



(12) 发明专利

(10) 授权公告号 CN 101331764 B

(45) 授权公告日 2010. 12. 22

(21) 申请号 200780000722. 5

(22) 申请日 2007. 05. 10

(30) 优先权数据

138701/2006 2006. 05. 18 JP

(85) PCT申请进入国家阶段日

2008. 01. 18

(86) PCT申请的申请数据

PCT/JP2007/060081 2007. 05. 10

(87) PCT申请的公布数据

W02007/135932 JA 2007. 11. 29

(73) 专利权人 索尼株式会社

地址 日本东京都

(72) 发明人 前笃 有留宪一郎 矶部幸雄

森本直树

(74) 专利代理机构 北京市柳沈律师事务所

11105

代理人 黄小临

(51) Int. Cl.

H04N 5/91 (2006. 01)

G11B 27/00 (2006. 01)

H04N 5/225 (2006. 01)

H04N 5/76 (2006. 01)

(56) 对比文件

CN 1698124 A, 2005. 11. 16, 全文.

JP 特开 2005-317076 A, 2005. 11. 10, 全文.

JP 特开 2005-27159 A5, 2005. 01. 27, 全文.

US 20040240848 A1, 2004. 11. 02, 全文.

W0 2006030767 A1, 2006. 03. 23, 全文.

审查员 于雷

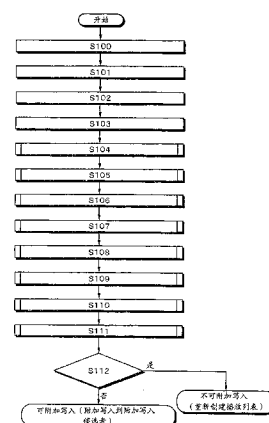
权利要求书 8 页 说明书 45 页 附图 54 页

(54) 发明名称

记录装置、记录方法、记录程序、摄像装置、摄像方法和摄像程序

(57) 摘要

将在记录的开始到记录期间生成的 AV 数据记录为文件,可改善用户友好性。将在记录的开始到记录期间生成的 AV 数据记录为文件,并且将表示再现部分的信息和表示跳转位置的标记信息附加写入到再现列表文件。这里,在附加写入之前,根据将作为附加可写候选者的再现列表文件和 AV 数据的再现时间和地址相关联的属性文件,进行判断,以决定是否满足对再现列表文件和属性文件所设置的约束。如果满足,则将再现部分信息和标记信息附加写入到作为附加可写候选者的再现列表文件。如果不满足,则生成新的再现列表文件。用户可执行 AV 数据的记录操作,而无需考虑对再现列表文件和属性文件所设置的约束。



1. 一种多路复用视频数据和音频数据并且将多路复用数据记录到记录介质中的记录设备,其特征在于

所述记录设备包括:

数据输入部分,其输入所述视频数据和所述音频数据;

记录命令输入部分,其输入用于所述视频数据和所述音频数据的记录开始命令和记录停止命令;

记录部分,其多路复用所述视频数据和所述音频数据,并将经多路复用的流作为流文件而记录到所述记录介质中;以及

管理信息创建部分,其创建用于控制在所述记录介质上记录的所述流文件的再现的再现管理信息,所述再现管理信息由表示所述流文件的属性信息的第一管理信息和包含表示用于所述流文件的再现方法的信息的第二管理信息组成,以及

所述管理信息创建部分基于所述再现管理信息的现有条目,确定是否根据所述记录命令输入部分的所述记录开始命令而将表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息附加写入到所述再现管理信息的所述现有条目中所包含的所述第二管理信息的预定条目中,

其中所述第一管理信息由存储至少表示所述流文件的再现时间信息和地址信息的关系的信息的文件组成,

所述第二管理信息由包括再现区域信息的至少一个条目的文件组成,所述再现区域信息指定具有对所述流文件设置的再现开始点和再现结束点的再现区域,并且所述再现区域信息能够存储表示所述流文件的再现时间信息的标记信息,以及

所述第二管理信息从所述再现区域信息参照所述第一管理信息,来表示用于所述流文件的所述再现方法。

2. 根据权利要求1所述的记录设备,其特征在于

当所述管理信息创建部分确定不根据所述记录命令输入部分的所述记录开始命令而将表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息附加写入到在所述再现管理信息的所述现有条目中所包含的所述第二管理信息的所述预定条目中时,所述管理信息创建部分根据所述记录命令输入部分的所述记录开始命令,新创建包括表示用于在所述记录介质上记录的所述流文件的再现方法的信息的所述第二管理信息。

3. 根据权利要求1所述的记录设备,其特征在于

所述管理信息创建部分将在记录所述流文件时更新的所述第二管理信息的所述第二管理信息的最近更新的条目当作所述第二管理信息的所述预定条目。

4. 根据权利要求1所述的记录设备,其特征在于

所述管理信息创建部分的所述附加记录能/不能确定是基于在所述第二管理信息的所述预定条目中存储的所述再现区域信息的条目数目。

5. 根据权利要求4所述的记录设备,其特征在于

当所述管理信息创建部分已经确定在所述第二管理信息的所述预定条目中已存储的所述再现区域信息的条目数目小于预先设置的上限时,所述管理信息创建部分执行所述附加记录能/不能确定,使得表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息被附加写入到所述第二管理信息的所述预定条目中。

6. 根据权利要求 1 所述的记录设备,其特征在于

所述管理信息创建部分的所述附加记录能 / 不能确定是基于在所述第二管理信息的所述预定条目中存储的所述标记信息的条目数目。

7. 根据权利要求 6 所述的记录设备,其特征在于

当所述管理信息创建部分已经确定存储在所述第二管理信息的所述预定条目中的所述标记信息的条目数目小于预先设置的上限时,所述管理信息创建部分执行所述附加记录能 / 不能确定,使得表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息被附加写入到所述第二管理信息的所述预定条目中。

8. 根据权利要求 1 所述的记录设备,其特征在于

所述第一管理信息还存储表示在相应的流文件中存储的所述视频数据的属性的视频属性信息,以及

所述管理信息创建部分的所述附加记录能 / 不能确定是基于在所述第一管理信息中存储的所述视频属性信息。

9. 根据权利要求 8 所述的记录设备,其特征在于

所述管理信息创建部分的所述附加记录能 / 不能确定是基于根据当前记录模式的视频属性和在从所述第二管理信息的所述预定条目参照的所述第一管理信息的至少一个条目中存储的视频属性这两者。

10. 根据权利要求 9 所述的记录设备,其特征在于

当基于所述记录模式的所述视频属性的帧大小、纵横比、帧速率和扫描系统中的至少一个与在所述第一管理信息的所述条目中存储的所述视频属性的所述帧大小、纵横比、帧速率和扫描系统中的所述至少一个不匹配时,所述管理信息创建部分执行所述附加记录能 / 不能确定,使得根据所述记录命令输入部分的所述记录开始命令不将表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息附加写入到所述第二管理信息的所述预定条目。

11. 根据权利要求 1 所述的记录设备,其特征在于

所述管理信息创建部分的所述附加记录能 / 不能确定是基于所述第二管理信息的所述预定条目的文件大小。

12. 根据权利要求 11 所述的记录设备,其特征在于

当所述管理信息创建部分已经基于所述再现区域信息和在已记录所述单个流文件时所创建的所述标记信息的数据大小而确定所述第二管理信息的所述文件大小没有超过预先设置的上限时,所述管理信息创建部分执行所述附加记录能 / 不能确定,使得表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息被附加写入到所述第二管理信息的所述预定条目中。

13. 根据权利要求 1 所述的记录设备,其特征在于

所述管理信息创建部分的所述附加记录能 / 不能确定是基于从所述第二管理信息的所述预定条目参照的所述第一管理信息的所有条目的总文件大小。

14. 根据权利要求 13 所述的记录设备,其特征在于

当所述管理信息创建部分已经基于在所确保的最小连续时间内在记录所述单个流文件时所创建的所述第一管理信息的数据大小而确定从所述第二管理信息的所述预定条目

参照的所述第一管理信息的所述总文件大小没有超过预先设置的上限时,所述管理信息创建部分执行所述附加记录能 / 不能确定,使得表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息被附加写入到所述第二管理信息的所述预定条目中。

15. 根据权利要求 1 所述的记录设备,其特征在于

所述管理信息创建部分的所述附加记录能 / 不能确定是基于表示在从所述第二管理信息的所述预定条目参照的所述第一管理信息的所有条目内包含的再现时间信息和地址信息的关系的信息的条目数目。

16. 根据权利要求 15 所述的记录设备,其特征在于

当所述管理信息创建部分已经基于表示在确保的最小连续时间内在已记录所述单个流文件时所创建的所述第一管理信息内包含的所述再现时间信息和所述地址信息的所述信息的条目数目而确定从所述第二管理信息参照的所述第一管理信息的所有条目的总数没有超过所述第二管理信息的所述条目的预先设置的上限时,所述管理信息创建部分执行所述附加记录能 / 不能确定,以便表示用于记录在所述记录介质上的所述流文件的所述再现方法的所述信息附加写入到所述第二管理信息的所述预定条目中。

17. 根据权利要求 1 所述的记录设备,其特征在于

所述第二管理信息能够存储唯一的设备信息,以及

所述管理信息创建部分的所述附加记录能 / 不能确定是基于在所述第二管理信息的所述预定条目中存储的所述唯一的设备信息。

18. 根据权利要求 17 所述的记录设备,其特征在于

如果所述管理信息创建部分已经在所述第二管理信息的所述预定条目中存储了所述唯一的设备信息时,确定已经由所述记录设备或者与其等同的另一设备创建了所述唯一的设备信息,

则所述管理信息创建部分执行所述附加记录能 / 不能确定,使得表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息被附加写入到所述第二管理信息的所述预定条目中。

19. 根据权利要求 1 所述的记录设备,其特征在于

所述第二管理信息还能够存储表示所述第二管理信息的最近更新设备的信息,以及

所述管理信息创建部分的所述附加记录能 / 不能确定是基于在所述第二管理信息的所述预定条目中存储的所述最近更新信息。

20. 根据权利要求 19 所述的记录设备,其特征在于

如果已经在所述第二管理信息的所述预定条目中存储了所述最近更新设备信息时,确定所述最近更新设备信息表示所述记录设备,则执行所述附加记录能 / 不能确定,使得表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息被附加写入到所述第二管理信息的所述预定条目中。

21. 根据权利要求 1 所述的记录设备,其特征在于

所述管理信息创建部分对一个或多个项目执行所述附加记录能 / 不能确定,以及

当所述附加记录能 / 不能确定的至少一个所述项目表示附加写入不能状态时,所述管理信息创建部分执行所述附加记录能 / 不能确定,使得与在所述记录介质上记录的所述流文件相对应的所述第二管理信息不被附加写入到所述第二管理信息的所述预定条目中。

22. 根据权利要求 21 所述的记录设备,其特征在于

所述一个或多个项目包括以下中的至少一个项目:在所述第二管理信息的所述预定条目中存储的再现区域信息的条目数目、在所述第二管理信息的所述预定条目中存储的标记信息的条目数目、在所述第一管理信息中存储的视频属性信息、所述第二管理信息的所述预定条目的文件大小、从所述第二管理信息的所述预定条目参照的所述第一管理信息的所有条目的总文件大小、表示在从所述第二管理信息的所述预定条目参照的所述第一管理信息的所有条目内包含的再现时间信息和地址信息的关系的信息的条目数目、在所述第二管理信息的所述预定条目中存储的唯一的设备信息、以及在所述第二管理信息的所述预定条目中存储的最近更新设备信息。

23. 一种多路复用视频数据和音频数据以及将多路复用的数据记录到记录介质中的记录方法,其特征在于所述记录方法包括:

记录命令输入步骤,输入用于已经输入到数据输入部分的所述视频数据和所述音频数据的记录开始命令和记录停止命令;

记录步骤,其多路复用所述视频数据和所述音频数据,并将经多路复用的流作为流文件而记录到所述记录介质中;以及

管理信息创建步骤,生成用于控制在所述记录介质上记录的所述流文件的再现的再现管理信息,所述再现管理信息由表示所述流文件的属性信息的第一管理信息和包括表示用于所述流文件的再现方法的信息的第二管理信息组成,

通过以下来执行所述管理信息创建步骤:基于所述再现管理信息的现有条目,确定是否根据所述记录命令输入部分的所述记录开始命令而将表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息附加写入到所述再现管理信息的所述现有条目中所包含的所述第二管理信息的预定条目中,

其中所述第一管理信息由存储至少表示所述流文件的再现时间信息和地址信息的关系的信息的文件组成,

所述第二管理信息由包括再现区域信息的至少一个条目的文件组成,所述再现区域信息指定具有对所述流文件设置的再现开始点和再现结束点的再现区域,并且所述再现区域信息能够存储表示所述流文件的再现时间信息的标记信息,以及

所述第二管理信息从所述再现区域信息参照所述第一管理信息,来表示用于所述流文件的所述再现方法。

24. 一种图像捕捉设备,其多路复用由图像捕捉部分从对象捕捉和获取的视频数据以及由声音收集部分收集和获取的音频数据,并且将多路复用的视频数据和音频数据记录到记录介质中,其特征在于

所述图像捕捉设备包括:

所述图像捕捉部分,其捕捉所述对象的图像并输出所述视频数据;

所述声音收集部分,其收集声音并输出音频数据,

记录部分,其多路复用所述视频数据和所述音频数据,并将经多路复用的流作为流文件而记录到所述记录介质中;以及

操作部分,其接受用于对所述记录介质的所述视频数据和所述音频数据的记录开始命令和记录停止命令的用户操作;以及

管理信息创建部分,其创建用于控制在所述记录介质上记录的所述流文件的再现的再现管理信息,所述再现管理信息由表示所述流文件的属性信息的第一管理信息和包含表示用于所述流文件的再现方法的信息的第二管理信息组成,

所述管理信息创建部分基于所述再现管理信息的现有条目,确定是否根据与对操作部分的用户操作相对应的所述记录开始命令而将表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息附加写入到所述再现管理信息的所述现有条目中所包含的所述第二管理信息的预定条目中,

其中所述第一管理信息由存储至少表示所述流文件的再现时间信息和地址信息的关系的信息的文件组成,

所述第二管理信息由包括再现区域信息的至少一个条目的文件组成,所述再现区域信息指定具有对所述流文件设置的再现开始点和再现结束点的再现区域,并且所述再现区域信息能够存储表示所述流文件的再现时间信息的标记信息,以及

所述第二管理信息从所述再现区域信息参照所述第一管理信息,来表示用于所述流文件的所述再现方法。

25. 根据权利要求 24 所述的图像捕捉设备,其特征在于

当所述管理信息创建部分确定不根据所述操作部分的所述记录开始命令而将表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息附加写入到在所述再现管理信息的所述现有条目中所包含的所述第二管理信息的所述预定条目中时,所述管理信息创建部分根据所述操作部分的所述记录开始命令,新创建包括表示用于在所述记录介质上记录的所述流文件的再现方法的信息的所述第二管理信息。

26. 根据权利要求 24 所述的图像捕捉设备,其特征在于

所述管理信息创建部分将在记录所述流文件时更新的所述第二管理信息的所述第二管理信息的最近更新的条目当作所述第二管理信息的所述预定条目。

27. 根据权利要求 24 所述的图像捕捉设备,其特征在于

所述管理信息创建部分的所述附加记录能 / 不能确定基于在所述第二管理信息的所述预定条目中存储的所述再现区域信息的条目数目。

28. 根据权利要求 27 所述的图像捕捉设备,其特征在于

当所述管理信息创建部分已经确定在所述第二管理信息的所述预定条目中已存储的所述再现区域信息的条目数目小于预先设置的上限时,所述管理信息创建部分执行所述附加记录能 / 不能确定,使得表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息被附加写入到所述第二管理信息的所述预定条目中。

29. 根据权利要求 24 所述的图像捕捉设备,其特征在于

所述管理信息创建部分的所述附加记录能 / 不能确定是基于在所述第二管理信息的所述预定条目中存储的所述标记信息的条目数目。

30. 根据权利要求 29 所述的图像捕捉设备,其特征在于

当所述管理信息创建部分已经确定存储在所述第二管理信息的所述预定条目中的所述标记信息的条目数目小于预先设置的上限时,所述管理信息创建部分执行所述附加记录能 / 不能确定,使得表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息被附加写入到所述第二管理信息的所述预定条目中。

31. 根据权利要求 24 所述的图像捕捉设备,其特征在于

所述第一管理信息还存储表示在相应的流文件中存储的所述视频数据的属性的视频属性信息,以及

所述管理信息创建部分的所述附加记录能 / 不能确定是基于在所述第一管理信息中存储的所述视频属性信息。

32. 根据权利要求 31 所述的图像捕捉设备,其特征在于

所述管理信息创建部分的所述附加记录能 / 不能确定是基于根据当前记录模式的视频属性和在从所述第二管理信息的所述预定条目参照的所述第一管理信息的至少一个条目中存储的视频属性这两者。

33. 根据权利要求 32 所述的图像捕捉设备,其特征在于

当基于所述记录模式的所述视频属性的帧大小、纵横比、帧速率和扫描系统中的至少一个与在所述第一管理信息的所述条目中存储的所述视频属性的所述帧大小、纵横比、帧速率和扫描系统中的所述至少一个不匹配时,所述管理信息创建部分执行所述附加记录能 / 不能确定,使得根据所述记录命令输入部分的所述记录开始命令不将表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息附加写入到所述第二管理信息的所述预定条目。

34. 根据权利要求 24 所述的图像捕捉设备,其特征在于

所述管理信息创建部分的所述附加记录能 / 不能确定是基于所述第二管理信息的所述预定条目的文件大小。

35. 根据权利要求 34 所述的图像捕捉设备,其特征在于

当所述管理信息创建部分已经基于所述再现区域信息和在已记录所述单个流文件时所创建的所述标记信息的数据大小而确定所述第二管理信息的所述文件大小没有超过预先设置的上限时,所述管理信息创建部分执行所述附加记录能 / 不能确定,使得表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息被附加写入到所述第二管理信息的所述预定条目中。

36. 根据权利要求 24 所述的图像捕捉设备,其特征在于

所述管理信息创建部分的所述附加记录能 / 不能确定是基于从所述第二管理信息的所述预定条目参照的所述第一管理信息的所有条目的总文件大小。

37. 根据权利要求 36 所述的图像捕捉设备,其特征在于

当所述管理信息创建部分已经基于在所确保的最小连续时间内在记录所述单个流文件时所创建的所述第一管理信息的数据大小而确定从所述第二管理信息的所述预定条目参照的所述第一管理信息的所述总文件大小没有超过预先设置的上限时,所述管理信息创建部分执行所述附加记录能 / 不能确定,使得表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息被附加写入到所述第二管理信息的所述预定条目中。

38. 根据权利要求 24 所述的图像捕捉设备,其特征在于

所述第二管理信息从所述再现区域信息参照所述第一管理信息,以便控制所述流文件的再现,以及

所述管理信息创建部分的所述附加记录能 / 不能确定是基于表示在从所述第二管理信息的所述预定条目参照的所述第一管理信息的所有条目内包含的再现时间信息和地址

信息的关系的信息的条目数目。

39. 根据权利要求 38 所述的图像捕捉设备,其特征在于

当所述管理信息创建部分已经基于表示在确保的最小连续时间内在已记录所述单个流文件时所创建的所述第一管理信息内包含的所述再现时间信息和所述地址信息的所述信息的条目数目而确定从所述第二管理信息参照的所述第一管理信息的所有条目的总数没有超过所述第二管理信息的所述条目的预先设置的上限时,所述管理信息创建部分执行所述附加记录能 / 不能确定,以便表示用于记录在所述记录介质上的所述流文件的所述再现方法的所述信息附加写入到所述第二管理信息的所述预定条目中。

40. 根据权利要求 24 所述的图像捕捉设备,其特征在于

所述第二管理信息能够存储唯一的设备信息,以及

所述管理信息创建部分的所述附加记录能 / 不能确定是基于在所述第二管理信息的所述预定条目中存储的所述唯一的设备信息。

41. 根据权利要求 40 所述的图像捕捉设备,其特征在于

如果所述管理信息创建部分已经在所述第二管理信息的所述预定条目中存储了所述唯一的设备信息时,确定已经由所述记录设备或者与其等同的另一设备创建了所述唯一的设备信息,

则所述管理信息创建部分执行所述附加记录能 / 不能确定,使得表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息被附加写入到所述第二管理信息的所述预定条目中。

42. 根据权利要求 24 所述的图像捕捉设备,其特征在于

所述第二管理信息还能够存储表示所述第二管理信息的最近更新设备的信息,以及

所述管理信息创建部分的所述附加记录能 / 不能确定是基于在所述第二管理信息的所述预定条目中存储的所述最近更新信息。

43. 根据权利要求 42 所述的图像捕捉设备,其特征在于

如果已经在所述第二管理信息的所述预定条目中存储了所述最近更新设备信息时,确定所述最近更新设备信息表示所述记录设备,则执行所述附加记录能 / 不能确定,使得表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息被附加写入到所述第二管理信息的所述预定条目中。

44. 根据权利要求 24 所述的图像捕捉设备,其特征在于

所述管理信息创建部分对一个或多个项目执行所述附加记录能 / 不能确定,以及

当所述附加记录能 / 不能确定的至少一个所述项目表示附加写入不能状态时,所述管理信息创建部分执行所述附加记录能 / 不能确定,使得与在所述记录介质上记录的所述流文件相对应的所述第二管理信息不被附加写入到所述第二管理信息的所述预定条目中。

45. 根据权利要求 44 所述的图像捕捉设备,其特征在于

所述一个或多个项目包括以下中的至少一个项目:在所述第二管理信息的所述预定条目中存储的再现区域信息的条目数目、在所述第二管理信息的所述预定条目中存储的标记信息的条目数目、在所述第一管理信息中存储的视频属性信息、所述第二管理信息的所述预定条目的文件大小、从所述第二管理信息的所述预定条目参照的所述第一管理信息的所有条目的总文件大小、表示在从所述第二管理信息的所述预定条目参照的所述第一管理信

息的所有条目内包含的再现时间信息和地址信息的关系的信息的条目数目、在所述第二管理信息的所述预定条目中存储的唯一的设备信息、以及在所述第二管理信息的所述预定条目中存储的最近更新设备信息。

46. 一种多路复用由图像捕捉部分从对象捕捉和获取的视频数据以及由声音收集部分收集和获取的音频数据并且将多路复用的视频数据和音频数据记录到记录介质中的图像捕捉方法,其特征在于

所述图像捕捉方法包括:

记录步骤,多路复用由图像捕捉部分从所述对象捕捉和获取的视频数据以及由声音收集部分收集和获取的音频数据,并将经多路复用的流作为流文件而记录到所述记录介质中;

接受步骤,接受针对操作部分的、用于对所述记录介质的所述视频数据和所述音频数据的记录开始命令和记录停止命令的用户操作;以及

管理信息创建步骤,生成用于控制在所述记录介质上记录的所述流文件的再现的再现管理信息,所述再现管理信息由表示所述流文件的属性信息的第一管理信息和包含表示用于所述流文件的再现方法的信息的第二管理信息组成,

通过以下来执行所述管理信息创建步骤:基于所述再现管理信息的现有条目,确定是否根据与对操作部分的用户操作相对应的所述记录开始命令而将表示用于在所述记录介质上记录的所述流文件的再现方法的所述信息附加写入到所述再现管理信息的所述现有条目中所包含的所述第二管理信息的预定条目中,

其中所述第一管理信息由存储至少表示所述流文件的再现时间信息和地址信息的关系的信息的文件组成,

所述第二管理信息由包括再现区域信息的至少一个条目的文件组成,所述再现区域信息指定具有对所述流文件设置的再现开始点和再现结束点的再现区域,并且所述再现区域信息能够存储表示所述流文件的再现时间信息的标记信息,以及

所述第二管理信息从所述再现区域信息参照所述第一管理信息,来表示用于所述流文件的所述再现方法。

记录装置、记录方法、记录程序、摄像装置、摄像方法和摄像程序

技术领域

[0001] 本发明涉及适合于将已经多路复用了视频数据和音频数据的流数据记录到记录介质中的记录设备、记录方法、记录程序、图像捕捉设备、图像捕捉方法和图像捕捉程序。

背景技术

[0002] 作为可记录的可从记录和再现设备分离、具有相对大的记录容量且适合于记录由视频数据和音频数据组成的 AV(音频/视频)数据的记录介质,迄今为止,具有 4.7GB(吉字节)或更多记录容量的 DVD(数字多用盘)已得到广泛使用。专利文件“日本专利申请特开 No. 2004-350251”描述了一种将 DVD-Video 格式的数据记录到可记录型 DVD 中的图像捕捉设备。

[0003] 通常以从记录开始操作到记录停止操作为止所创建的视频数据为单位,将视频数据和音频数据记录到记录介质上。通常用再现列表信息来指定从记录开始操作到记录停止操作为止所创建的视频数据的再现区域和再现顺序。当通过再现列表信息、以从记录开始操作到记录停止操作为止所创建的视频数据为单位来管理记录 AV 流时,通过随意设置 AV 流的再现区域和再现顺序,可容易地对 AV 流进行编辑,而不需要对记录介质上的 AV 流进行处理。

[0004] 假定将视频数据和音频数据作为文件而记录在记录介质上。例如,将视频数据和音频数据记录为以从记录开始操作到记录停止操作为止所创建的数据为单位的文件。因为将视频数据和音频数据记录为文件,所以它们对诸如计算机设备的其它设备来说更具有亲和力。因此,可预期将更有效地利用所记录的数据。

[0005] 因此,当根据记录开始操作和记录停止操作的序列以文件的形式连续地记录到一个记录介质上的视频数据和音频数据被按照记录它们的顺序来再现时,必须执行特定技术。例如,可以预期对有关根据记录开始操作和记录停止操作的序列连续地记录到一个记录介质上的视频数据和音频数据的多个单元的信息进行控制,以便将其附加写入到存储再现列表信息的一个再现列表文件中。当执行这样的记录控制时,连续地再现已经记录的视频数据和音频数据的多个单元是很容易的。

[0006] 当对有关连续地记录的视频数据和音频数据的信息进行控制以便每当执行记录开始操作和记录停止操作时将其附加写入到再现列表文件中时,每当执行记录开始操作和记录停止操作时,再现列表文件的数据大小增大。另一方面,预期在记录设备和再现设备中,将再现列表文件暂时读入存储器,而以文件的形式记录的视频数据和音频数据则根据再现列表文件再现。另外,将用来记录视频数据和音频数据到记录介质中的另一管理信息读入到存储器中。因此,由于再现列表文件的大小的变大,它会消耗存储空间。因此,作为传统设备的问题,可能对其处理产生不利影响。

[0007] 当以文件的形式记录视频数据和音频数据时,需要提供一种机制,能够在视频数据中搜索要再现的帧并且能够容易地对以一帧为单位的视频数据进行编辑。为此,可以预

期,例如视频数据的再现时间信息和视频数据的文件地址是相关联的。当记录视频数据时,表示再现时间信息和文件地址之间的关系的数据增加。因此,在这种情况下,当记录或者再现视频数据和音频数据时,表示这样的关系的数据变大并且消耗存储空间。因此,作为传统设备的问题,可能对其它处理产生不利影响。

[0008] 另外,作为传统设备的问题,当在存储空间不够的状态下尝试记录视频数据时,系统可能强制停止或者暂停记录视频数据和音频数据。

[0009] 此外,近年来,通常使用多种视频格式。关于扫描线和扫描方法,这些格式包括作为隔行扫描格式的具有 480 扫描线的 480i、具有 576 扫描线的 576i、具有 1080 扫描线的 1080i 等等,以及作为逐行扫描格式的具有 480 扫描线的 480p、具有 576 扫描线的 576p、具有 720 扫描线的 720p、具有 1080 扫描线的 1080p 等等。

[0010] 另外,记录设备需要按照这样的各种类型的格式来记录视频数据。此外,需要用多种格式将视频数据记录到记录介质上。然而,如上所述,当对以文件的形式连续地记录的視頻数据和音频数据的信息进行控制以便将其附加写入到一个再现列表文件中时,很有可能在从记录介质再现文件的再现设备端发生故障。作为传统设备的问题,当以文件的形式记录的視頻数据和音频数据根据再现列表文件连续地再现时,如果下一文件的视频数据的格式与之前文件的不同,则解码器不能按适当的定时执行其处理,或者可能由于解码器的暂停而停止设备。

发明内容

[0011] 因此,本发明的目的是提供在记录从记录开始操作之后到记录停止操作之间所创建的視頻数据和音频数据时,使得用户友好性得以改善的记录设备、记录方法、记录程序、图像捕捉设备、图像捕捉方法和图像捕捉程序。

[0012] 为了解决上述问题,本发明的第一方面是多路复用视频数据和音频数据以及将多路复用数据记录到记录介质中的记录设备,其特征在于记录设备包括输入视频数据和音频数据的数据输入部分、输入用于视频数据和音频数据的记录开始命令和记录停止命令的记录命令输入部分、多路复用视频数据和音频数据以及以流文件形式将多路复用流记录到记录介质中的记录部分、以及创建用于控制记录在记录介质上的流文件的再现的再现管理信息的管理信息创建部分,再现管理信息由表示流文件的属性信息的第一管理信息和包含表示流文件的再现方法的信息的第二管理信息组成,而管理信息创建部分基于再现管理信息的现有条目,而确定是否根据记录命令输入部分的记录开始命令而将表示记录在记录介质上的流文件的再现方法的信息附加写入到包含在再现管理信息的现有条目内的第二管理信息的预定条目中。

[0013] 另外,本发明的第二方面是多路复用视频数据和音频数据以及将多路复用数据记录到记录介质中的记录方法,其特征在于记录方法包括输入用于已经输入到数据输入部分的视频数据和音频数据的记录开始命令和记录停止命令的记录命令输入步骤、多路复用视频数据和音频数据以及以流文件形式将多路复用流记录到记录介质中的记录步骤、以及生成用于控制记录在记录介质上的流文件的再现的再现管理信息的管理信息创建步骤,再现管理信息由表示流文件的属性信息的第一管理信息和包含表示流文件的再现方法的信息的第二管理信息组成,通过基于再现管理信息的现有条目而确定是否根据记录命令输入部

分的记录开始命令将表示记录在记录介质上的流文件的再现方法的信息附加写入到包含在再现管理信息的现有条目内的第二管理信息的预定条目中,来执行管理信息创建步骤。

[0014] 本发明的第三方面是使计算机设备执行多路复用视频数据和音频数据以及将多路复用数据记录到记录介质中的记录方法的记录程序,其特征在于记录方法包括输入用于已经输入到数据输入部分的视频数据和音频数据的记录开始命令和记录停止命令的记录命令输入步骤、多路复用视频数据和音频数据以及以流文件形式将多路复用流记录到记录介质中的记录步骤、以及生成用于控制记录在记录介质上的流文件的再现的再现管理信息的管理信息创建步骤,再现管理信息由表示流文件的属性信息的第一管理信息和包含表示流文件的再现方法的信息的第二管理信息组成,通过基于再现管理信息的现有条目而确定是否根据记录命令输入部分的记录开始命令将表示记录在记录介质上的流文件的再现方法的信息附加写入到包含在再现管理信息的现有条目内的第二管理信息的预定条目中,来执行管理信息创建步骤。

[0015] 本发明的第四方面是多路复用由图像捕捉部分从对象中捕捉和获取的视频数据以及由声音收集部分收集和获取的音频数据并且将多路复用视频数据和音频数据记录到记录介质中的图像捕捉设备,其特征在于图像捕捉设备包括捕捉对象的图像以及输出视频数据的图像捕捉部分、收集声音以及输出音频数据的的声音收集部分、多路复用视频数据和音频数据以及以流文件形式将多路复用流记录到记录介质中的声音收集部分、接受对用于将视频数据和音频数据记录到记录介质中的记录开始命令和记录停止命令的用户操作的操作部分、以及创建用于控制记录在记录介质上的流文件的再现的再现管理信息的管理信息创建部分,再现管理信息由表示流文件的属性信息的第一管理信息和包含表示流文件的再现方法的信息的第二管理信息组成,管理信息创建部分基于再现管理信息的现有条目而确定是否根据与操作部分的用户操作相对应的记录开始命令将表示记录在记录介质上的流文件的再现方法附加写入到包含在再现管理信息的现有条目内的第二管理信息的预定条目中。

[0016] 本发明的第五方面是多路复用由图像捕捉部分从对象中捕捉和获取的视频数据以及由声音收集部分收集和获取的音频数据并且将多路复用视频数据和音频数据记录到记录介质中的图像捕捉方法,其特征在于所述图像捕捉方法包括多路复用由图像捕捉部分从对象捕捉和获取的视频数据和音频数据以及由声音收集部分收集和获取的音频数据并且以流文件形式将多路复用流记录到记录介质中的记录步骤、相对于操作部分,接受将视频数据和音频数据记录到记录介质中的记录开始命令和记录停止命令的用户操作的接受步骤、以及生成用于控制记录在记录介质上的流文件的再现的再现管理信息的管理信息创建步骤,再现管理信息由表示流文件的属性信息的第一管理信息和包含表示流文件的再现方法的信息的第二管理信息组成,通过基于再现管理信息的现有条目而确定是否根据与操作部分的用户操作相对应的记录开始命令将表示记录在记录介质上的流文件的再现方法的信息附加写入到包含在再现管理信息的现有条目内的第二管理信息的预定条目中,来执行管理信息创建步骤。

[0017] 本发明的第六方面是使计算机设备执行多路复用由图像捕捉部分从对象中捕捉和获取的视频数据以及由声音收集部分收集和获取的音频数据并且将多路复用视频数据和音频数据记录到记录介质中的图像捕捉方法的图像捕捉程序,其特征在于图像捕捉方法

包括多路复用由图像捕捉部分从对象捕捉和获取的视频数据和音频数据以及由声音收集部分收集和获取的音频数据并且以流文件形式将多路复用流记录到记录介质中的记录步骤、相对于操作部分接受将视频数据和音频数据记录到记录介质中的记录开始命令和记录停止命令的用户操作的接受步骤、以及创建用于控制记录在记录介质上的流文件的再现的再现管理信息的管理信息创建步骤,再现管理信息由表示流文件的属性信息的第一管理信息和包含表示流文件的再现方法的信息的第二管理信息组成,通过基于再现管理信息的现有条目而确定是否根据与操作部分的用户操作相对应的记录开始命令将表示记录在记录介质上的流文件的再现方法的信息附加写入到包含在再现管理信息的现有条目内的第二管理信息的预定条目中,来执行管理信息创建步骤。

[0018] 如上所述,在本发明的第一、第二、和第三方面中,创建用于控制记录在记录介质上的流文件的再现的再现管理信息。再现管理信息由表示由已经多路复用视频数据和音频数据的流组成的流文件的属性信息的第一管理信息和包含表示流文件的再现方法的信息的第二管理信息组成。附加记录能/不能确定根据对视频数据的记录开始命令,基于再现管理信息的现有条目而确定表示记录在记录介质上的流文件的再现方法的信息是否附加写入到包含在再现管理信息的现有条目内的第二管理信息的预定条目中。因此,即使重复地执行记录开始操作和记录停止操作,也能控制以恰当地创建第二管理信息,而不需要用户识别第二管理信息。

[0019] 如上所述,在本发明的第四、第五、和第六方面中,创建用于控制记录在记录介质上的流文件的再现的再现管理信息。再现管理信息由表示由已经多路复用捕捉和获取的视频数据以及收集和获取的音频数据的流组成的流文件的属性信息的第一管理信息和包含表示用于流文件的再现方法的信息的第二管理信息组成。附加记录能/不能确定基于再现管理信息的现有条目而确定是否根据相对于操作部分的与用户操作相对应的记录开始命令而将表示由从对象捕捉和获取的视频数据组成的并且记录在记录介质上的流文件的再现方法的信息附加写入到包含在再现管理信息的现有条目内的第二管理信息的预定条目中。因此,即使重复地执行用于捕捉的记录开始操作和记录停止操作,也能控制以恰当地创建第二管理信息,而不需要用户识别第二管理信息。

[0020] 如上所述,在本发明的第一、第二、和第三方面中,创建用于控制记录在记录介质上的流文件的再现的再现管理信息。再现管理信息由表示由已经多路复用视频数据和音频数据的流组成的流文件的属性信息的第一管理信息和包含表示用于流文件的再现方法的信息的第二管理信息组成。附加记录能/不能确定基于再现管理信息的现有条目而确定是否根据对记录视频数据的记录开始命令将表示记录在记录介质上的流文件的再现方法的信息附加写入到包含在再现管理信息的现有条目内的第二管理信息的预定条目中。因此,提供了如下效果:即使重复地执行记录开始操作和记录停止操作,也能控制以恰当地创建第二管理信息,而不需要用户识别第二管理信息。

[0021] 如上所述,在本发明的第四、第五、和第六方面中,创建用于控制记录在记录介质上的流文件的再现的再现管理信息。再现管理信息由表示由已经多路复用捕捉和获取的视频数据以及收集和获取的音频数据的流组成的流文件的属性信息的第一管理信息和包含表示用于流文件的再现方法的信息的第二管理信息组成。附加记录能/不能确定基于再现管理信息的现有条目而确定是否根据相对于操作部分的与用户操作相对应的记录开始命

令将表示由从对象中捕捉和获取的视频数据组成的并且记录在记录介质上的流文件的再现方法的信息附加写入到在再现管理信息的现有条目内包含的第二管理信息的预定条目中。因此,提供了以下效果:即使重复地执行用于捕捉的记录开始操作和记录停止操作,也能控制以恰当地创建第二管理信息,而不需要用户识别第二管理信息。

附图说明

- [0022] 图 1 是示出在可应用于本发明的 AVCHD 格式中规定的模型的数据模型的概略的示意图;
- [0023] 图 2 是描述索引表的示意图;
- [0024] 图 3 是示出剪辑 AV 流、剪辑信息、剪辑、播放项目和播放列表的关系的 UML 示意图;
- [0025] 图 4 是描述参照多个播放列表中的相同剪辑的方法的示意图;
- [0026] 图 5 是描述记录在记录介质上的文件的管理结构的示意图;
- [0027] 图 6 是示出表示文件“index.bdmv”的结构示例的语法的示意图;
- [0028] 图 7 是示出表示块 blkIndexes() 的结构示例的语法的示意图;
- [0029] 图 8 是示出表示文件“MovieObject.bdmv”的结构示例的语法的示意图;
- [0030] 图 9 是示出表示块 blkMovieObjects() 的结构示例的语法的示意图;
- [0031] 图 10 是示出表示播放列表文件“xxxxx.mpls”的结构示例的语法的示意图;
- [0032] 图 11 是示出表示块 blkPlayList() 的结构示例的语法的示意图;
- [0033] 图 12 是示出表示块 blkPlayItem() 的结构示例的语法的示意图;
- [0034] 图 13A 和图 13B 是分别描述第一和第二无缝连接的示意图;
- [0035] 图 14 是示出表示块 blkPlayListMark() 的结构示例的语法的示意图;
- [0036] 图 15 是示出表示剪辑信息文件的结构示例的语法的示意图;
- [0037] 图 16 是示出表示块 blkClipInfo() 的结构示例的语法的示意图;
- [0038] 图 17 是示出表示块 blkProgramInfo() 的结构示例的语法的示意图;
- [0039] 图 18 是示出表示块 blkStreamCodingInfo(stream_index) 的结构示例的语法的示意图;
- [0040] 图 19 是示出以字段 VideoFormat 表示的视频数据的格式列表的示例的示意图;
- [0041] 图 20 是示出以字段 FrameRate 表示的帧速率列表的示例的示意图;
- [0042] 图 21 是示出以字段 AspectRatio 表示的帧速率列表的示例的示意图;
- [0043] 图 22 是示出表示块 blkCPI() 的结构示例的语法的示意图;
- [0044] 图 23 是示出表示块 blkEPMap() 的结构示例的语法的示意图;
- [0045] 图 24 是示出表示块 blkEPMapForOneStreamPID(EP_stream_type, Nc, Nf) 的结构示例的语法的示意图;
- [0046] 图 25 是示出条目 PTSEPCoarse 和条目 PTSEPFine 的格式的示例的示意图;
- [0047] 图 26 是示出条目 SPNEPCoarse 和条目 SPNEPFine 的格式的示例的示意图;
- [0048] 图 27 是示出表示块 blkExtensionData() 的结构示例的语法的示意图;
- [0049] 图 28 是示出块 blkExtensionData() 的数据的每个条目的参照关系的示意图;
- [0050] 图 29 是示出将数据写入块 blkExtensionData() 的处理的示例的流程图;
- [0051] 图 30 是示出从块 blkExtensionData() 中读取扩展数据的处理的示例的流程图;
- [0052] 图 31 是示出表示文件“index.bdmv”中的字段 blkExtensionData() 的块

DataBlock() 的结构示例的语法的示意图；

[0053] 图 32 是示出表示块 blkTableOfPlayList() 的结构示例的语法的示意图；

[0054] 图 33 是示出表示播放列表文件“xxxxx.mpls”中的块 blkExtensionData() 的块 DataBlock() 的结构示例的语法的示意图；

[0055] 图 34 是示出表示块 blkPlayListMeta() 的结构示例的语法的示意图；

[0056] 图 35 是示出表示播放列表文件中的块 blkMakersPrivateData() 的结构示例的语法的示意图；

[0057] 图 36 是示出表示剪辑信息文件中的块 blkExtensionData() 的块 DataBlock() 的结构示例的语法的示意图；

[0058] 图 37 是示出表示块 blkProgramInfoExt() 的结构示例的语法的示意图；

[0059] 图 38 是示出表示块 blkStreamCodingInfoExt() 的结构示例的语法的示意图；

[0060] 图 39A 和图 39B 是示出虚拟播放器的操作的概略的流程图；

[0061] 图 40 是示出虚拟播放器的操作的概略的示意图；

[0062] 图 41 是示出根据本发明的实施例的记录设备的结构示例的方框图；

[0063] 图 42 是示出根据本发明的实施例的数据结构的示例的示意图；

[0064] 图 43 是示出确定是否能够将章(chapter)附加写入到播放列表文件的处理示例的流程图；

[0065] 图 44 是示出基于播放项目数的附加写入能 / 不能确定的示例的流程图；

[0066] 图 45 是示出基于播放列表标记数的附加写入能 / 不能确定的处理示例的流程图；

[0067] 图 46 是示出基于视频属性的附加写入能 / 不能确定的处理示例的流程图；

[0068] 图 47 是示出基于文件大小的附加写入能 / 不能确定的处理示例的流程图；

[0069] 图 48 是示出基于参照作为其他可写候选者(candidate)的剪辑信息文件的总文件大小的附加写入能 / 不能确定的处理示例的流程图；

[0070] 图 49 是示出基于存储在参照作为其他可写候选者的剪辑信息文件中的入口点总数的附加写入能 / 不能确定的处理示例的流程图；

[0071] 图 50 是示出基于有无另一设备的唯一扩展数据的附加写入能 / 不能确定的处理示例的流程图；

[0072] 图 51 是示出基于作为另外的可写候选者的播放列表文件的最近更新设备的附加写入能 / 不能确定的处理示例的流程图；

[0073] 图 52 是示出根据本发明的实施例的剪辑记录方法的示例的流程图；

[0074] 图 53 是示出根据本发明另一实施例的摄像机设备的结构示例的方框图。

具体实施方式

[0075] 接下来,将参考附图描述本发明的实施例。为了易于理解,将描述可应用于本发明的格式示例(以下,称为 AVCHD 格式)。AVCHD 格式是目前被提议为以预定方式在可记录的记录介质上记录多路复用了视频数据和音频数据的 AV(音频/视频)流的记录格式。AVCHD 格式允许通过播放列表以一个剪辑为单位来管理在记录介质上记录的 AV 流。

[0076] 例如,根据 ITU-U(国际电信联盟-电信标准部)推荐 H.264 或者 ISO(国际标准

组织)/IEC(国际电子技术协会)国际标准 14496-10(MPEG-4 部分 10)高级视频编码(以下,称为 H.264|AVC)来编码的且已根据 MPEG2 系统多路复用的位流被称为剪辑 AV 流(或者 AV 流)。通过预定文件系统,将剪辑 AV 流作为文件而记录到盘上。该文件称为剪辑 AV 流文件(或者 AV 流文件)。

[0077] 剪辑 AV 流文件是文件系统上的管理单元。因此,剪辑 AV 流文件并不总是用户友好的管理单元。为了促进用户友好性,需要提供连续地再现由视频内容分成的多个剪辑 AV 流文件的机制和部分地再现剪辑 AV 流文件的机制,并将顺利执行特殊再现操作、提示再现操作等等所必需的信息以数据库的形式记录到盘中。

[0078] 图 1 示出在可应用于本发明的 AVCHD 格式中规定的模型数据的概略。如图 1 所示,AVCHD 格式具有四层数据结构。最底层是放置剪辑 AV 流的层(为方便起见,以下称为剪辑层)。放在剪辑层上的是用来指定剪辑 AV 流的再现位置的播放列表(PlayList)和播放项目(PlayItem)的层(为方便起见,称为播放列表层)。放在播放列表层上的是电影对象(Movie Object)的层(为方便起见,称为对象层),其中每个电影对象由指定播放列表的再现顺序等的命令所组成的。最顶层是用于管理在记录介质上存储的标题等的索引表的层(为方便起见,称为索引层)。

[0079] 接下来,将描述剪辑层。例如,剪辑 AV 流是根据 MPEG2 TS(传输流)格式多路复用了视频数据和音频数据的位流。有关剪辑 AV 流的信息作为剪辑信息(Clip Information)记录在文件中。

[0080] 另外,作为字幕的图形流的 OB 流(覆层位流)和作为菜单指示的数据流(诸如按钮图像数据)的 MB 流(菜单位流)可与剪辑 AV 流进行多路复用。

[0081] 将其中记录了与剪辑 AV 流文件相对应的剪辑信息的一对剪辑 AV 流文件和剪辑信息文件视为对象,并且称之为剪辑(Clip)。换句话说,剪辑是由剪辑 AV 流和剪辑信息所组成的一个对象。

[0082] 通常将文件视为字节序列。基于时间而映射剪辑 AV 流文件的内容。主要基于时间而指定剪辑的入口点(entry point)。当向预定剪辑给定存取点的时间戳时,可使用剪辑信息文件来查找用来从剪辑 AV 流文件中读取数据的地址信息。

[0083] 接下来,将描述播放列表层。播放列表指定将要再现的 AV 流文件。播放列表由用来指定所指定的 AV 流文件的再现位置的再现开始点(IN 点)和再现结束点(OUT 点)的对组成。一对再现开始点和再现结束点称为播放项目(PlayItem)。播放列表由一组播放项目组成。当再现播放项目时,再现参照(refer)播放项目的 AV 流文件的一部分。换句话说,基于播放项目的 IN 点和 OUT 点的信息而再现剪辑区域。

[0084] 接下来,将描述对象层。电影对象包括与电影对象相关联的导航命令程序和终点信息。导航(navigation)程序是用于控制播放列表的再现的命令(导航命令)。

[0085] 接下来,将描述索引层。索引层由索引表(Index Table)组成。索引表是最顶层上的定义在盘上记录的内容的标题的表。播放器常驻系统软件的模块管理器基于存储在索引表中的标题信息而控制记录介质的数据的再现。

[0086] 换句话说,如图 2 所示,索引表中的任何条目都称为标题。作为索引表中的条目的所有第一重放标题(First PlaybackTitle)、菜单标题(MenuTitle)、和电影标题(MovieTitle)#1、#2 等都是标题。每个标题都表示对电影对象的链接。

[0087] 为了易于理解,例如,在只再现记录介质中,当存储在该记录介质中的内容是电影时,第一重放标题与电影制造者的广告画面相对应,其后面是电影的主体。当内容是电影时,菜单标题与用来选择操作的菜单屏幕相对应,所述操作用于再现主体、搜索章、设置字幕和语言、以及再现奖金(bonus)画面。电影标题是从菜单标题中选取的画面。标题可以构造为菜单屏幕。

[0088] 图3是示出上述剪辑AV流、剪辑信息(流属性)、剪辑、播放项目和播放列表的关系的UML(统一建模语言)示图。播放列表与一个或者多个播放项目相关联。播放项目与一个剪辑相关联。一个剪辑可以与开始点和/或结束点相互不同的多个播放项目相关联。一个剪辑指示一个剪辑AV流文件。一个剪辑同样也指示一个剪辑信息文件。剪辑AV流文件和剪辑信息文件具有一对一的关系。用这样的结构,能够非破坏性地指定再现顺序,以便仅仅再现剪辑AV流文件所期望的部分,而不需要改变它。

[0089] 另外,如图4所示,相同的剪辑可以从多个播放列表参照。另外,多个剪辑可以由一个播放列表指定。用播放列表中的播放项目的IN点和OUT点参照剪辑。在图4中示出的示例中,从播放列表310的播放项目320参照剪辑300。另外,对于由具有播放项目321和播放项目322的播放列表311的播放项目321的IN点和OUT点所指定的区域来参照剪辑300。另外,对于由播放列表312的播放项目322的IN点和OUT点指定的区域来参照剪辑301。另外,对于由具有播放项目323和播放项目324的播放列表312的播放项目323的IN点和OUT点指定的区域来参照剪辑301。在图4中示出的示例中,还从另一播放列表参照剪辑301。

[0090] 接下来,将参考图5描述记录介质上的AVCHD格式的文件的管理结构。文件用目录结构进行分级管理。在记录介质上创建一个目录(在图5中,根目录)。在该目录下是由一个记录和再现系统管理的范围。

[0091] 放在根目录下的是目录“BDMV”。当必要时,放在根方向下的是目录“AVCHDTN”。放在目录“AVCHDTN”中的是表示已经缩小到预定大小的剪辑的代表图像的缩略图(thumbnail)文件。放在目录“BDMV”中的是参考图1所描述的数据结构。

[0092] 直接放在目录“BDMV”下的只有两个文件,文件“index.bdmv”和文件“MovieObject.bdmv”。放在目录“BDMV”下的是目录“PLAYLIST”、目录“CLIPINF”、目录“STREAM”和目录“BACKUP”。目录“BACKUP”存储各个目录和文件的备份。

[0093] 文件“index.bdmv”描述目录“BDMV”的内容。换句话说,该文件“index.bdmv”与索引层中的索引表相对应,该索引层是上述的最高层。文件“MovieObject.bdmv”中存储有关至少一个电影对象的信息。换句话说,该文件“MovieObject.bdmv”与上述对象层相对应。

[0094] 目录“PLAYLIST”是放置播放列表的数据库的目录。换句话说,目录“PLAYLIST”包括文件“xxxxx.mpls”,其是关于播放列表的文件。文件“xxxxx.mpls”是为每个播放列表所创建的文件。在该文件名中,“.”(句点)之前的“xxxxx”是五位数字,句点之后的“mpls”是该文件类型的固定扩展名。

[0095] 目录“CLIPINF”是放置剪辑的数据库的目录。换句话说,目录“CLIPINF”包括文件“zzzzz.clpi”,其是与剪辑AV流文件相对应的剪辑信息文件。在该文件名中,“.”(句点)之前的“zzzzz”是五位数字,句点之后的“clpi”是该文件类型的固定扩展名。

[0096] 目录“STREAM”是放置作为实体的AV流文件的目录。换句话说,目录“STREAM”包

括与剪辑信息文件相对应的剪辑 AV 流文件。剪辑 AV 流文件由 MPEG2 (运动画面专家组 2) 传输流 (以下简写成 MPEG2 TS) 组成。剪辑 AV 流文件的文件名是“zzzzz.m2ts”。当剪辑 AV 流文件的文件名的句点之前的“zzzzz”与相对应的剪辑信息文件的“zzzzz”相同时,可以容易地识别它们的关系。

[0097] 在目录“AVCHDTN”中可放置两种缩略图文件 thumbnail.tidx 和 thumbnail.tdt2。缩略图文件 thumbnail.tidx 存储在预定系统中加密的缩略图图像。相反,缩略图文件 thumbnail.tdt2 存储没有加密的缩略图图像。因为与用户用诸如摄像机所捕捉的剪辑相对应的缩略图图像有可能是自由复制的,所以将该缩略图图像存储在缩略图文件 thumbnail.tdt2 中。

[0098] 接下来,在图 5 中示出的文件当中,将更详细地描述那些与本发明密切相关的内容。首先,将描述直接放在目录“BDMV”下的文件“index.bdmv”。图 6 示出表示文件“index.bdmv”的结构示例的语法。在该示例中,根据作为计算机装置等的编程语言使用的 C 语言的语法来表示语法。这也应用于示出其它文件的语法的附图。

[0099] 在图 6 中,字段 TypeIndicator 具有 32 位的数据长度,指示该文件是索引表。字段 TypeIndicator2 具有 32 位的数据长度,表示文件“index.bdmv”的版本。字段“IndexesStartAddress”具有 32 位的数据长度,表示在该语法中的块 blkIndexes() 的开始地址。

[0100] 字段 ExtensionDataStartAddress 具有 32 位数据长度,表示该语法中的块 blkExtensionData() 的开始地址。块 blkExtensionData() 是可存储预定扩展数据的块。字段 ExtensionDataStartAddress 表示块 blkExtensionData() 的开始地址,作为从该文件“index.bdmv”的字节开始的相对字节数。相对字节数从“0”开始。如果字段 ExtensionDataStartAddress 的值是“0”,则指示该文件“index.bdmv”不包括块 blkExtensionData()。

[0101] 字段 ExtensionDataStartAddress 后面是具有 192 字节的数据长度的区域 reserved (保留)。区域 reserved 是用于字节对齐 (alignment)、未来字段添加等的备用区域。这也适用于以下的描述。块 blkAppInfoBDMV() 是其中内容创建者可描述任何信息而不影响播放器的操作等的块。

[0102] 块 blkIndexes() 是该文件“index.bdmv”的实质内容。在该文件“index.bdmv”中描述的内容指定当盘放入播放器时再现的第一重放、和从顶层菜单中调用的标题 (电影对象)。根据由索引表调用的电影对象等等中所描述的命令而读取播放列表文件 (稍后将描述)。

[0103] 图 7 示出表示块 blkIndexes() 的结构示例的语法。字段 Length 具有 32 位的数据长度,表示紧跟该字段 Length 之后直至该块 blkIndexes() 的结尾的数据长度。字段 Length 后面是块 FirstPlaybackTitle() 和块 MenuTitle()。

[0104] 块 FirstPlaybackTitle() 描述有关在第一重放中使用的对象的信息。块 FirstPlaybackTitle() 描述具有 1 位数据长度的区域 reserved,后面是固定值“1”。块 FirstPlaybackTitle() 还通过具有 31 位数据长度的区域 reserved 描述固定值“1”。紧跟具有 14 位数据长度的区域 reserved 之后,放置具有 16 位数据长度的字段 FirstPlaybackTitleMobjIDRef。该字段 FirstPlaybackTitleMobjIDRef 表示在第一重放

标题中使用的电影对象的 ID。

[0105] 电影对象的 ID 用值 `mobj_id` 表示,例如,在稍后将参考图 8 和图 9 描述的基于电影对象的语法的电影对象的 for 循环语句中,该值 `mobj_id` 作为循环变量使用。在该示例中,字段 `FirstPlaybackTitleMobjIDRef` 存储与将要参照的电影对象相对应的值 `mobj_id`。

[0106] 块 `blkIndexes()` 中的块 `FirstPlaybackTitle()` 的字段 `FirstPlaybackTitleMobjIDRef` 可以表示顶层菜单或者标题的电影对象。

[0107] 块 `MenuTitle()` 描述有关在顶层菜单中使用的对象的信息。块 `MenuTitle()` 描述具有 1 位数据长度的区域 `reserved`,后面是固定值“1”。紧跟具有 31 位数据长度的区域 `reserved` 之后,描述固定值“1”。另外,紧跟具有 14 位数据长度的区域 `reserved` 之后,放置具有 16 位数据长度的字段 `MenuTitleMobjIDRef`。该字段 `MenuTitleMobjIDRef` 表示在菜单标题中使用的电影对象的 ID。

[0108] 块 `MenuTitle()` 后面是具有 16 位数据长度的并且表示用户能够选择以及能够再现的标题数的字段 `NumberOfTitles`。根据下一 for 循环语句,由该字段 `NumberOfTitles` 表示次数,描述具有值 `title_id` 的自变量的块 `MovieTitle[title_id]()`。块 `MovieTitle[title_id]()` 描述每个标题的信息。值 `title_id` 是从“0”到由字段 `NumberOfTitles` 表示的值之间的数值,并且它标识标题。

[0109] 在块 `MovieTitle[title_id]()` 中,在具有 1 位数据长度的区域 `reserved` 之后,描述固定值“1”。另外,在具有 46 位数据长度的区域 `reserved` 之后,描述字段 `MovieTitleMobjIDRef`。字段 `MovieTitleMobjIDRef` 具有 16 位数据长度,表示在该标题中使用的电影对象的 ID。字段 `MovieTitleMobjIDRef` 后面是具有 32 位数据长度的区域 `reserved`。

[0110] 图 8 示出表示直接放在目录“BDMV”下的文件“`MovieObject.bdmv`”的结构示例的语法。字段 `TypeIndicator` 具有 32 位(4 字节)数据长度,并且指示该文件是文件“`MovieObject.bdmv`”。字段 `TypeIndicator` 描述根据 ISO(国际标准化组织)646 规定的编码系统编码的四个字母的字符串。在图 8 中示出的示例中,字段 `TypeIndicator` 描述根据 ISO 646 规定的系统编码的四个字母的字符串“MOBJ”,指示该文件是文件“`MovieObject.bdmv`”。

[0111] 字段 `TypeIndicator2` 具有 32 位(4 字节)数据长度,表示该文件“`MovieObject.bdmv`”的版本号。在该文件“`MovieObject.bdmv`”中,字段 `TypeIndicator2` 应该是根据 ISO 646 规定的编码系统编码的四个字母的字符串“0100”。

[0112] 字段 `ExtensionDataStartAddress` 具有 32 位数据长度,表示该语法中的块 `blkExtensionData()` 的开始地址。字段 `ExtensionDataStartAddress` 表示块 `blkExtensionData()` 的开始地址,如从该文件“`MovieObject.bdmv`”的字节开始的相对字节数。相对字节数从“0”开始。如果该字段 `ExtensionDataStartAddress` 的值是“0”,则指示在该文件“`MovieObject.bdmv`”中没有块 `blkExtensionData()`。

[0113] 图 8 中示出的语法中的字段 `padding_word` 具有 16 位数据长度。根据该文件“`MovieObject.bdmv`”的语法,将字段 `padding_word` 插入 for 循环语句中 N1 次或者 N2 次。值 N1 或者值 N2 是“0”或者任何正整数。作为字段 `padding_word`,可使用任何值。

[0114] 在字段 `ExtensionDataStartAddress` 之后,放置具有 224 位数据长度的区域

reserved。其后,存储块 blkMovieObjects(),其是该文件“MovieObject.bdmv”的主体。

[0115] 图9示出表示块 blkMovieObjects() 的结构示例的语法。字段 Length 具有 32 位数据长度,表示直接在字段 Length 之后直至该块 blkMovieObjects() 的结尾的数据长度。在具有 32 位数据长度的区域 reserved 之后,放置字段 NumberOfMobjs。字段 NumberOfMobjs 表示根据紧跟字段 NumberOfMobjs 之后的 for 循环语句存储的电影对象的数目(number)。电影对象由用作 for 循环语句的循环变量的值 mobj_id 唯一地标识。值 mobj_id 从“0”开始。电影对象按 for 循环语句中描述的顺序定义。

[0116] for 循环语句中的块 TerminalInfo() 描述固定值“1”,后面是具有 15 位数据长度的区域 reserved。其后,放置具有 16 位的数据长度的字段 NumberOfNavigationCommands[mobj_id]。该字段 NumberOfNavigationCommands[mobj_id] 表示包含在由值 mobj_id 表示的电影对象 MovieObject[mobj_id]() 内的导航命令(NavigationCommand) 的数目。

[0117] 根据具有循环变量值 command_id 的 for 循环语句,对于由字段 NumberOfNavigationCommands[mobj_id] 表示的数目来描述导航命令。换句话说,放在该 for 循环语句中的字段 NavigationCommand[mobj_id][command_id] 包含按照由值 mobj_id 表示的块 MovieObject[mobj_id]() 中包含的值 command_id 所表示的顺序的导航命令 NavigationCommand。值 command_id 从“0”开始。导航命令 NavigationCommand 按该 for 循环语句中描述的顺序定义。

[0118] 图10示出表示播放列表文件“xxxxx.mpls”的结构示例的语法。字段 TypeIndicator 具有 32 位(4 字节)数据长度,指示该文件是播放列表文件。字段 TypeIndicator2 具有 32 位(4 字节)数据长度,表示该播放列表文件的版本。字段 PlayListStartAddress 具有 32 位数据长度,表示在该语法中的块 blkPlayList() 的开始地址。

[0119] 字段 PlayListMarkStartAddress 具有 32 位数据长度,表示在该语法中的块 blkPlayListMark() 的开始地址。字段 ExtensionDataStartAddress 具有 32 位数据长度,表示在该语法中块 blkExtensionData() 的开始地址。字段 ExtensionDataStartAddress 表示块 blkExtensionData() 的开始地址,如从该文件“xxxxx.mpls”的字节开始的相对字节数。相对字节数从“0”开始。如果字段 ExtensionDataStartAddress 的值是“0”,则指示在该文件“xxxxx.mpls”中没有块 blkExtensionData()。

[0120] 在具有 160 位数据长度的区域 reserved 之后,放置块 blkAppInfoPlayList()。块 blkAppInfoPlayList() 描述诸如在下一块 blkPlayList() 中描述的播放列表的类型和再现约束的信息。块 blkPlayList() 描述播放列表。块 blkPlayListMark() 描述如在章跳转中将要跳转(jump)的点。块 blkExtensionData() 是可存储预定扩展数据的块。

[0121] 图11示出表示块 blkPlayList() 的结构示例的语法。字段 Length 具有 32 位数据长度,表示直接在字段 Length 之后直至块 blkPlayList() 的结尾的数据长度。字段 Length 后面是具有 16 位数据长度的区域 reserved,后面是字段 NumberOfPayItems。字段 NumberOfPlayItems 具有 16 位数据长度,表示包含在该块 blkPlayList() 内的播放项目数。字段 NumberOfSubPath 表示包含在该块 blkPlayList() 内的子路径数。

[0122] 根据下一 for 循环语句,对于由字段 NumberOfPlatItems 表示的数目来描述用于

描述播放项目的块 blkPlayItem()。基于 for 循环语句的计数值变成块 blkPlayItem() 的标识符 PlayItem_id。根据下一 for 循环语句,对于由字段 NumberOfSubPath 表示的数目来描述块 blkSubPath()。基于 for 循环语句的计数值变成块 blkSubPath() 的标识符 SubPath_id。

[0123] 可与子播放项目相对应地提供子路径,而与主再现的播放项目相对应地提供主路径。子路径可用于指定例如记录之后的音频数据,以及将要与播放项目中指定的剪辑同步地再现的子图像。

[0124] 图 12 示出表示块 blkPlayItem() 的结构示例的语法。字段 Length 具有 16 位数据长度,表示紧跟该字段 Length 之后直至该块 blkPlayItem() 的结尾的数据长度。

[0125] 字段 ClipInformation FileName 具有 40 位 (5 字节) 数据长度,表示该块 blkPlayItem() 参照的剪辑信息文件的文件名。在该播放项目中,读取具有由字段 ClipInformationFileName 表示的文件的剪辑信息文件。字段 ClipCodecIdentification 具有 32 位 (4 字节) 数据长度,表示在该块 blkPlayItem() 的该播放项目中使用的剪辑 AV 流的编解码器系统。

[0126] 紧跟具有 12 位数据长度的区域 reserved 之后,放置字段 ConnectionCondition。字段 ConnectionCondition 具有 4 位数据长度,表示有关剪辑的连接状态的信息。对于可记录的记录介质,作为字段 ConnectionCondition 的值,使用“1”、“5”或“6”。当字段 ConnectionCondition 的值是“1”时,指示从当前播放项目参照的剪辑和从紧接的前一播放项目参照的剪辑之间不是无缝连接的。当字段 ConnectionCondition 的值是“5”或者“6”时,指示从当前播放项目参照的剪辑和从紧接的前一播放项目参照的剪辑之间是无缝连接的。术语“无缝连接”指的是剪辑的再现被控制为使得按照帧定时 (timing) 成功地再现一个剪辑和下一剪辑。

[0127] 当字段 ConnectionCondition 的值是“5”时,当前播放项目参照的剪辑的音频数据的记录长度比它的视频数据的记录长度更长 (参考图 13A)。因此,当连接剪辑时,能够对音频数据执行渐隐处理。例如,当通过用户的记录停止操作关闭剪辑时,字段 ConnectionCondition 的值是“5”。以下,字段 ConnectionCondition 的值是“5”的剪辑连接方法称为第一无缝连接。

[0128] 当字段 ConnectionCondition 的值是“6”时,当前播放项目参照的剪辑的音频数据的记录长度与它的视频数据的记录长度一样 (参考图 13B)。因此,剪辑可进行无缝连接。例如,当由于系统而不是用户的记录停止操作的原因而关闭剪辑时,字段 ConnectionCondition 的值是“6”。以下,该字段 ConnectionCondition 值的剪辑连接方法称为第二无缝连接。

[0129] 字段 RefToSTCID 具有 8 位数据长度,表示有关系统时基 (STC) 的间断点的信息。每个字段 INTime 和字段 OUTTime 都具有 32 位数据长度,表示主剪辑 AV 流的再现范围。字段 INTime 表示开始点 (IN 点),而字段 OUTTime 表示结束点 (OUT 点)。

[0130] 块 blkUOMaskTable() 是对用户输入设置接受约束的表。具有 1 位数据长度的标志 PlayItemRandomAccessFlag 规定对该块 blkPlayItem() 的播放项目的随机存取许可 / 约束。其后,在具有 7 位数据长度的区域 reserved 之后,放置字段 StillMode。字段 StillMode 具有 8 位数据长度,指示块 blkPlayItem() 中播放项目的最后画面是否显示为静止图像。当

字段 StillMode 的值是“0x01”(二进制)时,具有 16 位数据长度的字段 StillTime 表示基于 if 语句的静止时间。当字段 StillMode 的值不是“0x01”时,具有 16 位数据长度的区域是 reserved。

[0131] 数值“0x”指示用十六进制记数法表示。这也适用于类似的计数法。

[0132] 块 blkSTNTable() 管理由该块 blkPlayItem() 的播放项目所管理的剪辑 AV 流的属性、PID 数、记录介质上的记录位置、等等。

[0133] 图 14 示出表示块 blkPlayListMark() 的结构示例的语法。字段 Length 具有 32 位数据长度,表示紧跟该字段 Length 之后直至该块 blkPlayListMark() 的结尾的数据长度。

[0134] 字段 NumberOfPlatListMark 具有 16 位数据长度,表示包含在该块 blkPlayListMark() 内的播放列表标记的数目。根据下一 for 循环语句,描述用于字段 NumberOfPlayListMarks 所表示的数的播放列表标记的信息。

[0135] 在 for 循环语句中,紧跟具有 8 位数据长度的区域 reserved 之后,放置字段 MarkType。字段 MarkType 具有 8 位数据长度,表示标记的类型。将播放列表标记定义为两种标记,即入口标记(Entry Mark)和链接点(Link Point)。该字段 MarkType 表示这些类型之一。为了定义章,使用入口标记。因为链接点较少涉及本发明,所以将省略对它的描述。上述字段 NumberOfPlayListMarks 表示入口标记和链接点的总数值。

[0136] 字段 RefToPlayItemID 具有 16 位数据长度,描述表示放置标志的播放项目的标识信息 PlayItem_id。字段 MarkTimeStamp 具有 32 位数据长度,描述表示放置标志的点的的时间戳。字段 EntryESPID 具有 16 位数据长度,表示包含由标志表示的基本流的 TS 分组的 PID 值。字段 Duration 是具有 32 位数据长度的无符号整数,表示在 45kHz 时钟的单元中的测量结果。当存储在该字段 Duration 中的值是“0”时,该字段是无意义的。

[0137] 图 15 示出表示剪辑信息文件的结构示例的语法。字段 TypeIndicator 具有 32 位(4 字节)的数据长度,指示该文件是剪辑信息文件。字段 TypeIndicator2 具有 32 位(4 字节)数据长度,表示该剪辑信息文件的版本。

[0138] 该剪辑信息文件具有块 blkClipInfo()、块 blkSequenceInfo()、块 blkProgramInfo()、块 blkCPI()、块 blkClipMark() 和块 blkExtensionData()。每个字段 SequenceInfoStartAddress、字段 ProgramInfoStartAddress、字段 CPIStartAddress、字段 ClipMarkStartAddress 和字段 ExtensionDataStartAddress 都具有 32 位数据长度,表示它们相应的块的开始地址。

[0139] 字段 ExtensionDataStartAddress 表示块 blkExtensionData() 的开始地址,如从该剪辑信息文件的字节开始的相对字节数。相对字节数从“0”开始。如果该字段 ExtensionDataStartAddress 的值是“0”,则指示在该文件“index.bdmv”中没有块 blkExtensionData()。

[0140] 块 blkClipInfo() 在表示它们的开始地址的字段之后的、具有 96 位数据长度的区域 reserved 之后。块 blkClipInfo() 描述有关该剪辑信息文件所管理的剪辑 AV 流的信息。块 blkSequenceInfo() 描述用来作为一个序列而管理具有连续 STC 和 ATC(基于到达时间)的序列的信息。例如,块 blkProgramInfo() 描述有关编码系统和在该剪辑信息文件中管理的剪辑 AV 流的纵横比(aspect ratio)的信息。例如,块 blkCPI() 存储有关表示 AV 流中的特征点(如随机存取开始点)的特性点信息 CPI 的信息。

[0141] 块 blkClipMark() 描述索引点（跳转点），诸如添加到用于提示操作的剪辑的章位置。块 blkExtensionData() 是能够存储扩展数据的区域。因为剪辑信息文件中的块 blkClipMark() 和块 blkSequenceInfo() 较少涉及本发明，将省略对它们的描述。

[0142] 图 16 示出表示块 blkClipInfo() 的结构示例的语法。字段 Length 具有 32 位数据长度，表示紧跟该字段 Length 之后直至块 blkClipInfo() 的结尾的数据长度。紧跟具有 16 位数据长度的区域 reserved 之后，放置字段 ClipStreamType。

[0143] 字段 ClipStreamType 具有 8 位数据长度，表示剪辑 AV 流的类型。该字段 ClipStreamType 的值是固定的，例如“1”。字段 ApplicationType 具有 8 位数据长度，指示剪辑 AV 流（具有扩展名“m2ts”的文件）是如何多路复用的。当字段 ApplicationType 的值是“1”时，相对应的剪辑 AV 流再现为规则的运动画面。字段 ApplicationType 后面是具有 31 位数据长度的区域 reserved。

[0144] 标记 IsCC5 具有 1 位数据长度，指示相对应的剪辑和下一剪辑是否通过上述第一无缝连接（即字段 ConnectionCondition 的值是“5”的方法）而与播放列表的块 blkPlayItem() 连接。当标记 IsCC5 的值是“1”（二进制值）时，指示通过第一无缝连接来连接剪辑。

[0145] 字段 TSRecordingRate 表示以字节 / 秒为单位的剪辑 AV 流文件的记录速率。字段 NumberOfSourcePackets 表示包含在剪辑 AV 流内的源分组的数目。紧跟具有 1024 位数据长度的区域 reserved 之后，放置块 TSTypeInfoBlock()。块 TSTypeInfoBlock() 存储表示存储剪辑 AV 流的分组类型的信息。因为该块 TSTypeInfoBlock() 较少涉及本发明，所以将省略对它的详细描述。

[0146] 当上述标记 IsCC5 的值是“1”时，紧跟 if 语句之后，描述信息。紧跟 if 语句之后的具有 8 位数据长度的区域 reserved 之后，放置字段 FollowingClipStreamType。字段 FollowingClipStreamType 具有 8 位数据长度，表示与剪辑信息文件相对应的剪辑之后的剪辑的类型。紧跟具有 32 位数据长度的区域 reserved 之后，放置字段 FollowingClipInformationFileName。

[0147] 字段 FollowingClipInformationFileName 具有 40 位数据长度，描述与该剪辑信息文件相对应的剪辑的下一剪辑相对应的剪辑信息文件的文件名。下一字段 ClipCodecIdentifier 具有 32 位（4 字节）数据长度，表示下一剪辑的编码系统。在该示例中，字段 ClipCodecIdentifier 是根据 ISO 646 规定的系统编码的固定的四个字母的字符串“M2TS”。接下来，放置具有 8 位数据长度的区域 reserved。

[0148] 图 17 示出表示块 blkProgramInfo() 的结构示例的语法。字段 Length 具有 32 位数据长度，表示紧跟字段 Length 之后直至块 blkProgramInfo() 的结尾的数据长度。紧跟具有 15 位数据长度的区域 reserved 之后，描述具有 1 位数据长度的固定值“1”。

[0149] 字段 SPNProgramSequenceStart 具有 32 位数据长度，描述在相对应的剪辑 AV 流中节目序列开始所使用的源分组数。字段 ProgramMapPID 具有 16 位数据长度，表示包含可应用于节目序列的节目映射部分的 TS 分组的 PID 值。字段 NumberOfStreamsInPS 具有 8 位数据长度，表示在节目序列中定义的基本流的数目。字段 NumberOfStreamsInPS 后面是具有 8 位数据长度的区域 reserved。

[0150] 根据下一 for 循环语句，以值 [stream_index] 为循环变量，对于由字

段 NumberOfStreamInPS 表示的数来存储字段 StreamPID[stream_index] 和块 blkStreamCodingInfo(stream_index) 的对。字段 StreamPID[stream_index] 表示与由节目序列参照的 PMT(节目映射表)中描述的基本流相对应的 PID 值。下一块 blkStreamCodingInfo(stream_index) 描述有关由相对应的字段 StreamPID[stream_index] 表示的基本流的编码系统的信息。

[0151] 图 18 示出表示块 blkStreamCodingInfo(stream_index) 的结构示例的语法。字段 Length 具有 8 位数据长度,表示紧跟该字段 Length 之后直至该块 blkStreamCodingInfo(stream_index) 的结尾的数据长度。

[0152] 字段 Length 后面是具有 8 位数据长度的字段 StreamCodingType。字段 StreamCodingType 表示由值 [stream_index] 表示的基本流的编码类型。对于字段 StreamCodingType 的值,例如,定义为值“0x1B”、值“0x80”、值“0x81”、值“0x90”和值“0x91”。当字段 StreamCodingType 的值是“0x1B”时,指示流的编码类型是视频流。当字段 StreamCodingType 的值是“0x80”或者“0x81”,指示流的编码类型是音频流。当字段 StreamCodingType 的值是“0x90”时,指示流的编码类型是 OB 流。当字段 StreamCodingType 的值是“0x91”时,指示流的编码类型是 MB 流。根据下一 if 语句,基于该字段 StreamCodingType 的值而描述信息。

[0153] 例如,当字段 StreamCodingType 的值是“0x1B”时,由值 [stream_index] 表示的基本流是视频流,根据 if 语句,描述字段 VideoFormat、字段 FrameRate 和字段 AspectRatio,之后紧跟具有 2 位数据长度的区域 reserved,然后描述标志 CCFlag。在标志 CCFlag 之后,放置具有 17 位数据长度的区域 reserved 和具有 128 位数据长度的区域 reserved。

[0154] 字段 VideoFormat 具有 4 位数据长度,表示由值 [stream_index] 表示的视频数据的格式。图 19 示出由字段 VideoFormat 表示的视频数据的格式列表的示例。如图 19 所示,视频数据的格式用 4 位表示的值“0”至值“15”来标识。值“0”和值“8”至值“15”被保留。值“1”至值“7”分别表示视频数据的格式 480i、576i、480p、1080i、720p、1080p 和 576p。

[0155] 视频格式的后缀“i”和“p”分别表示隔行扫描和逐行扫描。另一方面,视频格式的数字表示扫描线的数目。这些视频格式已经由 ITU(国际电信联盟)-R、BT. 601-4(480i 和 576i)、ITU-R BT. 1358(576p)、SMPTE(电影与电视工程师协会)293M(480p)、SMPTE 274M(1080i 和 1080p) 以及 SMPTE296M(720p) 标准化。

[0156] 在块 blkStreamCodingInfo(stream_index) 中,字段 FrameRate 具有 4 位数据长度,表示由值 [stream_index] 表示的视频数据的帧速率(frame rate)。图 20 示出由字段 FrameRate 表示的帧速率列表。如图 20 所示,视频数据的帧速率用 4 位表示的值“0”至值“15”来标识。值“0”、值“5”、以及值“8”至值“15”被保留。值“1”至值“4”分别表示帧速率(字段频率)(24000/1001)Hz(即大约 23.97Hz)、24Hz、25Hz、以及(30000/1001)Hz(即大约 29.97Hz)。值“6”和值“7”分别表示帧速率 50Hz 和(60000/1001)Hz(即大约 59.94Hz)。

[0157] 在块 blkStreamCodingInfo(stream_index) 中,字段 AspectRatio 具有 4 位数据长度,表示由值 [stream_index] 表示的视频数据的纵横比。图 21 示出由字段 AspectRatio 表示的帧速率列表的示例。如图 21 所示,视频数据的纵横比用 4 位表示的字段 AspectRatio 的值“0”至值“15”标识。值“0”、值“1”、和值“4”至值“15”被保留。值“2”表示纵横比为 4 : 3。值“3”表示纵横比为 16 : 9。

[0158] 在块 `blkStreamCodingInfo(stream_index)` 中, 当字段 `StreamCodingType` 的值是“0x80”、“0x81”、“0x90”或者“0x91”时, 根据 `if` 语句的“`else if`”的描述, 基于由值表示的编码类型而描述信息。因为有关除了视频流以外的编码类型的流的描述较少涉及本发明, 所以将省略其详细描述。

[0159] 图 22 示出表示剪辑信息文件的块 `blk(PIC)` 的结构示例的语法。在诸如已经进行帧间压缩的 MPEG 流的编码流中, 解码开始位置限制在编码流的部分, 如 GOP (画面组) 的开始处。CPI (特征点信息) 是解码开始位置的信息的数据库。CPI 使文件的再现时间和地址相关联。换句话说, CPI 具有表示解码单元的开始位置的信息的表。

[0160] 当定义了这样的数据库时, 如果从任何时间再现视频数据, 则可基于再现时间, 通过参照 CPI 获取与再现位置相对应的文件的地址。因为该地址是解码单元的开始, 所以播放器能够从文件的地址中读取数据并且快速地显示画面。

[0161] 包含在 CPI 内的解码单元的开始位置 (在该示例中, GOP 的开始位置) 称为 EP (入口点) 条目。

[0162] 在图 22 中, 字段 `Length` 具有 32 位数据长度, 表示紧接字段 `Length` 之后直至块 `blkCPI()` 的结尾的数据长度。根据下一 `if` 语句, 除非字段 `Length` 的值是“0”, 否则紧跟具有 12 位数据长度的区域 `reserved` 之后, 放置字段 `CPIType`。字段 `CPIType` 具有 4 位数据长度, 表示 CPI 的类型。下一块 `blkEPMMap` 存储将 PTS 值与相对应的剪辑 AV 流文件的字节地址相关联的表。

[0163] 图 23 示出表示块 `blkEPMMap()` 的结构示例的语法。紧跟具有 8 位数据长度的区域 `reserved` 之后, 放置字段 `NumberOfStreamPIDEntries`。字段 `NumberOfStreamPIDEntries` 具有 8 位数据长度, 表示块 `blkEPMMap()` 中的块 `blkEPMMapForOneStreamPID` 的条目数目。根据 `for` 循环语句, 用值 `[k]` 作为循环变量, 对由字段 `NumberOfStreamPIDEntries` 表示的数来描述有关入口点的信息。

[0164] 在 `for` 循环语句中, 字段 `StreamPID[k]` 具有 16 位数据长度, 表示传输分组的 PID 值, 该传输分组传输从块 `blkEPMMapForOneStreamPID`, 如块 `blkEPMMap()` 的第 14 条目 (以下, 描述为第 `[k]` 块 `blkEPMMapForOneStreamPID`) 参照的基本流。

[0165] 在具有 10 位数据长度的区域 `reserved` 之后, 放置字段 `EPStreamType[k]`。字段 `EPStreamType[k]` 具有 4 位数据长度, 并且表示从第 `[k]` 块 `blkEPMMapForOneStreamPID` 参照的基本流的类型。字段 `NumberOfEPCoarseEntries[k]` 具有 16 位数据长度, 并且表示在第 `[k]` 块 `blkEPMMapForOneStreamPID` 的粗 (`coarse`) 单元搜索子表 (EP 粗表) 中的条目数目。字段 `NumberOfEPFineEntries[k]` 具有 18 位数据长度, 表示在第 `[k]` 块 `blkEPMMapForOneStreamPID` 的细 (`fine`) 单元搜索子表 (EP 细表) 中的条目数目。字段 `EPMMapForOneStreamPIDStartAddress[k]` 具有 32 位数据长度, 表示块 `blkEPMMap()` 中的第 `[k]` 块 `blkEPMMapForOneStreamPID` 的开始的相对字节位置。该值用从块 `blkEPMMap()` 的第一字节开始的字节数来表示。

[0166] 在描述了上述 `for` 循环语句之后, 在具有 16 的整数倍位的数据长度的填充字之后, 根据用值 `[k]` 作为循环变量的 `for` 循环语句, 对于由字段 `NumberOfStreamPIDEntries` 表示的数目来存储块 `blkEPMMapForOneStreamPID(EPStreamType[k], NumberOfEPCoarseEntries[k], 和 NumberOfEPFineEntries[k])`。换

句话说,自变量 `NumberOfEPCoarseEntries[k]` 表示存储在子表 (EP 粗表) 中的条目 `PTSEPCoarse` 和条目 `SPNEPCoarse` 的数目。同样地,自变量 `NumberOfEPFineEntries[k]` 表示存储在子表 (EP 细表) 中的条目 `PTSEPFine` 和条目 `SPNEPFie` 的数目。以下,自变量 `NumberOfEPCoarseEntries[k]` 和自变量 `NumberOfEPFineEntries[k]` 有时分别称为条目数 `Nc` 和条目数 `Nf`。

[0167] 图 24 示出表示块 `blkEMapForOneStreamPID(EP_stream_type, Nc, Nf)` 的结构示例的语法。为了描述块 `blkEMapForOneStreamPID(EP_stream_type, Nc, Nf)` 的语义,将描述作为存储在块 `blkEMapForOneStreamPID(EP_stream_type, Nc, Nf)` 中的数据源的条目的条目 `PTSEPStart` 和条目 `SPNEPStart` 的意义。

[0168] 条目 `PTSEPStart` 和与条目 `PTSEPStart` 相关联的条目 `SPNEPStart` 表示 AV 流的入口点。条目 `PTSEPFine` 和与条目 `PTSEPFine` 相关联的条目 `PTSEPCoarse` 从同一条目 `PTSEPStart` 中获取。另外,条目 `SPNEPFine` 和与条目 `SPNEPFine` 相关联的条目 `SPNEPCoarseassociated` 从同一条目 `SPNEPStart` 中获取。

[0169] 图 25 示出条目 `PTSEPCoarse` 和条目 `PTSEPFine` 的格式示例。PTS,即条目 `PTSEPStart`,具有 33 位数据长度。当条目 `PTSEPStart` 的 MSB 和 LSB 分别是第 32 位和第 0 位时,在图 25 中示出的示例中,从条目 `PTSEPStart` 的第 32 位至第 19 位的 14 位用于粗单元搜索的条目 `PTSEPCoarse`。条目 `PTSEPCoarse` 允许范围从 5.8 秒至 26.5 小时的分辨率的搜索。从条目 `PTSEPStart` 的第 19 至第 9 位的 11 位用于细单元搜索的条目 `PTSEPFine`。条目 `PTSEPFine` 允许范围从 5.7 毫秒至 11.5 秒的分辨率的搜索。条目 `PTSEPStart` 的第 19 位通常用于条目 `PTSEPCoarse` 和条目 `PTSEPFine`。从条目 `PTSEPStart` 的 LSB 端的第 0 位至第 8 位的 9 位是不使用的。

[0170] 图 26 示出条目 `SPNEPCoarse` 和条目 `SPNEPFine` 的格式示例。源分组数,即条目 `SPNEPStart` 具有 32 位的数据长度。当条目 `SPNEPStart` 的 MSB 和 LSB 是第 31 位和第 0 位时,在图 26 中示出的示例中,从条目 `SPNEPStart` 的第 31 位至第 0 位的所有位用于粗单元搜索的条目 `SPNEPCoarse`。从条目 `SPNEPStart` 的第 16 位至第 0 位的 17 位用于细单元搜索的条目 `SPNEPFine`。条目 `SPNEPFine` 允许直到大约 25MB (兆字节) 的 AV 流文件的探索。

[0171] 对于源分组数,可以使用条目 `SPNEPCoarse` 的 MSB 端的预定位数。例如,从条目 `SPNEPStart` 的第 31 位至第 16 位的 17 位可以用于条目 `SPNEPCoarse`,而从条目 `SPNEPStart` 的第 16 位至第 0 位的 17 位可以用于条目 `SPNEPFine`。

[0172] 基于上文,条目 `PTSEPStart` 和条目 `SPNEPStart` 定义如下。

[0173] 如图 25 所示,条目 `PTSEPStart` 是具有 33 位数据长度的无符号整数,表示具有 33 位数据长度的、从 AV 流的随机可存取的画面 (例如, IDR (瞬间解码刷新) 画面或者 I (内) 画面) 开始的视频存取单元的 PTS。

[0174] 如图 26 所示,条目 `SPNEPStart` 是具有 32 位数据长度的无符号整数,表示在 AV 流文件中包括与条目 `PTSEPStart` 相关联的视频存取单元的第一字节的源分组的地址。条目 `SPNEPStart` 被表示为源分组数。条目 `SPNEPstart` 被计数为从“1”开始的值,并且从 AV 流文件的第一源分组开始,对于每个源分组,该值递增 1。

[0175] 参考图 24,块 `blkEMapForOneStreamPID(EP_stream_type, Nc, Nf)` 描述第一 for 循环语句和第二 for 循环语句。第一 for 循环语句描述粗单元搜索的子表 (EP 粗表)。第

二 for 循环语句描述基于子表 (EP 粗表) 的搜索结果而执行的细单元搜索的子表 (EP 细表)。

[0176] 第一 for 循环语句之前紧接字段 EPFineTableStartAddress。字段 EPFineTableStartAddress 具有 32 位数据长度, 表示第二 for 循环的字段 ReservedEPFine[EP_fine_id] 的第一字节的开始地址, 如从块 blkEPMAPForOneStreamPID(EP_stream_type, Nc, Nf) 的第一字节开始的相对字节数。相对字节数从值“0”开始。

[0177] 第一 for 循环语句通过循环变量 [i] 重复由子表 (EP 粗表) 的条目数表示的次数。对于由条目数 Nc 表示的数来存储字段 RefToEPFineID[i] 条目 PTSEPCoarse[i] 和条目 SPNEPCoarse[i] 的对。在第一 for 循环语句中, 字段 RefToEPFine[i] 具有 18 位数据长度, 表示与由字段 RefToEPFineID[i] 后的字段 PTSEPCoarse[i] 表示的条目 PTSEPCoarse 相关联的条目 PTSEPFine 的子表 (EP 细表) 中的条目数。条目 PTSEPFine 和与该条目 PTSEPFine 相关联的条目 PTSEPCoarse 从同一条目 PTSEPStart 中获取。通过按第二 for 循环语句中描述的顺序定义的循环变量 [EP_fine_id] 的值获取字段 RefToEPFineID[i]。

[0178] 在第一 for 循环语句之后, 在填充字之后, 描述第二 for 循环语句。第二 for 循环语句用循环变量 [EP_fine_id] 重复由子表 (EP 细表) 的条目数 Nf 表示的次数。对于由条目数 Nf 表示的数来存储具有 1 位数据长度的字段 ReservedEPFine[EP_fine_id]、具有 3 位数据长度的字段 IEndPositionOffset[EP_fine_id]、具有 11 位数据长度的字段 PTSEPFine[EP_fine_id] 和具有 17 位数据长度的字段 SPNEPFine[EP_fine_id] 的对。在这些字段当中, 字段 PTSEPFine[EP_fine_id] 和字段 SPNEPFine[EP_fine_id] 存储基于循环变量 [EP_fine_id] 而分别从子表 (EP 细表) 参照的条目 PTSEPFine 和条目 SPNEPFine。

[0179] 条目 PTSEPCoarse 和条目 PTSEPFine 以及条目 SPNEPCoarse 和条目 SPNEPFine 按以下方式获取。假定子表 (EP 细表) 包括按相关数据 SPNEPStart 的值的升序排列的 Nf 个条目。每个条目 PTSEPFine 根据以下公式 (1) 从相对应的条目 PTSEPStart 中获取。

$$[0180] \quad \text{PTSEPFine}[\text{EP_fine_id}] = (\text{PTSEPStart}[\text{EP_fine_id}] \gg 9) / 2^{11} \quad (1)$$

[0181] 条目 PTSEPCoarse 和相对应的条目 PTSEPFine 之间的关系表示为以下公式 (2) 和 (3)。

$$[0182] \quad \text{PTSEPCoarse}[i] = (\text{PTSEPStart}[\text{RefToEPFineID}[i]] \gg 19) / 2^{14} \quad (2)$$

$$[0183] \quad \text{PTSEPFine}[\text{RefToEPFineID}[i]] =$$

$$[0184] \quad \text{PTSEPStart}[\text{RefToEPFineID}[i]] \gg 9) / 2^{11} \quad (3)$$

[0185] 根据以下公式 (4) 从相对应的条目 SPNEPStart 中获取每个条目 SPNEPFine。

$$[0186] \quad \text{SPNEPFine}[\text{EP_fine_id}] = \text{SPNEPStart}[\text{EP_fine_id}] / 2^{17} \quad (4)$$

[0187] 条目 SPNEPCoarse 和相对应的条目 SPNEPFine 之间的关系用以下公式 (5) 和 (6) 表示。

$$[0188] \quad \text{SPNEPCoarse}[i] = \text{SPNEPStart}[\text{RefToEPFineID}[i]] \quad (5)$$

$$[0189] \quad \text{SPNEPFine}[\text{RefToEPFineID}[i]] = \text{SPNEPStart}[\text{RefToEPFineID}[i]] / 2^{17} \quad (6)$$

[0190] 在上述公式 (1) 至 (6) 中, 符号“ $\gg x$ ”指示使用在 LSB 端具有超过 x 位的位数的位。

[0191] 接下来, 将描述用于存储扩展数据的块 blkExtensionData()。将该块

blkExtensionData() 定义为使得它能够存储预定扩展数据。在存储索引表的文件“index.bdmv”、存储播放列表的文件“xxxxx.mpls”和剪辑信息文件“zzzzz.dpi”中的每一个中,都可描述块 blkExtensionData()。

[0192] 图 27 示出表示块 blkExtensionData() 的结构示例的语法。字段 Length 具有 32 位数据长度,表示以字节表示的紧跟该字段 Length 之后直至块 blkExtensionData() 的结尾的数据长度。除非由字段 Length 表示的数据长度是“0”,否则描述之后的 if 语句的信息。

[0193] 字段 DataBlockStartAddress 具有 32 位数据长度,表示存储该语法的扩展数据的主体的块 DataBlock() 的开始地址,如从块 blkExtensionData() 的字节开始的相对字节数。换句话说,相对字节数从“0”开始。字段 DataBlockStartAddress 需要满足以下的 32 位对齐条件。

[0194] $\text{DataBlockStartAddress} \% 4 = 0$

[0195] 在具有 24 位数据长度的区域 reserved 之后,放置字段 NumberOfExtDataEntries。字段 NumberOfExtDataEntries 具有 8 位数据长度,并表示在块 blkExtensionData() 的块 DataBlock() 中存储的扩展数据的条目数目。扩展数据的条目存储获取扩展数据的主体所必需的信息。在该示例中,扩展数据的条目是由字段 ExtDataType、字段 ExtDataVersion、字段 ExtDataStartAddress 和字段 ExtDataLength 组成的块 ext_data_entry()。在块 blkExtensionData() 中,对于由字段 NumberOfExtDataEntries 根据第一 for 循环语句表示的数,存在块 ext_data_entry()。

[0196] 字段 ExtDataType 具有 16 位数据长度,指示包含在该块 blkExtensionData() 内的扩展数据是用于记录设备的扩展数据。该字段 ExtDataType 的值是用来标识扩展数据的第一值。字段 ExtDataType 的值可定义为由包括该块 blkExtensionData() 的标准的许可者 (licensor) 分配。字段 ExtDataVersion 是用来标识扩展数据的第二值。字段 ExtDataVersion 可定义为表示该扩展数据的版本号。在该块 blkExtensionData() 中,当这些条目的字段 ExtDataType 和字段 ExtDataVersion 的值相同时,不能存在块 ext_data_entry() 的两个或者更多条目。

[0197] 字段 ExtDataStartAddress 具有 32 位数据长度,表示与包含该字段 ExtDataStartAddress 的扩展数据的条目 (块 ext_data_entry()) 相对应的扩展数据的开始地址。字段 ExtDataStartAddress 表示扩展数据 ext_data 的开始地址,如从块 blkExtensionData() 的字节开始的相对字节数。字段 ExtDataStartAddress 需要满足以下 32 位对齐条件。

[0198] $\text{ExtDataStartAddress} \% 4 = 0$

[0199] 字段 ExtDataLength 具有 32 位数据长度,并且表示与包含字段 ExtDataStartAddress 的扩展数据的条目 (块 ext_data_entry()) 相对应的扩展数据的数据长度。数据长度以字节表示。

[0200] 当对于由字段 NumberOfExtDataEntries 表示的数来描述扩展数据的条目 (块 ext_data_entry()) 时,具有 16 位数据长度且由任何数据序列组成的字段 padding_word 重复 L1 次,作为两个字段的对。其后,描述存储扩展数据的主体的块 DataBlock()。块 DataBlock() 存储扩展数据的至少一个条目。扩展数据 ext_data 的每个条目基于上述字段

ExtDataStartAddress 和字段 ExtDataLength 而从块 DataBlock() 中获取。

[0201] 图 28 示出块 blkExtensionData() 中的数据条目的参照关系。字段 Length 表示紧接字段 Length 之后直至块 blkExtensionData() 的结尾的数据长度。字段 DataBlockStartAddress 表示块 DataBlock() 的开始位置。对于由字段 NumberOfExtDataEntries 表示的数来描述块 ext_data_entry。具有任何数据长度的字段 padding_word 放置在最后的块 ext_data_entry 和块 DataBlock() 之间。

[0202] 在块 DataBlock() 中, 放置由块 ext_data_entry() 表示的扩展数据 ext_data。每个扩展数据 ext_data 的位置和数据长度由相对应的块 ext_data_entry() 的字段 ExtDataStartAddress 和字段 ExtDataLength 表示。因此, 块 DataBlock() 中的扩展数据 ext_data 的顺序不需要与相对应的块 ext_data_entry() 的顺序相匹配。

[0203] 当将扩展数据构造成存储扩展数据的主体的块 DataBlock() 和存储用于包含在块 DataBlock() 内的扩展数据的存取信息的块 ext_data_entry() 的两层时, 能够存储扩展数据的多个条目。

[0204] 接下来, 将描述创建和读取上述扩展数据的方法的示例。图 29 是示出块 blkExtensionData() 中的写入数据的处理示例的流程图。图 29 示出扩展数据作为第 (n+1) 条目添加至块 blkExtensionData() 中以便重写块 blkExtensionData() 的示例。

[0205] 首先, 在步骤 S10 中, 获取将要写入的扩展数据的数据长度并将其设置为字段 ExtDataLength[n+1] 的值。符号 “[n+1]” 表示第 (n+1) 个条目编号。在下一步骤 S11, 检查在当前块 blkExtensionData() 中列出的块 ext_data_entry() 的字段 ExtDataLength 和字段 ExtDataStartAddress 的值, 并获取块 DataBlock() 的使用状态。

[0206] 在下一步骤 S12 中, 确定块 DataBlock() 是否具有等于或者大于由表示将要写入的扩展数据的数据长度的字段 ExtDataLength[n+1] 所表示的数据长度的连续空闲区域。当确定结果指示块 DataBlock() 具有这样的区域时, 处理流程前进到步骤 S14。

[0207] 相反, 当确定结果指示块 DataBlock() 不具有超过由字段 ExtDataLength[n+1] 表示的数据长度的连续空闲区域时, 处理流程前进到步骤 S13。在步骤 S13 中, 增加块 blkExtensionData() 的字段 Length 的值, 使得在块 DataBlock() 中创建等于或者大于由字段 ExtDataLength[n+1] 表示的数据长度的连续空闲区域。在创建这样的空闲区域之后, 处理流程前进到步骤 S14。

[0208] 在步骤 S14 中, 决定写入扩展数据的区域的开始地址, 使得将开始地址的值设置为字段 ExtDataStartAddress[n+1]。在下一步骤 S15, 从字段 ExtDataStartAddress[n+1] 中写入在步骤 S10 中设置的具有由字段 ExtDataLength[n+1] 表示的数据长度的扩展数据 ext_data[n+1]。

[0209] 在写入数据之后, 处理流程前进到步骤 S16。在步骤 S16 中, 将字段 ExtDataLength[n+1] 和字段 ExtDataStartAddress[n+1] 添加到块 ext_data_entry() 中。

[0210] 在上文中, 假定将要重写的块 blkExtensionData() 已经从诸如盘的记录介质中读取并且存储在记录设备的存储器中。因此, 由系统通过在步骤 S13 中改变字段 Length 的值来执行块 blkExtensionData() 的增加。系统适当地分配存储器。

[0211] 图 30 是示出从块 blkExtensionData() 中读取扩展数据的处理示例。图 30 中示出的流程图的处理可应用于只再现记录介质和可记录的记录介质两者。首先, 在步骤 S20,

从将要读取的扩展数据所基于的标准获取字段 ExtDataType 的值。在步骤 S21, 从将要读取的扩展数据的类型中获取字段 ExtDataVersion 的值。

[0212] 在下一步骤 S22 中, 逐一读取在块 blkExtensionData() 中列出的每个块 ext_data_entry()。在步骤 S23 中, 确定每个块 ext_data_entry() 的字段 ExtDataType 和字段 ExtDataVersion 的值是否与在上述步骤 S20 和 S21 中获取的字段 ExtDataType 和字段 ExtDataVersion 的值相匹配。

[0213] 当确定结果指示它们不匹配时, 处理流程前进到步骤 S26。在步骤 S26 中, 确定是否已经读取块 blkExtensionData() 中列出的所有块 ext_data_entry()。当确定结果指示已经读取所有块 ext_data_entry() 时, 处理流程前进到步骤 S27。在步骤 S27 中, 因为在块 blkExtensionData() 中没有将要读取的扩展数据, 所以处理完成。相反, 当确定结果指示没有读取所有块 ext_data_entry() 时, 处理流程回到步骤 S22。在步骤 S22 中, 读取下一块 ext_data_entry()。

[0214] 当在步骤 S23 中的确定结果指示块 ext_data_entry() 的字段 ExtDataType 和字段 ExtDataVersion 的值与在上述步骤 S20 和 S21 中获取的字段 ExtDataType 和字段 ExtDataVersion 的值相匹配时, 处理流程前进到步骤 S24。在该示例中, 假定它们与块 blkExtensionData() 的第 [i] 条目相匹配。

[0215] 在步骤 S24, 从块 ext_data_entry() 的第 [i] 条目中读取字段 ExtDataLength[i] 的值和字段 ExtDataStartAddress[i] 的值。在步骤 S25, 从由在步骤 S24 中读取的字段 ExtDataStartAddress[i] 所表示的地址中读取作为字段 ExtDataLength[i] 所表示的数据长度的数据。

[0216] 接下来, 将描述存储扩展数据的且可在上述索引文件“index.bdmv”、电影对象文件“MovieObject.bdmv”、播放列表文件“xxxxx.mpls”和剪辑信息文件“zzzzz.clpi”中定义的扩展数据块 blkExtensionData()。

[0217] 首先, 将描述在索引文件“index.bdmv”中定义的扩展数据块的示例。在该示例中, 将描述向每个播放列表中添加对可记录的记录介质唯一的属性信息的扩展数据块。图 31 示出表示文件“index.bdmv”中的块 blkExtensionData() 的块 DataBlock() 的结构示例 (参考图 27) 的语法。在图 31 中示出的示例中, 块 DataBlock() 被描述为块 blkIndexExtensionData()。

[0218] 首先, 参考图 27, 分别设置块 blkExtensionData() 的字段 ExtDataType 和字段 ExtDataVersion 的值“0x1000”和值“0x0100”。在这些字段 ExtDataType 和字段 ExtDataVersion 中描述的值参考存储在例如再现设备端的 ROM (只读存储器) 中的表而标识。块 blkIndexExtensionData() 存储在由块 DataBlock() 中的字段 ExtDataStartAddress 和字段 ExtDataLength 表示的区域中。

[0219] 在块 blkIndexExtensionData() 中, 字段 TypeIndicator 描述根据 ISO 646 中规定的编码系统编码的四个字母的字符串。四个字母的字符串表示随后的数据类型。在图 31 中示出的示例中, 字段 TypeIndicator 描述根据 ISO 646 中规定的系统编码的四个字母的字符串“IDEX”, 指示随后的数据类型是索引文件的扩展数据。

[0220] 字段 TypeIndicator 后面是具有 32 位数据长度的区域 reserved, 之后紧跟具有 32 位数据长度的字段 TableOfPlayListStartAddress。字段 TableOfPlayListStartAddress

表示基于该块 blkIndexExtensionData() 的开始的块 blkTableOfPlayList() 的开始地址。

[0221] 字段 TableOfPlayListStartAddress 后面是具有 32 位数据长度的字段 MakersPrivateDataStartAddress。字段 MakersPrivateDataStartAddress 表示基于块 blkIndexExtensionData() 的开始的块 blkMakersPrivateData() 的开始地址。在具有 192 位数据长度的区域 reserved 之后,放置块 blkUIAppInfoAVCHD()。具有 16 位数据长度的填充字 padding_word 重复由值 N1 表示的次数。其后,放置块 blkTableOfPlayList()。其后,具有 16 位数据长度的填充字 padding_word 重复由值 N2 表示的次数。其后,放置块 blkMakersPrivateData()。块 blkMakersPrivateData() 后面是具有 16 位数据长度的填充字 padding_word。填充字 padding_word 重复由值 N3 表示的次数。

[0222] 因为块 blkUIAppInfoAVCHD() 和块 blkMakersPrivateData() 较少涉及本发明,所以将省略对它们的描述。

[0223] 图 32 示出表示上述块 blkTableOfPlayList() 的结构示例的语法。字段 Length 具有 32 位数据长度,表示以字节表示的紧接字段 Length 之后直至块 blkTableOfPlayList() 的结尾的字节的长度。字段 Length 后面是描述有关再现重放标题所使用的播放列表的信息的块 blkFirstPlaybackTitlePlayLists() 和描述有关菜单标题的信息的块 blkMenuTitlePlayLists()。因为这些块 blkFirstPlaybackTitlePlayLists() 和块 blkMenuTitlePlayLists() 较少涉及本发明,所以将省略对它们的描述。

[0224] 其后,放置具有 16 位数据长度的字段 NumberOfTitlePlayListPair。字段 NumberOfTitlePlayListPair 描述用来再现除了重放标题和菜单标题之外的标题的播放列表数。根据下一 for 循环语句,对于字段 NumberOfTitlePlayListPair 所表示的数目描述块 blkMovieTitlePlayListPair()。块 blkMovieTitlePlayListPair() 包含字段 PlayListFileName、字段 PlayListAttribute 和字段 RefToTitleId。换句话说,块 blkMovieTitlePlayListPair() 将信息构造为由用于由该 for 循环语句表示的第 [i] 播放列表的文件名、属性和参照标题 ID 所组成。

[0225] for 循环语句中的条目基于记录它们的顺序。换句话说,当添加一个播放列表时,字段 NumberOfTitlePlayListPair 的值递增“1”,而所添加的播放列表的信息被附加写入现有播放列表的信息之后。

[0226] 字段 PlayListFileName 具有 40 位 (5 字节) 数据长度,描述根据 ISO 646 中规定的编码系统编码的播放列表的文件名。字段 PlayListFileName 后面是具有 6 位数据长度的区域 reserved,之后紧跟字段 PlayListAttribute。字段 PlayListAttribute 具有 2 位数据长度,表示分配给当前播放列表的属性。播放列表基于它们的原因 (cause) 而分类为与剪辑一起创建的第一类型、与用现有标题或者全部或者部分的播放列表创建的播放列表相对应的第二类型、和用于再现菜单的第三类型。取决于播放列表的类型给每个播放列表分配属性“Real”(第一类型)、属性“Virtual”(第二类型)、或者属性“Menu”(第三类型)。

[0227] 以下,分配属性“Real”的播放列表称为真实播放列表。分配属性“Virtual”的播放列表称为虚拟播放列表。分配属性“Menu”的播放列表称为菜单播放列表。

[0228] 字段 RefTo TitleId 描述当创建播放列表时,在同一循环的字段 PlayListFileName 中表示的播放列表所属的标题的 ID(编号)。作为更具体的示例,字段 RefTo TitleId 描述索引文件“index.bdmv”中的块 blkIndexes() 的相应值 title_id。当

仅仅从第一重放标题中再现该播放列表时,字段 RefTo TitleId 的值是第一固定值,例如“0xFFFF”。另一方面,当仅仅从菜单标题中再现该播放列表时,字段 RefTo TitleId 的值是第二固定值,例如“0xFFFE”。

[0229] 接下来,将简单描述真实播放列表、虚拟播放列表和菜单播放列表。例如,分配属性“Real”的真实播放列表被与剪辑一起创建并记录在盘上。因为真实播放列表是表示素材的第一播放列表,所以也将其称为原始播放列表。例如,在真实播放列表中,所创建剪辑的开始和末尾分别被指定为 IN 点和 OUT 点。

[0230] 真实播放列表是参照作为流的实体的剪辑的播放列表。当新创建剪辑时,总是创建参照所创建的剪辑的真实播放列表。换句话说,没有不从任何真实播放列表参照的剪辑。因此,关于盘的真实播放列表的总的再现时间与记录在盘上的剪辑的总的再现时间相匹配。换句话说,从用户的观点来看,盘的剩余可记录时间与真实播放列表或者仅仅由真实播放列表组成的标题的可记录时间相匹配。

[0231] 作为素材的真实播放列表和剪辑之间总是具有一对一的关系。因此,当例如,通过编辑来删除真实播放列表时,也从盘上删掉了相应的剪辑。另外,当删除剪辑所参照的真实播放列表部分时,也删除了与真实播放列表相对应的剪辑部分。因此,当编辑真实播放列表时,也改变了记录在盘上的剪辑的实体。因此,例如,该编辑称为实体编辑或者真实编辑。

[0232] 分配属性“Virtual”的虚拟播放列表是用现有标题或者部分或全部播放列表而创建的播放列表。虚拟播放列表是通过给现有剪辑设置 IN 点和 OUT 点以及参照由 IN 点和 OUT 点定义的区域而创建的播放列表。

[0233] 例如,虚拟播放列表指定用于上述真实播放列表的 IN 点和 OUT 点。例如,虚拟播放列表指定用于多个真实播放列表的每一个的 IN 点和 OUT 点,以及指定由这些 IN 点和 OUT 点指定的多个区域的再现顺序。另外,基于虚拟播放列表,可创建另一虚拟播放列表。换句话说,可创建为多个虚拟播放列表之一或每一个指定 IN 点和 OUT 点的虚拟播放列表。

[0234] 可快速创建虚拟播放列表,而不需要改变例如具有大尺寸的剪辑(流实体)。另外,可通过删除与剪辑的参照关系来删除虚拟播放列表,而不需要改变剪辑的实体。因为编辑虚拟播放列表时不需要改变剪辑的实体,所以该编辑称为虚拟编辑等等。

[0235] 分配属性“Menu”的菜单播放列表是用于再现菜单的播放列表。当创建或者更新菜单时创建菜单播放列表。菜单播放列表是从电影对象中调用的用于再现菜单标题的播放列表。

[0236] 接下来,将描述为播放列表文件“xxxxx.mpls”定义的扩展数据块的示例。图 33 示出表示在播放目录文件“xxxxx.mpls”中的块 blkExtensionData() 的块 DataBlock() 的结构示例(参考图 27)的语法。在图 33 中示出的示例中,块 DataBlock() 被表示为块 blkPlaylistExtensionData()。

[0237] 首先,参考图 27,假定播放列表文件“xxxxx.mpls”中的块 blkExtensionData() 的字段 ExtDataType 和字段 ExtDataVersion 的值分别是预定值,例如,值“0x1000”和值“0x0100”,如上述示例中所示。

[0238] 在块 blkPlaylistExtensionData() 中,字段 TypeIndicator 描述已经根据 ISO646 中规定的编码系统编码的预定的四个字母的字符串,表示随后的数据类型。在该字段 TypeIndicator 中描述的字符串指示随后的数据类型是播放列表文件的扩展数据。

[0239] 字段 TypeIndicator 后面是具有 32 位 (4 字节) 数据长度的区域 reserved, 之后紧跟具有 32 位数据长度的字段 PlaylistMarkExtStartAddress, 再之后紧跟具有 32 位数据长度的字段 MakersPrivateDataStartAddress。字段 PlaylistMarkExtStartAddress 和字段 MakersPrivateDataStartAddress 表示基于块 blkPlaylistExtensionData() 的开始的块 blkPlaylistExt() 和块 blkMakersPrivateData() 的开始地址。

[0240] 字段 MakersPrivateDataStartAddress 后面是具有 192 位数据长度的区域 reserved, 之后紧跟块 blkPlaylistMenu()。具有 16 位数据长度的填充字 padding_word 重复由值 N1 表示的次数。其后, 放置块 blkPlaylistMarkExt()。另外, 具有 16 位数据长度的填充字 padding_word 重复由值 N2 表示的次数, 之后紧跟块 blkMakersPrivateData()。块 blkMakersPrivateData() 后面是具有 16 位数据长度的填充字 padding_word。填充字 padding_word 重复由值 N3 表示的次数。

[0241] 图 34 示出表示块 blkPlaylistMeta() 的结构示例的语法。字段 Length 具有 32 位数据长度, 表示直接在该字段 Length 之后直至该块 blkPlaylistMenu() 的结尾的数据长度。字段 Length 后面是每个都具有 16 位数据长度的字段 MakerID 和字段 MakerModelCode。字段 MakerID 和字段 MakerModelCode 分别描述已经更新该播放列表文件的制造者和标识该制造者的模型的信息。

[0242] 字段 MakerModelCode 后面是具有 32 位数据长度的区域 reserved, 之后紧跟具有 16 位数据长度的字段 RefToMenuThumbnailIndex。当存在表示由该播放列表文件再现的剪辑的序列的缩略图图像时, 该字段 RefToMenuThumbnailIndex 描述标识缩略图图像的缩略图编号。其后, 放置具有 8 位数据长度的块 blkTimeZone()。当更新该播放列表文件时, 在块 blkTimeZone() 中描述表示已经为设备设置的时区的信息。其后, 放置具有 56 位数据长度的字段 RecordTimeAndData。字段 RecordTimeAndData 表示该播放列表文件更新的时间和数据。

[0243] 字段 RecordTimeAndData 后面是具有 8 位数据长度的区域 reserved, 之后紧跟每个都具有 8 位数据长度的字段 PlaylistCharacterSet 和字段 PlaylistNameLength, 再之后紧跟具有 255 字节数据长度的字段 PlaylistName。这些字段 PlaylistCharacterSet、字段 PlaylistNameLength、和字段 PlaylistName 描述关于由该播放列表文件表示的播放列表的名字的信息。

[0244] 换句话说, 字段 PlaylistCharacterSet 表示在字段 PlaylistName 中描述的字符串的字符集。字段 PlaylistNameLength 表示在字段 PlaylistName 中描述的播放列表名字的字节长度。字段 PlaylistName 描述分配给该播放列表的名字。在字段 PlaylistName 中, 由字段 PlaylistNameLength 表示的从左边开始的字节数是该播放列表的名字的有效字符串。在字段 PlaylistName 中, 由字段 PlaylistNameLength 表示的有效字符串之后可以跟随任何值。

[0245] 字段 PlaylistName 后面是块 Additional_data()。该块 Additional_data() 是存储附加信息的保留区域。在块 Additional_data() 中, 具有由具有 32 位数据长度的字段 Length2 表示的字节数的数据长度的区域被保留。

[0246] 图 35 示出表示播放列表文件中的块 blkMakersPrivateData() 的结构示例的语法。块 blkMakersPrivateData() 是描述有关该播放列表文件的制造者的唯

一信息的块。字段 Length 具有 32 位数据长度,表示紧接字段 Length 之后直至该块 blkMakersPrivateData() 的结尾的数据长度。当该字段 Length 的值是“0”时,指示该 MakersPrivateData() 不描述信息。除非字段 Length 的值是“0”,否则描述之后的 if 语句的信息。

[0247] 字段 DataBlockStartAddress 具有 32 位数据长度,表示在该语法中存储制造者的唯一信息的主体的块 DataBlock() 的开始地址,如从该块 blkMakersPrivateData() 开始的相对字节数。紧跟具有 24 位数据长度的区域 reserved 之后,放置具有 8 位数据长度的字段 NumberOfMakerEntries。

[0248] 根据下一 for 语句,对于由字段 NumberOfMakerEntries 表示的数描述扩展数据的条目,即字段 MakerID、字段 MakerModelCode、字段 MpdStartAddress 和字段 MpdLength。每个字段 MakerID 和字段 MakerModelCode 都具有 16 位数据长度,分别表示制造者的标识符信息和制造者的模型的标识符信息。每个字段 MpdStartAddress 和字段 MpdLength 都具有 32 位数据长度,分别表示从块 blkExtensionData() 开始的存储扩展数据的主体的块 DataBlock() 的开始地址,如块 DataBlock() 的相对字节数和数据长度。

[0249] 当对于由字段 NumberOfMakerEntries 表示的数描述扩展数据的条目时,具有 16 位数据长度且由任何数据序列组成的字段 padding_word 重复 L1 次,如两个字段的对。其后,描述存储扩展数据的主体的块 DataBlock()。块 DataBlock() 存储至少一个扩展数据 ext_data。换句话说,制造者的唯一扩展数据存储在块 DataBlock() 中,分别用于由字段 MakerID 和字段 MakerModelCode 表示的每个制造者和每个模型。扩展数据的每个条目基于上述字段 MpdStartAddress 和字段 MpdStartAddress 而从块 DataBlock() 中获取。

[0250] 因为播放列表文件的扩展数据的块 blkPlayListExtensionData() 中的块 blkPlayListMarkExt() 较少涉及本发明,所以将省略对它的描述。

[0251] 接下来,将描述为剪辑信息文件“zzzzz.clpi”定义的扩展数据块的示例。图 36 示出表示在剪辑信息文件的块 blkExtensionData() 中的块 DataBlock() 的结构示例(参考图 27)的语法。在图 36 中示出的示例中,块 DataBlock() 表示为块 blkClipExtensionData()。

[0252] 在块 blkClipExtensionData() 中,字段 TypeIndicator 描述由表示随后的数据类型的并且已经根据 ISO 646 中规定的编码系统编码的四个字母组成的预定字符串。在该字段 TypeIndicator 中描述的字符串指示随后的数据类型是剪辑信息文件的扩展数据。

[0253] 字段 TypeIndicator 后面是具有 32 位(4 字节)数据长度的区域 reserved,之后紧跟每个都分别具有 32 位数据长度的字段 ProgramInfoExtStartAddress 和字段 MakersPrivateDataStartAddress。字段 ProgramInfoExtStartAddress 和字段 MakersPrivateDataStartAddress 表示基于该块 blkClipExtensionData() 的开始的块 blkClipExtensionData() 中的块 blkProgramInfoExt() 和块 blkMakersPrivateData() 的开始地址。

[0254] 字段 MakersPrivateDataStartAddress 后面是具有 192 位数据长度的区域 reserved,之后紧跟块 blkClipInfoExt()。具有 16 位数据长度的填充字 padding_word 重复由值 N1 表示的次数,之后紧跟块 blkProgramInfoExt()。其后,具有 16 位数据长度的填充字 padding_word 重复由值 N2 表示的次数,之后紧跟块 blkMakersPrivateData()。

块 blkMakersPrivateData() 后面是具有 16 位数据长度的填充字 padding_word。填充字 padding_word 重复由值 N3 表示的次数。

[0255] 图 37 示出表示块 blkClipInfoExt() 的结构示例的语法。字段 Length 具有 32 位数据长度, 表示紧跟该字段 Length 之后直至该块 blkProgramInfoExt() 的结尾的数据长度。其后, 紧跟具有 8 位数据长度的区域 reserved 之后, 放置字段 NumberOfProgramSequence。

[0256] 字段 NumberOfProgramSequence 具有 8 位数据长度, 表示包含在该块 blkProgramInfoExt() 内的节目序列的信息的条目数。在该示例中, 字段 NumberOfProgramSequence 的值固定为“1”。根据下一第一 for 循环语句, 字段 NumberOfStreamCodingInfoExtInPs[i] 和下一第二 for 循环语句重复由字段 NumberOfProgramSequence 表示的次数。第二 for 循环语句对于由具有 8 位数据长度的字段 NumberOfStreamCodingInfoExtInPs[i] 表示的数目来存储字段 StreamPIP[i][j] 和块 blkStreamCodingInfoExt(i, j) 的对。

[0257] 字段 StreamPIP[i][j] 具有 16 位数据长度。字段 StreamPIP[i][j] 组成由值 [i] 和值 [j] 指定的表, 并且表示与在从第 [i] 节目序列参照的 PMT(程序映射表) 中描述的基本流相对应的 PID 值。下一块 blkStreamCodingInfoExt(i, j) 描述关于由值 [i] 和值 [j] 指定的基本流的编码系统的信息。

[0258] 图 38 示出表示块 blkStreamCodingInfoExt(i, j) 的结构示例的语法。字段 Length 具有 8 位数据长度, 并且表示紧接字段 Length 之后直至块 blkStreamCodingInfoExt(i, j) 的结尾的数据长度。下一字段 StreamCodingType 具有 8 位数据长度, 表示相对应的流的编码类型。当字段 StreamCodingType 的值是“0x1B”时, 之后紧跟 if 语句, 描述信息。具有 8 位数据长度的第一字段 HorizontalSize 表示屏幕的垂直尺寸, 即屏幕的线数。因为在块 blkStreamCodingInfoExt 中字段 HorizontalSize 之后的信息较少涉及本发明, 所以将省略对其的描述。

[0259] 另外, 因为剪辑信息文件的块 blkClipExtensionData() 中的块 ClipEnfoExt() 和块 blkMakersPrivateData() 较少涉及本发明, 所以将省略对它们的描述。

[0260] 接下来, 将简单描述虚拟播放器。当将具有上述数据结构的盘加载到播放器中时, 播放器需要将在已经从盘中读取的电影对象等中包含的命令转换成用来控制播放器的硬件的唯一命令。播放器已经将用来对这些命令执行这样的转换的软件存储在播放器中内置的 ROM(只读存储器) 中。因为该软件通过盘和播放器使播放器按照 AVCHD 格式的规定进行操作, 所以该软件称为虚拟播放器。

[0261] 图 39A 和图 39B 示出虚拟播放器的操作的概略。图 39A 示出虚拟播放器加载盘的示例。当盘被加载到播放器中并且初始存取盘(在步骤 S30) 时, 初始化用来存储共用于一个盘的公共参数的寄存器(在步骤 S31)。在下一步骤 S32, 从盘中读取并且执行在电影对象等中描述的程序。例如, 初始存取表示: 当加载盘时, 从盘中开始再现指示数据。

[0262] 图 39B 示出在用户(例如)按下处于停止状态的播放键、以命令播放器从盘中再现数据的情况下的播放器的操作示例。在初始停止状态(在步骤 S40) 中, 例如, 用户通过遥控命令器等(U0: 用户操作) 命令播放器从盘中再现数据。当命令播放器从盘中再现数据时, 初始化寄存器, 即公共参数(在步骤 S41)。在下一步骤 S42, 电影对象执行阶段(phase) 开始。

[0263] 接下来,将参考图 40 描述电影对象执行阶段中的播放列表的再现。假定通过 U0 等等,播放器已经命令开始再现盘中的标题 #_1 的内容。当已经命令播放器开始再现盘中的内容时,播放器参照上述图 2 中示出的索引表 (IndexTable),并获取与标题 #_1 的内容的再现相对应的对象编号。假定完成标题 #_1 内容的执行的对象编号是 #_1,播放器开始执行电影对象 #_1。

[0264] 在图 40 中示出的示例中,假定电影对象 #_1 中描述的程序由两行组成,第一行的命令是“播放 Playlist(1)”,播放器开始再现播放列表 #_1。播放列表 #_1 由一个或多个播放项目组成。播放器连续地再现播放项目。在播放器已经完成播放列表 #_1 的所有播放项目的再现之后,播放器恢复 (restore) 电影对象 #_1 的执行,然后执行第二行的命令。在图 40 中示出的示例中,第二行的命令是“跳转 MenuTitle”。播放器执行该命令以及开始执行完成索引表中描述的菜单标题 (MenuTitle) 的电影对象。

[0265] 接下来,将描述本发明的实施例。根据本发明,当将剪辑记录到记录介质中时,基于将要记录的剪辑的章被附加写入到现有的播放列表中。当将章附加写入到现有的播放列表中时,基于播放列表的预定约束而确定基于剪辑的章是否能够被附加写入到播放列表中。当确定结果指示能够将章附加写入时,基于剪辑的章被附加写入播放列表中。相反,当确定结果指示不能将章附加写入时,新创建播放列表,并且向其登记基于所记录剪辑的章。

[0266] 当执行这样的记录控制时,用户可记录剪辑,而无需识别播放列表的约束。因此,设备的可操作性得以改善,并且用户友好性变得很好。

[0267] 图 41 示出根据本发明的实施例的记录设备的结构示例的概略。该记录设备将 AV 流记录到记录介质,其中该 AV 流的输入数字视频数据和数字音频数据已经在预定系统中压缩编码并且多路复用。

[0268] 图 41 中示出的记录设备可以是将从外部输入的视频数据和音频数据记录到记录介质中的单个记录设备,或者是与具有光学系统、图像捕捉装置等的相机块一起用于摄像机设备的、且将基于所捕捉图像信号的视频数据记录到记录介质的记录块。

[0269] 可预期存在多种可应用的压缩编码系统和多路复用系统。例如,在 H. 264|AVC 中定义的系统可以应用于本发明的实施例的压缩编码系统。作为替代,可基于 MPEG2 系统而执行压缩编码。另一方面,对于多路复用系统,例如,可使用 MPEG2 系统。在下文中,假定视频数据基于 H. 264|AVC 中规定的系统而压缩编码,以及视频数据和音频数据基于 MPEG2 系统中规定的系统而多路复用。

[0270] 例如,控制部分 30 由 CPU(中央处理单元)、RAM(随机存取存储器)、ROM(只读存储器)等等(未示出)组成。控制部分 30 使用 RAM 作为工作存储器,基于在 ROM 中存储的程序和数据而控制记录设备的记录部分 10 的每个部分。在图 41 中,省略连接控制部分 30 和记录部分 10 的每个部分的线以防止附图变得复杂。

[0271] UI(用户接口)部分 31 具有操作开关,根据操作开关的操作,用户使用该操作开关来操作记录设备和输出控制信号。控制信号被提供给控制部分 30。通过基于根据用户的操作从 UI 部分 31 提供的控制信号而执行的软件的处理,控制部分 30 控制记录部分 10 的每个部分的操作。例如,控制部分 30 根据 UI 部分 31 的操作而控制记录设备的记录开始和停止操作。

[0272] 例如,控制部分 30 执行应用软件,其在由预定程序组成并提供基本软件功能的

OS(操作系统)上提供用户界面,并控制记录部分 10 的每个部分。另外,控制部分 30 执行协调记录部分 10 的每个部分的软件和硬件的驱动器程序。OS 还提供管理在记录介质 20 上记录的文件和数据的文件系统。稍后将描述文件系统。应用软件通过由 OS 提供的文件系统存取在记录介质 20 上记录的文件。

[0273] 从终端 40 输入基带数字视频数据。当输入数字视频数据时,从终端 41 输入基带数字音频数据。

[0274] 数字视频数据从终端 40 输入到记录部分 10 中,并提供给视频编码器 11。视频编码器 11 根据预定系统对所提供的数字视频数据进行压缩编码。在基于 H. 264|AVC 中规定的系统而执行压缩编码的该示例中,例如,使用 DCT(离散余弦变换)和屏幕内预测,执行帧内压缩。使用运动向量执行帧间压缩。执行加密编码以便改善压缩效率。已经由视频编码器 11 压缩编码的数字视频数据被作为 H. 264|AVC 基本流(ES)而提供给多路复用器(MUX)13。

[0275] 数字音频数据从终端 41 输入到记录部分 10 并被提供给音频编码器 12。根据预定压缩编码系统,例如,AC3(音频码数 3),音频编码器 12 压缩编码数字音频数据。用于音频数据的压缩编码系统不限于 AC3。作为替代,预期使用基带音频数据而不压缩编码音频数据。已经压缩编码的数字音频数据被提供给多路复用器 13。

[0276] 多路复用器 13 根据预定系统多路复用已经压缩编码和提供的数字视频数据和数字音频数据,并且作为一个数据流而输出多路复用的数据。在基于 MPEG2 系统而执行多路复用的该示例中,使用 MPEG2 传输流在时分的基础上多路复用所提供的压缩视频数据和音频数据。例如,多路复用器 13 具有缓冲存储器。多路复用器 13 在缓冲存储器中存储已经提供的压缩视频数据和音频数据。

[0277] 存储在缓冲存储器中的压缩视频数据是通过将它们分成预定大小并且给分成的每个部分添加标头(header)而分组化的 PES(分组化基本流)。类似地,压缩音频数据是通过将它们分成预定大小并且给分成的每个部分添加标头而被 PES 分组化。标头存储 MPEG2 系统中规定的预定信息,诸如表示存储在分组中的数据的再现时间的 PTS 和表示解码时间的 DTS(解码时戳)。PES 分组被进一步划分并存储在传输分组(TS 分组)的有效荷载中。TS 分组的标头存储标识在有效荷载中存储的数据的 PID(分组标识)。从多路复用器 13 输出的 TS 分组暂时存储在流缓冲器 14 中。

[0278] 具体地,由多路复用器 13 将具有预定大小的标头添加到 TS 分组。输出具有标头的 TS 分组。将预定标头添加到 TS 分组的分组被称为源分组。

[0279] 记录控制部分 15 控制数据到记录介质 20 的记录。对于记录介质 20,例如,可以使用可记录型 DVD(数字多用盘)。作为替代,对于记录介质 20,可以使用硬盘驱动器。作为替代,半导体存储器可以应用于记录介质 20。实现比其它记录介质更大容量的 Blu-ray Disc(注册商标)预期可应用于记录介质 20。

[0280] 记录控制部分 15 监视存储在流缓冲器 14 中的数据量。当存储在流缓冲器 14 中的数据超过预定量时,从流缓冲器 14 中读取用于记录介质 20 的记录单元量的数据并且将其写入到记录介质 20 中。

[0281] 例如,管理信息处理部分 16 由 CPU、作为工作存储器的 RAM、以及预存程序和预定数据的 ROM(这些部分未示出)组成。另外,例如,管理信息处理部分 16 可通过控制部分 30 的程序处理来实现其功能。在这种情况下,例如,控制部分 30 的 RAM 用作非易失性存储器

17,而非易失性存储器 18 连接到控制部分 30。

[0282] 管理信息处理部分 16 基于记录数据而使用非易失性存储器 17 作为工作存储器,并创建将要在上述索引文件“index.bdmv”、电影对象文件“MovieObject.bdmv”、播放列表文件“xxxxx.mpls”和剪辑信息文件“zzzzz.dpi”中存储的信息。按预定定时将所创建的信息写入到记录介质 20。

[0283] 例如,管理信息处理部分 16 获取多路复用器 13 中的记录数据的时间信息和记录控制部分 15 中的用于记录介质 20 的记录数据的地址信息,并且基于所获取的这些时间信息和地址信息而创建 EP_map 信息。另外,基于根据 UI 部分 31 的记录开始操作或者记录停止操作从控制部分 30 输出的控制信号和有关从多路复用器 13a 和记录控制部分 15 提供的记录数据的信息,而创建剪辑信息文件“zzzzz.dpi”。有关创建剪辑信息文件的信息被附加写入到播放列表文件中。用这样的方式,更新播放列表文件“xxxxx.mpls”。

[0284] 当更新播放列表文件时,确定对附加写入有关剪辑信息文件的信息的播放列表文件的预定约束是否满足。当确定结果指示不满足约束时,新创建播放列表文件。有关创建剪辑信息文件的信息被记录到新创建的播放列表文件中。

[0285] 另外,当将数据新记录到记录介质 20 中时,创建或更新索引文件“index.bdmv”和电影对象文件“MovieObject.bdmv”。

[0286] 图 42 示出可应用于本发明的实施例的数据结构的示例。索引文件“index.bdmv”具有一个或者多个标题。电影对象文件“MovieObject.bdmv”包含与索引文件“index.bdmv”具有的标题相对应的一个或者多个电影对象。每个电影对象调用一个播放列表文件“xxxxx.mpls”。播放列表文件“xxxxx.mpls”包含一个或者多个播放项目。每个播放项目参照剪辑信息文件“zzzzz.dpi”。剪辑信息文件“zzzzz.clpi”和作为剪辑的实体的剪辑 AV 流文件“zzzzz.m2ts”具有一对一的关系。

[0287] 在这样的结构中,用户可以以索引文件“index.bdmv”具有的标题为单元来看到记录介质上记录的剪辑。当用户选择他或她希望的标题时,从电影对象文件“MovieObject.bdmv”参照对应于该标题的电影对象。调用在所参照的电影对象中描述的播放列表文件“xxxxx.mpls”,根据包含在播放列表文件内的播放项目,参照剪辑信息文件“zzzzz.clpi”,并再现相应的剪辑 AV 流文件“zzzzz.m2ts”。

[0288] 当将表示时间信息的标记(播放列表标记)添加到播放列表文件“xxxxx.mpls”中时,可设置跳转位置。章用标记来定义。章是用户可看到的标题的再现单元。标记总是放置在记录开始位置。标记可以放置在除了记录开始位置之外的位置。

[0289] 换言之,当在播放列表文件“xxxxx.mpls”中设置用于记录开始操作的播放列表标记并登记参照剪辑的播放项目时,在播放列表文件中形成章。换句话说,当将播放列表标记记录到播放列表文件并记录播放项目时,可以说是将章记录到播放列表文件。

[0290] 如上所述,与剪辑一起创建真实播放列表。在图 42 中示出的示例中,播放列表文件“00000.mpls”、“00200.mpls”和“00018.mpls”具有真实播放列表的属性。

[0291] 在它们之中,播放列表文件“00000.mpls”是附加写入有关新创建的剪辑的信息的播放列表的示例。例如,参照与新记录的剪辑 AV 流文件“00125.m2ts”相对应的剪辑信息文件“00125.m2ts”的播放项目 #_1 被附加记录到播放列表文件“00000.mpls”,所述播放列表文件“00000.mpls”已经存储了参照与剪辑 AV 流文件“00001.m2ts”相对应的剪辑信息文

件“00001. dpi”的播放项目 #_0。在播放项目表示的开始时间处放置标记。当再现播放列表文件“00000. mpls”时,基于播放项目 #_0 而再现剪辑 AV 流文件“00001. m2ts”。其后,基于播放项目 #_1 而再现剪辑 AV 流文件“00125. m2ts”。

[0292] 播放列表文件“00200. mpls”是为一个剪辑创建一个播放列表文件以及该播放列表文件仅仅包含一个播放项目的示例。

[0293] 另外,播放列表文件“00018. mpls”是多个播放项目参照一个剪辑的示例。可预期,当执行记录开始操作或者记录停止操作时,执行控制以便创建播放项目并将数据附加写入到一个剪辑中。在播放项目 #_0 的开始处放置标记。通过连续地再现播放项目 #_0 和播放项目 #_1,再现整个剪辑 AV 流文件“00002. m2ts”。

[0294] 另一方面,如图 42 中示出的播放列表文件“00005. mpls”所示,虚拟播放列表指定现有剪辑的再现区域。在该示例中,在播放列表文件“00005. mpls”内包含的播放项目 #_0 参照剪辑信息文件“00028. dpi”并指定区域。播放项目 #_1 参照剪辑信息文件“00002. dpi”并指定区域。在播放项目 #_0 和播放项目 #_1 的开始处放置标记。当再现播放列表文件“00005. mpls”时,基于播放项目 #_0 而再现剪辑 AV 流文件“00028. m2ts”的指定区域。其后,基于播放项目 #_1 而再现剪辑 AV 流文件“00002. m2ts”。

[0295] 根据本发明,如上所述,确定基于新记录的剪辑的章是基于播放列表文件的约束而被附加写入到现有播放列表文件中还是被记录到新创建的播放列表文件中。播放列表文件的约束示例如下。播放列表文件的约束可预期为包括对格式的约束、对安装规格的约束、以及对产品规格的约束。

[0296] 对格式的约束可预期为以下约束 (1) 至约束 (7)。

[0297] 约束 (1):在一个播放列表文件中能够存在的播放项目数的上限。

[0298] 约束 (2):在一个播放列表文件中能够存在的播放列表标记数的上限。

[0299] 约束 (3):当一个播放列表文件参照多个剪辑时,它们的预定视频属性需要相互匹配。

[0300] 约束 (4):一个播放列表文件的文件大小的上限。

[0301] 约束 (5):与一个播放列表文件相关联的剪辑信息文件的总文件大小的上限。

[0302] 约束 (6):在与一个播放列表文件相关联的剪辑信息文件中用于粗单元搜索的入口点总数的上限。

[0303] 约束 (7):在与一个播放列表文件相关联的剪辑信息文件中用于细单元搜索的入口点总数的上限。

[0304] 其中,关于约束 (1) 的播放项目数和约束 (2) 的播放列表标记数,对于格式,存在于一个播放列表中的播放项目数的上限约束为例如 999,而存在于一个播放列表文件中的播放列表标记数的上限约束为例如 999。因此,当基于新记录的剪辑的章被附加写入到播放列表文件中时,需要确定是否满足这些约束。

[0305] 关于约束 (3) 的视频属性,格式规定在一个播放列表文件参照的多个剪辑 AV 流文件中,它们的有关视频数据编码的属性,诸如一个播放列表文件参照的多个剪辑 AV 流文件的帧大小、扫描系统、纵横比、帧速率和编解码器类型,需要相互匹配。因此,当新记录剪辑时,需要将从在附加写入了基于剪辑的章的播放列表文件中已经记录的播放项目参照的剪辑的属性与新记录的剪辑的属性进行比较。

[0306] 关于约束 (4) 的一个播放列表的文件大小和约束 (5) 的与一个播放列表文件相关联的剪辑信息文件的总文件大小,规定它们的上限。文件大小的规定与记录设备和再现设备的存储器容量相关联。

[0307] 换句话说,当执行记录操作时,与已经记录的剪辑相对应的播放列表和已经记录的剪辑的剪辑信息文件被暂时存储在记录设备的存储器中。当执行记录操作时,对存储器上的播放列表文件和剪辑信息文件执行更新处理。存储在存储器中的这些播放列表文件和剪辑信息文件按预定定时写入到记录介质中。使用有关播放列表文件和与一个播放列表文件相关联的剪辑信息文件的总文件大小的约束,能够防止在记录操作期间存储器已满、从而强迫其停止的情形。

[0308] 播放列表文件的文件大小的上限是例如 600kB(千字节)。当一个剪辑信息文件的文件大小的上限是例如 1MB(兆字节)时,与一个播放列表文件相关联的剪辑信息文件的总文件大小的上限是例如 2MB。

[0309] 约束 (6) 和约束 (7) 的上限与上述剪辑信息文件的文件大小的上限相关联。换句话说,如上所述,在剪辑信息文件中,粗单元搜索的入口点和细单元搜索的入口点是存储在块 blkCPI() 的块 blkEMap() 中的信息。换句话说,粗单元搜索的入口点是块 blkEMap() 的字段 PTSEPCoarse 和字段 SPNEPCoarse。细单元搜索的入口点是块 blkEMap() 的字段 PTSEPFine 和字段 SPNEPFine。

[0310] 在与一个播放列表文件相关联的剪辑信息文件中,对于格式,规定粗单元搜索的入口点总数的上限和细单元搜索的入口点总数的上限。因此,当剪辑新记录时,需要确定在附加写入参照剪辑的播放项目的播放列表文件中,这些约束是否满足。

[0311] 对安装规格的约束可预期为具有以下项目。

[0312] 约束 (8):关于播放列表文件的扩展数据块中的块 blkMakersPrivateData(),当不能接管 (take over) 由另一设备记录的播放列表文件的块 blkMakersPrivateData() 的信息时,禁止更新播放列表文件。

[0313] 例如,如图 35 中示出的块 DataBlock() 所示,没有约束播放列表文件的扩展数据块的块 blkMakersPrivateData() 的数据大小。因此,可预期,根据设备的规格而写入具有相对较大的数据大小的数据,例如,多个 100kB。由于存储容量的约束,可能难以使用由另一设备记录的具有大尺寸的块 blkMakersPrivateData() 的接管信息来更新或者编辑播放列表文件。

[0314] 对产品规格的约束可预期为以下项目。

[0315] 约束 (9):即使一个设备的标题和章的概念与另一设备不同,必须防止用户混淆它们。

[0316] 约束 (10):必须能够连续地执行捕捉操作和记录操作一段规定时期或更长时期。

[0317] 关于约束 (9) 的标题和章,可预期,取决于设备,播放列表和章具有不同的结构。因此,当已经由一个设备在其上记录了剪辑的记录介质由另一设备再现时,存在标题可能异常显示的可能性。

[0318] 例如,假定记录电视广播的固定式记录 and 再现设备(如视频传动机(deck))具有以下规格。一个节目与一个播放列表相对应。每隔 5 分钟自动放置作为章的播放列表标记。每当摄像机设备捕捉一个图像,即设备执行记录开始操作时,放置作为章的播放列表标记。

将多个章登记到一个播放列表中。另外,假定固定式记录和再现设备具有以下规格。仅仅显示播放列表的开始的缩略图图像。因此,不显示每一章的缩略图图像。

[0319] 在这些条件下,假定将已经记录了由固定式记录和再现设备所创建的播放列表文件的记录介质加载到摄像机设备中,以及参照由摄像机设备记录的剪辑的播放项目被附加写入和记录到由固定式记录和再现设备创建的播放列表文件中。在这种情况下,作为由固定式记录和再现设备记录的节目的稍后场景 (later scene),将由摄像机设备捕捉的画面登记为章。当摄像机设备的记录介质再次加载到固定式记录和再现设备从而再现节目时,由摄像机设备捕捉的画面的章不显示为缩略图图像。作为替代,不自然地将章再现为由固定式记录和再现设备记录的节目的最后场景。

[0320] 关于约束 (10) 的连续捕捉和记录一段规定时期或者更长时期,再现设备的规格通常确保连续记录一段预定时期或者更长时期。另一方面,如上所述,格式规定在与一个播放列表相关联的剪辑信息文件中入口点总数的上限。即使记录介质具有充足的空闲空间,由于参照剪辑信息文件的入口点总数和在格式中规定的上限,在参照将要新记录的剪辑的播放项目播放列表文件中将要确保一段预定时期的入口点不能得到确保。因此,当新记录剪辑时,需要确定是否可基于与附加写入了参照剪辑的播放项目的播放列表文件相关联的入口点数而执行连续记录一段预定时期或者更长的时期。

[0321] 接下来,将描述根据本发明的实施例基于将要新记录的剪辑的章是否可附加写入到播放列表文件中的确定处理。逐一执行上述约束 (1) 至约束 (10) 的确定。当确定结果指示这些约束的至少一个不能满足时,新创建播放列表,并将基于新记录的剪辑的章记录到新创建的播放列表中。

[0322] 图 43 是示出确定基于新记录的剪辑的章是否可附加写入到播放列表中的处理示例的流程图。首先,参考图 43 中示出的流程图,将简单描述整体处理的流程。在该示例中,假定记录介质 20 是诸如可记录型 DVD 的盘形记录介质 (以下,称为盘)。在参考图 41 描述的记录设备中,由控制部分 30 执行随后的每个确定处理等。

[0323] 当将记录介质 20 加载到记录设备中时,控制部分 30 控制记录控制部分 15 从记录介质 20 中读取索引文件 “index.bdmv” (在步骤 S100)。例如,通过管理信息处理部分 16 将读取的索引文件 “index.bdmv” 存储在非易失性存储器 17 中。

[0324] 在下一步骤 S101,控制部分 30 基于存储在非易失性存储器 17 中的索引文件 “index.bdmv” 中描述的信息而标识附加写入基于接下来将要记录的剪辑的章的播放列表 (播放列表文件) 的候选者 (candidate)。

[0325] 例如,关于块 blkIndexExtensionData() 中的块 blkTableOfPlayLists(), 其中块 blkIndexExtensionData() 是索引文件 “index.bdmv” 的扩展数据块,搜索块 blkTableOfPlayLists(), 以找到已经记录的最近真实播放列表,从而获取它的文件名。

[0326] 更具体地,参考图 29,搜索在块 blkIndexExtensionData() 中列出的块 blkMovieTitlePlayListPair(), 以找到其具有分配了属性 “Real” 的字段 PlayListattribute 且被描述为块 blkIndexExtensionData() 中的最后块 blkMovieTitlePlayListPair() 的块 blkMovieTitlePlayListPair(), 从而从检索到的块 blkMovieTitlePlayListPair() 中获取字段 PlayListFileName 的数据。

[0327] 在下一步骤 S102,从记录介质 20 中读取作为在步骤 S101 中标识的附加可写候选

者的播放列表文件并将其存储到非易失性存储器 17。其后,从记录介质 20 中读取与作为已读取的附加可写候选者的播放列表文件相关联的所有剪辑信息文件。将已经读取的剪辑信息文件存储在非易失性存储器 17 中。

[0328] 更具体地,参考图 10 至图 12,在作为附加可写候选者的播放列表文件中,参照块 blkPlayList() 中的所有的块 blkPlayItem()。从记录介质 20 中读取在获取的字段 ClipInformationFileName 的数据中表示的所有剪辑信息文件。

[0329] 在以下步骤 S104 至步骤 S111,根据上述约束确定是否能够基于作为附加可写候选者的播放列表文件以及存储在非易失性存储器 17 中的剪辑信息文件而附加写入章。换句话说,在步骤 S104 中,根据上述约束 (1),基于包含在步骤 S101 中所获取的作为附加可写候选者的播放列表文件内的播放项目数而执行附加写入能 / 不能 (ablitiy/inability) 确定。在下一步骤 S105 中,根据上述约束 (2),基于包含在作为附加可写候选者的播放列表文件内的播放列表标记数而执行附加写入能 / 不能确定。在下一步骤 S106 中,根据上述约束 (3),基于在作为附加可写候选者的播放列表文件中描述的视频属性和将要新记录的剪辑的视频属性而执行附加写入能 / 不能确定。

[0330] 在下一步骤 S107 中,根据上述约束 (4),基于附加可写候选者的播放列表文件的文件大小而执行附加写入能 / 不能确定。在下一步骤 S108 中,根据上述约束 (5),基从作为附加可写候选者的播放列表文件于参照的剪辑信息文件的总文件大小而执行附加写入能 / 不能确定。在下一步骤 S109 中,根据上述约束 (6) 和 (7),基于存储在从作为附加可写候选者的播放列表文件参照的剪辑信息文件中的入口点总数而执行附加写入能 / 不能确定。

[0331] 在下一步骤 S110 中,根据上述约束 (8),基于在作为附加可写候选者的播放列表文件中有或者没有另一设备的唯一扩展数据而执行附加写入能 / 不能确定。在下一步骤 S111 中,根据上述约束 (9),基于作为附加可写候选者的播放列表文件的最近更新设备而执行附加写入能 / 不能确定。

[0332] 在下一步骤 S112 中,基于上述步骤 S104 至步骤 S111 中的每个确定处理的确定结果,最终确定是否能够将基于新记录的剪辑的章附加写入到作为附加可写候选者的播放列表文件中。例如,将步骤 S104 至步骤 S111 的每个处理的确定结果都存储在控制部分 30 的寄存器等中。在步骤 S112 中,基于存储在寄存器中的每个处理的确定结果而执行确定。

[0333] 换句话说,当步骤 S104 至步骤 S111 的每个步骤的每个处理的确定结果都表示附加可写时,处理流程前进到步骤 S112。在步骤 S112 中,确定是否将基于新记录的剪辑的章附加写入到在步骤 S101 中获取的作为附加可写候选者的播放列表文件中。

[0334] 相反,当步骤 S104 至步骤 S111 的附加写入能 / 不能确定的至少一个确定结果表示不可附加写入时,确定不能附加写入到基于步骤 S101 中获取的作为附加可写候选者的播放列表文件中。在此情况下,新创建真实播放列表的播放列表文件,并且将基于新记录的剪辑的章记录到新创建的播放列表文件中。

[0335] 步骤 S104 至步骤 S111 的处理顺序并不局限于上述顺序。换句话说,步骤 S104 至步骤 S111 的处理顺序可以用任何方式进行改变。作为替代,可以并行地执行步骤 S104 至步骤 S111 的处理。可以不执行步骤 S104 至步骤 S111 的所有处理。作为替代,可以有选择地执行一个或者多个这样的处理。

[0336] 接下来,将逐一地更详细地描述步骤 S104 至步骤 S111 的处理。图 44 示出在步骤

S104 中,根据约束 (1),基于包含在作为附加可写候选者的播放列表文件内的播放项目数的附加写入能 / 不能确定的处理示例。在该处理中,如步骤 S120 所示,对于作为附加可写候选者的播放列表文件的块 blkPlayList() (参考图 11),获取字段 NumberOfPlayItems 的值。将字段 NumberOfPlayItems 的获取值与已经为一个播放列表文件中可存在的播放项目数而设置的上限值 (例如值“999”)进行比较。

[0337] 当比较结果指示字段 NumberOfPlayItems 的值小于预定上限值时,确定基于新记录的剪辑的可将章附加写入到作为附加可写候选者的播放列表文件中。相反,当字段 NumberOfPlayItems 的值等于或者大于预定上限值时,确定不能将章附加写入到作为附加可写候选者的播放列表文件中。

[0338] 图 45 示出在步骤 S105 中,根据上述约束 (2),基于包含在作为附加可写候选者的播放列表标记内的播放列表标记数的附加写入能 / 不能确定的处理示例。在该处理中,如在步骤 S130 中所示,对于作为附加可写候选者的播放列表文件中的块 blkPlayListMark() (参考图 14) 的值,获取字段 NumberOfPlayListMarks 的值。将字段 NumberOfPlayListMarks 的获取值与已经为一个播放列表文件中可存在的播放列表标记数而设置的预定上限值 (例如值“999”)进行比较。

[0339] 当比较结果指示字段 NumberOfPlayListMarks 的值小于预定上限值时,确定基于新记录的剪辑的可将章附加写入到作为附加可写候选者的播放列表文件中。相反,当字段 NumberOfPlayListMarks 的值等于或者大于预定上限值时,确定不能将章附加写入到作为附加可写候选者的播放列表文件中。

[0340] 如上所述,存在两类播放列表标记,它们是入口标记和链接点。在步骤 S130 中,确定这些入口标记和链接点的总数是否小于上限值。

[0341] 图 46 中示出在步骤 S106 中,根据上述约束 (3),基于作为附加可写候选者的播放列表文件中描述的视频属性和新记录的剪辑的视频属性的附加写入能 / 不能确定的处理示例。首先,在第一步骤 S140 中,获取将要记录的视频数据的属性信息。在该实施例中,基于目前为记录设备所设置的捕捉模式或者记录模式,获取作为将要记录的视频数据的属性信息的帧大小、纵横比、帧速率和扫描类型 (隔行 / 逐行)。

[0342] 在以下步骤 S141 至步骤 S144 中,获取与作为附加可写候选者的播放列表文件相关联的剪辑的视频属性。在该实施例中,当多个播放项目存储在作为附加可写候选者的播放列表文件中时,从作为最旧的播放项目记录的播放项目参照的剪辑信息文件中获取视频属性。简单地说,作为视频属性,在步骤 S141 中,获取图像宽度;在步骤 S142 中,获取图像高度 (线数) 和扫描类型;在步骤 S143 中,获取纵横比;在步骤 S144 中,获取帧速率。

[0343] 更具体地,对于存储在作为附加可写候选者的播放列表文件中的块 blkPlayList() 的最上端的块 blkPlayItem() (参考图 12),获取作为播放项目参照的剪辑信息文件的文件名的字段 ClipInformationFileName 的数据。从在图 43 中示出的流程图的步骤 S103 的处理中所读取的并且与作为附加可写候选者的播放列表相关联的所有剪辑信息文件参照与获取的文件名相对应的剪辑信息文件。

[0344] 在步骤 S141 中,对于作为剪辑信息文件的扩展数据的块 blkClipExtensionData() 中的块 blkProgramInfoExt() (参考图 37) 和块 blkProgramInfoExt() 中的块 blkStreamCodingInfoExt(i, j) (参考图 38),从块

blkStreamCodingInfoExt(i, j) 中获取字段 HorizontalSize 的值。

[0345] 在步骤 S142 中,对于剪辑信息文件的块 blkProgramInfo() (参考图 17) 中的块 blkStreamCodingInfo() (参考图 18),获取字段 VideoFormat 的值。如参考图 19 所述,字段 VideoFormat 表示视频数据的线数和扫描系统(隔行/逐行)与用 4 位表示的值的组合。

[0346] 如在步骤 S142,在步骤 S143,对于块 blkStreamCodingInfo(),获取字段 AspectRatio 的值。如参考图 21 所述,字段 AspectRatio 通过用 4 位表示的值来表示视频数据的帧的纵横比。

[0347] 如在步骤 S143 和步骤 S143,在步骤 S144,对于块 blkStreamCodingInfo(),获取字段 FrameRate 值。如参考图 20 所述,字段 FrameRate 通过用 4 位表示的值来表示视频数据的帧速率。

[0348] 在步骤 S145 中,将所记录的已经在步骤 S140 中从记录设备中获取的视频数据的属性信息的每个条目,和与作为附加可写候选者的播放列表文件相关联的并在步骤 S141 至步骤 S144 中获取的剪辑的每个视频属性进行比较。

[0349] 当比较结果指示所记录的视频数据的属性信息的所有条目匹配于与作为附加可写候选者的播放列表文件相关联的剪辑的所有视频属性时,确定基于新记录的剪辑的章能够被附加写入到作为附加可写候选者的播放列表文件中。相反,即使记录的视频数据的每一属性信息不匹配于作为附加可写候选者的播放列表文件相关联的剪辑的每一视频属性,也确定不能将章附加写入到作为附加可写候选者的播放列表文件中。

[0350] 图 47 示出在步骤 S107 中,根据约束(4),基于作为附加可写候选者的播放列表文件的文件大小的附加写入能/不能确定的处理示例。在该附加写入能/不能确定处理中,确定附加写入了至少预定数目的章的、作为附加可写候选者的播放列表的文件大小是否超过一个播放列表的文件大小的上限。

[0351] 取决于记录设备的设计概念,预定数目的章可预期为一个章、多个章、相对于可在一章中存在的章的最大数目的剩余章数等等。在该示例中,考虑所述预定数目的章是相对于可在一个播放列表文件中存在的章的最大数目的剩余章数。

[0352] 首先,在步骤 S150 中,获取已经预先计算的播放列表文件每一章的大小增量 SIZE_1CHAP。换句话说,在该步骤 S150 中,计算组成一章的播放项目和播放列表标记的数据量。

[0353] 参考图 12,在播放项目中,字段 StillMode、字段 StillTime、块 blkUOMaskTable() 和块 blkSTNTable() 是数据长度可以变化的数据。具体地,这些字段和块的数据具有取决于记录模式或者记录设备的固定长度。参考图 14,播放列表标记的每个字段的值具有固定长度。因此,能够预先计算在记录一个剪辑时所创建的播放项目和播放列表标记的数据大小。另外,所计算的值在特性上并不取决于记录时间。例如,预先计算并在记录设备的 ROM 等中存储每一章的播放列表文件的大小增量 SIZE_1_CHAP 的值。作为替代,每当执行图 47 中示出的流程图的处理时,可计算大小增量 SIZE_1_CHAP 的值。

[0354] 当将章附加写入到播放列表中时,添加了播放项目和播放列表标记。因此,当将章附加写入到播放列表中时,需要根据约束(1)和约束(2)考虑可在一个播放列表文件中存在的播放项目数和播放列表标记数。

[0355] 在步骤 S151 中,计算能够附加写入到作为附加可写候选者中的播放列表文件的

播放项目的数目 PI_REMAIN。换句话说,对于作为附加可写候选者的播放列表文件中的块 blkPlayList() (参考图 11), 获取字段 NumberOfPlayItems 的值, 作为包含在作为附加可写候选者的播放列表内的播放项目的数目。通过从为一个播放项目中存在的播放项目的数目而设置的预定上限值 PI_MAX (例如, “999”) 中减去包含在作为附加可写候选者的播放列表内的播放项目的数目, 来获取能够附加写入到作为附加可写候选者的播放列表中的播放项目的数目 PI_REMAIN’。

[0356] 换句话说, 能够附加写入到作为附加可写候选者的播放列表文件中的播放项目的数目 PI_REMAIN 可通过以下公式 (1) 获得。

$$[0357] \quad PI_REMAIN = PI_MAX - NumberOfPlayItems \dots \quad (1)$$

[0358] 在步骤 S152, 计算能够附加写入到作为附加可写候选者的播放列表文件中的播放列表标记的数目 MARK_REMAIN。换句话说, 对于作为附加可写候选者的播放列表文件中的块 blkPlayListMark() (参考图 14), 获取字段 NumberOfPlayListMarks 的值, 作为包含在作为附加可写候选者的播放列表内的播放列表标记的数目。其后, 通过从为一个播放列表中存在的播放项目的数目设置的预定上限值 MARK_MAX (例如 “999”) 中减去包含在作为附加可写候选者的播放列表内的播放列表标记的数目, 来获取能够附加写入到作为附加可写候选者的播放列表中的播放列表标记的数目 MARK_REMAIN。

[0359] 换句话说, 能够附加写入到作为附加可写候选者的播放列表文件中的播放列表标记的数目 MARK_REMAIN 可从以下公式 (2) 中获得。

$$[0360] \quad MARK_REMAIN = MARK_MAX - NumberOfPlayListMarks \dots \quad (2)$$

[0361] 在下一步骤 S153, 在步骤 S151 中获取的能够附加写入到作为附加可写候选者的播放列表中的播放项目的数目 PI_REMAIN 和在步骤 S152 中获取的能够附加写入到作为附加可写候选者的播放列表文件中的播放列表标记的数目 MARK_REMAIN 中的较小者被视为能够附加写入到附加可写候选者中的章的数目 CHAP_REMAIN。

[0362] 换句话说, 假定 “MIN” 指示选取括号内的值的较小者, 通过上述公式 (1) 和公式 (2) 的结果, 从以下公式 (3) 中可获得能够附加写入到作为附加可写候选者的播放列表文件中的章的数目 CHPA_REMAIN。在公式 (3) 中, “MIN” 指示选取括号内的值的较小者。

$$[0363] \quad CHAP_REMAIN = MIN(PI_REMAIN, MARK_REMAIN) \dots \quad (3)$$

[0364] 在下一步骤 S154 中, 通过获取作为附加可写候选者的播放列表文件的文件大小, 来获取相对于根据约束 (4) 的一个播放列表文件的文件大小的上限 PL_SIZE_MAX 的剩余数据大小 SIZE_REMAIN。例如, 使用 OS 的文件系统提供的功能, 可获取播放列表文件的文件大小。数据大小 SIZE_REMAIN(kB) 可从以下公式 (4) 获得。在以下公式中, 通过 FILE_SIZE(kB) 指定作为附加可写候选者的播放列表文件的文件大小。

$$[0365] \quad SIZE_REMAIN(kB) = PL_SIZE_MAX(kB) - FILE_SIZE(kB) \dots \quad (4)$$

[0366] 在下一步骤 S155, 确定作为附加可写候选者的播放列表文件的文件大小是否超过一个播放列表文件的文件大小的上限, 其中将能够附加写入到作为附加可写候选者的播放列表文件中的用于章的播放项目和播放列表标记附加写入到所述播放列表文件中。

[0367] 换句话说, 确定在步骤 S150 中获取的每一章的播放列表文件的大小增量 SIZE_1CHAP 和公式 (3) 与公式 (4) 的结果是否满足以下公式 (5)。

$$[0368] \quad SIZE_1CHAP \times CHAP_REMAIN \geq SIZE_REMAIN \dots \quad (5)$$

[0369] 如果满足公式 (5), 则确定基于新记录的剪辑的章能够被附加写入到作为附加可写候选者的播放列表文件中。相反, 如果公式 (5) 不满足, 则确定不能将章附加写入到作为附加可写候选者的播放列表文件中。

[0370] 图 48 示出在步骤 S108 中, 根据上述约束 (5), 基于从作为附加可写候选者的播放列表文件参照的剪辑信息文件的总文件大小的附加写入能 / 不能确定的处理示例。

[0371] 根据上述约束 (5), 为从作为附加可写候选者的播放列表文件参照的剪辑信息文件的总文件大小设置上限值 CLIP_SUM_MAX, 例如 2MB。当在附加写入章时将剪辑信息文件添加到作为附加可写候选者的播放列表中时, 与作为附加可写候选者的播放列表文件相关联的剪辑信息文件的总大小需要不超过上限值 CLIP_SUM_MAX。

[0372] 在步骤 S160 中, 获取已经为一个剪辑信息文件计算的最大尺寸 SIZE_1CLIP。在剪辑信息文件中, 块 blkEPMAP() 存储将 PTS 值与剪辑 AV 流文件的字节地址相关联的入口点的信息 (参考图 23 至图 26)。该入口点的信息是随记录时间而变化的值。存储在剪辑信息文件中的其它信息不是取决于记录时间、而是取决于编码系统的固定值。另一方面, 如上所述, 确保连续记录的最小时间设置为记录设备的规格。确保连续记录的最小时间的入口点的数目取决于编码器的视频和音频的编码系统。通过计算用于确保连续记录的最小时间的入口点的总数据大小, 基于计算结果而获取最大尺寸 SIZE_1CLIP。

[0373] 在下一步骤 S161 中, 通过获取从作为附加可写候选者的播放列表文件参照的所有剪辑信息文件的文件大小, 计算总大小 SIZE_TOTAL_CLIP。

[0374] 换句话说, 对于作为附加可写候选者的播放列表文件中的所有块 blkPlayItem() (参考图 12), 获取每个块 blkPlayItem() 的字段 ClipInformationFileName 的数据。对于作为附加可写候选者的播放列表文件中的每个块 blkPlayItem(), 获取用字段 ClipInformationFileName 的获取数据表示的剪辑信息文件的文件大小。添加每个块 blkPlayItem() 的文件大小。例如, 使用 OS 的文件系统提供的功能, 可获取每个剪辑信息文件的文件大小。

[0375] 在下一步骤 S162 中, 根据以下公式 (6), 将从与一个播放列表文件相关联的剪辑信息文件的总文件大小的上限值中减去从在步骤 S161 中所计算的作为附加可写候选者的播放列表文件参照的所有剪辑信息文件的总大小 SIZE_TOTAL_CLIP 所得到的值, 与在步骤 S160 中所计算的一个剪辑信息文件的最大尺寸 SIZE_1CLIP 进行比较, 确定具有最大尺寸 SIZE_1CLIP 的剪辑信息文件是否能够被附加写入到作为附加可写候选者的播放列表文件中。

[0376] $CLIP_SUM_MAX - SIZE_TOTAL_CLIP \geq SIZE_1CLIP \dots (6)$

[0377] 当从与一个播放列表文件相关联的剪辑信息文件的总文件大小的上限值中减去从作为附加可写候选者的播放列表文件参照的所有剪辑信息文件的总大小 SIZE_TOTAL_CLIP 所得到的值与一个剪辑信息文件的最大尺寸 SIZE_1CLIP 进行比较的比较结果指示大于一个剪辑信息文件的最大尺寸 SIZE_1CLIP 时, 确定基于新记录的剪辑的章能够被附加写入到作为附加可写候选者的播放列表文件中。

[0378] 相反, 当从与一个播放列表文件相关联的剪辑信息文件的总文件大小的上限值中减去从作为附加可写候选者的播放列表文件参照的所有剪辑信息文件的总大小 SIZE_TOTAL_CLIP 所得到的值小于一个剪辑信息文件的最大尺寸 SIZE_1CLIP 时, 确定不能将章

附加写入到作为附加可写候选者的播放列表文件中。

[0379] 图 49 示出在上述步骤 S109 中,根据约束 (6) 和约束 (7),基于在从作为附加可写候选者的播放列表文件参照的剪辑信息文件中存储的入口点的总数的附加写入能 / 不能确定的处理示例。

[0380] 如上所述,对于从一个播放列表文件参照的剪辑信息文件,为存储在块 blkEPMaP() 中的入口点的总数设置上限。当在附加写入章时将剪辑信息文件附加写入到作为附加可写候选者的播放列表中时,在与作为附加可写候选者的播放列表文件相关联的剪辑信息文件中存储的入口点的总数需要不超过上限值。

[0381] 根据约束 (6) 和约束 (7),设置粗单元搜索的入口点的上限值和细单元搜索的入口点的上限值。因此,需要分别对粗单元搜索的入口点和细单元搜索的入口点执行确定。

[0382] 在与一个播放列表文件相关联的剪辑信息文件中用于粗单元搜索的入口点的总数值的上限值 MAX_EP_COARSE 是例如 24576。另一方面,用于细单元搜索的入口点的总数值的上限值 MAX_EP_FINE 是例如 180000。

[0383] 首先,在步骤 S170 中,获取预先计算的每一章的入口点的最大数。如上所述,通常作为记录设备的规格,设置确保连续记录的最小时间。预先计算确保连续记录的最小时间的入口点的最大数。在步骤 S170 中,获取该值。

[0384] 如参考图 25 和图 26 所述,每间隔 11.5 秒 (PTS : 条目 PTSEPCoarse) 或者 25MB (源分组数 : 条目 SPNEPCoarse) 放置粗单元搜索的入口点。当假定确保连续记录的最小时间由时间 MIN_TIME (hr) 指定以及在最小时间内的记录中创建的剪辑 AV 流文件的数据量由数据量 MIN_SIZE (MB) 指定时,从以下公式 (7) 获取用于粗单元搜索的入口点的、每一章的入口点的最大数 NEEDED_EP_COARSE。在稍后将描述的公式 (7) 和公式 (8) 中,“°C EIL”指示将小数点后的部分向上舍入 (round up) 到相邻整数。

$$[0385] \quad \text{NEEDED_EP_COARSE} = \text{CEIL}(3600[\text{SEC}] \times \text{MINJTIME}[\text{hr}] / 11.5[\text{sec}] + \text{CEIL}(\text{MIN_SIZE}[\text{MB}] / 25[\text{BM}])) \dots \quad (7)$$

[0386] 每隔具有 PTS (条目 PTSEPFine) 精度或者源分组 (条目 SPNEPFine) 精度的 GOP,放置细单元搜索的入口点。在该实施例中,对于细单元搜索的入口点,仅仅使用条目 PTSEPFine。当帧速率是 29.97Hz 以及 1 个 GOP 由 15 帧组成时,对于细单元搜索的入口点,从以下公式 (8) 中获取每一章的入口点最大数 NEEDED_EP_FINE。在公式 (8) 中,“90[kH]/13003”表示基于 PTS 精度的帧速率 (29.97Hz)。

$$[0387] \quad \text{NEEDED_EP_FINE} = \text{CEIL}(3600[\text{sec}] \times \text{MINJTIME}[\text{hr}] \times 90[\text{kHz}] / 3003 / 15[\text{Frames/GOP}])) \dots \quad (8)$$

[0388] 在下一步骤 S171 中,对于从作为附加可写候选者的播放列表文件参照的所有剪辑信息文件,获取粗单元搜索的入口点和细单元搜索的入口点。获取粗单元搜索的入口点总数 TOTAL_EP_COARSE 和细单元搜索的入口点总数 TOTAL_EP_FINE。

[0389] 更具体地,可基于剪辑信息文件中的字段 NumberOfStreamPIDEntries、字段 NumberOfEPCoarseEntries[k] 和字段 NumberOfEPFineEntries[k] 而获取粗单元搜索的入口点总数 TOTAL_EP_COARSE 和细单元搜索的入口点总数 TOTAL_EP_FINE。

[0390] 在下一步骤 S172 中,当将章附加写入到作为附加可写候选者的播放列表文件中时,确定粗单元搜索的入口点是否超过 MAX_EP_COARSE 的上限值。换句话说,根据以下公

式 (9), 基于在上述步骤 S170 中获取的每一章的粗单元搜索的入口点最大数 $NEEDED_EP_COARSE$ 和在步骤 S171 中获取的粗单元搜索的入口点总数 $TOTAL_EP_COARSE$ 而执行确定。

[0391] $MAX_EP_COARSE - TOTAL_EP_COARSE \geq NEEDED_EP_COARSE \dots (9)$

[0392] 如果每个值的关系不满足该公式 (9), 即每一章的粗单元搜索的入口点的最大数 $NEEDED_EP_COARSE$ 大于与一个播放列表文件相关联的剪辑信息文件的粗单元搜索的入口点总数值的上限值 MAX_EP_COARSE 和与作为附加可写候选者的播放列表文件相关联的剪辑信息文件的粗单元搜索的入口点总数 $TOTAL_EP_COARSE$ 两者之间的差值, 则确定不能将章附加写入到作为附加可写候选者的播放列表文件中。

[0393] 相反, 当确定结果指示每个值的关系满足公式 (9) 时, 处理流程前进到步骤 S173。在步骤 S173 中, 对于细单元搜索的入口点, 执行相同的确定。换句话说, 根据以下公式 (10), 基于在上述步骤 S170 中获取的每一章的细单元搜索的入口点的最大数 $NEEDED_EP_FINE$ 和在步骤 S171 中获取的细单元搜索的入口点总数 $TOTAL_EP_FINE$, 对粗单元搜索 MAX_EP_FINE 的入口点的上限值执行确定。

[0394] $MAX_EP_FINE - TOTAL_EP_FINE \geq NEEDED_EP_FINE \dots (10)$

[0395] 如果确定结果指示每个值的关系不满足该公式 (10), 则确定不能将章附加写入到作为附加可写候选者的播放列表文件中。相反, 当每个值的关系满足该公式 (10) 时, 确定能够将章附加写入到作为附加可写候选者的播放列表文件中。

[0396] 图 50 示出在步骤 S110 中, 根据约束 (8), 基于在作为附加可写候选者的播放列表文件中有 / 没有另一设备的唯一扩展数据的附加写入能 / 不能确定的处理示例。

[0397] 首先, 在第一步骤 S180 中, 在作为附加可写候选者的播放列表文件的扩展数据中搜索基于 AVCDH 格式的扩展数据。

[0398] 换句话说, 参考图 10, 获取作为附加可写候选者的播放列表文件的字段 $ExtensionDataStartAddress$ 的值, 并确定所获取的值是否为“0”(未示出)。当值不是“0”时, 因为在播放列表文件中存在扩展数据的块 $blkExtensionData()$, 所以基于字段 $ExtensionDataStartAddress$ 的值而参照块 $blkExtensionData()$ 。参考图 33, 确定在播放列表文件中的扩展数据的块 $blkPlayListExtensionData()$ 的字段 $ExtDataType$ 和字段 $ExtDataVersion$ 的值是否是以 AVCHD 格式规定的值。

[0399] 在下一步骤 S181 中, 基于步骤 S180 中的搜索结果而确定在作为附加可写候选者的播放列表文件所包含的扩展数据中是否存在基于 AVCHD 格式的扩展数据。

[0400] 换句话说, 如果基于步骤 S180 中的搜索结果, 在作为附加可写候选者的播放列表文件中字段 $ExtensionDataStartAddress$ 的值是“0”, 或者在扩展数据的块 $blkPlayListExtensionData()$ 中字段 $ExtDatatype$ 和字段 $ExtDataVersion$ 的值不是以 AVCHD 格式定义的值, 则确定在作为附加可写候选者的播放列表文件的扩展数据中不存在以 AVCHD 格式定义的扩展数据。在这种情况下, 确定不能将章附加写入到作为附加可写候选者的播放列表文件中。

[0401] 相反, 当在步骤 S181 中的确定结果指示在作为附加可写候选者的播放列表文件的扩展数据中存在以 AVCHD 格式定义的扩展数据时, 处理流程前进到步骤 S182。在步骤 S182 中, 确定在作为附加可写候选者的播放列表文件的扩展数据中是否存在除了基于 AVCDH 格式的扩展数据之外的其它扩展数据。如果确定结果指示它们存在, 则确定不能将章

附加写入到作为附加可写候选者的播放列表文件中。

[0402] 当步骤 S182 中的确定结果指示在作为附加可写候选者的播放列表文件的扩展数据中除了基于 AVCHD 格式的扩展数据之外不存在其它扩展数据时,处理流程前进到步骤 S183。在步骤 S183 中,对于作为附加可写候选者的播放列表文件的扩展数据的块 blkPlaylistExtensionData() 的块 blkMakersPrivateData(),搜索该设备的数据。换句话说,参考图 35,在块 blkMakersPrivateData() 中的字段 MakerID 和字段 MakerModelCode 中搜索表示该设备的数据。

[0403] 在下一步骤 S184 中,基于步骤 S183 中的搜索结果而执行确定。在步骤 S184 中,如果基于步骤 S183 中的搜索结果,确定结果指示不存在该设备的扩展数据,则确定不能将章附加写入到作为附加可写候选者的播放列表文件中。相反,当基于步骤 S183 中的搜索结果,确定结果指示存在该设备的扩展数据时,处理流程前进到步骤 S185。

[0404] 在步骤 S185 中,基于步骤 S183 中的搜索结果,确定在块 blkMakersPrivateData() 中是否存在除了该设备之外的其它设备扩展数据。换句话说,当在块 blkMakersPrivateData() 的字段 MakerID 和字段 MakerModelCode 中存在除了表示该设备的数据之外的数据时,确定在块 blkMakersPrivateData() 中存在除了该设备之外的其它设备的扩展数据。

[0405] 当在步骤 S185 中的确定结果指示在块 blkMakersPrivateData() 中存在除了该设备之外的其它设备的扩展数据时,确定不能将章附加写入到作为附加可写候选者的播放列表文件中。相反,当确定结果指示在块 blkMakersPrivateData() 中只存在该设备的扩展数据时,确定能够将章附加写入到作为附加可写候选者的播放列表文件中。

[0406] 在上述步骤 S181 中,当在作为附加可写候选者的播放列表文件中不存在基于 AVCHD 格式的扩展数据时,确定不能将章附加写入到作为附加可写候选者播放列表文件中。然而,在该步骤中执行的确定并不局限于该示例。换句话说,根据记录设备的规格,当在作为附加可写候选者的播放列表文件中不存在块 blkMakersPrivateData 时,可以预期能够将章附加写入到作为附加可写候选者的播放列表文件中。

[0407] 图 51 示出在上述步骤 S111 中,根据约束 (9),基于作为附加可写候选者的播放列表文件的最近更新设备的附加写入能 / 不能确定的处理示例。当作为附加可写候选者的播放列表文件的最近更新设备不是该本地 (local) 设备时,作为附加可写候选者的播放列表文件的标题和章的概念与该设备不同。因此,当数据再现时存在出现问题的可能性。当执行控制使得基于作为另外可记录的候选者的播放列表文件的最近更新设备的信息而将章附加写入到作为附加可写候选者的播放列表文件中时,只有在最近更新的设备是该设备的情况下,由于在一个播放列表文件中标题和章的概念的差别而出现的故障才能够得到解决。

[0408] 首先,在第一步骤 S190 中,在作为附加可写候选者的播放列表文件的扩展数据中搜索基于 AVCHD 格式的扩展数据。在下一步骤 S191 中,基于步骤 S190 中的搜索结果而确定在作为附加可写候选者的播放列表文件的扩展数据中是否存在基于 AVCHD 格式的扩展数据。因为在步骤 S190 和步骤 S191 中的处理与参考图 50 描述的步骤 S180 和步骤 S181 中的处理相同,所以为了防止复杂化,将省略对它们的详细描述。

[0409] 当在步骤 S191 中的确定结果指示在作为附加可写候选者的播放列表文件的扩展

数据中没有基于 AVCHD 格式的扩展数据时,确定不能将章附加写入到作为附加可写候选者的播放列表文件中。

[0410] 相反,当基于步骤 S190 中的搜索结果,在步骤 S191 中的确定结果指示在作为附加可写候选者的播放列表文件的扩展数据中存在基于 AVCHD 格式的扩展数据时,处理流程前进到步骤 S192。在步骤 S192 中,检查作为附加可写候选者的播放列表文件的最近更新设备。换句话说,对于扩展数据的块 blkPlayListExtensionData() 的块 blkPlayListMeta(), 获取字段 MakerID 和字段 MakerModelCode 的数据。

[0411] 当在步骤 S192 中已经检查了作为附加可写候选者的播放列表文件的最近更新设备时,处理流程前进到步骤 S193。在步骤 S193,确定检查的最近更新设备是否是该设备。换句话说,当扩展数据的块 blkPlayListExtensionData() 的块 blkPlayListMeta() 中的字段 MakerID 和字段 MakerModelCode 的数据表示该设备时,能够确定最近更新设备是该设备。

[0412] 当在步骤 S193 中的确定结果指示作为附加可写候选者的播放列表文件的最近更新设备是该设备时,确定能够将章附加写入到作为附加可写候选者的播放列表文件中。相反,当确定结果指示作为附加可写候选者的播放列表文件的最近更新设备不是该设备时,确定不能将章附加写入作为附加可写候选者的播放列表文件中。

[0413] 还可将有关最近更新设备的信息存储在索引文件中。可预期,例如,将有关最近更新设备的信息存储在索引文件的扩展数据的块 blkMakersPrivateData() 中的块 blkMakersPrivateData() 中。在这种情况下,使用在索引文件中存储的有关最近更新设备的信息来执行图 51 中示出的流程图的确定。

[0414] 至于根据上述约束 (10) 的连续捕捉和记录,它们被包括在参考图 43(步骤 S108) 和图 48 描述的关于剪辑信息文件的文件大小的处理和参考图 43(步骤 S109) 和图 49 描述的关于剪辑信息文件中的入口点的处理中,对它们的详细描述将省略。

[0415] 在参考图 43 进行的以上描述中,当将记录介质 20 加载到记录设备中时,执行将章附加写入到作为附加可写候选者的播放列表文件中还是新创建的播放列表文件的确定的序列。然而,可不按这样的方式执行这些确定。作为替代,每当附加写入章,例如,执行记录开始操作时,可执行这些确定。

[0416] 接下来,将描述作为参考上述图 43 至图 51 描述的处理结果、将章附加写入到作为附加可写候选者的播放列表文件中的处理。图 52 是示出将章附加写入到播放列表文件中的处理示例的流程图。

[0417] 当在步骤 S200 中执行记录开始操作时,处理流程前进到步骤 S201。在步骤 S201 中,开始将剪辑 AV 流记录到记录介质 20。例如,当操作位于 UI 部分 31 中的且命令记录开始的记录开始开关时,命令记录开始的控制信号从 UI 部分 31 提供给控制部分 30。控制部分 30 控制记录部分 10 的每个部分,以基于命令记录开始的控制信号而将从终端 40 输入的基带视频数据和从终端 41 输入的基带音频数据记录到记录介质 20 中。

[0418] 当控制记录的开始时,将剪辑 AV 流记录到记录介质 20 中 (在步骤 S201)。换句话说,输入视频数据和音频数据分别由视频编码器 11 和音频编码器 12 进行压缩编码、由多路复用器 13 分组化为 TS 分组 (具体地,添加预定标头到其上的源分组)、并且被提供给流缓冲器 14。当已经在流缓冲器 14 中存储了超过预定量的 TS 分组时,记录控制部分 15 从流缓冲器 14 中读取 TS 分组。将已读取的 TS 分组存储在已经分配了预定文件名并被记录到记

录介质 20 中的剪辑 AV 流文件中。

[0419] 例如,当已经将具有文件名“00001.m2ts”的剪辑 AV 流文件记录在记录介质 20 上时,选取与已经记录的剪辑 AV 流文件的文件名不重复的文件名,例如,“00002.m2ts”。

[0420] 当将剪辑 AV 流记录到记录介质 20 中时,由管理信息处理部分 16 实时地创建表示所记录的数据的再现时间和地址之间的关系的信息。作为存储在上述剪辑信息文件“zzzzz.clpi”的块 blkEPMa() 中的数据而存储该数据。

[0421] 在下一步骤 S202 中,确定是否已经执行记录停止操作。当用户已经操作位于 UI 部分 31 中的记录停止开关以及确定结果指示已经停止记录时,处理流程前进到步骤 S203。相反,当记录没有停止时,处理流程回到步骤 S201。在步骤 S201 中,继续将剪辑 AV 流记录到记录介质 20 中。

[0422] 在步骤 S203 中,当记录已经停止时,将存储在流缓冲器 14 中的所有流写入到记录介质 20 中。例如,根据从控制部分 30 接收的记录停止命令,记录控制部分 15 读取存储在流缓冲器 14 中的所有流(TS 分组),并且将它们写入到记录介质 20 中。

[0423] 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 的操作。此时,为了执行参考图 13A 描述的第一无缝连接,执行控制,例如,使得在停止视频编码器 11 的操作之后的预定时间,停止音频编码器 12 的操作。

[0424] 在下一步骤 S204 至步骤 S208 中,创建关于写入到记录介质 20 中的剪辑 AV 流文件的剪辑信息文件,以及由管理信息处理部分 16 更新作为附加可写候选者的播放列表文件。

[0425] 首先,在步骤 S204 中,由管理信息处理部分 16 创建剪辑信息文件“zzzzz.clpi”。例如,剪辑信息文件的文件名与该剪辑信息文件所表示的剪辑 AV 流文件的文件名相对应。当剪辑 AV 流文件的文件名是“00002.m2ts”时,该剪辑信息文件的文件名是“00002.dpi”,其中扩展名之前的部分与剪辑 AV 流文件的文件名的扩展名之前的部分相同。

[0426] 根据图 15 至图 21 所示的每个语法,设置字段和标志的值,并且以预定方式将它们存储在剪辑信息文件“00002.dpi”中。例如,当正在记录剪辑时,基于由管理信息处理部分 16 从多路复用器 13 中获取的信息而创建关于 TS 分组的信息和关于再现时间(PTS)的信息。另外,当正在记录剪辑时,基于由管理信息处理部分 16 从记录控制部分 15 中获取的信息而创建关于记录介质 20 上的记录地址的信息。对于系统唯一的值基于在例如 ROM(未示出)中存储的信息。另外,表示再现时间和地址之间的关系的上述块 blkEPMa() 的信息被存储在剪辑信息文件“00002.dpi”的块 blkCPI() 中。

[0427] 当由用户操作停止记录剪辑时,块 blkClipInfo() 中的标记 IsCC5 的值设置为“1”(二进制值)。因此,以预定方式设置由块 blkClipInfo() 中的 if 语句(参考图 16)表示的数据。

[0428] 在已经创建剪辑信息文件之后,处理流程前进到下一步骤 S205。从步骤 S205 至步骤 S208 的处理是对播放列表文件的处理。在从步骤 S205 至步骤 S208 的处理中,将与新记录的剪辑 AV 流文件“00002.m2ts”相对应的播放项目添加到存在于记录介质 20 上的播放列表文件中。

[0429] 首先,在步骤 S205 中,将播放列表文件中的块 blkPlayItem() 中的字段 ConnectionCondition 的值设置为“5”,其指示该剪辑作为第一无缝连接(参考图 12)而连接到下一剪辑。在下一步骤 S206 中,播放项目文件的字段 NumberOfPlayItems 的值递增

“1”,其指示已经将一个播放项目添加到播放列表中(参考图 11)。

[0430] 在下一步骤 S207 中,设置块 blkPlayItem() 的字段 ClipInformationFileName、字段 INTime 和字段 OUTTime,从而创建在记录剪辑时所添加的块 blkPlayItem()。在上述步骤 S205 中创建的剪辑信息文件的文件名“00002. dpi”被存储在字段 ClipInformationFileName 中。具体地,因为剪辑信息文件的扩展名是固定的,所以存储句点之前的部分“00002”。例如,基于剪辑信息文件“00002. dpi”中的块 blkCPI() 的块 blkEPMMap() 的信息,字段 INTime 和字段 OUTTime 是表示在相对应的剪辑 AV 流文件“00002. m2ts”中存储的视频流的开始和末尾端的时间的信息。

[0431] 在下一步骤 S208 中,作为附加可写候选者的播放列表文件中的块 blkPlayListMark() 中的字段 NumberOfPlayListMarks 的值递增“1”。因此,添加到 for 循环语句的字段 MarkTimeStamp 的值被设置为上述步骤 S207 中的块 blkPlayItem() 中的字段 INTime 的值。换句话说,在新记录剪辑的开始处放置播放列表标记。当放置播放列表标记时,形成章。换句话说,因此,将章附加写入到作为附加可写候选者的播放列表文件中。

[0432] 用这样的方式,创建用于新记录剪辑 AV 流文件“00002. m2ts”的剪辑信息文件“00002. dpi”,并更新作为附加可写候选者的播放列表文件。另外,此时,可以更新播放列表文件的扩展数据的块 blkPlayListExtensionData() 中的块 blkPlayListMeta() 的信息。

[0433] 在步骤 S208 的处理之后,可执行在上述步骤 S203 中的写入在流缓冲器 14 中存储的数据的处理。

[0434] 当通过新创建播放列表文件而形成章时,步骤 S205 之后的处理稍微不同于上述处理。换句话说,新创建播放列表文件的每个字段的数据。作为替代,可以预期将要改变为播放列表文件提供的模板的数据。

[0435] 接下来,将描述本发明的另一实施例。在上文中,描述了本发明应用于单个记录设备的示例(参考图 41)。相反,在其它实施例中,本发明应用于具有图像捕捉装置和光学系统的摄像机设备,其将从对象发出的光输入到图像捕捉装置,并且基于由图像捕捉装置所捕捉的捕捉图像信号而将视频数据记录到记录介质中。

[0436] 图 53 示出根据本发明的其它实施例的摄像机设备 100 的结构示例。因为参考图 41 描述的记录设备的结构几乎能够应用于摄像机设备 100 的记录系统的结构,所以与图 41 中示出的部分相同的部分用相同的附图标记来指定,并且将省略对它们的详细描述。

[0437] 在图 53 中示出的结构中,相机部分 50 具有光学系统 51、图像捕捉装置 52、捕捉图像信号处理部分 53、相机控制部分 54 和显示部分 55,作为关于视频信号的结构。相机部分 50 具有麦克风(MIC)56 和音频信号处理部分 57,作为关于音频信号的结构。通过与相机部分 50 的每个部分交换各种控制信号和信息,控制部分 30 控制相机部分 50 的操作。另外,相机部分 50 基于根据用户操作从 UI 部分 31 提供的控制信号而控制相机部分 50 的操作。

[0438] 当构造为摄像机设备 100 时,正常地执行记录开始操作和记录停止操作,例如,使得使用位于 UI 部分 31 中的单个记录开关,并在每当按下记录开关时,交替地命令记录开始和停止。在该摄像机设备 100 中,诸如可记录型 DVD 或者 Blu-ray Disc 的盘记录介质应用于记录介质 20。

[0439] 在相机部分 50 中,光学系统 51 具有引导光线从对象到图像捕捉装置 52 的透镜系统、光圈调整机构、焦点调整机构、缩放机构、快门机构等等。基于从控制部分 30 提供的控

制信号,由摄像机控制部分 54 控制光圈调整机构、焦点调整机构、缩放机构和快门机构的操作。

[0440] 例如,图像捕捉装置 52 由 CCD(电荷耦合器件)组成。图像捕捉装置 52 通过光电变换将通过光学系统 51 发出的光转换为电信号,执行预定的信号处理,以及输出结果信号作为捕捉图像信号。捕捉图像信号处理部分 53 对从图像捕捉装置输出的捕捉图像信号执行预定信号处理,并输出所得的信号作为基带数字视频数据。

[0441] 例如,捕捉图像信号处理部分 53 使得 CDS(相关双倍采样)电路仅仅采样具有从图像捕捉装置 52 输出的捕捉图像信号的图像信息的信号以及从采样信号中去除噪音,并使得 AGC(自动增益控制)电路调节信号的增益。其后,捕捉图像信号处理部分 53 通过 A/D 转换将增益调节后的信号转换为数字信号。其后,捕捉图像信号处理部分 53 对数字信号执行检测信号处理,从数字信号中提取 R(红)、G(绿)和 B(蓝)的颜色成分,执行诸如 γ 校正、白色平衡校正等的处理,并且最终输出所得的信号,作为一个基带的数字视频数据。

[0442] 另外,捕捉图像信号处理部分 53 向控制部分 30 提供有关已经从图像捕捉装置 52 输出的捕捉图像信号的信息。控制部分 30 基于该信息而生成用于控制光学系统 51 的控制信号并向摄像机控制部分 54 提供该控制信号。摄像机控制部分 54 基于该控制信号而控制焦点调整机构、光圈调整机构等。

[0443] 另外,捕捉图像信号处理部分 53 基于已经从图像捕捉装置 52 输出的捕捉图像信号而生成图像信号,其使得使用例如 LCD(液晶显示器)作为显示设备的显示部分 55 显示。

[0444] 另一方面,麦克风 56 收集周围的声音,将它转换成电信号,并且输出电信号。从麦克风 56 输出的音频信号被提供给音频信号处理部分 57。音频信号处理部分 57 通过限幅器对所提供的音频信号执行 A/D 转换,对数字音频数据执行预定音频信号处理,诸如减噪和声音质量校正,并输出所得的数据作为基带数字音频数据。

[0445] 将已经从相机部分 50 的捕捉图像信号处理部分 53 输出的基带数字视频数据提供给记录部分 10 的终端 40。另一方面,将从音频信号处理部分 57 输出的基带数字音频数据提供给记录部分 10 的终端 41。

[0446] 当记录介质 20 加载到处于用于捕捉的预定方式下的摄像机设备 100 时,标识用于基于通过捕捉获取的视频数据的章、作为附加可写候选者的播放列表文件,并且根据参考图 43 描述的处理,确定是否能够将该章附加写入到作为附加可写候选者的标识播放列表文件中。

[0447] 例如,在控制部分 30 的控制下从记录介质 20 中读取索引文件并且通过管理信息处理部分 16 将其存储到非易失性存储器 17 中。控制部分 30 基于有关在非易失性存储器 17 中存储的索引文件的信息而标识作为附加可写候选者的播放列表文件,并且向记录控制部分 15 发出命令以从记录介质 20 中读取作为附加可写候选者的播放列表文件。通过管理信息处理部分 16,将已经基于该命令从记录介质 20 中读取的、作为附加可写候选者的播放列表文件存储在非易失性存储器 17 中。

[0448] 基于在非易失性存储器 17 中存储的有关作为附加可写候选者的播放列表文件的信息,控制部分 30 向记录控制部分 15 发出命令以从记录介质 20 中读取与作为附加可写候选者的播放列表文件相关联的所有剪辑信息文件。将已经基于该命令而从记录介质 20 中读取的剪辑信息文件存储到非易失性存储器 17 中。基于在非易失性存储器 17 中存储的作

为附加可写候选者的播放列表文件和与作为附加可写候选者的播放列表文件相关联的所有剪辑信息文件,控制部分 30 执行图 43 中示出的流程图中从步骤 S104 至步骤 S111 的确定。将确定结果存储到例如控制部分 30 的寄存器中。

[0449] 在控制部分 30 执行步骤 S104 至步骤 S111 的确定之后,处理流程前进到图 43 中示出的步骤 S112。在步骤 S112,控制部分 30 执行对步骤 S104 至步骤 S111 的确定结果的总确定,以及确定是否将基于已经捕捉的视频数据的章附加写入到作为附加可写候选者的播放列表文件中。控制部分 30 控制管理信息处理部分 16,使得当确定结果指示将章附加写入到播放列表文件中时,将章附加写入到在非易失性存储器 17 中存储的作为附加可写候选者的播放列表文件中,而当确定结果指示不将章附加写入到播放列表文件时,新创建播放列表文件并且将该章记录到其中。

[0450] 当在记录停止状态下按下位于 UI 部分 31 的记录开关时,命令记录开始的控制信号从 UI 部分 31 提供给控制部分 30。在控制部分 30 的控制下,开始将已经从相机部分 50 输出的基带数字视频信号和数字音频数据记录到记录介质 20 中。

[0451] 换句话说,如上所述,在控制部分 30 的控制下,开始视频编码器 11 和音频编码器 12 的操作。视频数据和音频数据分别由视频编码器 11 和音频编码器 12 压缩编码、由多路复用器 13 以预定方式分组化并多路复用为 AV 流数据。AV 流数据通过流缓冲器 14 提供给记录控制部分 15,并被作为剪辑 AV 流文件而记录到记录介质 20 中。

[0452] 当再次按下 UI 部分 31 的记录开关时,记录停止,创建剪辑信息文件,并更新播放列表文件。管理信息处理部分 16 基于从多路复用器 13 和记录控制部分 15 提供的信息,创建与记录到记录介质 20 中的剪辑 AV 流文件相对应的剪辑信息文件和参照剪辑信息文件的播放项目。

[0453] 当确定结果指示将章附加写入到作为附加可写候选者的播放列表文件时,将所创建的播放项目添加到作为附加可写候选者的播放列表文件中,并放置播放列表标记,以便形成章。相反,当确定结果指示不能将章附加写入作为附加可写候选者的播放列表文件中时,将所创建的播放项目添加到新创建的播放列表文件中,并设置播放列表标记。

[0454] 当在该状态下再次按下记录开关时,命令记录开始,从而开始将新的剪辑 AV 流记录到记录介质 20 中。当再次开始记录时,根据图 43 中示出的流程图,可对于基于新记录到作为附加可写候选者的播放列表文件中的新章,执行附加记录能 / 不能确定。

[0455] 在上文中,图 41 中示出的记录设备和图 53 中示出的摄像机设备 100 的记录部分 10 被描述为硬件结构。然而,本发明并不限于这样的示例。换句话说,记录部分 10 可被构造为软件。在这种情况下,将软件预存在例如控制部分 30 的 ROM(未示出)中。作为替代,可以在诸如个人计算机上的计算机设备上构造记录部分 10。在这种情况下,使得计算机设备执行记录部分 10 的软件被记录在诸如 CD-ROM 或者 DVD-ROM 的记录介质中,并通过它们提供。当计算机设备可与网络相连接时,可通过诸如因特网的网络提供软件。

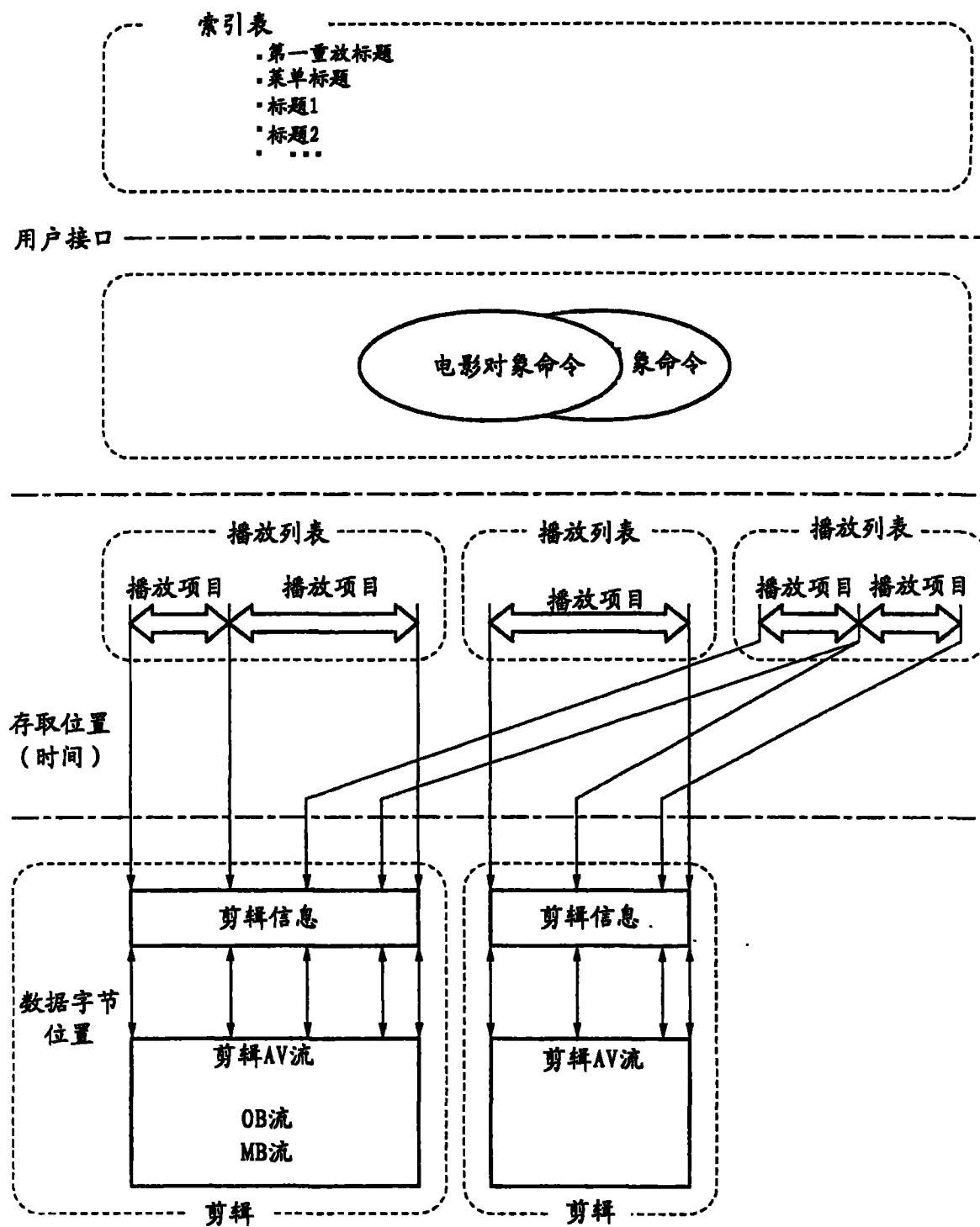


图 1

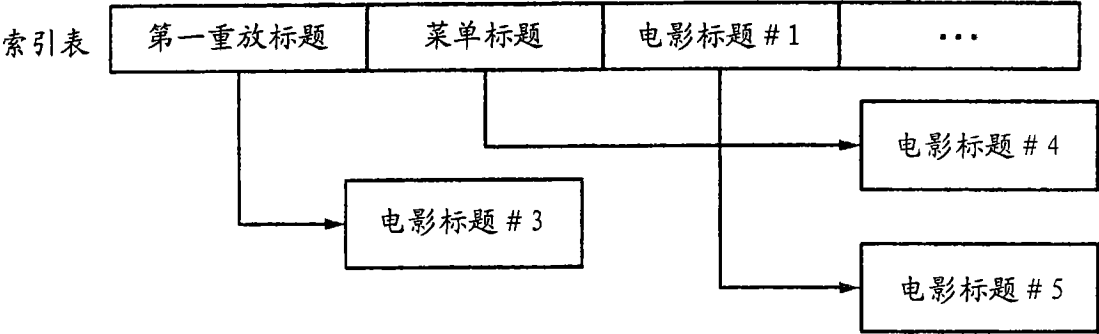


图 2

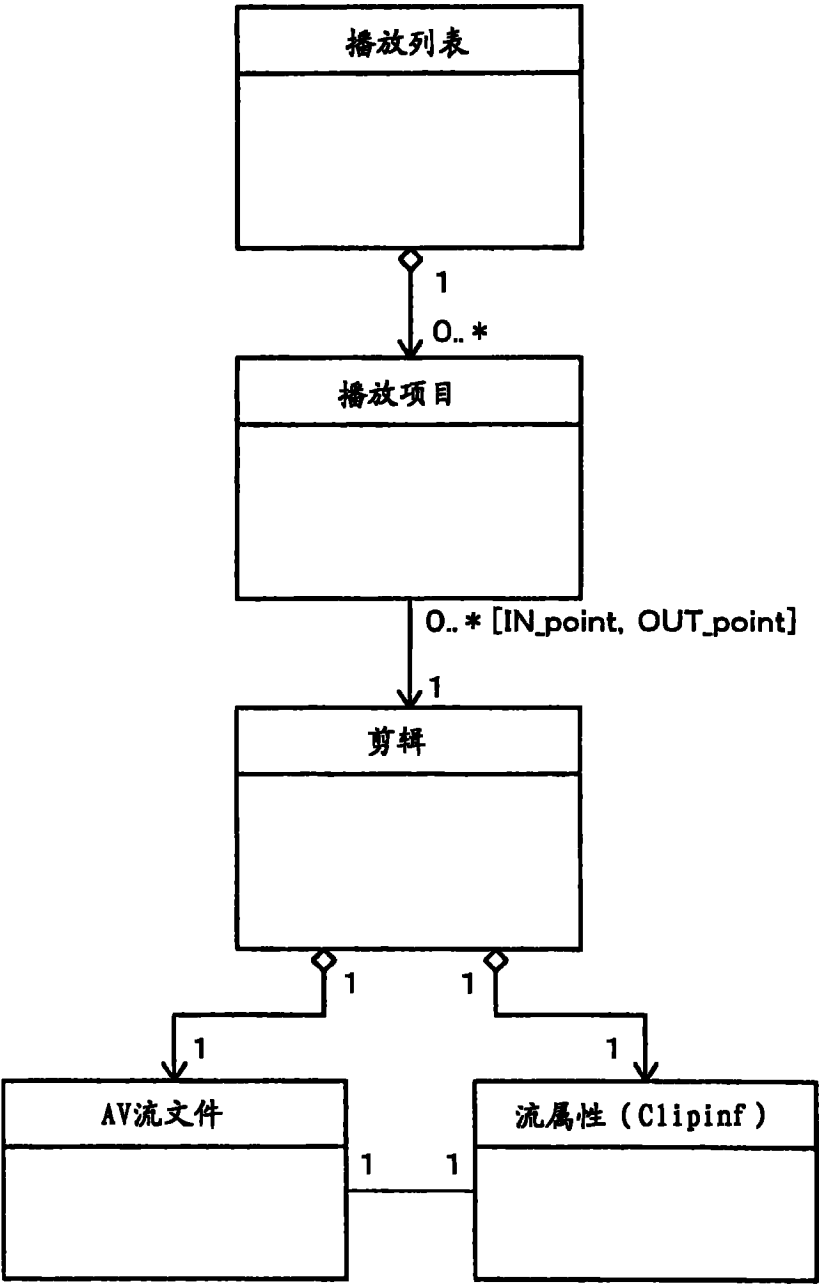


图 3

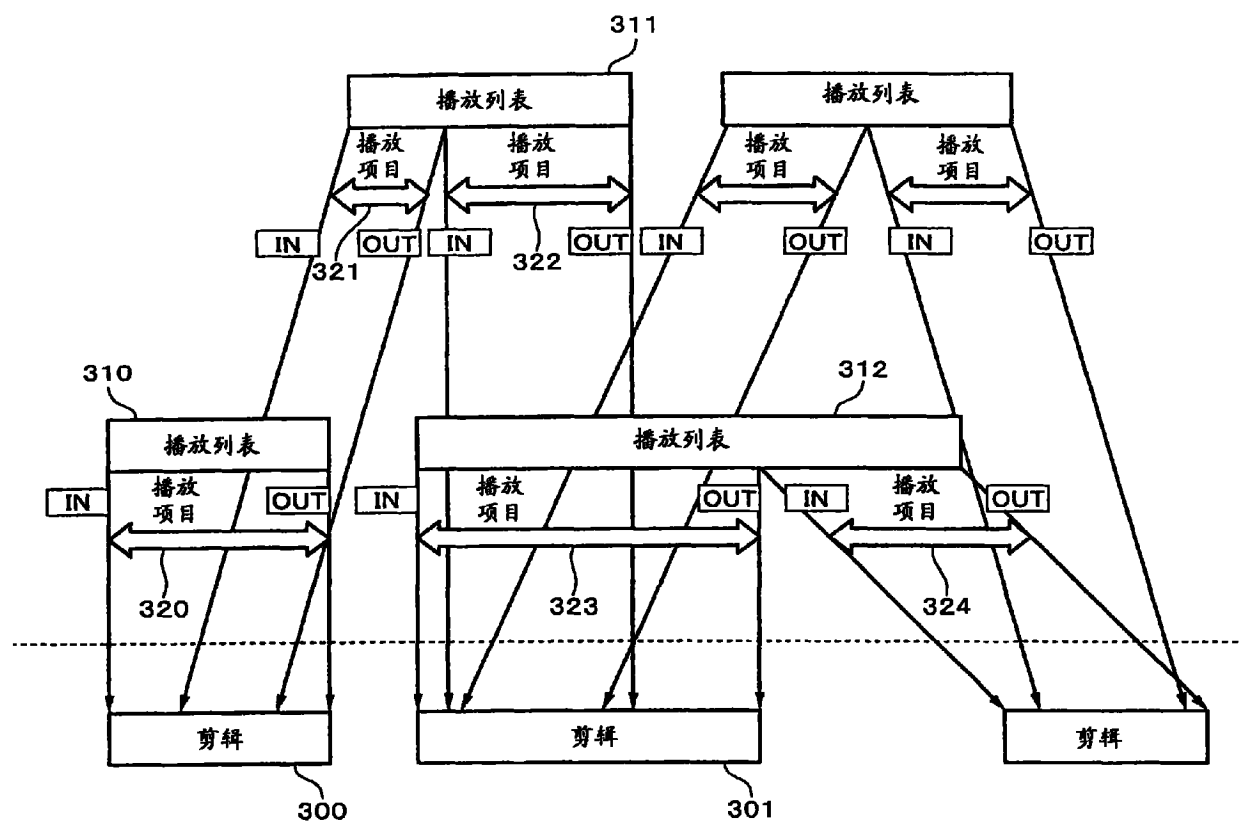


图 4

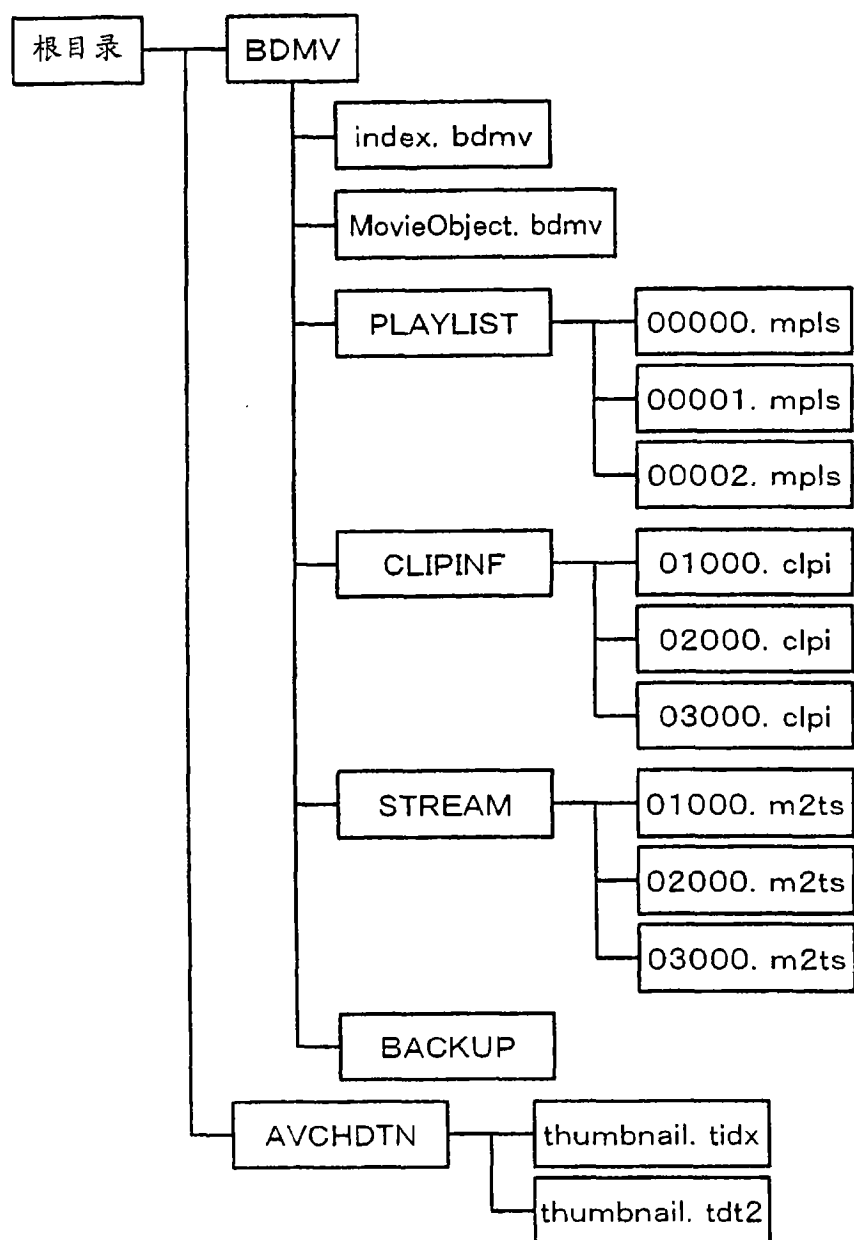


图 5

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
FirstPlaybackTitleMobIDRef	16	uimsbf
reserved	32	bslbf
}		
MenuTitle() {		
reserved	1	bslbf
'1'	1	bslbf
reserved	31	bslbf
'1'	1	bslbf
reserved	14	bslbf
MenuTitleMobIDRef	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
MovieTitleMobIDRef[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	uimbsf
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
ClipCodeIdentifier	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

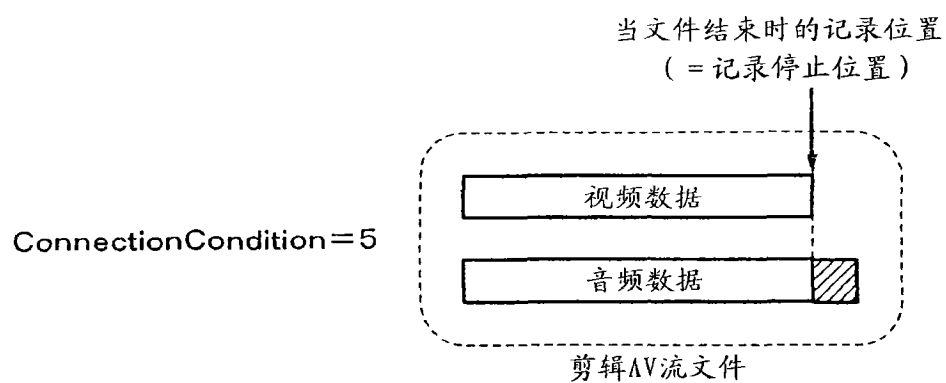


图 13A

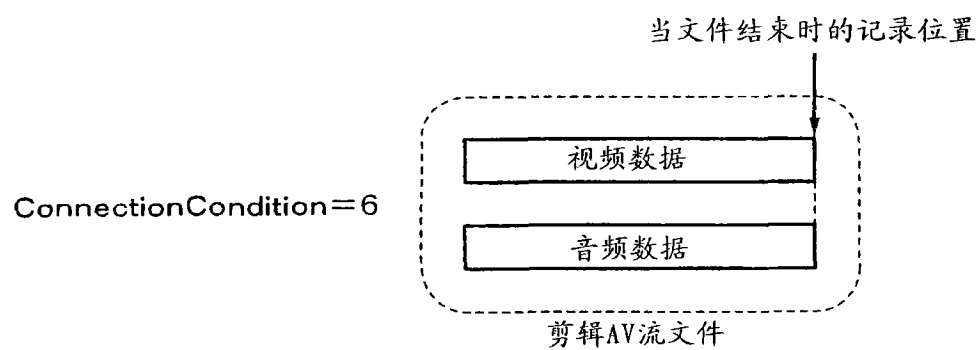


图 13B

语法	数据长度 (位)	助记符
blkPlayListMark() {		
Length	32	uimsbf
NumberOfPlayListMarks	16	uimsbf
for(<i>PL_mark_id</i> =0; <i>PL_mark_id</i> < <i>NumberOfPlayListMarks</i> ; <i>PL_mark_id</i> ++) {		
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 == 1 _b) {		
reserved	8	bslbf
FollowingClipStreamType	8	bslbf
reserved	32	bslbf
FollowingClipInformationFileName	8 * 5	bslbf
ClipCodecIdentifier	8 * 4	bslbf
reserved	8	bslbf
}		
}		

图 16

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[<i>stream_index</i>]	16	uimsbf
blkStreamCodingInfo(<i>stream_index</i>)		
}		
}		

图 17

语法	数据长度 (位)	助记符
blkStreamCodingInfo (<i>stream_index</i>) {		
Length	8	uimsbf
StreamCodingType	8	bslbf
if (<i>StreamCodingType</i> ==0x1B) {		
VideoFormat	4	bslbf
FrameRate	4	bslbf
AspectRatio	4	bslbf
reserved	2	bslbf
CCFlag	1	bslbf
reserved	17	bslbf
reserved	128	bslbf
} else if (<i>StreamCodingType</i> ==0x80 <i>StreamCodingType</i> ==0x81) {		
AudioPresentationType	4	bslbf
SamplingFrequency	4	bslbf
AudioLanguageCode	8 * 3	bslbf
reserved	128	bslbf
} else if (<i>StreamCodingType</i> ==0x90) { // Overlay Bitmap stream		
OBLanguageCode	8 * 3	bslbf
reserved	136	bslbf
} else if (<i>StreamCodingType</i> ==0x91) { // Menu Bitmap stream		
MBLanguageCode	8 * 3	bslbf
reserved	136	bslbf
}		
}		

图 18

VideoFormat	描述	标准
0	保留	
1	480i	ITU-R BT.601-4 ^[38]
2	576i	ITU-R BT.601-4 ^[38]
3	480p	SMPTE 293M ^[39]
4	1080i	SMPTE 274M ^[40]
5	720p	SMPTE 296M ^[41]
6	1080p	SMPTE 274M ^[40]
7	576p	ITU-R BT.1358 ^[42]
8-14	保留	
15	保留	

图 19

FrameRate	描述 [Hz]
0	保留
1	24000/1001 (23.976...)
2	24
3	25
4	30000/1001 (29.97...)
5	保留
6	50
7	60000/1001 (59.94...)
8	保留
9-14	保留
15	保留

图 20

AspectRatio	描述
0	保留
1	保留
2	显示纵横比=4: 3
3	显示纵横比=16: 9
4	保留
5-14	保留
15	保留

图 21

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

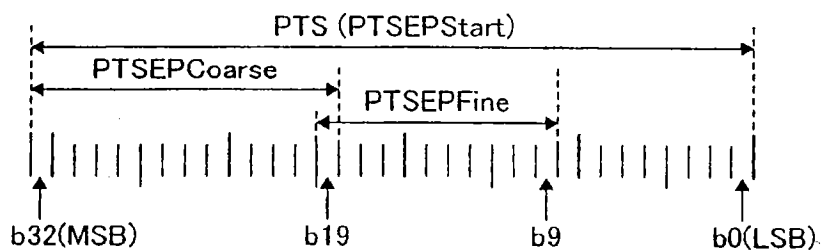
图 22

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 23

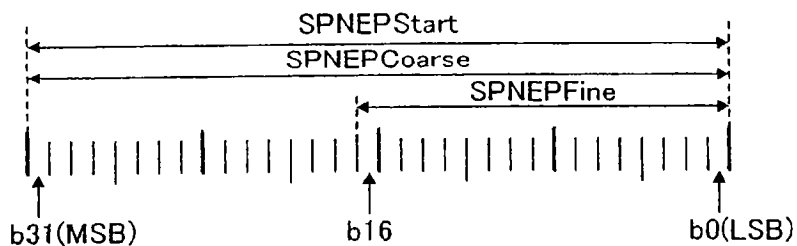
语法	数据长度 (位)	助记符
blkEPMapForOneStreamPID (<i>EP_stream_type</i> , <i>Nc</i> , <i>Nf</i>) {		
EPFineTableStartAddress	32	uimsbf
for (<i>i</i> =0; <i>i</i> < <i>Nc</i> ; <i>i</i> ++) {		
// EP coarse table		
RefToEPFineID [<i>i</i>]	18	uimsbf
PTSEPCoarse [<i>i</i>]	14	uimsbf
SPNEPCoarse [<i>i</i>]	32	uimsbf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

图 24



- PTS :b0 ... b32 (33位, 90kHz)
- PTSEPFine :b9 ... b19 (11位, 具有5.7msec的分辨率, 在大约接近11.5sec时绕回)
- PTSEPCoarse :b19 ... b32 (14位, 具有5.8sec的分辨率, 在大约接近26.5h时绕回)

图 25



- SPNEPStart :b0 ... b31 (32位)
- SPNEPFine :b0 ... b16 (17位, 在大约接近AV流文件的25MB时绕回)
- SPNEPCoarse :b0 ... b31 (32位)

图 26

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<L1; i++) {		
padding_word	16	bslbf
padding_word	16	bslbf
}		
DataBlock()	32+ 8*(Length - DataBlockStartAddress)	
}		
}		

图 27

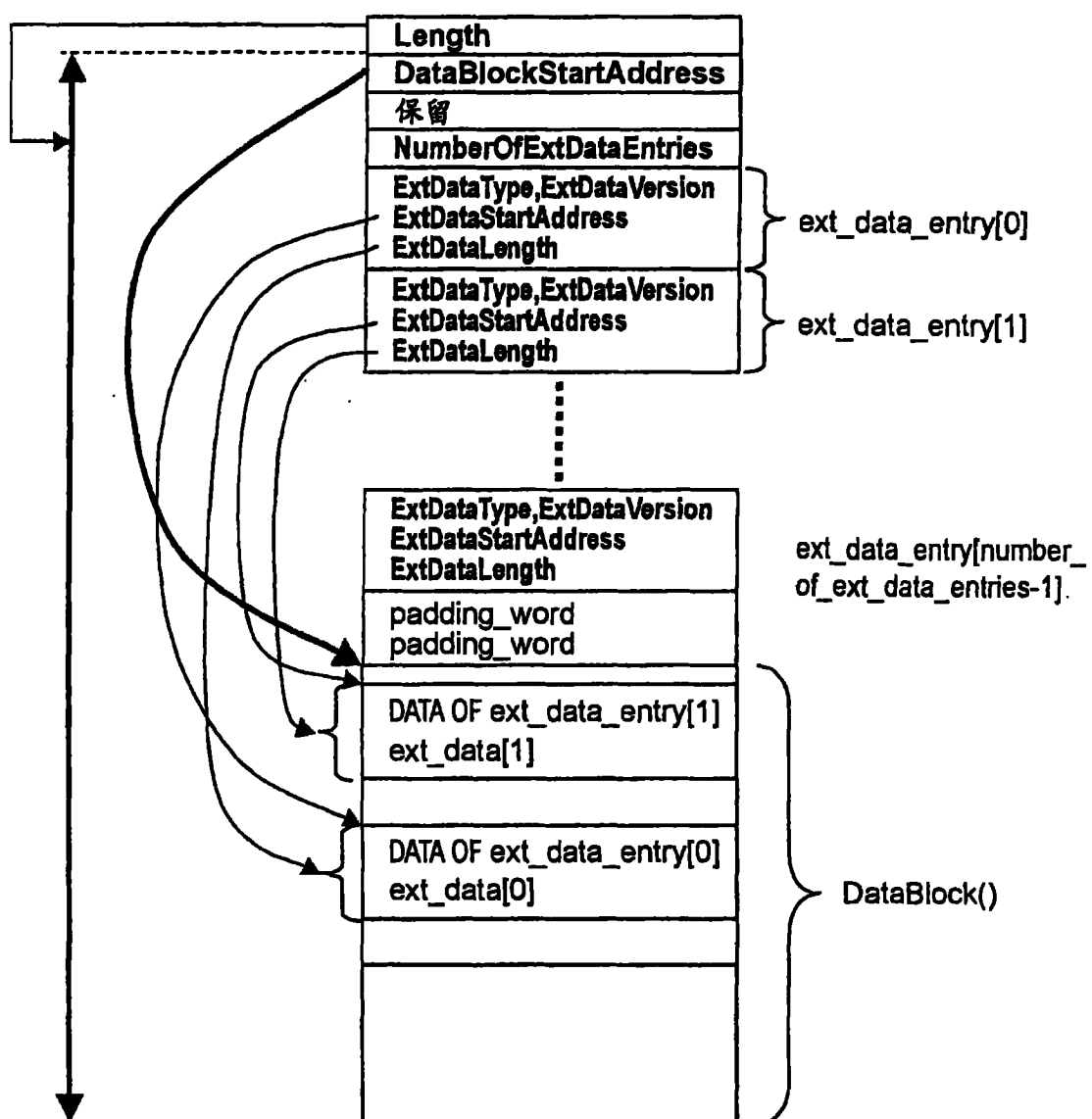


图 28

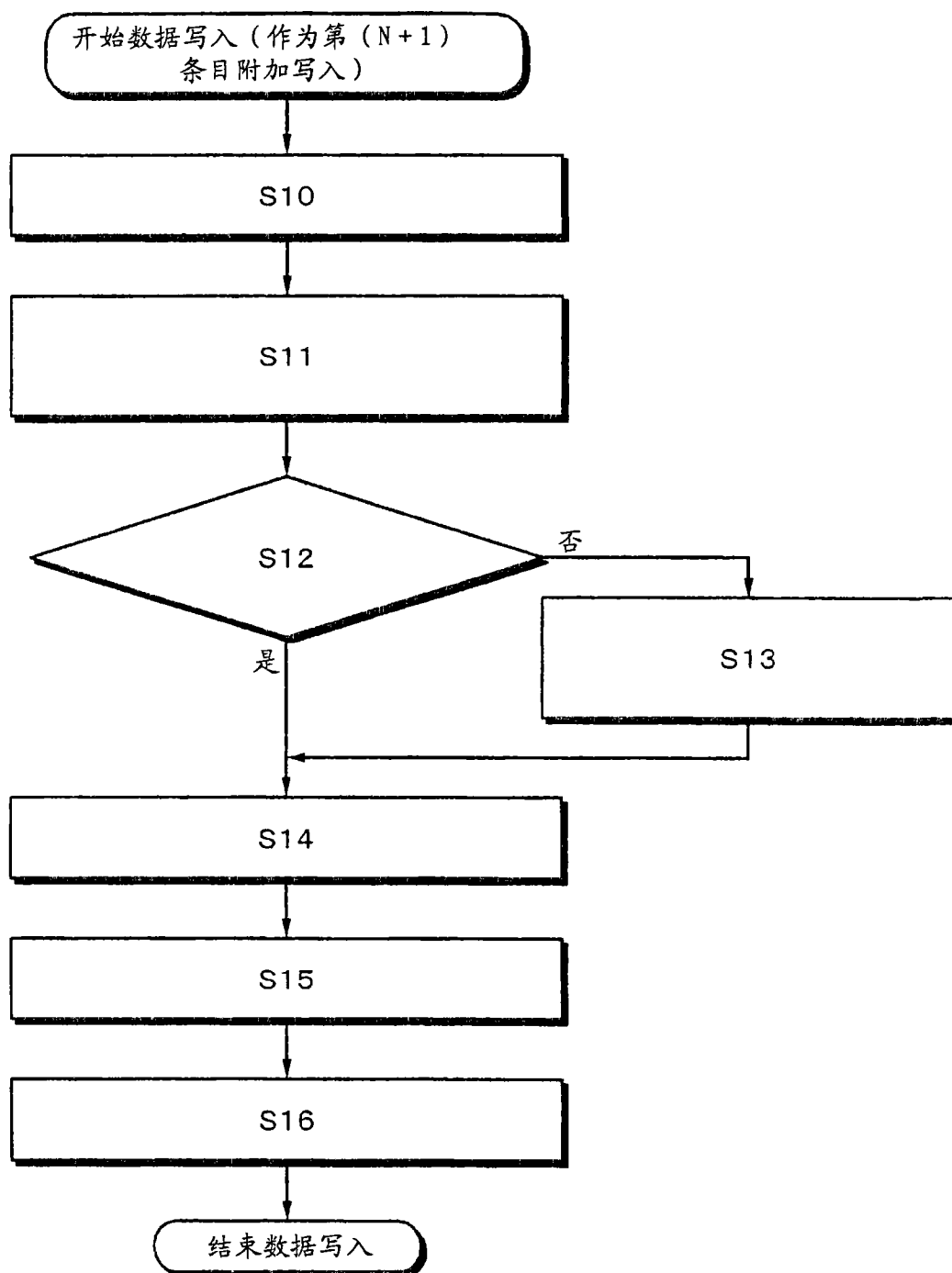


图 29

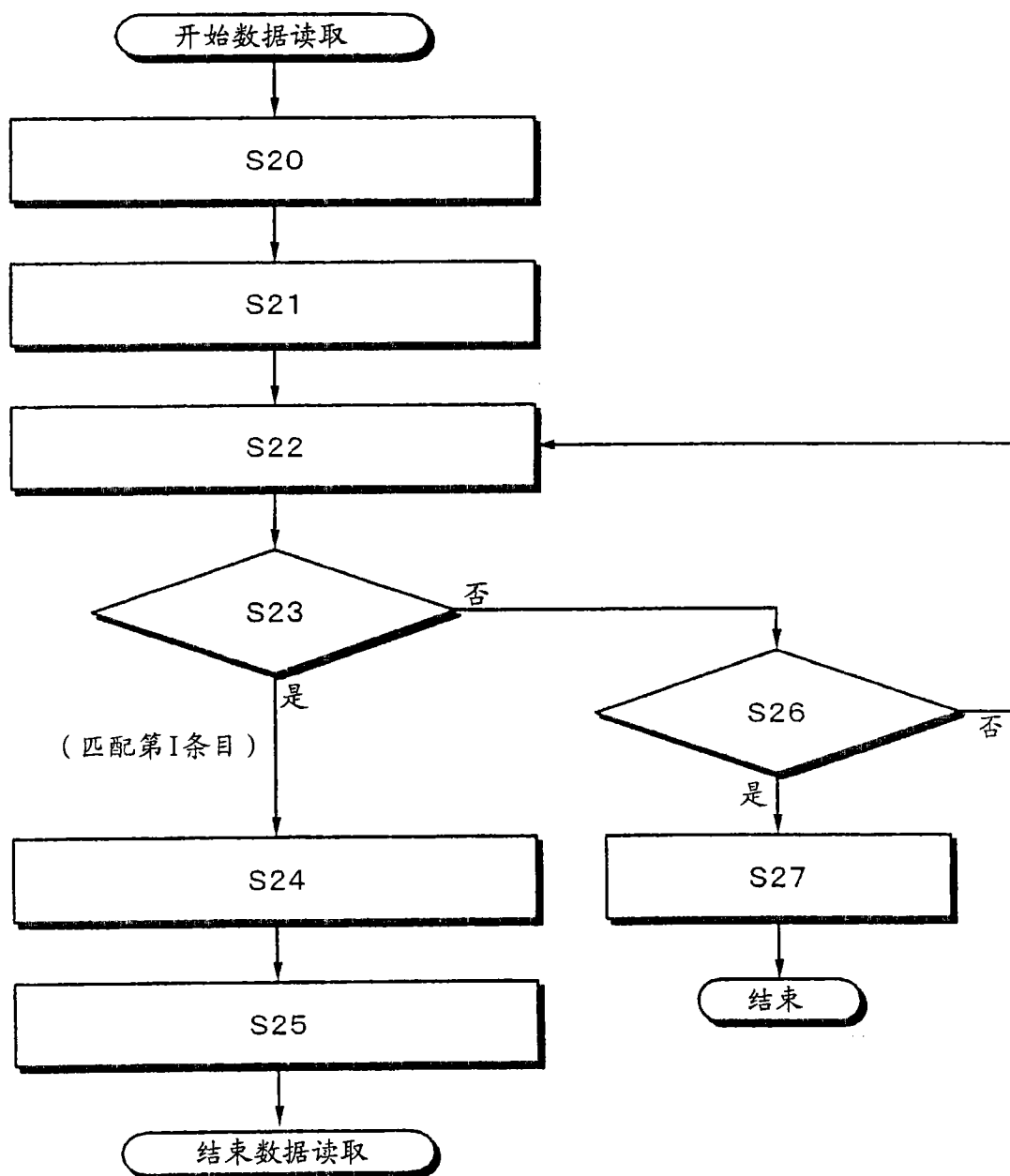


图 30

语法	数据长度 (位)	助记符
blkIndexExtensionData() {		
TypeIndicator	8 * 4	uimsbf
保留	8 * 4	bslbf
TableOfPlayListsStartAddress	32	uimsbf
MakersPrivateDataStartAddress	32	uimsbf
保留	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 31

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
保留	6	bslbf
PlayListAttribute	2	uimsbf
保留	16	bslbf
RefToTitleId	16	uimsbf
}		
}		
}		

图 32

语法	数据长度 (位)	助记符
blkPlaylistExtensionData() {		
TypeIndicator	8 * 4	bslbf
保留	8 * 4	bslbf
PlaylistMarkExtStartAddress	32	uimsbf
MakersPrivateDataStartAddress	32	uimsbf
保留	192	bslbf
blkPlaylistMeta()		
for (i=0; i<N1; i++){		
padding_word	16	bslbf
}		
blkPlaylistMarkExt()		
for (i=0; i<N2; i++){		
padding_word	16	bslbf
}		
blkMakersPrivateData()		
for (i=0; i<N3; i++){		
padding_word	16	bslbf
}		
}		

图 33

语法	数据长度 (位)	助记符
blkPlaylistMeta() {		
Length	32	uimsbf
MakerID	16	uimsbf
MakerModelCode	16	uimsbf
保留	32	bslbf
RefToMenuThumbnailIndex	16	uimsbf
blkTimeZone()	8	uimsbf
RecordTimeAndDate	4 * 14	uimsbf
保留	8	bslbf
PlayListCharacterSet	8	uimsbf
PlayListNameLength	8	bslbf
PlayListName	8 * 255	uimsbf
Additional_data(){		
Length2	32	uimsbf
保留	8 * Length2	bslbf
}		
}		

图 34

语法	数据长度 (位)	助记符
blkMakersPrivateData() {		
Length	32	uimsbf
if(Length !=0){		
DataBlockStartAddress	32	uimsbf
保留	24	bslbf
NumberOfMakerEntries	8	uimsbf
for (i=0; i<NumberOfMakerEntries; i++){		
MakerID	16	uimsbf
MakerModelCode	16	uimsbf
MpdStartAddress	32	uimsbf
MpdLength	32	uimsbf
}		
for (i=0; i<L1; i++) {		
padding_word	16	bslbf
padding_word	16	bslbf
}		
DataBlock()	32+ 8 * (Length - DataBlockStartAddress)	
}		
}		

图 35

语法	数据长度 (位)	助记符
blkClipExtensionData() {		
TypeIndicator	8 * 4	bslbf
保留	8 * 4	bslbf
ProgramInfoExtStartAddress	32	uimsbf
MakersPrivateDataStartAddress	32	uimsbf
保留	192	bslbf
blkClipInfoExt()		
for(i=0; i<N1;i++){		
padding_word	16	bslbf
}		
blkProgramInfoExt()		
for(i=0; i<N2;i++){		
padding_word	16	bslbf
}		
blkMakersPrivateData()		
for(i=0; i<N3;i++){		
padding_word		
}	16	bslbf
}		

图 36

语法	数据长度 (位)	助记符
blkProgramInfoExt() {		
Length	32	uimsbf
保留	8	bslbf
NumberOfProgramSequences /* Fixed to 1	8	uimsbf
for (i=0; i<NumberOfProgramSequences; i++) {		
NumberOfStreamCodingInfoExtInPs[i]	8	uimsbf
保留	8	bslbf
for (j=0; j<NumberOfStreamCodingInfoExtInPs[i]; j++) {		
StreamPID[i][j]	16	uimsbf
blkStreamCodingInfoExt(i,j)		
}		
}		
}		

图 37

语法	数据长度 (位)	助记符
blkStreamCodingInfoExt(<i>i,j</i>) {		
Length	8	uimsbf
StreamCodingType	8	bslbf
if (<i>StreamCodingType</i> ==0x1B) {		
HorizontalSize	8	uimsbf
profile_idc	8	uimsbf
constraint_set0_flag	1	bslbf
constraint_set1_flag	1	bslbf
constraint_set2_flag	1	bslbf
constraint_set3_flag	1	bslbf
保留	4	bslbf
level_idc	8	uimsbf
保留	48	bslbf
}		
}		

图 38

图 39A

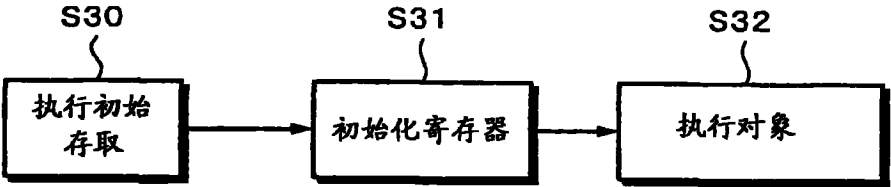
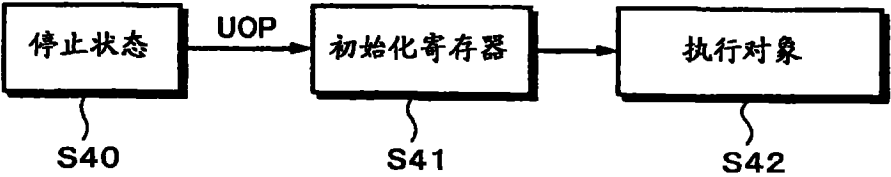


图 39B



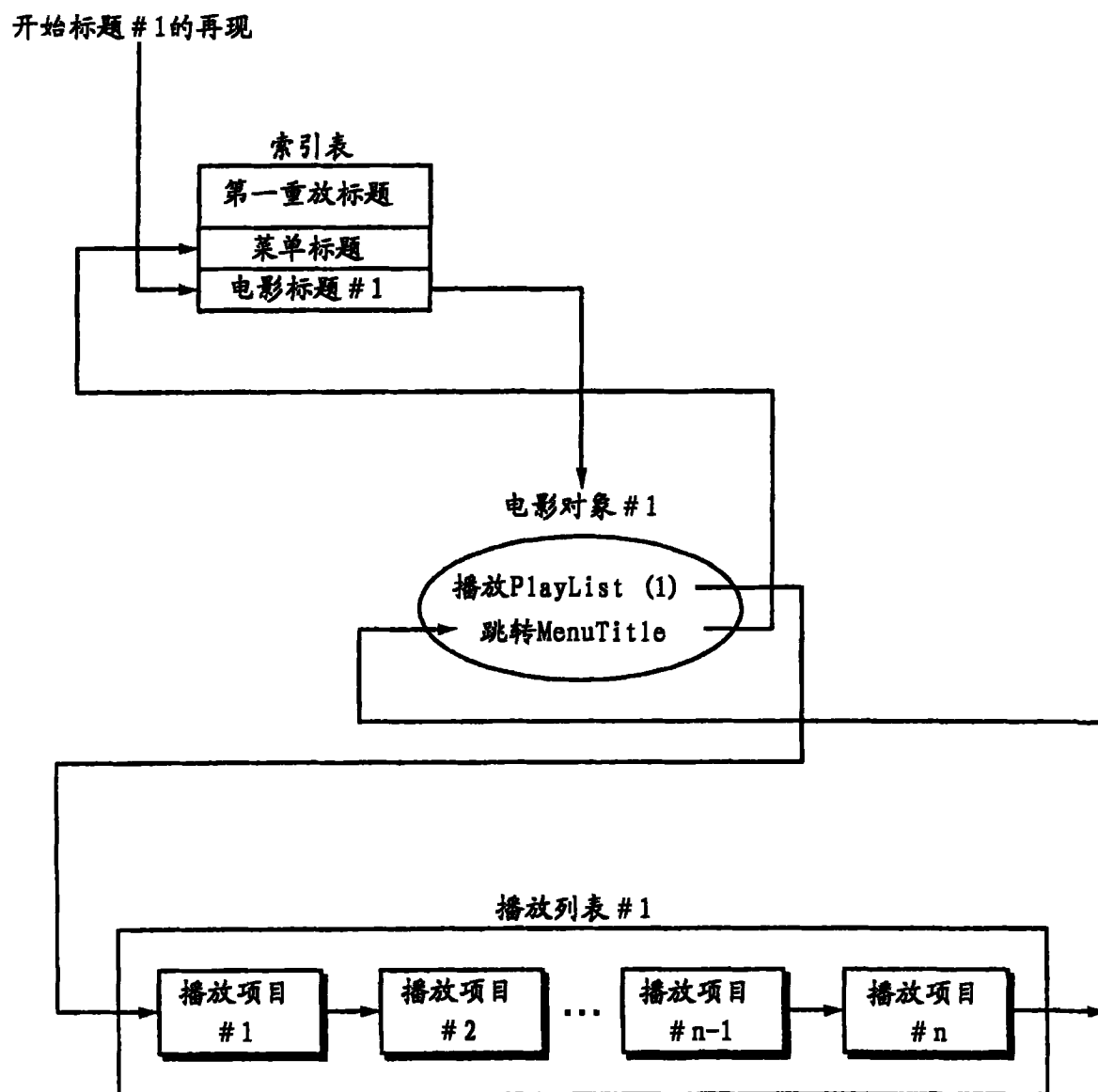


图 40

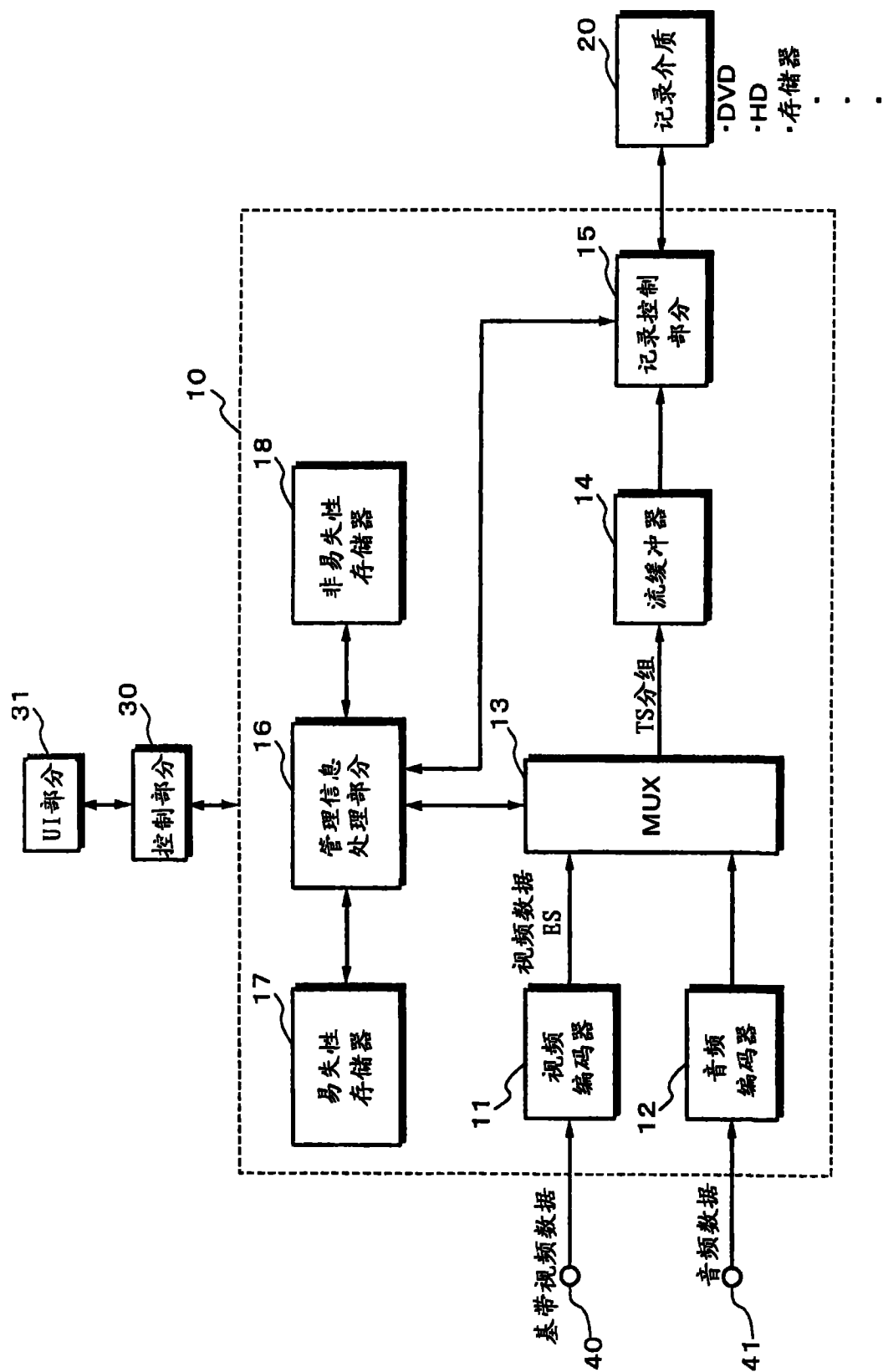


图 41

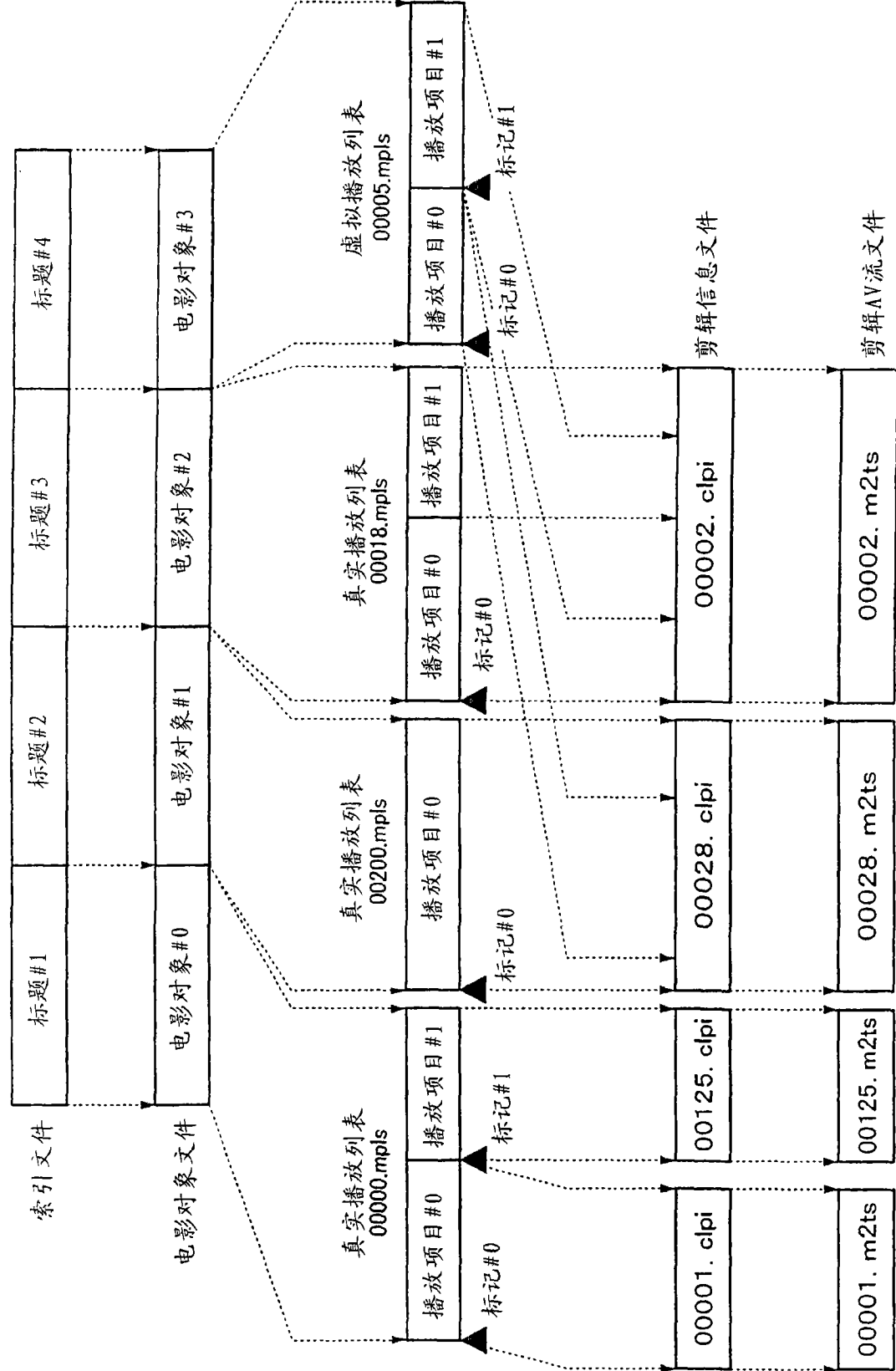


图 42

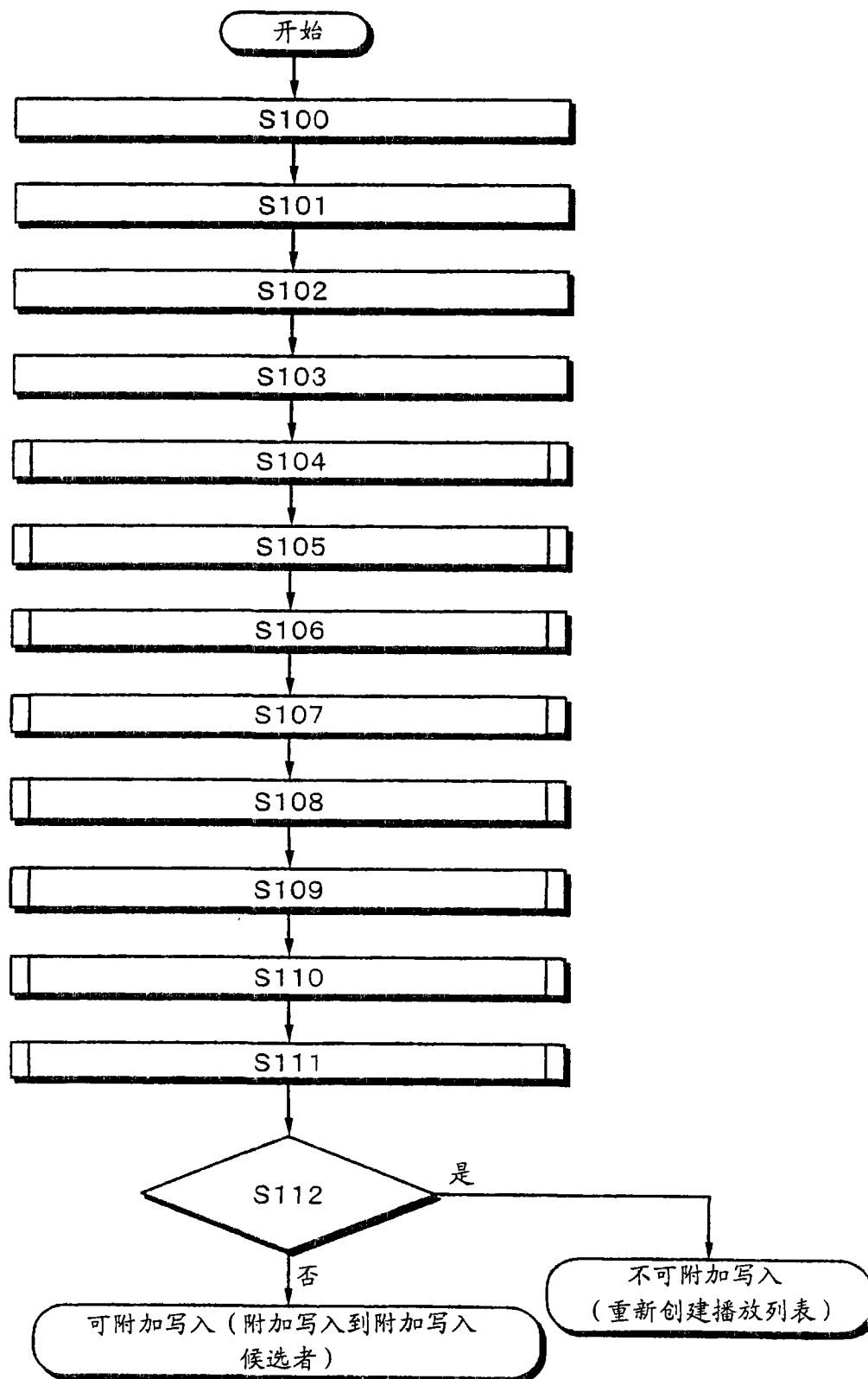


图 43

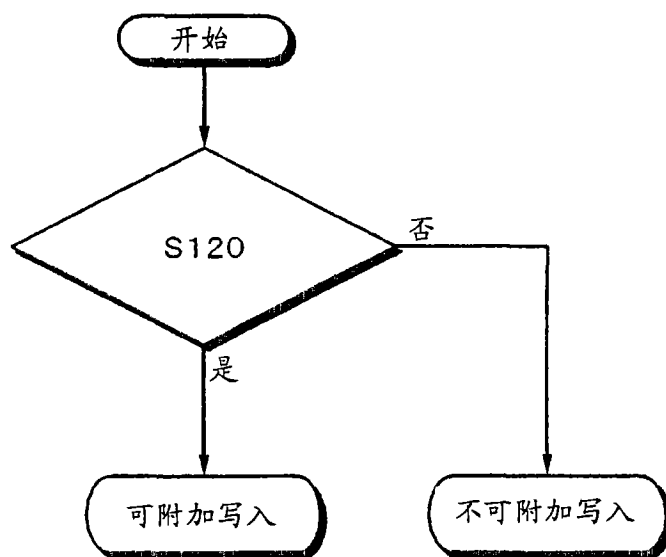
S104

图 44

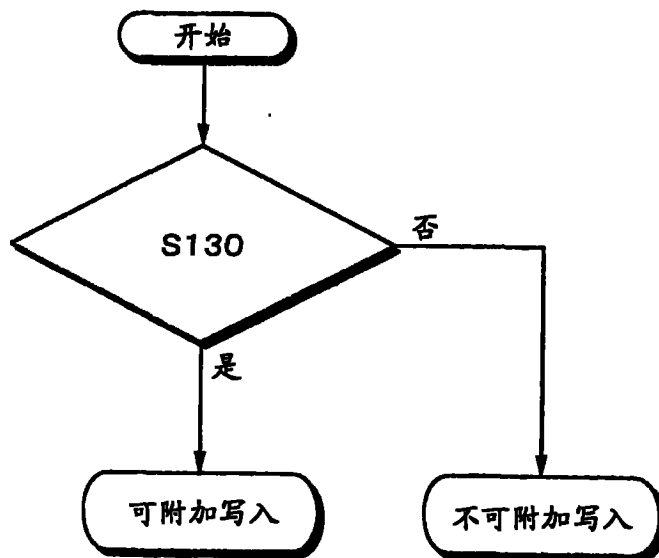
S105

图 45

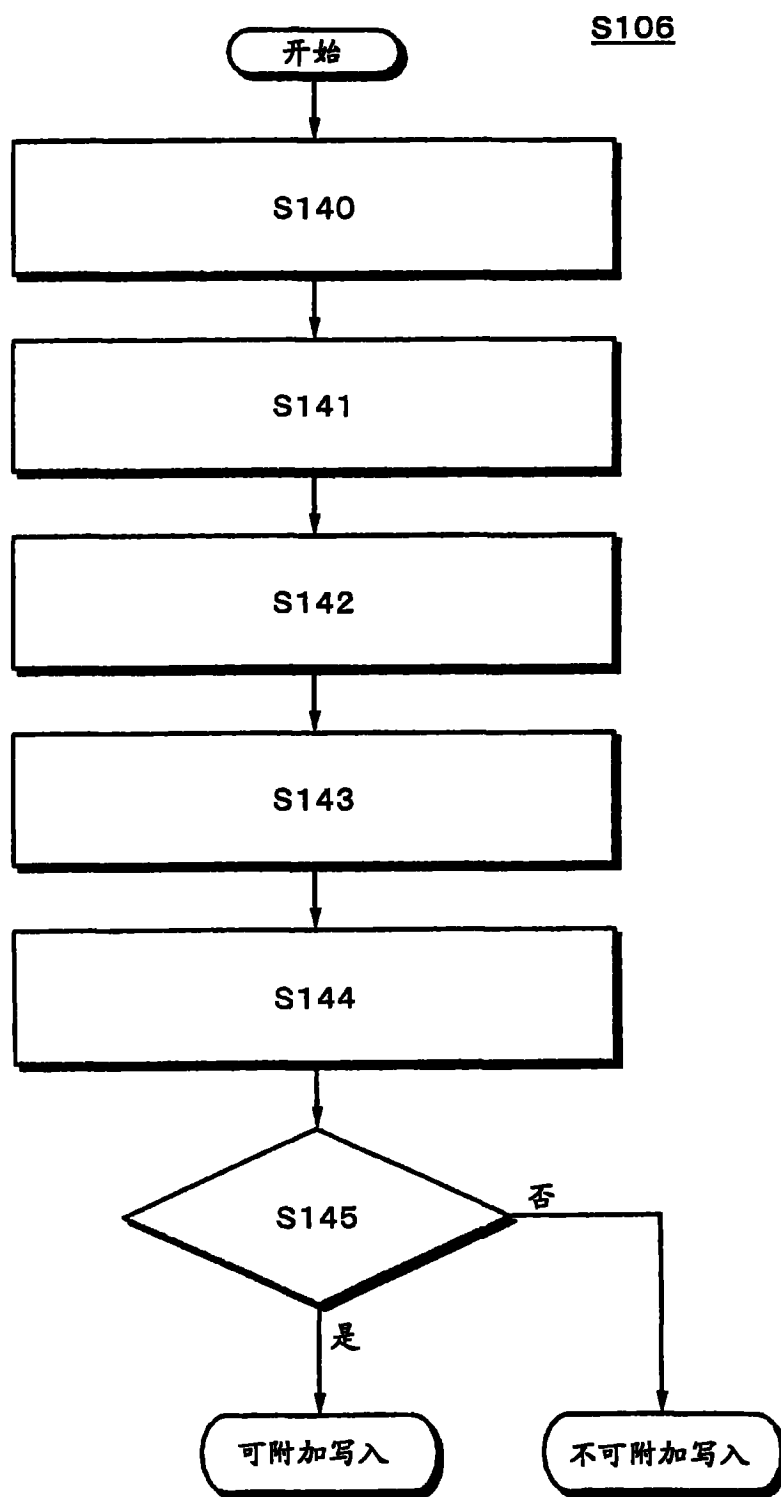


图 46

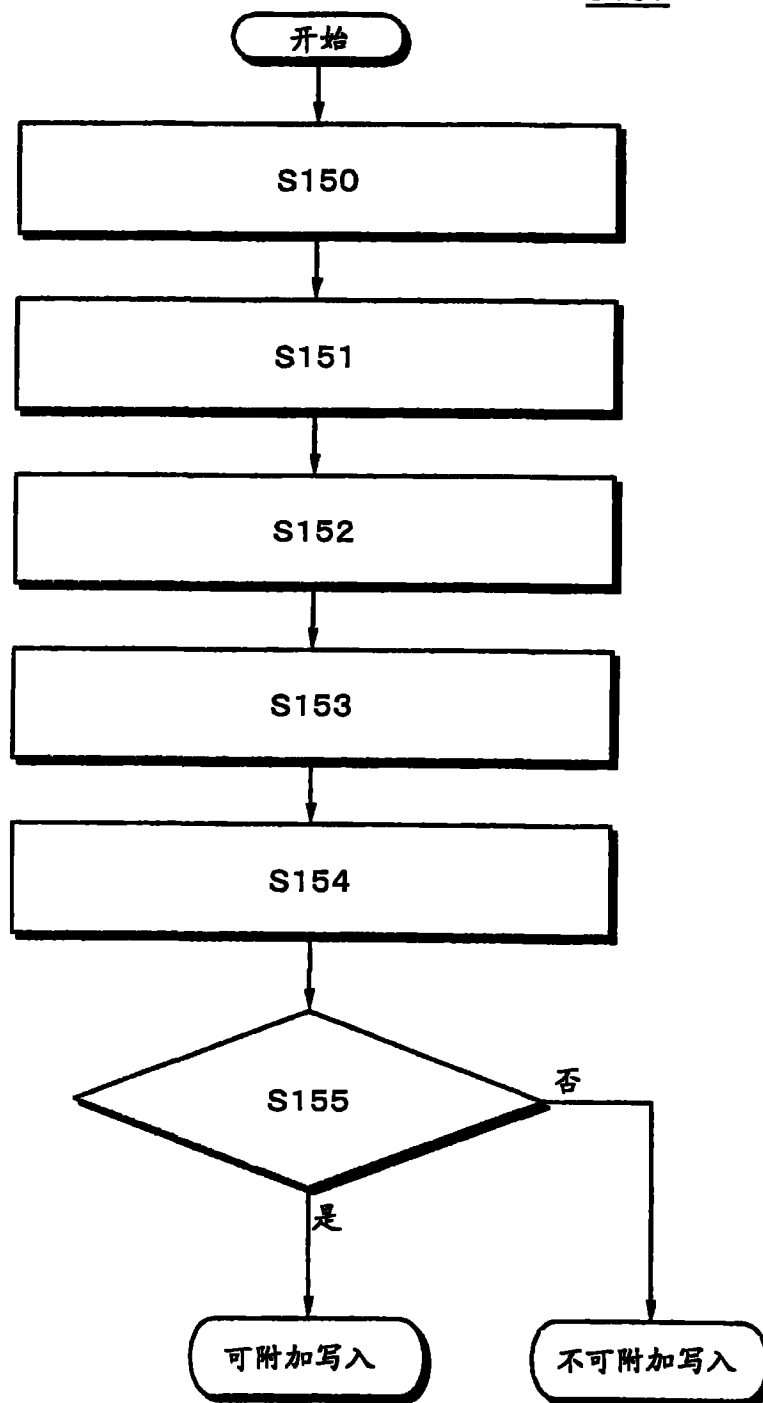
S107

图 47

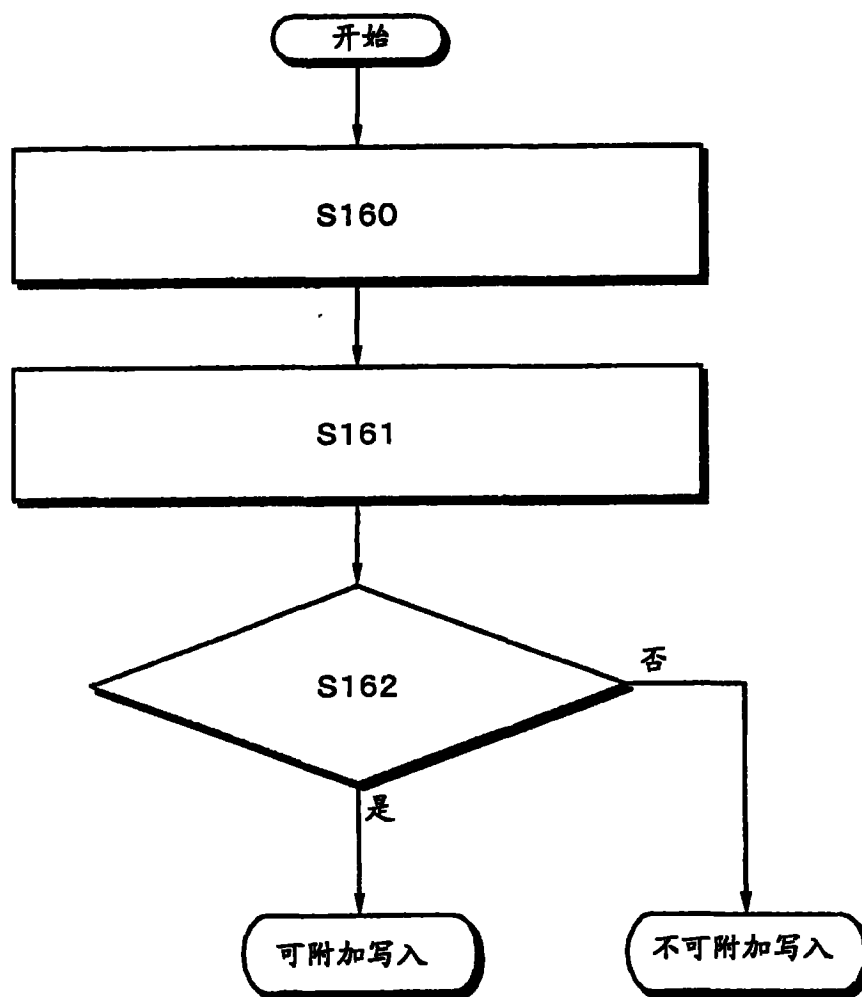
S108

图 48

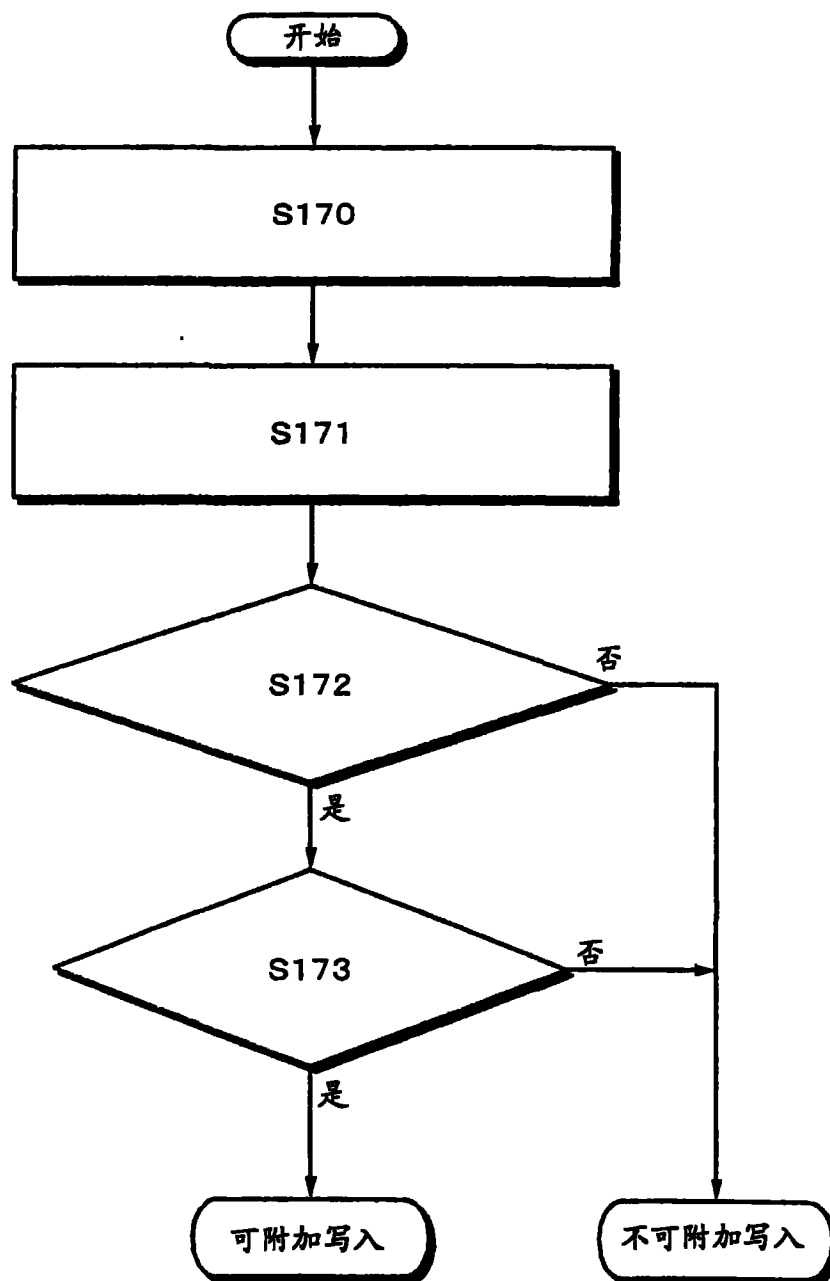
S109

图 49

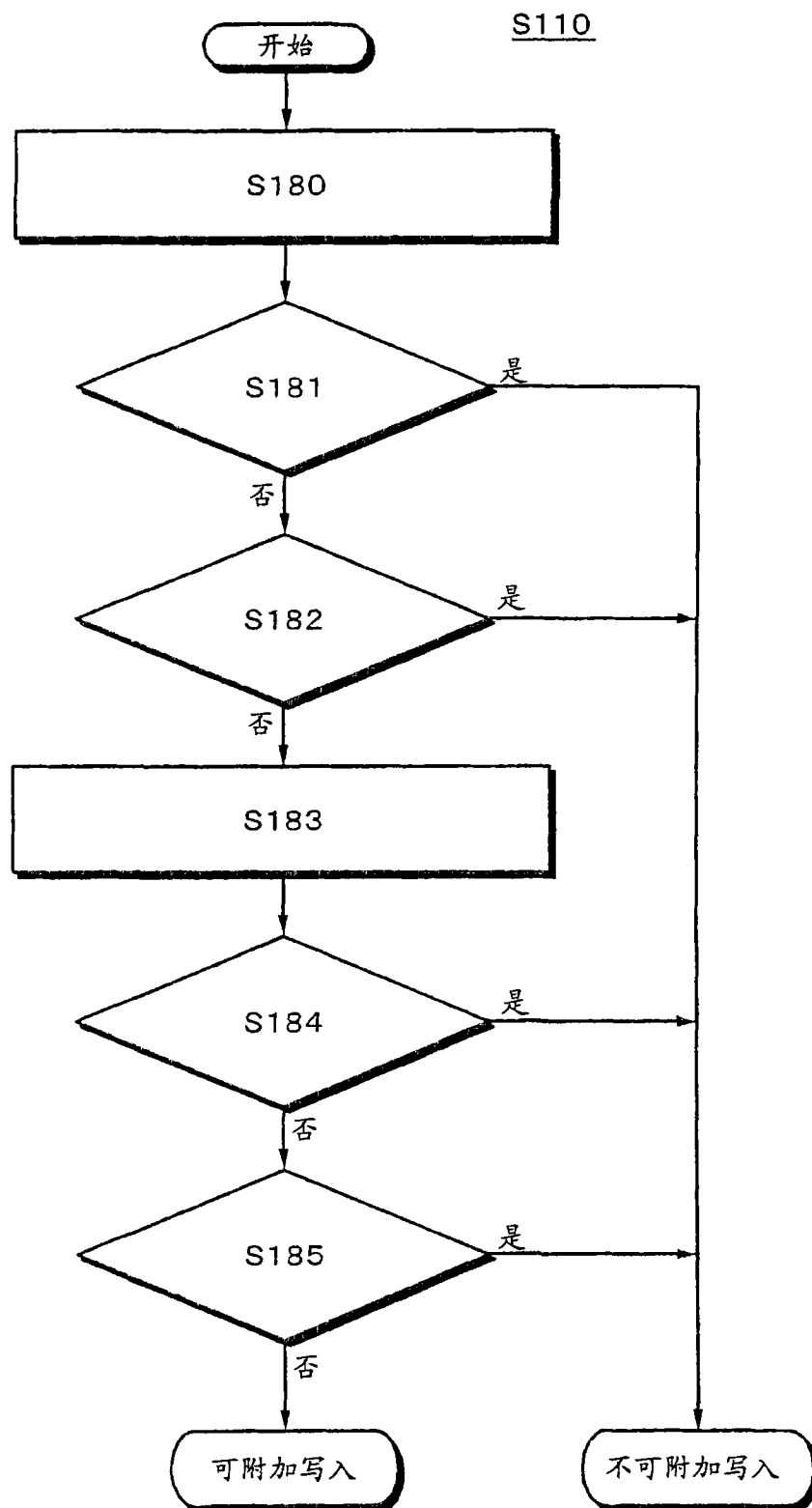


图 50

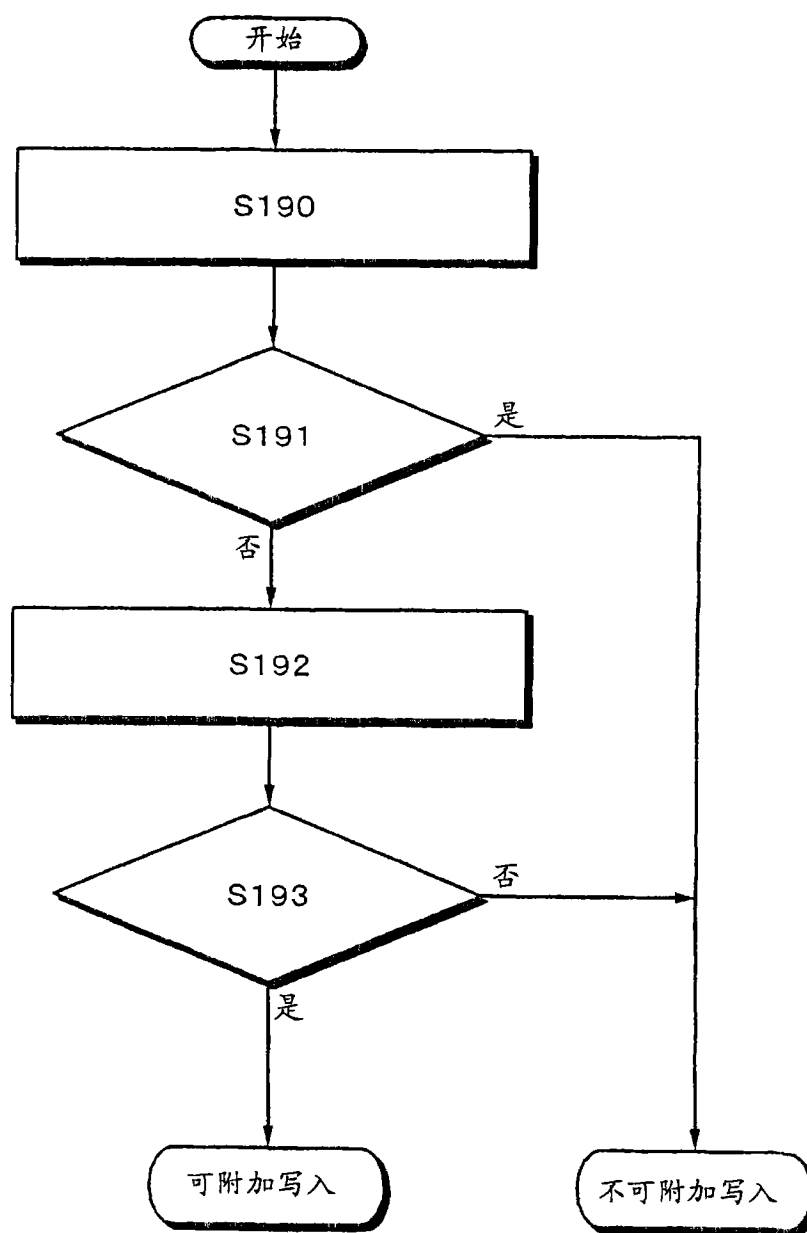
S111

图 51

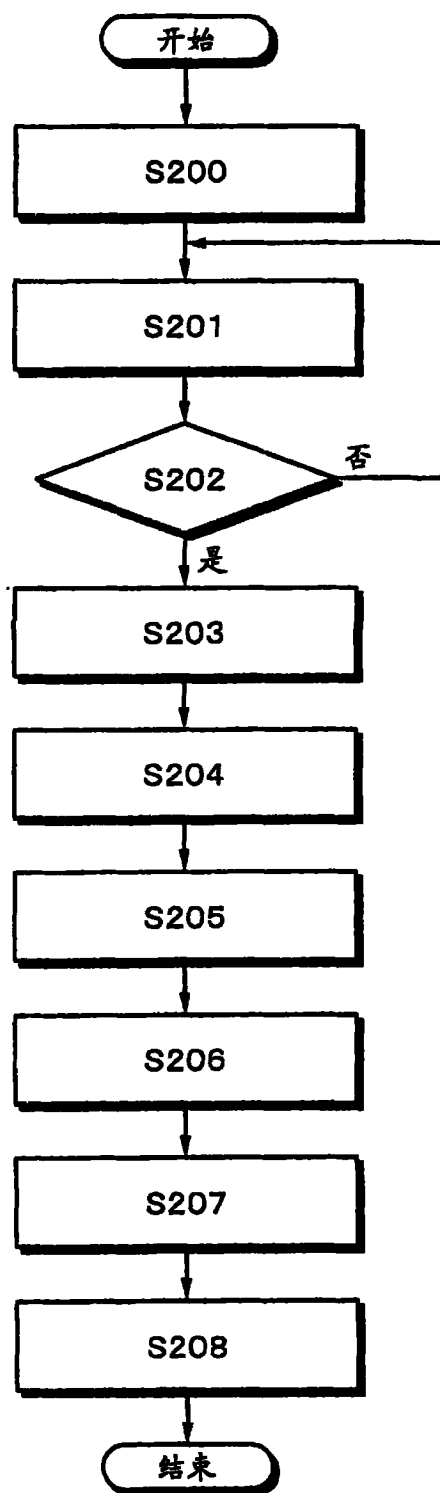


图 52

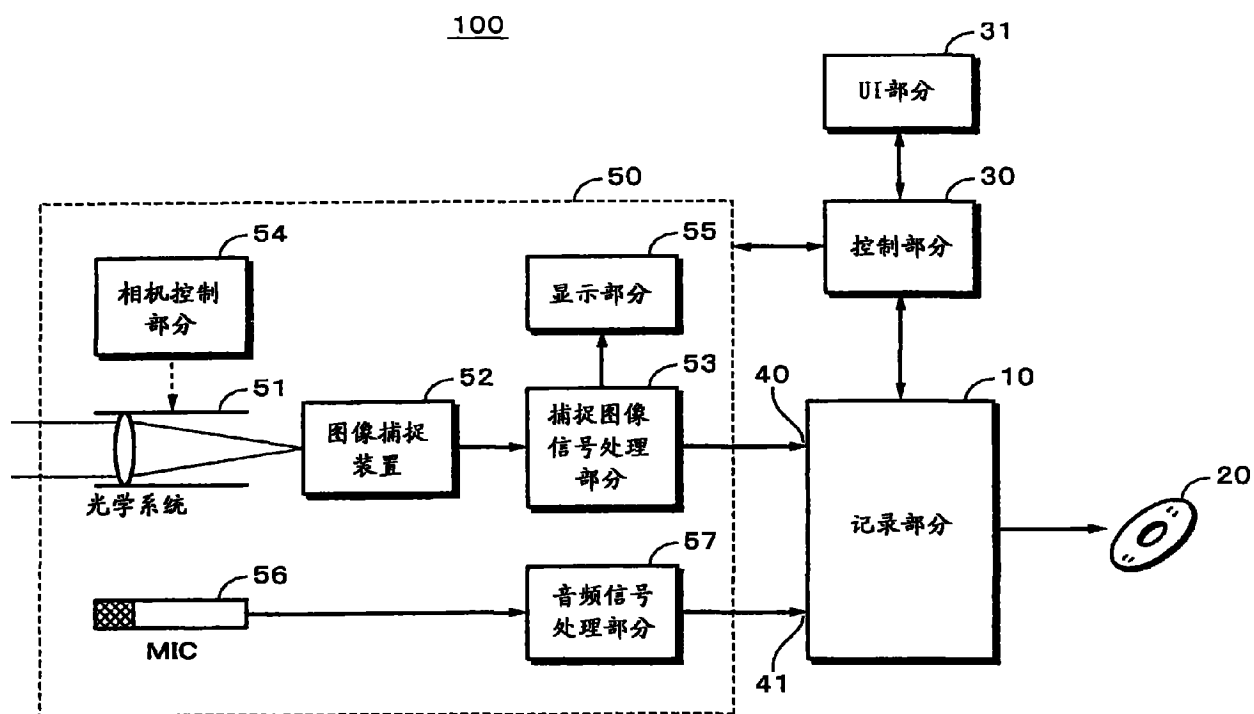


图 53

附图标记的说明

- 10: 记录部分
- 11: 视频编码器
- 12: 音频编码器
- 13: 多路复用器
- 14: 流缓冲器
- 15: 记录控制部分
- 16: 管理信息处理部分
- 17: 非易失性存储器
- 18: 非易失性存储器
- 20: 记录介质
- 30: 控制部分
- 31: 用户接口部分
- 50: 相机部分
- 100: 视频相机设备
- S10: 获取将要记录的数据长度并且将它设置为 ExtDataLength(n+1)的值
- S11: 检查在当前 blkExtensionData()中列出的每个 ext_data_entry()的 ExtDataLength 和 ExtDataStartAddress 以及获取 DataBlock 的使用状态
- S12: 块 DataBlock()具有等于或大于 ExtDataLength(n+1)的连续空闲区域吗
- S14: 决定存储数据的区域的开始地址并将其设置为 ExtDataStartAddress(n+1)

- S15:** 从由 ExtDataStartAddress(n+1)表示的地址写入具有 ExtDataLength(n+1) 的长度的数据
- S16:** 将 ExtDataLength(n+1)和 ExtDataStartAddress(n+1)添加到 ext_data_entry()
- S20:** 从要读取的数据所基于的标准中获取 ExtDataType
- S21:** 基于 ExtDataType 从要读取的数据的类型获取 ExtDataVersion
- S22:** 逐一读取 BlkExtensionData()中列出的 ext_data_entry()
- S23:** 两者的 ExtDataType 和 ExtDataVersion 相匹配?
- S24:** 读取 ExtDataLength(i)和 ExtDataStartAddress(i)
- S25:** 从由 ExtDataStartAddress(i)表示的地址读取具有 ExtDataLength (i)的长度的数据
- S26:** 已经读取 ext_data_entry()?
- S27:** 没有尝试要读取的数据
- S100:** 从介质中读取索引文件
- S101:** 从索引文件的信息中标识作为附加可写候选者的播放列表
- S102:** 读取作为附加可写候选者的播放列表
- S103:** 读取与作为附加可写候选者的播放列表相关联的所有剪辑信息文件
- S104:** 基于播放项目数而执行附加写入能/不能确定
- S105:** 基于播放列表标志数而执行附加写入能/不能确定
- S106:** 基于视频属性而执行附加写入能/不能确定

- S107:** 基于作为附加可写候选者的播放列表文件的文件大小而执行附加写入能/不能确定
- S108:** 基于参照作为附加可写候选者的播放列表文件的剪辑信息文件的总文件大小而执行附加写入能/不能确定
- S109:** 基于参照作为附加可写候选者的播放列表文件的剪辑信息文件中的 EP 的总数而执行附加写入能/不能确定
- S110:** 基于在作为附加可写候选者的播放列表文件中有/没有另一设备的唯一扩展数据而执行附加写入能/不能确定
- S111:** 基于作为附加可写候选者的播放列表文件中的最近更新设备而执行附加写入能/不能确定。
- S112:** 确定至少一个结果是否指示附加可写是不可能的
- S120:** 作为附加可写候选者的播放列表的播放项目数小于上限值?
- S130:** 作为附加可写候选者的播放列表的播放列表标记数小于上限值?
- S140:** 根据当前捕捉/记录模式, 获取记录图像大小、记录纵横比、记录帧速率和记录扫描类型

S141: 从作为附加可写候选者的播放列表的第一播放项目所参考的剪辑信息文件中获取视频的图像宽度

S142: 从作为附加可写候选者的播放列表的第一播放项目所参考的剪辑信息文件中获取视频图像高度和扫描类型

S143: 从作为附加可写候选者的播放列表的第一播放项目所参考的剪辑信息文件中获取视频的纵横比

S144: 从作为附加可写候选者的播放列表的第一播放项目所参考的剪辑信息文件中获取视频的帧速率

S145: 视频属性与作为附加可写候选者的播放列表的视频属性相匹配?

S150: 获取预先计算的播放列表文件每一章的文件大小的增加的字节数。[SIZE_1CHAP=增加的字节数]

S151: 获取作为附加可写候选者的播放列表的播放项目数。

[PI - REMAIN = 上限值 - 播放项目数]

S152: 获取作为附加可写候选者的播放列表的播放列表标记数。[MARK - REMAIN = 上限值 - 播放列表标记数]

S153: 将能够附加写入的播放项目数和播放列表标记数中的较小者视为能够附加写入的章数。[CHAP - REMAIN = MIN (PI - REMAIN, MARK - REMAIN)]

S154: 使用文件系统的功能获取作为附加可写候选者的播放列表的文件大小。

[SIZE_REMAIN=上限值 - 文件大小]

S155: $SIZE - 1CLIP \times CHAP_REMAIN < SIZE_REMAIN?$

S160: 获取预先计算的每一剪辑信息文件的最大尺寸。

[SIZE_1CLIP = 最大尺寸]

S161: 获取参照作为附加可写候选者的播放列表文件的剪辑信息文件的文件大小并且计算所参照的剪辑信息文件的字节总数。 [SIZE_TOTAL_CLIP = 字节总数]

S162: $CLIP_SUM_MAX - SIZE_TOTAL_CLIP > SIZE_1CLIP?$

S170: 获取预先计算的 EP 的每一章的最大数。

[NEEDED_EP_COARSE, NEEDED_EP_FINE]

S171: 参照作为附加可写候选者的播放列表的剪辑信息文件的 EP 并且计算参考的 EP 的总数。

[TOTAL_EP_COARSE, TOTAL_EP_FINE]

S172: $MAX_EP_COARSE - TOTAL_EP_COARSE > NEEDED_EP_COARSE?$

S173: $MAX_EP_FINE - TOTAL_EP_FINE > NEEDED_EP_FINE?$

S180: 在作为附加可写候选者的播放列表文件的扩展数据中搜索 AVCDH 格式的扩展数据

S181: 有 AVCHD 格式的扩展数据吗?

S182: 除了 AVCDH 格式的扩展数据之外还有扩展数据吗?

S183: 在 AVC 的标记唯一扩展数据的扩展数据中搜索该设备的数据。

S184: 没有该设备的数据?

S185: 除了该设备的数据外还有数据?

S190: 基于作为附加可写候选者的播放列表的扩展数据而检查是否存在 AVCHD 格式的扩展数据。

S191: 没有 AVCHD 的扩展数据?

S192: 基于作为附加可写候选者的播放列表的扩展数据而检查最近更新设备

S193: 该设备是最近更新设备?

S200: 执行记录开始操作。

S201: 将流记录到记录介质中。

S202: 记录停止操作?

S203: 将在流缓冲器中存储的流写入到记录介质。

S204: 创建剪辑信息文件。

S205: 设置 ConnectionCondition = 5。

S206: NumberOfPlayItems + 1

S207: 设置 ClipInformationFileName、INTime 和 OUTTime 并创建 blkPlayItem()

S208: 将 MarkTimeStamp 设置为 INTime 并使 blkPlayListMark 加 1。