



(12) 发明专利

(10) 授权公告号 CN 101351845 B

(45) 授权公告日 2011.01.26

(21) 申请号 200780001025.1

代理人 余刚 吴孟秋

(22) 申请日 2007.07.25

(51) Int. Cl.

(30) 优先权数据

203037/2006 2006.07.26 JP

G11B 27/00 (2006.01)

G11B 20/10 (2006.01)

H04N 5/85 (2006.01)

H04N 5/91 (2006.01)

(85) PCT申请进入国家阶段日

2008.03.25

(86) PCT申请的申请数据

PCT/JP2007/065030 2007.07.25

(56) 对比文件

JP 特表 2005-524191 A, 2005.08.11, 全文.

JP 特开 2004-350251 A, 2004.12.09, 全文.

CN 1609984 A, 2005.04.27, 全文.

(87) PCT申请的公布数据

W02008/013308 JA 2008.01.31

审查员 王宁

(73) 专利权人 索尼株式会社

地址 日本东京

(72) 发明人 矶部幸雄 有留宪一郎 森本直树

前笃 前田哲裕

(74) 专利代理机构 北京康信知识产权代理有限

责任公司 11240

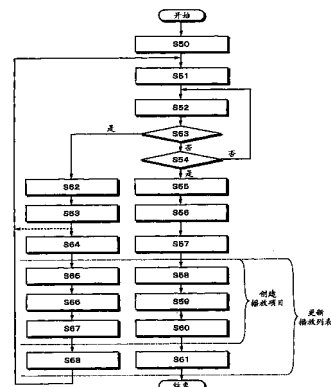
权利要求书 4 页 说明书 35 页 附图 35 页

(54) 发明名称

记录装置、记录方法和记录程序、以及摄像装置、摄像方法和摄像程序

(57) 摘要

本发明提供了一种记录装置、记录方法和记录程序、以及摄像装置、摄像方法和摄像程序。在将记录开始直至停止期间生成的 AV 数据作为文件进行记录时,易于长时间的连续记录和连续再生。以包为单位,将多路复用 AV 数据的流记录到文件中。当由于下一包的记录导致文件大小超出 2GB 时,执行记录控制,以便使该文件的音频数据的后端与下一个记录流的文件的音频数据的开头在时间上相对应,并在关闭该文件的同时,指定关闭的文件与下一个记录流的文件以音频数据的没有重复的方式无缝连接。新创建文件并继续流的记录。通过停止操作来停止文件的记录。对于长时间的记录,文件可被自动分割。再生时,可在无意识文件分割的情况下连续再生长时间记录的文件。



1. 一种记录装置,所述记录装置用于对视频数据和音频数据进行多路复用并记录于记录介质,其特征在于,包括:

数据输入部,被输入所述视频数据和所述音频数据;记录指示输入部,被输入所述视频数据和所述音频数据的记录开始和记录停止的指示;

记录部,以包为单位多路复用所述视频数据和所述音频数据,并按每个所述包依次将多路复用后的流记录到所述记录介质上的流文件中;

管理信息生成部,生成流信息文件和再生列表文件,其中,所述流信息文件针对记录到所述记录介质中的所述流文件,至少使所述流文件的再生时刻信息和地址信息相对应,所述再生列表文件存储通过对所述流文件设定再生开始点和再生结束点而指定再生区间的再生区间数据,并且,所述再生列表文件能够存储表示对所述流文件再生的再生时刻信息的标志信息;以及

控制部,用于控制所述记录部和所述管理信息生成部,

其中,所述控制部响应来自所述记录指示输入部的所述记录开始和所述记录停止的指示,控制所述记录部,使得分别开始和停止将所述流向所述流文件的记录,

在通过将由一个或多个所述包构成的规定单位记录到所述流文件中,从而使表示所述流文件的属性的信息没有满足规定限制的情况下,所述控制部控制所述记录部,使得所述记录部关闭所述流文件,创建新的所述流文件,并从所述规定单位开始,对所述新的流文件依次记录所述流。

2. 根据权利要求1所述的记录装置,其特征在于,

所述控制部控制所述记录部的记录,使得响应基于所述规定限制的所述流文件的关闭,将被关闭的所述流文件中记录的所述音频数据的后端时刻、与所述新的流文件中存储的所述音频数据的前端时刻相对应,并且,

所述控制部控制所述记录部,使得所述记录部将表示记录到所述被关闭的流文件中的视频数据和音频数据、与记录到所述新的流文件中的视频数据和音频数据连续再生的信息设置在将所述被关闭的流文件指定作为再生区间的再生区间数据内,并将所述再生区间数据记录到所述记录介质上。

3. 根据权利要求1所述的记录装置,其特征在于,

所述流文件的属性是文件大小,

所述规定限制是所述文件大小的上限。

4. 根据权利要求1所述的记录装置,其特征在于,

所述流文件的属性是存储在所述流信息文件中的入口点信息,所述入口点信息表示存储在所述流文件中的所述视频数据的时刻的信息、与所述流文件内的地址相关联,所述规定限制是一个所述流信息文件能够存储的所述入口点信息的数量的上限。

5. 根据权利要求1所述的记录装置,其特征在于,

所述流文件的属性是表示存储在所述流文件中的所述视频数据的时刻的信息,

所述规定限制是表示所述时刻的信息能够表现的时刻的上限。

6. 根据权利要求1所述的记录装置,其特征在于,

所述控制部根据所述记录指示输入部的记录停止的指示来控制所述记录部,使得在所述流文件内,所述视频数据的前端的再生时刻与所述音频数据的前端的再生时刻一致,且

所述音频数据的终端的再生时刻比所述视频数据的终端的再生时刻靠后,并且,

所述控制部控制所述记录部,使得所述记录部在将被记录的所述流文件指定作为再生区间的所述再生区间数据内,设置表示在帧定时连续再生所述流文件和所述流文件的后续流文件的信息,并将所述再生区间数据记录到所述记录介质上。

7. 一种记录方法,用于对视频数据和音频数据进行多路复用并记录于记录介质,其特征在于,包括:

记录指示输入步骤,输入通过数据输入而输入的所述视频数据和所述音频数据的记录开始和记录停止的指示;

记录步骤,以包为单位多路复用所述视频数据和所述音频数据,并按每个所述包依次将多路复用后的流记录到所述记录介质上的流文件中;

管理信息生成步骤,生成流信息文件和再生列表文件,其中,所述流信息文件针对记录到所述记录介质中的所述流文件,至少使所述流文件的再生时刻信息和地址信息相对应,所述再生列表文件存储通过对所述流文件设定再生开始点和再生结束点而指定再生区间的再生区间数据,并且,所述再生列表文件能够存储表示对所述流文件再生的再生时刻信息的标志信息;以及

控制步骤,控制所述记录步骤和所述管理信息生成步骤,

其中,在所述控制步骤中,

根据在所述记录指示输入步骤中的所述记录开始和所述记录停止的指示,控制所述记录步骤,使得分别开始和停止将所述流向所述流文件的记录,

在通过将由一个或多个所述包构成的规定单位记录到所述流文件中,从而使表示所述流文件的属性的信息没有满足规定限制的情况下,控制所述记录步骤,使得在所述记录步骤中关闭所述流文件,创建新的所述流文件,并从所述规定单位开始,对所述新的流文件依次记录所述流。

8. 一种摄像装置,对视频数据和音频数据进行多路复用并记录于记录介质,其中,所述视频数据是由摄像部拍摄被摄体而获得的,所述音频数据是由声音收集部收集声音而获得的,所述摄像装置的特征在于,包括:

所述摄像部,拍摄被摄体,并输出所述视频数据;

所述声音收集部,收集声音,并输出所述音频数据;记录部,以包为单位多路复用所述视频数据和所述音频数据,并按每个所述包依次将多路复用后的流记录到记录介质上的流文件中;

操作部,受理用户操作,其中,所述用户操作指示对所述记录介质开始记录和停止记录所述视频数据和所述音频数据;

管理信息生成部,生成流信息文件和再生列表文件,其中,所述流信息文件针对记录到所述记录介质中的所述流文件,至少使所述流文件的再生时刻信息和地址信息相对应,所述再生列表文件存储通过对所述流文件设定再生开始点和再生结束点而指定再生区间的再生区间数据,并且,所述再生列表文件能够存储表示对所述流文件再生的再生时刻信息的标志信息;以及

控制部,用于控制所述记录部和所述管理信息生成部,其中,所述控制部响应基于对所述操作部的操作的所述记录开始和所述记录停止的指示,控制所述记录部,使得分别开始

和停止将所述流向所述流文件的记录,

在通过将由一个或多个所述包构成的规定单位记录到所述流文件中,从而使表示所述流文件的属性的信息没有满足规定限制的情况下,所述控制部控制所述记录部,使得所述记录部关闭所述流文件,创建新的所述流文件,并从所述规定单位开始,对所述新的流文件依次记录所述流。

9. 根据权利要求 8 所述的摄像装置,其特征在于,

所述控制部控制所述记录部的记录,使得响应基于所述规定限制的所述流文件的关闭,将被关闭的所述流文件中记录的所述音频数据的后端时刻、与所述新的流文件中存储的所述音频数据的前端时刻相对应,并且,

所述控制部控制所述记录部,使得所述记录部将表示记录到所述被关闭的流文件中的视频数据和音频数据、与记录到所述新的流文件中的视频数据和音频数据连续再生的信息设置在将所述被关闭的流文件指定作为再生区间的再生区间数据内,并将所述再生区间数据记录到所述记录介质上。

10. 根据权利要求 8 所述的摄像装置,其特征在于,

所述流文件的属性是文件大小,

所述规定限制是所述文件大小的上限。

11. 根据权利要求 8 所述的摄像装置,其特征在于,

所述流文件的属性是存储在所述流信息文件中的入口点信息,所述入口点信息使表示存储在所述流文件中的所述视频数据的时刻的信息、与所述流文件内的地址相关联,所述规定限制是一个所述流信息文件能够存储的所述入口点信息的数量的上限。

12. 根据权利要求 8 所述的摄像装置,其特征在于,

所述流文件的属性是表示存储在所述流文件中的所述视频数据的时刻的信息,

所述规定限制是表示所述时刻的信息能够表现的时刻的上限。

13. 根据权利要求 8 所述的摄像装置,其特征在于,

所述控制部根据与对所述操作部的操作相对应的记录停止的指示来控制所述记录部,使得在所述流文件内,所述视频数据的前端的再生时刻与所述音频数据的前端的再生时刻一致,且所述音频数据的终端的再生时刻比所述视频数据的终端的再生时刻靠后,并且,

所述控制部控制所述记录部,使得所述记录部在将被记录的所述流文件指定作为再生区间的所述再生区间数据内,设置表示在帧定时连续再生所述流文件和所述流文件的后续流文件的信息,并将所述再生区间数据记录到所述记录介质上。

14. 一种摄像装置的摄像方法,在所述摄像装置中,对视频数据和音频数据进行多路复用并记录于记录介质,其中,所述视频数据是由摄像部拍摄被摄体而获得的,所述音频数据是由声音收集部收集声音而获得的,所述摄像方法的特征在于,包括:记录步骤,以包为单位多路复用拍摄被摄体而获得的视频数据和收集声音而获得的音频数据,并按每个所述包依次将多路复用后的流记录到记录介质上的流文件中;

受理步骤,受理对操作部的用户操作,其中,所述用户操作指示对所述记录介质开始记录和停止记录所述视频数据和所述音频数据;

管理信息生成步骤,生成流信息文件和再生列表文件,其中,所述流信息文件针对记录到所述记录介质中的所述流文件,至少使所述流文件的再生时刻信息和地址信息相对应,

所述再生列表文件存储通过对所述流文件设定再生开始点和再生结束点而指定再生区间的一个以上的再生区间数据,并且,所述再生列表文件能够存储表示对所述流文件再生的再生时刻信息的标志信息;以及

控制步骤,控制所述记录步骤和所述管理信息生成步骤,

其中,在所述控制步骤中,

响应基于所述操作部的操作的所述记录开始和所述记录停止的指示,控制所述记录步骤,使得分别开始和停止将所述流向所述流文件的记录,

在通过将由一个或多个所述包构成的规定单位记录到所述流文件中,从而使表示所述流文件的属性的信息没有满足规定限制的情况下,控制所述记录步骤,使得在所述记录步骤中关闭所述流文件,创建新的所述流文件,并从所述规定单位开始,对所述新的流文件依次记录所述流。

记录装置、记录方法和记录程序、以及摄像装置、摄像方法和摄像程序

技术领域

[0001] 本发明涉及适用于将用于对音频数据和视频数据进行多路复用后的流数据记录于记录介质的记录装置、记录方法和记录程序、以及摄像装置、摄像方法和摄像程序。

背景技术

[0002] 现有技术中,可以使用磁带作为在例如摄像机等中,对通过摄像元件拍摄的映像信号转换成数字视频数据并对其压缩编码,从而进行记录时的记录介质。磁带适于连续记录大量的数据。

[0003] 另一方面,近年来,作为可记录并可从记录再生装置上卸除,同时记录容量较大,适于记录由视频数据和音频数据构成的 AV(Audio/Video, 音频/视频)数据的记录介质,具有 4.7GB(Giga Byte, 千兆字节)以上记录容量的 DVD(Digital Versatile Disc, 数字化视频光盘)得以广泛普及。在专利文献(日本特开 2004-350251)中公开了一种以 DVD-Video 格式对可记录型 DVD 进行记录的摄像装置。

[0004] 这种可记录型 DVD 可以使用 UDF(Universal Disk Format, 通用盘格式)作为文件系统,并可通过与 UDF 相对应的计算机装置进行存取。UDF 包含基于 ISO(International Organization for Standardization, 国际标准化组织)9660 的格式,并可通过计算机装置所使用的各种文件系统进行存取。由于将视频数据和音频数据作为文件并记录于可记录型 DVD,从而可增加与计算机装置等其它装置的亲和性,并可更为有效地活用记录的数据。

[0005] 近年来,还出现了一种在摄像机中内置硬盘驱动器,并将拍摄获得的视频数据和音频数据作为文件而记录到该硬盘驱动器中的产品。此外,还公开了一种将大容量的半导体存储器用作记录介质的产品。

[0006] 在现有技术中的、使用磁带作为记录介质的情况下,一般以从记录开始操作直至记录停止操作之间生成的视频数据为单位,将视频数据和音频数据记录于记录介质。因此,即使在将视频数据和音频数据作为文件而进行记录的情况下,若也以从记录开始操作直至记录停止操作之间生成的数据为单位来生成文件,则易于与现有技术的记录方式相协调,因而优选。

[0007] 但是,当将连续提供的视频数据和音频数据作为文件而记录于记录介质时,由于系统上的制约,在从记录开始操作直至记录停止操作之间,对可记录的记录时间产生限制。作为系统上的限制,可以考虑有:诸如适用于记录介质的文件系统上的文件大小的限制、以及存储有视频数据和音频数据的文件中的管理信息相关的限制等。

[0008] 对文件系统上的限制进行概略说明。当通过计算机装置处理记录于记录介质的文件时,记录于记录介质的文件的形式需与计算机装置的文件系统相对应。在此,通过考虑用于计算机装置的更低级(low level)的文件系统,所以可被更多的计算机装置所读取。例如,可考虑与作为计算机装置的 OS(Operating System, 操作系统)之一的 Windows(注册商标)中所使用的文件系统、即 FAT16(File Allocation Table 16, 文件分配表)具有互换

性。

[0009] 在 FAT16 中,一个文件的最大尺寸被限制为 2GB(Giga Byte,千兆字节)。另一方面,在 UDF 中,一个文件的最大尺寸远大于 2GB。因此,当以 UDF 标准为基准而记录的文件的尺寸超过 2GB 时,则该文件在 FAT16 中无法处理。

[0010] 因此,当用户考虑例如摄像机时,存在以下问题:用户在考虑使用作为文件而记录的数字视频数据的计算机装置的文件系统的同时,必须执行记录开始及停止,从而导致操作性显著下降。而且,因此,还存在以下问题:在不进行可对记录介质执行记录时间的记录中,考虑不得不停止记录,从而无法确保与记录容量相应的连续记录时间。

[0011] 为了避免这种问题,例如可考虑由系统对记录中的文件大小进行监视,一旦文件大小达到规定尺寸,便关闭该文件,重新创建文件,以便继续记录。但是,在这种情况下,也存在以下问题:在再生时,需要用于连续再生通过连续记录而生成的多个文件的操作,此对于用户增添了负担。

发明内容

[0012] 因此,本发明的目的在于提供在将记录开始直至记录停止期间生成的视频数据和音频数据作为文件而记录时,易于长时间的连续记录和连续再生的记录装置、记录方法和记录程序、以及摄像装置、摄像方法和摄像程序。

[0013] 为解决上述问题,本发明的第一方面提供了记录装置,其对视频数据和音频数据进行多路复用并记录于记录介质,在记录装置中,包括:数据输入部,被输入视频数据和音频数据;记录指示输入部,被输入视频数据和音频数据的记录开始和记录停止的指示;记录部,以包为单位多路复用视频数据和音频数据,并按每个包依次将多路复用后的流记录到记录介质上的流文件中;管理信息生成部,生成流信息文件和再生列表文件,其中,流信息文件针对记录到记录介质中的流文件,至少使流文件的再生时刻信息和地址信息相对应,再生列表文件存储通过对流文件设定再生开始点和再生结束点而指定再生区间的一个以上的再生区间数据,并且,再生列表文件可存储表示对流文件再生的再生时刻信息的标志信息;以及控制部,用于控制记录部和管理信息生成部,其中,控制部根据基于记录指示输入部的记录开始和记录停止的指示,控制记录部,以便分别开始和停止将流向流文件的记录,在通过将由一个或多个包构成的规定单位记录到流文件中,从而使表示流文件的属性的信息没有满足规定限制的情况下,控制部控制记录部,以便使得记录部关闭流文件,创建新的流文件,并从规定单位开始,对新的流文件依次记录流。

[0014] 而且,本发明的第二方面提供了记录方法,其对视频数据和音频数据进行多路复用并记录于记录介质,在记录方法中,包括:记录指示输入步骤,输入通过数据输入而输入的视频数据和音频数据的记录开始和记录停止的指示;记录步骤,以包为单位多路复用视频数据和音频数据,并按每个包依次将多路复用后的流记录到记录介质上的流文件中;管理信息生成步骤,生成流信息文件和再生列表文件,其中,流信息文件针对记录到记录介质中的流文件,至少使流文件的再生时刻信息和地址信息相对应,再生列表文件存储通过对流文件设定再生开始点和再生结束点而指定再生区间的一个以上的再生区间数据,并且,再生列表文件可存储表示对流文件再生的再生时刻信息的标志信息;以及控制步骤,控制记录步骤和管理信息生成步骤,其中,在控制步骤中,根据在记录指示输入步骤中的记录开

始和记录停止的指示,控制记录步骤,以便分别开始和停止将流向流文件的记录,在通过将由一个或多个包构成的规定单位记录到流文件中,从而使表示流文件的属性的信息没有满足规定限制的情况下,控制记录步骤,以便使得在记录步骤中关闭流文件,创建新的流文件,并从规定单位开始,对新的流文件依次记录流。

[0015] 此外,本发明的第三方面提供了记录程序,其使计算机装置执行对视频数据和音频数据进行多路复用并记录于记录介质的记录方法,在记录程序中,记录方法包括:记录指示输入步骤,输入通过数据输入而输入的视频数据和音频数据的记录开始和记录停止的指示;记录步骤,以包为单位多路复用视频数据和音频数据,并按每个包依次将多路复用后的流记录到记录介质上的流文件中;管理信息生成步骤,生成流信息文件和再生列表文件,其中,流信息文件针对记录到记录介质中的流文件,至少使流文件的再生时刻信息和地址信息相对应,再生列表文件存储通过对流文件设定再生开始点和再生结束点而指定再生区间的一个以上的再生区间数据,并且,再生列表文件可存储表示对流文件再生的再生时刻信息的标志信息;以及控制步骤,控制记录步骤和管理信息生成步骤,其中,在控制步骤中,根据在记录指示输入步骤中的记录开始和记录停止的指示,控制记录步骤,以便分别开始和停止将流向流文件的记录,在通过将由一个或多个包构成的规定单位记录到流文件中,从而使表示流文件的属性的信息没有满足规定限制的情况下,控制记录步骤,以便使得在记录步骤中关闭流文件,创建新的流文件,并从规定单位开始,对新的流文件依次记录流。

[0016] 并且,本发明的第四方面提供了摄像装置,对视频数据和音频数据进行多路复用并记录于记录介质,其中,视频数据是由摄像部拍摄被摄体而获得的,音频数据是由声音收集部收集声音而获得的,在摄像装置中,包括:摄像部,拍摄被摄体,并输出视频数据;声音收集部,收集声音,并输出音频数据;记录部,以包为单位多路复用视频数据和音频数据,并按每个包依次将多路复用后的流记录到记录介质上的流文件中;操作部,受理用户操作,其中,用户操作指示对记录介质开始记录和停止记录视频数据和音频数据;管理信息生成部,生成流信息文件和再生列表文件,其中,流信息文件针对记录到记录介质中的流文件,至少使流文件的再生时刻信息和地址信息相对应,再生列表文件存储通过对流文件设定再生开始点和再生结束点而指定再生区间的一个以上的再生区间数据,并且,再生列表文件可存储表示对流文件再生的再生时刻信息的标志信息;以及控制部,用于控制记录部和管理信息生成部,其中,控制部根据基于对操作部的操作的记录开始和记录停止的指示,控制记录部,以便分别开始和停止将流向流文件的记录,在通过将由一个或多个包构成的规定单位记录到流文件中,从而使表示流文件的属性的信息没有满足规定限制的情况下,控制部控制记录部,以便使得记录部关闭流文件,创建新的流文件,并从规定单位开始,对新的流文件依次记录流。

[0017] 此外,本发明的第五方面提供了摄像装置的摄像方法,在摄像装置中,对视频数据和音频数据进行多路复用并记录于记录介质,其中,视频数据是由摄像部拍摄被摄体而获得的,音频数据是由声音收集部收集声音而获得的,在摄像方法中,包括:记录步骤,以包为单位多路复用拍摄被摄体而获得的视频数据和收集声音而获得的音频数据,并按每个包依次将多路复用后的流记录到记录介质上的流文件中;受理步骤,受理对操作部的用户操作,其中,用户操作指示对记录介质开始记录和停止记录视频数据和音频数据;管理信息生成步骤,生成流信息文件和再生列表文件,其中,流信息文件针对记录到记录介质中的流文

件,至少使流文件的再生时刻信息和地址信息相对应,再生列表文件存储通过对流文件设定再生开始点和再生结束点而指定再生区间的一个以上的再生区间数据,并且,再生列表文件可存储表示对流文件再生的再生时刻信息的标志信息;以及控制步骤,控制记录步骤和管理信息生成步骤,其中,在控制步骤中,根据基于操作部的操作的记录开始和记录停止的指示,控制记录步骤,以便分别开始和停止将流向流文件的记录,在通过将由一个或多个包构成的规定单位记录到流文件中,从而使表示流文件的属性的信息没有满足规定限制的情况下,控制记录步骤,以便使得在记录步骤中关闭流文件,创建新的流文件,并从规定单位开始,对新的流文件依次记录流。

[0018] 并且,本发明的第六方面提供了使计算机装置执行摄像装置的摄像方法的摄像程序,在摄像装置中,对视频数据和音频数据进行多路复用并记录于记录介质,其中,视频数据是由摄像部拍摄被摄体而获得的,音频数据是由声音收集部收集声音而获得的,在摄像程序中,摄像方法包括:记录步骤,以包为单位多路复用拍摄被摄体而获得的视频数据和收集声音而获得的音频数据,并按每个包依次将多路复用后的流记录到记录介质上的流文件中;受理步骤,受理对操作部的用户操作,其中,用户操作指示对记录介质开始记录和停止记录视频数据和音频数据;管理信息生成步骤,生成流信息文件和再生列表文件,其中,流信息文件针对记录到记录介质中的流文件,至少使流文件的再生时刻信息和地址信息相对应,再生列表文件存储通过对流文件设定再生开始点和再生结束点而指定再生区间的一个以上的再生区间数据,并且,再生列表文件可存储表示对流文件再生的再生时刻信息的标志信息;以及控制步骤,控制记录步骤和管理信息生成步骤,其中,在控制步骤中,根据基于操作部的操作的记录开始和记录停止的指示,控制记录步骤,以便分别开始和停止将流向流文件的记录,在通过将由一个或多个包构成的规定单位记录到流文件中,从而使表示流文件的属性的信息没有满足规定限制的情况下,控制记录步骤,以便使得在记录步骤中关闭流文件,创建新的流文件,并从规定单位开始,对新的流文件依次记录流。

[0019] 如上述,根据本发明的第一、第二及第三方面,将以包为单位多路复用从数据输入输入的视频数据和音频数据后的流按每个包依次记录到所述记录介质上的流文件中,并生成流信息文件和再生列表文件,其中,所述流信息文件针对记录到所述记录介质中的所述流文件,至少使所述流文件的再生时刻信息和地址信息相对应,所述再生列表文件存储通过对所述流文件设定再生开始点和再生结束点而指定再生区间的一个以上的再生区间数据,并且,所述再生列表文件可存储表示对所述流文件再生的再生时刻信息的标志信息,根据记录开始和记录停止的指示,控制所述记录部,以便分别开始和停止将所述流向所述流文件的记录,在通过将由一个或多个所述包构成的规定单位记录到所述流文件中,从而使表示所述流文件的属性的信息没有满足规定限制的情况下,进行控制,以便关闭所述流文件,创建新的所述流文件,并从所述规定单位开始,对所述新的流文件依次记录所述流,所以可以无意识对流文件的规定限制地进行长时间的连续记录。

[0020] 此外,根据本发明第四、第五及第六方面,以包为单位多路复用由摄像部拍摄被摄体而获得的所述视频数据和由声音收集部收集声音而获得的所述音频数据,并按每个所述包依次将多路复用后的流记录到记录介质上的流文件中,生成流信息文件和再生列表文件,其中,所述流信息文件针对记录到所述记录介质中的所述流文件,至少使所述流文件的再生时刻信息和地址信息相对应,所述再生列表文件存储通过对所述流文件设定再生开始

点和再生结束点而指定再生区间的一个以上的再生区间数据,并且,所述再生列表文件可存储表示对所述流文件再生的再生时刻信息的标志信息,所述控制部根据基于对所述操作部的操作的所述记录开始和所述记录停止的指示,控制记录,以便分别开始和停止将所述流向所述流文件的记录,在通过将由一个或多个所述包构成的规定单位记录到所述流文件中,从而使表示所述流文件的属性的信息没有满足规定限制的情况下,控制记录,以便关闭所述流文件,创建新的所述流文件,并从所述规定单位开始,对所述新的流文件依次记录所述流,因此,可以进行长时间的连续的摄影及拍摄获得的视频数据及音频数据的记录。

[0021] 如上述,本发明的第一、第二及第三方面可以获得如下效果:将以包为单位多路复用从数据输入输入的视频数据和音频数据后的流按每个包依次记录到所述记录介质上的流文件中,并生成流信息文件和再生列表文件,其中,所述流信息文件针对记录到所述记录介质中的所述流文件,至少使所述流文件的再生时刻信息和地址信息相对应,所述再生列表文件存储通过对所述流文件设定再生开始点和再生结束点而指定再生区间的一个以上的再生区间数据,并且,所述再生列表文件可存储表示对所述流文件再生的再生时刻信息的标志信息,根据记录开始和记录停止的指示,控制所述记录部,以便分别开始和停止将所述流向所述流文件的记录,在通过将由一个或多个所述包构成的规定单位记录到所述流文件中,从而使表示所述流文件的属性的信息没有满足规定限制的情况下,进行控制,以便关闭所述流文件,创建新的所述流文件,并从所述规定单位开始,对所述新的流文件依次记录所述流,所以可以无意识对流文件的规定限制地进行长时间的连续记录。

[0022] 如上述,本发明第四、第五及第六方面可以获得如下效果:以包为单位多路复用由摄像部拍摄被摄体而获得的所述视频数据和由声音收集部收集声音而获得的所述音频数据,并按每个所述包依次将多路复用后的流记录到记录介质上的流文件中,生成流信息文件和再生列表文件,其中,所述流信息文件针对记录到所述记录介质中的所述流文件,至少使所述流文件的再生时刻信息和地址信息相对应,所述再生列表文件存储通过对所述流文件设定再生开始点和再生结束点而指定再生区间的一个以上的再生区间数据,并且,所述再生列表文件可存储表示对所述流文件再生的再生时刻信息的标志信息,所述控制部根据基于对所述操作部的操作的所述记录开始和所述记录停止的指示,控制记录,以便分别开始和停止将所述流向所述流文件的记录,在通过将由一个或多个所述包构成的规定单位记录到所述流文件中,从而使表示所述流文件的属性的信息没有满足规定限制的情况下,控制记录,以便关闭所述流文件,创建新的所述流文件,并从所述规定单位开始,对所述新的流文件依次记录所述流,因此,可以进行长时间的连续的摄影及拍摄获得的视频数据及音频数据的记录。

附图说明

[0023] 图1是概略地表示被规定为可适用于本发明的 AVCHD 格式的数据模型(data model)的示意图;

[0024] 图2是用于说明索引表的简略线图;

[0025] 图3是表示片段(clip)AV流、片段信息、片段、播放项目以及播放列表的关系的UML图;

[0026] 图4是用于说明从多个播放列表参照同一片段的方法的简略线图;

- [0027] 图 5 是用于说明记录于记录介质的文件的管理结构的简略线图；
- [0028] 图 6 是表示文件“index.bdmv”的一例结构的语法的示意图；
- [0029] 图 7 是表示块 blkIndexes() 的一例结构的语法的示意图；
- [0030] 图 8 是表示文件“MovieObject.bdmv”的一例结构的语法的示意图；
- [0031] 图 9 是表示块 blkMovieObjects() 的一例结构的语法的示意图；
- [0032] 图 10 是表示播放列表文件“xxxxx.mpls”的一例结构的语法的示意图；
- [0033] 图 11 是表示块 blkPlayList() 的一例结构的语法的示意图；
- [0034] 图 12 是表示块 blkPlayItem() 的一例结构的语法的示意图；
- [0035] 图 13A 及图 13B 是用于说明第一和第二无缝连接的示意图；
- [0036] 图 14 是表示块 blkPlayListMark() 的一例结构的语法的示意图；
- [0037] 图 15 是表示片段信息文件的一例结构的语法的示意图；
- [0038] 图 16 是表示块 blkClipInfo() 的一例结构的语法的示意图；
- [0039] 图 17 是表示块 blkSequenceInfo() 的一例结构的语法的示意图；
- [0040] 图 18 是表示块 blkProgramInfo() 的一例结构的语法的示意图；
- [0041] 图 19 是表示块 blkCPI() 的一例结构的语法的示意图；
- [0042] 图 20 是表示块 blkEPMMap() 的一例结构的语法的示意图；
- [0043] 图 21 是表示块 blkEPMMapForOneStreamPID(EP_stream_type, Nc, Nf) 的一例结构的语法示意图；
- [0044] 图 22 是示出入口 PTSEPCoarse 及入口 PTSEPFine 的一例格式的示意图；
- [0045] 图 23 是示出入口 SPNEPCoarse 及入口 SPNEPFine 的一例格式的示意图；
- [0046] 图 24 是表示块 blkExtensionData() 的一例结构的语法的示意图；
- [0047] 图 25 是模式地表示块 blkExtensionData() 中的各数据的参照关系的示意图；
- [0048] 图 26 是表示对块 blkExtensionData() 写入数据时的一例处理的流程图；
- [0049] 图 27 是表示从块 blkExtensionData() 读出扩展数据时的一例处理的流程图；
- [0050] 图 28 是表示文件“index.bdmv”内的域 blkExtensionData() 中的块 DataBlock() 的一例结构的语法的示意图；
- [0051] 图 29 是表示块 blkTableOfPlayList() 的一例结构的语法的示意图；
- [0052] 图 30A 及图 30B 是概略地表示虚拟播放器(virtual player)的动作的流程图；
- [0053] 图 31 是概略地表示虚拟播放器的动作的示意图；
- [0054] 图 32 是概略地表示可适用于本发明的一实施例的记录装置的一例结构的框图；
- [0055] 图 33 是表示本发明的一实施例涉及的一例记录方法的流程图；
- [0056] 图 34 是表示通过本发明的一实施例的方法来记录的一例文件结构的示意图；
- [0057] 图 35A 及图 35B 是用于说明记录时间经过的例子的示意图；
- [0058] 以及
- [0059] 图 36 是表示本发明的一实施例的其他例子涉及的摄像机装置的一例结构的框图。

具体实施方式

[0060] 下面,参照附图,对本发明的一实施例进行说明。首先,为了便于理解,对可适用于

本发明的一例格式（下面称为 AVCHD 格式）进行说明。由于 AVCHD 格式是现有技术中已提出的一种将对视频数据和音频数据规定进行多路复用后的 AV(Audio/Video, 音频 / 视频) 流记录于可记录的记录介质的记录格式, 所以可使用播放列表, 以片段为单位来管理记录于记录介质的 AV 流。

[0061] 例如, 以 ITU-T(International Telecommunication Union-Telecommunication Standardization Sector, 国际电信联盟电信标准局) 建议的 H. 264 或 ISO(International Organization for Standardization, 国际标准化组织)/IEC(International Electrotechnical Commission, 国际电工委员会) 国际标准 14496-10(MPEG-4 part 10) Advanced Video Coding(以下简称为“H. 264|AVC”) 规定的编码方式进行编码, 并将基于 MPEG2 系统的多路复用后的位流称为片段 AV 流(或者 AV 流)。片段 AV 流通过规定的文件系统作为文件而记录在盘中。将该文件称作片段 AV 流文件(或 AV 流文件)。

[0062] 片段 AV 流文件是文件系统上的管理单位, 且是用户容易明白的管理单位, 但并不仅限于此。考虑到用户使用的方便性, 则需要将用于将分割成多个片段 AV 流文件的映像内容合成为一个加以再生的机制(mechanism)、用于只再生片段 AV 流文件的一部分的机制、以及用于顺利进行特殊再生、辅助文件再生(cue reproduction) 的信息等作为数据库预先记录在盘中。

[0063] 图 1 概略地表示被规定为可适用于本发明的 AVCHD 格式的数据模型。根据该 AVCHD 格式, 如图 1 所示, 数据结构由四层级的层构成。最下层级的层是配置片段 AV 流的层(为了方便而将其称作“片段层”)。其上的层是配置用于指定片段 AV 流的再生位置的、播放列表(PlayList) 和播放项目(PlayItem) 的层(为了方便而将其称作“播放列表层”)。再其上的层是配置影片夹对象(MovieObject) 等的层(为了方便将其称作“对象层”), 其中, 该影片夹对象由对播放列表指定再生顺序等的命令构成。最上层级的层配置有用于管理记录介质所存储的标题(title) 等的索引表(为了方便而将其称作“索引层”)。

[0064] 片段层的说明如下。片段 AV 流是视频数据和音频数据被多路复用为 MPEG2 TS(传输流) 的形式等的位流。与该片段 AV 流相关的信息被作为片段信息(Clip Information) 记录在文件中。

[0065] 另外, 可在片段 AV 流中多路复用 OB 流(Overlay Bitmapstream) 和 MB 流(Menu Bitmap stream), 其中, OB 流是显示字幕的图形流(graphic stream), MB 流将用于菜单显示等的信息(按钮图像数据(button image data) 等) 流形式化。

[0066] 片段 AV 流文件和记录有对应的片段信息的片段信息文件被视为归为一体的对象, 且被称作片段(Clip)。即, 片段是一个由片段 AV 流和片段信息构成的对象。

[0067] 一般而言, 文件被当作字节序(byte sequence) 处理。片段 AV 流文件的内容在时间轴上展开, 片段中的入口点(entry point) 主要在时基上指定。当存取点的时间戳被赋予规定的片段时, 片段信息文件可用于找到应在片段 AV 流文件中开始数据读出的地址信息。

[0068] 播放列表层的说明如下。播放列表由再生的 AV 流文件的指定、以及用于指定被指定了的 AV 流文件的再生位置的再生开始点(IN 点) 和再生结束点(OUT 点) 的集合构成。一组再生开始点和再生结束点的信息被称作播放项目(PlayItem)。播放列表由播放项目的集合构成。再生播放项目是指再生从其播放项目参照的 AV 流文件的一部分。即, 根据播放项目中的 IN 点和 OUT 点信息, 再生片段中的对应区间。

[0069] 对象层的说明如下。影片夹对象包含导航命令程序和与影片夹对象相关联的终端信息。导航程序是用于控制播放列表的再生的命令 (navigation command, 导航命令)。

[0070] 索引层的说明如下。索引层由索引表 (Index Table) 构成。索引表是定义记录介质中记录的内容的标题的顶层级 (top level) 的表。根据索引表中存储的标题信息, 由常驻播放器的系统软件中的模块管理 (module manager) 来控制记录介质的再生。

[0071] 即, 概略地如图 2 所示, 索引表中的任意入口被称为标题, 索引表中的入口的第一回放标题 (First PlaybackTitle)、菜单标题 (MenuTitle)、以及影片夹标题 (MovieTitle)#1、#2、..... 全部都是标题。各标题表示对影片夹对象的链接。

[0072] 为便于理解, 若以再生专用的记录介质为例, 则例如当存储在该记录介质中的内容是电影时, 第一回放标题对应于在电影正片之前放映的电影公司的宣传映像 (trailer)。菜单标题在内容例如是电影时, 对应于选择正片再生、章节搜索、字幕、语言设定、附赠影像 (bonus picture) 再生等的菜单画面。而且, 影片夹标题是从菜单标题选择的各个图像。标题还可以构成菜单画面。

[0073] 图 3 是表示上述的片段 AV 流、片段信息 (Stream Attributes)、片段、播放项目以及播放列表的关系的 UML (Unified Modeling Language, 统一建模语言) 图。播放列表对应于一个或多个播放项目, 播放项目对应于一个片段。对于一个片段, 其可以分别对应于开始点和 / 或结束点不同的多个播放项目。从一个片段参照一个片段 AV 流文件。同样, 从一个片段参照一个片段信息文件。并且, 片段 AV 流文件和片段信息文件具有一对一的对应关系。通过定义这样的结构, 从而便可以在无需更改片段 AV 流文件的情况下进行只再生任意部分的非破坏的再生顺序指定。

[0074] 而且, 如图 4 所示, 也可以从多个播放列表参照同一片段。并且, 还可以从一个播放列表指定多个片段。根据播放列表中的播放项目所示的 IN 点和 OUT 点来参照片段。在图 4 的例子中, 从播放列表 310 的播放项目 320 参照片段 300, 同时, 从构成播放列表 311 的播放项目 321 和 322 中的播放项目 321 参照以 IN 点和 OUT 点所示的区间。而且, 至于片段 301, 从播放列表 311 的播放项目 322 参照以 IN 点和 OUT 点所示的区间, 并且, 参照以播放列表 312 的播放项目 323 和 324 中的播放项目 323 的 IN 点和 OUT 点所示的区间。在图 4 的例子中, 还可以从其它的播放列表参照片段 301。

[0075] 接着, 使用图 5 来说明基于 AVCHD 格式的、记录在记录介质中的文件的管理结构。文件以目录结构被分层管理。在记录介质上首先创建一个目录 (图 5 所示的例子中为根 (root) 目录)。该目录下是由一个记录再生系统管理的范围。

[0076] 在根目录下设置目录“BDMV”。进一步根据需要, 在根目录下设置目录“AVCHDTN”。在目录“AVCHDTN”中设置例如将片段的代表图像缩小至规定尺寸的缩略文件 (thumbnail file)。在目录“BDMV”下存储使用图 1 说明的数据结构。

[0077] 可在目录“BDMV”下设置的文件只有两个文件、即文件“index.bdmv”以及文件“MovieObject.bdmv”。此外, 在目录“BDMV”下设置目录“PLAYLIST”、目录“CLIPINF”、目录“STREAM”以及目录“BACKUP”。目录“BACKUP”存储各目录及文件的备份。

[0078] 文件“index.bdmv”描述目录“BDMV”的内容。即, 该文件“index.bdmv”对应上述最上层级的层的索引层中的索引表。此外, 文件“MovieObject.bdmv”存储一个以上的影片夹对象的信息。即, 该文件“MovieObject.bdmv”对应上述的对象层。

[0079] 目录“PLAYLIST”是设置播放列表的数据库的目录。即，目录“PLAYLIST”包含文件“xxxxx.mpls”，其中，该文件“xxxxx.mpls”是与播放列表相关的文件。文件“xxxxx.mpls”是针对每个播放列表而创建的文件。在文件名中，“.”(period,点)前的“xxxxx”是5数位(digit)的数字，点后的“mpls”是这种类型文件的固定扩展名。

[0080] 目录“CLIPINF”是设置片段的数据库的目录。即，目录“CLIPINF”包含文件“zzzzz.clpi”，其中，该文件“zzzzz.clpi”是针对各片段AV流文件的片段信息文件。在文件名中，“.”(period,点)前的“zzzzz”是5数位的数字，点后的“clpi”是这种类型文件的固定的扩展名。

[0081] 目录“STREAM”是设置作为实体的AV流文件的目录。即，目录“STREAM”包含对应于各片段信息文件的片段AV流文件。片段AV流文件由MPEG2(Moving Pictures Experts Group 2,运动图像专家组)的传输流(以下简称为“MPEG2 TS”)构成，其文件名是“zzzzz.m2ts”。在文件名中，点前的“zzzzz”与对应的片段信息文件相同，从而可以便于把握片段信息文件和该片段AV流文件的对应关系。

[0082] 此外，目录“AVCHDTN”可以设置两种缩略文件“thumbnail.tidx”和“thumbnail.tdt2”。缩略文件“thumbnail.tidx”存储按规定方式加密的缩略图。缩略文件“thumbnail.tdt2”存储未加密的缩略图。例如，由于考虑到与用户通过摄像机拍摄到的片段相对应的缩略图是自由拷贝，即无需加密，所以将其存储在缩略文件“thumbnail.tdt2”中。

[0083] 下面，对在图5所示的各文件中，与本发明关系密切的文件作进一步的说明。首先，对目录“BDMV”下设置的文件“index.bdmv”进行说明。图6中示出表示该文件“index.bdmv”的一例结构的语法。在此，根据用作计算机装置等的编程语言的C语言来表示语法。关于这点，在其它的表示语法的图中也是同样。

[0084] 图6中，域TypeIndicator具有32位数据长度，表示该文件是索引表。域TypeIndicator2具有32位数据长度，表示文件“index.bdmv”的版本。域IndexesStartAddress具有32位数据长度，表示处于该语法内的块blkIndexes()的开始地址。

[0085] 域ExtensionDataStartAddress具有32位数据长度，表示处于该语法内的块blkExtensionData()的开始地址。块blkExtensionData()是可用于存储规定的扩展数据的块。域ExtensionDataStartAddress是通过以该文件“index.bdmv”的最初的字节为始的相对字节数来表示块blkExtensionData()的开始地址。相对字节数从“0”开始。如果该域ExtensionDataStartAddress的值为0，则表示在该文件“index.bdmv”内不存在块blkExtensionData()。

[0086] 接着域ExtensionDataStartAddress，设置有数据长度为192字节的区域reserved。此外，区域reserved是用于字节对齐(bytealignment)和以后追加域等的区域。关于这点，在以下的描述中也是同样。块blkAppInfoBDMV()是内容创建者能够描述任意信息的块，不会对播放器的动作等造成影响。

[0087] 块blkIndexes()是该文件“index.bdmv”的实质内容，通过描述在该块blkIndexes()中的内容来指定将盘装入播放器中时再生的第一回放、以及从主菜单(top menu)调出的标题(影片夹对象)等。根据由索引表调用的影片夹对象等描述的命令，读入后述的播放列表文件。

[0088] 图 7 中示出表示块 `blkIndexes()` 的一例结构的语法。域 `Length` 具有 32 位数据长度,表示从紧跟域 `Length` 之后直到该块 `blkIndexes()` 结束为止的数据长度。然后,在域 `Length` 之后设置块 `FirstPlaybackTitle()` 和块 `MenuTitle()`。

[0089] 块 `FirstPlaybackTitle()` 描述与在第一回放中使用的对象相关的信息。块 `FirstPlaybackTitle()` 接着具有 1 位数据长度的区域 `reserved`,且描述固定值“1”。此外,接着具有 31 位数据长度的区域 `reserved`,描述固定值“1”。接着,接着具有 14 位数据长度的区域 `reserved`,设置具有 16 位数据长度的域 `FirstPlaybackTitleMobjIDRef`。通过该域 `FirstPlaybackTitleMobjIDRef` 表示在第一回放标题中使用的影片夹对象的 ID。

[0090] 根据使用图 8 和图 9 后述的影片夹对象的语法,通过在影片夹对象的 `for` 循环语句中被用作循环变量的值 `mobj_id` 来表示影片夹对象的 ID。在该例中,域 `FirstPlaybackTitleMobjIDRef` 存储与参照的影片夹对象相对应的值 `mobj_id`。

[0091] 此外,块 `blkIndexes()` 中的块 `FirstPlaybackTitle()` 内的域 `FirstPlaybackTitleMobjIDRef` 既可以指主菜单的影片夹对象,也可以指标题。

[0092] 块 `MenuTitle(·)` 描述与主菜单中使用的对象相关的信息。块 `MenuTitle()` 接着具有 1 位数据长度的区域 `reserved`,被描述固定值“1”。此外,接着具有 31 位数据长度的区域 `reserved`,描述固定值“1”。并且,接着具有 14 位数据长度的区域 `reserved`,设置具有 16 位数据长度的域 `MenuTitleMobjIDRef`。域 `MenuTitleMobjIDRef` 表示在菜单标题中使用的影片夹对象的 ID。

[0093] 块 `MenuTitle()` 的后面的域 `NumberOfTitles` 具有 16 位数据长度,表示用户可选择、再生的标题数量。根据后面的 `for` 循环语句,以值 `title_id` 为自变量并按该域 `NumberOfTitles` 所示的次数来描述块 `MovieTitle[title_id]()`。块 `MovieTitle[title_id]()` 描述每个标题的信息。值 `title_id` 是从“0”直至域 `NumberOfTitles` 所示的值为止的数值,其用于识别标题。

[0094] 在块 `MovieTitle[title_id]()` 中,接着具有 1 位数据长度的区域 `reserved`,描述固定值 1,并且,接着具有 46 位数据长度的区域 `reserved`,描述域 `MovieTitleMobjIDRef`。域 `MovieTitleMobjIDRef` 具有 16 位数据长度,表示在该标题中使用的影片夹对象的 ID。域 `MovieTitleMobjIDRef` 的后面设置具有 32 位数据长度的区域 `reserved`。

[0095] 图 8 中示出表示设置在目录“BDMV”下的文件“`MovieObject.bdmv`”的一例结构的语法。域 `TypeIndicator` 具有 32 位(4 字节)的数据长度,表示该文件是文件“`MovieObject.bdmv`”。域 `TypeIndicator` 描述以 ISO(International Organization for Standardization, 国际标准化组织)646 标准的编码方式编码的 4 个字符构成的字符串。图 8 的例子中,在域 `type_indicator` 中描述以 ISO646 标准的方式编码的 4 个字符的字符串“MOBJ”,表示该文件是文件“`MovieObject.bdmv`”。

[0096] 域 `TypeIndicator2` 具有 32 位(4 字节)的数据长度,其表示文件“`MovieObject.bdmv`”的版本号。在该文件“`MovieObject.bdmv`”中,域 `TypeIndicator2` 必须是以 ISO646 标准的编码方式编码的 4 个字符的字符串“0100”。

[0097] 域 `ExtensionDataStartAddress` 具有 32 位数据长度,其表示处于该语法内的块 `blkExtensionData()` 的开始地址。域 `ExtensionDataStartAddress` 通过该文件“`MovieObject.bdmv`”的最初的字节开始的相对字节数来表示块 `blkExtensionData()` 的开

始地址。相对字节数从“0”开始。如果该域 ExtensionDataStartAddress 的值为“0”，则表示在该文件“MovieObject.bdmv”内不存在块 blkExtensionData()。

[0098] 此外，图 8 所示的语法内的域 padding_word 具有 16 位数据长度，根据文件“MovieObject.bdmv”的语法，在 for 循环语句中仅插入由值 N1 或值 N2 所表示次数。值 N1 或值 N2 是 0 或任意的正整数。并且，域 padding_word 可采用任意的值。

[0099] 接着域 ExtensionDataStartAddress，设置数据长度为 224 位的区域 reserved，其后存储文件“MovieObject.bdmv”的主体、即块 blkMovieObjects()。

[0100] 图 9 中示出表示块 blkMovieObjects() 的一例结构的语法。域 Length 具有 32 位数据长度，表示从紧跟域 Length 之后直到该块 blkMovieObjects() 结束为止的数据长度。接着具有 32 位数据长度的区域 reserved，设置域 NumberOfMobjs。域 NumberOfMobjs 表示根据其后的 for 循环语句而存储的影片夹对象的数量。通过作为 for 循环语句的循环变量而使用的值 mobj_id 来唯一指定影片夹对象。值 mobj_id 是从“0”开始的值，影片夹对象以 for 循环语句中描述的顺序定义。

[0101] for 循环语句中的块 TerminalInfo() 描述固定值“1”，接着设置具有 15 位数据长度的区域 reserved。接下来设置具有 16 位数据长度的域 NumberOfNavigationCommands[mobj_id]。该域 NumberOfNavigationCommands[mobj_id] 表示在由值 mobj_id 指示的影片夹对象 MovieObject[mobj_id]() 中所包括的导航命令 (NavigationCommand) 的数量。

[0102] 通过后续的、将值 command_id 作为循环变量的 for 循环语句，仅用域 NumberOfNavigationCommands[mobj_id] 所示的数量来描述导航命令。即，设置在该 for 循环语句中的域 NavigationCommands[mobj_id][command_id] 存储在由值 mobj_id 指示的块 MovieObject[mobj_id]() 中所包括的、以值 command_id 所示的顺序的导航命令 NavigationCommand。值 command_id 是从 0 开始的值，导航命令 NavigationCommand 以 for 循环语句中描述的顺序被定义。

[0103] 图 10 中示出表示播放列表文件“xxxxx.mpls”的一例结构的语法。域 TypeIndicator 具有 32 位 (4 字节) 的数据长度，表示该文件是播放列表文件。域 TypeIndicator2 具有 32 位 (4 字节) 的数据长度，其表示该播放列表文件的版本。域 PlaylistStartAddress 具有 32 位的数据长度，其表示该语法中的块 blkPlaylist() 的开始地址。

[0104] 域 PlaylistMarkStartAddress 具有 32 位数据长度，其表示该语法中的块 blkPlaylistMark() 的开始地址。域 ExtensionDataStartAddress 具有 32 位数据长度，其表示该语法中的块 blkExtensionData() 的开始地址。域 ExtensionDataStartAddress 是将块 blkExtensionData() 的开始地址表示作为从文件“xxxxx.mpls”的最初的字节开始的相对字节数的值。相对字节数从“0”开始。如果该域 ExtensionDataStartAddress 的值为“0”，则表示在该文件“xxxxx.mpl”内不存在块 blkExtensionData()。

[0105] 接着具有 160 位数据长度的区域 reserved，设置块 blkAppInfoPlaylist()。块 blkAppInfoPlaylist() 描述后面的块 blkPlaylist() 中描述的播放列表的类型、再生限制等信息。块 blkPlaylist() 描述播放列表。块 blkPlaylistMark() 描述在章节转移 (chapter jump) 等中被转移的点。块 blkExtensionData() 是用于可存储规定扩展数据的块。

[0106] 此外,该图 10 所示语法内的域 padding_word 具有 16 位数据长度,根据文件“xxxxx.mpls”的语法,在 for 循环语句中仅插入由值 N1、值 N2 及值 N3 所示次数。值 N1、值 N2 或值 N3 是 0 或任意的正整数。并且,域 padding_word 可采用任意的值。

[0107] 图 11 中示出表示块 blkPlayList() 的一例结构的语法。域 Length 具有 32 位数据长度,其表示从紧跟域 Length 之后直到该块 blkPlayList() 结束为止的数据长度。接着域 Length,配置有具有 16 位数据长度的区域 reserved,然后配置域 NumberOfPlayItems。域 NumberOfPlayItems 具有 16 位数据长度,其表示包含在该块 blkPlayList() 中的播放项目的数量。域 NumberOfSubPath 表示包含在该块 blkPlayList() 中的子路径的数量。

[0108] 根据后续的 for 循环语句,通过域 NumberOfPlayItems 所示的数量来描述块 blkPlayItem(),其中,该块 blkPlayItem() 描述播放项目。基于 for 循环语句的计数数量变为块 blkPlayItem() 的标识符 PlayItem_id。然后,根据后续的 for 循环语句,通过域 NumberOfSubPath 所示的数量来描述块 blkSubPath()。基于 for 循环语句的计数数量变为块 blkSubPath() 的标识符 SubPath_id。

[0109] 此外,相对于与被主要再生的播放项目相对应的主路径,可以与子播放项目相对应地具有子路径。子路径例如可以用于指定后期录音(after recording)用的音频数据、以及在合成两个映像时指定与播放项目中指定的片段同步再生的副映像。

[0110] 图 12 中示出表示块 blkPlayItem() 的一例结构的语法。域 Length 具有 16 位数据长度,表示从紧跟该域 Length 之后直到该块 blkPlayItem() 结束为止的数据长度。

[0111] 域 ClipInformationFileName 具有 40 位(5 字节)的数据长度,表示该块 blkPlayItem() 参照的片段信息文件的文件名。在该播放项目中,读出域 ClipInformationFileName[0] 所示文件名的片段信息文件。域 ClipCodeIdentifier[0] 具有 32 位(4 字节)的数据长度,表示该块 blkPlayItem() 中的播放项目所使用的片段 AV 流的编码方式(codec)。

[0112] 接着具有 12 位数据长度的区域 reserved,设置域 ConnectionCondition。域 ConnectionCondition 具有 4 位数据长度,表示与片段间的连接状态有关的信息。对于记录用途的记录介质,采用“1”、“5”或“6”作为域 ConnectionCondition 的值。若域 ConnectionCondition 的值是“1”,则表示从当前播放项目参照的片段与从前一个播放项目参照的片段没有无缝连接(seamless),若域 ConnectionCondition 的值是“5”或“6”,则表示从当前播放项目参照的片段与从前一个播放项目参照的片段是无缝连接。此外,所谓的无缝连接是指进行片段间的再生控制,以便使得片段与下一片段在帧定时(frame timing)被连续地再生。

[0113] 若域 ConnectionCondition 的值是“5”,则在当前播放项目参照的片段中,音频数据的记录长度比视频数据的记录长度长(参照图 13A)。由此,当连接片段和片段时,可进行音频数据的淡出处理(fade-out process)。例如,当片段由于用户的记录停止操作而被关闭(close)时,域 ConnectionCondition 的值被设定为“5”。下面,将域 ConnectionCondition 的值被表示为“5”的片段的连接方法称作第一无缝连接。

[0114] 若域 ConnectionCondition 的值是“6”,则在该播放项目参照的片段中,音频数据的记录长度与视频数据的记录长度相同(参照图 13B)。由此,可无缝连接片段和片段。例如,当片段由于除与用户的操作相对应的记录停止之外的原因、例如基于系统因素而关闭

片段时,域 ConnectionCondition 的值被设定为“6”。下面,将域 ConnectionCondition 的值被示为“6”的片段的连接方法称作第二无缝连接。

[0115] 域 RefToSTCID[0] 具有 8 位数据长度,表示与系统时基 (STC) 的不连续点有关的信息。域 INTime 及域 OUTTime 分别具有 32 位数据长度,表示主片段 AV 流的再生范围。域 INTime 表示开始点 (IN 点),域 OUTTime 表示结束点 (OUT 点)。

[0116] 块 blkUOMaskTable() 是设定有用户输入的受理限制的表。具有 1 位数据长度的标签 PlayItemRandomAccessFlag 规定是否允许对该块 blkPlayItem() 的播放项目进行随机存取。接下来,接着具有 7 位数据长度的区域 reserved,设置域 StillMode。域 StillMode 具有 8 位数据长度,其表示是否在块 blkPlayItem() 的播放项目中,将最后显示的映像显示作为静态图像。如果域 StillMode 的值是“0x01”(二进制),则根据 if 语句,通过具有 16 位数据长度的域 StillTime 表示静止时间。如果域 StillMode 的值是“0x01”以外的值,则该具有 16 位数据长度的区域被作为区域 reserved。

[0117] 此外,关于数值的表述,“0x”表示其数值为十六进制的表述。对于这点,也适用于下面的同样的表述。

[0118] 块 blkSTNTable() 管理该块 blkPlayItem() 的播放项目所管理的片段 AV 流的属性、PID 编号、记录介质上的记录位置等。

[0119] 图 14 中示出表示块 blkPlayListMark() 的一例的结构的语法。域 Length 具有 32 位数据长度,表示从紧跟域 Length 之后直到该块 blkPlayList Mark() 结束为止的数据长度。

[0120] 域 NumberOfPlayListMarks 具有 16 位数据长度,其表示在该块 blkPlayList Mark() 中包括的播放列表标签的数量。根据后面的 for 循环语句,仅以域 NumberOfPlayListMarks 所表示的数量来描述播放列表标签 (mark) 的信息。

[0121] 在 for 循环语句内,接着具 8 位数据长度的区域 reserved,设置域 MarkType。域 MarkType 具有 8 位数据长度,表示标签的类型。在播放列表的标签中定义有入口标签 (Entry Mark) 和连接点 (LinkPoint) 两种类型,通过该域 MarkType 来表示是哪种类型。使用入口标签来定义章节。连接点与本发明关联较小,故省略对其的说明。上述的域 NumberOfPlayListMarks 表示入口标签和连接点的总共值。

[0122] 域 RefToPlayItemID 具有 16 位数据长度,描述参照打标签的播放项目的识别信息 PlayItem_id。域 MarkTimeStamp 具有 32 位数据长度,描述表示打标签的点的时戳。域 EntryESPID 具有 16 位数据长度,表示含有通过标签指示的基本流 (elementary stream) 的 TS 包的 PID 值。域 Duration 是基于以 45kHz 时钟为单位的计测的、具有 32 位数据长度的无符号的整数。如果存储在该域 Duration 中的值是“0”,则该域 Duration 没有任何意义。

[0123] 图 15 中示出表示片段信息文件的一例结构的语法。域 TypeIndicator 具有 32 位 (4 字节) 的数据长度,表示该文件是片段信息文件。域 TypeIndicator2 具有 32 位 (4 字节) 的数据长度,表示该片段信息文件的版本。

[0124] 该片段信息文件包括块 blkClipInfo()、块 blkSequenceInfo()、块 blkProgramInfo()、块 blkCPI()、块 blkClipMark() 及块 blkExtensionData(), 分别具有 32 位数据长度的域 SequenceInfoStartAddress、域 ProgramInfoStartAddress、域 CPIStartAddress、域 ClipMarkStartAddress 和域 ExtensionDataStartAddress 表示各自

对应的块的开始地址。

[0125] 域 ExtensionDataStartAddress 是从该片段信息文件的最初的字节开始的相对字节数,其表示块 blkExtensionData() 的开始地址。相对字节数从“0”开始。如果该域 ExtensionDataStartAddress 的值为“0”,则表示在该文件“index.bdmv”内不存在块 blkExtensionData()。

[0126] 块 blkClipInfo() 后续表示这些表示开始地址的域,且从具有 96 位数据长度的区域 reserved 之后开始。块 blkClipInfo() 描述与该片段信息文件所管理的片段 AV 流有关的信息。块 blkSequenceInfo() 描述将具有 STC 和 ATC(arrival time base,到达时基)连续的序列统一管理的信息。块 blkProgramInfo() 描述在该片段信息文件中被管理的片段 AV 流的编码方式、片段 AV 流中的视频数据的屏幕高宽比(aspect ratio)等信息。块 blkCPI() 存储与特征点信息 CPI 相关的信息,其中,特征点信息 CPI 表示随机存取开始点等的、AV 流中的特征位置。

[0127] 块 blkClipMark() 描述章节位置等的、附加给片段中用于再生辅助文件的索引点(跳跃点(jump point))。块 blkExtensionData() 是能够存储扩展数据的区域。此外,由于这些块 blkClipMark() 和片段信息文件内的块 blkExtensionData() 与本发明的关联较小,故省略对其的详细说明。

[0128] 图 16 中示出表示块 blkClipInfo() 的一例结构的语法。域 Length 具有 32 位数据长度,其表示从紧跟域 Length 之后直到该块 blkClipInfo() 结束为止的数据长度。接着具有 16 位数据长度的区域 reserve,设置域 ClipStreamType。

[0129] 域 ClipStreamType 具有 8 位数据长度,表示片段 AV 流的种类。域 ClipStreamType 的值例如被固定为“1”。域 ApplicationType 具有 8 位数据长度,表示片段 AV 流(扩展名为“m2ts”的文件)是通过怎样的多路复用而制作的。若域 ApplicationType 的值是“1”,则对应的片段 AV 流再生作为通常的动态图像。接着,设置具有 31 位数据长度的区域 reserved。

[0130] 数据长度为 1 位的标记(flag)IsCC5 通过播放列表中的块 blkPlayItem(),表示是否通过上述的第一无缝连接、即域 ConnectionCondition 的值被表示为“5”的方法来进行对应的片段和下一片段的连接。如果标记 IsCC5 的值是“1”(二进制值),则通过第一无缝连接来进行片段间的连接。

[0131] 域 TSRecordingRate 用字节/秒表示片段 AV 流文件的记录速度。域 NumberOfSourcePackets 表示在片段 AV 流中包括的源包(sourcepacket)数量。接着数据长度是 1024 位的区域 reserved,设置块 TSTypeInfoBlock()。块 TSTypeInfoBlock() 存储表示存储有片段 AV 流的包的类型的信息。由于该块 TSTypeInfoBlock() 与本发明的关联较小,故省略对其的详细说明。

[0132] 在上述的标签 IsCC5 的值是“1”的情况下,描述后面的 if 语句以后的信息。接着 if 语句后面的、具 8 位数据长度的区域 reserved,设置域 FollowingClipStreamType,域 FollowingClipStreamType 具有 8 位数据长度,其描述与该片段信息文件相对应的片段的下一片段的类型。接着具有 32 位数据长度的区域 reserved,设置域 FollowingClipInformationFileName。

[0133] 域 FollowingClipInformationFileName 具有 40 位(5 字节)的数据长度,其描述与该片段信息文件相对应片段的下一片段相对应的片段信息文件的文件名。后面的域

ClipCodecIdentifier 具有 32 位 (4 字节) 的数据长度,其表示该下一片段的编码方式。在本实施例中,域 ClipCodecIdentifier 被固定为通过 ISO646 标准的方式编码的 4 字符的字符串值“M2TS”。然后,设置具有 8 位数据长度的区域 reserved。

[0134] 图 17 中示出表示块 blkSequenceInfo() 的一例结构的语法。域 Length 具有 32 位数据长度,其表示从紧跟域 Length 之后直到该块 blkSequenceInfo() 结束为止的数据长度。接着具有 15 位数据长度的区域 reserved,以 1 位的数据长度来描述固定值“1”。

[0135] 后面的域 SPNATCStart 具有 32 位数据长度,其通过包编号来表示序列 (称为序列 ATCSequence) 的开始,其中,序列表示在连续的时间中记录数据。在图 17 所示的例子中,域 SPNATCStart 具有值“0”,其表示序列的开始与片段 AV 流文件的开始相一致。域 NumberOfSTCSequence 表示序列 ATCSequence 上的序列 STCSequence 的数量。域 NumberOfSTCSequence 的值在“1”以上。

[0136] 根据后面的 for 循环语句,仅通过域 NumberOfSTCSequence 所示的数量来描述序列 STCSequence 的信息。序列 STCSequence 表示作为 MPEG2 TS (Transport Stream, 传输流) 中的时间轴的基准的 PCR (Program Clock Reference, 程序时钟参考) 连续的范围。对序列 STCSequence 分配在片段内唯一的编号 stc_id。在序列 STCSequence 内,因为能够定义不间断的连续时间轴,所以能够唯一确定播放项目的开始时刻和结束时刻。即,各播放项目的开始点和结束点必须存在于同一序列 STCSequence 中。在该 for 循环语句中,通过值 stc_id 指定序列 STCSequence。

[0137] 域 PCRPID[stc_id] 具有 16 位数据长度,表示在 MPEG2TS 中含有 PCR (Program Clock Reference, 程序时钟参考) 的 TS 包的 PID。域 SPNSTCStart[stc_id] 具有 32 位数据长度,用包编号表示序列 STCSequence 的开始。域 PresentationStartTime 和域 PresentationEndTime 分别具有 32 位的数据长度,表示片段 AV 流中的有效范围。域 PresentationStartTime 和域 PresentationEndTime 所示的范围是可以从播放项目参照的范围。

[0138] 图 18 中示出表示块 blkProgramInfo() 的一例结构的语法。域 Length 具有 32 位数据长度,表示从紧跟域 Length 之后直到该块 blkProgramInfo() 结束为止的数据长度。接着具有 15 位数据长度的区域 reserved,通过 1 位的数据长度来描述固定值“1”。

[0139] 域 SPNprogramSequenceStart 具有 32 位数据长度,其描述在对应的片段 AV 流文件中开始程序序列的源包的编号。域 ProgramMapPID 具有 16 位数据长度,表示 TS 包的 PID 值,其中,TS 包被当作包含可适用于程序序列中的程序映射部 (program mapsection)。域 NumberOfStreamsInPS 具有 8 位的数据长度,表示在程序序列中被定义的基本流的数量。接着域 NumberOfStreamsInPS,设置具 8 位数据长度的区域 reserved。

[0140] 根据后面的 for 循环语句,将值 [stream_index] 作为循环变量,并以域 NumberOfStreamsInPS 所表示的数量来存储域 StreamPID[stream_index] 和块 blkStreamCodingInfo(stream_index) 的组。域 StreamPID[stream_index] 表示被程序序列参照的 PMT (Program Map Table, 程序映射表) 中描述的基本流相对应的 PID 的值。下一块 blkStreamCodingInfo(stream_index) 描述与对应的域 StreamPID[stream_index] 所示的基本流的编码方式相关的信息。

[0141] 图 19 中示出表示块 blkCPI() 的一例结构的语法。在 MPEG 流这样的、进行帧间压

缩的编码流中,可开始译码的位置多被限制在诸如 GOP(Group OfPicture,图像组)的开头等一部分的位置上。对于 CPI(Characteristic Point Information,特征点信息),其是收集可译码开始点的位置信息的数据库,且是使再生时刻和文件内地址相对应的表。即,CPI将表示译码单位的开头位置的信息进行制表。

[0142] 通过这样地确定数据库,从而在例如想从任意时刻开始再生时,都能够通过基于再生时刻并参照 CPI 来知晓再生位置的文件内地址。该地址是译码单位的开头,所以播放器能够从该位置读出数据并译码,从而迅速地显示图像。

[0143] 此外,将该 CPI 中存储的、译码单位的开头位置(本例中为 GOP 的开头位置)称为 EP(Entry Point,入口点)入口。

[0144] 在图 19 中,域 Length 具有 32 位数据长度,表示从紧跟该域 Length 之后直到块 blkCPI() 结束为止的数据长度。根据后面的 if 语句,当域 Length 的值不是“0”时,接着具有 12 位数据长度的区域 reserved,设置域 CPIType。域 CPIType 具有 4 位数据长度,表示 CPI 的种类。后面的块 blkEMap() 存储用于使对应的片段 AV 流文件中的 PTS 值和字节地址相关联的表。

[0145] 图 20 中示出表示块 blkEMap() 的一例结构的语法。接着具有 8 位数据长度的区域 reserved,设置域 NumberOfStreamPIDEntries。域 NumberOfStreamPIDEntries 具有 8 位数据长度,表示块 blkEMap() 中的块 blkEMapForOneStreamPID 的入口数量。根据 for 循环语句,将值 [k] 作为循环变量,并以域 NumberOfStreamPIDEntries 中所示的数量来描述入口点相关的信息。

[0146] 在 for 循环语句内,域 StreamPID[k] 具有 16 位数据长度,表示用于传输通过块 blkEMapForOneStreamPID 参照的基本流的传输包的 PID 值,其中,块 blkEMapForOneStreamPID 是块 blkEMap() 中的第 [k] 个入口(以下表述为第 [k] 块 blkEMapForOneStreamPID)。

[0147] 接着具有 10 位数据长度的区域 reserved,设置域 EPStreamType[k]。域 EPStreamType[k] 具有 4 位的数据长度,表示通过第 [k] 块 blkEMapForOneStreamPID 参照的基本流的类型。域 NumberOfEPCoarseEntries[k] 具有 16 位数据长度,表示位于第 [k] 块 blkEMapForOneStreamPID 中的用于粗检的子表(EP coarse table,EP 粗检表)的入口数量。域 NumberOfEPFineEntries[k] 具有 18 位数据长度,表示位于第 [k] 块 blkEMapForOneStreamPID 中的用于精检的子表(EP fine table,EP 精检表)的入口数量。域 EMapForOneStreamPIDStartAddress[k] 具有 32 位数据长度,表示块 blkEMap() 中第 [k] 块 blkEMapForOneStreamPID 开始的相对字节位置。该值通过从块 blkEMap() 的第 1 字节开始的字节数加以表示。

[0148] 在上述的 for 循环语句的描述之后,根据夹着具有 16 位的整数倍的数据长度的附加字而描述的 for 循环语句,将值 [k] 作为循环变量,并仅以域 NumberOfStreamPIDEntries 所示的数量来存储块 blkEMapForOneStreamPID(EPStreamType[k]、NumberOfEPCoarseEntries[k]、NumberOfEPFineEntries[k])。即,自变量 NumberOfEPCoarseEntries[k] 表示在子表(EP coarse table,EP 粗检表)中存储的入口 PTSEPCoarse 和入口 SPNEPCoarse 的数量。同样,自变量 NumberOfEPFineEntries[k] 表示在子表(EP fine table,EP 精检表)中存储的入口 PTSEPFine 和入口 SPNEPFine 的数量。

下面,将自变量 NumberOfEPCoarseEntries[k] 和自变量 NumberOfEPFineEntries[k] 分别适当地称作入口数 Nc 和入口数 Nf。

[0149] 图 21 中示出表示块 blkEMapForOneStreamPID(EP_streamtype, Nc, Nf) 的一例结构的语法。为了说明块 blkEMapForOneStreamPID(EP_stream_type, Nc, Nf) 的语义(semantics),首先对作为在块 blkEMapForOneStreamPID(EPstream_type, Nc, Nf) 中存储的数据源的入口的、入口 PTSEPStart 和入口 SPNEPStart 的意义进行说明。

[0150] 入口 PTSEPStart 和与入口 PTSEPStart 相关联的入口 SPNEPStart 分别指 AV 流上的入口点。而且,入口 PTSEPFine 和与入口 PTSEPFine 相关联的入口 PTSEPCoarse 通过同一入口 PTSEPStart 导出。并且,入口 SPNEPFine 和与入口 SPNEPFine 相关联的入口 SPNEPCoarse 通过同一入口 SPNEPStart 导出。

[0151] 图 22 中示出入口 PTSEPCoarse 和入口 PTSEPFine 的一例格式。PTS、即入口 PTSEPStart 是数据长度为 33 位的值。当将 MSB 的位设为第 32 位,将 LSB 的位设为第 0 位时,在图 22 的例子中,以粗略的单位进行检索时所使用的入口 PTSEPCoarse 采用从入口 PTSEPStart 的第 32 位到第 19 位为止的 14 位。通过入口 PTSEPCoarse,从而可在分辨率为 5.8 秒至 26.5 个小时为止的范围内进行检索。此外,用于进行更精确检索的入口 PTSEPFine 采用从入口 PTSEPStart 的第 19 位到第 9 位为止的 11 位。通过入口 PTSEPFine,从而可在分辨率为 5.7 毫秒至 11.5 秒为止的范围内进行检索。此外,入口 PTSEPCoarse 和入口 PTSEPFine 共用第 19 位。另外,从 LSB 侧的第 0 位至第 8 位为止的 9 位没有被使用。

[0152] 图 23 中示出的是入口 SPNEPCoarse 和入口 SPNEPFine 的一例格式。源包编号、即入口 SPNEPStart 是数据长度为 32 位的值。当将 MSB 的位作为第 31 位,并将 LSB 的位作为第 0 位时,在图 23 的例子中,当通过粗略的单位进行检索时所使用的入口 SPNEPCoarse 使用从入口 SPNEPStart 的第 31 位到第 0 位为止的所有位。而且,用于进行更精确检索的入口 SPNEPFine 使用从入口 SPNEPStart 的第 16 位到第 0 位为止的 17 位。通过入口 SPNEPFine,从而可在例如大致 25MB(Mega Byte,兆字节)的 AV 流文件为止的范围内进行检索。

[0153] 此外,在源包编号的情况下,作为入口 SPNEPCoarse,也可以仅使用 MSB 侧的规定位数的值。例如,作为入口 SPNEPCoarse,可使用从入口 SPNEPStart 的第 31 位至第 16 位为止的 17 位,而入口 SPNEPFine 可使用从入口 SPNEPStart 的第 16 位至第 0 位为止的 17 位。

[0154] 综上所述,入口 PTSEPStart 和入口 SPNEPStart 如下所述被定义。

[0155] 如图 22 所示,入口 PTSEPStart 是数据长度为 33 位的无符号整数,其表示在 AV 流中,从可随机存取的图片 (picture) (例如, IDR (Instantaneous Decoding Refresh, 即时译码刷新) 图片和 I (Intra, 内部) 图片) 开始的视频存取单元的 33 位数据长度的 PTS。

[0156] 如图 23 所示,入口 SPNEPStart 是 32 位的无符号整数,表示源包在 AV 流中的地址,其中,源包包含与入口 PTSEPStart 相关联的视频存取单元的第一字节。入口 SPNEPStart 以源包编号的单位表示,从 AV 流文件中的最初的源包开始计数,以值“0”为初始值,每增加一个源包其值就加 1。

[0157] 参照图 21,块 blkEMapForOneStreamPID(EP_stream_type, Nc, Nf) 通过第一 for 循环语句描述用于以粗略的单位进行检索的子表 (EPcoarse table, EP 粗检表),通过第二 for 循环语句描述用于根据子表 (EP coarse table, EP 粗检表) 的检索结果,进行更为详细检索的子表 (EP fine table, EP 精检表)。

[0158] 在紧靠第一 for 循环语句的前面, 设置域 EPFineTableStartAddress。域 EPFineTableStartAddress 具有 32 位数据长度, 通过从块 blkEMapForOneStreamPID(EP_stream_type, Nc, Nf) 的第一字节开始的相对字节数来表示最初的第二 for 循环语句中的域 ReservedEPFine[EP_fine_id] 的第一字节的开始地址。相对字节数从值“0”开始。

[0159] 第一 for 循环语句通过循环变量 [i], 反复直至子表 (EP_coarsetable, EP 粗检表) 的入口数 Nc, 第一 for 循环语句存储有入口数 Nc 所代表的组数的域 RefToEPFineID[i]、入口 PTSEPCoarse[i]、及入口 SPNEPCoarse[i]。在第一 for 循环语句中, 域 RefToEPFineID[i] 具有 18 位数据长度, 并具有与域 PTSEPCoarse[i] 所表的入口 PTSEPCoarse 相关联的入口 PTSEPFine, 其中, 域 PTSEPCoarse[i] 后续在域 RefToEPFineID[i] 之后, 域 RefToEPFineID[i] 表示子表 (EPfine table, EP 精检表) 内的入口编号。入口 PTSEPFine 和与该入口 PTSEPFine 相关联的入口 PTSEPCoarse 从同一入口 PTSEPStart 被导出。通过以第二 for 循环语句中描述的顺序定义的循环变量 [EPfine_id] 的值来赋予域 RefToEPFineID[i]。

[0160] 在第一 for 循环语句之后, 隔着附加字而进行的是第二 for 循环语句的描述。第二 for 循环语句通过循环变量 [EP_fine_id], 反复直至子表 (EP_fine table, EP 精检表) 的入口数 Nf, 第二 for 循环语句存储有入口数 Nf 所代表的组数的、具有 1 位数据长度的域 ReservedEPFine[EP_fine_id]、具有 3 位数据长度的域 IEndPositionOffset[EP_fine_id]、具有 11 位数据长度的域 PTSEPFine[EP_fine_id]、和具有 17 位数据长度的域 SPNEPFine[EP_fine_id]。在这些域中, 域 PTSEPFine[EP_fine_id] 和域 SPNEPFine[EP_fine_id] 分别存储根据循环变量 [EP_fine_id], 从子表 (EP_fine table, EP 精检表) 参照的入口 PTSEPFine 和入口 SPNEPFine。

[0161] 如下所述地导出入口 PTSEPCoarse 和入口 PTSEPFine、以及入口 SPNEPCoarse 和入口 SPNEPFine。以在子表 (EP_fine table, EP 精检表) 中存在按相关的数据 SPNEPStart 的值的升序排列的 Nf 个入口为例。各个入口 PTSEPFine 按下式 (1) 从对应的入口 PTSEPStart 被导出。

[0162] $PTSEPFine[EP_fine_id] = (PTSEPStart[EP_fine_id] \gg 9) / 2^{11} \dots (1)$

[0163] 入口 PTSEPCoarse 和对应的入口 PTSEPFine 的关系如下式 (2)、(3) 所示。

[0164] $PTSEPCoarse[i] = (PTSEPStart[RefToEPFineID[i]] \gg 19) / 2^{14} \dots (2)$

[0165] $PTSEPFine[RefToEPFineID[i]] = (PTSEPStart[RefToEPFineID[i]] \gg 9) / 2^{11} \dots (3)$

[0166] 各个入口 SPNEPFine 按下式 (4) 从对应的入口 SPNEPStart 被导出。

[0167] $SPNEPFine[EP_fine_id] = SPNEPStart[EP_fine_id] / 2^{17} \dots (4)$

[0168] 入口 SPNEPCoarse 和对应的入口 SPNEPFine 的关系如下式 (5)、(6) 所示。

[0169] $SPNEPCoarse[i] = SPNEPStart[RefToEPFineID[i]] \dots (5)$

[0170] $SPNEPFine[RefToEPFineID[i]] = SPNEPStart[RefToEPFineID[i]] / 2^{17} \dots (6)$

[0171] 在上述式 (1) ~ 式 (6) 中, 符号 “ $\gg x$ ” 表示使用具有超过数据的 LSB 侧的 x 位的数位的位。

[0172] 接下来, 对用于存储扩展数据的块 blkExtensionData() 进行说明。块 blkExtensionData() 被定义为可存储规定的扩展数据, 并能够被描述在存储有索引表的文

件“index.bdmv”、存储有播放列表的文件“xxxxx.mpls”以及片段信息文件“zzzzz.clpi”各文件中。

[0173] 图 24 中示出表示块 blkExtensionData() 的一例的结构的语法。域 Length 具有 32 位数据长度,以字节数表示从紧跟该域 Length 之后直到块 blkExtensionData() 结束为止的数据长度。如果该域 Length 所示的数据长度为“0”,则进行 if 语句后面的描述。

[0174] 域 DataBlockStartAddress 具有 32 位数据长度,其通过从该块 blkExtensionData() 的开头字节开始的相对字节数来表示存储有该语法中的、扩展数据本体的块 DataBlock() 的开始地址。即,相对字节数从“0”开始。此外,域 DataBlockStartAddress 必须满足如下所示的 32 位对齐(alignment)的条件。

[0175] $\text{DataBlockStartAddress} \% 4 = 0$

[0176] 接着具有 24 位数据长度的区域 reserved,设置域 NumberOfExtDataEntries。域 NumberOfExtDataEntries 具有 8 位数据长度,表示存储在该块 blkExtensionData() 的块 DataBlock() 中的扩展数据的入口数量。扩展数据的入口存储有用于获取扩展数据主体的信息。在本实施例中,扩展数据的入口是由域 ExtDataType、域 ExtDataVersion、域 ExtDataStartAddress、及域 ExtDataLength 组成的块 ext_data_entry(),在块 blkExtensionData() 中,根据第一 for 循环语句,存在由该域 NumberOfExtDataEntries 所示的个数的该块 ext_data_entry()。

[0177] 域 ExtDataType 具有 16 位数据长度,表示在该块 blkExtensionData() 中描述的扩展数据是记录装置用的扩展数据。域 ExtDataType 的值是用于识别扩展数据的第一值,其可被定义为由包含该块 blkExtensionData() 的规格书的许可方(licence)(合法使用者)来分配域 ExtDataType 的值。域 ExtDataVersion 是用于识别扩展数据的第二值,其可被定义为表示该扩展数据的版本号。此外,在块 blkExtensionData() 中,必须存在两个以上的域 ExtDataType 和域 ExtDataVersion 的值相同的块 ext_data_entry()。

[0178] 域 ExtDataStartAddress 具有 32 位数据长度,表示与含有该域 ExtDataStartAddress 的扩展数据的入口(块 ext_data_entry())相对应的扩展数据的开始地址。域 ExtDataStartAddress 是以从块 blkExtensionData() 的开头字节开始的相对字节数来表示扩展数据 ext_data 的开始地址。此外,域 ExtDataStartAddress 必须满足如下所示的 32 位对齐的条件。

[0179] $\text{ExtDataStartAddress} \% 4 = 0$

[0180] 域 ExtDataLength 具有 32 位的数据长度,表示与含有该域 ExtDataStartAddress 的扩展数据的入口(块 ext_data_entries())相对应的扩展数据的数据长度。数据长度用字节数表示。

[0181] 如果描述由域 NumberOfExtDataEntries 所示个数的扩展数据的入口(块 ext_data_entry()),则分别具有 16 位数据长度并由任意数据序列(data sequence)组成的附加字 padding_word 以两个域作为一组,重复任意的次数 L1。然后,描述存储有扩展数据主体的块 DataBlock()。块 DataBlock() 存储 1 个以上的扩展数据。根据上述的域 ExtDataStartAddress 和域 ExtDataLength,从块 DataBlock() 中提取各扩展数据 ext_data。

[0182] 图 25 模式地示出了块 blkExtensionData() 中的各数据的参照关系。通过

域 Length, 表示从紧跟该域 Length 之后的位置直到块 blkExtensionData() 结束为止的数据长度。通过域 DataBlockStartAddress, 表示块 DataBlock() 的开始位置。以域 NumberOfExtDataEntries 所示的个数来描述块 ext_data_entry()。从最后的块 ext_data_entry() 到块 DataBlock() 之间以任意的长度设置域 padding_word。

[0183] 在块 DataBlock() 内设置有通过块 ext_data_entry() 所示的扩展数据 ext_data。由对应的块 ext_data_entry() 内的域 ExtDataStartAddress 和域 ExtDataLength 表示各扩展数据 ext_data 的位置和数据长度。因此, 块 DataBlock() 内的扩展数据 ext_data 的排列顺序也可以不与对应的块 ext_data_entry() 的排列顺序一致。

[0184] 这样, 通过使扩展数据具有块 DataBlock() 和块 ext_data_entry() 的双层结构, 从而能够存储多个扩展数据, 其中, 块 DataBlock() 存储有扩展数据主体, 块 ext_data_entry() 存储有对块 DataBlock() 内的扩展数据进行存取信息等。

[0185] 接下来, 对上述扩展数据的一例的创建方法和读出方法进行说明。图 26 是表示将数据写入块 blkExtensionData() 中时的一例处理的流程图。图 26 是将扩展数据追加作为块 blkExtensionData() 中的第 (n+1) 个入口, 并重写块 blkExtensionData() 时的示例。

[0186] 首先, 在步骤 S10 中, 获取想要写入的扩展数据的数据长度, 并设置为域 ExtDataLength[n+1] 的值。此外, “[n+1]” 的表述对应于第 (n+1) 个入口的编号。接着, 在步骤 S11 中, 调查当前块 blkExtensionData() 中列出的块 ext_data_entry() 的域 ExtDataLength 和域 ExtDataStartAddress 的值, 并获得块 DataBlock() 的使用情况。

[0187] 并且, 在接下来的步骤 S12 中, 判断在块 DataBlock() 中是否存在大于等于想要写入的扩展数据的数据长度、即域 ExtDataLength[n+1] 中所示的数据长度的连续的空区域。如果判断的结果是存在这样的空区域, 则进入步骤 S14 中的处理。

[0188] 另一方面, 如果判断的结果是不存在大于等于域 ExtDataLength[n+1] 中所示的数据长度的连续的空区域, 则进入步骤 S13 中的处理, 增大块 blkExtensionData() 中的域 Length 的值, 并在块 DataBlock() 内创建大于等于域 ExtDataLength[n+1] 中所示的数据长度的连续空区域。若可创建空区域, 则进入步骤 S14 中的处理。

[0189] 在步骤 S14 中, 确定写入扩展数据区域的开头地址, 并将该开头地址的值作为域 ExtDataStartAddress[n+1]。在接下来的步骤 S15 中, 从域 ExtDataStartAddress[n+1] 写入在上述步骤 S10 中设定的域 ExtDataLength[n+1] 的长度的扩展数据 ext_data[n+1]。

[0190] 若数据的写入结束之后, 在步骤 S16 中, 对块 ext_data_entry() 追加域 ExtDataLength[n+1] 和域 ExtDataStartAddress[n+1]。

[0191] 此外, 在以上的描述中, 进行重写的块 blkExtensionData() 是已经从盘等记录介质中读出并存储在记录装置的存储器中的块。因此, 步骤 S13 中的、通过改变域 Length 的值来扩大块 blkExtensionData() 是由系统来执行的, 系统适当地进行存储器分配。

[0192] 图 27 是表示从块 blkExtensionData() 读出扩展数据时的一例处理的流程图。此外, 图 27 中的流程处理可对适用于再生专用的记录介质和可记录记录介质两者。首先, 在最初的步骤 S20 中, 根据以想要读入的扩展数据为基准的标准, 获取域 ExtDataType 的值, 在步骤 S21 中, 根据想要读入的扩展数据的类别, 获取域 ExtDataVersion 的值。

[0193] 在接下来的步骤 S22 中, 逐个依次读入块 blkExtensionData() 中列举的块 ext_data_entry()。并且, 在步骤 S23 中, 判断在读入的块 ext_data_entry() 中所包括的域

ExtDataType 和域 ExtDataVersion 的值是否与在上述步骤 S20 和步骤 S21 中获取的域 ExtDataType 和域 ExtDataVersion 的值一致。

[0194] 如果判断为不一致,则进入步骤 S26 中的处理,判断是否将块 blkExtensionData() 中列举的块 ext_data_entry() 全部读完。如果判断为全部读完,则处理进入步骤 S27,以该块 blkExtensionData() 中不存在想要读入的扩展数据而结束一连串的处理。如果判断为尚未全部读完,则处理返回到步骤 S22,读入下面的块 ext_data_entry()。

[0195] 如果在上述步骤 S23 中,判断包含在块 ext_data_entry() 中的域 ExtDataType 和域 ExtDataVersion 的值、与获取的域 ExtDataType 和域 ExtDataVersion 的值一致,则处理进入步骤 S24。在此,以在块 blkExtensionData() 中的第 [i] 个入口一致为例。

[0196] 在步骤 S24 中,从第 [i] 个入口的块 ext_data_entry() 读入域 ExtDataLength[i] 的值和域 ExtDataStartAddress[i] 的值。接着,在步骤 S25 中,从在步骤 S24 中读入的域 ExtDataStartAddress[i] 所示的地址读出由域 ExtDataLength[i] 所示的数据长度的数据。

[0197] 下面,对能够分别在上述的索引文件“index.bdmv”、影片夹对象文件“MovieObject.bdmv”、播放列表文件“xxxxx.mpls”和片段信息文件“zzzzz.clpi”中被定义的、用来存储扩展数据的扩展数据块 blkExtensionData() 进行说明。

[0198] 首先,对在索引文件“index.bdmv”中被定义的一例扩展数据块进行说明。在此描述的是:在每个播放列表中都附加相对于可记录的记录介质的特有的属性信息的、一例扩展数据块。图 28 中示出用于描述播放列表属性的、表示域“index.bdmv”内的域 blkExtensionData() 中的块 DataBlock() (参照图 24) 的一例结构的语法。在图 28 的示例中,块 DataBlock() 被描述作为块 blkIndexExtensionData()。

[0199] 首先,参照上述图 24,在块 blkExtensionData() 中,将域 ExtDataType 设为值“0x1000”,将域 ExtDataVersion 设为值“0x0100”。被描述在这些域 ExtDataType 和域 ExtDataVersion 中的值在例如再生装置侧,通过参照预先存储在 ROM(Read Only Memory,只读存储器)等中的表而加以识别。块 blkIndexExtensionData() 被存储在由块 DataBlock() 内的域 ExtDataAddress 和域 ExtDataLength 所表示的区域中。

[0200] 在块 blkIndexExtensionData() 中,域 TypeIndicator 描述以 ISO646 标准的编码方式编码的 4 字符字符串,其中,该 4 字符字符串表示后续数据的种类。在图 28 的示例中,以 ISO646 标准的编码方式编码的 4 字符字符串“IDEX”被描述在域 TypeIndicator 中,表示后续的数据的种类是索引文件中的扩展数据。

[0201] 接着域 TypeIndicator,设置有具有 32 位数据长度的区域 reserved,然后再设置具有 32 位数据长度的域 TableOfPlayListStartAddress。域 TableOfPlayListStartAddress 表示块 blkTableOfPlayList() 的、以该块 blkIndexExtensionData() 的开头为基准的开始地址。

[0202] 接着域 TableOfPlayListStartAddress,设置有具有 32 位数据长度的域 MakersPrivateDataStartAddress,表示块 blkMakersPrivateData() 的、以该块 blkIndexExtensionData() 的开头为基准的开始地址,接着具有 192 位数据长度的区域 reserved,设置块 blkUIAppInfoAVCHD()。将具有 16 位数据长度的附加字 padding_word 重

复值 N1 所代表的次数,然后设置块 blkTableOfplayLists()。再接下来,将具有 16 位数据长度的附加字 padding_word 重复值 N2 所代表的次数,随后设置块 blkMakersPrivateData()。在块 blkMakersPrivateData() 之后,将具有 16 位数据长度的附加字 padding_word 重复值 N3 所代表的次数。

[0203] 块 blkUIAppInfoAVCHD() 和块 blkMakersPrivateData() 与本发明的关联较小,故不予详细说明。

[0204] 图 29 中示出表示上述块 blkTableOfPlayLists() 的一例结构的语法。域 Length 具有 32 位数据长度,以字节数表示从紧跟该域 Length 之后直到块 blkTableOfPlayLists() 的最后字节为止的数据长度。接着域 Length,设置块 blkFirstPlayBackTitlePlayLists() 和块 blkMenuTitlePlayLists(),其中,块 blkFirstPlayBackTitlePlayLists() 描述与用于再生回放标题的播放列表相关的信息,块 blkMenuTitlePlayLists() 描述与菜单标题相关的信息。这些块 blkFirstPlayBackTitlePlayLists() 和块 blkMenuTitlePlayLists() 与本发明的关联较小,故不予说明。

[0205] 接着,设置具有 16 位数据长度的域 NumberOfTitlePlayListPair。域 NumberOfTitlePlayListPair 描述用于再生回放标题和菜单标题以外的标题的播放列表的数量。根据后面的 for 循环语句,以域 NumberOfTitlePlayListPair 所表示的数量来描述块 blkMovieTitlePlayListPair()。块 blkMovieTitlePlayListPair() 包括域 PlayListFileName、域 PlayListAttribute 和域 RefToTitleID。即,块 blkMovieTitlePlayListPair() 针对该 for 循环语句中所示的第 [i] 个播放列表,将包括该播放列表的文件名、赋予该播放列表的属性以及该播放列表的参照标题 ID 的播放列表信息构造化。

[0206] 将基于该 for 循环语句的排列顺序作为记录顺序。即,当追加一个播放列表时,域 NumberOfTitlePlayListPair 的值就增加“1”,被追加的播放列表的信息被追记在现有播放列表信息的后面。

[0207] 域 PlayListFileName 具有 40 位(5 字节)的数据长度,描述以 ISO646 标准的编码方式编码的播放列表的文件名。在域 PlayListFileName 之后隔着具有 6 位数据长度的区域 reserved,设置有域 PlayListAttribute。域 PlayListAttribute 具有 2 位数据长度,表示赋予该播放列表的属性。播放列表根据其成因可分为以下三种:对应于与片段的生成同时生成的播放列表的第一种类;对应于使用现有标题或播放列表的一部分或全部而创建的播放列表的第二种类;以及用于再生菜单的第三种类,根据播放列表的种类,将各自对应的属性“Real”(第一种类)、属性“Virtual”(第二种类)和属性“Menu”(第三种类)赋予给各播放列表。

[0208] 此外,下面,适当地将被赋予属性“Real”的播放列表称作实播放列表(real play list),将被赋予属性“Virtual”的播放列表称作虚播放列表(virtual play list),将被赋予属性“Menu”的播放列表称作菜单播放列表。

[0209] 域 RefToTitleId 描述同一循环内的域 PlayListFileName 中所示的播放列表在创建时所属的标题的 ID(编号)。作为更具体的例子,域 RefToTitleId 描述的是索引文件“index.bdmv”内的块 blkIndexes() 中的对应值 title_id。此外,当该播放列表只从第一回放标题再生时,域 RefToTitleId 的值是第一固定值,例如为“0xFFFF”。当该播放列表只

从菜单标题再生时,域 RefToTitleId 的值是第二固定值,例如为“0xFFFE”。

[0210] 下面,对虚拟播放器进行概括说明。当将具有上述数据结构的盘装入播放器中时,播放器需要将描述在从盘读出的影片夹对象等中的命令转换为用于控制播放器内部的硬件的固有命令。播放器将用于执行这样的转换的软件预先存储在内置于播放器的 ROM (Read Only Memory,只读存储器) 中。由于这种软件通过盘和播放器来使播放器进行基于 AVCHD 格式标准的操作,所以将其称为虚拟播放器。

[0211] 图 30A 和图 30B 概略地示出了该虚拟播放器的动作。图 30A 示出盘安装时的动作例。当将盘装入播放器,并对盘进行原始存取(步骤 S30)时,对在一个盘中存储有共同使用的公有参数的寄存器进行初始化(步骤 S31)。而且,在接下来的步骤 S32 中,从盘读入描述在影片夹对象等中的程序并执行。此外,初始化存取是指装入盘时这样的、初次进行盘的再生。

[0212] 图 30B 示出用户在播放器的停止状态按下例如播放键并指示再生时的动作例。用户针对最初的停止状态(步骤 S40),通过例如遥控指示(remote control commander)等来指示再生(U0:UserOperation,用户操作)。一旦指示再生,则首先寄存器、即共通寄存器被初始化(步骤 S41),并在随后的步骤 S42 中移动到影片夹对象执行阶段。

[0213] 使用图 31 来说明影片夹对象的执行阶段中的播放列表的再生。考虑通过 U0 等,用于开始再生标题编号 #1 的内容的指示存在的情况。播放器响应内容的再生开始指示,参照上述图 2 所示的索引表(Index Table),获取与标题 #1 的内容再生相对应的对象的编号。例如,如果例如,用于实现标题 #1 的内容再生的对象的编号是 #1,则播放器开始执行影片夹对象 #1。

[0214] 在图 31 的示例中,影片夹对象 #1 所描述的程序由两行组成,如果第一行的命令是“Play Playlist(1)”,则播放器开始播放列表 #1 的再生。播放列表 #1 由 1 个以上的播放项目构成,并依次再生播放项目。如果播放列表 #1 中的播放项目的再生结束,则返回执行影片夹对象 #1,并执行第二行的命令。在图 31 的示例中,第二行的命令是“jump MenuTitle”,执行该命令,并开始执行实现索引表中描述的菜单标题(MenuTitle)的影片夹对象。

[0215] 下面,对本发明一实施进行说明。在本发明中,当记录视频数据和音频数据时,将存储有视频数据和音频数据的文件的信息、与系统上的规定限制进行比较,并根据比较结果,判断继续记录时是否分割该文件。例如,当判断该文件的信息超过系统上的规定限制时,在关闭正在生成的文件的同时新创建文件,并将连续记录的视频数据和音频数据存储在該新创建的文件中。

[0216] 由于存储有视频数据和音频数据的文件根据系统上的限制而自动分割,所以用户在记录时可以进行连续的记录,而无需介意系统上的限制。

[0217] 而且,在被关闭文件的管理信息中存储表示新创建文件的信息,并且,存储表示以下内容的识别信息:无缝再生存储在被关闭文件中的视频数据和音频数据、与存储在新创建文件中的视频数据和音频数据。

[0218] 由此,再生时,即使由于系统上的限制而使存储有连续记录的视频数据和音频数据的文件被分割,也可在不介意此的情况下进行连续的再生。

[0219] 作为系统上的限制的例子,可以考虑最大文件大小。当通过计算机装置等外部设备来处理存储有通过适用本发明的记录机记录的视频数据和音频数据的文件时,根据外部

设备所使用的文件系统中的最大文件大小的限制,对可通过记录机创建的文件大小设置限制。作为可对应于大部分设备的文件系统的一例,可以列举有 FAT16。在 FAT16 中,1 个文件的最大文件大小是 2GB。例如,在记录机系统中,也根据此来确定 1 个文件的最大尺寸,并监视存储有被记录的视频数据和音频数据的片段 AV 流文件的尺寸,当文件大小超过 2GB 时,关闭当前记录中的文件而新创建文件,将视频数据和音频数据记录在该新创建的文件中。

[0220] 作为系统上的限制的其它例子,可以考虑用于表示地址信息和时刻信息的对应关系的信息的最大值。即,在 AVCHD 中,针对采用图 19 至图 23 而说明的 EP 入口,对可存储在一个片段信息文件中的 EP 入口数设置上限。因此,在视频数据和音频数据的记录中,监视该 EP 入口的数量,当 EP 入口数量超过可在一个片段信息文件中存储的最大值时,关闭当前记录中的文件并新创建文件。

[0221] 图 32 概略地示出可适用于本发明的一实施例的记录装置的一例结构。该记录装置将按规定方式对输入的数字视频数据和数字音频数据进行了压缩编码和多路复用的 AV 流记录于记录介质。

[0222] 图 32 中例示的记录装置既可用作将从外部输入的视频数据和音频数据记录于记录介质的单独的记录装置,也可以与包括光学系统或摄像元件等的照相机模块相组合,被用于将基于拍摄到的摄像信号的视频数据记录于记录介质的摄像机装置的记录模块。

[0223] 作为可适用的压缩编码和多路复用方式,可以有多种考虑。例如,可将 H. 264|AVC 中规定的方式适用于本发明的一实施例的压缩编码。但并不仅限于此,也可以根据 MPEG2 方式来进行压缩编码。另外,多路复用的方式可适用例如 MPEG2 系统。下面,以根据 H. 264|AVC 中规定的方式来进行视频数据的压缩编码,并根据 MPEG2 系统中规定的方式来进行视频数据和音频数据的多路复用为例进行说明。

[0224] 控制部 30 例如包括:CPU(Central Processing Unit,中央处理单元)、RAM(Random Access Memory,随机存取存储器)和 ROM(Read Only Memory,只读存储器)(未图示)等,根据预先存储在 ROM 中的程序或数据,将 RAM 用作工作存储器,并控制该记录装置的记录部 10 的各部分。此外,为避免繁琐,图 32 中省略了连接控制部 30 和记录部 10 中各部分的路径。

[0225] 通过在控制部 30 上动作的程序,而提供在该记录装置上使用的文件系统。例如,控制部 30 根据该文件系统,使在数据被记录于记录介质 20 时的、记录介质 20 的物理地址和存储有这些数据的数据文件相关联,同时,生成存储有各数据的文件的逻辑的管理信息。上述图 5 中所示的目录结构是文件的逻辑管理信息的一例。根据文件系统,由控制部 30 控制新文件的创建或文件的打开、关闭。

[0226] UI(User Interface,用户界面)部 31 规定地设置有助于用户操作该记录装置的动作的操作元件,并输出与对操作元件的操作相对应的控制信号。该控制信号被提供给控制部 30。控制部 30 根据用户的操作,通过基于从 UI 部 31 提供的控制信号的程序的处理来控制记录部 10 各部分的动作。例如,根据对 UI 部 31 进行的操作,由控制部 30 控制记录装置的记录动作的开始和停止操作。

[0227] 从端子 40 输入基本频带(bese band)的数字视频数据。并且,基本频带的数字音频数据也随着该数字视频数据而从端子 41 输入。

[0228] 数字视频数据从端子 40 输入到记录部 10,并被提供给视频编码器 11。视频编码

器 11 以规定方式,对提供的数字视频数据进行压缩编码。在本实施例中,以 H. 264|AVC 中规定的方式为基准,进行压缩编码,例如,通过 DCT(Discrete Cosine Transform,离散余弦变换)和图像内预测来进行帧内压缩,并进行使用移动矢量的帧间压缩,进而进行熵编码(entropy encoding)来提高压缩效率。将在视频编码器 11 中被压缩编码后的数字视频数据作为 H. 264|AVC 的基本流(ES),并被提供给多路复用器(MUX)13。

[0229] 数字音频数据从端子 41 输入到记录部 10,并被提供给音频编码器 12。音频编码器 12 根据规定的压缩编码方式,例如 AC3(AudioCode number 3)方式,对提供的数字音频数据进行压缩编码。音频数据的压缩编码方式并不限于 AC3 方式。也可以考虑不压缩编码音频数据而直接使用基本频带的数据。压缩编码的数字音频数据被提供到多路复用器 13。

[0230] 多路复用器 13 按规定方式对被分别压缩编码后被提供的数字视频数据和数字音频数据进行多路复用,将其作为 1 个数据流输出。在以 MPEG2 系统为基准进行多路复用的本实施例中,使用 MPEG2 的传输流,对提供的压缩视频数据和压缩音频数据时分复用。例如,多路复用器 13 具有缓冲存储器,将提供的压缩视频数据和压缩音频数据提供给缓冲存储器。

[0231] 在 H. 264|AVC 方式中,在可开始译码的位置的译码单位,通过使用利用帧间相关的帧间压缩来进行视频数据的编码。该译码单位例如是 GOP。存储在片段 AV 流文件中的视频数据也需要与该译码单位的界限相配合,多路复用器 13 与该译码单位的界限相配合地创建 TS 包。

[0232] 存入缓冲存储器的压缩视频数据以每个规定尺寸被分割并被附加包头(header),打包成 PES(Packetized Elementary Stream,打包基本流)包。同样,压缩音频数据也以每个规定尺寸被分割并被附加包头,打包成 PES 包。包头中存储有 PTS 和 DTS(Decoding TimeStamp,译码时间戳)等 MPEG2 系统中规定的规定信息,其中,PTS 表示存储在包中的数据的再生时刻,而 DTS 表示译码时刻。PES 包被进一步分割后塞入传输包(TS 包)的有效载荷(payload)中。TS 包的包头中存储有用于识别塞入有效载荷中的数据的 PID(Packet Identification,包标识符)。

[0233] 多路复用器 13 对 TS 包进一步附加规定尺寸的包头并输出。将该规定包头附加到 TS 包中的包被称作源包。各源包被赋予源包编号,该源包编号用来识别各源包并表示源包的顺序。源包编号例如可存储在被附加到 TS 包的包头中。从多路复用器 13 输出的源包被暂时存入流缓冲器 14。

[0234] 记录控制部 15 控制对记录介质 20 记录数据。作为记录介质 20,可以使用例如,可记录型 DVD(Digital Versatile Disc,数字化视频光盘)。但并不仅限于此,也可以使用硬盘驱动器作为记录介质 20,还可以将半导体存储器适用于记录介质 20。此外,作为记录介质 20,可考虑使用更大容量的 Blu-ray Disc(蓝光盘,注册商标)。

[0235] 记录控制部 15 监视存入流缓冲器 14 的数据量,当流缓冲器 14 中存入规定量以上的数据时,从流缓冲器 14 读出记录介质 20 的记录单位大小的数据,并写入记录介质 20。在基于文件系统的控制部 30 的控制下,以源包为单位,依次将数据追记到规定文件中,从而写入记录介质 20 中。

[0236] 管理信息处理部 16 例如由 CPU、作为工作存储器的 RAM 以及预先存储有程序和规定数据的 ROM 构成(未图示)。但并不仅限于此,管理信息处理部 16 例如也可以通过控制

部 30 中的程序处理来实现管理信息处理部 16 的功能。在这种情况下,例如,控制部 30 所具有的 RAM 被用作易失性存储器 17,且非易失性存储器 18 与控制部 30 连接。

[0237] 管理信息处理部 16 根据记录数据,将易失性存储器 17 用作工作存储器,管理信息处理部 16 用于生成存储在上述的索引文件“index.bdmv”、影片夹对象文件“MovieObject.bdmv”、播放列表文件“xxxxx.mpls”和片段信息文件“zzzzz.clpi”中的信息。生成的信息在规定定时被写入记录介质 20 中。

[0238] 作为一例,管理信息处理部 16 从多路复用器 13 取得记录数据的时间信息,同时,从记录控制部 15 取得记录数据的相对于记录介质 20 的地址信息,根据取得的这些时间信息和地址信息来生成 EP 入口信息。而且,根据对应对 UI 部 31 的记录开始、记录停止的操作而从控制部 30 输出的控制信号、以及与来自多路复用器 13 和记录控制部 15 的记录数据相关的信息,进行播放列表文件“xxxxx.mpls”的生成或更新、片段信息文件“zzzzz.clpi”的生成等。此外,当对记录介质 20 重新进行记录时,进行索引文件“index.bdmv”和影片夹对象文件“MovieObject.bdmv”的生成或更新。

[0239] 下面,对基于本发明的一实施例的片段记录控制进行说明。图 33 是表示本发明的一实施例中的一例记录方法的流程图。此外,在本发明的一实施例中,新记录的片段的信息被依次追记到已经存在的播放列表中。以在该流程的处理之前,已经在记录介质 20 中记录有播放列表文件、索引文件和影片夹对象文件为例。

[0240] 在步骤 S50 中,当进行记录开始操作时,在后面的步骤 S51 中,通过控制部 30 的控制,新创建片段 AV 流文件。例如,由文件系统设定新创建的片段 AV 流文件的逻辑结构,并与记录介质 20 上的物理地址相关。在接下来的步骤 S52 中,开始对步骤 S51 中创建的片段 AV 流文件记录片段 AV 流。

[0241] 例如,按如下述执行记录开始操作。例如,在 UI 部 31 中设置有例如用于指示记录开始的记录开始开关、和用于指示记录停止的记录停止开关。在步骤 S50 中,由用户操作记录开始开关。根据该次操作,从 UI 部 31 输出用于指示记录开始的控制信号并被提供给控制部 30。根据该指示记录开始的控制信号,控制部 30 控制记录部 10 的各部分,以便将从端子 40 输入的基本频带的视频数据、和从端子 41 输入的基本频带的音频数据记录到片段 AV 流文件中。

[0242] 作为与记录开始相关的控制部 30 的控制的一例,可以考虑:在记录停止状态下,停止视频编码器 11 和音频编码器 12 的动作,并根据记录开始的指示,开始这些视频编码器 11 和音频编码器 12 的动作。但并不仅限于此,也可以通过控制多路复用器 13、流缓冲器 14 以及记录控制部 15 的动作的开始、停止来控制记录开始和停止。

[0243] 根据记录开始控制,将片段 AV 流记录在记录介质 20 上的片段 AV 流文件中(步骤 S52)。即,输入的视频数据和音频数据在视频编码器 11 和音频编码器 12 中被分别压缩编码,并在多路复用器 13 中被打包,并被打包成将规定包头附加给 TS 包的源包,从而提供给流缓冲器 14。若在流缓冲器 14 中提供规定量以上的源包,则由记录控制部 15 从流缓冲器 14 中读出源包。根据与被规定赋予有文件名的片段 AV 流文件相关的记录介质 20 上的地址,依次将读出的源包记录于记录介质 20。

[0244] 选择不与已经记录于记录介质 20 的其他片段 AV 流文件重复的文件名作为新创建的片段 AV 流文件的文件名。

[0245] 此外,随着对记录介质 20 记录片段 AV 流,由管理信息处理部 16 实时地生成 EP 入口信息,其中,EP 入口信息表示被记录数据的再生时间和地址(片段 AV 流文件内的逻辑地址)的对应关系。该数据作为存储在上述片段信息文件“zzzzz.clpi”内的块 blkEPMAP() 中的数据,并被存储在易失性存储器 17 中。也可在非易失性存储器 18 中存储同一数据作为该数据的备份。

[0246] 在后面的步骤 S53 中,判断当前记录中的片段 AV 流文件的文件大小是否超过预定尺寸,例如 2GB。作为一例,控制部 30 总是监视当前记录中的片段 AV 流文件的文件大小,例如当从流缓冲器 14 中读出后续的源包并记录到该片段 AV 流文件中时,判断该片段 AV 流文件的文件大小是否超过规定尺寸。

[0247] 作为更具体的例子,控制部 30 监视多路复用器 13 中的处理,并从存储在片段 AV 流文件的开头的源包中累计从多路复用器 13 输出的源包的容量。根据该累计值,通过后续从多路复用器 13 输出的源包来判断数据量是否超过规定值。

[0248] 此外,如上述,与基于可开始译码位置的译码单位的界限相配合地记录片段 AV 流文件。因此,步骤 S53 中的判断实际上是按译码单位进行的。例如,当从流缓冲器 14 中读出相当于后续译码单位的源包并记录到片段 AV 流文件中时,控制部 30 判断该片段 AV 流文件的文件大小是否超过规定尺寸。

[0249] 在步骤 S53 中,若当将下一源包记录到片段 AV 流文件中时,判断该片段 AV 流文件的文件大小超过规定尺寸,则处理移至后述的步骤 S62。这时,控制部 30 保持由于被记录到片段 AV 流文件中而使文件大小超过规定尺寸的该下一源包的源包编号。

[0250] 另一方面,在步骤 S53 中,若当后续的源包被记录到片段 AV 流文件中时,该片段 AV 流文件的文件大小没有超过规定尺寸,则处理移至后述的步骤 S54,判断是否进行了记录停止操作。例如,若判断由用户操作设置于 UI 部 31 的记录停止开关,且停止记录时,处理移至步骤 S55。另一方面,若判断记录未停止,则处理返回到步骤 S52,继续对记录介质 20 记录片段 AV 流。此外,例如,在多路复用器 13 每次输出源包时,进行基于该步骤 S53 和步骤 S54 判断的循环处理。

[0251] 在步骤 S55 中,随着记录的停止,存入流缓冲器 14 的流被全部写入记录介质 20。例如,记录控制部 15 根据来自控制部 30 的记录停止命令,将存入流缓冲器 14 的流(源包)全部读出,并写入记录介质 20 上的当前记录中的片段 AV 流文件。接着,在步骤 S56 中,在文件系统中,该片段 AV 流文件被关闭。

[0252] 此外,根据记录停止命令,例如停止视频编码器 11 和音频编码器 12 的动作。这时,为了进行使用图 13A 所描述的第一无缝连接而进行控制,以便在例如视频编码器 11 的动作停止之后的规定时间后停止音频编码器 12 的动作。

[0253] 在随后的步骤 S57 至 S61 中,生成与写入记录介质 20 的片段 AV 流文件相关的片段信息文件,并执行播放列表文件的更新。

[0254] 首先,在步骤 S57 中,通过管理信息处理部 16 生成对应于在步骤 S56 中关闭的片段 AV 流文件的片段信息文件“zzzzz.clpi”。文件名是与在步骤 S51 中创建的片段 AV 流文件的文件名相对应的文件名,当该片段 AV 流文件的文件名是“zzzzz.m2ts”时,该片段信息文件的文件名是“zzzzz.clpi”,与扩展名前面部分相同。

[0255] 根据如图 15 至图 21 例示的各语法,将各域和标记的值规定地设定并存储到片段

信息文件“zzzzz.clpi”中。作为一例,管理信息处理部 16 根据在片段记录中从多路复用器 13 取得的信息,生成源包相关信息、和再生时间 (PTS) 相关信息。管理信息处理部 16 根据在片段记录中从记录控制部 15 取得的信息,生成与记录介质 20 上的记录地址相关的信息。系统的固有值是基于例如预先存储在 ROM(未图示)等中的信息。此外,表示再生时间和地址的对应关系的上述块 blkEMap() 的信息被存储在片段信息文件“zzzzz.clpi”的块 blkCPI() 中。

[0256] 当通过用户操作停止片段的记录时,块 blkClipInfo() 内的标记 IsCC5 的值是“1”(二进制值)。与此相对,规定设置由块 blkClipInfo() 内的 if 语句(参照图 16)所示的数据。

[0257] 若片段信息文件的创建结束,则处理移至随后的步骤 S58。步骤 S58 至步骤 S61 中的处理是与播放列表文件相关的处理。通过该步骤 S58 至步骤 S61 中的处理,对已经存在于记录介质 20 上的播放列表文件追加对应于新记录的片段 AV 流文件“00002.m2ts”的播放项目。

[0258] 首先,在步骤 S58 中,将播放列表文件内的块 blkPlayItem() 中的域 ConnectionCondition 的值设定为“5”,表示该片段 AV 流文件“00002.m2ts”通过第一无缝连接而连接于下一个记录的片段 AV 流文件(参照图 12)。在接下来的步骤 S59 中,播放项目文件的域“NumberOfPlayItems”的值增加“1”,表示对该播放列表追加了一个播放项目(参照图 11)。

[0259] 在后面的步骤 S60 中,分别设置块 blkPlayItem() 中的域 ClipInformationFileName、域 INTime 和域 OUTTime,并创建随片段的记录而追加的块 blkPlayItem()。域 ClipInformationFileName 存储有在上述步骤 S55 中创建的片段信息文件的文件名“zzzzz.clpi”。实际上,由于片段信息文件的扩展名是固定的,所以存储的是点前面的部分“zzzzz”。域 INTime 和域 OUTTime 是表示存储在对应的片段 AV 流文件“00002.m2ts”中的视频流的开头和结束时间的信息,其基于例如片段信息文件“zzzzz.clpi”内的块 blkCPI() 中的块 blkEMap() 的信息。

[0260] 在后面的步骤 S61 中,播放列表文件内的块 blkPlayListMark() 中的域 NumberOfPlayListMarks 的值增加“1”,随此,追加到 for 循环语句内的域 MarkTimeStamp 的值被设定为上述步骤 S60 中设定的、块 blkPlayItem() 中的域 INTime 的值。即,在与新记录片段 AV 流文件的开始相对应的时刻,打播放列表标签。

[0261] 如上述,对新记录片段 AV 流文件“zzzzz.m2ts”创建片段信息文件“zzzzz.clpi”,并更新现有的播放列表文件。

[0262] 在停止记录状态下,通过对 UI 部 31 进行记录开始操作,从而从步骤 S50 再次开始处理,同样地,当向记录介质 20 记录新的片段 AV 流文件时,进行基于片段 AV 流文件的文件大小的判断,创建相应的片段信息文件,并生成参照创建的片段信息文件的播放项目,进一步将生成的播放项目追加到播放列表中。

[0263] 也可以在步骤 S61 的处理之后,执行上述步骤 S55 中的、将存入流缓冲器 14 中的数据写入记录介质 20 的写入处理。

[0264] 根据上述步骤 S53 的判断,对本发明的一实施例的、基于系统情况来分割片段 AV 流文件时的处理进行说明。在上述步骤 S53 中,当将下一个源包记录到片段 AV 流文件中时,

如果判断该片段 AV 流文件的文件大小超过规定尺寸,则处理移至步骤 S62。在步骤 S62 中,存入流缓冲器的流被写到片段 AV 流文件“zzzzz.m2ts”中。

[0265] 这时,根据在执行步骤 S53 的判断时保持在控制部 30 中的源包编号,将由于在记录到片段 AV 流文件“zzzzz.m2ts”中而使其文件大小超出规定尺寸的该下一个源包之前、即比该下一个源包更早存入流缓冲器中的流写入片段 AV 流文件“zzzzz.m2ts”。

[0266] 在后面的步骤 S63 中,由文件系统将当前记录中的片段 AV 流文件关闭。若文件被关闭,则在步骤 S64 中,由管理信息处理部 16 与步骤 S57 中描述的处理同样地规定生成与在步骤 S51 中创建的片段 AV 流文件相对应的片段信息文件“zzzzz.clpi”。

[0267] 在后面的步骤 S65 中,播放列表文件内的块 blkPlayItem() 中的域 ConnectionCondition 的值被设定为“6”,表示该片段 AV 流文件“zzzzz.m2ts”与下一个记录的片段 AV 流文件通过第二无缝连接而连接(参照图 12)。

[0268] 在步骤 S66 中,播放列表文件的域“NumberOfPlayItems”的值增加“1”,表示对该播放列表追加了一个播放项目(参照图 11)。接着,在步骤 S67 中,设定块 blkPlayItem() 中的域 ClipInformationFileName,并根据片段信息文件内的块 blkCPI() 中的块 blkEPMMap() 的信息,分别设定域 INTime 和域 OUTTime,并创建随片段的记录而增加的块 blkPlayItem()。

[0269] 在后面的步骤 S68 中,播放列表文件内的块 blkPlayListMark() 中的域 NumberOfPlayListMarks 的值增加“1”,随此追加到 for 循环语句内的域 MarkTimeStamp 的值被设定为在上述步骤 S67 中设定的、块 blkPlayItem() 中的域 INTime 的值。即,在新记录片段的开头打播放列表标签。

[0270] 若步骤 S68 的处理结束,则处理返回到步骤 S51,新创建片段 AV 流文件。在该步骤 S53 中的判断后的步骤 S51 中创建的片段 AV 流文件是通过步骤 S53 的判断,当将下一个源包存储到文件时,该文件的文件大小超出规定尺寸的、从该下一个源包开始依次存储有源包的文件。

[0271] 在此,这时创建的片段 AV 流文件与前一个创建的片段 AV 流文件的视频数据和音频数据分别是连续的。例如,对音频数据而言,控制记录,以便使记录到前一个创建的片段 AV 流文件中的音频数据的后端(last end)时刻、与记录到在此创建的片段 AV 流文件中的音频数据的开头时刻相对应。例如,控制部 30 对音频编码器 12、多路复用器 13 和流缓冲器 14 进行规定控制,从而执行像这样的记录控制。

[0272] 此外,虽然上面描述的是在步骤 S68 的处理之后返回到步骤 S51,创建新的片段 AV 流文件,但并不仅限于此例。即,当通过步骤 S53 的判断,根据系统的情况,片段 AV 流文件被分割时,由于分割前后的片段 AV 流文件以第二无缝连接相连接,所以优选尽可能缩短在步骤 S63 中关闭文件开始直至创建新的片段 AV 流文件为止的时间。

[0273] 因此,例如,可考虑在步骤 S63 中关闭片段 AV 流文件之后直接使处理返回到步骤 S51,新创建片段 AV 文件。在这种情况下,步骤 S64 至步骤 S68 的处理与创建新片段 AV 流文件的处理并行执行。播放列表文件和在步骤 S63 中创建的片段信息文件被暂时存储在易失性存储器 17 和 / 或非易失性存储器 18 中。

[0274] 并且,例如,在根据系统情况,对随着步骤 S50 的记录开始操作而新创建的片段 AV 流文件进行分割时,除初次分割时以外,都可以省略上述步骤 S61 和步骤 S68 中的设置播放

列表标签的处理。作为一例,可考虑省略与通过第二无缝连接而连接的片段 AV 流文件的开头相对应的播放列表标签。但并不仅限于此,也可以针对在记录停止操作时,通过与该记录停止操作对应的记录开始操作,使记录开始的时刻设置播放列表标签。

[0275] 图 34 示出了根据上述图 33 中的步骤创建的一例文件结构。作为一例,根据已经按图 35A 所示的时间经过来执行记录。即,参照图 35A,在时刻 ST_1 执行对记录介质 20 记录的记录开始操作,在时刻 ED_1 执行相应的记录停止操作,从而形成记录区间 400 的记录。在时刻 ED_1 之后的时刻 ST_2 再次对同一记录介质 20 进行记录开始操作,并在时刻 ED_2 进行记录停止操作,从而形成记录区间 401 的记录。同样,在时刻 ED_2 之后的时刻 ST_3 又一次对同一记录介质 20 进行记录开始操作,并在时刻 ED_3 进行记录停止操作,从而形成记录区间 402 的记录。

[0276] 在此,例如由图 35B 所示的长度来表示相当于被设定为片段 AV 流文件的文件大小上限的规定尺寸(例如 2GB)的记录时间。在这种情况下,在记录区间 400 中,从记录开始时刻 ST_1 至时间点 a 之间进行与 2GB 的规定尺寸相应时间的记录,再从时间点 a 至时间点 b 之间进行与规定尺寸相应时间的记录。将从时间点 b 至记录停止时刻 ED_1 这段作为小于与规定尺寸相应的记录时间。

[0277] 另一方面,记录区间 401 和记录区间 402 分别小于与作为文件大小上限的 2GB 相对应的记录时间。

[0278] 参照图 33 的流程和图 34 的文件结构,对在图 35A 所示的时间经过(time elapse)中执行记录的情况下,上述图 33 中所示的处理进行更为具体的说明。

[0279] 首先,随着在时刻 ST_1 的记录开始操作,在步骤 S51 中,创建片段 AV 流文件“00001.m2ts”。对创建的片段 AV 流文件“00001.m2ts”记录流,例如,控制部 30 以源包为单位,检查片段 AV 流文件“00001.m2ts”的文件大小,并判断文件大小是否超出规定尺寸。实际上,如上述,根据从多路复用器 13 输出的源包的数据量的累计结果进行判断。在图 35A 的示例中,判断刚好在时间点 a 之前,若记录下一个源包,则片段 AV 流文件“00001.m2ts”的文件大小超过规定尺寸。

[0280] 处理移至步骤 S62,规定读出存入流缓冲器中的流,并写入片段 AV 流文件“00001.m2ts”。接着,在步骤 S63 中关闭片段 AV 流文件“00001.m2ts”并创建相应的片段信息文件“00001.clpi”(步骤 S64)。此外,如上述,也可以为:若关闭片段 AV 流文件“00001.m2ts”,则与步骤 S64 之后的处理并行地将处理返回至步骤 S51,创建下一个片段 AV 流文件。

[0281] 从后面的步骤 S65 开始,创建参照片段 AV 流文件“00001.m2ts”的播放项目 #1,并更新播放列表 #1。首先,播放项目 #1 内的域 ConnectionCondition 的值被设置为表示第二无缝连接的值“6”(步骤 S65),根据片段信息文件“00001.clpi”,例如规定创建将时刻 ST_1 和时刻 a 分别作为域 INTime 和 OUTTime 的播放项目 #1(步骤 S66)。接着,播放项目 #1 被描述在播放列表 #1 中。此外,将域 MarkTimeStamp 的值设为域 INTime 的值、即时刻 ST_1 ,并设置播放列表标签 Mark#1。

[0282] 处理返回至步骤 S51,为了记录时刻 a 之后的流,新创建片段 AV 流文件“00002.m2ts”。将流记录到该片段 AV 流文件“00002.m2ts”中(步骤 S52)。

[0283] 在此,如上述,在对应于前一个创建并被记录的片段 AV 流文件“00001.m2ts”的播放项目 #1 中,域 ConnectionCondition 的值被设定为“6”,指定第二无缝连接。因此,执行

记录控制,以便可连续地再生记录到前一个被记录的片段 AV 流文件“00001.m2ts”中的音频数据、和记录到在此创建的片段 AV 流文件“00002.m2ts”中的音频数据。例如,执行记录控制,以便使记录到片段 AV 流文件“00001.m2ts”中的音频数据的后端时刻、与片段 AV 流文件“00002.m2ts”的开头时刻相对应。

[0284] 接着,在步骤 S53 中判断片段 AV 流文件“00002.m2ts”的文件大小是否超出规定尺寸。在图 35A 的示例中,判断刚好在时间点 b 之前,若记录下一个源包,则片段 AV 流文件“00002.m2ts”的文件大小超过规定尺寸。于是,处理移至步骤 S62,与上述同样地,在将存入流缓冲器中的流写入片段 AV 流文件“00002.m2ts”(步骤 S62)之后关闭该片段 AV 流文件“00002.m2ts”(步骤 S63)。然后,创建与该片段 AV 流文件“00002.m2ts”相应的片段信息文件“00002.clpi”(步骤 S64)。

[0285] 接着,将对应于片段信息文件“00002.clpi”的播放项目 #2 的域 ConnectionCondition 设定为表示第二无缝连接的值“6”(步骤 S65),并在后面的步骤 S66 中,规定创建将时刻 a 和时刻 b 分别作为域 INTime 和 OUTTime 的播放项目 #2,并将其追记到播放列表 #1 中。在步骤 S68 中,设定表示域 INTime 的值、即时刻 a 的播放列表标签。此外,可省略此处的播放列表标签的设置处理。

[0286] 处理再次返回到步骤 S51,为了记录时刻 b 之后的流,新创建片段 AV 流文件“00003.m2ts”。将流记录到该片段 AV 流文件“00003.m2ts”中(步骤 S52)。

[0287] 在此,如上述,在对应于前一个创建并被记录的片段 AV 流文件“00001.m2ts”的播放项目 #1 中,域 ConnectionCondition 的值被设定为“6”,指定第二无缝连接。因此,与上述同样地,执行的记录控制,以便使记录到片段 AV 流文件“00002.m2ts”的音频数据的后端时刻、与片段 AV 流文件“00003.m2ts”的开头时刻相对应。

[0288] 接着,在步骤 S53 中,判断片段 AV 流文件“00003.m2ts”的文件大小是否超出规定尺寸。在图 35A 的示例中,在片段 AV 流文件“00003.m2ts”的文件大小超出文件大小的上限、即 2GB 之前,在时刻 ED₁ 进行了记录停止操作。因此,根据步骤 S54 的判断,处理移至步骤 S55。在步骤 S55 中,存入流缓冲器的流被记录到片段 AV 流文件“00003.m2ts”中,并在步骤 S56 中关闭该片段 AV 流文件“00003.m2ts”。在步骤 S57 中,创建相应的片段信息文件“00003.clpi”。

[0289] 接着,参照该片段信息文件“00003.clpi”的播放项目 #3 的域 ConnectionCondition 被设定为表示第一无缝连接的值“5”(步骤 S58)。然后,按规定创建播放项目 #3,并将该播放项目 #3 追加记录到播放列表文件 #1 中(步骤 S59、步骤 S60)。进而,可在对应于播放项目 #3 的域 INTime 的时刻,对播放列表文件 #1 打播放列表标签(步骤 S61)。此外,可省略此处的播放列表标签的设置处理。

[0290] 如果在时刻 ED₁ 执行了记录停止操作之后,再次在时刻 ST₂ 开始记录开始操作(步骤 S50),则在步骤 S51 中新创建片段 AV 流文件“00004.m2ts”,将流记录到该片段 AV 流文件“00004.m2ts”中(步骤 S52)。在图 35A 的示例中,在片段 AV 流文件“00004.m2ts”的文件大小超出规定尺寸之前,在时刻 ED₂ 进行了记录停止操作。因此,根据步骤 S54 的判断,处理移至步骤 S55。

[0291] 在步骤 S55 中,存入流缓冲器的流被记录到片段 AV 流文件“00004.m2ts”中,并在步骤 S56 中,关闭该片段 AV 流文件“00004.m2ts”,在步骤 S57 中,创建相应的片段信息文

件“00004.clpi”。在步骤 S58 中,参照该片段信息文件“00004.clpi”的播放项目 #4 的域 ConnectionConditon 的值被设定为表示第一无缝连接的值“5”。并且,规定创建播放项目 #4,并将该播放项目 #4 追加记录到播放列表 #1 中(步骤 S59、步骤 S60)。进而,在对应于播放项目 #4 的域 INTime 的时刻,对播放列表 #1 设置播放列表标签 Mark#2(步骤 S61)。

[0292] 此外,由于图 35A 中的、从下一记录开始时刻 ST_3 至记录停止时刻 ED_3 的处理与上述的、从记录开始时刻 ST_2 至记录停止时刻 ED_2 的处理相同,所以为避免繁琐,省略对其的说明。

[0293] 此外,随着最初的片段 AV 流文件“00001.m2ts”的记录,可创建描述调用播放列表 #1 的影片夹对象 #1 的影片夹对象文件、和用于将影片夹对象 #1 选择及再生作为标题 #1 的索引文件。影片夹对象文件和索引文件可预先创建有模板(template)并将其更新。由于这些影片夹对象文件和索引文件的创建与本发明的主旨关联较小,故不予对其详细说明。

[0294] 对记录介质 20 的一例再生进行概略的说明,其中,该记录介质 20 正如使用图 34 和图 35A 所说明的一样,记录有片段 AV 流文件。此外,在此,以记录介质 20 是可记录型 DVD 为例。当将记录介质 20 装入到再生装置中后,读入索引文件,并进一步读入影片夹对象文件。若对索引文件指示标题 #1 的再生,则参照影片夹对象文件内的相应影片夹对象 #1,从记录介质 20 读入描述有播放列表 #1 的播放列表文件。

[0295] 依照播放列表 #1,依次再生播放项目 #1、播放项目 #2、播放项目 #3、播放项目 #4.....。从播放项目 #1 开始参照片段信息文件“00001.clpi”,并再生片段 AV 流文件“00001.m2ts”,按照播放列表 #1 的描述,然后从播放项目 #2 开始参照片段信息文件“00002.clpi”,并再生片段 AV 流文件“00002.m2ts”。

[0296] 由于播放项目 #1 的域 ConnectionCondition 的值被设定为“6”,且指示第二无缝连接,所以片段 AV 流文件“00001.m2ts”和片段 AV 流文件“00002.m2ts”被无缝连接并再生。这时,对音频数据执行记录控制,以便使片段 AV 流文件“00001.m2ts”的后端时刻、与片段 AV 流文件“00002.m2ts”的开头时刻相对应,所以基于片段 AV 流文件“00001.m2ts”的音频、和基于片段 AV 流文件“00002.m2ts”的音频被作为连续的音频被再生。

[0297] 按照播放列表 #1 的描述,在播放项目 #2 之后进行播放项目 #3 的再生。即,从播放项目 #3 开始参照片段信息文件“00003.clpi”,并再生片段 AV 流文件“00003.m2ts”。

[0298] 由于播放项目 #2 的域 ConnectionCondition 的值被设定为“6”,且指示第二无缝连接,所以片段 AV 流文件“00002.m2ts”和片段 AV 流文件“00003.m2ts”被无缝连接并再生。即使在这种情况下,也对音频数据执行记录控制,以便使片段 AV 流文件“00002.m2ts”的后端时刻、与片段 AV 流文件“00003.m2ts”的开头时刻相对应,所以基于片段 AV 流文件“00002.m2ts”的音频、和基于片段 AV 流文件“00003.m2ts”的音频被作为连续的音频被再生。

[0299] 因此,在从连续记录的记录区间 400(参照图 35A)的记录开始时刻 ST_1 到记录停止时刻 ED_1 的期间,按规定尺寸分割为三个文件并记录的片段 AV 流文件“00001.m2ts”、“00002.m2ts”及“00003.m2ts”可以恰如在一个连续的片段 AV 流文件中记录的视频数据和音频数据这样地被连续再生,而无需在再生时使用户意识到文件分割。

[0300] 另一方面,在时刻 ED_1 进行了记录停止操作之后,通过在时刻 ST_2 进行记录开始操作并记录的片段 AV 流文件“00003.m2ts”根据播放列表 #1,在上述播放项目 #3 的再生之

后,根据播放项目 #4 进行再生。这时,对播放项目 #3 的域 ConnectionCondition 设定表示第一无缝连接的值“5”。因此,在播放项目 #3 中,音频数据在片段 AV 流文件“00003.m2ts”的后端,比视频数据记录得稍微长些。

[0301] 例如,对该部分增加淡出等期望的声音效果,与下一播放项目 #4 的片段 AV 流文件“00004.m2ts”的开头部分的音频数据相混合(mix),连续地再生片段 AV 流文件“00003.m2ts”和片段 AV 流文件“00004.m2ts”。通过这种方式,从而可以抑制场景不同的两个片段 AV 流文件间的不适感,并能无缝再生这两个文件。

[0302] 下面,对本发明一实施例的其它的例子进行说明。在上述已经对将本发明适用于单独的记录装置中的例子进行了说明(参照图 32)。对此,在本实施例的其它例子中,本发明被适用于摄像机装置中,该摄像机装置具有摄像元件、和用于将来自被摄体的光射入摄像元件的光学系统,摄像机装置根据通过摄像元件拍摄的摄像信号,将视频数据记录于记录介质。

[0303] 图 36 中示出了根据本发明的一实施例的其他例子的摄像机装置 100 的一例的结构。在摄像机装置 100 中,记录系统的结构可大致直接适用图 32 中说明的记录装置的结构,因此与图 32 中相同的部分标注了相同的标记,并省略对其的说明。

[0304] 在图 36 的结构中,在照相机部 50 中,作为与映像信号相关的结构,其包括光学系统 51、摄像元件 52、摄像信号处理部 53、照相机控制部 54 和显示部 55,作为与音频信号相关的结构,其包括麦克风(MIC)56 和音频信号处理部 57。控制部 30 与照相机部 50 的各部分之间进行各种控制信号和信息的交换,并控制照相机部 50 的动作。此外,控制部 30 响应用户操作,根据从 UI 部 31 提供的控制信号来控制照相机部 50 的动作。

[0305] 此外,在构成为摄像机装置 100 的情况下,例如使用设置于 UI 部 31 上的单个的记录开关,每按一下该记录开关就交替指示记录开始和记录停止,一般都以这种方式来进行记录开始操作和记录停止操作。在该摄像机装置 100 中,可将 Blu-ray Disc(蓝光盘)或可记录型 DVD 等的、盘记录介质用作记录介质 20。

[0306] 当然并不仅限于此,也可将硬盘驱动器内置于摄像机装置 100 中(未图示),还可以将该硬盘驱动器用作记录介质 20。另外,也可以将非易失性半导体存储器内置于或者可拆卸地设置于摄像机装置 100 中,将该非易失性存储器用作记录介质 20,其中,该非易失性存储器具有可存储进行了规定的连续记录时间的记录时的数据的容量。

[0307] 在照相机部 50 中,光学系统 51 包括用于将来自被摄体的光导向摄像元件 52 的透镜系统、光圈调整机构、焦距调整机构、变焦机构和快门机构等。根据从控制部 30 提供的控制信号,由照相机控制部 54 控制光圈调整机构、焦距调整机构、变焦机构和快门机构的动作。

[0308] 摄像元件 52 例如由 CCD(Charge Coupled Device,电荷耦合器件)构成,其通过光电转换,将经由光学系统 51 照射的光转换成电信号,进行规定的信号处理,并将其作为摄像信号加以输出。摄像信号处理部 53 对从摄像元件输出的摄像信号进行规定的信号处理,并将其作为基本频带的数字视频数据加以输出。

[0309] 例如,摄像信号处理部 53 通过 CDS(Correlated DoubleSampling,相关双采样)电路,对从摄像元件 52 输出的摄像信号只采样具有图像信息的信号,同时除去噪声,并通过 AGC(Auto GainControl,自动增益控制)电路调整增益。此外,通过 A/D 转换,被转换成数

字信号。而且,摄像信号处理部 53 对该数字信号进行检波系统的信号处理,取出 R(红色)、G(绿色)和 B(蓝色)各颜色的成分,执行 γ 校正和白平衡校正等处理,最终将其作为一条基本频带的数字视频数据加以输出。

[0310] 摄像信号处理部 53 将从摄像元件 52 输出的摄像信号的信息发送给控制部 30。控制部 30 根据该信息,生成用于控制光学系统 51 的控制信号,并提供给照相机控制部 54。照相机控制部 54 根据该控制信号,进行焦距调整机构和光圈调整机构等的控制。

[0311] 此外,摄像信号处理部 53 根据从摄像元件 52 输出的摄像信号,生成使将 LCD(Liquid Crystal Display,液晶显示器)用作显示元件的显示部 55 映出的映像信号。

[0312] 另一方面,麦克风 56 拾取周围的声音,并将其转换成电信号加以输出。从麦克风 56 输出的音频信号被提供到音频信号处理部 57。音频信号处理部 57 使提供的音频信号通过限幅器(limiter)之后,进行 A/D 转换,将其转换为数字音频数据,并在进行了除去噪声及音质校正等规定音频信号处理之后,将其作为基本频带的数字音频数据加以输出。

[0313] 从照相机部 50 的摄像信号处理部 53 输出的基本频带的数字视频数据被提供到记录部 10 的端子 40。从音频信号处理部 57 输出的基本频带的数字音频数据被提供到记录部 10 的端子 41。

[0314] 当在记录停止状态下,按下设置于 UI 部 31 的记录开关时,指示记录开始的控制信号被从 UI 部 31 提供到控制部 30,根据控制部 30 的控制,开始将从照相机部 50 输出的基本频带的数字视频信号和数字音频数据记录于记录介质 20。

[0315] 即,正如上面所描述地,根据控制部 30 的控制,开始视频编码器 11 和音频编码器 12 的动作,分别通过视频编码器 11 和音频编码器 12 对视频数据和音频数据进行压缩编码,并在多路复用器 13 中规定地进行打包及多路复用,从而形成 AV 流数据。AV 流数据通过流缓冲器 14 而被提供到记录控制部 15,并将其作为片段 AV 流文件记录于记录介质 20。

[0316] 当从记录开始操作开始连续记录规定时间以上,且使片段 AV 流文件的文件大小超过规定尺寸时,正如已经描述过地,关闭正在记录的片段 AV 流文件,创建相应的片段信息文件,并生成参照该片段信息文件的播放项目。在此生成的播放项目指定第二无缝连接。生成的播放项目被追记到播放列表中。并且,新创建片段 AV 流文件,将由于继续记录而提供的 AV 流数据记录到该新创建的片段 AV 流文件中。

[0317] 若按下 UI 部 31 的记录开关,则记录停止,创建片段信息文件并更新播放列表文件。管理信息处理部 16 根据来自多路复用器 13 和记录控制部 15 的信息,创建对应于记录于记录介质 20 的片段 AV 流文件的片段信息文件。并且,管理信息处理部 16 生成参照该片段信息文件的播放项目,在已经存在播放列表的情况下,将生成的播放项目追加到该播放列表中,并对播放列表打播放列表标签。

[0318] 若在该状态下又一次按下记录开关,则再次指示记录开始,将新的片段 AV 流文件记录于记录介质 20,并进行以下处理:对应的片段信息文件的创建、生成参照该片段信息文件的播放项目、以及对播放列表追加生成的播放项目等。片段 AV 流文件间的再生以第一无缝连接相连接。

[0319] 如本发明的一实施例的其他例子所述,在将本发明运用于摄像机装置 100 的情况下,当进行长时间的连续拍摄,并从记录开始操作直至记录停止操作期间记录的片段 AV 流文件的文件大小超过规定尺寸时,自动分割片段 AV 流文件,以使 1 个文件的尺寸不超过规

定尺寸。并且,分割时,由于设定第二无缝连接作为分割前后的片段 AV 流文件的连接方法,因此,在再生时,无需意识到基于规定尺寸的片段 AV 流文件的分割,从而可以连续地再生由一次连续拍摄所记录到的视频数据和音频数据。

[0320] 此外,上述已经对根据文件大小来判断是否进行片段 AV 流文件分割的例子进行了说明,但并不仅限于此。即,也可考虑基于对片段 AV 流文件的管理信息来判断是否进行文件分割。例如,如上述,根据 EP 入口信息的上限来判断是否进行片段 AV 流文件的分割,其中,该 EP 入口信息表示片段 AV 流文件的地址信息、和记录到该文件中的视频数据及音频数据的时刻信息间的对应关系。

[0321] 如采用图 20 至图 23 所述,EP 入口分别具有基于源包的入口和基于 PTS 的入口,并且,各入口构成为包括进行粗检的入口和进行精检的入口。如图 22 和图 23 所示,由于以格式来规定用于表示入口的位数,所以通过该位数可表示的最大值对应于最大的入口数。作为一例,控制部 30 根据与来自管理信息处理部 16 的入口点相关的信息,判断正在记录的片段 AV 流文件的入口点数是否达到最大值。

[0322] 作为一例,参照图 33 的流程,控制部 30 监视在管理信息处理部 16 中生成的 EP 入口信息,并判断相对于一个片段 AV 流文件而生成的 EP 入口的数量是否变为规定数量。并且,如果判断 EP 入口数已变为规定数量,则处理移至步骤 S62,规定地读出存入流缓冲器 14 中的流,并将其记录到片段 AV 流文件中,关闭该片段 AV 流文件(步骤 S63)。接着,创建对应于该片段 AV 流文件的片段信息文件(步骤 S64),将对应的播放项目内的域 ConnectionCondition 的值设定为表示第二无缝连接的值“6”(步骤 S65)。此外,在步骤 S66 和步骤 S67 中生成播放项目的信息,并将生成的播放项目追加记录到播放列表中。根据需要,在步骤 S68 中进行播放列表标签的设定。

[0323] 此外,并不限于根据 EP 入口数量来判断是否分割片段 AV 流文件,例如,也可以根据 PTS 的上限来判断是否进行片段 AV 流文件的分割。

[0324] 并且,如上述,虽然已经对如图 32 所示的记录装置和如图 36 所示的摄像机装置 100 的记录部 10 的硬件构成进行了说明,但并不仅限于这些例子。即,记录部 10 也可以作为软件构成。在这种情况下,例如将软件预先存储在控制部 30 所具有的未图示的 ROM 中。并不仅限于此,也可以将记录部 10 构成在个人计算机等计算机装置中。在这种情况下,使计算机装置执行记录部 10 的软件被记录到 CD-ROM 或 DVD-ROM 等记录介质中而加以提供。当计算机装置可与网络连接时,也可以经由英特网等网络提供该软件。

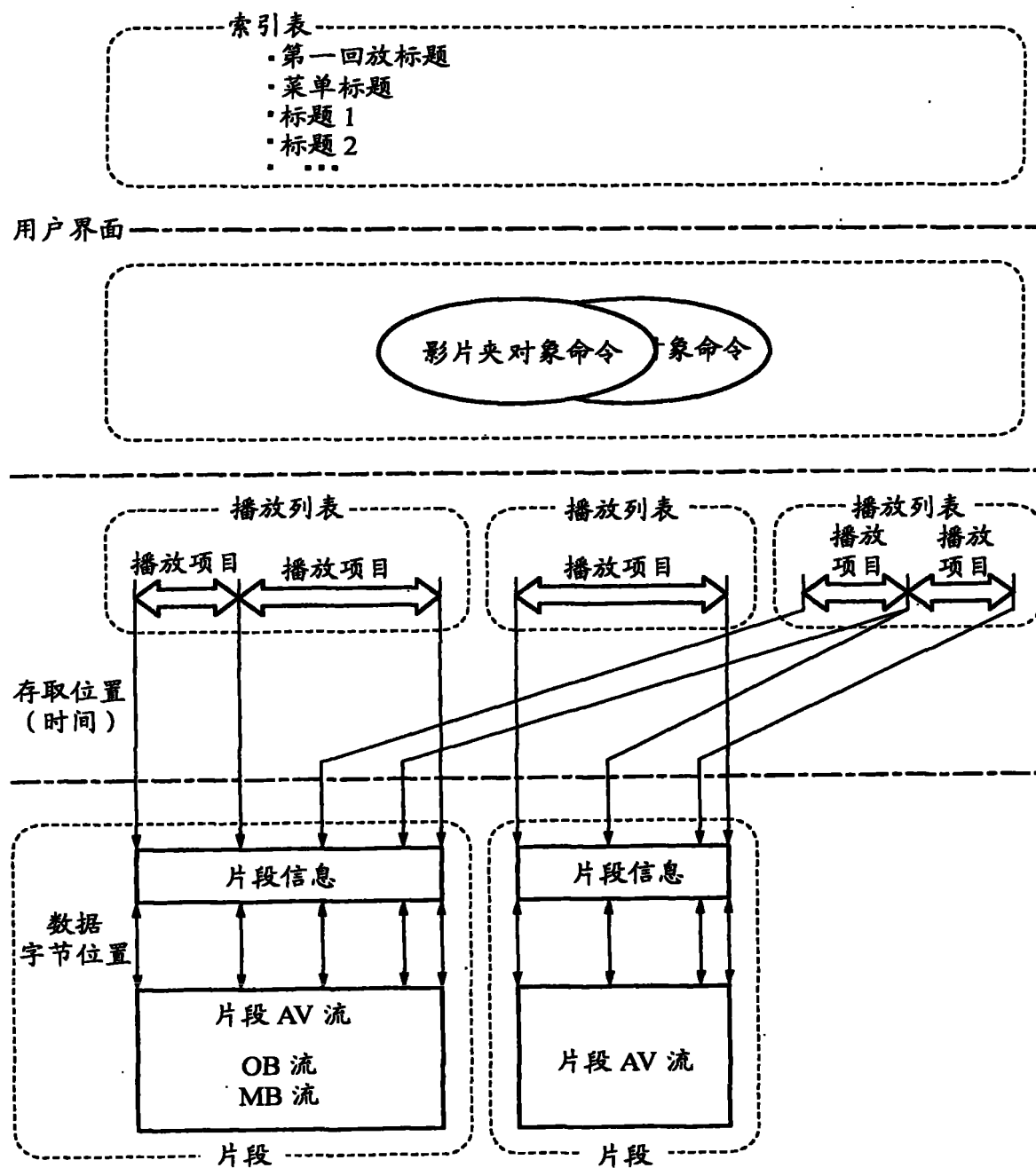


图 1

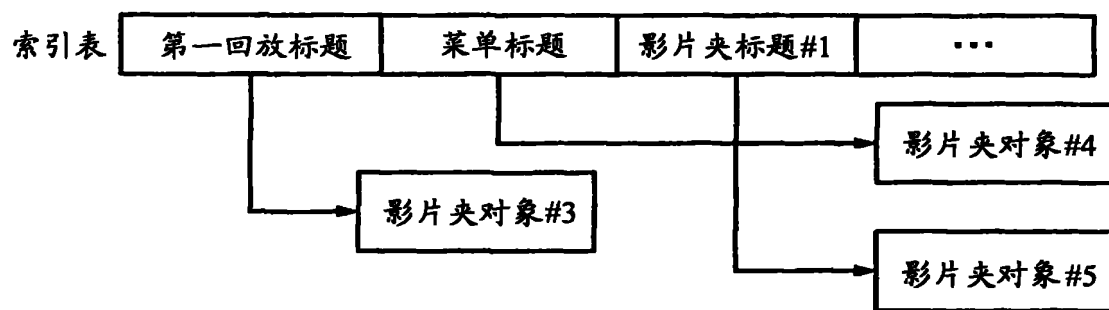


图 2

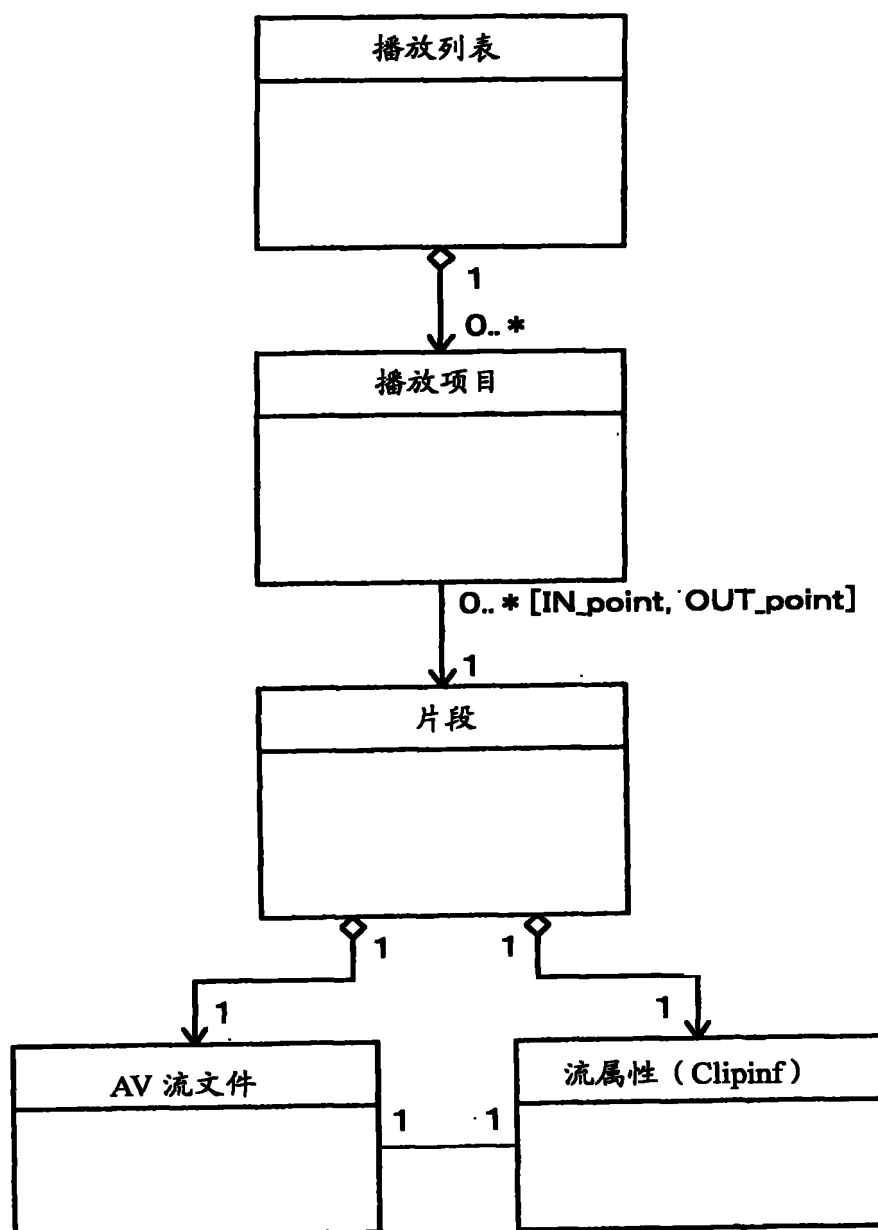
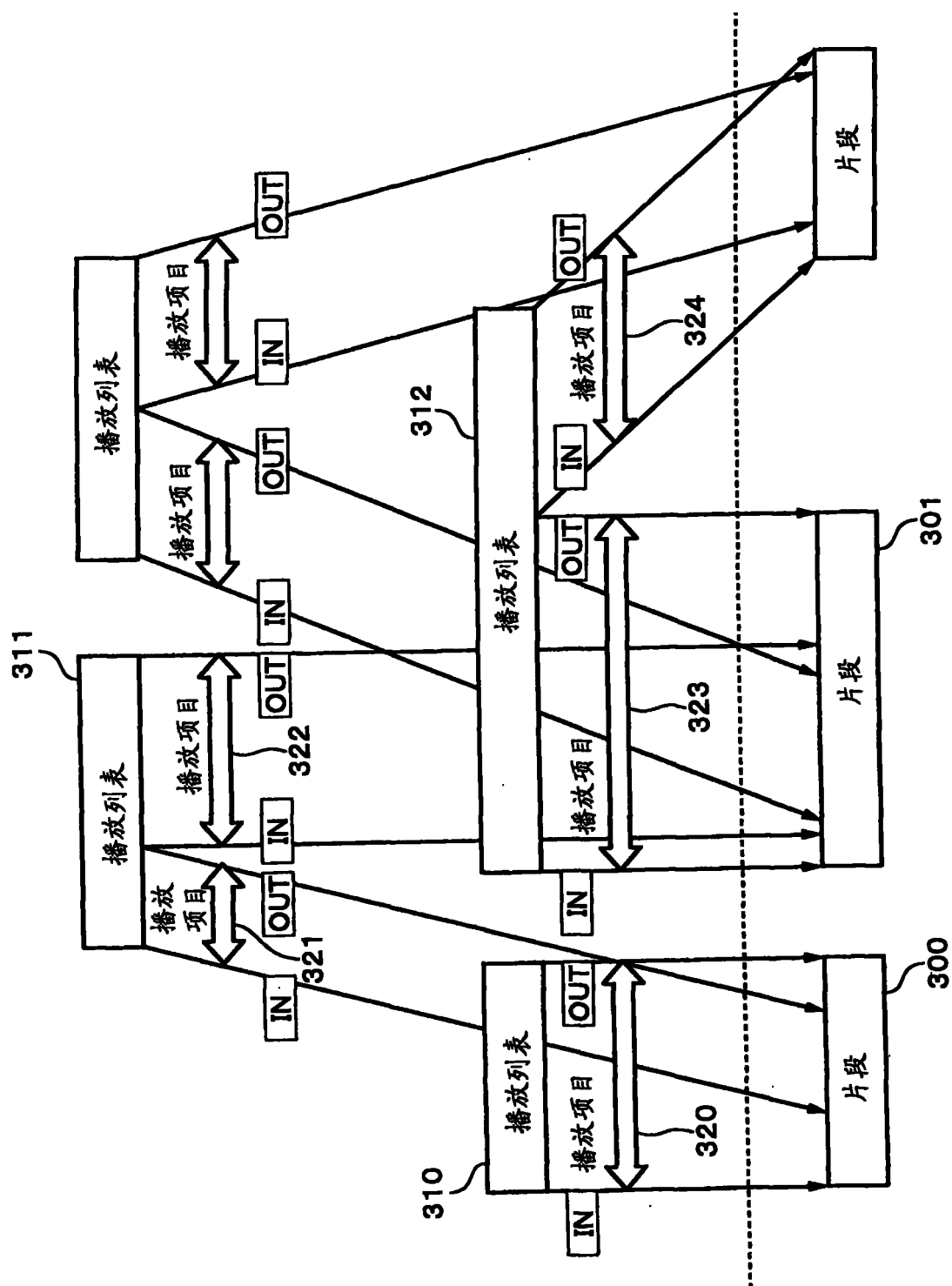


图 3



4
图

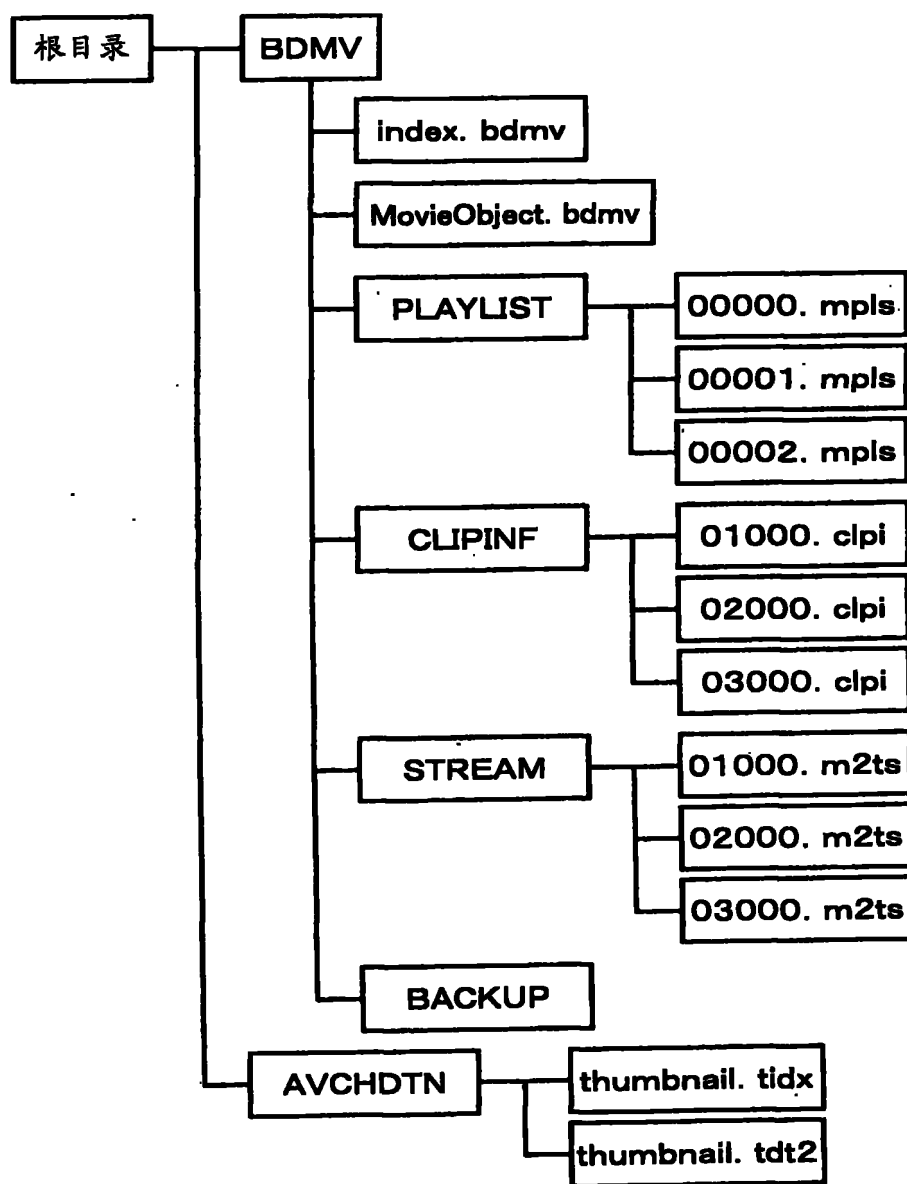


图 5

语法	数据长度 (位)	助记符
Index table file {		
TypeIndicator	8 * 4	bslbf
TypeIndicator2	8 * 4	bslbf
IndexesStartAddress	32	uimsbf
ExtensionDataStartAddress	32	uimsbf
reserved	192	bslbf
blkAppInfoBDMV()		
for (i=0; i<N1; i++) {		
padding_word	16	bslbf
}		
blkIndexes()		
for (i=0; i<N2; i++) {		
padding_word	16	bslbf
}		
blkExtensionData()		
for (i=0; i<N3; i++) {		
padding_word	16	bslbf
}		
}		

图 6

语法	数据长度 (位)	助记符
blkIndexes() {		
Length	32	uimsbf
FirstPlaybackTitle() {		
reserved	1	bslbf
'1'	1	bslbf
reserved	31	bslbf
'1'	1	bslbf
reserved	14	bslbf
FirstPlaybackTitleMobjIDRef	16	uimsbf
reserved	32	bslbf
}		
MenuTitle() {		
reserved	1	bslbf
'1'	1	bslbf
reserved	31	bslbf
'1'	1	bslbf
reserved	14	bslbf
MenuTitleMobjIDRef	16	uimsbf
reserved	32	bslbf
}		
NumberOfTitles	16	uimsbf
for (title_id=0; title_id < NumberOfTitles; title_id++) {		
MovieTitle[title_id]() {		
reserved	1	bslbf
'1'	1	bslbf
reserved	46	bslbf
MovieTitleMobjIDRef[title_id]	16	uimsbf
reserved	32	bslbf
}		
}		
}		

图 7

语法	数据长度 (位)	助记符
MovieObject file {		
TypeIndicator	8 * 4	bslbf
TypeIndicator2	8 * 4	bslbf
ExtensionDataStartAddress	32	uimsbf
reserved	224	bslbf
blkMovieObjects()		
for (i=0; i<N1; i++) {		
padding_word	16	bslbf
}		
blkExtensionData()		
for (i=0; i<N2; i++) {		
padding_word	16	bslbf
}		
}		

图 8

语法	数据长度 (位)	助记符
blkMovieObjects() {		
Length	32	uimsbf
reserved	32	bslbf
NumberOfMobjs	16	uimsbf
for (mobj_id=0; mobj_id<NumberOfMobjs; mobj_id++) {		
MovieObject[mobj_id]() {		
TerminalInfo() {		
'1'	1	bslbf
reserved	15	bslbf
}		
NumberOfNavigationCommands[mobj_id]	16	uimsbf
for (command_id=0;		
command_id<NumberOfNavigationCommands[mobj_id];		
command_id++){		
NavigationCommand[mobj_id][command_id]	96	bslbf
}		
}		
}		
}		

图 9

语法	数据长度 (位)	助记符
Movie PlayList file {		
TypeIndicator	8 * 4	bslbf
TypeIndicator2	8 * 4	bslbf
PlayListStartAddress	32	uimsbf
PlayListMarkStartAddress	32	uimsbf
ExtensionDataStartAddress	32	uimsbf
reserved	160	bslbf
blkAppInfoPlayList()		
for (i=0; i<N1; i++) {		
padding_word	16	bslbf
}		
blkPlayList()		
for (i=0; i<N2; i++) {		
padding_word	16	bslbf
}		
blkPlayListMark()		
for (i=0; i<N3; i++) {		
padding_word	16	bslbf
}		
blkExtensionData()		
for (i=0; i<N4; i++) {		
padding_word	16	bslbf
}		
}		

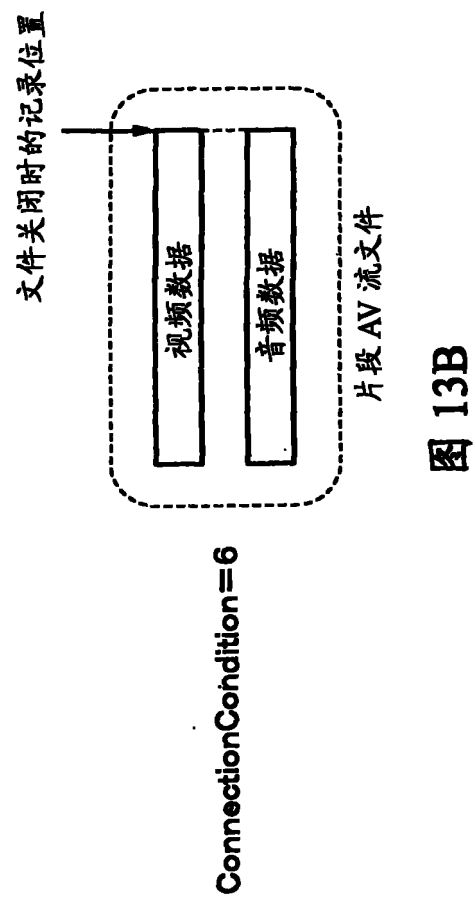
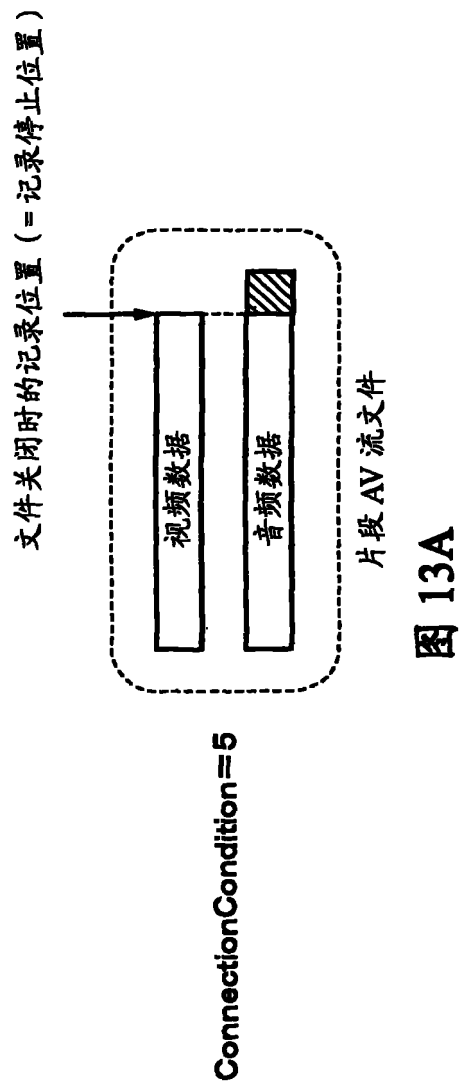
图 10

语法	数据长度 (位)	助记符
blkPlaylist() {		
Length	32	uimsbf
reserved	16	bslbf
NumberOfPlayItems	16	uimsbf
NumberOfSubPaths	16	uimsbf
for(<i>PlayItem_id</i> =0; <i>PlayItem_id</i> < <i>NumberOfPlayItems</i> ; <i>PlayItem_id</i> ++) {		
blkPlayItem()		
}		
for(<i>SubPath_id</i> =0; <i>SubPath_id</i> < <i>NumberOfSubPaths</i> ; <i>SubPath_id</i> ++){		
blkSubPath()		
}		
}		

图 11

语法	数据长度 (位)	助记符
blkPlayItem() {		
Length	16	uimsbf
ClipInformationFileName	8 * 5	bslbf
ClipCodecIdentifier	8 * 4	bslbf
reserved	12	bslbf
ConnectionCondition	4	bslbf
RefToSTCID	8	uimsbf
INTime	32	uimsbf
OUTTime	32	uimsbf
blkUOMaskTable()		
PlayItemRandomAccessFlag	1	bslbf
reserved	7	bslbf
StillMode	8	bslbf
if (StillMode== 0x01){		
StillTime	16	uimsbf
} else {		
reserved	16	bslbf
}		
blkSTNTable()		
}		

图 12



语法	数据长度 (位)	助记符
blkPlayListMark() {		
Length	32	uimsbf
NumberOfPlayListMarks	16	uimsbf
for(PL_mark_id=0; PL_mark_id< NumberOfPlayListMarks; PL_mark_id++) {		
reserved	8	bslbf
MarkType	8	bslbf
RefToPlayItemID	16	uimsbf
MarkTimeStamp	32	uimsbf
EntryESPID	16	uimsbf
Duration	32	uimsbf
}		
}		

图 14

语法	数据长度 (位)	助记符
Clip information file {		
TypeIndicator	8 * 4	bslbf
TypeIndicator2	8 * 4	bslbf
SequenceInfoStartAddress	32	uimsbf
ProgramInfoStartAddress	32	uimsbf
CPIStartAddress	32	uimsbf
ClipMarkStartAddress	32	uimsbf
ExtensionDataStartAddress	32	uimsbf
reserved	96	bslbf
blkClipInfo()		
for (i=0; i<N1; i++) {		
padding_word	16	bslbf
}		
blkSequenceInfo()		
for (i=0; i<N2; i++) {		
padding_word	16	bslbf
}		
blkProgramInfo()		
for (i=0; i<N3; i++) {		
padding_word	16	bslbf
}		
blkCPI()		
for (i=0; i<N4; i++) {		
padding_word	16	bslbf
}		
blkClipMark()		
for (i=0; i<N5; i++) {		
padding_word	16	bslbf
}		
blkExtensionData()		
for (i=0; i<N6; i++) {		
padding_word	16	bslbf
}		
}		

图 15

语法	数据长度 (位)	助记符
blkClipInfo() {		
Length	32	uimsbf
reserved	16	bslbf
ClipStreamType	8	bslbf
ApplicationType	8	bslbf
reserved	31	bslbf
IsCC5	1	bslbf
TSRecordingRate	32	uimsbf
NumberOfSourcePackets	32	uimsbf
reserved	1024	bslbf
TSTypeInfoBlock()		
if (IsCC5 == 1b) {		
reserved	8	bslbf
FollowingClipStreamType	8	bslbf
reserved	32	bslbf
FollowingClipInformationFileName	8 * 5	bslbf
ClipCodeIdentifier	8 * 4	bslbf
reserved	8	bslbf
}		
}		

图 16

语法	数据长度 (位)	助记符
blkSequenceInfo() {		
Length	32	uimsbf
reserved	15	bslbf
'1'	1	bslbf
SPNATCStart	32	uimsbf
NumberOfSTCSequences	8	uimsbf
reserved	8	uimsbf
for (stc_id=0; stc_id<NumberOfSTCSequences; stc_id++) {		
PCRPID[stc_id]	16	uimsbf
SPNSTCStart[stc_id]	32	uimsbf
PresentationStartTime[stc_id]	32	uimsbf
PresentationEndTime[stc_id]	32	uimsbf
}		
}		

图 17

语法	数据长度 (位)	助记符
blkProgramInfo() {		
Length	32	uimsbf
reserved	15	bslbf
'1'	1	bslbf
SPNProgramSequenceStart	32	uimsbf
ProgramMapPID	16	uimsbf
NumberOfStreamsInPS	8	uimsbf
reserved	8	bslbf
for (<i>stream_index</i> =0; <i>stream_index</i> < <i>NumberOfStreamsInPS</i> ; <i>stream_index</i> ++) {		
StreamPID[stream_index]	16	uimsbf
blkStreamCodingInfo(stream_index)		
}		
}		

图 18

语法	数据长度 (位)	助记符
blkCPI() {		
Length	32	uimsbf
if(<i>Length</i> !=0) {		
reserved	12	bslbf
CPIType	4	bslbf
blkEPMap()		
}		
}		

图 19

语法	数据长度 (位)	助记符
blkEPMAP(){		
reserved	8	bslbf
NumberOfStreamPIDEntries	8	uimsbf
for (k=0; k<NumberOfStreamPIDEntries; k++) {		
StreamPID[k]	16	bslbf
reserved	10	bslbf
EPStreamType[k]	4	bslbf
NumberOfEPCoarseEntries[k]	16	uimsbf
NumberOfEPFineEntries[k]	18	uimsbf
EPMAPForOneStreamPIDStartAddress[k]	32	uimsbf
}		
for (i=0; i<X; i++) {		
padding_word	16	bslbf
}		
for (k=0; k<NumberOfStreamPIDEntries; k++) {		
blkEPMAPForOneStreamPID(EPStreamType[k], NumberOfEPCoarseEntries[k], NumberOfEPFineEntries[k])		
for (i=0; i<Y[k]; i++) {		
padding_word	16	bslbf
}		
}		
}		

图 20

语法	数据长度 (位)	助记符
blkEPMapForOneStreamPID(<i>EP_stream_type</i>, <i>Nc</i>, <i>Nf</i>) {		
EPFineTableStartAddress	32	uimsbf
for (i=0; i<<i>Nc</i>; i++) {		
// EP coarse table		
RefToEPFineID[i]	18	uimsbf
PTSEPCoarse[i]	14	uimsbf
SPNEPCoarse[i]	32	uimsbf
}		
for (i=0; i<<i>X</i>; i++) {		
padding_word	16	bslbf
}		
for (EP_fine_id =0; EP_fine_id < <i>Nf</i>, EP_fine_id++) {		
// EP fine table		
ReservedEPFine[EP_fine_id]	1	bslbf
!EndPositionOffset[EP_fine_id]	3	bslbf
PTSEPFine[EP_fine_id]	11	uimsbf
SPNEPFine[EP_fine_id]	17	uimsbf
}		
}		

图 21

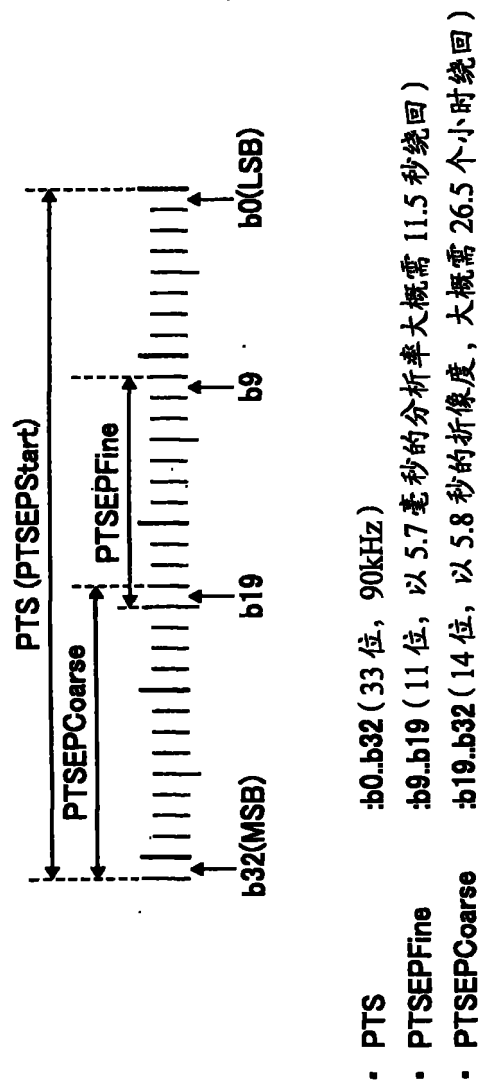
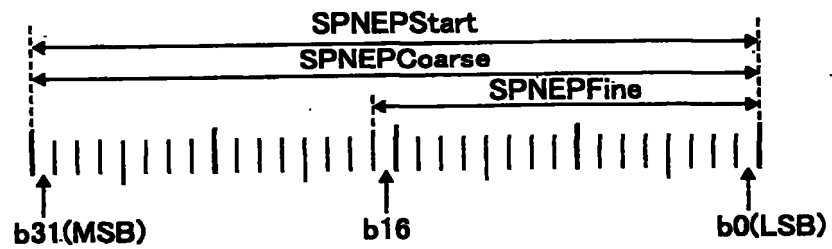


图 22



- **SPNEPStart** :b0..b31 (32 位)
- **SPNEPFine** :b0..b16 (17 位, 在 AV 流文件中大概以 25 兆字节绕回)
- **SPNEPCoarse** :b0..b31 (32 位)

图 23

语法	数据长度 (位)	助记符
blkExtensionData() {		
Length	32	uimsbf
if(Length !=0){		
DataBlockStartAddress	32	uimsbf
reserved	24	
NumberOfExtDataEntries	8	uimsbf
for (i=0; i<NumberOfExtDataEntries; i++) {		
ext_data_entry() {		
ExtDataType	16	uimsbf
ExtDataVersion	16	uimsbf
ExtDataStartAddress	32	uimsbf
ExtDataLength	32	uimsbf
}		
}		
for (i=0; i<L-1; i++) {		
padding_word	16	bslbf
padding_word	16	bslbf
}		
DataBlock()	32 + 8 * (Length - DataBlockStartAddress)	
}		
}		

图 24

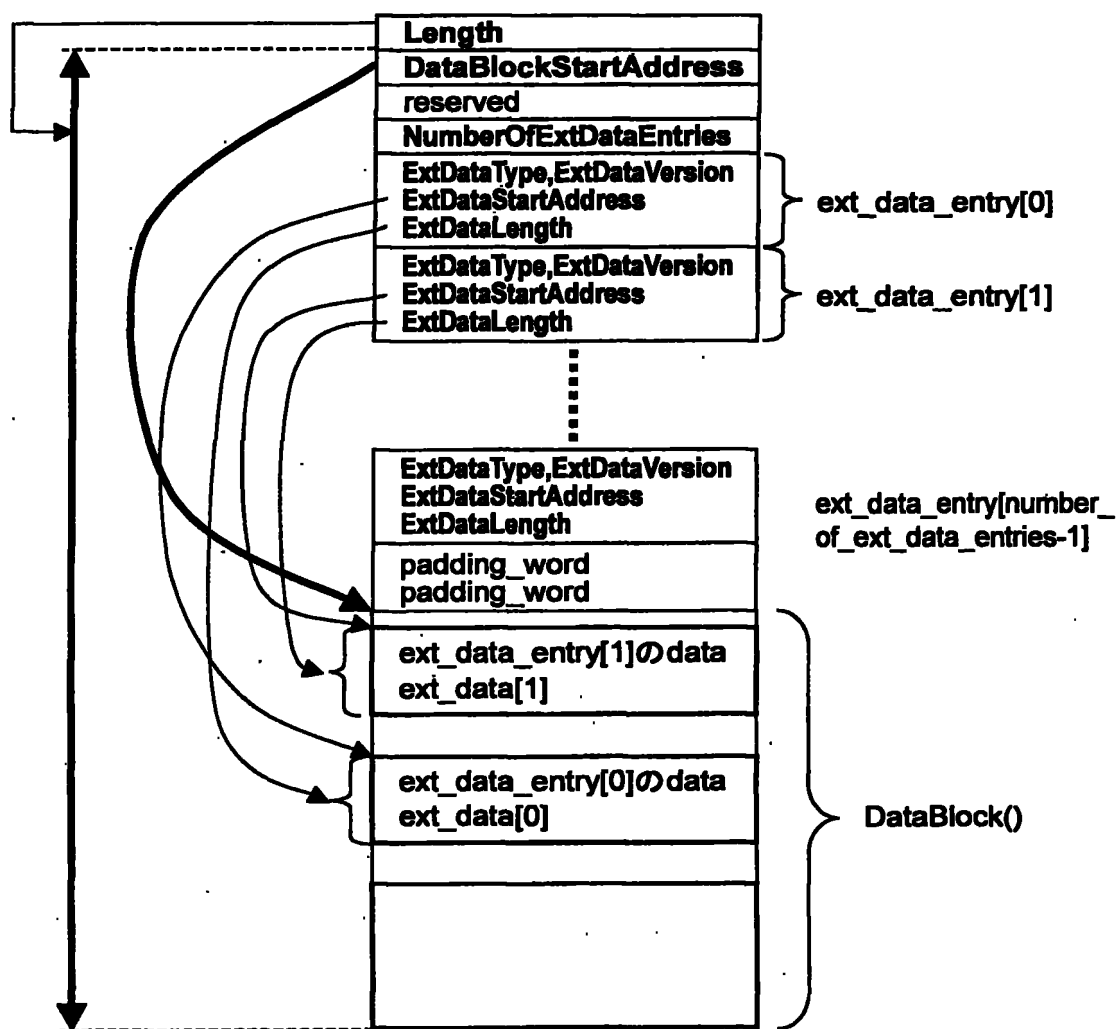


图 25

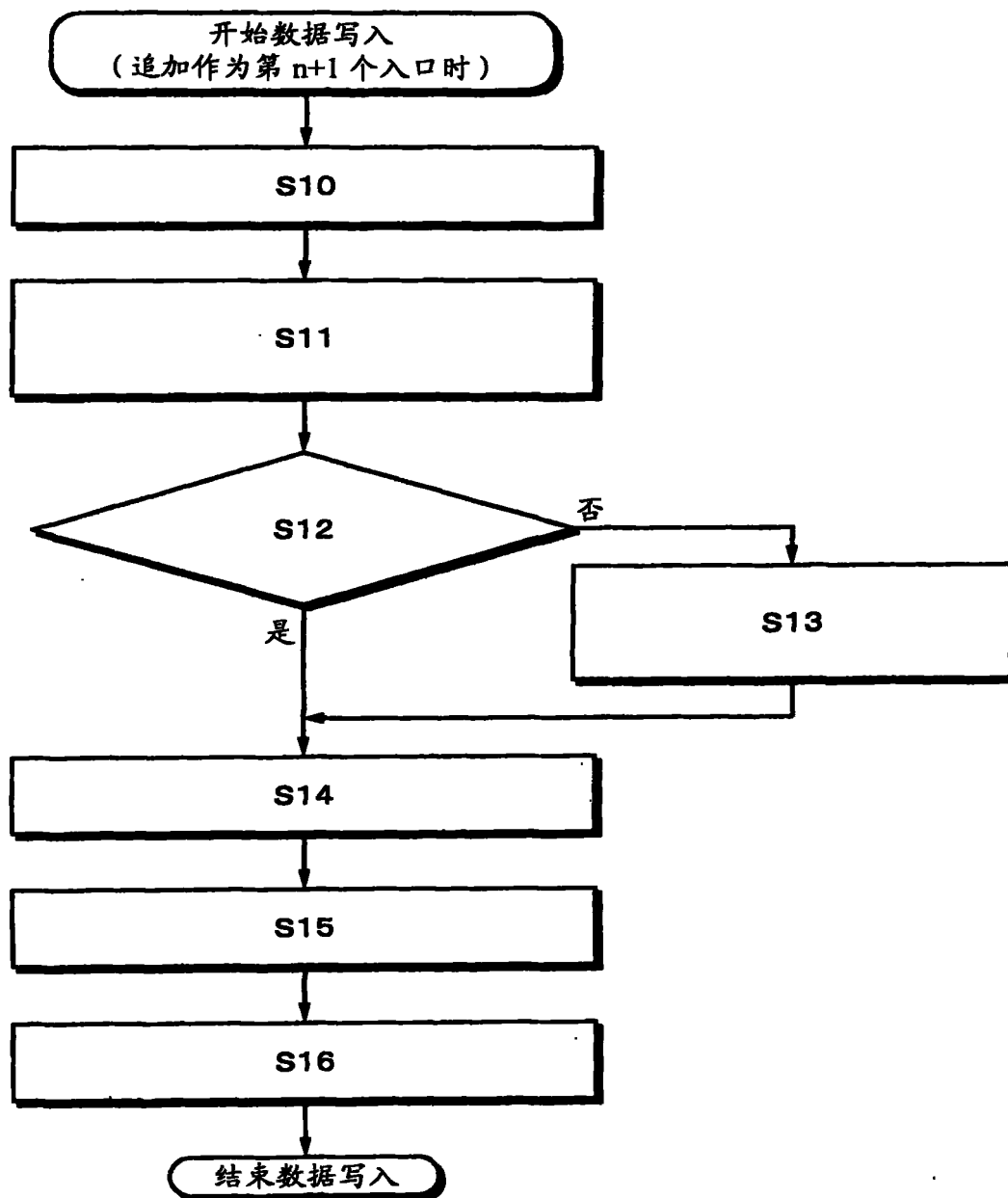


图 26

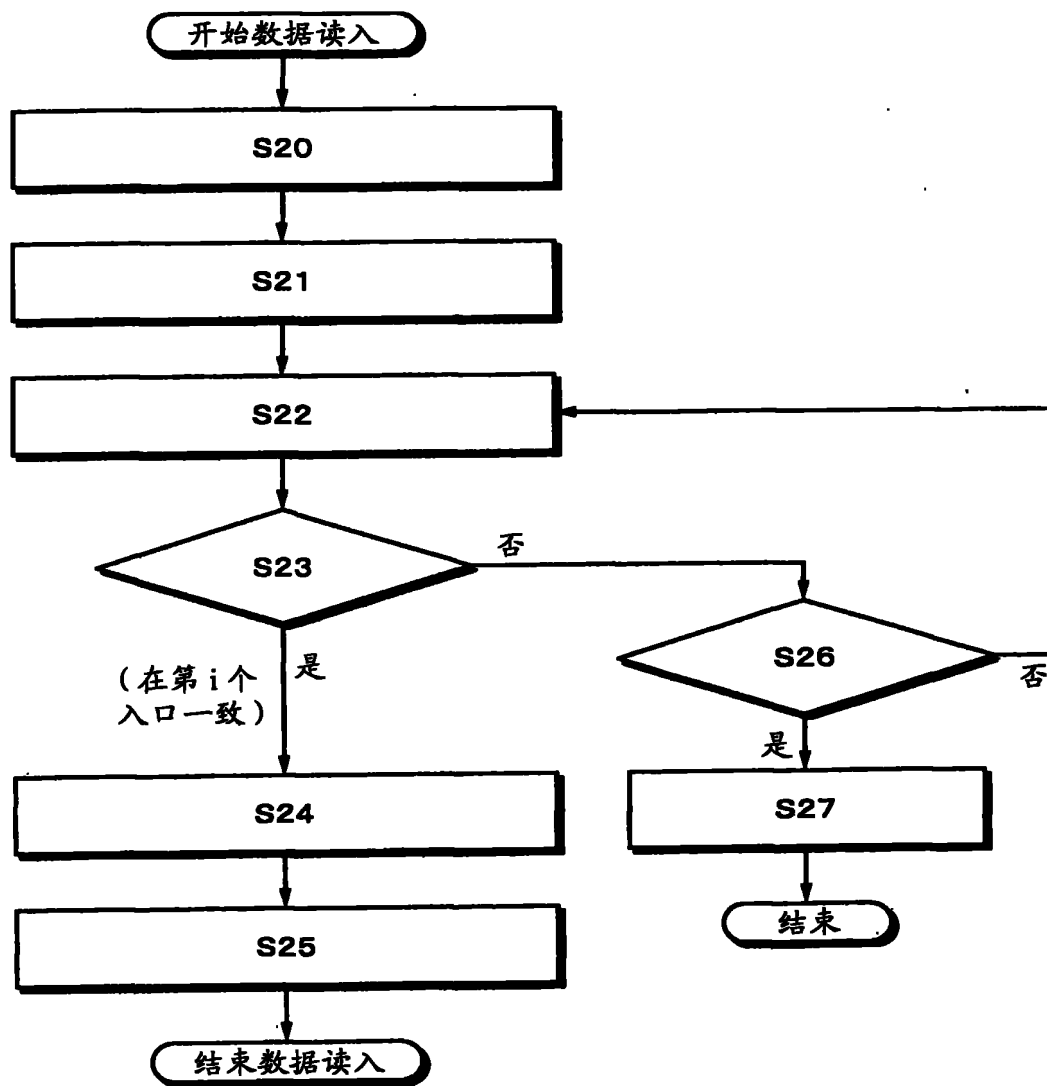


图 27

语法	数据长度 (位)	助记符
blkIndexExtensionData(){		
TypeIndicator	8 * 4	uimsbf
reserved	8 * 4	bslbf
TableOfPlayListsStartAddress	32	uimsbf
MakersPrivateDataStartAddress	32	uimsbf
reserved	192	bslbf
blkUIAppInfoAVCHD()		
for(i=0; i<N1;i++) {		
padding_word	16	bslbf
}		
blkTableOfPlayLists()		
for(i=0; i<N2;i++) {		
padding_word	16	bslbf
}		
blkMakersPrivateData()		
for (i=0; i<N3;i++){		
padding_word	16	bslbf
}		
}		

图 28

语法	数据长度 (位)	助记符
blkTableOfPlayLists() {		
Length	32	uimsbf
blkFirstPlaybackTitlePlayLists()		
blkMenuTitlePlayLists()		
NumberOfTitlePlayListPair	16	bslbf
for(i=0; i< NumberOfTitlePlayListPair;i++){		
blkMovieTitlePlayListPair() {		
PlayListFileName	8 * 5	bslbf
reserved	6	bslbf
PlayListAttribute	2	uimsbf
reserved	16	bslbf
RefToTitleId	16	uimsbf
}		
}		
}		

图 29

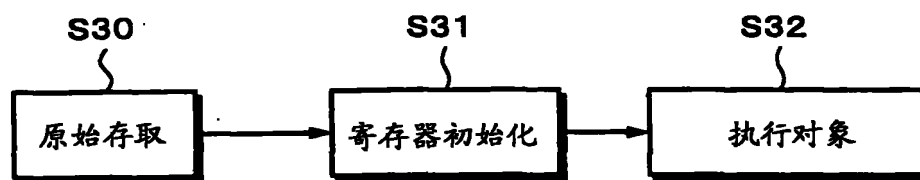


图 30A

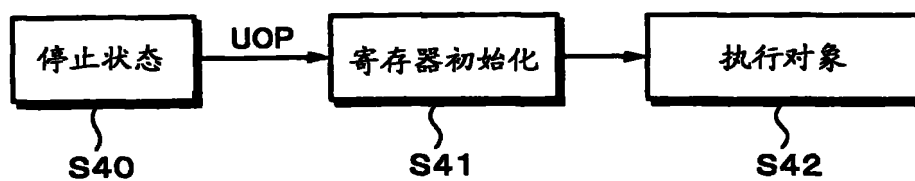


图 30B

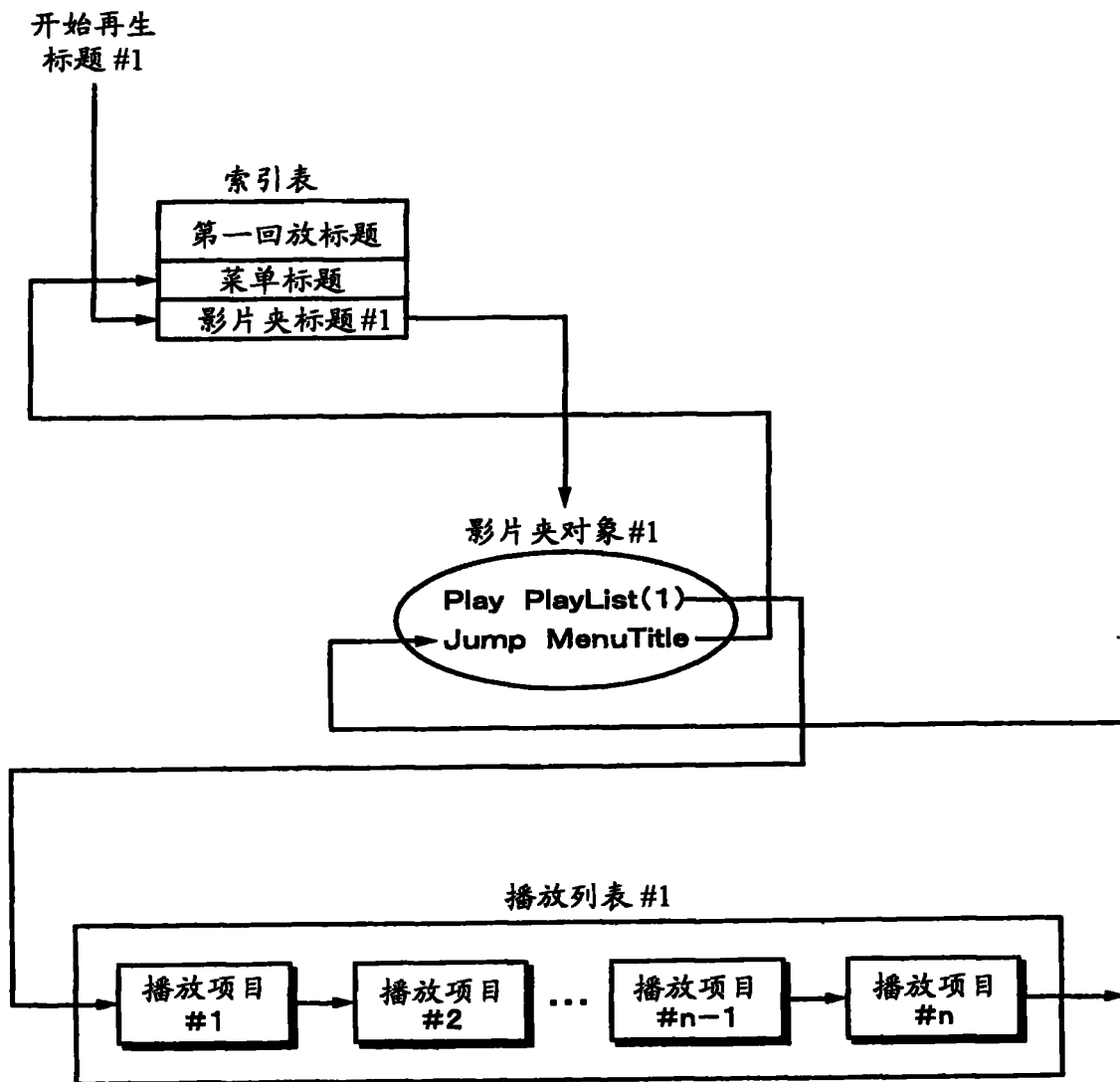


图 31

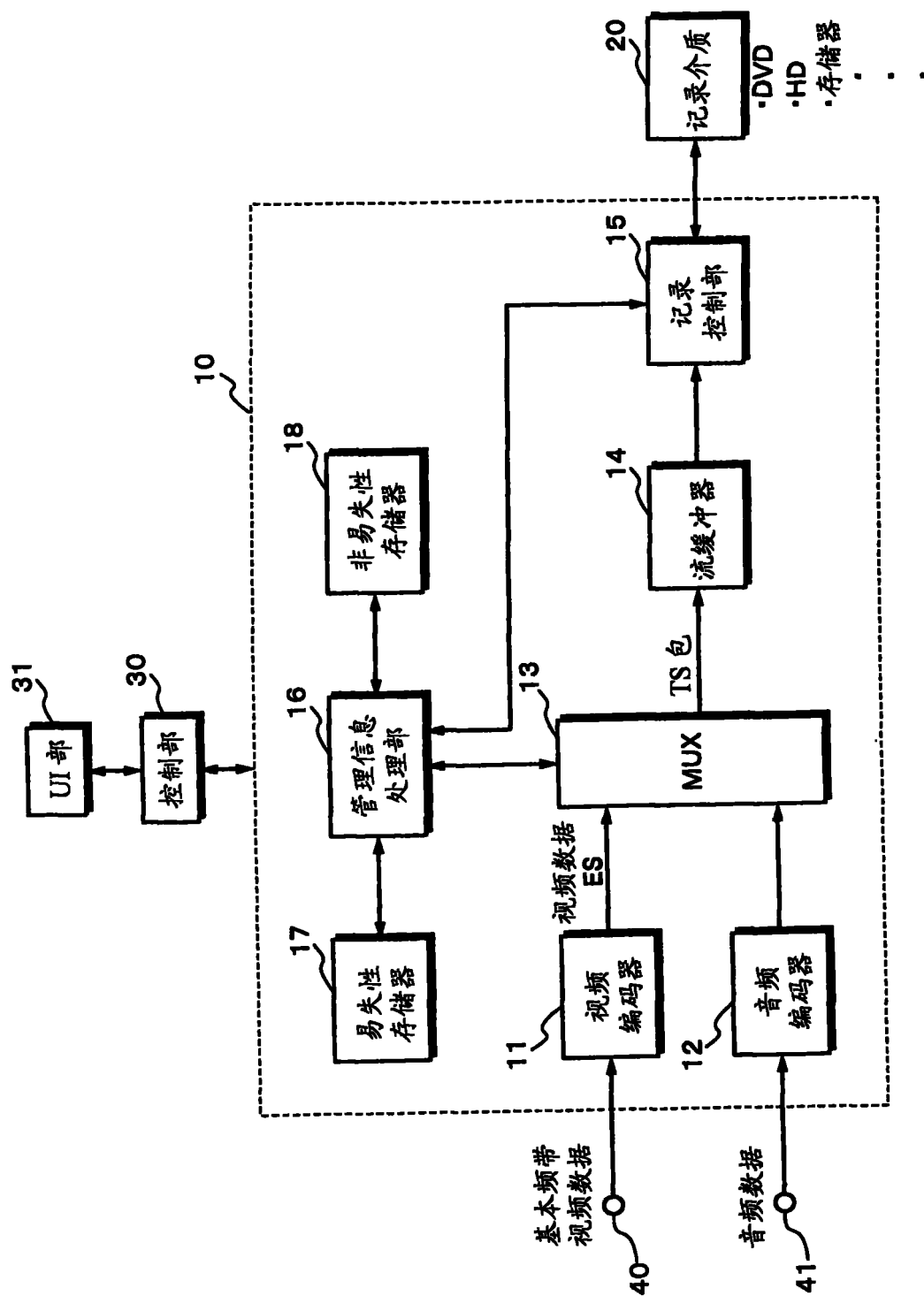


图 32

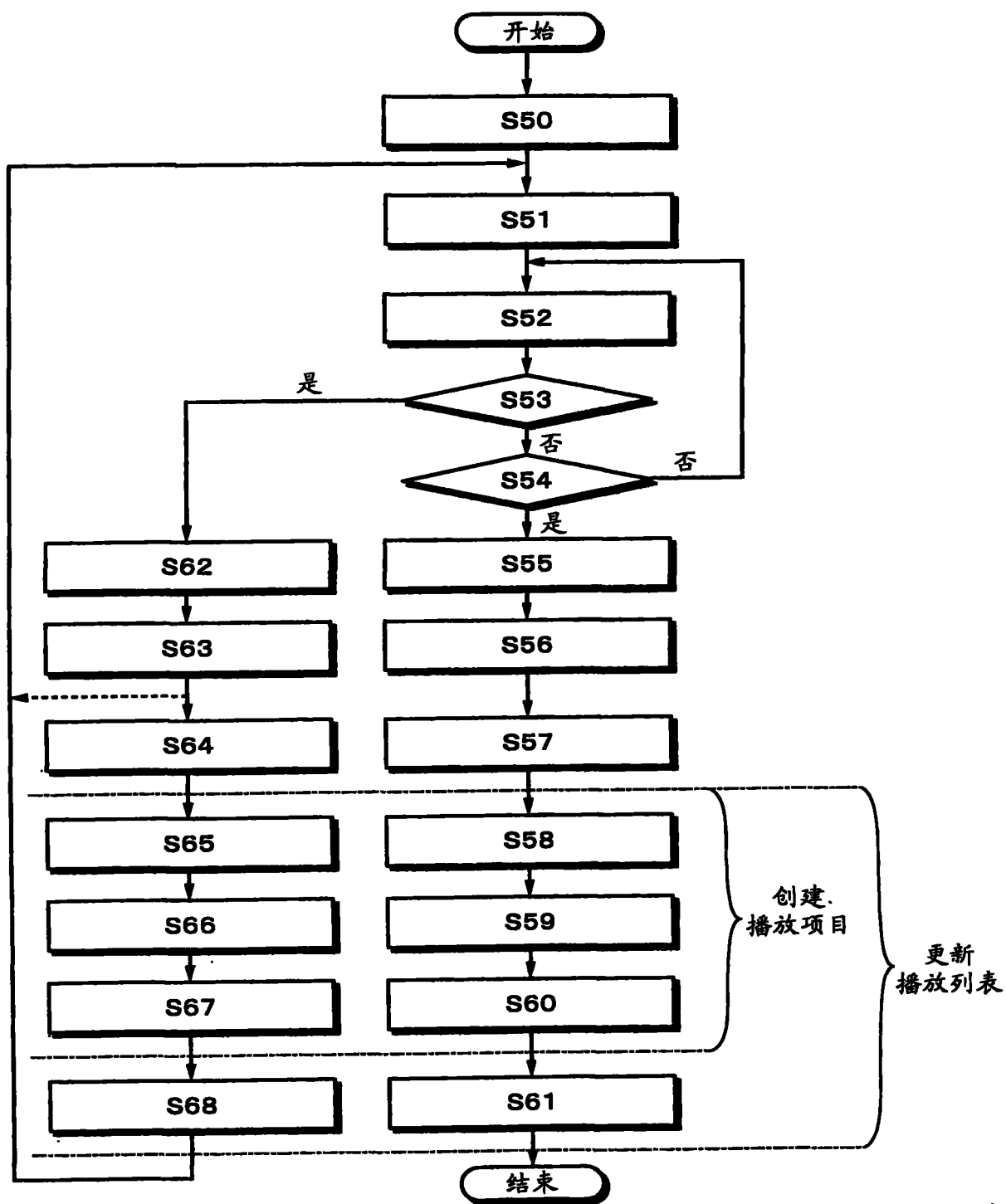


图 33

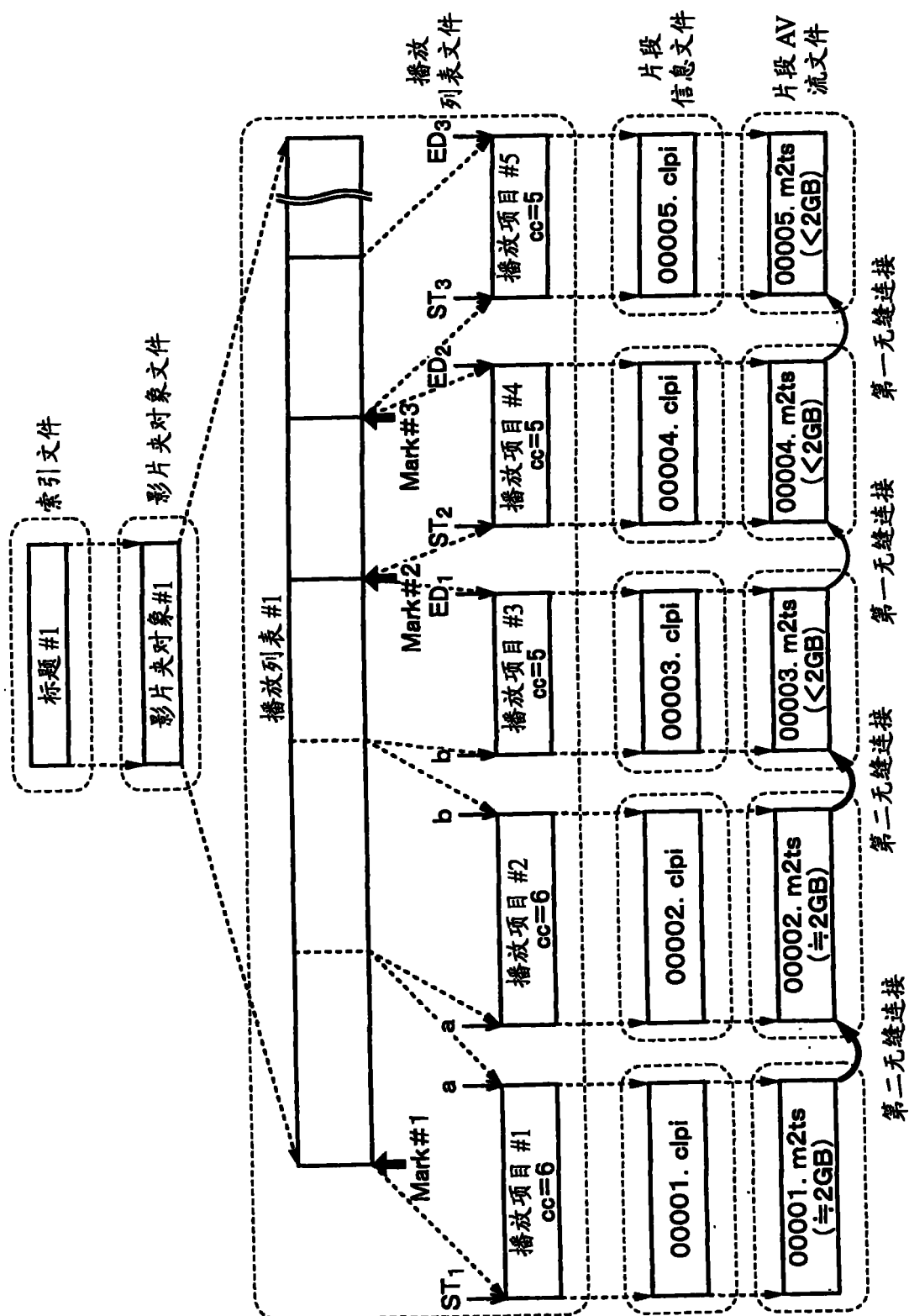


图 34

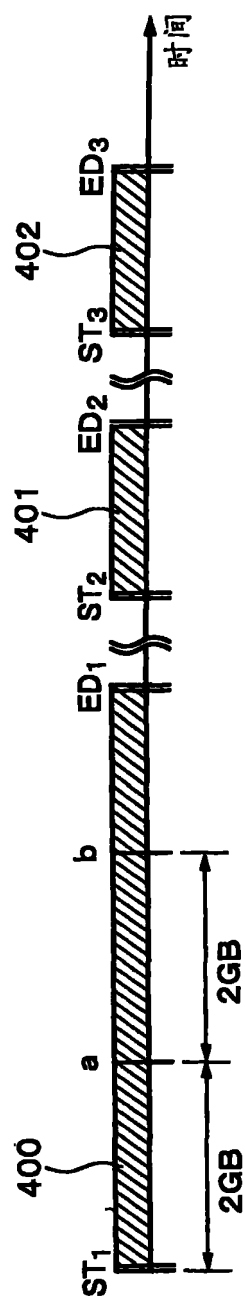


图 35A

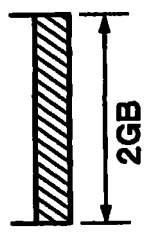


图 35B

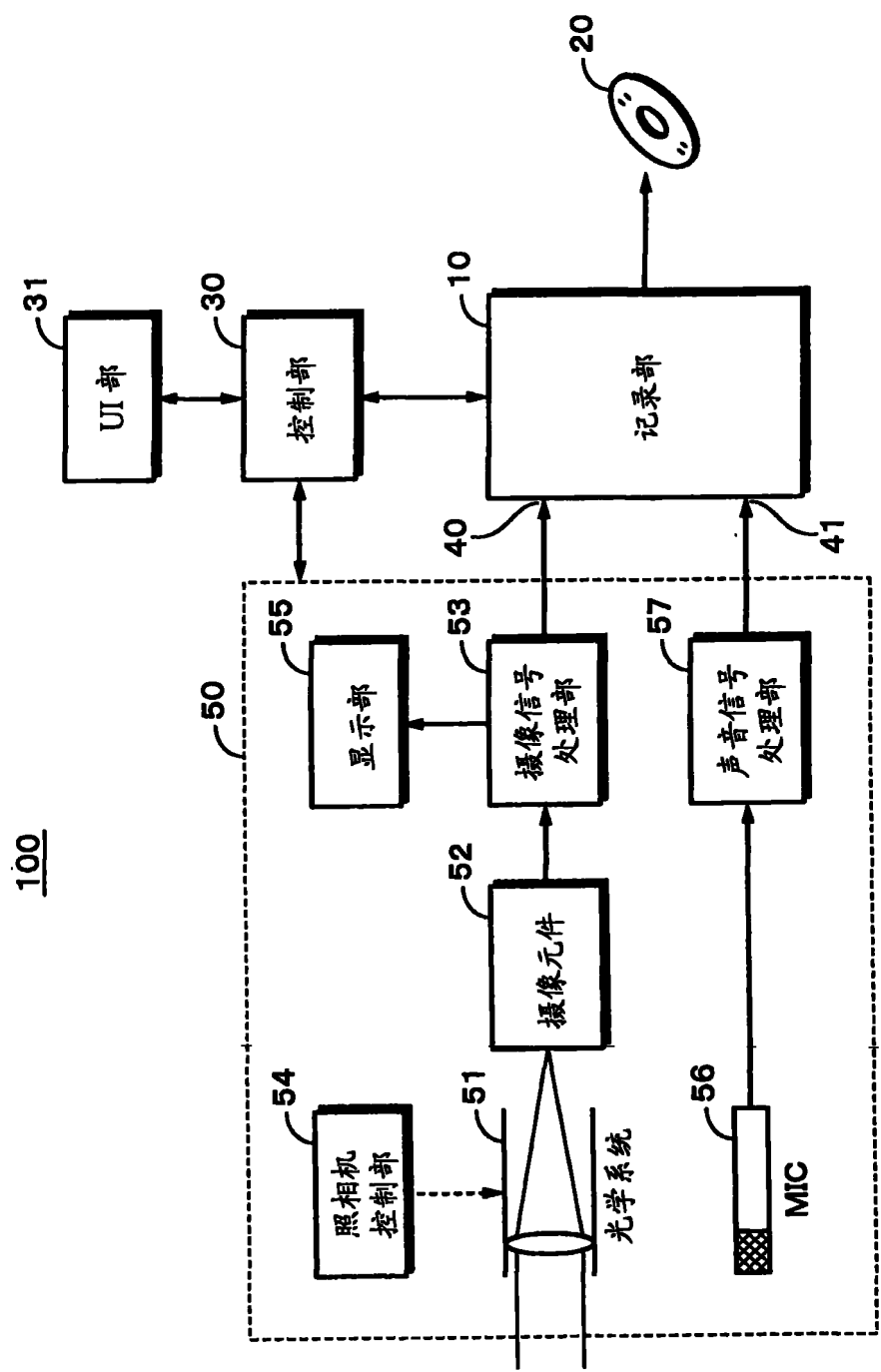


图 36

符号说明
10 记录部

11 视频编码器

12 音频编码器

13 多路复用器

14 流缓冲器

15 记录控制部

16 管理信息处理部

17 易失性存储器

18 非易失性存储器

20 记录介质

30 控制部

31 用户界面部

50 照相机部

100 摄像机装置

S10 获取想要记录的数据的长度,并设置为域 ExtDataLength[n+1] 的值

S11 根据当前块 blkExtensionData() 中列出的块 ext_data_entry(), 调查域 ExtDataLength 和域 ExtDataStartAddress, 并获知块 DataBlock() 中的使用状况

S12 在 DataBlock 中存在 ExtDataLength(n+1) 以上的连续空区域?

S13 增大 blkExtensionData() 的 Length 的值, 创建 ExtDataLength(n+1) 以上的连续空区域

S14 确定存储数据的区域的开头地址, 将其作为 ExtDataStartAddress(n+1)

S15 从 ExtDataStartAddress(n+1) 所示的地址写入 ExtDataLength(n+1) 长度的数据

S16 对 ext_data_entry() 追加 ExtDataLength(n+1) 和 ExtDataStartAddress(n+1)

S20 根据以想要读入的数据为基准的标准, 获知 ExtDataType

S21 基于 ExtDataType, 根据想要读入的数据的种类, 获知 ExtDataVersion

S22 逐个依次读入块 blkExtensionData() 中列举的块 ext_data_entry()

S23 ExtDataType、ExtDataVersion 是否一致?

S24 读入 ExtDataLength(i) 和 ExtDataStartAddress(i)

S25 从 ExtDataStartAddress(i) 所示的地址读入 ExtDataLength(i) 长度的数据

S26 已读完 ext_data_entry() ?

S27 想要读入的数据不存在

S50 记录开始操作

S51 新创建片段 AV 流文件

S52 将流记录于片段 AV 流文件

S53 超过 2GB ?

S54 记录停止操作?

S55 将存入流缓冲器的流写入文件

S56 关闭片段 AV 流文件

S57 创建片段信息文件

S58 设置 ConnectionCondition = 5

S59 将 NumberOfPlayItems+1

S60 设置 ClipInformationFileName、INTime、OUTTime,并创建 blkPlayItem()

S61 将 MarkTimeStamp 设定为 INTime,对 blkPlayListMark 增加 1

S62 将存入流缓冲器的流写入文件

S63 关闭片段 AV 流文件

S64 创建片段信息文件

S65 设置 ConnectionCondition = 6

S66 对 NumberOfPlayItems+1

S67 设定 ClipInformationFileName、INTime、OUTTime,并创建 blkPlayItem()

S68(追加播放列表标签)