[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G06F 12/00 (2006.01)
G11B 27/00 (2006.01)

[12] 发明专利说明书

专利号 ZL 01808483.4

[45] 授权公告日 2008 年 10 月 15 日

[11] 授权公告号 CN 100426255C

[22] 申请日 2001.2.22 [21] 申请号 01808483.4

[30] 优先权

[32] 2000.2.28 [33] JP [31] 50428/00

[86] 国际申请 PCT/JP2001/001309 2001.2.22

[87] 国际公布 WO2001/063419 日 2001.8.30

[85] 进入国家阶段日期 2002.10.23

[73] 专利权人 夏普公司

地址 日本大阪

[72] 发明人 岩野裕利 池田奈津子 木山次郎
西村元秀 山村博幸 山口孝好

[56] 参考文献

CN1229485A 1999.9.22

JP8-95835A 1996.4.12

JP63-163642A 1988.7.7

CN1208891A 1999.2.24

审查员 王学睿

[74] 专利代理机构 中国专利代理(香港)有限公司

代理人 杨 凯 王忠忠

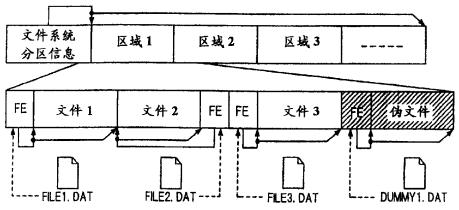
权利要求书 2 页 说明书 16 页 附图 12 页

[54] 发明名称

文件管理方法

[57] 摘要

从多个应用存取的一个记录媒体通常被划分成多个区域。当采用分区控制这种区域划分时，分区的大小难以更改，从而妨碍了记录媒体的有效利用。记录媒体被划分成多个区域，而各个区域中未使用的区域用伪数据写入，该伪数据只能被特定应用或特定文件盖写，从而使得容易划分成多个区域以及容易更改大小。



1. 一种文件管理方法，用于保护记录媒体上要存储数据的数据区和要存储管理信息的区域，并且将所述数据作为文件来管理，它包括以下步骤:

保护所述记录媒体上用于存储第一管理信息的第一区域，以及

把所述第一区域中未使用的空间设置为伪文件，

其中所述伪文件由文件名来定义，从用来执行将所述第一管理信息写入所述记录媒体的驱动器的角度来看，所述文件名被识别为空白区。

2. 如权利要求 1 所述的文件管理方法，其特征在于，从用来执行将所述数据写入所述记录媒体的驱动器的角度来看，所述文件名被识别为占用区。

3. 如权利要求 1 所述的文件管理方法，其特征在于，在所述记录媒体上保护用于存储管理信息的多个区域，用于存储所述管理信息的所述各区域的伪文件的文件名对于不同区域是不相同的。

4. 如权利要求 1 所述的文件管理方法，其特征在于，所述第一管理信息是关于所述数据的应用管理信息。

5. 如权利要求 1 所述的文件管理方法，其特征在于，所述第一管理信息包括关于所述数据的应用管理信息以及读出所述应用管理信息所需的文件系统管理信息。

6. 如权利要求 1 所述的文件管理方法，其特征在于，所述第一管理信息是用于管理目录层次结构的文件管理信息。

7. 如权利要求 1 所述的文件管理方法，其特征在于，所述伪文件的文件管理信息位于所述第一区域中。

8. 如权利要求 1 所述的文件管理方法，其特征在于还包括以下步骤:

当出现对于写入所述第一管理信息的请求时，将所述伪文件的大

小与所请求写入的所述第一管理信息的大小进行比较，以及

当所述伪文件的大小不够写入所述第一管理信息时，通过将邻接所述第一区域的相邻未使用区域包含在内来扩大所述伪文件。

9. 如权利要求 1 所述的文件管理方法，其特征在于还包括以下步骤：

当出现对于写入所述第一管理信息的请求时，将所述伪文件的大小与所请求写入的所述第一管理信息的大小进行比较，以及

当所述伪文件的大小不够写入所述第一管理信息时，把与所述第一区域分开的非相邻未使用区域设置成非相邻伪文件，

其中所述非相邻伪文件由与所述伪文件相关的文件名来定义。

10. 如权利要求 8 所述的文件管理方法，其特征在于，所述相邻未使用区域是通过将所述数据或第二管理信息转移来产生的。

11. 如权利要求 1 所述的文件管理方法，其特征在于，所述第一区域的尾端始终由所述伪文件构成。

12. 如权利要求 1 所述的文件管理方法，其特征在于，所述第一区域的前端和尾端始终由所述伪文件构成。

13. 一种文件管理方法，用于保护记录媒体上要存储数据的数据区和要存储管理信息的区域，并且将所述数据作为文件来管理，它包括以下步骤：

保护所述记录媒体上用于存储第一管理信息的第一区域，以及

把所述第一区域中未使用的空间设置为伪文件，

其中所述伪文件的文件管理信息具有对于预定应用指定隐藏文件或写保护文件的属性。

14. 如权利要求 13 所述的文件管理方法，其特征在于，所述预定应用是 PC 应用。

文件管理方法

技术领域

本发明涉及一种文件管理方法，当要把不同应用的文件写入记录媒体时，通过该方法把单个记录媒体划分成多个区。

背景技术

如果有基于不同平台、如 PC 用途和 AV 用途的可以通用的多用途盘媒体，会为用户带来显著的便利。例如，如果利用 AV 盘记录器记录的盘可以从连接到 PC 的盘驱动器来存取，而且如果可以容易地进行逆向存取，则意味着 AV 盘记录器记录的 AV 数据可以从 PC 进行存取或编辑，而编辑后的产物等可以容易地被 AV 盘记录器等复制。再者，单个盘可以通用于 AV 用途和 PC 用途，使得在单个盘上可以记录 AV 数据而且可以存储用于 PC 的应用软件。

但是，为 AV 用途和 PC 用途记录的数据之间有特征上的差异。例如，当复制盘上记录的 AV 数据时，需要在预定的定时从盘中读出 AV 数据，并将其显示在显示器上。如果无法在预定的定时再现该数据，则意味着再现视图的画面会变得不连续，从而导致异常，这是不可接受的。

因为盘媒体在随机存取性方面具有优势，所以不必将一连串的数据安排在盘上的连续位置中，而是可以通过有效利用盘上的空白区将数据分成各部分来记录。例如，对于 PC 用途，如果文档数据文件被记录在盘上分散的位置，在读取文件期间在数据断开位置上会发生寻道或轨迹跳转，而这时停止从盘读取数据。但是，即使这种配置由于某种原因而需要比可连续读出数据的情况要长一些的数据读取时间，在功能方面也完全不会发生任何问题。

但是，对于上述的 AV 用途来说，如果要再现的数据在盘上是不连续的，在不连续的位置上，在数据读取期间会产生中断，这可能导致问题。一般，从盘读出的 AV 数据一次存储在缓冲存储器中到某种程度，以便利用存储在缓冲存储器中的数据对数据读取时发生的间断(如寻道、轨迹跳转等)进行补偿，从而防止再现画面的中断。虽然缓冲存储器缓解了再现画面的这种不连续情况，但是如果寻道或轨迹跳转的情况经常发生，则无法跟上进度。因此，当在盘上记录 AV 数据时，最好连续记录数据，以免发生导致数据读取时中断的寻道和轨迹跳转。

现在，如果设想一个单个盘为 AV 用途和 PC 用途所共用，每种数据最好记录在彼此不同的区域。如果 PC 用途的数据(在数据量上比 AV 数据小得多)随机地记录在盘区上，其中已经记录了需要连续记录的 AV 数据，这会妨碍 AV 数据的连续记录，可能在某些情况下导致 AV 数据的记录和再现上的困难。

要为 AV 用途而记录的数据不仅包括 AV 数据，而且包括诸如用于再现 AV 数据和静止图像的管理信息文件的各类数据。因为要记录的数据可能用于相同用途，但类型不同，所以出现根据要记录的数据的类型管理分离的区域中数据的需要。记录逻辑文件系统的管理信息的区域、记录用于再现 AV 数据的管理信息等的区域、记录 AV 数据本身的区域以及记录静止图像的区域，均可作为示例来陈述。

为了通过将单个盘划分成多个区域来使用该盘，现有如下方法。首先，就清晰地分区的意义上来说，现有基于逻辑文件系统的分区功能。例如，通过定义 AV 用途和 PC 用途的分区，可以为每一个用途提供专用区。

图 16 中所示实例表示将一个盘分成三个分区、即分区 1、分区 2 和分区 3 的方式。

作为其中未采用分区功能的第二种方法，可以考虑在实现级上将各区作为应用层上的管理信息来管理，而不是由逻辑文件系统来

管理 AV 用途和 PC 用途的区域。例如，其中记录了区域的位置信息的管理信息文件被记录在盘上，使得各区的位置信息可以通过读取该文件来获取。在图 17 所示的实例中，文件“AREA.DAT”中记录了关于区域 1、区域 2 和区域 3 的位置信息。因此，能够理解文件“AREA.DAT”的应用只能获取这些区域的范围。

不同于上述第一方法，这里的区域 1 至 3 属于逻辑文件系统上的单个分区，并在其中被分割。

还可以考虑第三种方法，其中为了保留用于记录文件或数据的区域，创建伪文件，尽管并未记录任何实际数据，但好像该区域已被使用一样。在图 18 所示的区域中，用于待记录文件的区域被三个伪文件 DUMMY1.DAT、DUMMY2.DAT 和 DUMMY3.DAT 保护。

这样，大小等于要记录的数据的伪数据已被预先记录，当写入预定数据类型的文件时，伪数据被擦除，就地写入该文件，从而实现不同数据类型的文件的连续写入。

在上述第一种方法中，可以利用分区功能创建专用于 AV 用途和 PC 用途的分区。但是，一般对于分区来说，需要在盘初始化时确定分区的数量及其大小。因此，在使用过程中不容易更改分区的配置。这是因为每个分区通常都具有独立的逻辑地址和空白区的管理信息，因此如果需要更改配置，则需要重建许多条管理信息。再者，用户还需要在初始化时确定为 AV 用途和 PC 用途分配的大小。但是，随着不断使用，可能会发生任一者变得不够用的情况。

如上所述，因为分区的大小等不容易更改，所以如果例如在用于 PC 用途的区域中有空白区，就不再可能当用于 AV 用途的区域已满时记录 AV 数据，因此造成无法有效利用盘的问题。此外，有些情况中，甚至还需要多个区域来管理用于 AV 用途的同类数据，如记录逻辑文件系统的管理信息的区域、记录用于再现 AV 数据的管理信息的区域、记录 AV 数据的区域以及记录静止图像的区域。在此情况中，如果各个区域被作为分区来管理，则需要构建许多分区。因此，当

一个分区已满时，即使盘上尚有空白区，无法记录更多数据等的概率也会增高。

在第二种方法中，当在实现级上根据应用层上的管理信息管理各区，而不采用逻辑文件系统提供的分区功能来管理时，各区是由特定应用来控制的。因此，其中记录的含有各区的位置数据的管理信息对于该特定应用以外的应用而言是无用的。只要盘仅由指定应用来使用，就不会发生问题。当从该指定应用以外的应用存取这些区时，这些区就无法被识别，因为受保护的区域未建立在逻辑文件系统级。因此，存在非期望数据被写入受保护区域的可能性，从而导致问题。

再者，在第三种方法中，利用伪文件保护某个区域，以便预留允许记录一个文件或数据的区域。在此方法中，例如若需要保护记录管理信息的连续区域，必须预测将来要记录的文件的大小和数量，从而应该记录在大小和数量上相当的伪数据来保护该区域。因此，伪文件最初按预期要记录的文件的最大大小来准备。当实际写入的文件在大小上小于伪文件时，就会发生许多空间被浪费的问题。

在图 18 中所示的区域 1 中，当假定每个要写入的文件的大小是 1 兆字节时，将创建大小为 1 兆字节的伪文件。当写入 0.8 兆字节的文件时，0.8 兆字节的数据首先被写入第一伪文件(DUMMY1.DAT)的位置。当写入 0.9 兆字节的第二文件时，该数据则被写入第二伪文件(DUMMY2.DAT)的位置。这就导致 0.1 兆字节的浪费。再者，如果要写入的文件的大小无法估计(无法预计最大大小)，则有无法处理文件的问题。

因此，本发明的目的是解决上述三种方法中存在的问题，本发明防止所需数据之外的数据被写入预定区域，而且不利用基于逻辑文件系统的分区法来分区。

发明的公开

本发明的第一方面在于用于记录装置的文件管理方法，所述记录装置具有记录媒体和用于管理针对记录媒体的数据输入和输出的记录控制装置，其中，在所述记录媒体中保护至少一个区域，并且每个区域中的空白区由伪文件来管理，所述方法的特征在于，当文件写入该区域时，写入的文件覆盖了所述区域中的伪文件并且伪文件的大小被更新为该区域中空白区的大小。

本发明的第二方面的特征在于，各个伪文件允许从预定应用写入。

本发明的第三方面的特征在于，各个伪文件允许写入预定类型的文件。

本发明的第四方面的特征在于，各个区域的尾端始终由伪块构成。

本发明的第五方面的特征在于，各个区域的前端和尾端始终由伪块构成。

根据本发明，通过创建伪文件以便管理受保护的连续区域内的未使用部分，该区域的整个部分看上去像是正在用于除被指定使用此区域的应用以外的应用，使得有可能禁止写入数据。

此外，通过获取伪文件的文件大小和位置信息，定义伪文件的应用可以容易地获取受保护区域中未使用的空间的大小及其位置，并且容易地在区域内执行更新和文件创建。

当提供多个都可以识别伪文件的不同制造商的应用时，在制造商之间，受保护的连续区域的大小并非始终一致。而且在此情况中，仅参考伪文件信息就能获知未使用的部分，从而可以提供制造商之间的兼容性。

因为不采用作为基于逻辑文件系统的功能的分区功能，所以仅通过更新伪文件即可更改区域大小，是一种比改变分区的大小等的情况更为简单的程序。当需要知道受保护的连续区域的范围时，可以通过将伪文件的一个或多个逻辑块按照需要安排在该连续区域的

尾端或同时安排在尾端和前端来管理该连续区域，从而可以容易地获取正在管理的连续区域的范围。

附图简介

图 1 是说明作为本发明的一个实施例的盘装置的配置的框图；

图 2 是本发明的区域管理方法的实施例的示意图，说明采用伪文件来保护区域的方式；

图 3 是本发明的区域管理方法的实施例的示意图，说明当擦除文件时伪文件的安排；

图 4 是本发明的区域管理方法的实施例的示意图，说明 UDF 管理信息的关系；

图 5 是说明以本发明的区域管理方法的方式保护连续区域的程序的流程图；

图 6 是说明在本发明的区域管理方法的实施例中保护的连续区域内创建文件的程序的流程图；

图 7 是说明在本发明的区域管理方法的实施例中、进行伴随连续区域内保护文件的大小更改的更新时的操作流程的流程图；

图 8 是说明在本发明的区域管理方法的实施例中擦除受保护的连续区域中的文件时的操作流程的流程图；

图 9 是说明在本发明的区域管理方法的实施例中扩充受保护的连续区域时的操作流程的流程图；

图 10 是本发明的区域管理方法的实施例的示意图，表示当扩充受保护的连续区域时的实例；

图 11 是本发明的区域管理方法的实施例的示意图，说明了区域中的伪文件必须管理最后一个逻辑块的方式；

图 12 是本发明的区域管理方法的实施例的示意图，说明了区域中的伪文件必须管理第一和最后一个逻辑块的方式；

图 13 是本发明的区域管理方法的实施例的示意图，说明了区域

中的伪文件必须管理最后一个逻辑块并且将该伪文件的文件入口(file entry)设在第一个逻辑块的方式;

图 14 是说明本发明的区域管理方法的实施例中实际应用实例的示意图;

图 15 是说明本发明的区域管理方法的实施例中、图 12 的实例的目录层次的示意图;

图 16 是说明通过分区保护区域的现有技术方法的示意图;

图 17 是说明通过记录有关各区域的位置信息的文件来进行区域保护的现有技术方法的示意图; 以及

图 18 是说明采用单元伪文件的区域保护的现有技术方法的示意图。

本发明的最佳实施方式

下面参考附图详细地说明本发明的区域管理方法的实施例。在本实施例中，假定盘装置是采用针对 AV 记录和再现的盘、与 PC 等连接的外部存储装置的手持摄像机和录像机。盘媒体最好是可移动型的，但是也可以是安装型的，如硬盘。为便于说明，假定盘所用的逻辑文件系统是基于作为 OSTA(光学存储技术协会)的标准的 UDF(通用盘格式)，但是也可以采用其他通用的逻辑文件系统。

图 1 表示典型盘装置的配置。数据输入/输出部分 1 从摄像机等输入视频信号并将要再现的数据输出到监视器等。数据处理器 2 是处理部分，它执行诸如对 MPEG 码进行编码和解码的信号处理。处理后的数据被存储在存储器 3 中。当记录数据时，盘控制器 5 控制盘 6，使得数据可以被记录在盘上的目标位置。当再现数据时，控制器控制盘 6，使得数据从盘上的目标位置读出并存储在存储器 3 中。各个处理部分由系统控制器 4 控制。

在此盘装置中，当需要将盘划分为多个区域并进行管理时，通常采用基于逻辑文件系统的分区功能。但是，如上所述，一旦盘被

划分成多个分区，就难以更改分区的大小或将一个分区与另一个分区合并来创建一个扩大的分区。因为在此实施例中，区域划分不是通过分区功能执行的，所以形成一种分区，以便于管理盘的整个用户区域，如图 2 所示。用于记录指定类型的信息的区域是在创建的分区内定义的。因为不采用分区功能，而且因为采用了作为通用文件系统之一的 UDF，所以逻辑文件系统级上不存在管理一个分区内的多个区域的管理信息。这里，重要的是：基于多个平台使用一个盘的可能性是存在的。例如，存在这样的可能性：从 PC 的驱动器存取已经为 AV 用途记录的盘。因此，如果仅仅基于可以仅被特定应用、如 AV 应用理解的信息执行区域保护，这是不够好的。换言之，定义用于记录特定类型的信息的区域不仅意味着记录区域的管理是根据 AV 用途的数据类型，而且还限制数据被写入到受保护区域，即使存取是从 PC 应用等进行的。为了实现此目的，从逻辑文件系统的角度来看，受保护区域必须看上去像是它已经记录了数据并正被占用。

为此，在本发明中，为了保护用于记录特定类型的信息的任意数量的连续区域，创建了伪文件来管理区域中未记录任何数据的未使用部分。也就是说，使区域中的未使用部分看上去像是已经写入由伪文件管理的盘上的数据。

在 UDF 中，有卷级上的管理信息和文件系统级上的管理信息。这里，将简要地说明其中涉及的文件系统级的管理信息。在卷级的管理信息所定义的分区内，管理空白区的空间位图描述符、包括分区中目录结构的基本信息和指向管理根目录的文件入口的指针信息的文件集描述符、表示文件集描述符终止的终止描述符以及根目录的文件入口都基本上作为文件系统级的管理信息来记录。

作为空间位图描述符，对分区中管理的每个逻辑块分配一比特，使得各个逻辑块是否被使用由该比特数据的值 0 或 1 来指示。这里，逻辑块是文件系统中的最小可存取单元，而且 UDF 分区内的各个逻辑块被分配按升序排列的逻辑块编号。

一旦定义了目录，管理目录的文件入口将管理目录中包含的文件和对应于该目录的文件标识符描述符集合的记录位置。文件标识符描述符保存了地址信息，其中记录了文件或目录的名称和对应于属性数据的文件入口。文件入口管理诸如日期等的属性数据以及该文件(文件入口)管理的数据在盘上的位置信息，或者(若是目录的话)记录目录信息(在文件标识符描述符的情况中)之处的位置信息。由此，可以通过跟踪文件入口和文件标识符描述符来获取目录层次结构。这些信息是 UDF 文件系统的管理信息，通常仅供文件系统驱动器使用，使得用户通常无法看到它们。图 4 说明了代表 UDF 中目录层次结构的文件入口与文件标识符描述符之间的关系。

作为采用伪文件的区域保护的实例，图 2 示出一种方法，将区域 1、2 和 3 定义为用于记录任意条数的特定类型的信息的连续区域，并将文件 FILE1.DAT、FILE2.DAT 和 FILE3.DAT 记录在连续区域 2 中。在实际的盘中，记录作为对应于 FILE1.DAT 的 UDF 管理信息的 FE(文件入口)和数据本身或文件 1。在本实例中，未示出包括指向作为文件管理信息的 FE(文件入口)的指针信息的文件标识符描述符，假定它记录在例如区域 1 中。

盘上的位置信息由分配描述符集合管理，这些描述符各指示起始逻辑块编号和逻辑块的数量。即，当已经连续记录单个文件(文件入口)所管理的数据时，文件的位置信息由一个分配描述符来管理。当数据记录在分离的两个部分时，文件的位置信息由两个分配描述符来管理。

与 FILE1.DAT 类似，对应于 FILE2.DAT 的 FE(文件入口)及其实际数据被记录在受保护的连续区域。在此阶段，为该受保护的连续区域内未记录数据的未使用部分创建伪文件 DUMMY1.DAT。即，形成伪文件 DUMMY1.DAT，好像是数据已被写入该未使用的区域。这个伪文件由 FE(文件入口)的分配描述符管理。作为 UDF 文件系统管理信息在伪文件 DUMMY1.DAT 所管理的部分中管理空白区的空

间位图描述符被当作正在使用来处理。

在使用过程中，例如，如果擦除图 2 中的 FILE2.DAT，其中记录对应于 FILE2.DAT 的 FE(文件入口)和数据的区域由它们的空间位图描述符指示为正在未使用中，并且变成被释放。但是，在本发明中，擦除数据的部分将不被释放，而当作正在被上述伪文件 DUMMY1.DAT 管理来处理，如图 3 所示。因此，DUMMY1.DAT 的 FE(文件入口)中的分配描述符被更新，使得 DUMMY1.DAT 由两个分离的部分构成。

当创建伪文件时，伪文件的名称应该被定义成可被使用该受保护的连续区域的文件系统的驱动器所理解。在附图的实例中，受保护的连续区域中的伪文件被定义为 DUMMY1.DAT。从对应于本发明的文件系统驱动器的角度来看，此 DUMMY1.DAT 被识别为空白区。

因此，通过总是创建伪文件，以便控制受保护的连续区域内的未使用部分，该区域的整个部分将被看作正在被除指定使用此区域的应用以外的应用所使用，使得不写入任何数据。再者，此配置不仅禁止其他应用写入数据，而且使得定义此伪文件的文件系统驱动器有可能获取有关伪文件的信息，从而容易获取受保护区域中的未使部分的大小和位置。例如，可以通过参考伪文件的文件大小来获知未使用部分的大小，而且可以通过参考伪文件的 FE(文件入口)中的分配描述符来容易地获取位置信息。简言之，伪文件自身代表受保护区域中的未使用部分。

当提供多个都可以识别伪文件的不同制造商的文件系统驱动器时，在制造商之间，受保护的连续区域在大小上并非始终一致。而且在此情况中，仅参考伪文件信息可以获知受保护区域中的未使用部分，使得可以实现制造商之间的兼容性，而不需要限制区域大小。另外，如果需要，可以从关于要记录到该区域内的一个或多个伪文件的信息中获知区域大小。

现参考图 5 所示的流程图，详细说明保护用于记录特定类型的新信息的指定大小的连续区域的程序。当在步骤 S1 保护连续区域的请求出现时，在步骤 S2，通过参考 UDF 逻辑文件系统的管理信息的空间位图信息，搜索和获取盘上的空白区。在步骤 S3，当无法在盘上的任何地方保护需要的连续区域时，在步骤 S5 执行错误例程且操作结束。在步骤 S3，当可以在盘上某个位置保护所需的连续区域时，在步骤 S4 以这样的方式创建伪文件，使得它管理要保护的连续区域，并且结束操作。具体来说，指向文件入口的伪文件的指针信息、即文件标识符描述符被添加到其上创建了该伪文件的目录的目录信息中，并且记录用于管理该伪文件的数据的位置信息和属性信息的文件入口。此时，作为受保护区域的位置信息的起始逻辑块编号和逻辑块的数量被记录在文件入口中的分配描述符中。

接下来，描述当出现文件创建请求时的程序。当已经产生文件创建请求时，判断该文件可以写入哪个区域。例如，如图 2 所示，有多个划分的区域，即区域 1、区域 2、区域 3…。在此情况中，假定当某个 AV 应用发出文件写入请求时，该文件适合于写入区域 1。例如通过指示文件系统驱动器使 AV 应用写入 DUMMY1.DAT 的位置来作出确定。其他应用、例如除 AV 应用之外的 PC 应用将不会盖写 DUMMY1.DAT，而会写入其他区域。

也可以进行控制，通过根据文件的使用目的来定义(对应于伪文件的)写入区域，使视频数据例如写在 DUMMY1.DAT 上而音频数据写在 DUMMY2.DAT 上。

总而言之，可以实现将预定类型的文件写入预定区域。

接下来参考图 6 所示的流程图，详细说明确定写入区域之后、在伪文件保护的区域内创建文件时的程序。

当在步骤 S10 发出文件创建请求时，在步骤 S11 获取伪文件信息。具体来说，参考与基于其文件名被定义为伪文件的文件对应的文件入口，从该信息中获取有关伪文件的文件大小和位置信息。在

步骤 S12，确定伪文件的文件大小或者受保护区域中的空闲空间大小是否大于要创建的文件的大小。当伪文件的大小小于要创建的文件的大小时，受保护区域中没有空间来创建文件，所以在步骤 S15 执行错误例程并且结束操作。在步骤 S12，当伪文件的大小大于要创建的文件的大小时，则可以在受保护区域中创建该文件，使得在步骤 S13 中，该文件在伪文件所管理的空闲部分中创建。

具体来说，指向管理该文件的文件入口的指针信息、即文件标识符描述符被添加到要创建的文件所属的目录的目录信息中，并且记录用于管理该文件的数据的位置信息和属性信息的文件入口。在步骤 S14，更新伪文件的文件入口以结束该操作。为此，通过从文件入口中的分配描述符信息中扣除为新文件分配的部分，从而进行更新。

接下来，参考图 7 所示的流程图，当利用大小变化来更新属于伪文件所保护的区域的文件时要实施的程序。

在步骤 S20，当发出利用文件大小的变化更新文件的请求时，在步骤 S21，确定该文件的大小是变得更大还是更小。在步骤 S28，当文件大小减小时，要减小的文件的文件入口的分配描述符被更新，以便管理减小的数据。在步骤 S29，通过将文件减小的量加到伪文件的文件入口的分配描述符来执行更新。当文件大小扩充时，在步骤 S22 获取关于伪文件的信息，以便了解受保护区域内的空闲部分。在步骤 23，伪文件的大小或受保护区域中的未使用空间的大小与新近扩充的文件的数据大小进行比较。当伪文件的大小较大，或者受保护区域中的空闲部分足够大时，步骤 S24，扩充的数据被记录到伪文件所管理的未使用空间中。在步骤 S25，扩充文件的文件入口的分配描述符被更新。在步骤 S26，更新伪文件的文件入口的分配描述符以结束该操作。在步骤 S23，如果扩充数据的大小大于伪文件，或者当没有足够大的空闲空间以允许扩充时，则在步骤 S27 执行错误例程以结束操作。

涉及图 8 所示的流程图详情说明的是擦除属于伪文件所管理的区域的文件的情况。在步骤 S30，当出现擦除文件的请求时，在步骤 S31 执行文件的擦除。具体来说，从包括要擦除的文件的目录的目录信息中，擦除相应的文件标识符描述符。在步骤 S32，更新伪文件的文件入口以结束该操作。实际上，更新伪文件的分配描述符，使得该伪文件管理记录过所擦除文件的部分。

参考图 9 所示的流程图，详细说明在伪文件保护的区域不够时需要扩充区域大小的情况下的程序。

在步骤 S40，当出现区域扩充请求时，在步骤 S41 通过空间位图获取盘上的空白区。在步骤 S42，确定是否存在与要扩充的区域毗邻、具有相当于所需扩充量的空间的连续区域。如果有，则在步骤 S44 以这样的方式更新伪文件的文件入口，使得更新的文件包括扩充的区域。如果不存在具有相当于所需扩充量的空间的邻接连续区域，则执行图 9 所示的连续区域保护处理以结束该操作。在此情况中，扩充的区域不与现有的区域相接。如果扩充的区域应该是连续的，则该操作可以通过错误例程来结束，而不执行步骤 S43 的连续区域保护处理。图 10 表示扩充某个区域的实例。在此实例中，FILE1.DAT、FILE2.DAT 和 FILE3.DAT 已被记录在区域 1 中，没有余留空白区，并且由于相邻区域 2 已经被占用，在尝试利用相邻区域扩充该区域无效之后，保护了盘上的一个分开的连续区域。区域 1 中的空白区由文件名为 DUMMY1.DAT 的伪文件来管理，而区域 1 的扩充区域中的空白区由文件名为 DUMMY2.DAT 的伪文件来管理。这里，因为定义了文件名，所以可以根据 DUMMY2.DAT 的存在来获知在盘上分离的位置存在扩充的连续区域的情况。此外，当需要扩充现有的连续区域，但是由于相邻区域正被占用而无法连续扩充时，可以将相邻数据移位到盘上的另一个位置以使得该区域空置，并且执行连续扩充。

这样，因为不采用作为逻辑文件系统的功能的分区功能，所以

有可能仅通过更新伪文件来更改区域大小，这是一种比变更分区的大小等的情况简单得多的程序。

要利用伪文件管理已定义的连续区域中的空白区，还可以增加一个要求：连续区域中的最后一个逻辑块必须由伪文件管理，如图 11 所示。也就是说，在连续区域中最末端的一个逻辑块始终预留为由伪文件管理的空间。因此，除所需的连续区域的大小之外，应该预留一个逻辑块和一个用于记录管理伪文件的文件入口的逻辑块，或者总共多留两个逻辑块。因此，在管理伪文件 DUMMY1.DAT 的文件入口中的分配描述符之中，最大的逻辑块编号将指示受保护的连续区域的尾端。利用这种配置，可以防止连续区域的边界在将数据文件记录到连续区域的最后位置时变得不明确，并且使获取连续区域的边界更容易。

当已确定要在受保护连续区域中记录的文件时，就可以根据文件入口的分配描述符和伪文件的信息了解该连续区域的范围。但是，如果尚未确定记录在连续区域中的文件的名称等，则无法根据文件名判断记录的数据属于哪个连续区域。因此，在这种情况中，可以根据伪文件了解空白区，但无法了解受保护的连续区域的范围。因此，要处理此类情况，也可以增加一个要求：连续区域中的第一个和最后一个逻辑块必须由伪文件管理，如图 12 所示。也就是说，连续区域中第一个和最后一个逻辑块始终预留作由伪文件管理的空间。因此，除所需的连续区域的大小之外，应该预留两个逻辑块和一个用于记录管理伪文件的文件入口的逻辑块，或者总共多预留三个逻辑块。

或者，如图 13 所示，还可以提出一个要求：要保护的连续区域的第一个逻辑块应该是管理伪文件的文件入口，而尾端的一个逻辑块应该是要由伪文件管理的空间。

通过这种配置，管理伪文件 DUMMY1.DAT 的文件入口中的分配描述符中的最小逻辑块编号或者其中记录该伪文件的文件入口本

身的逻辑块的数量将指示受保护的连续区域的起始位置，而最大逻辑块编号将指示该连续区域的末尾。因而，这使得更容易了解正被管理的连续区域的范围。

接下来参考图 14，描述本发明的实际应用实例。此图表示用于记录 AV 数据的 AV 用途的区域管理的实例。首先，通过将整个区域划分为文件系统管理信息区、应用管理信息区和数据区来管理整个区域。记录在文件系统管理信息区中的是基于 UDF 文件系统的基本管理信息，包括空间位图描述符(SBD)、文件集描述符(FSD)和终止描述符(TD)、AV 应用所定义的目录的文件入口(FE)以及文件标识符描述符(FID)，它作为指向管理文件的文件入口和目录中定义的目录的指针信息。在此图的实例中，记录了 ROOT 目录的配置、用于记录 AV 数据的管理信息的 AVMAN 目录、用于管理记录 AV 数据的 AVDAT 目录的文件入口、在不同目录中定义的文件的文件标识符描述符(FID)集合。伪文件 LOGI.DMY 管理文件系统管理信息区中的未使用部分。

应用管理信息区是其中记录 AV 应用的管理信息、如用于再现 AV 数据的管理信息的区域。在此区域中，记录了作为 AV 数据的管理信息的 AVINF01.DAT 和 AVINF02.DAT。对应于每个的文件入口和数据实体被记录在盘上。类似于文件系统管理信息区，伪文件 APPL.DMY 管理应用管理信息区中的未使用部分。当存取这样配置的盘时，用户将看到图 15 所示的目录层次结构。

在数据区中，记录了 AV 数据，即 AV1.DAT 和 AV2.DAT。对应于每个的文件入口和数据实体被记录在盘上。在此图的实例中，虽然对于数据区，不利用伪文件在数据区中作任何区域预留，但是可以根据需要定义伪文件来保护某个区域。

这样，文件系统管理信息区中的未使用部分由伪文件 LOGI.DMY 来管理。应用管理信息区中的未使用部分由伪文件 APPL.DMY 来管理。对于利用这些区域的 AV 应用，这些伪文件被处理为管理相关区

域中的空闲部分。例如，当从与 PC 连接的盘驱动器存取有这样条件的盘时，它看上去像是在文件系统管理信息区和应用管理信息区中已经记录了文件，而未留有任何空白区。这相当于这些区域受到保护。因此，不会有任何数据写入这些区域。

因为若从 PC 应用擦除 AV 应用中使用的目录、有关 AV 数据的管理信息等，就会导致严重的问题，所以可将这些文件和目录的属性指定成正被写保护。即，PC 应用只能写入数据区中的空闲空间。更不用说，如果伪文件本身被擦除，会相当不便，因此可以将文件属性设置为阻止任何数据写入。伪文件可以设置成具有“隐藏”文件属性，以便使它的文件名不出现在 PC 应用上。如果伪文件意外地被擦除，则可以根据有关当前定义的文件和目录的逻辑文件系统管理信息以及有关管理空白区的空间位图描述符的信息来重构伪文件。

工业实用性

如上所述，根据本发明的文件管理方法，由于它可以将所需数据之外的数据写入预定区域，而不用执行基于逻辑文件系统的分区操作，所以该方法可以应用于诸如 PC 用途和 AV 用途的不同平台所通用的多用途盘媒体。

图 1

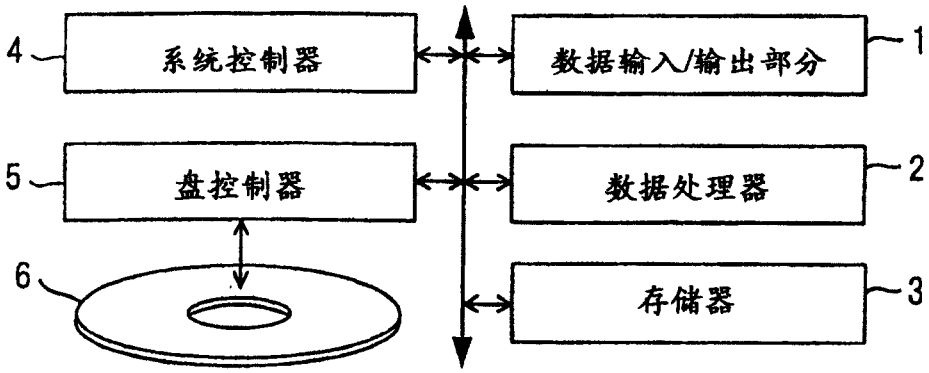
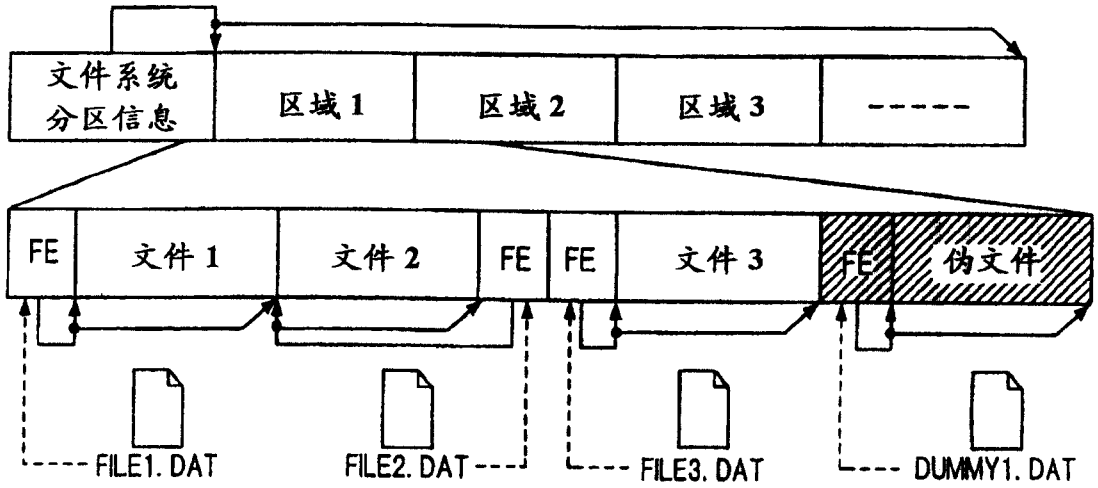
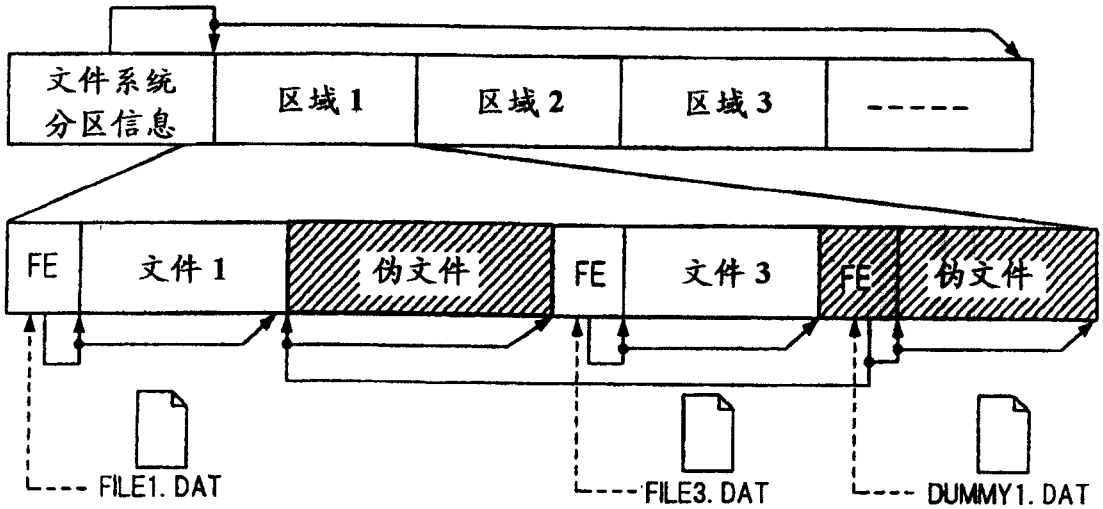

4
系统控制器
数据输入/输出部分
1
5
盘控制器
数据处理器
2
6
存储器
3

图 2


文件系统
分区信息
区域 1
区域 2
区域 3

FE
文件 1
文件 2
FE
FE
文件 3
FE
伪文件
FILE1. DAT
FILE2. DAT
FILE3. DAT
DUMMY1. DAT

图 3


文件系统
分区信息
区域 1
区域 2
区域 3

FE
文件 1
伪文件
FE
文件 3
FE
伪文件
FILE1. DAT
FILE3. DAT
DUMMY1. DAT

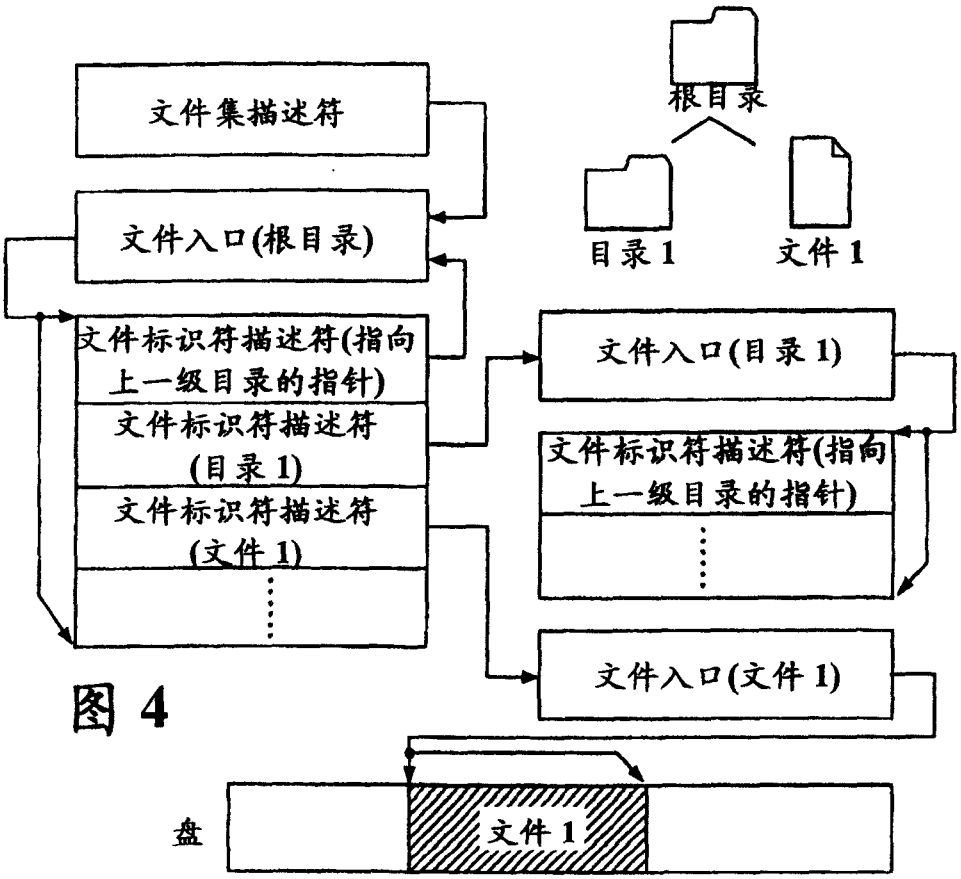

文件集描述符
文件入口(根目录)
文件标识符描述符(指向上一级目录的指针)
文件标识符描述符(目录 1)
文件标识符描述符(文件 1)
根目录
目录 1
文件 1
文件入口(目录 1)
文件标识符描述符(指向上一级目录的指针)
文件入口(文件 1)
盘
文件 1

图 4

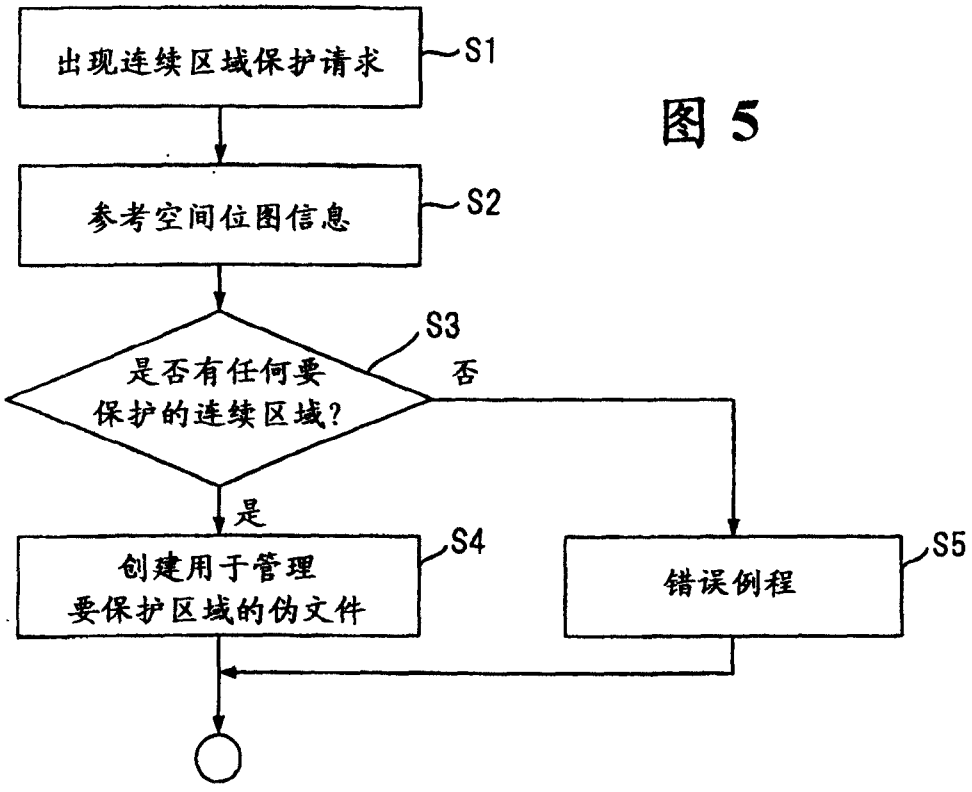

出现连续区域保护请求
S1
参考空间位图信息
S2
是否有任何要保护的连续区域?
S3
否
是
创建用于管理要保护区域的伪文件
S4
错误例程
S5

图 5

图 6

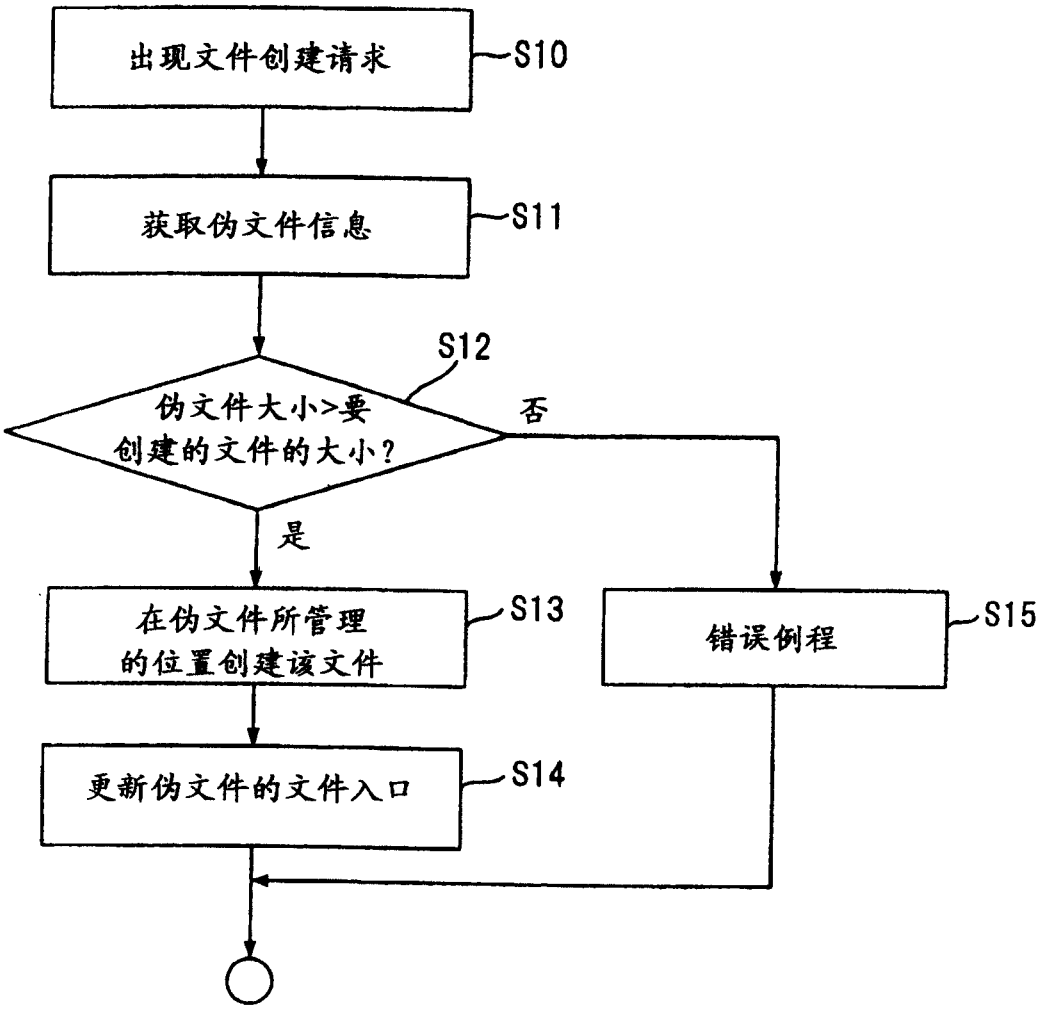

出现文件创建请求
S10
获取伪文件信息
S11
S12
伪文件大小>要创建的文件的大小?
否
是
在伪文件所管理的位置创建该文件
S13
更新伪文件的文件入口
S14
错误例程
S15

图 7

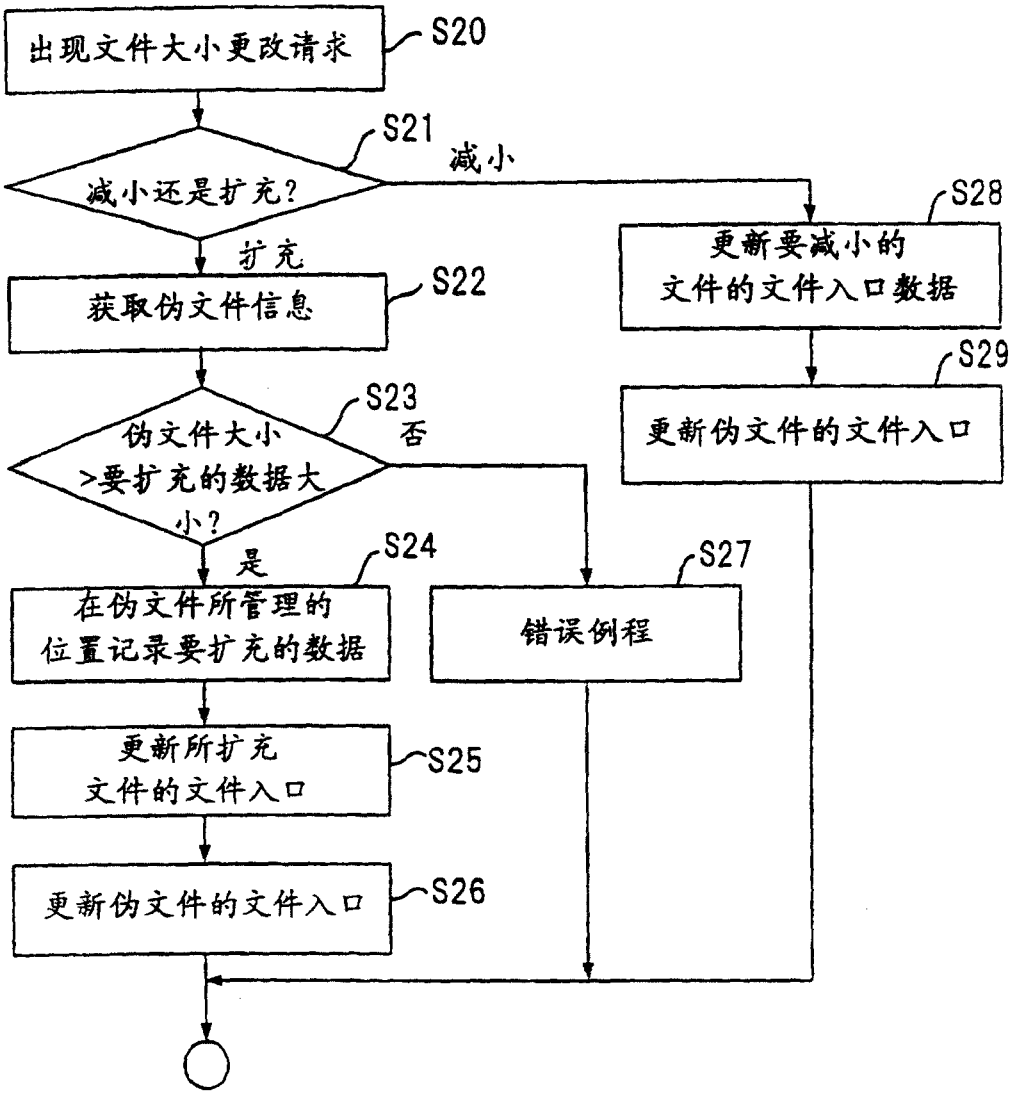

出现文件大小更改请求
S20
S21
减小还是扩充?
减小
扩充
S22
获取伪文件信息
S23
伪文件大小
>要扩充的数据大
小?
否
是
S24
在伪文件所管理的
位置记录要扩充的数据
S25
更新所扩充
文件的文件入口
S26
更新伪文件的文件入口
S27
错误例程
S28
更新要减小的
文件的文件入口数据
S29
更新伪文件的文件入口

图 8

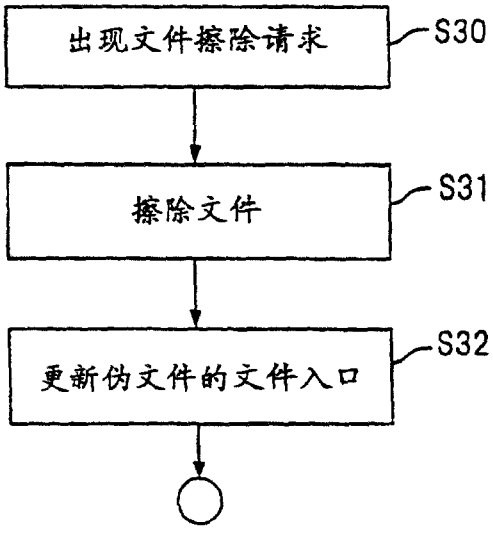

出现文件擦除请求
S30
擦除文件
S31
更新伪文件的文件入口
S32

图 9

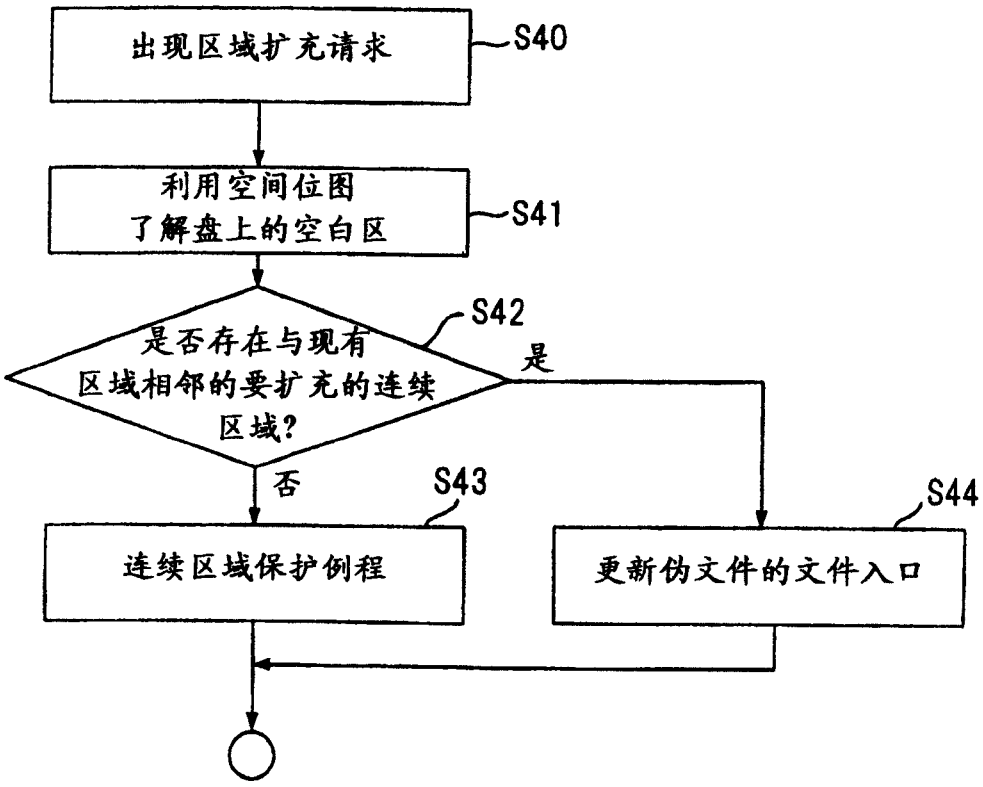

出现区域扩充请求
S40
利用空间位图
了解盘上的空白区
S41
是否存在与现有
区域相邻的要扩充的连续
区域?
S42
是
否
S43
S44
连续区域保护例程
更新伪文件的文件入口

图 10

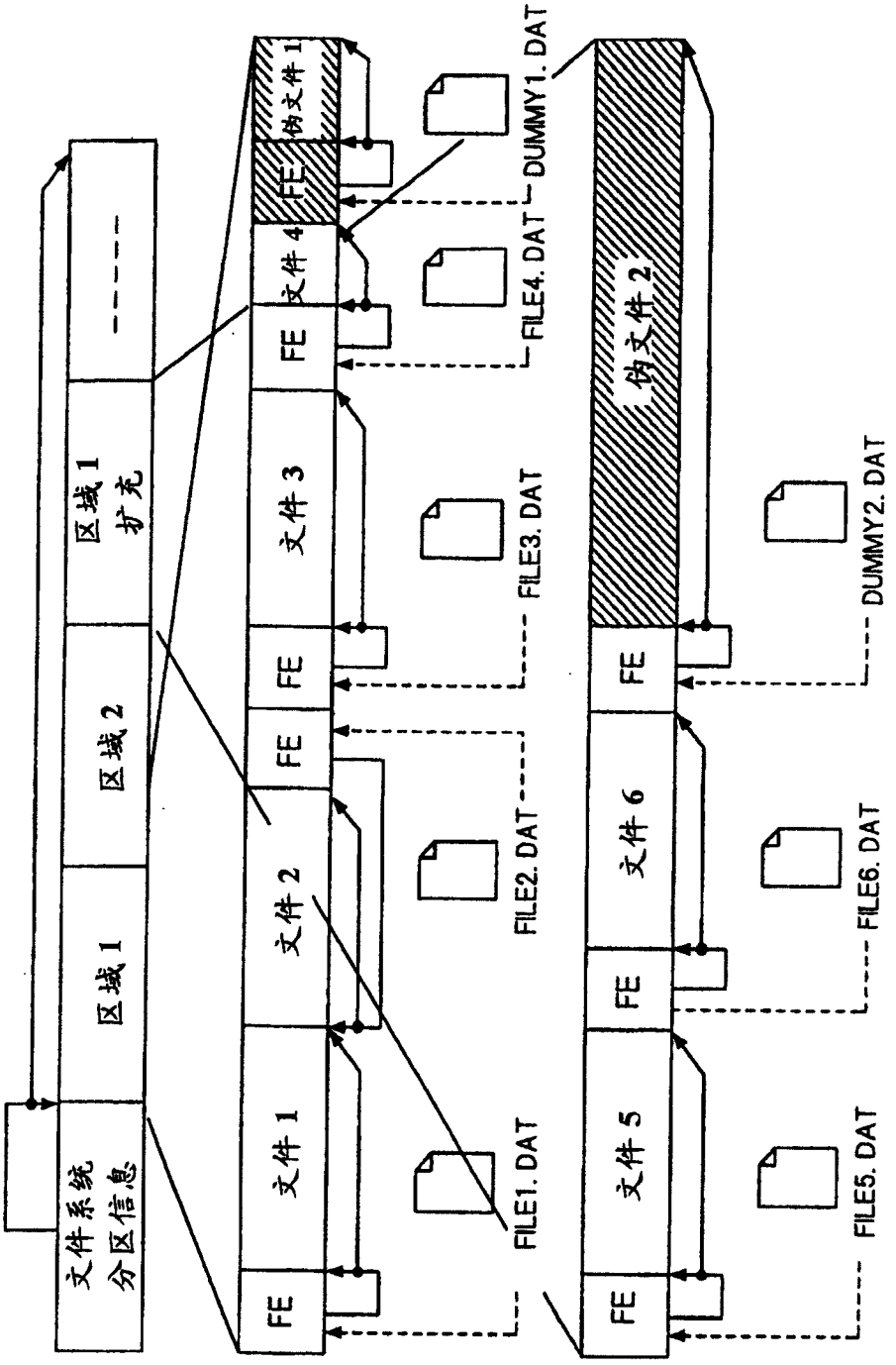

文件系统
分区信息
区域 1
区域 2
区域 1
扩充
FE
文件 1
文件 2
FE
FE
文件 3
FE
文件 4
FE
伪文件 1
FE
文件 5
FE
文件 6
FE
伪文件 2
FILE1. DAT
FILE2. DAT
FILE3. DAT
FILE4. DAT
DUMMY1. DAT
FILE5. DAT
FILE6. DAT
DUMMY2. DAT

图 11

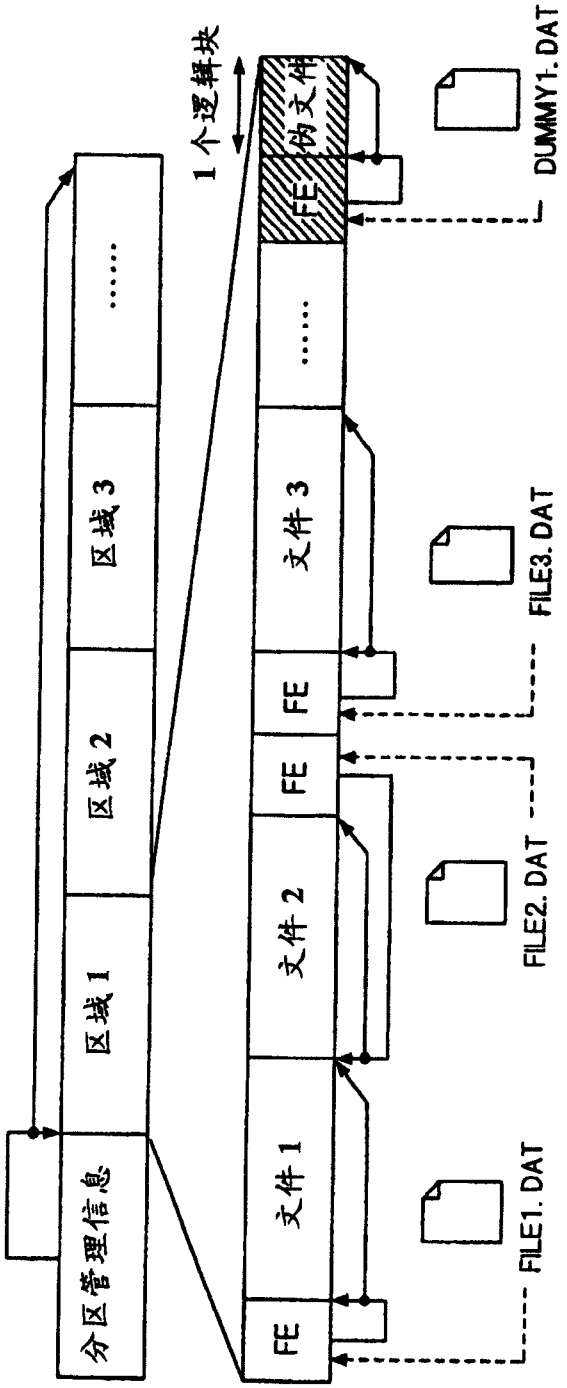

分区管理信息
区域 1
区域 2
区域 3
.....
1 个逻辑块
FE
文件 1
FE
文件 2
FE
文件 3
FE
伪文件
.....
FILE1. DAT
FILE2. DAT
FILE3. DAT
DUMMY1. DAT

图 12

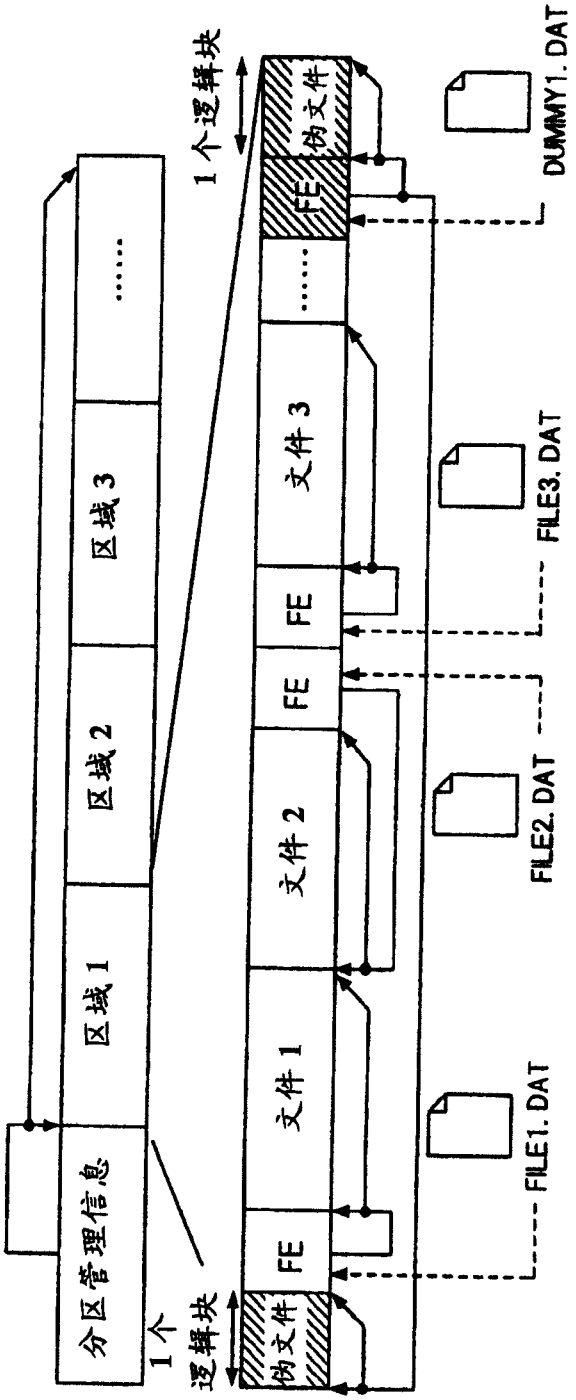

分区管理信息
区域 1
区域 2
区域 3
.....
1个逻辑块
伪文件
FE
文件 1
文件 2
文件 3
FE
FE
FE
伪文件
.....
1个逻辑块
FILE1.DAT
FILE2.DAT
FILE3.DAT
DUMMY1.DAT

图 13

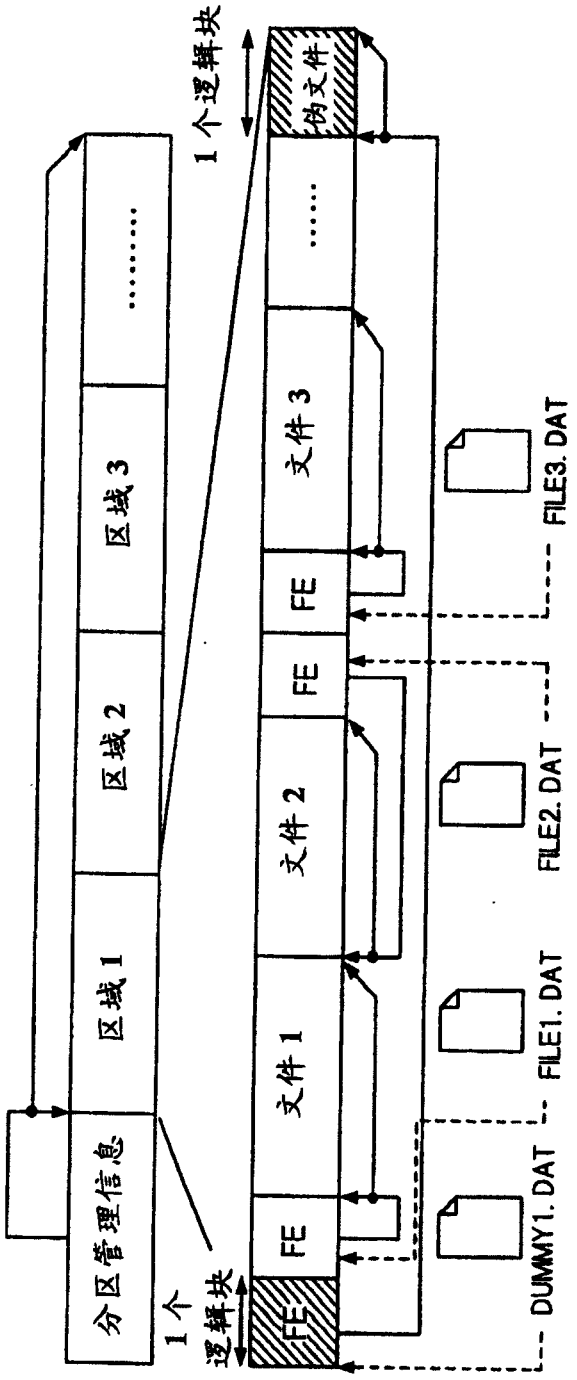

分区管理信息
区域 1
区域 2
区域 3
.....
1个逻辑块
FE
文件 1
文件 2
文件 3
伪文件
DUMMY1. DAT
FILE1. DAT
FILE2. DAT
FILE3. DAT

图 14

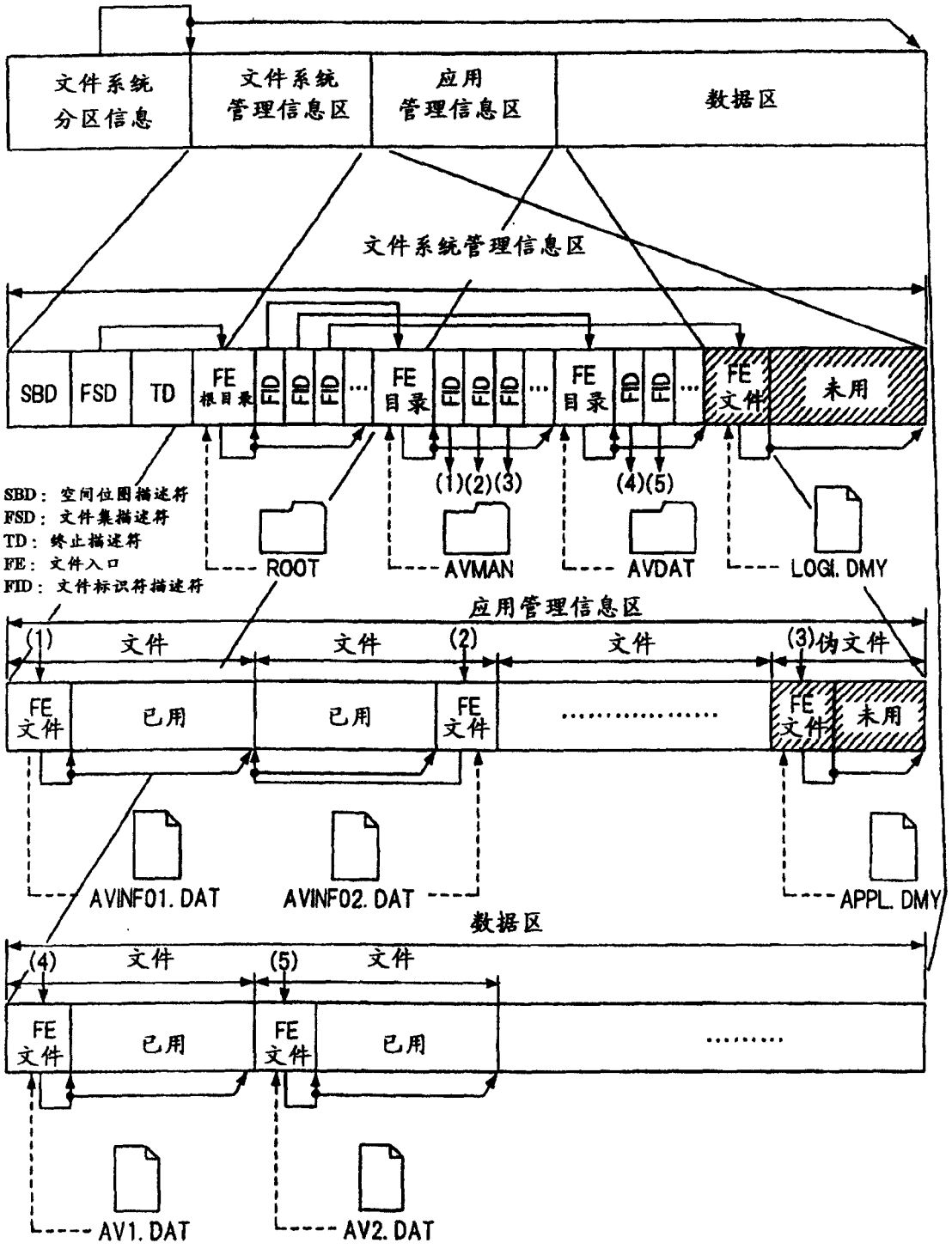
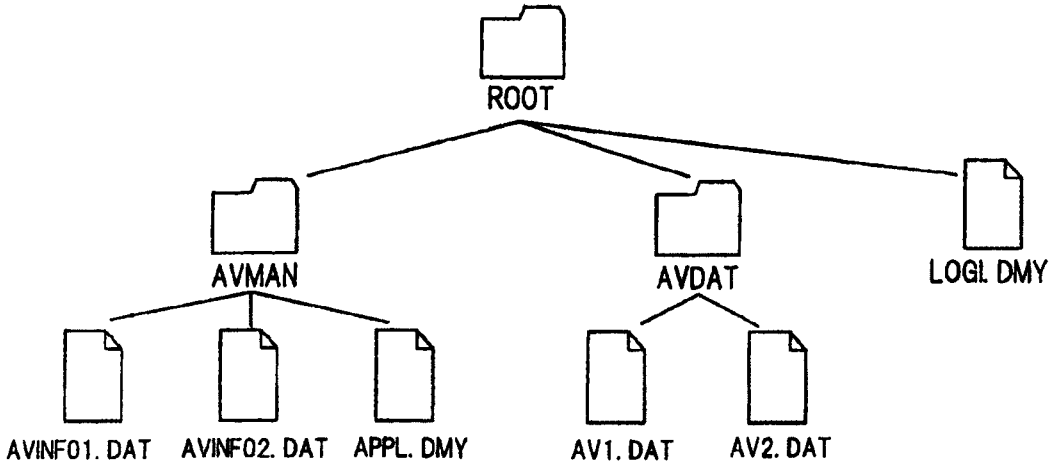

文件系统
分区信息
文件系统
管理信息区
应用
管理信息区
数据区
文件系统管理信息区
SBD
FSD
TD
FE
根目录
FE
目录
FE
目录
FE
文件
未用
(1)(2)(3)
(4)(5)
SBD：空间位图描述符
FSD：文件集描述符
TD：终止描述符
FE：文件入口
FID：文件标识符描述符
ROOT
AVMAN
AVDAT
LOGI. DMY
应用管理信息区
(1)
文件
文件
(2)
文件
(3)伪文件
FE
文件
已用
已用
FE
文件
FE
文件
未用
AVINF01. DAT
AVINF02. DAT
APPL. DMY
数据区
(4)
文件
(5)
文件
FE
文件
已用
FE
文件
已用
AV1. DAT
AV2. DAT

图 15


ROOT
AVMAN
AVDAT
LOGI. DMY
AVINF01. DAT
AVINF02. DAT
APPL. DMY
AV1. DAT
AV2. DAT

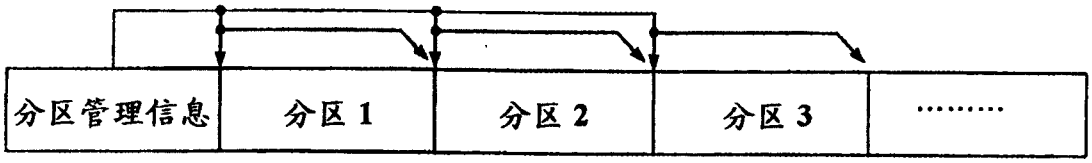
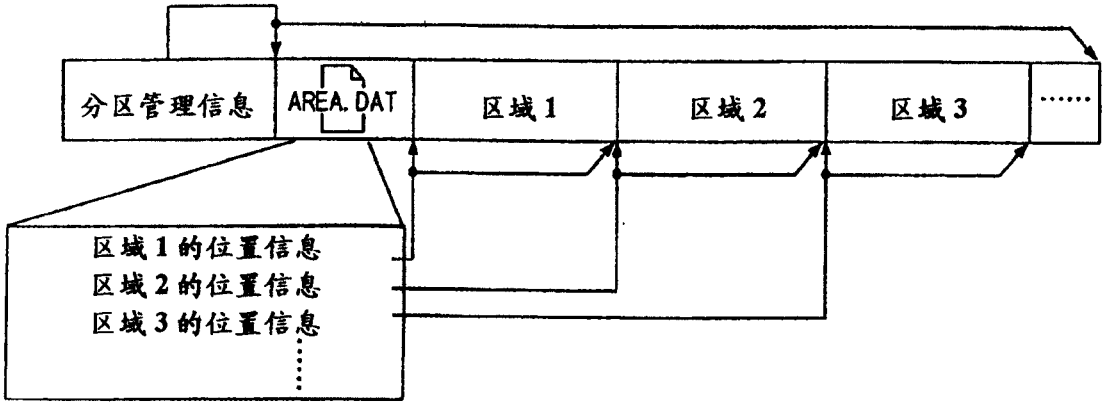

分区管理信息
分区 1
分区 2
分区 3
.....

图 16

图 17


分区管理信息
AREA.DAT
区域 1
区域 2
区域 3
区域 1 的位置信息
区域 2 的位置信息
区域 3 的位置信息

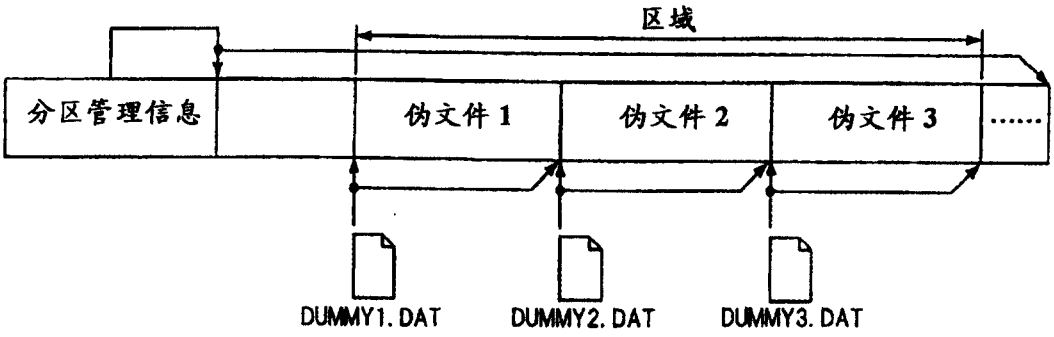

区域
分区管理信息
伪文件 1
伪文件 2
伪文件 3
DUMMY1.DAT
DUMMY2.DAT
DUMMY3.DAT

图 18