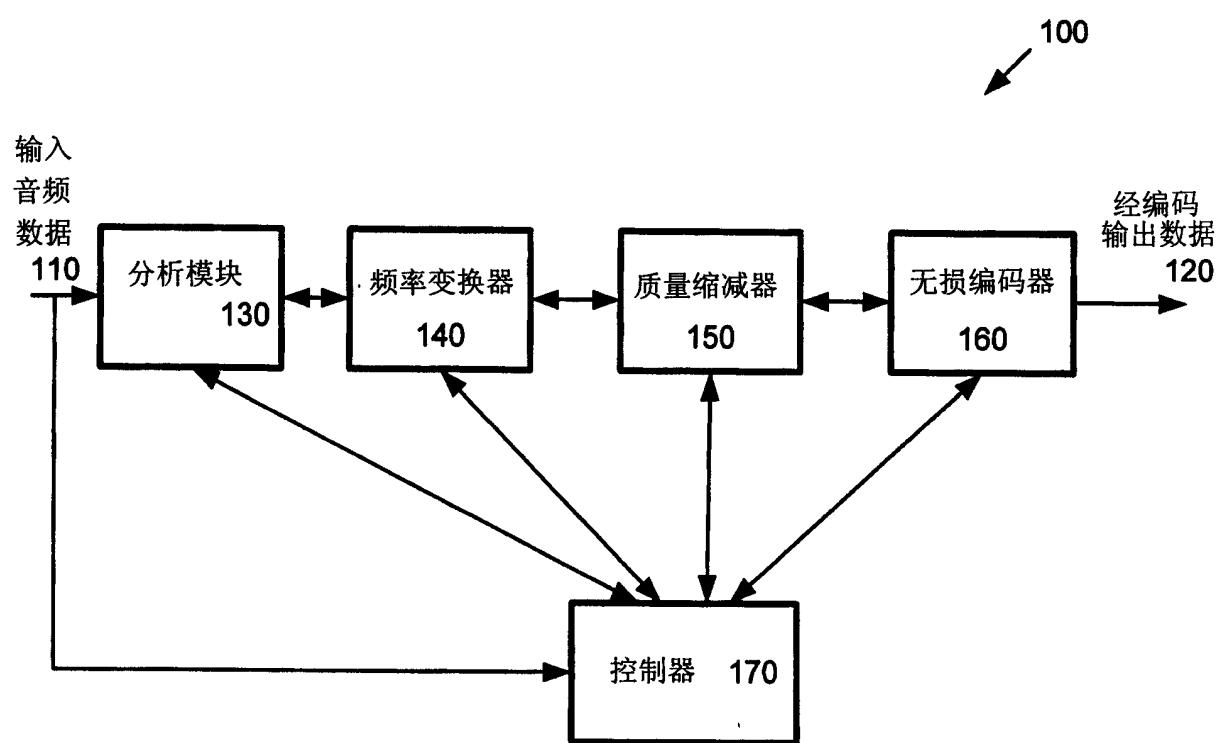


图 1

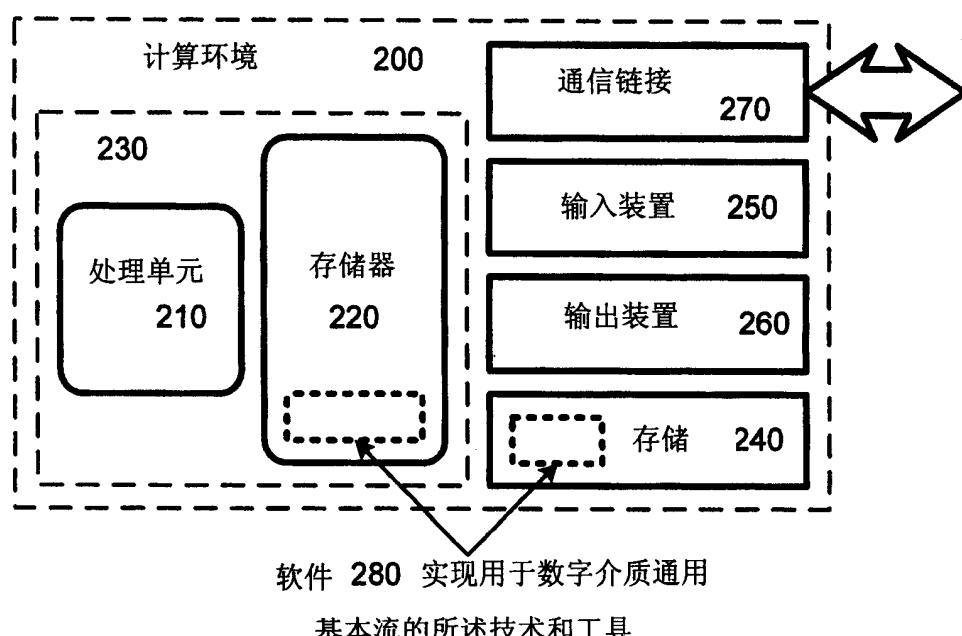


图 2

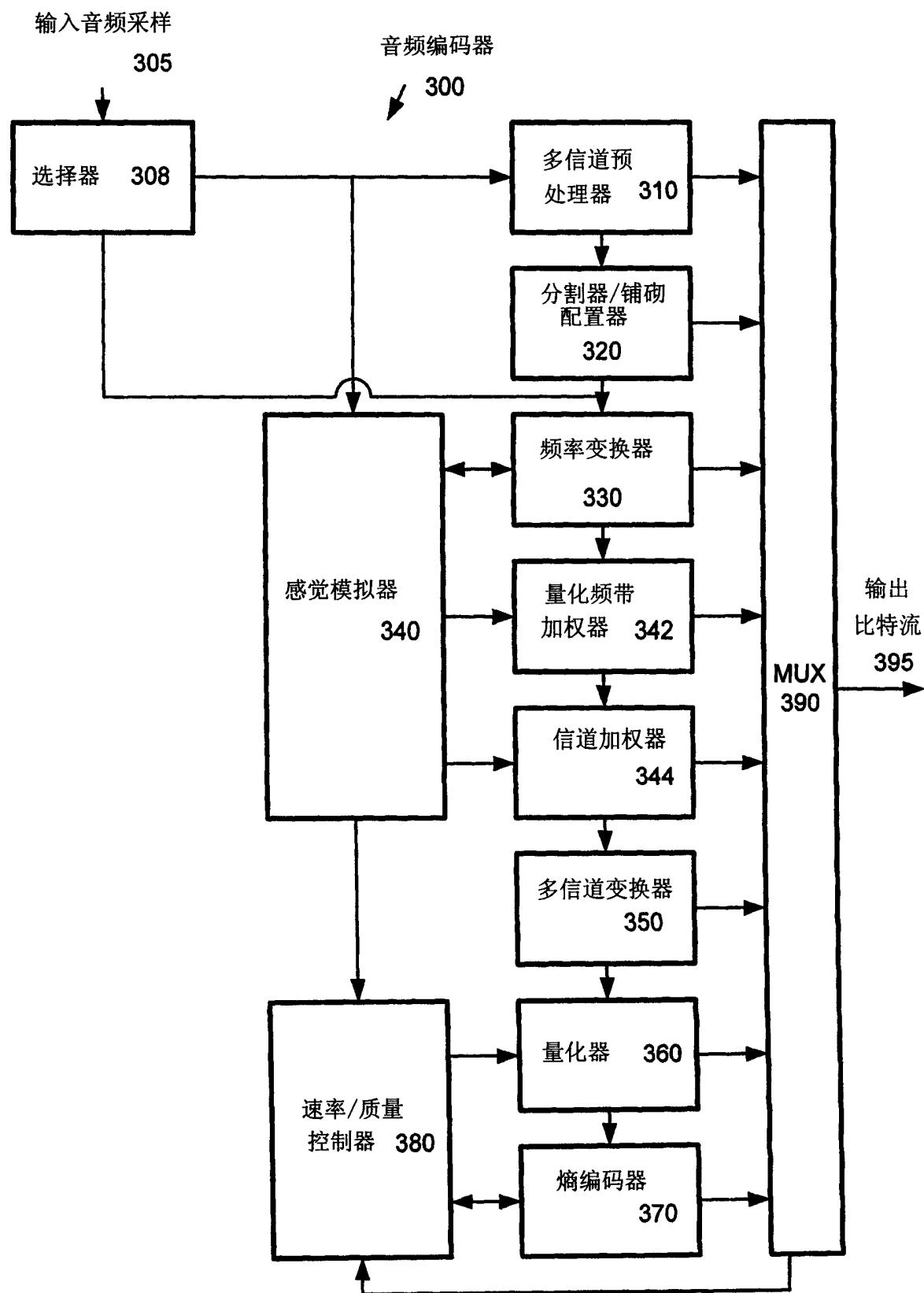


图 3

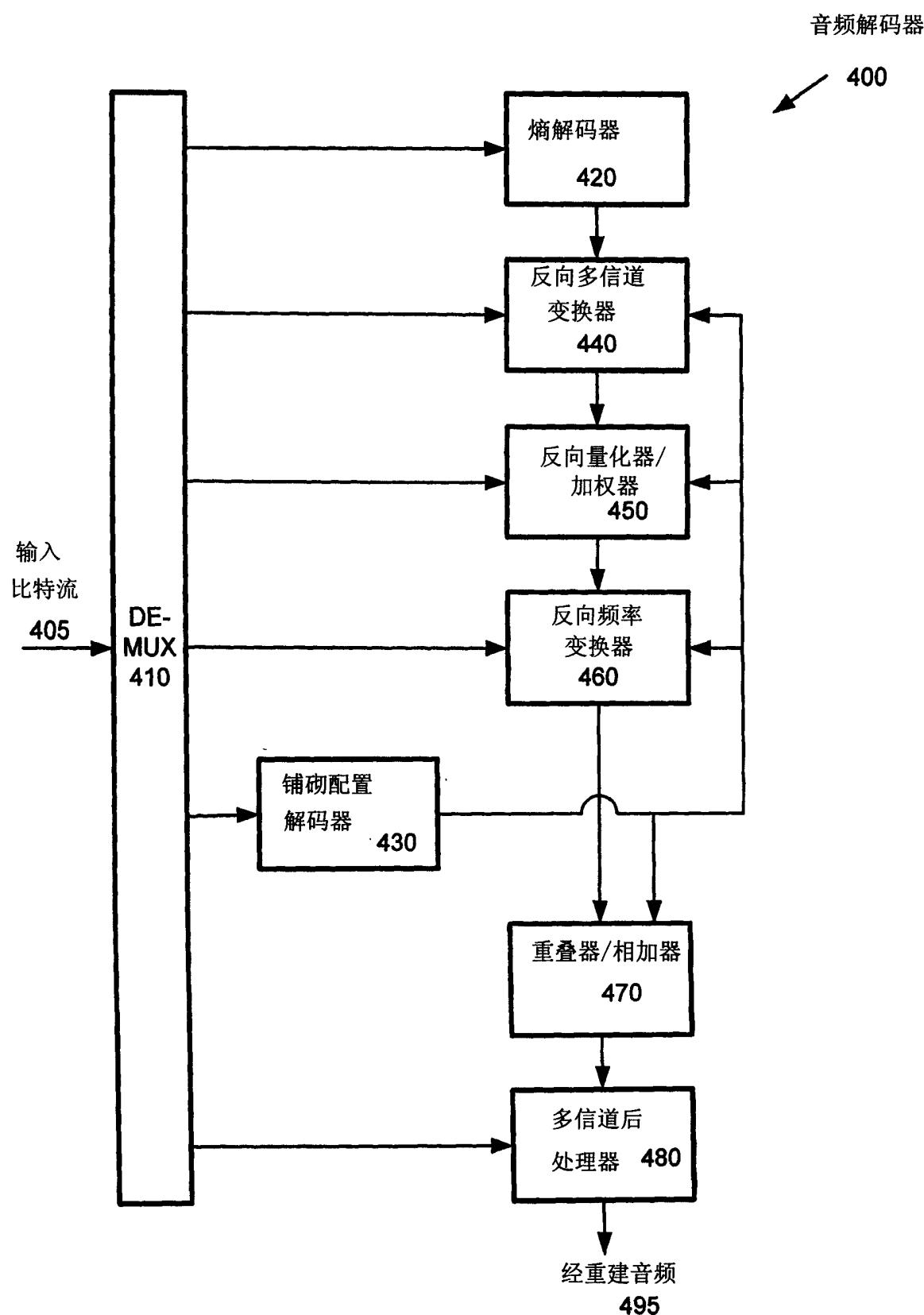


图 4

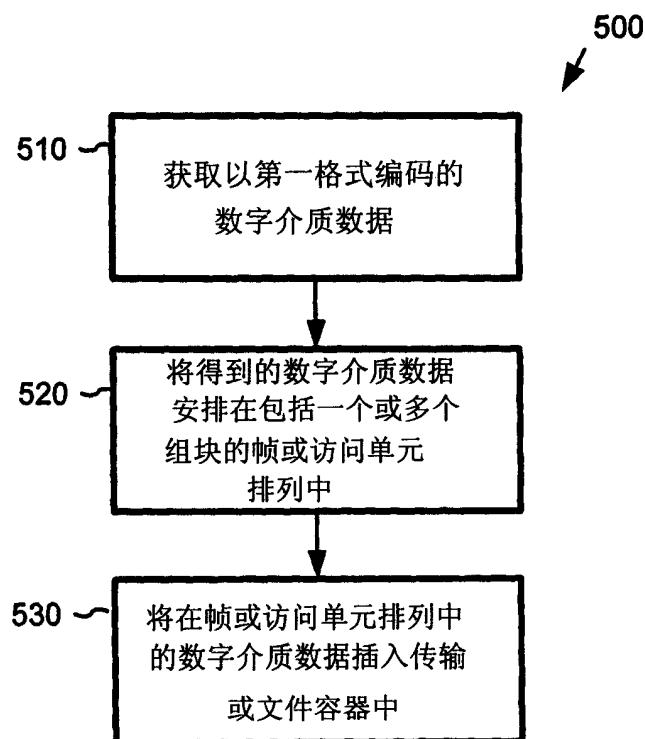


图 5

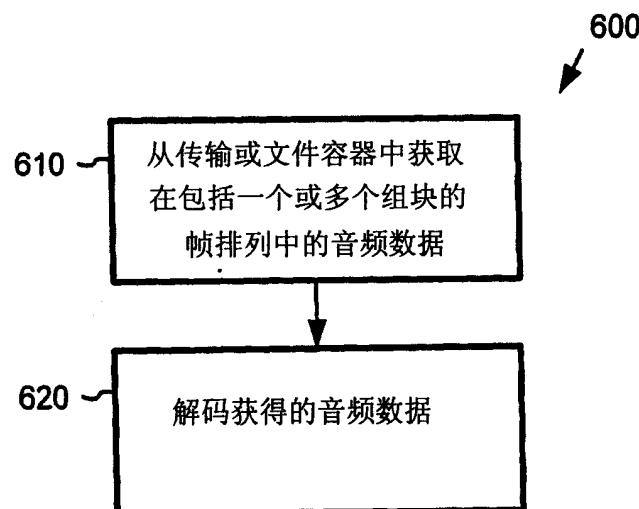


图 6

WMA Pro映射到DVD-A CA区域

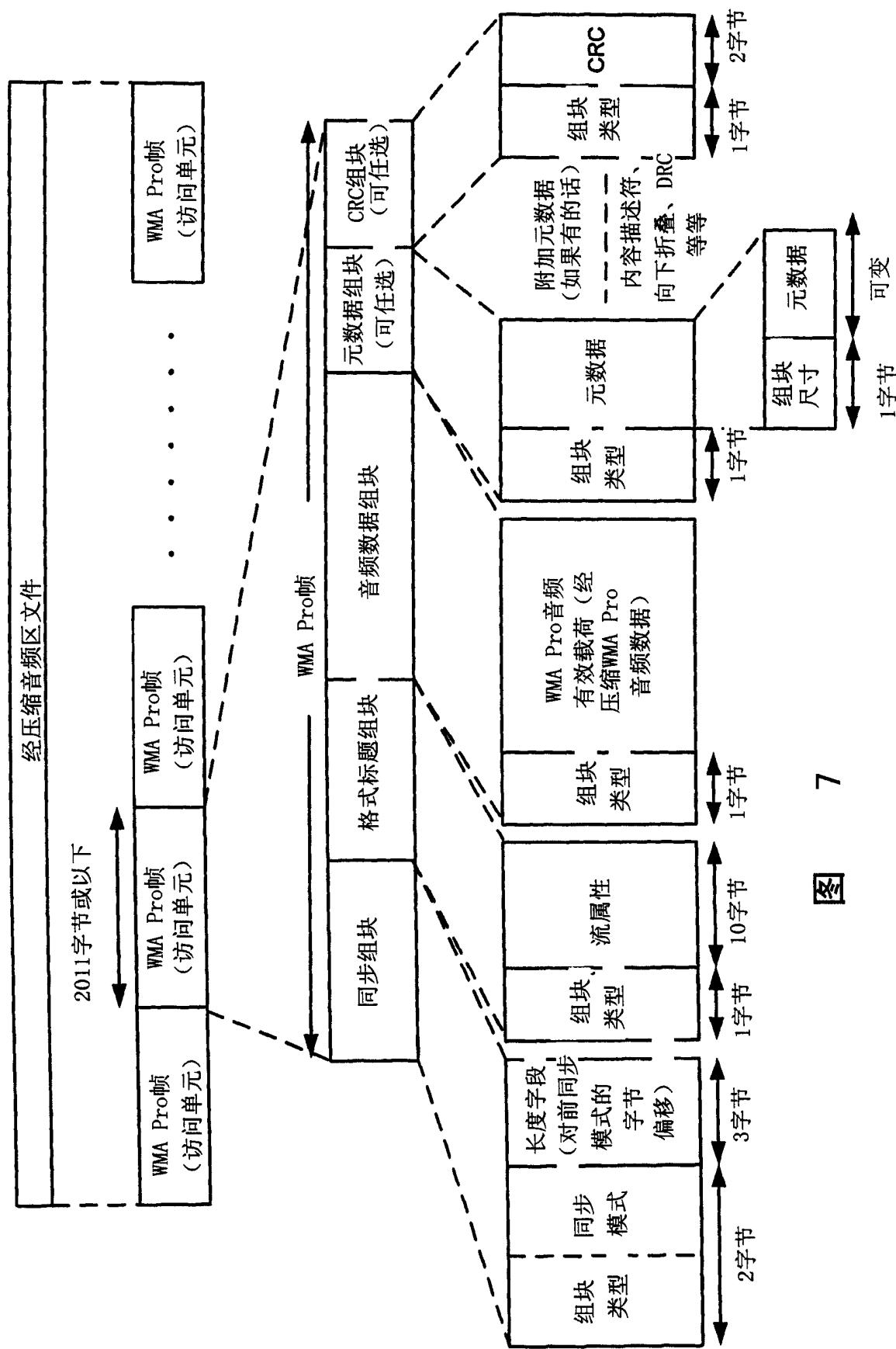
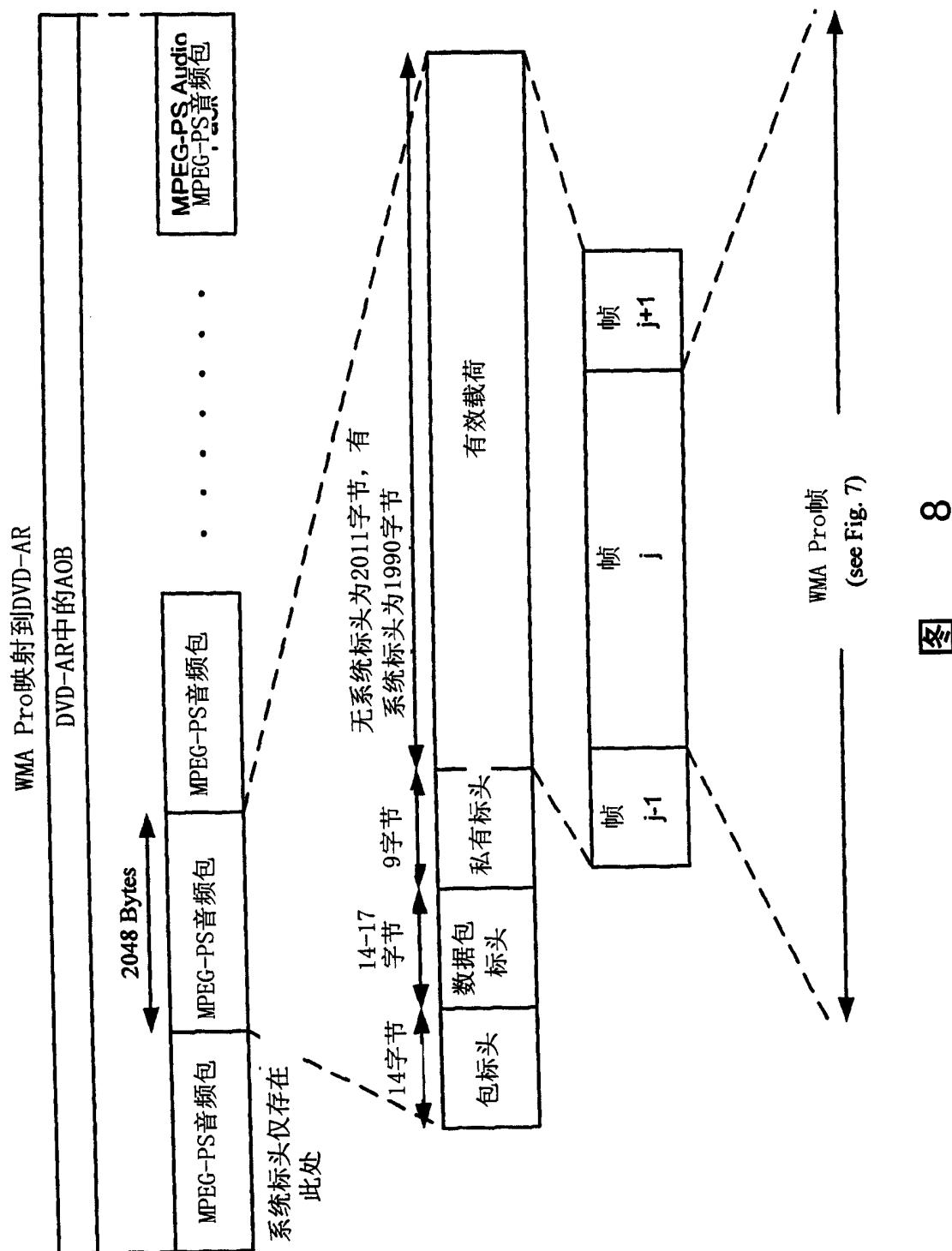


图 7



Windows介质音频基本流

