

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.  
G06F 12/06 (2006.01)  
G06F 12/08 (2006.01)



# [12] 发明专利申请公布说明书

[21] 申请号 200710069256.8

[43] 公开日 2007 年 11 月 21 日

[11] 公开号 CN 101075211A

[22] 申请日 2007.6.8

[21] 申请号 200710069256.8

[71] 申请人 马彩艳

地址 310012 浙江省杭州市文二西路 176 号  
恩济花苑 2-3-402 室

[72] 发明人 马彩艳

[74] 专利代理机构 杭州杭诚专利事务所有限公司  
代理人 王鑫康

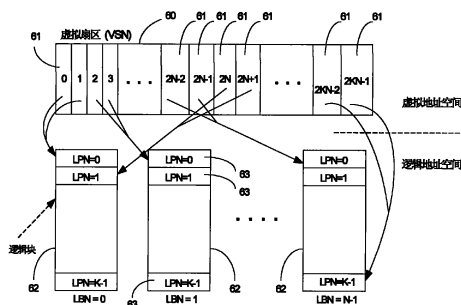
权利要求书 7 页 说明书 21 页 附图 8 页

## [54] 发明名称

基于 sector 访问的 flash 存储器的存储管理

## [57] 摘要

本发明提供基于 sector 访问的 flash 存储器的存储管理方法，把连续的虚拟扇区均匀地分散映射到各个逻辑块中，建立逻辑块和物理块的映射，并且建立该逻辑块内逻辑页和对应物理块内物理页的映射，使得虚拟扇区分散地映射到了各个物理块中，实现了各物理块的均衡使用和满足了对物理块内物理页的顺序写要求，提供了虚拟地址空间的多次读写能力。同时 flash 存储器的访问接口和磁盘兼容，使整个应用系统的构建简便、通用性好。本发明实现系统对 flash 上数据更改的可靠性高，flash 空间利用率高，能有效延长 flash 使用寿命，适用于各型 flash 以及其他与 flash 有类似操作特性的设备。



1. 基于 sector 访问的 flash 存储器的存储管理，其存储管理方法包括下列步骤：

(1) 定义 Flash 存储器的物理空间

a. 首先根据 flash 存储器的物理结构，将其存储器划分为一个以上的块，称之为物理块，各物理块的大小相同，按照物理块起始地址从小到大的顺序给每个物理块指定一个编号 PBN（物理块号）；

b. 然后把每个物理块划分为一个以上的页，称之为物理页，各物理页的大小相同；物理块中的每个物理页按照它在物理块内起始偏移地址从小到大的顺序给每个物理页指定一个在该物理块内唯一的编号 PPN（物理页号）；

c. 把由所有物理块组成的空间称为物理空间；

(2) 定义一个逻辑存储区

a. 将其划分为一个以上的块，称之为逻辑块，各逻辑块的大小相同，并且逻辑块的大小和物理块的大小相同，每个逻辑块都有一个逻辑地址，按照逻辑块在逻辑空间中的起始地址从小到大的顺序给每个逻辑块指定一个编号 LBN（逻辑块号）；

b. 然后把每个逻辑块划分为一个以上的页，称之为逻辑页，各逻辑页的大小相同，并且逻辑页的大小和物理页的大小相同；逻辑块中的每个逻辑页按照它在逻辑块内起始偏移地址从小到大的顺序给每个逻辑页指定一个在该逻辑块内唯一的编号 LPN（逻辑页号）；

c. 把由所有逻辑块组成的空间称为逻辑空间；

(3) 建立逻辑空间和物理空间的映射

a. 每个逻辑块都会映射到唯一的一个物理块，但不是每个物理块都会映射到一个逻辑块，每个物理块至多被映射到一个逻辑块中；物理空间被保留一些至少一个物理块以作为坏块的替换或传输块或存储配置管理数据，这些被保留的物理块没有映射到任何逻辑块，因此逻辑空间中的逻辑

块数和物理空间中的物理块数不必是相同的；

b. 当物理块映射有逻辑块时，把对应的逻辑块号保存在物理块中，此时该物理块除了有一个根据其起始地址赋予的物理块号 PBN 外，另外还对应有一个逻辑块号 LBN；

c. 然后把逻辑块中的逻辑页映射到所对应物理块中的物理页；当物理页映射有逻辑页时，把对应的逻辑页号 LPN 保存在物理页中，此时该物理页除了有一个根据其在物理块内起始偏移地址赋予的物理页号 PPN 外，另外还对应有一个逻辑页号 LPN；当物理块或页没有和逻辑块或页映射时，物理块或页中就不保存逻辑块号或页号；

(4) 根据物理块中的逻辑块号和物理页中的逻辑页号，在 RAM 中建立两级反向的映射表

a. 先建立第一级映射表，把逻辑空间中的每个逻辑块映射到唯一的一个物理块上去：

第一级映射表的第一项属于逻辑块 0，用于保存逻辑块 0 对应的物理块号，第二项属于逻辑块 1，用于保存逻辑块 1 对应的物理块号；依此类推，第一级映射表的最后一项属于最后一个逻辑块，用于保存最后一个逻辑块对应的物理块号；

b. 然后为每个逻辑块建立一张第二级映射表，把在这个逻辑块中的逻辑页映射到对应物理块中的物理页上去：

第二级映射表的第一项属于逻辑页 0，用于保存逻辑页 0 对应的物理页号，第二项属于逻辑页 1，用于保存逻辑页 1 对应的物理页号，依此类推，一个逻辑块中的所有逻辑页都映射到在同一个物理块内的各物理页上；

(5) 定义一个虚拟磁盘

a. 它由一个以上的扇区组成，称之为虚拟扇区；各虚拟扇区的大小相同，并且虚拟扇区的大小和现实磁盘中的扇区的大小相同，通常是 512 字节；

b. 把由所有虚拟扇区组成的空间称为虚拟空间；按照虚拟扇区在虚拟空间中的起始地址从小到大的顺序给每个虚拟扇区指定一个编号 VSN（虚

拟扇区号);

c. 计算机的程序 (比如文件系统) 在读写 flash 存储器上的内容时, 提供的地址被解释成虚拟扇区号;

(6) 建立虚拟空间到逻辑空间的映射, 把虚拟空间中的每个虚拟扇区映射到唯一的一个逻辑页上去; 该映射方法是:

a. 先把最前面编号为 0~(M-1) 的 M 个虚拟扇区映射到第一个逻辑块中的第一个逻辑页, 再把后面紧接着编号为 M~(2M-1) 的 M 个虚拟扇区映射到第二个逻辑块中的第一个逻辑页, 依次类推, 一直到把后面紧接着的 M 个虚拟扇区映射到最后一个逻辑块中的第一个逻辑页;

b. 然后, 继续在第一个逻辑块中建立映射, 把后续紧接着的 M 个虚拟扇区映射到第一个逻辑块中的第二个逻辑页, 一直到把后面紧接着的 M 个虚拟扇区映射到最后一个逻辑块中的第二个逻辑页; 依此类推, 一直到把后面紧接着的 M 个虚拟扇区映射到最后一个逻辑块中的最后一个逻辑页;

M 是每个逻辑页包含的虚拟扇区的个数, 即 M 值为逻辑页的大小除以虚拟扇区的大小的商, 虚拟空间的大小和逻辑空间的大小是相同的;

c. 虚拟地址到逻辑地址映射方法的计算公式是:

$$\text{offset}=(\text{VSN mod M});$$

$$\text{LBN}=\left(\left(\text{VSN}-\text{offset}\right) \div \text{M}\right) \text{ mod N};$$

$$\text{LPN}=\left(\left(\left(\text{VSN}-\text{offset}\right) \div \text{M}\right)-\text{LBN}\right) \div \text{N};$$

式中: mod 是模运算 (相除取余数);

VSN 是虚拟扇区的虚拟扇区号;

LBN 是逻辑地址中的逻辑块号;

LPN 是逻辑地址中的逻辑页号;

offset 是逻辑页的页内偏移, 即逻辑页内第几个虚拟扇区;

M 是每个逻辑页 (也是每个物理页) 包含的虚拟扇区的个数, 即逻辑页大小是虚拟扇区大小的 M 倍;

N 是逻辑空间中逻辑块的块数。

2. 根据权利要求1所述的方法，其特征在于：所述的步骤（5）的计算机程序（比如文件系统）在读 flash 存储器上的数据时，提供的地址被解释成虚拟扇区号，读操作包括：

- a. 先计算出虚拟扇区号对应的逻辑块号和逻辑页号，以及页内的偏移；
- b. 再根据两级映射表把逻辑块号和逻辑页号转换为对应的物理块号和物理页号；
- c. 然后在对应物理块的物理页中从页内偏移开始读取数据。

3. 根据权利要求1所述的方法，其特征在于：所述的步骤（5）的计算机程序（比如文件系统）在写数据到 flash 存储器时，提供的地址也被解释成虚拟扇区号，写操作包括：

- a. 先计算出虚拟扇区号对应的逻辑块号和逻辑页号，以及页内的偏移；
- b. 再根据两级映射表把逻辑块号和逻辑页号转换为对应的物理块号和物理页号；
- c. 把该物理块号和物理页号所寻址的物理页的数据读到 RAM 中，并根据页内偏移量在该 RAM 中更新数据；然后在该物理块中按物理页号从小到大的顺序找第一个未曾写数据的物理页（第一个空闲的物理页），把在 RAM 中更新后的整页数据写入到该空闲物理页；
- d. 更新该逻辑块的第二级映射表，使得逻辑块中的逻辑页映射到对应物理块中的写有新数据的物理页上。

4. 根据权利要求1或3所述的方法，其特征在于：所述的步骤（5）的计算机程序（比如文件系统）在写数据到 flash 存储器时，提供的地址也被解释成虚拟扇区号，写操作包括：

- a. 先计算出虚拟扇区号对应的逻辑块号和逻辑页号，以及页内的偏移；
- b. 如果根据第一级映射表逻辑块没有对应的物理块，则分配一个新的物理块映射到该逻辑块，然后在该新物理块中找第一个空闲的物理页，在该空闲物理页中从页内偏移开始写入数据；
- c. 如果根据第一级映射表逻辑块有对应的物理块，但是根据该逻辑块

的第二级映射表逻辑页没有对应的物理页，则在对应物理块中找第一个空闲的物理页，然后在该空闲物理页中从页内偏移开始写入数据；

d. 更新该逻辑块的第二级映射表，使得逻辑块中的逻辑页映射到对应物理块中的写有新数据的物理页上；

5. 根据权利要求3所述的方法，其特征在于写操作进一步包括：

a. 如果在对应的物理块中找不到一个空闲物理页，则先回收该物理块，以释放该物理块中被旧数据占用的物理页；

b. 回收完成后，在该物理块中找到第一个空闲物理页来保存更新后的整页数据；

c. 更新该逻辑块的第二级映射表，使逻辑页映射到这个新的物理页上。

6. 根据权利要求4所述的方法，其特征在于写操作进一步包括：

a. 如果在对应的物理块中找不到一个空闲物理页，则先回收该物理块，以释放该物理块中被旧数据占用的物理页；

b. 回收完成后，在该物理块中找到第一个空闲物理页来保存更新后的整页数据；

c. 更新该逻辑块的第二级映射表，使逻辑页映射到这个新的物理页上。

7. 根据权利要求5所述的方法，其特征在于回收操作包括：

a. 保留一些至少一个物理块，作为传输物理块，用于回收处理时的数据转存；

b. 把被回收物理块上的所有包含有效数据的物理页上的数据拷贝到一个传输物理块中，同时更新该物理块对应的逻辑块的第二级映射表；

c. 最后擦除被回收物理块，该被回收物理块就成为一个空闲的物理块，随后的回收操作可以用该块来转存数据。

8. 根据权利要求1所述的基于 sector 访问的 flash 存储器的存储管理，其特征在于存储管理方法进一步包括：

(1) 对 NOR flash 提供了一个划分方法，把一组物理地址连续的定长

区域组成一个物理页，每个物理页包括用于保存文件数据的区域，称之为数据区，以及一小段用于保存物理页的管理信息的区域，称之为备用区；

物理页数据区的大小设定为 512B，备用区的大小是 2B~16B，根据物理页的管理信息的多少决定；

(2) 对 NOR flash，每个物理块有一小部分至少 6B 被保留用于保存物理块的管理信息；管理信息有物理块对应的逻辑块号，物理块的擦除次数，物理块的状态等。

9. 根据权利要求 1 所述的基于 sector 访问的 flash 存储器的存储管理，其特征在于存储管理方法进一步包括：为了减少对 RAM 的需求，省略每个逻辑块的第二级映射表，在完成了逻辑块号到物理块号的转换后，通过扫描对应物理块中的物理页的方法来完成逻辑页号到物理页号的转换，即通过在物理块中查找包含该逻辑页号的最新物理页的方法，实现逻辑块第二级映射表的功能。

10. 根据权利要求 1 所述的基于 sector 访问的 flash 存储器的存储管理，其特征在于存储管理方法进一步包括：本发明所使用的建在 RAM 中的两级映射表可以在系统启动的时候进行重构，重构方法包括：

(1) 第一级映射表重构方法包括：

- a. 系统启动时，扫描每个物理块；
- b. 如果物理块有对应的逻辑块号，则把物理块号写到该逻辑块号对应的表项中；
- c. 如果有两个物理块对应到同一个逻辑块号，扫描两个物理块，则选择映射有较多有效逻辑页的物理块和该逻辑块号对应，另外一个物理块被擦除；如果两个物理块包含的有效物理页数相同，则任选一个和该逻辑块号对应，另外一个物理块被擦除；

d. 当所有的物理块都被扫描后，完成第一级映射表重构；

(2) 第二级映射表重构方法包括：

- a. 如果系统需要在 RAM 中构建第二级映射表，则从逻辑块 0 开始检查；

- 
- b. 如果有物理块和逻辑块对应，扫描和该逻辑块对应的物理块上的每个物理页；
  - c. 如果物理页有对应的逻辑页号，则把物理页号写到该逻辑页号对应的表项中；
  - d. 如果有两个物理页对应到同一个逻辑页号，则选择物理页号大的物理页和该逻辑页号对应；
  - e. 当该逻辑块对应的物理块上所有的物理页都被扫描后，完成该逻辑块的第二级映射表重构。
  - f. 当所有逻辑块都被扫描后，完成所有逻辑块的第二级映射表重构。



## 基于 sector 访问的 flash 存储器的存储管理

### 所属领域

本发明属于存储管理技术领域，涉及以 flash 存储器设备为介质的存储管理系统，尤其涉及基于 sector 访问的 flash 存储器的组织方法和管理方法以及数据访问方法，使得 flash 存储器设备能象磁盘一样的运作，即实现固态内存磁盘。

### 背景技术

现有技术的 Flash 存储器是一种类似于 EEPROM 的非易失存储器设备，目前广泛的用于嵌入式设备中。Flash 存储器有如下特性：

1. Flash 的写操作可以把 flash 存储器上任意 bit 的数据的从 1 变成 0。但是对任一 bit，写操作都不能把它从 0 变成 1。

2. Flash 的擦除操作只能把 flash 存储器上每个 bit 的数据从 0 变成 1。而且相对于写操作，擦除操作需要更多的执行时间。无论是 NAND flash 还是 NOR flash，擦除都是一个很耗时的操作。

3. Flash 最小可擦除区域的大小和最小可写区域的大小是不一样的，而且相差很大。Flash 的最小可擦除区域称为块 (block)，对 NOR flash，块的大小通常是 8KB 或者 64KB 或者 128KB 等。对 NAND flash，块的大小通常是 16KB、128KB 或者 256KB 等 (不计备用区)。而最小可写区域的大小比块的大小要小很多。对 NOR flash 存储器，块直接由一系列连续的字节组成，最小可写区域为单字节或者双字节。对 NAND flash 存储器，块由一系列大小相等的页 (page) 组成，最小可写区域是页或者页的一小部分，页的大小通常是 512B 或者 2KB 等 (不计备用区)，但是在两次连续的擦除操作之间，每个页的写入次数是有限制的，通常是 3-10 次，超过这个次数，该页上的数据就会变得不可靠。

4. Flash 存储器设备由一系列块组成，每个块的可擦除次数是有限的，超过这个最大可擦除次数的限制后，该块的工作稳定性就没有保证。

5. 对于一些大容量 NAND flash，比如 SAMSUNG 的 NAND flash K9K8G08U0A，块内的各个页必须被顺序写入，不能以随机的顺序对块内的各个页写入数据。

因此每次在修改 flash 存储器中已有的数据前，需要把对应块中的全部内容先读到 RAM 中，在 RAM 中对相应的数据进行修改，然后擦除包含原始数据的 flash 块，最后把 RAM 中的数据写回到原来的块中。这种 flash 存储管理方法存在下列缺陷：

1) 写数据时容易导致数据丢失。在擦除块之后但是在把数据写回块之前，发生任何原因的系统掉电，都会造成该块的全部数据包括不要求修改的数据的丢失。

2) 存储器写操作速度慢效率低。每次数据修改即使是一个 Bit 的修改都要求先擦除对应的块，增加了写操作的时间。

3) 频繁的修改一个特定块中的数据，会导致该块被频繁的擦除，该块就会比其他块更快的达到最大可擦除次数，因此影响整个系统的使用寿命。

许多现有的操作系统（OS）都带有管理数据的文件系统，那些文件系统通常以磁盘作为管理的对象，而磁盘是以扇区（sector）为单位读写的，扇区的大小通常是 512 字节。文件系统会对磁盘上的任意扇区进行反复的数据读和写操作。Flash 存储器的上述特性使得它无法直接用于现存的文件系统。为了让现存的文件系统可以不经修改的用于 flash 存储器设备，需要提供一种方法来分配和管理 flash 存储器，使文件系统可以用它的地址来寻址相应的扇区，并可以反复的读写，而且其读写可以以磁盘扇区的大小为单位。

美国专利 6381176 提出的 flash 管理方法，提高了写的效率和系统的健壮性，但是该技术方案需要在 flash 存储器的每个块内预留一些区域，用于数据更新，这影响了 flash 存储器的空间利用率，同时，因为预留更新区和原始数据区在同一块（block）内的两个不同区域，该技术方案无法适用于

某些要求对块内的页顺序写入的 NAND flash。另外该方案把一段地址相邻的虚拟扇区映射到一个块中，在文件数据更新时，几个相邻的虚拟扇区中的数据可能被同时更新，因此会很快用光预留的更新区域，导致频繁的擦除该块，影响 flash 的使用寿命以及写的速度。

中国专利公开号：CN 1805049 提出的 flash 的管理方法，提高了写的效率和系统的健壮性，但是该技术方案在跟踪状态变化时，需要对 flash 同一个区域进行多次的写，以记录各个过程。因为 NAND flash 对同一个页有前面描述的写次数的限制，所以该方案不能用于 NAND flash。而且，因为 NAND flash 写整个页和写页的一小部分消耗的时间相差不多，因此该技术方案即使能减少写的次数，对 NAND flash 依然有写效率的问题，因此需要有一个优化的方法，并且该方法还能适用于 NOR flash 和 NAND flash。

## 发明内容

为了克服现有技术的上述缺陷，本发明提供了 flash 存储器的一种组织方法、管理方法以及数据读写方法，以提高 flash 用于数据存储时的健壮性，提高在 flash 上写数据时的写效率，提高每个 flash 块的使用均衡性，并使 flash 存储器能模拟出和内存磁盘一样的基于 sector 的数据访问接口，使得现存的文件系统不用修改就能把 flash 存储器用作文件数据存储设备。虽然这些方法以 NOR flash 和 NAND flash 为例子来描述，但是这些方法也可以用于其他的有类似读写特性的存储设备。

本发明的目的是通过以下的技术方案来实现：基于 sector 访问的 flash 存储器，其方法包括下列步骤：

(1) 定义 Flash 存储器的物理空间。根据 flash 存储器的物理结构，将其存储器划分为一个以上的块，称之为物理块，各物理块的大小相同，按照物理块起始地址从小到大的顺序给每个物理块指定一个编号（物理块号 PBN）。然后把每个物理块划分为一个以上的页，称之为物理页，各物理页的大小相同。物理块中的每个物理页按照它在物理块内起始偏移地址从小到大的顺序给每个物理页指定一个在该物理块内唯一的编号（物理页号

PPN)。把物理块组成的空间称为物理空间。

(2) 定义一个逻辑存储区，并将其划分为一个以上的块，称之为逻辑块，各逻辑块的大小相同，并且逻辑块的大小和物理块的大小相同，每个逻辑块都有一个逻辑地址，把逻辑块组成的空间称为逻辑空间。按照逻辑块在逻辑空间中的起始地址从小到大的顺序给每个逻辑块指定一个编号（逻辑块号 LBN）。然后把逻辑块划分为一个以上的页，称之为逻辑页，各逻辑页的大小相同，并且逻辑页的大小和物理页的大小相同。逻辑块中的每个逻辑页按照它在逻辑块内起始偏移地址从小到大的顺序给每个逻辑页指定一个在该逻辑块内唯一的编号（逻辑页号 LPN）。

(3) 建立逻辑空间和物理空间的映射。每个逻辑块都会映射到唯一的一个物理块，但不是每个物理块都会映射到一个逻辑块，有一些至少一个物理块被保留以作为坏块的替换或传输块或存储配置管理数据，这些被保留的物理块没有映射到逻辑块，每个物理块至多被映射到一个逻辑块中。因此逻辑空间中的逻辑块数和物理空间中的物理块数不必是相同的。当物理块映射有逻辑块时，把对应的逻辑块号保存在物理块中，此时该物理块除了有一个根据其起始地址赋予的物理块号 PBN 外，另外还有一个逻辑块号 LBN。然后把逻辑块中的逻辑页映射到所对应物理块中的物理页。当物理页映射有逻辑页时，把对应的逻辑页号（LPN）保存在物理页中，此时该物理页除了有一个根据其在物理块内起始偏移地址赋予的物理页号 PPN 外，另外还有一个逻辑页号 LPN。当物理块（页）没有和逻辑块（页）映射时，物理块（页）中就不保存逻辑块（页）号。

(4) 根据物理块中的逻辑块号和物理页中的逻辑页号，在 RAM 中建立两级反向的映射表。第一级映射表把逻辑空间中的每个逻辑块映射到唯一的一个物理块上去，第一级映射表的第一项属于逻辑块 0，用于保存逻辑块 0 对应的物理块号，第二项属于逻辑块 1，用于保存逻辑块 1 对应的物理块号，依此类推。然后为每个逻辑块建立一张第二级映射表，把在这个逻辑块中的逻辑页映射到对应物理块中的物理页上去。第二级映射表的第一项属于逻辑页 0，用于保存逻辑页 0 对应的物理页号，第二项属于逻辑页 1，

用于保存逻辑页 1 对应的物理页号，依此类推。一个逻辑块中的所有逻辑页都映射到同一个物理块中。

(5) 定义一个虚拟磁盘，其由一个以上的扇区 (sector) 组成，称之为虚拟扇区。各虚拟扇区的大小相同，并且虚拟扇区的大小和现实磁盘的扇区的大小相同，通常是 512 字节。把所有虚拟扇区组成的空间称为虚拟空间。按照虚拟扇区在虚拟空间中的起始地址从小到大的顺序给每个虚拟扇区指定一个编号 (虚拟扇区号 VSN)。计算机程序 (比如文件系统) 在读写 flash 存储器上的内容时提供的地址被解释成虚拟扇区号。

(6) 建立虚拟空间到逻辑空间的映射，把虚拟空间中的每个虚拟扇区映射到唯一的一个逻辑页上去。该映射方法是：把最前面的 M 个虚拟扇区 (编号为 0~(M-1)) 映射到第一个逻辑块中的第一个逻辑页，把紧接着的后 M 个虚拟扇区 (编号为 M~(2M-1)) 映射到第二个逻辑块中的第一个逻辑页，依次类推，一直到最后一个逻辑块中的第一个逻辑页；然后再回到第一个逻辑块，把后续紧接着的 M 个虚拟扇区映射到第一个逻辑块中的第二个逻辑页，一直到最后一个逻辑块中的第二个逻辑页，依此类推，一直到最后一个逻辑块中的最后一个逻辑页。这里的 M 就是每个逻辑页 (也即每个物理页) 包含的虚拟扇区的个数，即 M 就是逻辑页的大小除以虚拟扇区的大小的商。虚拟空间的大小和逻辑空间的大小是相同的。虚拟地址到逻辑地址映射方法的计算公式是：

$$\text{offset}=(\text{VSN mod } M),$$

$$\text{LBN}=\left(\left(\left(\text{VSN}-\text{offset}\right)\div M\right)\text{ mod } N\right),$$

$$\text{LPN}=\left(\left(\left(\text{VSN}-\text{offset}\right)\div M\right)-\text{LBN}\right)\div N,$$

mod 是模运算，也就是相除取余数的运算。

VSN 是虚拟扇区的虚拟扇区号。

LBN 是逻辑地址中的逻辑块号。

LPN 是逻辑地址中的逻辑页号。

offset 是逻辑页的页内偏移 (即逻辑页内第几个虚拟扇区)。

M 是每个逻辑页 (也即每个物理页) 包含的虚拟扇区的个数，即逻辑

页大小是虚拟扇区大小的  $M$  倍。

$N$  是逻辑空间中逻辑块的块数。

在读 flash 存储器上的数据时，计算机程序（比如文件系统）提供的地址被解释成虚拟扇区号，先根据前述第（6）步中的方法计算出虚拟扇区号对应的逻辑块号和逻辑页号，以及页内的偏移，再根据前述第（4）步中的两级映射表把逻辑块号和逻辑页号转换为对应的物理块号和物理页号，然后在对应物理块的物理页中从页内偏移开始读取数据。

在写数据时，计算机程序（比如文件系统）提供的地址也被解释成虚拟扇区号，先根据前述第（6）步中的方法计算出虚拟扇区号对应的逻辑块号和逻辑页号，以及页内的偏移，再根据前述第（4）步中的两级映射表把逻辑块号和逻辑页号转换为对应的物理块号和物理页号。如果逻辑块没有对应的物理块，则分配一个新的物理块映射到该逻辑块。如果逻辑块有对应的物理块，但是逻辑页没有对应的物理页，则在该物理块中找第一个未曾写数据的物理页（第一个空闲的物理页）以保存请求写入的数据。如果逻辑页有对应物理页，即逻辑页对应的物理页已经写有数据，则不能在该页上直接写，应先把整个物理页读到 RAM 中，并根据页内偏移量在该 RAM 中更新数据。然后在该物理块中找第一个空闲的物理页，把在 RAM 中更新后的整页数据写入到该空闲物理页。最后更新该逻辑块的第二级映射表，使得逻辑块中的逻辑页映射到对应物理块中的写有新数据的物理页上。原来的那个物理页包含旧的数据，成为脏页，脏页不能用于数据写入，直到包含它的物理块被擦除。如果前述的那个空闲物理页无法在该物理块中找到，根据第（6）步的映射方法，只有整个虚拟空间都用完了，才会发生整个逻辑空间都用完的情况，也即逻辑块中所有逻辑页都映射到了物理页。如果虚拟空间没有用完而对应的物理块中已经没有空闲的物理页了，则该物理块必然包含脏页。这时先回收该物理块，以释放该物理块中被旧数据占用的物理页。回收完成后，该物理块中必然可以找到一个空闲物理页来保存更新后的整页数据，同时更新该逻辑块的第二级映射表，使逻辑页映射到这个新的物理页上。原来的物理页包含旧的数据，成为脏页，将在下

次回收该物理块时释放。在物理块内找空闲的物理页时，总是优先使用物理页号最小的空闲物理页。

本发明方法保留了一些至少一个物理块，这些保留的物理块用于：1) 替换 flash 存储器在使用过程中形成的坏块，使得和逻辑块映射的物理块数保持不变。2) 存储本技术方案的实现系统的管理数据，比如所有坏块的索引表等。3) 回收处理时的数据转存（称为传输块）。回收操作先选择一个包含脏物理页的物理块（回收物理块），然后把该物理块上的包含有效数据的物理页上的内容拷贝到一个保留的物理块中，最后擦除回收物理块，该回收物理块就成为一个保留的空闲物理块，随后的回收操作可以用该块来转存数据。

NAND flash 的硬件自身提供了物理页划分的方法，物理块由一系列至少一个物理页组成。每个物理页包含用于保存文件数据的区域（后续称之为数据区），以及一小段用于保存物理块和物理页的管理信息的区域（后续称之为备用区）。NOR flash 的硬件没有提供这种物理页的划分方法，本发明提供了一个划分方法，把一组物理地址连续的定长区域组成一个物理页，每个物理页包括用于保存文件数据的区域（数据区），以及一小段用于保存物理页的管理信息的区域（备用区）。物理页数据区的大小设定为 512B，备用区的大小是 2B~16B，根据物理页的管理信息的多少决定。对 NOR flash，每个物理块有一小部分至少 6B 被保留用于保存物理块的管理信息，比如物理块对应的逻辑块号，物理块的擦除次数，物理块的状态等。

为了减少对 RAM 的需求，本发明方法的一个改进是省略每个逻辑块的第二级映射表，在完成了逻辑块号到物理块号的转换后，通过扫描对应物理块中的物理页的方法来完成逻辑页号到物理页号的转换，即通过在物理块中查找包含该逻辑页号的最新物理页的方法，实现逻辑块第二级映射表的功能。

本发明方法所使用的放在 RAM 中的两级映射表可以在系统启动的时候进行重构。系统启动时，扫描每个物理块，如果物理块有对应的逻辑块号，则把物理块号写到该逻辑块号对应的表项中。如果有两个物理块对应

到同一个逻辑块号，则是在回收物理块时发生了掉电，扫描两个物理块，然后选择映射有较多有效逻辑页的物理块和该逻辑块号对应，另外一个物理块被擦除。如果两个物理块包含的有效物理页数相同，则任选一个和该逻辑块号对应，另外一个物理块被擦除。当所有的物理块都被扫描后，第一级映射表重构成功。如果系统需要在 RAM 中构建第二级映射表，则从逻辑块 0 开始检查，如果有物理块和逻辑块对应，扫描和该逻辑块对应的物理块上的每个物理页，如果物理页有对应的逻辑页号，则把物理页号写到该逻辑页号对应的表项中。如果有两个物理页对应到同一个逻辑页号，则选择物理页号大的物理页和该逻辑页号对应。

本发明具有以下实质性效果：

1) 本发明实现系统更改文件数据时，在成功保存新数据到 flash 存储器前，不会破坏文件的原始数据，因此任何时候系统掉电，或者可以保证数据已经写入，或者可以恢复原始的文件数据，提高了数据的可靠性和系统的健壮性。

2) 写数据到 flash 存储器上时，不必每次写操作都要先擦除 flash 块，大大减少了 flash 块的擦除次数，延长了 flash 的使用寿命以及提高了写的速度。

3) 本发明采用的虚拟扇区到逻辑块逻辑页的映射方法和逻辑块逻辑页到物理块物理页的映射方法，（即虚拟地址到逻辑地址的转换方法和逻辑地址到物理地址的转换方法），使得文件系统的静态数据（很少修改的数据，比如程序代码等）能尽最大可能的分布在不同的物理块中，因此在最大程度上消除了静态区域的形成（静态区指全部空间用于静态数据存储的物理块），提高了各个 flash 物理块的使用均衡性，延长了 flash 的使用寿命。同时这些映射方法的使用，使得在整个地址空间用完之前，物理块内总会有未用的物理页，物理块内目前未用的物理页可以临时作为更新区域使用，因此不必在每个物理块内预留固定数量的更新区域。提高了 flash 存储器的空间利用率。

4) 本发明的空闲物理页的分配方法，保证了每个物理块内的物理页是



以顺序的方式写入的。因此本发明的技术方案可以适用于各种 NAND flash 型号。通过对 NOR flash 的物理块进行页划分,本发明的技术方案同时可用于 NOR flash。

5)提供基于扇区的访问接口以及 flash 存储器的分配和管理方法,使 flash 存储器能和磁盘兼容,最大程度的重用了现存的 OS 程序(文件系统),让现存的文件系统不经修改可以用于 flash 存储器设备,方便了整个应用系统的构建以及提高了通用性。本发明方法适用于各型 flash 存储器,并具有良好的存储管理性能,展示了广阔的应用领域,经济效益十分显著。

### 附图说明

图 1 是根据本发明实现的一个系统的各功能部件的示意图。

图 2 是根据本发明的 NAND flash 存储器的组织方法的应用例的示意图。

图 3 是根据本发明的 NAND flash 物理页备用区中的管理数据的应用例的示意图。

图 4 是根据本发明的 NOR flash 存储器的组织方法的应用例的示意图。

图 5a 是根据本发明的 NOR flash 块头部中的管理数据的实施例的示意图。

图 5b 是根据本发明的 NOR flash 物理页备用区中的管理数据的实施例的示意图。

图 6 是根据本发明的虚拟地址到逻辑地址的映射方法的实施例的示意图。

图 7 是根据本发明的逻辑地址到物理地址的映射方法的实施例的示意图。

图 8 是图 7 所示实施例的物理块的各物理页的状态示意图。

图 8 中: Valid—有效页、Dirty—脏页、Free—空闲页。

图 9 是本发明实施例的虚拟扇区读流程框图。

图 10 是本发明实施例的 NOR flash 的虚拟扇区写流程框图。

图 11 是本发明实施例的 NAND flash 的虚拟扇区写流程框图。

图 12 是本发明实施例的物理块上脏页的回收处理流程框图。

图 13a 是本发明实施例的一个物理块在回收操作执行前的物理块的状态。

图 13b 是本发明实施例的一个物理块在回收操作执行后的物理块的状态。

图 14 是本发明实施例的 NOR flash 的虚拟扇区删除流程框图。

图 15 是本发明实施例的 NAND flash 的虚拟扇区删除流程框图。

图 16 是本发明实施例的逻辑块号 (LBN) 到物理块号 (PBN) 的映射表的建立流程框图。

图 17 是本发明实施例用于 NOR flash 时逻辑块中的逻辑页号 (LPN) 到物理页号 (PPN) 的映射表的建立流程框图。

图 18 是本发明实施例用于 NAND flash 时逻辑块中的逻辑页号 (LPN) 到物理页号 (PPN) 的映射表的建立流程框图。

## 具体实施方式

通过具体实施例，并结合附图，对本发明技术方案作进一步的详细说明，使本发明的优点和特点能更加清楚地呈现。但是本发明的方法不受下述实施例的限制，可用各种类似的方式来实现。

根据本发明实现的典型系统的各功能部件如图 1 所示。它由微处理器 10、ROM 12、RAM 14、Flash Memory 控制器 16 和 Flash 存储器 18 构成。通过使用本发明提供的方法，Flash Memory 控制器 16 提供了一个基于扇区访问的虚拟地址空间，Flash 存储器 18 被模拟成了一个可以多次读写的非易失存储设备，微处理器 10 以及相应的 OS 程序如同访问磁盘一样发出读写命令，用虚拟地址空间中的虚拟扇区号在 flash 存储器 18 中读写数据。

NAND flash 存储器的组织方法应用例如图 2 所示，在本发明的具体实施例中，NAND flash 存储器设备包括一个单独的 flash 分区 20。Flash 分区 20 被划分成一系列的物理块 21。每个物理块 21 再被划分成一系列的物理

页 22。每个物理页 22 由两部分组成，数据区 24 和备用区 26。数据区 24 用于存储文件数据，备用区 26 用于存储管理物理块和物理页的管理数据。对于 NAND flash 存储器，物理块和物理页的划分，以及物理页内部的划分由 NAND flash 的硬件特性决定。NAND flash 物理页备用区 26 中的管理数据的数据结构如图 3 所示，数据结构中包含的信息有：物理页中的每个扇区的状态（Sector 状态 32、Sector 状态 33）、本物理页对应的逻辑页的页号 LPN 34、本物理页所在物理块的擦除计数 35、本物理页所在物理块对应的逻辑块的块号 LBN 36、本物理页所在物理块的块状态 37 和本物理页中存储的数据的纠错码 ECC 38。物理块的状态有：Bad（坏块），Valid（有效块）。物理页中的每个扇区在备用区 26 中都有一个对应状态指示域，用于说明该物理页中的每个扇区的状态，扇区的状态有：Free（扇区是空闲的），Valid（扇区包含有效数据），Deleted（扇区中的数据被删除）。NAND flash 实施例中每个物理页数据区包含两个扇区只是为了举例说明，依据本发明的构思，物理页数据区的大小可以是任意整数个扇区的大小。

如果物理页中每个扇区都是 Free 状态，那就称该物理页是 Free 状态。如果物理页中每个扇区都是 Deleted 状态，那就称该物理页是 Deleted 状态。如果物理页中某个扇区的状态是 Valid，那就称该物理页是 Valid 状态。物理页的状态是 free，说明该物理页没有被写入数据，该物理页是一个空闲页，物理页的状态是 valid，说明该物理页包含有效的数据，但是该页中的数据不一定是最新的数据，同一个物理块中可能有多于 1 个的物理页对应到同一个逻辑页，并且状态都是 Valid，仅仅物理页号最大的物理页包含着最新的数据，其他物理页包含的是被替换的数据。物理页的状态是 Deleted，表示与该物理页映射的逻辑页在逻辑空间中被释放，也即同一物理块中在该物理页之前所有映射到该逻辑页的物理页也都同时被删除。NAND flash 的物理块的状态是 valid，说明该块可以用于数据的存储。物理块的状态是 bad，说明该块是坏块，不能再用于数据存储。

NOR flash 存储器的组织方法应用例如图 4 所示，在本发明的具体实施例中，NOR Flash 存储器设备包括一个单独的 Flash 分区 40。Flash 分区 40

被划分成一系列的物理块 42，对 NOR flash 存储器，物理块的划分由 NOR flash 的硬件特性决定。通过使用本发明提供的方法，NOR Flash 存储器的每个物理块 42 再被划分成：一个块头部 (Block Header) 43，一系列的物理页 44，以及根据物理块大小的不同，每个物理块尾部可能有一个剩余区 45，其大小小于一个物理页 44 的大小。每个物理页 44 又划分为备用区 46 和数据区 47 两部分，数据区 47 用于存储文件数据，备用区 46 用于存储管理物理页的管理数据。块头部 43 用于存储管理物理块的管理数据。块头部 43 中的管理数据的数据结构如图 5a 所示，数据结构中包含的信息有：本物理块的擦除计数 51，本物理块对应的逻辑块的块号 LBN 52，以及物理块的状态 53。物理块的状态有：Free (空闲块) 和 Valid (有效块)。物理页备用区 46 中的管理数据的数据结构如图 5b 所示，数据结构中包含的信息有：物理页中的每个扇区的状态 54，以及本物理页对应的逻辑页的页号 LPN 55。扇区状态有：Free (空闲扇区)，Pending (正在往扇区写数据)，Valid (扇区中包含有效数据)，Invalid (扇区中的数据无效)。NOR flash 实施例中每个物理页数据区包含一个扇区只是为了举例说明，依据本发明的原理，物理页数据区大小可以是任意整数个扇区的大小。对 NOR flash，考虑到具体实现的简便，推荐物理页数据区的大小是一个扇区的大小。

如果物理页中每个扇区都是 Free 状态，那就称该物理页是 Free 状态。如果物理页中每个扇区都是 Invalid 状态，那就称该物理页是 Invalid 状态。如果物理页中某个扇区的状态是 Valid，那就称该物理页是 Valid 状态。物理页的状态是 free，说明该物理页没有被写入数据，该物理页是一个空闲页；物理页的状态是 valid，说明该物理页包含有效的数据；物理页的状态是 invalid，说明该物理页包含的数据都无效了，该物理页是一个脏页。NOR flash 的物理块的状态是 valid，说明该物理块包含有效的数据信息；物理块的状态是 free，说明该物理块目前是空闲的，是保留的物理块。

本发明虚拟地址到逻辑地址的映射方法的实施例如图 6 所示。首先逻辑地址空间被划分为一系列的逻辑块 62，然后逻辑块 62 再被划分为一系列的逻辑页 63。逻辑页 63 的大小和物理页 22 的大小相同(如果是 NOR Flash，

则和物理页 44 的大小相同)。逻辑块 62 的大小和物理块 21 的大小相同(如果是 NOR Flash, 则和物理块 42 的大小相同)。虚拟地址空间 60 被划分为一系列的虚拟扇区 61。逻辑页 63 的大小是虚拟扇区 61 大小的两倍。逻辑页的大小是虚拟扇区大小的两倍只是为了举例说明, 依据本发明的构思, 逻辑页大小可以是虚拟扇区大小的任意整数倍。虚拟地址到逻辑地址的映射方法是: 每 2 个虚拟扇区 61 映射到逻辑块 62 中的一个逻辑页 63, 虚拟扇区 61 先依次映射到各逻辑块 62 中的第一个逻辑页 63, 接着依次映射到各逻辑块 62 中的第二个逻辑页 63, 依此类推, 直到最后 2 个虚拟扇区 (VSN=2KN-2) 和 (VSN=2KN-1) 映射到最后一个逻辑块 (LBN=N-1) 中的最后一个逻辑页 (LPN=K-1)。

在读写 flash 中的数据时, 读写的地址被解释成虚拟地址空间中的虚拟扇区号 (VSN)。首先该虚拟扇区号被转换成逻辑地址, 即逻辑块号 (LBN) 和逻辑页号 (LPN) 以及页内偏移 (offset), offset 指出是页内第几个扇区。转换方法的计算公式是:

$$\text{offset}=(\text{VSN mod } M),$$

$$\text{LBN}=\left(\left(\left(\text{VSN}-\text{offset}\right) \div M\right) \text{ mod } N\right),$$

$$\text{LPN}=\left(\left(\left(\text{VSN}-\text{offset}\right) \div M\right)-\text{LBN}\right) \div N,$$

式中: mod 是模运算, 也就是相除取余数的运算。

M 是每个逻辑页 (也即每个物理页) 包含的虚拟扇区的个数, 即逻辑页大小是虚拟扇区大小的 M 倍, 本实施例以 M=2 为例,

N 是逻辑空间中逻辑块的块数。

然后再把逻辑地址转换为物理地址, 如图 7 所示。第一级映射表 70, 也叫块映射表, 它以逻辑块号 LBN 为索引, 把逻辑块 62 的 LBN 转换成对应物理块 21 的 PBN, 实现逻辑块到物理块的映射。逻辑块的第二级映射表 72 以逻辑页号 LPN 为索引, 把逻辑页 63 的 LPN 转换成对应物理页 22 的 PPN, 实现该逻辑块中的逻辑页到对应物理块中的物理页的映射。首先以逻辑地址中的 LBN 为索引, 从第一级映射表 70 的表项 71 中获得对应物理块的 PBN。再通过该逻辑块的第二级映射表 72, 以 LPN 为索引从对应的表项

73 中获得对应物理页的 PPN。物理页内的偏移和逻辑页内的偏移是一样的，因此逻辑地址中的 offset 直接作为物理地址中的 offset。这样完成了逻辑地址 (LBN:LPN:offset) 到物理地址 (PBN:PPN:offset) 的转换。因为逻辑地址和物理地址中的 offset 相同，后续在描述逻辑地址到物理地址转换时，只描述 (LBN:LPN) 到 (PBN:PPN) 的转换。本例中，LBN 为 2 的逻辑块映射到 PBN 为 100 的物理块，且该逻辑块中 LPN=1 的逻辑页映射到 PBN=100 物理块中的 PPN=0 的物理页。

图 8 示出了物理块号 PBN=100 所指的物理块上的各物理页的状态，以及物理页—逻辑页的映射关系，该映射对应于图 7 示例的逻辑页—物理页映射。图 7 中的页映射表 72 把逻辑页映射到物理页，即逻辑页 0 (LPN=0) 映射到物理页 2 (PPN=2)，逻辑页 1 映射到物理页 0，逻辑页 2 映射到物理页 5，……。在图 8 中，示出了一个反向的映射，物理页到逻辑页的映射，即物理页 0 映射到逻辑页 1，物理页 1 映射到逻辑页 3，物理页 2 映射到逻辑页 0，物理页 5 映射到逻辑页 2，……。

图 9 示出了读操作的流程。在一个读操作中，相应的地址被解释成虚拟扇区号，该虚拟扇区号先被转换成逻辑块号和逻辑页号以及页内偏移，然后通过第一级映射表 70，逻辑块号转换成相应的物理块号，通过该逻辑块的第二级映射表 72，逻辑页号转换成相应的物理页号，最后从该物理页内指定的页内偏移处开始读取所需的数据。具体流程为：

S900: 把虚拟扇区号 (VSN) 转换成逻辑地址 (LBN:LPN:offset)；

S902: 通过第一级映射表 70，把逻辑块号 LBN 转换成相应的物理块号 PBN；

S904: 然后通过该逻辑块的第二级映射表 72，把逻辑页号 LPN 转换成对应的物理页号 PPN；

S906: 然后从 (PBN:PPN) 所指物理页的 offset 偏移处开始读取相应的数据。

在一个写操作中，相应的地址被解释成虚拟扇区号，该虚拟扇区号先被换成逻辑块号和逻辑页号以及页内偏移。如果该逻辑块号没有对应的有

效物理块号，则分配一个空闲的物理块和该逻辑块对应，然后更新第一级映射表使逻辑块和物理块对应，然后在该物理块中按照物理偏移从小到大的顺序（也即物理页号从小到大的顺序）分配一个空闲的物理页（即第一个空闲的物理页），把数据写入到该物理页，最后更新该逻辑块的第二级映射表。如果该逻辑块号有对应的有效物理块号，通过该逻辑块的第二级映射表，查看该逻辑页对应的物理页，如果没有对应的物理页，则在该物理块中，按照物理页号从小到大的顺序分配一个空闲的物理页（即第一个空闲的物理页），把数据写入该物理页，最后更新该逻辑块的第二级映射表。如果逻辑块有对应的物理块，且逻辑页有对应的物理页，则先把该物理页的内容读到 RAM 中，根据页内偏移，修改相关数据，然后在该物理块中找第一个空闲的物理页，把修改后的数据写到该空闲物理页，最后更新该逻辑块的第二级映射表。

参见图 10，实施例的 NOR flash 的虚拟扇区写操作流程，该流程图示出了 NOR flash 的写操作方法：

S1000：把虚拟扇区号（VSN）转换成逻辑地址（LBN:LPN:offset）；

S1010：如果 LBN 没有对应的有效物理块号（PBN），则转 S1020；如果 LBN 有对应的有效物理块号（PBN），转 S1030；

S1020：分配一个空闲的物理块，然后更新第一级映射表 70 使逻辑块和物理块对应，最后转 S1030；

S1030：通过该逻辑块的第二级映射表 72，查看 LPN 是否对应有效有效的 PPN，即查看逻辑页是否有对应的物理页；如果有对应的物理页，转 S1040；如果没有对应的物理页，转 S1050；

S1040：把对应物理页的内容读取到 RAM 中，并根据页内偏移(offset)，修改更新相应部分的数据，然后转 S1050；

S1050：检查该物理块内是否有空闲的物理页；如果没有空闲的物理页，转 S1060；如果有空闲的物理页，转 S1070；

S1060：回收该物理块，以使该物理块中的脏页能重新用于存储数据；

S1070：在该物理块中找第一个空闲的物理页（即页号小的空闲物理页

优先分配), 把最新的数据写入该物理页;

**S1080:** 如果该逻辑页原来有对应的物理页, 则把原来对应的物理页中的各个扇区的的状态设置为 invalid;

**S1090:** 更新该逻辑块的第二级映射表, 使 LPN 映射到新分配的物理页的 PPN。

NAND flash 的写操作流程参见图 11。先把虚拟地址 VSN 转换成逻辑地址 (LBN,LPN,offset) (S1100)。如果 LBN 没有对应的有效 PBN(S1110), 则分配一个空闲的物理块和该逻辑块对应, 然后更新第一级映射表 70 使逻辑块和物理块对应(S1120), 通过该逻辑块的第二级映射表 72, 查看是否逻辑页 (LPN) 有对应的物理页(S1130), 如果有对应的物理页, 先把对应物理页的内容读取到 RAM 中, 根据页内偏移, 修改相应部分的数据(S1140), 然后, 不管逻辑页是否有对应的物理页, 在该物理块中找第一个空闲的物理页 (即页号小的空闲物理页优先分配), 把最新的数据写入该物理页 (S1170), 最后更新该逻辑块的第二级映射表, 使 LPN 映射到新分配的物理页的 PPN(S1180)。如果该物理块中没有空闲的物理页(S1150), 则该物理块需要先做回收处理(S1160)。对 NAND flash, 没有把逻辑页原来对应的物理页中的各个扇区的状态设置为 invalid, 以使本发明可以用于对物理块中的物理页有顺序写入要求的 NAND flash, 同时也可以减少对 NAND flash 的物理页的写入次数。同一份数据的新旧, 可根据包含数据的物理页的页号大小来判断, 页号大的物理页包含新的数据。

在图 10 和图 11 的写流程中, 如果物理块中无法找到空闲的物理页, 则该物理块需要做回收处理, 以释放该物理块中包含旧数据或者无效数据的物理页 (脏页), 使这些物理页能用于存储更新的数据。如前所述, 使用本发明实现的系统在物理空间中保留着一些空闲物理块, 用作回收处理中的传输块。

参见图 12, 该图示出了本发明实施例的物理块上脏页的回收处理流程。先选择一个空闲的物理块作为传输物理块(S1200)。然后扫描被回收物理块对应的逻辑块的第二级映射表 72 的每个表项 73, 从 LPN 0 对应的表项开始



(S1210), 如果 LPN 对应的物理页号 (PPN) 有效(S1220), 则把该物理页号所指的物理页的内容先读出再写到传输块的第一个空闲物理页中(S1230), 即在传输物理块中优先写页号小的空闲物理页, 拷贝时每个物理页中只有处于 valid 状态的扇区的内容被拷贝, 然后更新第二级映射表, 把 LPN 映射到传输块中该物理页的 PPN(S1240), 然后对下一个逻辑页号执行同样的处理过程(S1250); 如果逻辑页号 LPN 对应的物理块号无效(S1220), 则直接对下一个逻辑页号执行前述的处理过程(S1250)。以上过程一直执行到逻辑页号 LPN 到达最大逻辑页号为止(S1260)。最后把被回收的物理块擦除, 把它作为一个保留的空闲块, 又可以作为新的传输块, 同时更新第一级映射表映射 LBN 到原来的传输块的 PBN(S1270)。为了降低写操作过程中执行物理块回收操作的机率, 尽可能的使写操作能在物理块中找到空闲的物理页, 本发明提出周期性的回收物理块的构思, 即图 12 所示的物理块回收流程会周期性的被执行。

参见图 13a 和图 13b, 它示出了回收前和回收后物理块的状态。被回收的物理块 (对应于逻辑块号 LBN=2) 上包含有效数据的物理页按顺序拷贝到传输物理块上, 然后将被回收的物理块 (对应于逻辑块 2) 擦除, 成为一个空闲物理块, 不再对应到逻辑块 2, 而拷贝有有效数据的原传输物理块对应到了逻辑块 2。

通过这种新旧物理页替换的方法, OS 程序 (比如文件系统) 所用的存储空间就有了多次读写的能力。由于不是每次写操作都需要先执行擦除, 从而延长了 flash 存储器设备的使用寿命, 也提高了系统的写性能。另外, 在新数据成功写入前, 不会先丢掉旧的数据, 即使在写新数据的过程中有掉电的情况发生, 原来的数据也不会丢失, 提高了系统的健壮性。而且本发明每次分配空闲物理页时, 按照 PPN 从小到大的顺序分配, 因此也满足了大容量 NAND flash 存储器设备对物理块中的物理页需要顺序写的要求。OS 程序 (比如文件系统) 在存储数据时, 总是尽可能的使用连续的虚拟扇区段, 而且也尽量用虚拟扇区号小的扇区 (从低地址开始分配), 根据前述的虚拟地址到逻辑地址的转换方法, 连续的虚拟扇区均匀分散在各个

逻辑块中，也即均匀分散在各个物理块中，这使得静态数据最大可能的分散在不同的物理块，也就最大可能的消除了静态区域的形成，使每个物理块能被平均的使用，延长了整个 flash 设备的使用寿命。而且因为连续的虚拟扇区均匀分散在各个逻辑块，所以除非逻辑块中的所有逻辑页都被分配出去了（也即所有的虚拟扇区都分配出去了，系统的虚拟空间已经用完了），否则对应的物理块中物理页的数量总是多于该逻辑块中已经被分配出去的逻辑页的数量，这样在更新该逻辑块的逻辑页中的内容时，这些数量上多出来的物理页可以作为更新区使用，因此每个物理块中就不用预留物理页专做更新区，提高了 flash 的利用效率。

当 OS 程序（比如文件系统）删除文件时，属于该文件的虚拟扇区的内容就无效了，这时需要删除虚拟扇区对应物理页中的对应的扇区以在 flash 设备上释放该部分空间，这样可以减少包含相关物理页的物理块在回收处理中需要拷贝的扇区数，因此节省了回收处理的时间，而且及时的释放出物理页，可以有更多的物理页用作更新区，减少了物理块的擦除次数，延长了 flash 的使用寿命。

图 14 是本发明用于 NOR flash 的实施例的虚拟扇区删除流程，当收到一个删除命令来删除某个虚拟扇区时，先把虚拟扇区号 VSN 转换成逻辑地址（LBN:LPN:offset）(S1401)，如果逻辑块没有对应的物理块(S1402)，结束删除操作。如果有对应的物理块(S1402)，但是逻辑页没有对应的物理页(S1403)，结束删除操作。如果有对应物理页(S1403)且偏移 offset 对应的扇区的状态不是 valid (S1404)，结束删除操作。如果偏移 offset 对应的扇区的状态是 valid (S1404)，则把该扇区的状态变成 invalid (S1405)，如果这时该物理页不再包含任何 valid 状态的扇区(S1406)，则更新 LPN 到 PPN 的映射，该 LPN 映射到无效的 PPN 值(S1407)，即该 LPN 不映射到任何 PPN。

图 15 是本发明用于 NAND flash 的实施例的虚拟扇区删除流程，前面部分 S1501 ~S1504 的处理过程和 NOR flash 一样。如果偏移 offset 对应的扇区的状态是 valid (S1504)，并且该物理块中有空闲的物理页(S1505)，把整个物理页的内容读取到 RAM 中，在 RAM 中更改 offset 对应的扇区的状

态为 **deleted** (S1506), 然后在该物理块中找出第一个空闲的物理页, 把更改后的数据写入该物理页, 这样该扇区对应的状态就是 **deleted** (S1506)。如果该物理页中不再包含任何 **valid** 状态的扇区(S1508), 则更新 LPN 到 PPN 的映射, 该 LPN 就不映射到任何 PPN, 也即该 LPN 映射到无效的 PPN 值 (S1509)。如果该物理块中没有空闲的物理块(S1505), 则回收该物理块, 并在回收过程中, 不拷贝该物理页中 **offset** 对应的扇区中的数据(S1507), 由此释放该扇区的空间。

参见图 16, 该图示出了本发明实施例的逻辑块号 (LBN) 到物理块号 (PBN) 的映射表的建立流程。在系统启动时, 为了建立第一级映射表, 从第一个物理块 (PBN=0) 开始扫描 flash 分区的每个物理块(S1601)。获取该物理块对应的逻辑块号 LBN 和块状态(S1602); 对 NAND flash, 物理块的 LBN 和块状态保存在属于该物理块的物理页的备用区; 对 NOR flash, 物理块的 LBN 和块状态保存在物理块的头部; 对 NAND flash 若块状态指示是坏块, 则直接略过该块。如果块状态不是 **valid**, 或者获取的 LBN 无效 (即大于系统的最大逻辑块数) (S1603), 则该物理块被当作是空闲的物理块(S1604)。如果块状态是 **valid** 且获取的 LBN 有效(S1603), 则扫描该物理块, 建立该物理块中 PPN 和 LPN 之间的映射表, 也即该逻辑块的第二级映射表(S1605)。第二级映射表的建立流程如图 17 和图 18 描述。如果第一级映射表中该 LBN 还没有对应的 PBN(S1606), 则把 LBN 映射到当前的 PBN(S1607)。如果在映射表中, LBN 已经和有效的 PBN 映射, 则说明在物理块的回收处理过程中发生了意外掉电, 映射到同一个 LBN 的两个物理块中有一个是原始物理块。比较这两个物理块中映射到逻辑页的物理页数目 (也即处于 **valid** 状态的扇区的数目) (S1608), 选择映射有更多逻辑页的物理块作为原始物理块, 并把 LBN 映射到该物理块的 PBN(S1609), 擦除另外一个物理块, 使他成为一个空闲的物理块(S1610)。如果两个物理块中的物理页映射有一样多的逻辑页, 则任选一个物理块的 PBN 和 LBN 映射, 然后擦除另外一个物理块使它成为一个空闲物理块。接着处理下一个物理块(S1611), 当所有的物理块都被扫描后(S1612), 第一级映射表就建立起来

了，同时每个逻辑块的第二级映射表也建立起来了。

参见图 17，该流程图示出了 NOR flash 物理块对应的逻辑块的第二级映射表的建立过程。从第一个物理页（PPN=0）开始，扫描整个物理块（S1701），获取当前物理页对应的逻辑页号 LPN，以及该物理块内所有扇区的状态（S1702）。如果物理页内的每个扇区都是 Free 状态（即物理页是空闲的）（S1703），根据前述的物理页分配方法，该物理块内后续的物理页也都是空闲的，因此该逻辑块的第二级映射表建立完成。如果物理页内有扇区是 valid 状态（S1704），则把该 LPN 和当前的 PPN 建立映射（S1705）。如果物理页内没有 valid 状态的扇区，则 LPN 不和当前 PPN 建立映射。接着用上述的同样方法处理下一个物理页（S1706），如果物理块中的所有物理页都扫描了（S1707），该物理块对应的逻辑块的第二级映射表建立完成。

参见图 18，该流程图示出了 NAND flash 物理块对应的逻辑块的第二级映射表的建立过程。从第一个物理页（PPN=0）开始，扫描整个物理块的物理页（S1801），获取该物理页对应的逻辑页号 LPN，以及该物理块内所有扇区的状态（S1802）。如果物理页内的每个扇区都是 Free 状态（即物理页是空闲的）（S1803），根据前述的物理页分配方法，该物理块内后续的物理页也都是空闲的，因此该逻辑块的第二级映射表建立完成。如果物理页内的每个扇区都是 Deleted 状态（S1804），则该 LPN 映射到无效的 PPN（S1805），如果物理页有扇区是 valid 状态，则把该 LPN 和当前的 PPN 建立映射（S1806）。接着用上述的同样方法处理下一个物理页（S1807）。如果物理块中的所有物理页都扫描了（S1808），该物理块对应的逻辑块的第二级映射表建立完成。

为了节省对 RAM 的需求量，本发明的一个优化方法是简化逻辑地址到物理地址转换所用的映射表，只建立逻辑块到物理块的映射表（即第一级映射表），而不对每个逻辑块建立逻辑页到物理页的映射表（即第二级映射表）。为完成 LPN 到 PPN 的映射，按照物理页号 PPN 从大到小的顺序扫描逻辑块对应的物理块中的各物理页，从物理页的备用区中查找对应的 LPN，第一个找到的包含该 LPN 的物理页就是该 LPN 逻辑页所映射的物理页，如

果无法找到包含该 LPN 的物理页，则认为该 LPN 映射到无效的 PPN，这样就完成了 LPN 到 PPN 的映射。由此本发明只需建立第一级映射表，即 LBN 到 PBN 的映射表，节省了大量的 RAM 需求。

虽然本发明的具体实现是以 NOR flash 和 NAND flash 为例来描述的，但是本发明存储管理的基本技术方案也可用于其他和 flash 有类似操作特性的设备上，比如 AND flash 等。在具体实现时，用本方法作一点适应性的修改。

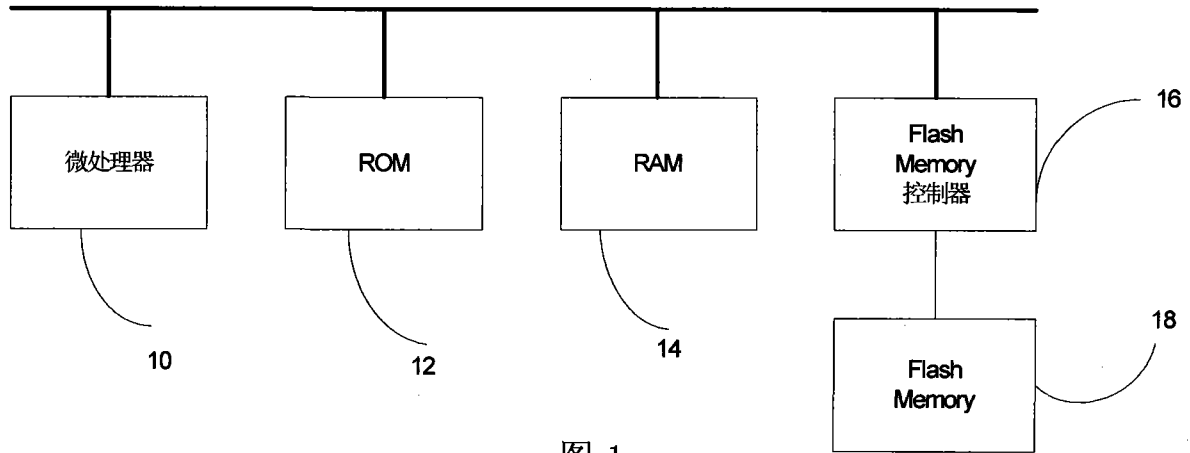


图 1

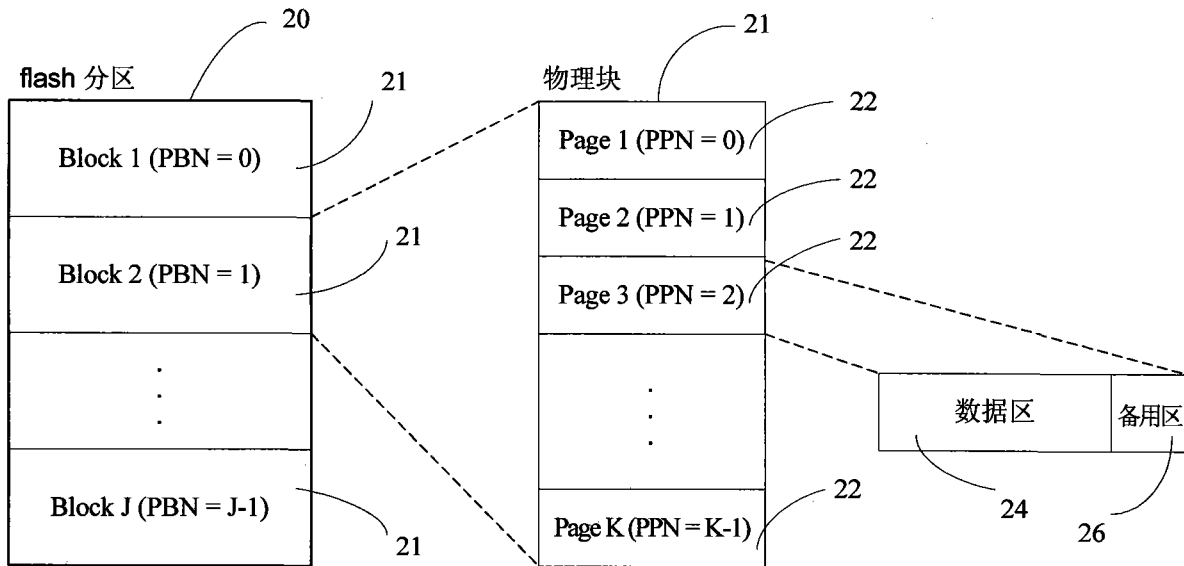


图 2

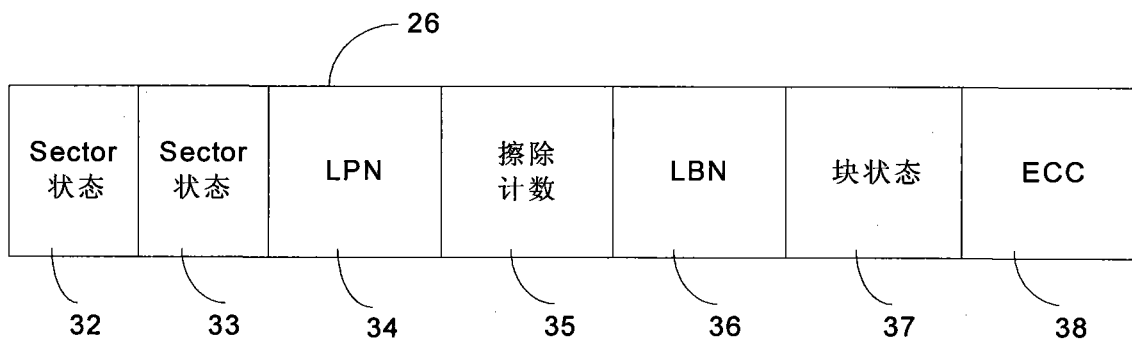


图 3

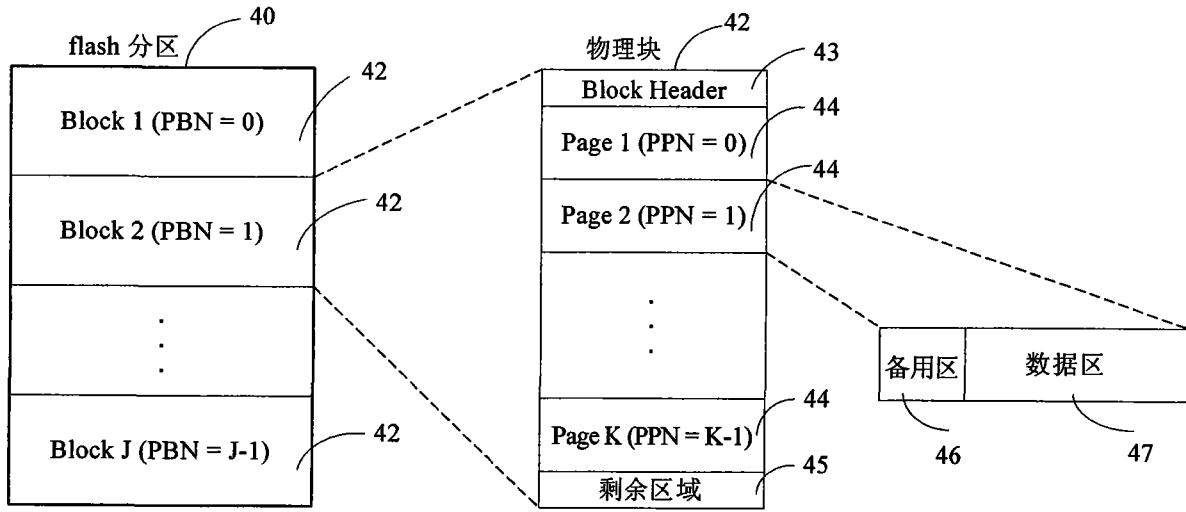


图 4

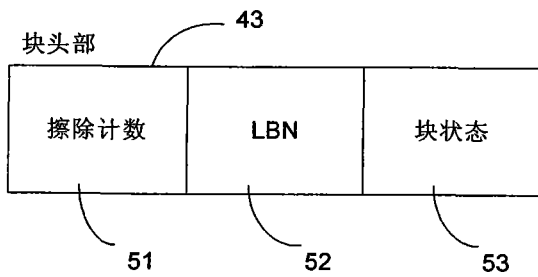


图 5a

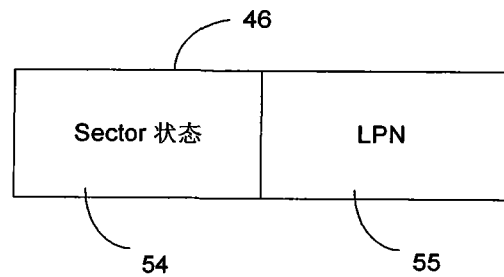


图 5b

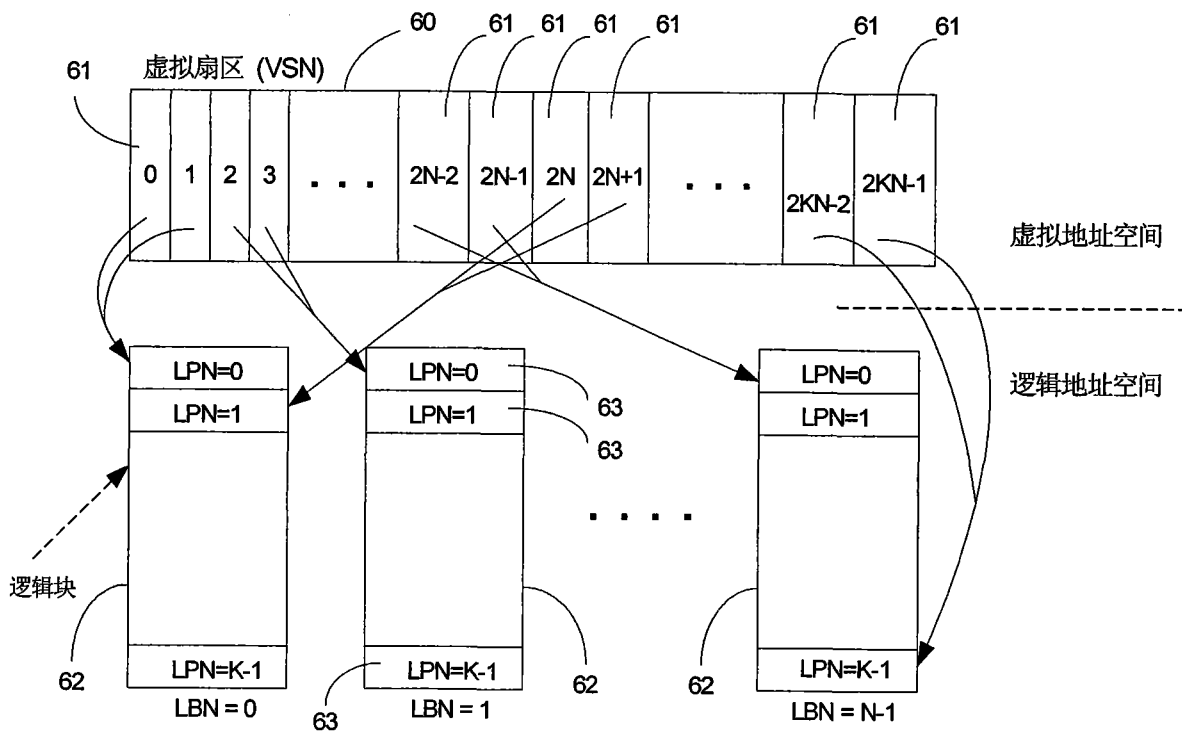


图 6

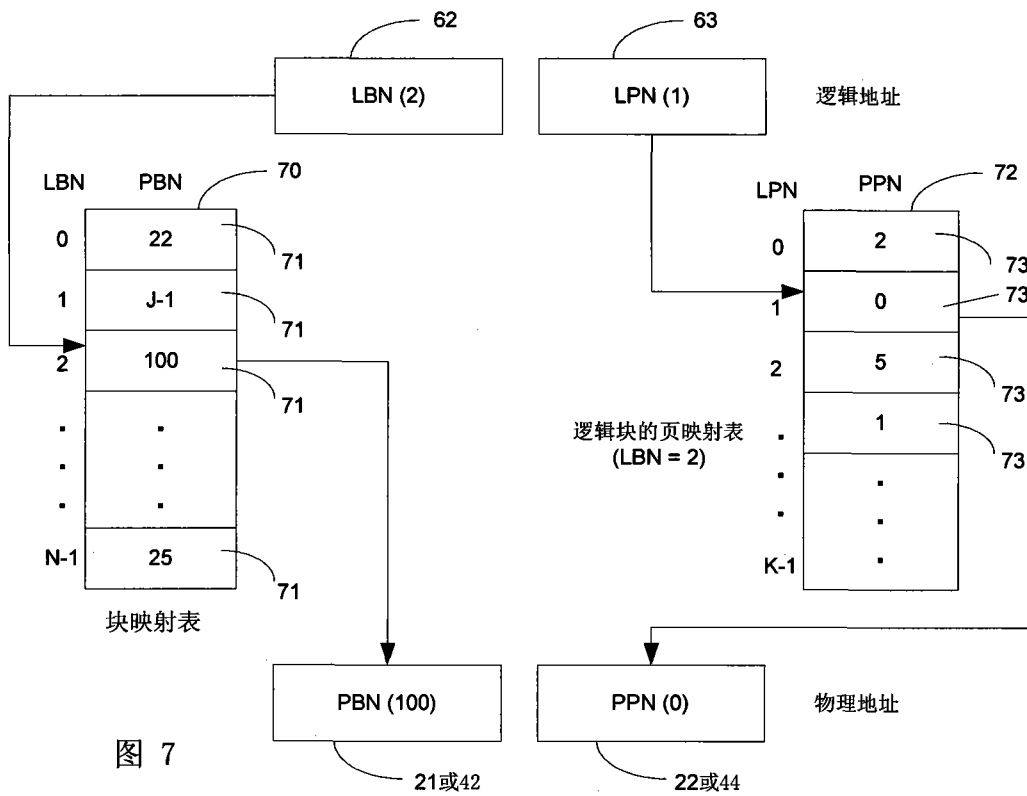


图 7

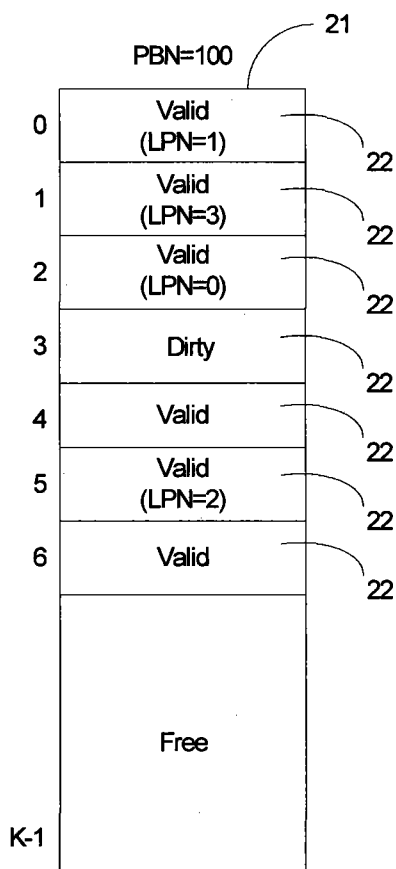


图 8

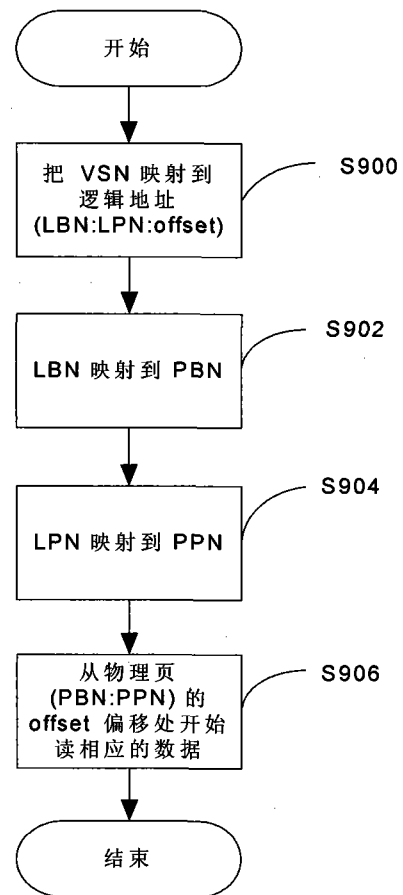


图 9



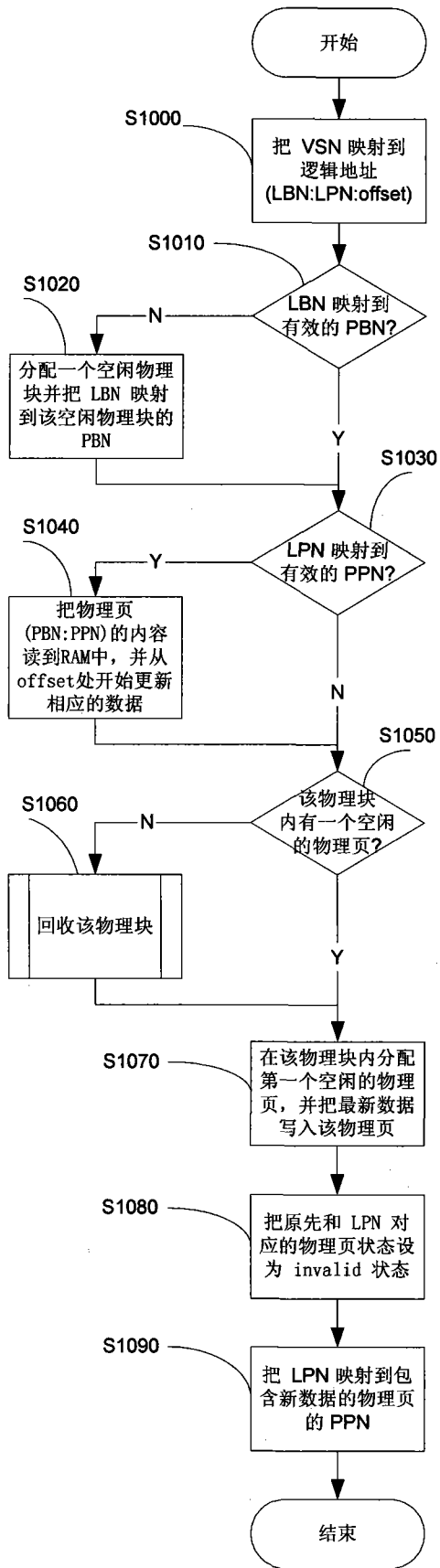


图 10

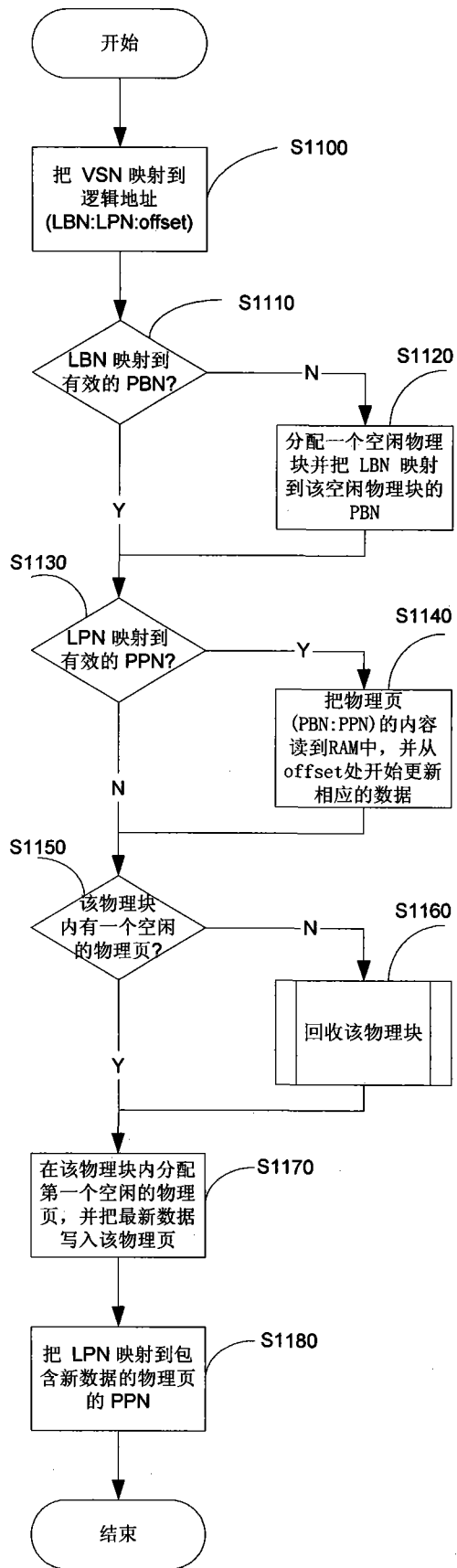


图 11

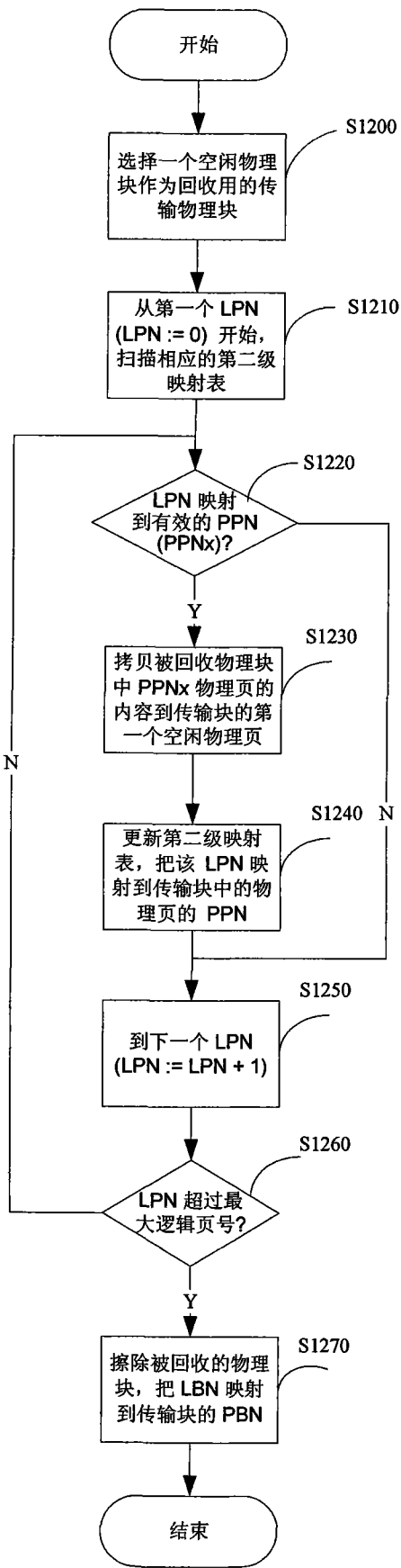


图 12

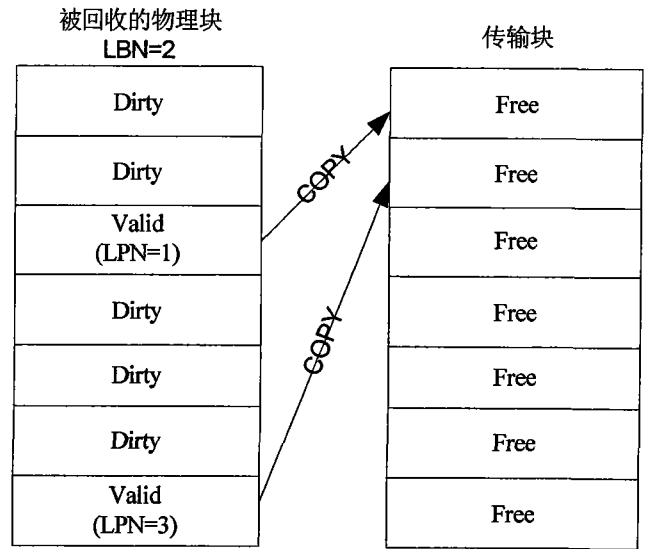


图 13a

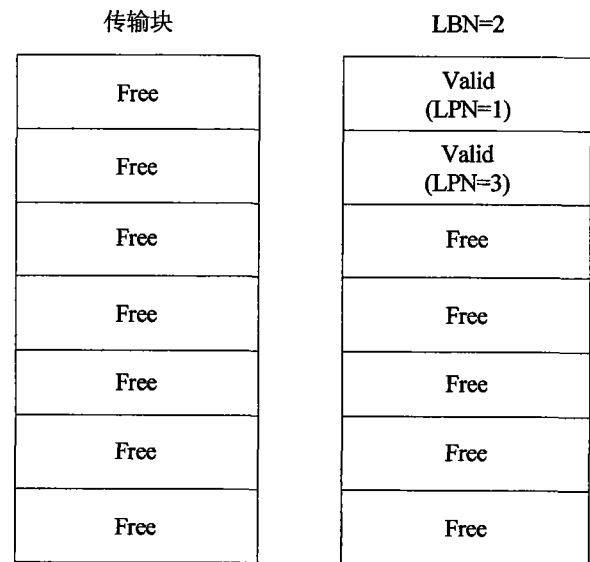


图 13b

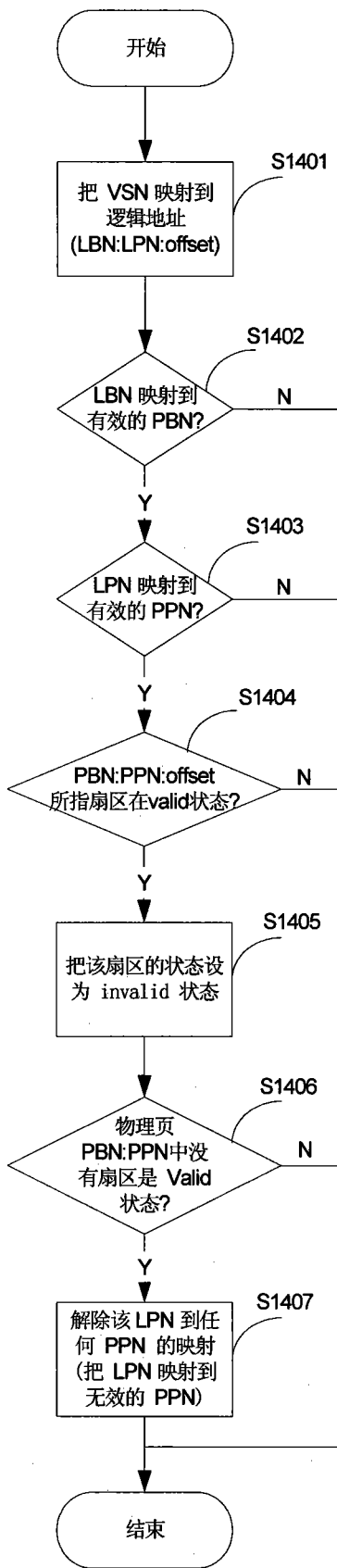


图 14

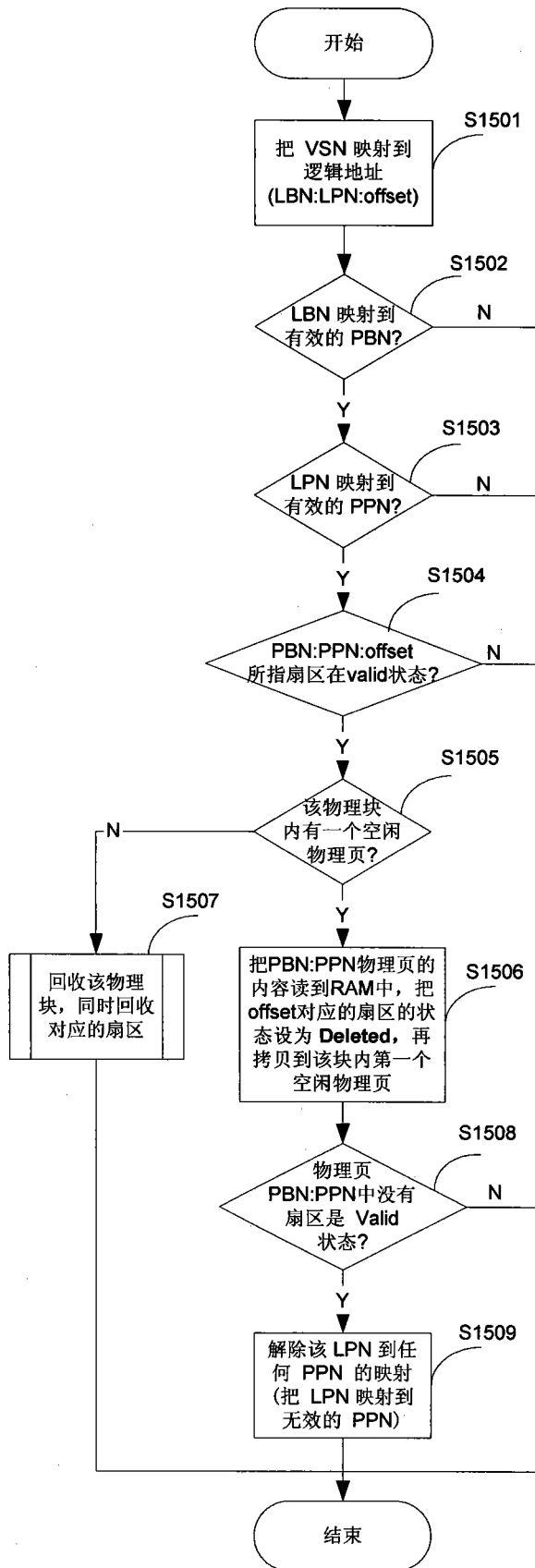


图 15

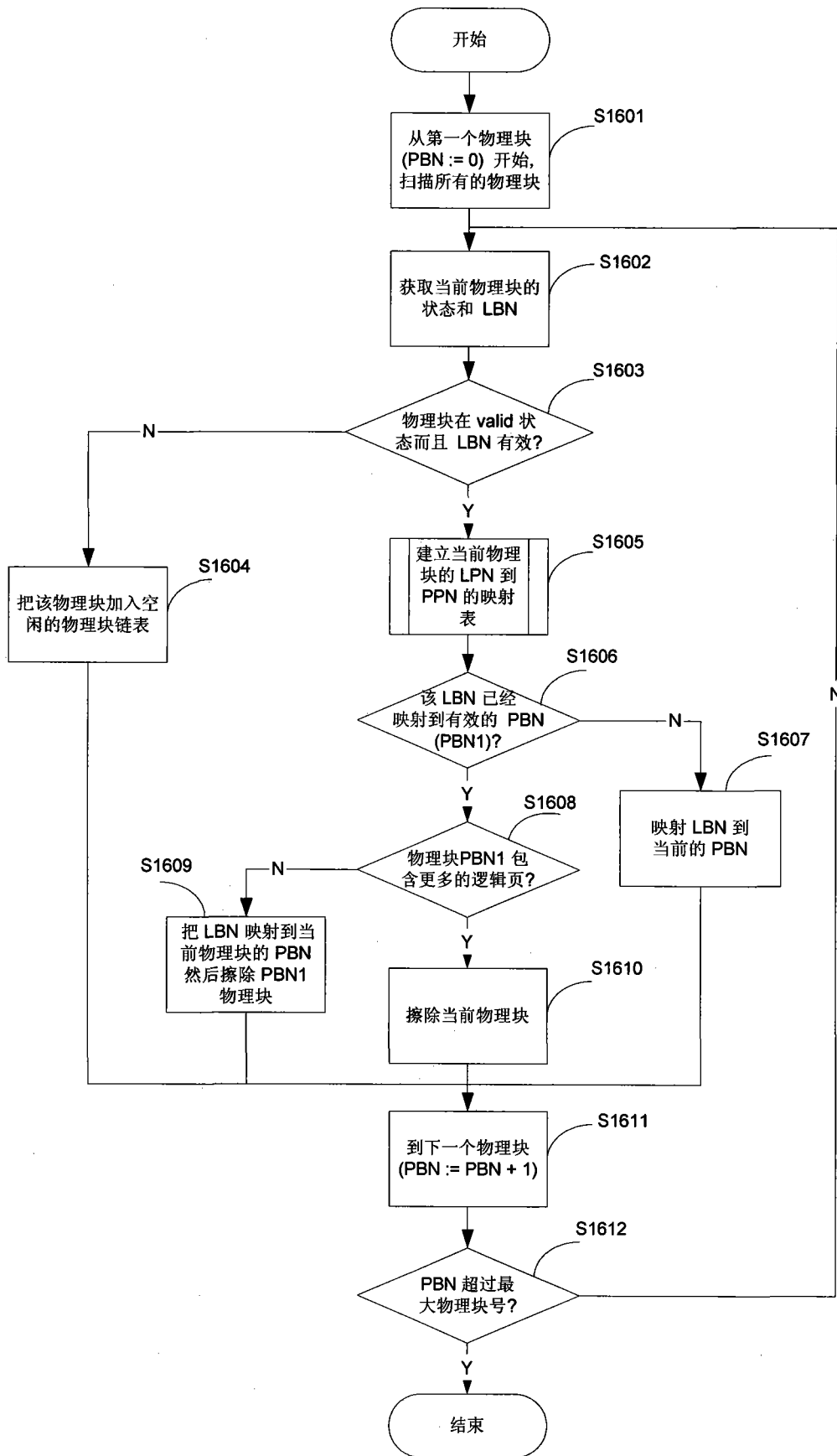


图 16

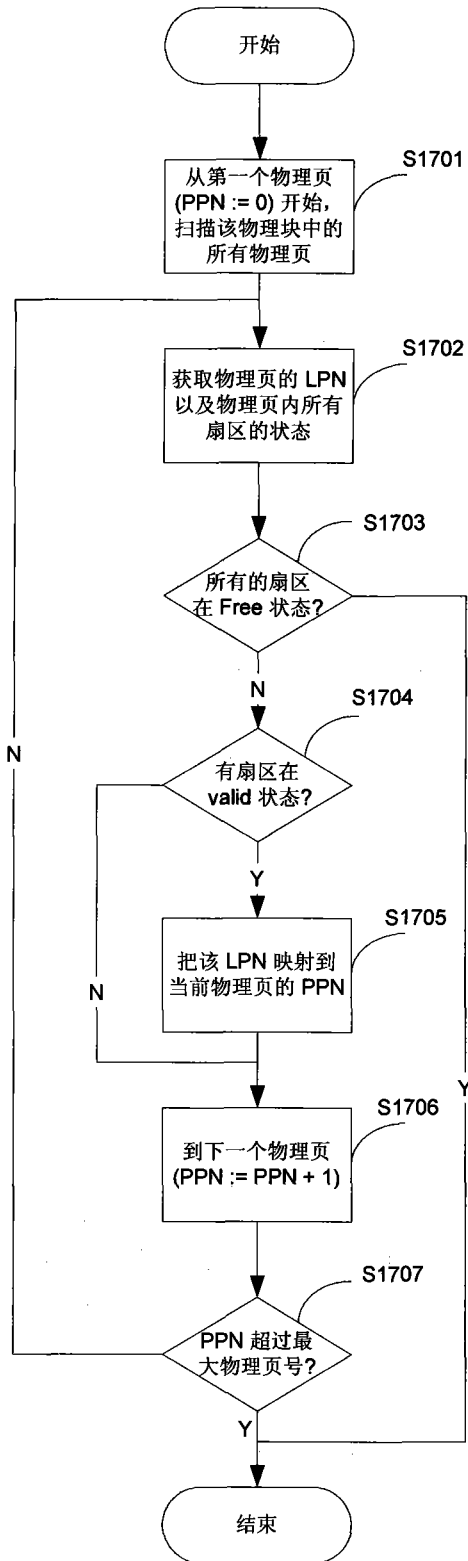


图 17

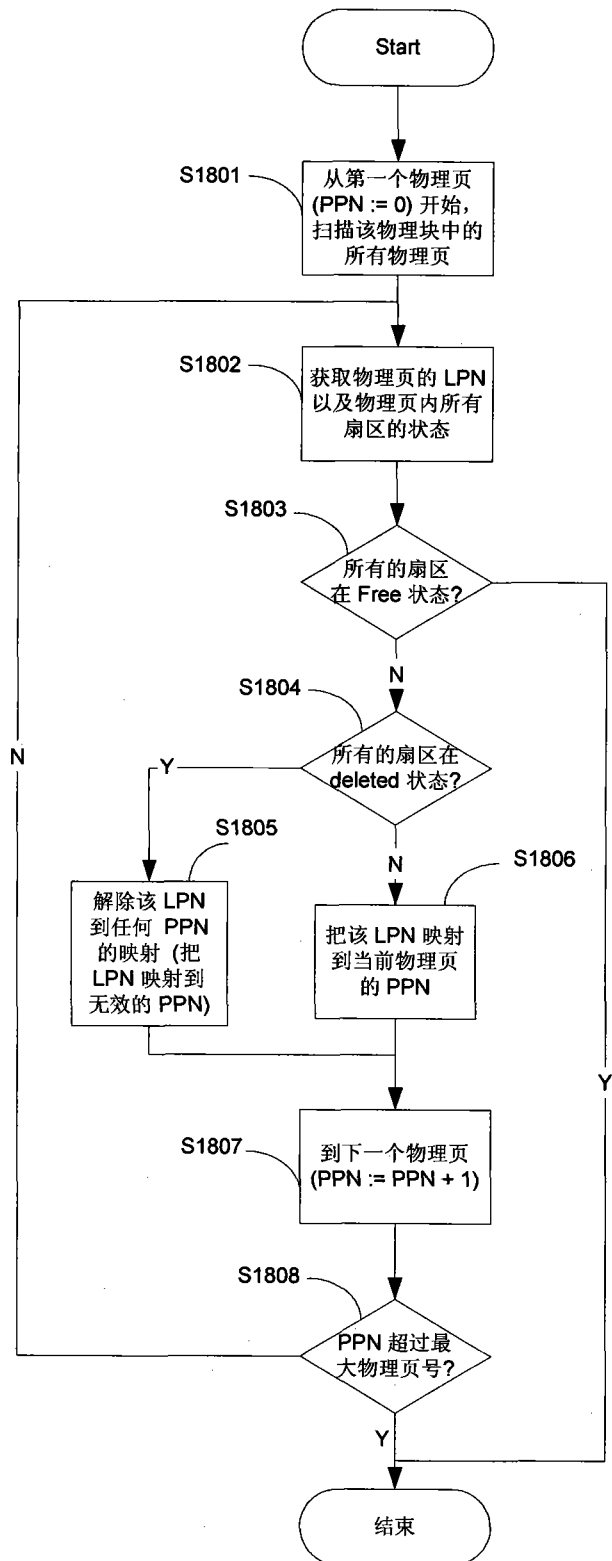


图 18