



(12)发明专利申请

(10)申请公布号 CN 109144888 A

(43)申请公布日 2019.01.04

(21)申请号 201810021540.6

(22)申请日 2018.01.10

(30)优先权数据

2017-126616 2017.06.28 JP

(71)申请人 东芝存储器株式会社

地址 日本东京都

(72)发明人 菅野伸一

(74)专利代理机构 北京市中咨律师事务所

11247

代理人 刘静 段承恩

(51)Int.Cl.

G06F 12/02(2006.01)

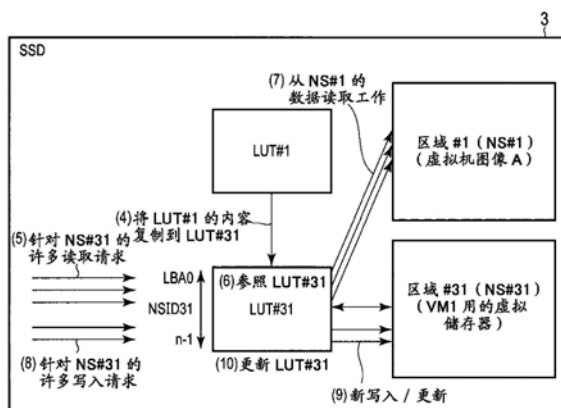
权利要求书4页 说明书23页 附图18页

(54)发明名称

存储器系统

(57)摘要

本发明实现能够经由其他区域的逻辑地址来利用某区域所保存的数据的存储器系统。实施方式... 将第1数据的一部分从第1区域读出并返回给主机计算机。存储器系统从分配给第1区域的所述非易失性存储器的第1区块群中检测满足刷新条件的区块,对所检测出的区块内的数据的错误进行纠正,将错误已被纠正的数据回写到与所述检测出的区块相同的区块。



1. 一种存储器系统,是能够与主机计算机连接的存储器系统,其中,具备:
包括多个区块的非易失性存储器;和
与所述非易失性存储器电连接的控制器,
所述控制器,

管理所述非易失性存储器内的多个区域,所述多个区域包括保存有从其他区域参照的第1数据的第1区域和参照所述第1数据的第2区域,

根据来自所述主机计算机的请求,将与所述第1区域对应的第1逻辑物理地址变换表的内容复制到与所述第2区域对应的第2逻辑物理地址变换表,

根据来自所述主机计算机的指定所述第2区域的逻辑地址的读取请求,基于所述第2逻辑物理地址变换表,将所述第1数据的一部分从所述第1区域读出并返回给所述主机计算机,

根据来自所述主机计算机的指定所述第2区域的逻辑地址的写入请求,向所述第2区域写入应被写入的第2数据,并且更新所述第2逻辑物理地址变换表以使得表示已写入所述第2数据的所述第2区域内的物理存储位置的物理地址映射到由所述写入请求指定的所述逻辑地址,

从分配给所述第1区域的所述非易失性存储器的第1区块群中检测满足刷新条件的区块,对所述检测出的区块内的数据的错误进行纠正,将错误已被纠正的数据回写到与所述检测出的区块相同的区块。

2. 根据权利要求1所述的存储器系统,其中,

所述第1区域保存有操作系统来作为所述第1数据,

所述第2区域作为由所述主机计算机上的虚拟机访问的虚拟存储器来利用,所述虚拟机执行所述操作系统。

3. 根据权利要求1所述的存储器系统,其中,

所述第1区域保存有操作系统以及应用程序来作为所述第1数据,

所述第2区域作为由所述主机计算机上的虚拟机访问的虚拟存储器来利用,所述虚拟机执行所述操作系统以及所述应用程序。

4. 根据权利要求1所述的存储器系统,其中,

所述控制器在所述第2数据已被写入所述第2区域之后从所述主机计算机接收到对所述第2数据的读取请求的情况下,基于所述更新后的第2地址变换表,将所述第2数据从所述第2区域读出并返回给所述主机计算机。

5. 根据权利要求1所述的存储器系统,其中,

所述多个区域还包括参照所述第1数据的第3区域,

所述控制器,

根据来自所述主机计算机的请求,将所述第1逻辑物理地址变换表的内容复制到与所述第3区域对应的第3逻辑物理地址变换表,

根据来自所述主机计算机的指定所述第3区域的逻辑地址的读取请求,基于所述第3逻辑物理地址变换表,将所述第1数据的一部分从所述第1区域读出并返回给所述主机计算机,

根据来自所述主机计算机的指定所述第3区域的逻辑地址的写入请求,向所述第3区域

写入应被写入的第3数据,并且更新所述第3逻辑物理地址变换表以使得表示已写入所述第3数据的所述第3区域内的物理存储位置的物理地址映射到由所述写入请求指定的所述逻辑地址。

6. 根据权利要求1所述的存储器系统,其中,

所述控制器,

根据来自所述主机计算机的请求,向所述第1区域写入所述第1数据,

在所述第1数据的写入完成之后,将所述第1区域设定为禁止向所述第1区域进行写入的只读模式。

7. 根据权利要求1所述的存储器系统,其中,

所述控制器,

在所述检测出的区块的误比特率或读取次数为阈值以上的情况下,将所述错误已被纠正的数据写入到与所述检测出的区块不同的另外的区块,

将所述另外的区块的区块地址保存于区块地址变换表,以使得从所述第1逻辑物理地址变换表或所述第2逻辑物理地址变换表取得的物理地址内所包含的所述检测出的区块的区块地址被变换为所述另外的区块的区块地址,所述区块地址变换表对变换前的各个区块地址与变换后的各个区块地址之间的映射进行管理。

8. 根据权利要求1所述的存储器系统,其中,

分配给所述第1区域的所述第1区块群,是每个存储器单元能够保存一位数据的第1类型的区块群,

分配给所述第2区域的所述非易失性存储器的第2区块群,是每个存储器单元能够保存多位数据的第2类型的区块群。

9. 根据权利要求1所述的存储器系统,其中,

所述控制器,

使用每个存储器单元被写入一位数据的第1写入模式,执行向所述第1区域的数据写入,

使用每个存储器单元被写入多位数据的第2写入模式,执行向所述第2区域的数据写入。

10. 一种存储器系统,是能够与主机计算机连接的存储器系统,其中,具备:

包括多个区块的非易失性存储器;和

与所述非易失性存储器电连接的控制器,

所述控制器,

管理所述非易失性存储器内的多个区域,所述多个区域包括保存有从其他区域参照的第1数据的第1区域和参照所述第1数据的第2区域,

根据来自所述主机计算机的请求,将与所述第1区域对应的第1逻辑物理地址变换表的内容复制到与所述第2区域对应的第2逻辑物理地址变换表,

根据来自所述主机计算机的指定所述第2区域的逻辑地址的读取请求,基于所述第2逻辑物理地址变换表,将所述第1数据的一部分从所述第1区域读出并返回给所述主机计算机,

根据来自所述主机计算机的指定所述第2区域的逻辑地址的写入请求,向所述第2区域

写入应被写入的第2数据,并且更新所述第2逻辑物理地址变换表以使得表示已写入所述第2数据的所述第2区域内的物理存储位置的物理地址映射到由所述写入请求指定的所述逻辑地址,

从分配给所述第1区域的所述非易失性存储器的第1区块群中检测满足刷新条件的区块,对所述检测出的区块内的数据的错误进行纠正,将错误已被纠正的数据写入到与所述检测出的区块不同的另外的区块,

将所述另外的区块的区块地址保存于区块地址变换表,以使得从所述第1逻辑物理地址变换表或所述第2逻辑物理地址变换表取得的物理地址内所包含的所述检测出的区块的区块地址被变换为所述另外的区块的区块地址,所述区块地址变换表对变换前的各个区块地址与变换后的各个区块地址之间的映射进行管理。

11. 根据权利要求10所述的存储器系统,其中,
所述第1区域保存有操作系统来作为所述第1数据,

所述第2区域作为由所述主机计算机上的虚拟机访问的虚拟存储器来利用,所述虚拟机执行所述操作系统。

12. 根据权利要求10所述的存储器系统,其中,
所述第1区域保存有操作系统以及应用程序来作为所述第1数据,

所述第2区域作为由所述主机计算机上的虚拟机访问的虚拟存储器来利用,所述虚拟机执行所述操作系统以及所述应用程序。

13. 一种存储器系统,是能够与包括虚拟机被执行的虚拟环境的服务器连接的存储器系统,其中,具备:

包括多个区块的非易失性存储器;和
与所述非易失性存储器电连接的控制器,
所述控制器,

管理所述非易失性存储器内的多个区域,所述多个区域包括保存有作为所述虚拟机用的客户操作系统来利用的操作系统的第1区域和作为由所述虚拟机访问的虚拟存储器来利用的第2区域,

将与所述第1区域对应的第1逻辑物理地址变换表的内容复制到与所述第2区域对应的第2逻辑物理地址变换表,

根据来自所述虚拟机的指定所述第2区域的逻辑地址的读取请求,基于所述第2逻辑物理地址变换表,将所述操作系统的一部分从所述第1区域读出并返回给所述虚拟机,

根据来自所述虚拟机的指定所述第2区域的逻辑地址的写入请求,向所述第2区域写入应被写入的第2数据,并且更新所述第2逻辑物理地址变换表以使得表示已写入所述第2数据的所述第2区域内的物理存储位置的物理地址映射到由所述写入请求指定的所述逻辑地址,

从分配给所述第1区域的所述非易失性存储器的第1区块群中检测满足刷新条件的区块,对所述检测出的区块内的数据的错误进行纠正,将错误已被纠正的数据回写到与所述检测出的区块相同的区块。

14. 根据权利要求13所述的存储器系统,其中,
所述第1区域保存有所述操作系统以及应用程序,

所述虚拟机执行所述操作系统以及所述应用程序。

15. 根据权利要求13所述的存储器系统,其中,
所述控制器,

在所述检测出的区块的误比特率或读取次数为阈值以上的情况下,将所述错误已被纠正的数据写入到与所述检测出的区块不同的另外的区块,

将所述另外的区块的区块地址保存于区块地址变换表,以使得从所述第1逻辑物理地址变换表或所述第2逻辑物理地址变换表取得的物理地址内所包含的所述检测出的区块的区块地址被变换为所述另外的区块的区块地址,所述区块地址变换表对变换前的各个区块地址与变换后的各个区块地址之间的映射进行管理。

存储器系统

[0001] 关联申请

[0002] 本申请享有以日本专利申请2017-126616号(申请日:2017年6月28日)为基础申请的优先权。本申请通过参照该基础申请而包含基础申请的全部内容。

技术领域

[0003] 本发明的实施方式涉及存储器系统。

背景技术

[0004] 近年来,具备非易失性存储器的存储器系统已被广泛普及。

[0005] 作为这样的存储器系统之一,已知基于NAND闪存技术的固态驱动器(SSD)。

[0006] 在SSD这样的存储器系统中,有时需要将某区域中保存的数据复制到其他区域。

[0007] 但是,这样的复制有时会需要比较多的时间。

发明内容

[0008] 本发明提供能够经由其他区域的逻辑地址来利用某区域中保存的数据的存储器系统。

[0009] 根据实施方式,能够与主机计算机连接的存储器系统具备:包括多个区块的非易失性存储器;和与所述非易失性存储器电连接的控制器。所述控制器管理所述非易失性存储器内的多个区域。所述多个区域包括保存有从其他区域参照的第1数据的第1区域和参照所述第1数据的第2区域。所述控制器根据来自所述主机计算机的请求,将与所述第1区域对应的第1逻辑物理地址变换表的内容复制到与所述第2区域对应的第2逻辑物理地址变换表。所述控制器根据来自所述主机计算机的指定所述第2区域的逻辑地址的读取请求,基于所述第2逻辑物理地址变换表,将所述第1数据的一部分从所述第1区域读出并返回给所述主机计算机。所述控制器根据来自所述主机计算机的指定所述第2区域的逻辑地址的写入请求,向所述第2区域写入应被写入的第2数据,并且更新所述第2逻辑物理地址变换表以使得表示已写入所述第2数据的所述第2区域内的物理存储位置的物理地址映射到由所述写入请求指定的所述逻辑地址。所述控制器从分配给所述第1区域的所述非易失性存储器的第1区块群中检测满足刷新条件的区块,对所述检测出的区块内的数据的错误进行纠正,将错误已被纠正的数据回写到与所述检测出的区块相同的区块。

附图说明

[0010] 图1是表示服务器(主机计算机)的构成例的框图。

[0011] 图2是用于说明在图1的服务器的各虚拟机所设置的访问目的地切换部的工作的框图。

[0012] 图3是用于说明通过对实施方式的存储器系统的非易失性存储器进行逻辑分割而得到的多个区域的图。

[0013] 图4是表示该实施方式的存储器系统内的多个区域、与这些区域对应的逻辑地址空间、和与这些区域对应的查找表的关系的框图。

[0014] 图5是用于说明向该实施方式的存储器系统内的区域#1写入操作系统的二值图像的工作和将区域#1设定为禁止写入模式(只读模式)的工作的图。

[0015] 图6是用于说明对该实施方式的存储器系统内的区域#31进行控制的工作的图。

[0016] 图7是用于说明对该实施方式的存储器系统内的区域#32进行控制的工作的图。

[0017] 图8是用于说明对区域#1的刷新工作的例子的图。

[0018] 图9是表示与区域#1对应的查找表#1的内容的例子和与区域#31对应的查找表#31的内容的例子图。

[0019] 图10是用于说明根据指定区域#31(或区域#32)的某逻辑地址的读取请求而执行的读取工作和根据指定区域#31(或区域#32)的某逻辑地址的写入请求而执行的写入工作的图。

[0020] 图11是表示分配给区域#1的SLC区块群和分配给区域#31(或区域#32)的MLC/TLC区块群的图。

[0021] 图12是用于说明对区域#1的刷新工作的另一例的图。

[0022] 图13是用于说明由与区域#1对应的查找表和与区域#31对应的查找表(或与区域#32对应的查找表)共用的区块地址变换表的图。

[0023] 图14是表示包括主机计算机和该实施方式的存储器系统的信息处理系统的构成例的框图。

[0024] 图15是表示该实施方式的存储器系统内的NAND接口与多个NAND型闪速存储器裸片的关系的图。

[0025] 图16是表示由该实施方式的存储器系统执行的向区域#1写入虚拟机图像的处理步骤的时序图。

[0026] 图17是表示由该实施方式的存储器系统执行的创建虚拟机用的虚拟存储器(区域#31或区域#32)的处理步骤的时序图。

[0027] 图18是表示由该实施方式的存储器系统执行的读/写处理的步骤的流程图。

[0028] 图19是表示由该实施方式的存储器系统执行的刷新处理的步骤的流程图。

[0029] 图20是表示由该实施方式的存储器系统执行的刷新处理的另一步骤的流程图。

[0030] 图21是表示由该实施方式的存储器系统执行的数据写入处理的步骤的流程图。

[0031] 图22是表示包括该实施方式的存储器系统的服务器(主机计算机)的构成例的图。

具体实施方式

[0032] 以下,参照附图来说明实施方式。

[0033] 图1表示能适用一实施方式的存储器系统的计算机的构成例。该计算机是服务器、个人计算机这样的执行各种各样的程序的信息处理装置。以下,设想该计算机作为服务器100来实现的情况。

[0034] 服务器100是经由网络向多个用户(多个终端)提供各种各样的服务的计算机(也称为“物理服务器”)。

[0035] 该服务器100也可以设置在数据中心内。对于由该服务器100提供的服务的例子,

存在将虚拟服务器这样的基础架构向许多用户(许多承租人)提供的基础架构即服务(IaaS)等。

[0036] 服务器100包括多个虚拟机被执行的虚拟环境,在该服务器100上执行这些多个虚拟机。图1中例示了在服务器100上执行3个虚拟机VM1、VM2、VM3的情况。

[0037] 这些虚拟机VM1、VM2、VM3作为构成为向与虚拟机VM1、VM2、VM3分别对应的用户A、用户B、用户C提供各种服务的虚拟服务器发挥功能。

[0038] 这些虚拟机VM1、VM2、VM3共用服务器100内的物理资源101。物理资源101通常包括一个以上的物理CPU、包含一个以上的物理存储器模块的物理存储器、一个以上的物理存储器(storage)、一个以上的物理网络接口卡(物理NIC)等。各物理存储器也可以是利用NAND型闪存存储器这样的非易失性存储器来作为存储介质的存储器设备。该存储器设备也可以是固态硬盘(SSD),还可以是包括各自包含非易失性存储器的多个存储器在内的闪存阵列。

[0039] 各虚拟机VM也可以由虚拟CPU、虚拟存储器、虚拟存储器(也称为虚拟盘)以及虚拟NIC来构成。

[0040] 在图1中,例示了如下情况:向用户A提供的虚拟机VM1包括虚拟CPU201、虚拟存储器202、虚拟存储器203、虚拟NIC204,向用户B提供的虚拟机VM2包括虚拟CPU211、虚拟存储器212、虚拟存储器213、虚拟NIC214,向用户C提供的虚拟机VM3包括虚拟CPU221、虚拟存储器222、虚拟存储器223、虚拟NIC224。

[0041] 希望利用虚拟机(虚拟服务器)的用户,能够指定想要利用的虚拟机(虚拟服务器)的性能、属性等。更详细而言,用户能够指定所希望的性能(例如,CPU速度、CPU核数量)、所希望的存储器(DRAM)容量、所希望的网络性能(例如NIC的数量)、所希望的存储器容量、应使用的操作系统的种类、应使用的应用程序的种类等。

[0042] 例如,如果在虚拟机VM1创建时用户A选择了某种类的操作系统的二值图像安装于虚拟机VM1用的虚拟存储器203。

[0043] 但是,通常,向虚拟存储器203安装操作系统需要比较多的时间。

[0044] 因此,在服务器100中,也可以预先准备几种虚拟机图像来作为模板。

[0045] 在图1中,例示了如下情况:预先准备了安装有某种类的操作系统的盘115和安装有另一种类的操作系统的盘116。来作为虚拟机图像A(即模板)的盘115和安装有另一种类的操作系统的盘116。来作为虚拟机图像B(即另一模板)的盘116。

[0046] 如果用户A选择了虚拟机图像A,则操作系统(OS#1)的二值图像、即操作系统(OS#1)的安装图像被从盘115复制到虚拟机VM1的虚拟存储器203。

[0047] 虚拟机图像A不仅包含操作系统(OS#1)的安装图像,还可以包含几个应用程序(APL#1、#2、#3)的安装图像。

[0048] 虚拟机图像B也不仅包含操作系统(OS#2)的安装图像,还可以包含几个应用程序(APL#4、#5、#6)的安装图像。

[0049] 各用户能够从作为模板准备的多个虚拟机图像(在此为虚拟机图像A、B)中选择任意的虚拟机图像。

[0050] 例如,如果用户A选择了虚拟机图像A,则虚拟机图像A的复制体(操作系统(OS#1)、应用程序(APL#1、#2、#3))被从盘115复制到虚拟机VM1的虚拟存储器203。已复制到虚拟储

存器203的操作系统(OS#1),作为虚拟机VM1用的客户操作系统而执行。换言之,虚拟机VM1访问虚拟储存器203,执行虚拟储存器203内的操作系统(OS#1)。

[0051] 同样地,如果用户B选择了虚拟机图像A,则虚拟机图像A的复制体(操作系统(OS#1)、应用程序(APL#1、#2、#3))被从盘115复制到虚拟机VM2的虚拟储存器213。已复制到虚拟储存器213的操作系统(OS#1),作为虚拟机VM2用的客户操作系统而执行。换言之,虚拟机VM2访问虚拟储存器213,执行虚拟储存器213内的操作系统(OS#1)。

[0052] 如果用户C选择了虚拟机图像B,则虚拟机图像B的复制体(操作系统(OS#2)、应用程序(APL#4、#5、#6))被从盘116复制到虚拟机VM3的虚拟储存器223。已复制到虚拟储存器223的操作系统(OS#2),作为虚拟机VM3用的客户操作系统而执行。换言之,虚拟机VM3访问虚拟储存器223,执行虚拟储存器223内的操作系统(OS#2)。

[0053] 如此,通过利用向虚拟储存器复制虚拟机图像的复制体这一结构,相比于进行将操作系统实际安装于虚拟储存器的作业的情况,能够以较短的时间完成虚拟机VM的设置(set up)。

[0054] 但是,通常,各虚拟机图像具有例如数10G字节等比较大的尺寸。因此,将虚拟机图像的复制体从盘115(或盘116)复制到虚拟储存器也需要某种程度的时间。

[0055] 因此,要求用于进一步缩短从某用户签订虚拟机的租赁服务合同到能够实际利用该虚拟机为止的时间的新功能。

[0056] 因此,如图2所示,也可以将各虚拟机包括访问目的地切换部的构成适用于图1的服务器100。

[0057] 在图2的服务器100的构成中,虚拟机VM1包括访问目的地切换部205。同样地,虚拟机VM2包括访问目的地切换部215,虚拟机VM3包括访问目的地切换部225。

[0058] 各访问目的地切换部在由对应的虚拟CPU请求的数据不存在于对应的虚拟储存器的情况下,取代访问该虚拟储存器而访问盘115或盘116。

[0059] 例如,在虚拟机VM1中,当在虚拟储存器203内不存在访问对象的数据(例如,操作系统(OS#1)的二值图像)的情况下,访问目的地切换部205根据来自虚拟CPU201的读取请求,从盘115中读出访问对象的数据(例如,操作系统(OS#1)的二值图像的一部分)。

[0060] 由此,不用将盘115内的数据复制到虚拟储存器,而能够利用盘115内的数据。换言之,不用将操作系统的二值图像复制到虚拟储存器,而能够立即开始虚拟机的利用。

[0061] 在本实施方式中,能够通过与服务服务器100连接的存储器系统来提供与盘115(或盘116)、虚拟储存器所分别对应的多个区域、以及访问目的地切换部对应的功能。

[0062] 该存储器系统是构成为向非易失性存储器写入数据、从非易失性存储器读出数据的半导体储存器设备。该存储器系统既可以是固态驱动器(SSD),也可以是包括各自包含非易失性存储器的多个储存器在内的闪存阵列。以下,并不限于该例,也设想如下情况:存储器系统如图3所示,例如作为基于NAND闪存技术的固态驱动器(SSD)3来实现。

[0063] 该SSD3在图2的服务器100中作为该服务器100的物理储存器而适用。服务器100作为向SSD3送出读取请求、写入请求、其他各种各样的请求的主机计算机(也称为“主机”或“主机设备”)发挥功能。

[0064] SSD3包括非易失性存储器,管理该非易失性存储器内的多个区域。例如,在SSD3中,非易失性存储器也可以在逻辑上分割为区域#1、区域#2、……区域#31、区域#32、区域#

33。

[0065] 区域#1是用于保存由其他区域参照的数据(original data:原创数据)的区域。

[0066] 原创数据可以是上述的虚拟机图像A。该情况下,在该区域#1中,至少保存操作系统(OS#1)的二值图像。该操作系统(OS#1)作为某一个以上的虚拟机用的客户操作系统来利用。进而,应用程序(APL#1、#2、#3)的二值图像也可以保存于区域#1。

[0067] 区域#2也是用于保存由其他区域参照的另一数据(另一原创数据)的区域。

[0068] 另一原创数据可以是上述的虚拟机图像B。该情况下,在该区域#2中,至少保存操作系统(OS#2)的二值图像。该操作系统(OS#2)也作为某一个以上的虚拟机用的客户操作系统来利用。进而,应用程序(APL#4、#5、#6)的二值图像也可以保存于区域#2。

[0069] 区域#31是参照区域#1的数据(原创数据)的区域。

[0070] 在区域#1保存有操作系统(OS#1)作为原创数据的情况下,区域#31作为由执行操作系统(OS#1)的虚拟机VM1访问的虚拟储存器(虚拟盘)来利用。

[0071] 在区域#1保存有操作系统(OS#1)以及应用程序(APL#1、#2、#3)作为原创数据的情况下,区域#31作为由执行操作系统(OS#1)以及应用程序(APL#1、#2、#3)的虚拟机VM1访问的虚拟储存器(虚拟盘)来利用。

[0072] 区域#32也是参照区域#1的数据(原创数据)的区域。

[0073] 在区域#1保存有操作系统(OS#1)作为原创数据的情况下,区域#32作为由执行操作系统(OS#1)的虚拟机VM2访问的虚拟储存器(虚拟盘)来利用。

[0074] 在区域#1保存有操作系统(OS#1)以及应用程序(APL#1、#2、#3)作为原创数据的情况下,区域#32作为由执行操作系统(OS#1)以及应用程序(APL#1、#2、#3)的虚拟机VM2访问的虚拟储存器(虚拟盘)来利用。

[0075] 如此,存在于区域#1的原创数据(模板)能够由其他多个区域(在此为区域#31、区域#32)共用。

[0076] 区域#33是参照区域#2的数据(另一原创数据)的区域。

[0077] 在区域#2保存有操作系统(OS#2)作为另一原创数据的情况下,区域#33作为由执行操作系统(OS#2)的虚拟机VM3访问的虚拟储存器(虚拟盘)来利用。

[0078] 在区域#2保存有操作系统(OS#2)以及应用程序(APL#4、#5、#6)作为原创数据的情况下,区域#33作为由执行操作系统(OS#2)以及应用程序(APL#4、#5、#6)的虚拟机VM3访问的虚拟储存器(虚拟盘)来利用。

[0079] 对这些多个区域(区域#1、区域#2、……区域#31、区域#32、区域#33)分别分配多个逻辑地址空间。作为逻辑地址,也可以使用逻辑区块地址(LBA)。

[0080] 虚拟机VM1经由区域#31的逻辑地址空间来访问区域#31。虚拟机VM2经由区域#32的逻辑地址空间来访问区域#32。虚拟机VM3经由区域#33的逻辑地址空间来访问区域#33。

[0081] 各区域也可以通过命名空间来实现。该情况下,为了将NAND型闪速存储器这样的非易失性存储器在逻辑上分割为多个区域,可利用多个命名空间。各命名空间是非易失性存储器内的一种区域(存储区域),被分配有逻辑地址空间(LBA空间)。

[0082] 各个命名空间通过这些命名空间的标识符来识别。在各区域通过命名空间来实现的情况下,在各区域分配有LBA范围(LBA0~LBA_{n-1})。LBA范围的尺寸(即LBA的个数)按每个区域(命名空间)而可变。各LBA范围从LBA0开始。

[0083] 图4表示多个区域、与这些区域对应的逻辑地址空间 (LBA空间) 以及与这些区域对应的查找表的关系。

[0084] 在与命名空间NS#1对应的区域#1, 分配有0~n-1的逻辑地址空间 (LBA空间) A1。在与命名空间NS#2对应的区域#2, 分配有0~n-1的逻辑地址空间 (LBA空间) A2。在与命名空间NS#31对应的区域#31, 分配有0~n-1的逻辑地址空间 (LBA空间) A31。在与命名空间NS#32对应的区域#32, 分配有0~n-1的逻辑地址空间 (LBA空间) A32。在与命名空间NS#33对应的区域#33, 分配有0~n-1的逻辑地址空间 (LBA空间) A33。

[0085] 在本实施方式中, 查找表LUT按每个区域即按每个命名空间来分割。也就是说, 与区域#1、区域#2、……区域#31、区域#32、区域#33分别对应的查找表LUT#1、LUT#2、……LUT#31、LUT#32、LUT#33, 通过SSD3来管理。

[0086] 查找表LUT#1是用于对与区域#1的逻辑地址 (LBA) 分别对应的非易失性存储器内的物理存储位置的物理地址分别进行管理的逻辑物理地址变换表 (logical-to-physical address translation table)。该查找表LUT#1被用于管理区域#1的各个逻辑地址 (LBA) 与分配给区域#1的非易失性存储器内的物理存储位置的各个物理地址之间的映射。

[0087] 在从服务器100 (主机计算机) 接收到指定区域#1的某逻辑地址 (某LBA) 的写入请求 (写入命令) 的情况下, SSD3向区域#1写入应写入的数据 (写入数据)。并且, SSD3更新查找表LUT#1, 以使得表示已写入了写入数据的区域#1内的物理存储位置的物理地址映射到该LBA。

[0088] 在从服务器100 (主机计算机) 接收到指定区域#1的某逻辑地址 (某LBA) 的读取请求 (读取命令) 的情况下, SSD3参照查找表LUT#1, 从查找表LUT#1取得与该LBA对应的物理地址。并且, SSD3从由该取得的物理地址指定的物理存储位置读出数据。

[0089] 另外, 查找表LUT#1在用于增加区域#1的空闲空间 (free space) 的垃圾回收中也进行利用。在区域#1的垃圾回收中, SSD3从分配给该区域#1的非易失性存储器内的各个区块中选择垃圾回收对象的一个以上的区块, 将该选择出的各区块内的有效数据复制到被分配给该区域#1的另外的区块 (空闲区块)。并且, SSD3更新查找表LUT#1, 以使得向与已被复制的数据对应的LBA映射表示复制目的地物理存储位置的物理地址。

[0090] 查找表LUT#2是用于管理与区域#2的逻辑地址 (LBA) 分别对应的非易失性存储器内的物理存储位置的各个物理地址的逻辑物理地址变换表。该查找表LUT#2被用于管理区域#2的各个逻辑地址 (LBA) 与被分配给区域#2的非易失性存储器内的物理存储位置的各个物理地址之间的映射。

[0091] 在从服务器100 (主机计算机) 接收到指定区域#2的某逻辑地址 (某LBA) 的写入请求 (写入命令) 的情况下, SSD3向区域#2写入应写入的数据 (写入数据)。并且, SSD3更新查找表LUT#2, 以使得表示已写入了写入数据的区域#2内的物理存储位置的物理地址映射到该LBA。

[0092] 在从服务器100 (主机计算机) 接收到指定区域#2的某逻辑地址 (某LBA) 的读取请求 (读取命令) 的情况下, SSD3参照查找表LUT#2, 从查找表LUT#2取得与该LBA对应的物理地址。并且, SSD3从由该取得的物理地址指定的物理存储位置读出数据。

[0093] 另外, 查找表LUT#2在用于增加区域#2的空闲空间的垃圾回收中也进行利用。在区域#2的垃圾回收中, SSD3从被分配给该区域#2的非易失性存储器内的各个区块中选择垃圾

回收对象的一个以上的区块,将该选择出的各区块内的有效数据复制到被分配给该区域#2的另外的区块(空闲区块)。并且,SSD3更新查找表LUT#2,以使得向与已被复制的数据对应的LBA分别映射表示复制目的地物理存储位置的物理地址。

[0094] 查找表LUT#31是用于管理与区域#31的逻辑地址(LBA)分别对应的非易失性存储器内的物理存储位置的各个物理地址的逻辑物理地址变换表。该查找表LUT#31被用于管理区域#31的各个逻辑地址(LBA)与被分配给区域#31的非易失性存储器内的物理存储位置的各个物理地址之间的映射。

[0095] 在从服务器100(主机计算机的虚拟机VM1)接收到指定区域#31的某逻辑地址(某LBA)的读取请求(读取命令)的情况下,SSD3参照查找表LUT#31,从查找表LUT#31取得与该LBA对应的物理地址。并且,SSD3从由该取得的物理地址指定的物理存储位置读出数据。

[0096] 在本实施方式中,为了实现虚拟机VM1内的上述的访问目的地切换部205,在虚拟机VM1创建时,查找表LUT#1的内容被复制到查找表LUT#31。因此,在虚拟机VM1刚创建之后的初始状态,查找表LUT#31保持与查找表LUT#1的内容相同的内容。也就是说,查找表LUT#31内的各个物理地址不是区域#31内的物理存储位置而是指区域#1内的物理存储位置。因此,向区域#31的各逻辑地址的读取访问,被改朝向区域#1。

[0097] 因此,在从服务器100(虚拟机VM1)接收到指定区域#31的某逻辑地址(某LBA)的读取请求(读取命令)的情况下,SSD3基于查找表LUT#31,将原创数据(例如,操作系统(OS#1)的一部分)从区域#1读出并返回给服务器100(虚拟机VM1)。因此,不用将区域#1内的原创数据(例如,操作系统(OS#1)等)复制到区域#31,而能够经由区域#31的逻辑地址空间来利用区域#1内所保存的全部原创数据。

[0098] 在从服务器100(主机计算机的虚拟机VM1)接收到指定区域#31的某逻辑地址(某LBA)的写入请求(写入命令)的情况下,SSD3向区域#31(被分配给区域#31的区块)写入应写入的数据(写入数据)。并且,SSD3更新查找表LUT#31,以使得表示已写入了写入数据的区域#31内的物理存储位置的物理地址映射到该LBA。

[0099] 如此,向区域#31的数据写入,并不改朝向区域#1,而是向区域#31写入数据。因此,区域#1内的原创数据不会被更新,仅是区域#31的数据被更新。区域#1内的原创数据继续维持原来的状态。由此,例如即使在区域#1内的原创数据(例如,虚拟机图像的模板)由多个用户参照(共用)的情况下,也不会发生与某用户对应的区域内的数据更新对与其他用户对应的区域的数据产生影响这一矛盾(问题)。

[0100] 在查找表LUT#31中,在与该已被写入的数据对应的逻辑地址映射了区域#31内的物理存储位置的物理地址。因此,指定该逻辑地址的后续的读取请求,并不改朝向区域#1,而是从区域#31读出该已被写入的数据。

[0101] 因此,能够从区域#1读出不存在于区域#31的数据(原创数据,例如操作系统(OS#1)),并且能够将新的写入数据或用于改写原创数据的一部分的更新数据写入到区域#31。例如,在需要对由虚拟机VM1执行的操作系统(OS#1)的一部分(文件等)进行升级(update)的情况下,操作系统(OS#1)的更新数据部分、即已被更新的文件被写入到区域#31。另外,通过由虚拟机VM1执行的应用而创建出的用户数据也被写入到区域#31。

[0102] 另外,当向区域#31写入了数据时,在与该数据对应的逻辑地址,映射了区域#31内的物理存储位置的物理地址。因此,当在某数据被写入到区域#31之后从服务器100(主机计

算机的虚拟机VM1) 接收到指定该数据的逻辑地址的后续的读取请求(读取命令)的情况下, SSD3基于查找表LUT#31, 将该数据从区域#31读出并返回给服务器100(虚拟机VM1)。由此, 能够从区域#31正确地读出新写入到区域#31的数据以及区域#31内的更新后的数据。

[0103] 查找表LUT#31在用于增加区域#31的空闲空间的垃圾回收中也进行利用。在区域#31的垃圾回收中, SSD3从被分配给该区域#31的非易失性存储器内的各个区块中选择垃圾回收对象的一个以上的区块, 将该选择出的各区块内的有效数据复制到被分配给该区域#31的另外的区块(空闲区块)。并且, SSD3更新查找表LUT#31, 以使得向与已被复制的数据对应的LBA分别映射表示复制目的地物理存储位置的物理地址。

[0104] 查找表LUT#32是用于管理与区域#32的逻辑地址(LBA) 分别对应的非易失性存储器内的物理存储位置的各个物理地址的逻辑物理地址变换表。该查找表LUT#32被用于管理区域#32的各个逻辑地址(LBA) 与被分配给区域#32的非易失性存储器内的物理存储位置的各个物理地址之间的映射。

[0105] 在从服务器100(主机计算机的虚拟机VM2) 接收到指定区域#32的某逻辑地址(某LBA) 的读取请求(读取命令)的情况下, SSD3参照查找表LUT#32, 从查找表LUT#32取得与该LBA对应的物理地址。并且, SSD3从由该取得的物理地址指定的物理存储位置读出数据。

[0106] 为了实现虚拟机VM2内的上述的访问目的地切换部215, 在虚拟机VM2创建时, 查找表LUT#1的内容被复制到查找表LUT#32。因此, 在虚拟机VM2的初始状态下, 查找表LUT#32也保持与查找表LUT#1的内容相同的内容。也就是说, 查找表LUT#32内的各个物理地址不是区域#32内的物理存储位置, 而是指区域#1内的物理存储位置。因此, 向区域#32的读取访问被改朝向区域#1。

[0107] 因此, 在从服务器100(主机计算机的虚拟机VM2) 接收到指定区域#32的某逻辑地址(某LBA) 的读取请求(读取命令)的情况下, SSD3基于查找表LUT#32, 将原创数据(例如, 操作系统(OS#1)的一部分)从区域#1读出并返回给服务器100(虚拟机VM2)。因此, 不用将区域#1内的原创数据(例如, 操作系统(OS#1)等)复制到区域#32, 而能够经由区域#32的逻辑地址空间来利用区域#1内的全部原创数据。

[0108] 在从服务器100(主机计算机的虚拟机VM1) 接收到指定区域#32的某逻辑地址(某LBA) 的写入请求(写入命令)的情况下, SSD3向区域#32(已分配给区域#32的区块)写入应写入的数据(写入数据)。并且, SSD3更新查找表LUT#32, 以使得表示已写入了写入数据的区域#32内的物理存储位置的物理地址映射到该LBA。

[0109] 如此, 向区域#32的数据写入并不改朝向区域#1, 而是向区域#32写入数据。在查找表LUT#32中, 在与该已被写入的数据对应的逻辑地址, 映射了区域#32内的物理存储位置的物理地址。因此, 指定该逻辑地址的后续的读取请求并不改朝向区域#1, 而是从区域#32读出该已被写入的数据。

[0110] 因此, 能够从区域#1读出不存在于区域#32的数据(原创数据, 例如操作系统(OS#1)), 并且能够向区域#32写入新的写入数据或用于改写原创数据的一部分的更新数据。例如, 在需要对由虚拟机VM2执行的操作系统(OS#1)的一部分(文件等)进行升级的情况下, 操作系统(OS#1)的更新数据部分、即已被更新的文件被写入到区域#32。另外, 通过由虚拟机VM2执行的应用而创建出的用户数据也被写入到区域#32。

[0111] 另外, 当向区域#32写入了数据时, 在与该数据对应的逻辑地址, 映射了区域#32内

的物理存储位置的物理地址。因此,在某数据已被写入到区域#32之后从服务器100(主机计算机的虚拟机VM2)接收到指定该数据的逻辑地址的后续的读取请求(读取命令)的情况下,SSD3基于查找表LUT#32,将该数据从区域#32读出并返回给服务器100(虚拟机VM2)。

[0112] 查找表LUT#32在用于增加区域#32的空闲空间的垃圾回收中也进行利用。在区域#32的垃圾回收中,SSD3从被分配给该区域#32的非易失性存储器内的各个区块中选择垃圾回收对象的一个以上的区块,将该选择出的各区块内的有效数据复制到已被分配给该区域#32的另外的区块(空闲区块)。并且,SSD3更新查找表LUT#32,以使得向与已被复制的数据对应的LBA分别映射表示复制目的地物理存储位置的物理地址。

[0113] 查找表LUT#33是用于管理与区域#33的逻辑地址(LBA)分别对应的非易失性存储器内的物理存储位置的各个物理地址的逻辑物理地址变换表。该查找表LUT#33被用于管理区域#33的各个逻辑地址(LBA)与被分配给区域#33的非易失性存储器内的物理存储位置的各个物理地址之间的映射。

[0114] 在从服务器100(主机计算机的虚拟机VM3)接收到指定区域#33的某逻辑地址(某LBA)的读取请求(读取命令)的情况下,SSD3参照查找表LUT#33,从查找表LUT#33取得与该LBA对应的物理地址。并且,SSD3从由该取得的物理地址指定的物理存储位置读出数据。

[0115] 为了实现虚拟机VM3内的上述的访问目的地切换部225,在虚拟机VM3创建时,查找表LUT#2的内容被复制到查找表LUT#33。因此,在虚拟机VM3的初始状态下,查找表LUT#33保持与查找表LUT#2的内容相同的内容。也就是说,查找表LUT#33内的各个物理地址不是区域#33内的物理存储位置,而是指区域#2内的物理存储位置。因此,向区域#33的读取访问被改朝向区域#2。

[0116] 因此,在从服务器100(主机计算机的虚拟机VM3)接收到指定区域#33的某逻辑地址(某LBA)的读取请求(读取命令)的情况下,SSD3基于查找表LUT#33,将另一原创数据(例如,操作系统(OS#2)的一部分)从区域#2读出并返回给服务器100(虚拟机VM3)。因此,不用将区域#2内的原创数据(例如,操作系统(OS#2)等)复制到区域#33,而能够经由区域#33的逻辑地址空间来利用区域#2内的全部原创数据。

[0117] 在从服务器100(主机计算机的虚拟机VM3)接收到指定区域#33的某逻辑地址(某LBA)的写入请求(写入命令)的情况下,SSD3向区域#33(已分配给区域#33的区块)写入应写入的数据(写入数据)。并且,SSD3更新查找表LUT#33,以使得表示已写入了写入数据的区域#33内的物理存储位置的物理地址映射到该LBA。

[0118] 如此,向区域#33的数据写入,并不改朝向区域#2,而是向区域#33写入数据。在查找表LUT#33中,在与该已被写入的数据对应的逻辑地址,映射了区域#33内的物理存储位置的物理地址。因此,指定该逻辑地址的后续的读取请求,并不改朝向区域#2,而是从区域#33读出该已被写入的数据。

[0119] 因此,能够从区域#2读出不存在于区域#33的数据(另一原创数据,例如操作系统(OS#2)),并且能够向区域#33写入新的写入数据或者用于改写另一原创数据的一部分的更新数据。例如,在需要对由虚拟机VM3执行的操作系统(OS#2)的一部分(文件等)进行升级的情况下,操作系统(OS#2)的更新数据部分、即更新后的文件被写入到区域#33。另外,通过由虚拟机VM3执行的应用而创建出的用户数据也被写入到区域#33。

[0120] 另外,当向区域#33写入了数据时,在与该数据对应的逻辑地址,映射了区域#33内

的物理存储位置的物理地址。因此,当在某数据已被写入到区域#33之后从服务器100(主机计算机的虚拟机VM3)接收到指定该数据的逻辑地址的后续的读取请求(读取命令)的情况下,SSD3基于查找表LUT#33,将该数据从区域#33读出并返回给服务器100(虚拟机VM3)。

[0121] 查找表LUT#33在用于增加区域#33的空闲空间的垃圾回收中也进行利用。在区域#33的垃圾回收中,SSD3从被分配给该区域#33的非易失性存储器内的各个区块中选择垃圾回收对象的一个以上的区块,将该选择出的各区块内的有效数据复制到被分配给该区域#33的另外的区块(空闲区块)。并且,SSD3更新查找表LUT#33,以使得向与已被复制的数据对应的LBA分别映射表示复制目的地物理存储位置的物理地址。

[0122] 此外,向区域#1(或区域#2)写入的数据并不限于虚拟机图像,也可以向区域#1(或区域#2)写入从其他区域参照的任意数据。例如,也可以向区域#1(或区域#2)写入共用文件这样的共用数据。

[0123] 图5表示向区域#1写入操作系统(OS#1)的二值图像的工作和将区域#1设定为禁止写入模式(只读模式)的工作。

[0124] (1) 基于服务器100(主机计算机)的管理者的操作,服务器100将用于向与命名空间NS#1对应的区域#1写入原创数据(例如,虚拟机图像A)的几个写入请求送出到SSD3。这些写入请求各自包含命名空间NS#1的标识符(NSID1)、开始LBA、转送长度。开始LBA表示写入数据应被写入的最初的LBA。转送长度表示写入数据的数据长度。

[0125] SSD3将原创数据、例如虚拟机图像A(操作系统(OS#1)的二值图像、应用程序(APL#1~#3)的二值图像)写入到区域#1。该情况下,既可以将操作系统(OS#1)以及应用程序(APL#1~#3)的安装图像从某盘复制到区域#1,也可以执行将操作系统(OS#1)以及应用程序(APL#1~#3)安装于区域#1的工作。

[0126] (2) SSD3更新与NSID1对应的LUT#1,以使得表示已写入原创数据(例如,虚拟机图像A)的区域#1内的多个物理存储位置的多个物理地址映射到与该虚拟机图像A对应的多个LBA。

[0127] (3) 在原创数据(例如,虚拟机图像A)的写入(安装)完成之后,SSD3也可以将区域#1设定为禁止向区域#1写入的只读模式(禁止写入模式)。该情况下,区域#1从允许读出和写入这双方的模式转换到只读模式(禁止写入模式)。由此,在原创数据(例如,虚拟机图像A)的写入(安装)完成之后,不执行向区域#1的新数据的写入/区域#1内的数据的改写。因此,已分配给区域#1的非易失性存储器内的各个区块的改写次数(编程/擦除循环的次数),维持为比作为虚拟存储器利用的其他区域少的值。

[0128] 图6表示控制区域#31的工作。

[0129] (4) 基于服务器100(主机计算机)的管理者的操作,服务器100将用于复制逻辑物理地址变换表的复制请求(复制命令)送出到SSD3。该复制请求包含指定复制源区域(复制源LUT)的复制源参数和指定复制目的地区域(复制目的地LUT)的复制目的地参数。在应该向与命名空间#31对应的区域#31创建参照原创数据(在此为虚拟机图像A)的虚拟机VM1用的虚拟存储器的情形下,服务器100将包含指定命名空间#1的标识符(NSID1)的复制源参数和指定命名空间#31的标识符(NSID31)的复制目的地参数的复制请求(复制命令)发送到SSD3。当接收到该复制命令时,SSD3将由复制源参数指定的LUT#1的内容(向区域#1的逻辑地址分别映射的物理地址的集合)复制到由复制目的地参数指定的LUT#31。当该复制完成

时,能够进行虚拟机VM1的利用。通常,各LUT的容量为对应的区域的容量的1/1000左右。因此,LUT的复制所需的时间为复制虚拟机图像A所需的时间的1/1000左右,LUT的复制可瞬间完成。因此,能够立即开始虚拟机VM1的利用。

[0130] (5) 服务器100 (主机计算机的虚拟机VM1) 为了从该虚拟机VM1用的虚拟储存器引导 (boot) 操作系统 (OS#1), 将各自指定区域#31的LBA的许多读取请求 (读取命令) 送出到SSD3。这些各读取请求包含命名空间NS#31的标识符 (NSID31)、开始LBA、转送长度。

[0131] (6) 当接收到一个读取请求时,SSD3参照与NSID31对应的LUT#31,从LUT#31取得与读取请求所包含的开始LBA对应的物理地址。

[0132] (7) 然后,SSD3使用该取得的物理地址,执行数据读出工作。因为该物理地址表示区域#1内的物理存储位置,所以原创数据的一部分及操作系统 (OS#1) 的一部分被从区域#1读出并返回给服务器100 (主机计算机的虚拟机VM1)。通过根据许多读取请求而执行的从区域#1的数据读出工作,可引导操作系统 (OS#1),该操作系统 (OS#1) 作为虚拟机VM1用的客户操作系统来执行。其结果是,虚拟机VM1成为能够执行各种各样的服务的运转状态。

[0133] 此外,在需要在该操作系统 (OS#1) 上执行应用程序的情况下,虚拟机VM1 (操作系统 (OS#1)) 为了执行应用程序,将各自指定区域#31的LBA的许多读取请求 (读取命令) 送出到SSD3。这些各读取请求也包含命名空间NS#31的标识符 (NSID31)、开始LBA、转送长度。当接收到一个读取请求时,SSD3参照与NSID31对应的LUT#31,从LUT#31取得与开始LBA对应的物理地址。并且,SSD3使用该取得的物理地址,执行数据读出工作。因为该物理地址表示区域#1内的物理存储位置,所以应用程序的一部分被从区域#1读出并返回给服务器100 (主机计算机的虚拟机VM1)。通过根据许多读取请求而执行的从区域#1的数据读出工作,可执行应用程序。

[0134] (8) 在虚拟机VM1处于运转状态的期间,服务器100 (虚拟机VM1) 为了进行用户数据的新建写入、该用户数据的改写、操作系统 (OS#1) 的升级等,将各自指定区域#31的LBA的许多写入请求 (写入命令) 送出到SSD3。

[0135] (9) 当接收到一个写入请求时,SSD3向区域#31写入从服务器100 (虚拟机VM1) 接收的数据 (写入数据)。

[0136] (10) 当数据 (写入数据) 被写入到区域#31时,SSD3更新LUT#31,以使得表示已写入该数据的区域#31内的物理存储位置的物理地址映射到该数据的LBA。由此,在接收到针对该数据的后续的读取请求的情况下,该数据被从区域#31读出,然后返回给服务器100 (虚拟机VM1)。另外,该数据的更新数据也被写入到区域#31。

[0137] 图7表示控制区域#32的工作。

[0138] (4) 基于服务器100的管理者的操作,服务器100将用于复制逻辑物理地址变换表的复制请求 (复制命令) 送出到SSD3。在应该向与命名空间#32对应的区域#32创建参照原创数据 (例如,虚拟机图像A) 的虚拟机VM2用的虚拟储存器的情形下,服务器100,该复制请求包含指定复制源区域 (复制源LUT) 的复制源参数和指定复制目的地区域 (复制目的地LUT) 的复制目的地参数。当接收到该复制命令时,SSD3将LUT#1的内容复制到LUT#32。当该复制完成时,能够进行虚拟机VM2的利用。

[0139] (5) 服务器100 (虚拟机VM2) 为了从该虚拟机VM2用的虚拟储存器引导操作系统 (OS#1),将各自指定区域#32的LBA的许多读取请求 (读取命令) 送出到SSD3。这些各读取请

求包含命名空间NS#32的标识符(NSID32)、开始LBA、转送长度。

[0140] (6) 当接收到一个读取请求时,SSD3参照与NSID32对应的LUT#32,从LUT#32取得与开始LBA对应的物理地址。

[0141] (7) 然后,SSD3使用该取得的物理地址,执行数据读出工作。因为该物理地址表示区域#1内的物理存储位置,所以操作系统(OS#1)的一部分被从区域#1读出。通过根据许多读取请求而执行的从区域#1的数据读出工作,可引导操作系统(OS#1),该操作系统(OS#1)作为虚拟机VM2用的客户操作系统来执行。其结果是,虚拟机VM2成为能够执行各种各样的服务的运转状态。

[0142] 此外,在需要在该操作系统(OS#1)上执行应用程序的情况下,虚拟机VM2(操作系统(OS#1))为了执行应用程序,将各自指定区域#32的LBA的许多读取请求(读取命令)送出到SSD3。这些各读取请求也包含命名空间NS#32的标识符(NSID32)、开始LBA、转送长度。当接收到一个读取请求时,SSD3参照与NSID32对应的LUT#32,从LUT#32取得与开始LBA对应的物理地址。并且,SSD3使用该取得的物理地址,执行数据读出工作。因为该物理地址表示区域#1内的物理存储位置,所以应用程序的一部分被从区域#1读出。通过根据许多读取请求而执行的从区域#1的数据读出工作,可执行应用程序。

[0143] (8) 在虚拟机VM2处于运转状态的期间,服务器100(虚拟机VM2)为了进行用户数据的新建写入、该用户数据的改写、操作系统(OS#1)的升级等,将各自指定区域#32的LBA的许多写入请求(写入命令)送出到SSD3。

[0144] (9) 当接收到一个写入请求时,SSD3向区域#31写入从服务器100(虚拟机VM2)接收的数据(写入数据)。

[0145] (10) 当数据(写入数据)被写入到区域#32时,SSD3更新LUT#32,以使得表示已写入该数据的区域#32内的物理存储位置的物理地址映射到该数据的LBA。由此,在接收到针对该数据的后续的读取请求的情况下,该数据会被从区域#32读出。另外,该数据的更新数据也会被写入到区域#32。

[0146] 图8表示由SSD3执行的对区域#1的刷新工作的例子。

[0147] 如上所述,对于区域#1的原创数据,会从区域#31、区域#32参照。因此,从区域#1的数据读出会以非常高的频度执行。也就是说,被分配给区域#1的各区块的改写次数非常少,但这些各区块的数据读出的次数非常多。数据读出的次数的增加会成为使读取干扰错误增加的要因。读取干扰错误是因数据读出工作而导致存储器单元(特别是,非选择存储器单元)的阈值电压等级发生变动(上升)的现象。因该读取干扰错误,从被分配给区域#1的各区块读取的数据的误比特率有可能会增加。

[0148] 因此,SSD3着眼于被分配给区域#1的各区块的改写次数少这一情况,执行以下的刷新工作。

[0149] SSD3从分配给区域#1的区块群中检测满足刷新条件的区块(步骤S11)。所谓满足刷新条件,既可以是误比特率为阈值X以上的区块,或者也可以是总读出次数为阈值Y以上的区块。SSD3对所检测出的区块内的数据的错误进行纠正(步骤S12)。在步骤S12中,SSD3使用对所检测出的区块内的数据付加的纠错码(ECC)来纠正该数据的错误。然后,SSD3将错误已被纠正的数据回写到与所检测出的区块相同的区块(步骤S13)。例如,如果分配给区域#1的区块BLK2的误比特率为阈值X以上,则可以将该区块BLK2检测为满足刷新条件的区块、即

应实施刷新工作的区块。并且,对该区块BLK2内的全部数据部的错误进行纠正。在执行了区块BLK2的擦除工作之后,将错误已被纠正的这些数据部回写到该区块BLK2。

[0150] 在通常的刷新工作中,从刷新对象的区块BLK2读出的数据,被移动到与该区块BLK2不同的区块(移动目的地区块)。然后,更新LUT#1,以使得移动目的地区块的各个物理地址映射到该移动的数据的各个逻辑地址。

[0151] 但是,在本实施方式中,LUT#1的内容(物理地址的集合)已被复制到LUT#31内,进一步也被复制到LUT#32内。因此,如果通过刷新工作将数据从区块BLK2移动到了另外的区块,则不仅需要更新LUT#1,还需要更新LUT#31、LUT#32。因此,需要更新的逻辑物理地址变换信息的量会非常大。这会成为使SSD3的性能降低的要因。

[0152] 在本实施方式中,因为错误已被纠正的数据部分别被回写到区块BLK2,所以不需要更新LUT#1、LUT#31、LUT#32。另外,被分配给区域#1的各区块的改写次数非常少。因此,即使错误已被纠正的这些数据部被回写到区块BLK2,区块BLK2的改写次数(编程/擦除循环的次数)也不会接近限制值。由此,将错误已被纠正的数据回写到原来的区块BLK2这一改良后的刷新工作,能够降低区域#1(命名空间NS#1)的读取干扰错误,能够高效率地改善已分配给区域#1(命名空间NS#1)的各区块的可靠性。

[0153] 图9表示与区域#1对应的LUT#1的内容的例子和与区域#31对应的LUT#31的内容的例子。

[0154] 在区域#1(命名空间NS#1),分配有非易失性存储器内的区块BLK1、BLK2、BLK3、……BLK50,在区域#31(命名空间NS#31),分配有非易失性存储器内的区块BLK51、BLK52、BLK53、……BLK100。分配给区域#1的BLK1、BLK2、BLK3、……BLK50分别仅被用于虚拟机图像A的保存,不被用于其他区域(其他命名空间)用的数据的保存。由此,能够将这些BLK1、BLK2、BLK3、……BLK50各自的改写次数维持为低的值。

[0155] LUT#1包含与命名空间#1所对应的LBA0、LBA1、LBA2、LBA3、LBA4、LBA5、LBA6、LBA7、LBA8、LBA9、LBA10、LBA11、LBA12、LBA13、LBA14、LBA15、……分别对应的多个入口。与在数据写入中利用的各LBA对应的入口,保持表示该数据被实际写入的非易失性存储器内的物理存储位置的物理地址。与在数据写入中尚未利用的各LBA对应的入口,不保持物理地址。该入口也可以保持表示该入口是未保持物理地址的空入口的空值。

[0156] 同样地,LUT#31也包含与命名空间#31所对应的LBA0、LBA1、LBA2、LBA3、LBA4、LBA5、LBA6、LBA7、LBA8、LBA9、LBA10、LBA11、LBA12、LBA13、LBA14、LBA15、……分别对应的多个入口。

[0157] 在图9中,例示了如下情况:在LUT#1中,在与命名空间#1对应的LBA0、LBA1、LBA2、LBA3、LBA4、LBA5、LBA6、LBA7、LBA8、LBA9、LBA10、LBA11、LBA12、LBA13、LBA14、LBA15、……,映射有区域#1的物理地址“100”、“101”、“102”、“103”、“104”、“105”、“106”、“107”、“200”、“201”、“202”、“203”、“204”、“205”、“206”、“207”、……。

[0158] 当LUT#1的内容被复制到LUT#31时,在LUT#31中,在与命名空间#31所对应的LBA0、LBA1、LBA2、LBA3、LBA4、LBA5、LBA6、LBA7、LBA8、LBA9、LBA10、LBA11、LBA12、LBA13、LBA14、LBA15、……分别对应的入口,保存了区域#1的物理地址“100”、“101”、“102”、“103”、“104”、“105”、“106”、“107”、“200”、“201”、“202”、“203”、“204”、“205”、“206”、“207”、……。其结果是,在与命名空间#31对应的LBA0、LBA1、LBA2、LBA3、LBA4、LBA5、LBA6、LBA7、LBA8、LBA9、

LBA10、LBA11、LBA12、LBA13、LBA14、LBA15、……，映射了区域#1的物理地址“100”、“101”、“102”、“103”、“104”、“105”、“106”、“107”、“200”、“201”、“202”、“203”、“204”、“205”、“206”、“207”、……。

[0159] 因此，查找表LUT#31内的各个物理地址不是区域#31内的物理存储位置，而是指区域#1内的物理存储位置。因此，向区域#31的各逻辑地址（命名空间#31的各逻辑地址）的读取访问，被改朝向区域#1。

[0160] 当SSD3从服务器100接收到指定区域#31的LBA_x的写入请求（向一次也未写入数据的LBA_x的新建的写入请求、或者已写入到LBA_x中的数据的改写请求）时，SSD3向区域#31写入应写入的数据（写入数据）。例如，如果已写入该写入数据的区域#31内的物理存储位置是由物理地址“500”指定的物理存储位置，则LUT#31被更新为使得表示区域#31内的物理存储位置的物理地址“500”映射到LBA_x。

[0161] 该情况下，在LUT#31中，如果与LBA_x对应的入口内的值为空值、即所接收到的写入请求是向LBA_x的新建的写入请求，则物理地址“500”被重新保存于与LBA_x对应的入口。其结果是，与LBA_x对应的入口的内容被从空值变更为物理地址“500”。

[0162] 或者，如果在与LBA_x对应的入口已经保存有表示区域#1内的某物理存储位置的物理地址、即所接收到的写入请求是已写入到LBA_x中的旧数据的改写请求，则在LUT#31中，与LBA_x对应的入口的内容被从表示保存有LBA_x的旧数据的物理存储位置的物理地址变更为物理地址“500”。

[0163] 在本实施方式中，说明了使用多个命名空间的环境，但本实施方式并不限于仅是使用多个命名空间的环境，例如，在与区域#1的LBA空间不同的LBA空间被分配给区域#31的环境下也可以适用。例如，也可以是，区域#1的LBA空间是LBA0、LBA1、LBA2、LBA3、LBA4、LBA5、LBA6、LBA7、LBA8、LBA9、LBA10、LBA11、LBA12、LBA13、LBA14、LBA15、……LBA_{n-1}，区域#31的LBA空间是LBA_n、LBA_{n+1}、LBA_{n+2}、LBA_{n+3}、LBA_{n+4}、LBA_{n+5}、LBA_{n+6}、LBA_{n+7}、LBA_{n+8}、LBA_{n+9}、LBA_{n+10}、LBA_{n+11}、LBA_{n+12}、LBA_{n+13}、LBA_{n+14}、LBA_{n+15}、……LBA_{2n-1}。

[0164] 该情况下，当LUT#1的内容被复制到LUT#31时，在与命名空间#31对应的LBA_n、LBA_{n+1}、LBA_{n+2}、LBA_{n+3}、LBA_{n+4}、LBA_{n+5}、LBA_{n+6}、LBA_{n+7}、LBA_{n+8}、LBA_{n+9}、LBA_{n+10}、LBA_{n+11}、LBA_{n+12}、LBA_{n+13}、LBA_{n+14}、LBA_{n+15}、……，映射了区域#1的物理地址“100”、“101”、“102”、“103”、“104”、“105”、“106”、“107”、“200”、“201”、“202”、“203”、“204”、“205”、“206”、“207”……。在该情况下，也能够将指定命名空间#31的某LBA的读取请求改朝向区域#1。

[0165] 此外，因为针对区域#1（命名空间#1）的数据的改写几乎不会执行，所以在区域#1（命名空间#1）也可以适用自动精简配置（Thin Provisioning）。该情况下，在区域#1（命名空间#1）中，可仅确保（预约）与比区域#1的尺寸（与LBA的个数对应的容量）少的容量相当的个数的区块群。同样地，在区域#2（命名空间#2）也可以适用自动精简配置。

[0166] 在区域#31（命名空间#31），可以适用自动精简配置、超额配置（Over Provisioning）的任一方。同样地，在区域#32（命名空间#32）、区域#33（命名空间#33）的各区域，可以适用自动精简配置、超额配置的任一方。

[0167] 图10表示根据指定区域#31（或区域#32）的某逻辑地址的读取请求而执行的读取工作和根据指定区域#31（或区域#32）的某逻辑地址的写入请求而执行的写入工作。

[0168] 在本实施方式中，LUT#1的内容被复制到LUT#31、LUT#32的各LUT。其结果是，朝向

区域#31的各LBA的读取访问,被改朝向区域#1。同样地,朝向区域#32的各LBA的读取访问,也被改朝向区域#1。

[0169] 在SSD3从服务器100接收到指定区域#31的某LBA(例如LBA_x)的写入请求时,SSD3将从服务器100接收的数据(DATA_x)写入到区域#31内的可利用的物理存储位置。并且,更新LUT#31,将表示已写入有DATA_x的物理存储位置的物理地址保存于与LUT#31内的LBA_x对应的入口。

[0170] 在SSD3从服务器100接收到指定区域#31的某LBA(例如LBA_y)的写入请求时,SSD3将从服务器100接收的数据(DATA_y)写入到区域#31内的可利用的物理存储位置。并且,更新LUT#31,将表示已写入有DATA_y的物理存储位置的物理地址保存于与LUT#31内的LBA_y对应的入口。

[0171] 在SSD3从服务器100接收到指定区域#32的某LBA(例如LBA_z)的写入请求时,SSD3将从服务器100接收的数据(DATA_z)写入到区域#32内的可利用的物理存储位置。并且,更新LUT#32,将表示已写入有DATA_z的物理存储位置的物理地址保存于与LUT#32内的LBA_z对应的入口。

[0172] 图11表示分配给区域#1的单层单元(SLC)区块群和分配给区域#31(或区域#32)的多层单元(MLC)/三层单元(TLC)区块群。

[0173] 在SLC区块中,每个存储器单元保存一位(1比特)。各存储器单元被设定为与2个状态(一个擦除状态、一个编程状态)对应的2个阈值电压分布的某一个。因此,在SLC区块中,用于对这些状态(擦除状态、编程状态)间进行分离的余裕的量比较大,因此不容易产生读取干扰错误。

[0174] 另一方面,在MLC区块中,每个存储器单元保存两位(2比特)。各存储器单元被设定为与4个状态(一个擦除状态、三个编程状态)对应的4个阈值电压分布的某一个。因此,在MLC区块中,这些4个状态以比较窄的阈值电压间隔排列,因此用于对相邻的2个状态间进行分离的余裕的量比SLC区块窄,与SLC区块相比容易产生读取干扰错误。

[0175] 在TLC区块中,每个存储器单元保存三位(3比特)。各存储器单元被设定为与8个状态(一个擦除状态、七个编程状态)对应的8个阈值电压分布的某一个。因此,在TLC区块中,用于对相邻的2个状态间进行分离的余裕的量进一步变窄,更容易产生读取干扰错误。

[0176] 在本实施方式中,如图11所示,在频繁地执行数据读出且几乎不会执行数据改写的区域#1(命名空间NS#1),分配有SLC区块群(SLC-BLK1、SLC-BLK2、SLC-BLK3、……SLC-BLK50)来作为物理资源。这是因为:对于区域#1,相比于容量成本,更重视针对读取干扰错误的抵抗性(抗干扰性)。

[0177] 另一方面,在区域#31(命名空间NS#31)中,数据读出次数比区域#1(命名空间NS#1)少,但随着时间经过,存在大量的数据被写入到区域#31的可能性。因此,在区域#31(命名空间NS#31),分配有MLC(或TLC)区块群(MLC/TLC-BLK51、MLC/TLC-BLK52、MLC/TLC-BLK53、……MLC/TLC-BLK100)来作为物理资源。这是因为:对于区域#31,相比于针对读取干扰错误的抵抗性,更重视容量成本。

[0178] 同样地,在区域#32(命名空间NS#32),也分配有MLC(或LC)区块群(MLC/TLC-BLK101、MLC/TLC-BLK102、MLC/TLC-BLK103、……MLC/TLC-BLK200)来作为物理资源。

[0179] 在SSD3内置有每个存储器单元能保存一位数据的SLC-NAND闪速存储器和每个存

存储器单元能保存多位数据的MLC/TLC-NAND闪速存储器的情况下,可以将SLC-NAND闪速存储器内的SLC区块群分配给区域#1,将MLC/TLC-NAND闪速存储器内的MLC/TLC区块群分配给区域#31以及区域#32。

[0180] 或者,在SSD3内置有每个存储器单元能保存多位数据的MLC/TLC-NAND闪速存储器的情况下,也可以将从MLC/TLC-NAND闪速存储器分配给区域#1的各MLC/TLC区块设定为SLC模式,将从MLC/TLC-NAND闪速存储器分配给区域#31(或区域#32)的各MLC/TLC区块设定为MLC/TLC模式。

[0181] 关于设定为SLC模式的MLC/TLC区块,SSD3在每个存储器单元被写入一位数据的第1写入模式下,将数据(在此为操作系统的二值图像等)写入到MLC/TLC区块。换言之,SSD3使用每个存储器单元被写入一位数据的第1写入模式,执行向区域#1的数据写入。例如,在第1写入模式下,可以向与同一字线连接的存储器单元群只写入快页(Lower page)数据。

[0182] 关于设定为MLC/TLC模式的MLC/TLC区块,SSD3在每个存储器单元被写入多位数据的第2写入模式下,将数据写入到MLC/TLC区块。换言之,SSD3使用每个存储器单元被写入多位数据的第2写入模式,执行向区域#31或区域#32的数据写入。例如,在第2写入模式下,可以向与同一字线连接的存储器单元群写入快页数据和慢页(Upper page)数据(MLC模式),也可以写入快页数据、中间页(Middle page)数据和慢页数据(TLC模式)。

[0183] 图12表示针对区域#1的刷新工作的另一例。

[0184] 图12的刷新工作可适用于图13所示这样的执行2个阶段的地址变换的构成。

[0185] 首先,对2个阶段的地址变换进行说明。

[0186] 如图13所示,在LUT#1、……LUT#31、LUT#32各自的后段,设有区块地址变换表T1。区块地址变换表T1管理变换前的各个区块地址与变换后的各个区块地址之间的映射。该区块地址变换表T1例如也可以包含与变换前的多个区块地址对应的多个入口。在各入口保存有与对应的变换前的区块地址对应的变换后的区块地址。

[0187] 从LUT#1、LUT#31或LUT#32取得的物理地址包含区块地址BLK-addr和页地址Page-addr。根据要取得的物理地址内的区块地址BLK-addr,选择区块地址变换表T1内的多个入口之一。并且,从区块地址变换表T1读出所选择出的入口所保持的区块地址BLK-addr',通过该区块地址BLK-addr'和所取得的物理地址内的页地址Page-addr,生成新的物理地址(区块地址BLK-addr',页地址Page-addr)。

[0188] 通过这样的2个阶段的地址变换,即使例如分配给区域#1的区块内的数据被移动到另外的区块(移动目的地区块),也不用更新LUT#1、LUT#31、LUT#32,而仅是简单地通过更新区块地址变换表T1,就能够获得与移动目的地区块对应的物理地址。

[0189] 在图12的刷新工作中,可执行以下的处理。

[0190] SSD3从分配给区域#1的区块群中检测满足刷新条件的区块(步骤S21)。所谓满足刷新条件的区块,既可以是误比特率为阈值X以上的区块,或者也可以是总读出次数为阈值Y以上的区块。SSD3对所检测出的区块内的数据的错误进行纠正(步骤S22)。在步骤S22中,SSD3使用对所检测出的区块内的数据附加的纠错码(ECC)来纠正该数据的错误。然后,SSD3将错误已被纠正的数据回写到与所检测出的区块不同的另外的区块(移动目的地区块)(步骤S23)。移动目的地区块也可以是读出次数少的空闲区块。并且,SSD3更新区块地址变换表T1,以使得向所检测出的区块的访问被改朝向移动目的地区块(步骤S24)。在步骤S24中,

SSD3将移动目的地区块的区块地址保存于区块地址变换表T1,以使得从LUT#1、LUT#31或LUT#32取得的物理地址内所包含的所检测出的区块的区块地址被变换为移动目的地区块的区块地址。该情况下,移动目的地区块的区块地址被保存于与所检测出的区块的区块地址对应的入口。

[0191] 图14表示包括SSD3和服务器(主机计算机)100的信息处理系统1。

[0192] 服务器(主机计算机)100也可以经由网络50与多个终端51连接。这些终端51与用图1、图2说明的用户A、用户B、用户C对应。服务器100向这些终端51提供IaaS这样的服务。

[0193] SSD3可以作为服务器100的物理储存器来使用。SSD3既可以内置于作为服务器100发挥功能的计算机,也可以经由电缆或网络连接于该计算机。

[0194] 作为用于使服务器100和SSD3相互连接的接口,可以使用SCSI、Serial Attached SCSI (SAS)、ATA、Serial ATA (SATA)、PCI Express (PCIe)、Ethernet (注册商标)、Fibre channel、NVM Express (NVMe) (注册商标)等。

[0195] SSD3具备控制器(存储器控制器)4以及非易失性存储器(NAND型闪速存储器)5。SSD3也可以具备随机存取存储器、例如DRAM6。

[0196] NAND型闪速存储器5包括包含呈矩阵状配置的多个存储器单元的存储器单元阵列。NAND型闪速存储器5既可以是二维结构的NAND型闪速存储器,也可以是三维结构的NAND型闪速存储器。

[0197] NAND型闪速存储器5的存储器单元阵列包括多个区块BLK0~BLKm-1。区块BLK0~BLKm-1的各区块由许多页(在此为页P0~Pn-1)组成。区块BLK0~BLKm-1作为擦除单位发挥功能。区块有时也称为“擦除区块”或“物理区块”。页P0~Pn-1的各页包含与同一字线连接的多个存储器单元。页P0~Pn-1是数据写入工作以及数据读取工作的单位。

[0198] 控制器4经由Toggle、开放式NAND接口(ONFI)这样的NAND接口13,与作为非易失性存储器的NAND型闪速存储器5电连接。NAND接口13作为构成为控制NAND型闪速存储器5的NAND控制电路发挥功能。

[0199] NAND型闪速存储器5如图15所示,也可以包含多个NAND型闪速存储器裸片。各个NAND型闪速存储器裸片能够独立地工作。因此,NAND型闪速存储器裸片作为能够并行工作的最小单位发挥功能。在图15中,例示了如下情况:在NAND接口13连接有8个通道Ch.1~Ch.8,在8个通道Ch.1~Ch.8的各通道连接有4个NAND型闪速存储器裸片。该情况下,能够使最多32个NAND型闪速存储器裸片#1、#2、……#32并行工作。

[0200] SSD3内的控制器4能够作为构成为执行NAND型闪速存储器5的数据管理和NAND型闪速存储器5的区块管理的闪存转换层(FTL)发挥功能。

[0201] 在由该FTL执行的数据管理中,包括:(1)表示各个逻辑地址与NAND型闪速存储器5的各个物理地址之间的对应关系的映射信息的管理;(2)用于对页单位的读/写和区块单位的擦除工作进行隐藏的处理等。

[0202] 对于各个逻辑区块地址(LBA)与各个物理地址之间的映射的管理,使用作为地址变换表(逻辑物理地址变换表)发挥功能的查找表(LUT)来执行。

[0203] 在本实施方式中,控制器4如上所述,使用与多个区域分别对应的多个查找表LUT#1、LUT#2、……LUT#31、LUT#32、LUT#33,按每个区域(命名空间),管理各个LBA与各个物理地址之间的映射。

[0204] 与某LBA对应的物理地址,表示已写入该LBA的数据的NAND型闪速存储器5内的物理存储位置。LUT#1、LUT#2、……LUT#31、LUT#32、LUT#33也可以在SSD3的电源接通时从NAND型闪速存储器5装载到DRAM6。一般而言,各LUT的尺寸比较大。因此,也可以将各LUT的至少一部分作为逻辑物理地址变换表缓存而保存于DRAM6。

[0205] 如果包含与由来自服务器100的读取请求(读取命令)指定的逻辑地址对应的目标物理地址的缓存行存在于逻辑物理地址变换表缓存(缓存命中),则立即从由该目标物理地址指定的NAND型闪速存储器5内的物理存储位置读取数据。

[0206] 另一方面,如果包含目标物理地址的缓存行不存在于逻辑物理地址变换表缓存(缓存未命中),则从NAND型闪速存储器5读出包含目标物理地址的查找表内的一部分(地址变换表部)。并且,从逻辑物理地址变换表缓存中赶出逻辑物理地址变换表缓存内的替换对象缓存行的内容,取而代之,将从NAND型闪速存储器5读出的地址变换表部保存于该缓存行。并且,从由该目标物理地址指定的NAND型闪速存储器5内的物理存储位置读取数据。

[0207] 在NAND型闪速存储器5的各区域中,对于向页的数据写入,每一个擦除循环只能进行一次。因此,控制器4不是将与某LBA对应的更新数据写入到保存有与该LBA对应的以前数据的物理存储位置而是写入到另外的物理存储位置。并且,控制器4更新与该区域对应的查找表LUT,对该LBA关联该另外的物理存储位置的物理地址,并且使以前的数据无效化。

[0208] 在区块管理中,包括不良区块的管理、耗损均衡(wear leveling)以及垃圾回收等。

[0209] 耗损均衡是用于使物理区块各自的消耗均匀化的工作。

[0210] 在垃圾回收中,控制器4为了使能够写入数据的空闲区块的个数增加,将有效数据和无效数据并存的几个对象区块内的有效数据复制到另外的区块(例如空闲区块)。在此,所谓有效数据,是可从LUT参照的数据(即从逻辑地址作为最新的数据而关联的数据),意味着后面有可能会从服务器(主机计算机)100读取的数据。所谓无效数据,意味着已经不可能从服务器(主机计算机)100读取的数据。例如,与某逻辑地址关联的数据是有效数据,与任何逻辑地址都没有关联的数据是无效数据。并且,控制器4将已被复制的有效数据的逻辑地址分别映射到复制目的地的物理地址。通过将有效数据复制到另外的区块而仅成为了无效数据的区块,被作为空闲区块释放。由此,该区块能够在执行了其擦除工作之后进行再利用。

[0211] 接着,对控制器4的构成进行说明。

[0212] 控制器4例如也可以通过一个LSI来实现。控制器4至少包括硬件逻辑。该硬件逻辑包括主机接口11、CPU12、NAND接口13、DRAM接口14以及ECC电路15等。这些CPU12、NAND接口13、DRAM接口14、ECC电路15经由总线10相互连接。

[0213] 该主机接口11是构成为执行与服务器(主机计算机)100通信的主机接口电路。主机接口11从服务器(主机计算机)100接收各种各样的命令(写入命令、读取命令、命名空间创建命令、复制命令等)。

[0214] 写入命令对SSD3请求写入由该写入命令指定的数据。写入命令也可以包含开始LBA、转送长度以及命名空间ID。写入命令内的命名空间ID是用于唯一识别应写入数据的区域(命名空间)的标识符。读取命令对SSD3请求读取由该读取命令指定的数据。读取命令也可以包含开始LBA、转送长度以及命名空间ID。读取命令内的命名空间ID是用于唯一识别应

读出数据的区域(命名空间)的标识符。

[0215] 命名空间创建命令是对SSD3请求创建命名空间的命令。复制命令对SSD3请求将某命名空间的内容复制到另外的命名空间。更详细而言,复制命令对SSD3请求将某命名空间的LUT的内容复制到另外的命名空间的LUT。

[0216] CPU12是构成为控制主机接口11、NAND接口13、DRAM接口14、ECC电路15的处理器。CPU12除了上述的FTL的处理之外,还执行用于处理来自服务器(主机计算机)100的各种各样的命令的命令处理等。

[0217] 这些FTL处理以及命令处理也可以通过由CPU12执行的固件来控制。该固件使CPU12作为命名空间管理部21、LUT复制控制部22、禁止写入控制部23、读/写控制部24以及刷新控制部25发挥功能。此外,命名空间管理部21、LUT复制控制部22、禁止写入控制部23、读/写控制部24以及刷新控制部25也可以通过硬件来实现。

[0218] 命名空间管理部21具有用于管理多个命名空间的多命名空间管理功能。各命名空间相当于作为非易失性存储器的NAND型闪速存储器5内的一种区域。命名空间管理部21基于来自服务器(主机计算机)100的各个命名空间创建命令,创建多个命名空间即多个区域。换言之,命名空间管理部21基于来自服务器(主机计算机)100的区域各自的命名空间创建命令,将NAND型闪速存储器5在逻辑上分割成多个区域(命名空间)。

[0219] LUT复制控制部22基于来自服务器(主机计算机)100的LUT复制命令,将与复制源命名空间对应的LUT的内容复制到与复制目的地命名空间对应的LUT。该LUT复制命令是用于将复制源命名空间所保存的数据复制到复制目的地的复制命令。

[0220] 禁止写入控制部23在向区域#1(命名空间#NS1)写入虚拟机图像A的写入完成之后,将区域#1(命名空间#NS1)设定为禁止向区域#1(命名空间#NS1)写入的只读模式(禁止写入模式)。同样地,禁止写入控制部23在向区域#2(命名空间#NS2)写入虚拟机图像B的写入完成之后,将区域#2(命名空间#NS2)设定为禁止向区域#2(命名空间#NS2)写入的只读模式(禁止写入模式)。

[0221] 在从服务器(主机计算机)100接收到针对区域#31(命名空间#NS31)的读取命令的情况下,读/写控制部24通过参照与命名空间#NS31对应的LUT#31,从LUT#31取得与读取对象的逻辑地址对应的物理地址。并且,读/写控制部24从由该取得的物理地址指定的NAND型闪速存储器5的物理存储位置读出数据,将该读出的数据返回给服务器(主机计算机)100。区域#31(命名空间#NS31)作为虚拟机VM1的虚拟储存器(虚拟盘)发挥功能。虚拟机VM1仅通过经由设备驱动器等发布包含命名空间#NS31的NSID31的读取请求(读取命令),就能够将区域#31作为自身的虚拟储存器进行读取访问。

[0222] 在从服务器(主机计算机)100接收到针对区域#31(命名空间#NS31)的写入命令的情况下,读/写控制部24向区域#31(命名空间#NS31)写入应写入的数据,并且,更新LUT#31,将表示已写入该数据的区域#31内的物理存储位置的物理地址映射到该数据的逻辑地址。虚拟机VM1仅通过经由设备驱动器等发布包含命名空间#NS31的NSID31的写入请求(写入命令),就能够将区域#31作为自身的虚拟储存器进行写入访问。

[0223] 在从服务器(主机计算机)100接收到针对区域#32(命名空间#NS32)的读取命令的情况下,读/写控制部24通过参照与命名空间#NS32对应的LUT#32,从LUT#32取得与读取对象的逻辑地址对应的物理地址。并且,读/写控制部24从由该取得的物理地址指定的NAND型

闪速存储器5的物理存储位置读出数据,将该读出的数据返回给服务器(主机计算机)100。区域#32(命名空间#NS32)作为虚拟机VM2的虚拟储存器(虚拟盘)发挥功能。虚拟机VM2仅通过经由设备驱动器等发布包含命名空间#NS32的NSID32的读取请求(读取命令),就能够将区域#32作为自身的虚拟储存器进行读取访问。

[0224] 在从服务器(主机计算机)100接收到针对区域#32(命名空间#NS32)的写入命令的情况下,读/写控制部24向区域#32(命名空间#NS32)写入应写入的数据,并且,更新LUT#32,将表示已写入该数据的区域#32内的物理存储位置的物理地址映射到该数据的逻辑地址。虚拟机VM2仅通过经由设备驱动器等发布包含命名空间#NS32的NSID32的写入请求(写入命令),就能够将区域#32作为自身的虚拟储存器进行写入访问。

[0225] 刷新控制部25从分配给区域#31(命名空间#NS31)的区块群中检测满足刷新条件的区块。并且,刷新控制部25对所检测出的区块内的数据的错误进行纠正,将错误已被纠正的数据回写到与所检测出的区块相同的区块。

[0226] 此外,如用图12、图13进行说明的那样,刷新控制部25也能够将错误已被纠正的数据写入到与所检测出的区块不同的另外的区块(移动目的地区块)。该情况下,刷新控制部25将移动目的地区块的区块地址保存于区块地址变换表T1,以使得指定所检测出的区块的区块地址被变换为移动目的地区块的区块地址。

[0227] NAND接口13在CPU12的控制下,控制NAND型闪速存储器5。DRAM接口14是构成为在CPU12的控制下控制DRAM6的DRAM控制器。DRAM6的存储区域的一部分被用于保存对来自服务器(主机计算机)100的写入数据进行暂时保存的写入缓冲器(WB)31。另外,DRAM6的存储区域的其他的一部分被用于保存LUT#1、LUT#2、……LUT#31、LUT#32、LUT#33。

[0228] ECC电路15包括ECC编码器和ECC译码器。ECC编码器对应写入的数据(用户数据)进行编码,生成包含用户数据和纠错码(ECC)的存储器写入数据。ECC译码器对包含用户数据和纠错码(ECC)的存储器读出数据进行译码,纠正用户数据的错误。

[0229] 图16表示向区域#1写入虚拟机图像的处理的步骤。

[0230] 服务器(主机计算机)100将用于向区域#1(命名空间#NS1)写入虚拟机图像A的写入请求送出到SSD3。SSD3的控制器4将虚拟机图像A写入到区域#1(命名空间#NS1)(步骤S101)。然后,控制器4更新LUT#1,向与虚拟机图像A对应的多个LBA,分别映射表示已写入虚拟机图像A的区域#1内的物理存储位置的物理地址(步骤S102)。

[0231] 在虚拟机图像A全部被写入到区域#1之后,基于服务器100(主机计算机)的管理者的操作,服务器(主机计算机)100也可以向SSD3通知虚拟机图像A的写入完成。对该通知进行响应,控制器4将区域#1(命名空间#NS1)设定为禁止写入模式(禁止写入且只允许读取的只读模式)(步骤S103)。此外,在步骤S103中,也可以不仅将区域#1(命名空间#NS1)设定为只读模式还禁止对区域#1(命名空间#NS1)执行垃圾回收。

[0232] 图17表示创建虚拟储存器(区域#31或区域#32)的处理的步骤。

[0233] 在需要创建新虚拟储存器时,服务器(主机计算机)100为了创建虚拟储存器而将命名空间创建命令送出到SSD3。SSD3的控制器4创建命名空间,将几个数的区块预约为该命名空间用(步骤S111)。然后,控制器4向所创建出的命名空间分配ID,将该命名空间的ID返回给服务器(主机计算机)100(步骤S112)。

[0234] 服务器(主机计算机)100将复制命令(地址变换表复制命令)送出到SSD3。SSD3的

控制器4将与由复制命令内的复制源参数指定的NSID对应的LUT的内容复制到与由复制命令内的复制目的地参数指定的NSID对应的LUT(步骤S113)。

[0235] 图18的流程图表示读/写处理的步骤。

[0236] 当从服务器(主机计算机)100接收到读取请求(读取命令)时(步骤S121:是),SSD3的控制器4参照与由读取命令内的NSID指定的命名空间对应的LUT,从该LUT取得与读取命令内的逻辑地址(LBA)对应的物理地址(步骤S122)。换言之,在接收到指定某区域的逻辑地址的读取命令时,控制器4参照与该区域对应的LUT,取得与读取命令内的逻辑地址对应的物理地址。

[0237] 控制器4从与该取得的物理地址对应的NAND型闪速存储器5内的物理存储位置读出数据,将该读出的数据返回给服务器(主机计算机)100(步骤S123)。在本实施方式中,例如,在与虚拟存储器用的区域#31对应的LUT#31复制有区域#1的LUT#1的内容。因此,在接收到指定区域#31的某逻辑地址的读取命令的情况下,参照LUT#31,并且,从LUT#31取得表示区域#1内的物理存储位置的物理地址。其结果是,从LUT#1的物理存储位置读出数据。

[0238] 当从服务器(主机计算机)100接收到写入请求(写入命令)时(步骤S124:是),SSD3的控制器4判定由写入命令内的NSID指定的写入目的地命名空间是否为禁止写入模式(只读模式)(步骤S125)。换言之,在接收到指定某区域的逻辑地址的写入命令时,控制器4判定该区域是否为禁止写入模式(只读模式)。

[0239] 如果写入目的地命名空间被设定为禁止写入模式(只读模式)(步骤S124:是),则控制器4将表示向写入目的地命名空间的写入已被禁止这一情况的错误通知返回给服务器(主机计算机)100(步骤S128)。

[0240] 如果写入目的地命名空间不是禁止写入模式(只读模式)(步骤S124:否),则控制器4向与写入目的地命名空间对应的区域写入数据(写入数据)(步骤S126)。然后,控制器4更新与写入目的地命名空间对应的LUT,以使得表示已写入了写入数据的物理存储位置的物理地址映射到该写入数据的逻辑地址(LBA)(步骤S127)。

[0241] 图19的流程图表示刷新处理的步骤。

[0242] SSD3的控制器4从与保存有虚拟机图像的命名空间对应的区块群、即保存有虚拟机图像的区块群中,检测满足刷新条件的区块(步骤S131)。控制器4从该检测出的区块中以页为单位来读出数据,使用ECC电路15来纠正所读出的数据的错误(步骤S132)。在步骤S132中,可纠正所检测出的区块的全部页的数据的错误。控制器4对所检测出的区块实施擦除工作,将检测出的区块的全部存储器单元设为擦除状态(步骤S133)。然后,控制器4将错误已被纠正的全部页的数据回写到与所检测出的区块相同的区块(步骤S134)。

[0243] 图20的流程图表示刷新处理的另一步骤。

[0244] SSD3的控制器4从与保存有虚拟机图像的命名空间对应的区块群、即保存有虚拟机图像的区块群中,检测满足刷新条件的区块(步骤S141)。控制器4从该检测出的区块中以页为单位来读出数据,使用ECC电路15来纠正所读出的数据的错误(步骤S142)。在步骤S142中,可纠正所检测出的区块的全部页的数据的错误。

[0245] 控制器4判定所检测出的区块的误比特率或读取次数是否为数据移动用的阈值以上(步骤S143)。该阈值是用于判定是否应该将错误已被纠正的数据向与所检测出的区块不同的另外的区块移动的阈值。

[0246] 如果所检测出的区块的误比特率或读取次数小于阈值(步骤S143:否),则控制器4对所检测出的区块实施擦除工作而将检测出的区块的全部存储器单元设为擦除状态(步骤S144)。然后,控制器4将错误已被纠正的全部页的数据回写到与所检测出的区块相同的区块(步骤S145)。

[0247] 如果所检测出的区块的误比特率或读取次数为阈值以上(步骤S143:是),则控制器4将错误已被纠正的全部页的数据写入到与所检测出的区块不同的另外的区块(移动目的地区块)(步骤S146)。然后,控制器4更新区块地址变换表T1,以使得所检测出的区块的区块地址被变换为移动目的地区块的区块地址,将移动目的地区块的区块地址保存于区块地址变换表T1(步骤S147)。在步骤S147中,移动目的地区块的区块地址被保存于与所检测出的区块的区块地址对应的入口。

[0248] 此外,在此,说明了仅在所检测出的区块的误比特率或读取次数为阈值以上的情况下,才将错误已被纠正的数据写入到与所检测出的区块不同的另外的区块(移动目的地区块),并更新区块地址变换表T1的情况,但也可以与所检测出的区块的误比特率或读取次数无关系地,将错误已被纠正的数据写入到与所检测出的区块不同的另外的区块(移动目的地区块),并更新区块地址变换表T1。另外,可能会存在如下情况:对所检测出的区块实施擦除工作的结果,导致该区块成为不良区块(不良区块)。在该情况下,也可以将错误已被纠正的数据写入到与所检测出的区块不同的另外的区块(移动目的地区块),并更新区块地址变换表T1。

[0249] 图21的流程图表示以每个存储器单元被写入一位数据的第1写入模式向区域#1(保存原创的虚拟机图像的原创命名空间)写入数据,以每个存储器单元被写入多位数据的第2写入模式向区域#31(或区域#32)写入数据的处理的步骤。

[0250] SSD3的控制器4判定写入目的地的命名空间是否为应保存原创的虚拟机图像的原创命名空间(区域#1或区域#2)(步骤S151)。

[0251] 如果写入目的地的命名空间是原创命名空间(区域#1或区域#2),则控制器4以每个存储器单元被写入一位数据的写入模式(SLC模式),将数据(在此为操作系统的二值图像等)写入到原创命名空间(区域#1或区域#2)(步骤S152)。换言之,控制器4使用每个存储器单元被写入一位数据的写入模式(SLC模式),执行向区域#1或区域#2的数据写入。

[0252] 另一方面,如果写入目的地的命名空间是作为虚拟机用的虚拟存储器利用的命名空间(区域#31、区域#32或区域#33),则控制器4以每个存储器单元被写入多位数据的写入模式(MLC模式或TLC模式),将数据写入到作为虚拟存储器利用的命名空间(区域#31、区域#32或区域#33)(步骤S153)。换言之,控制器4使用每个存储器单元被写入多位数据的写入模式(MLC模式或TLC模式),执行向区域#31或区域#32的数据写入。

[0253] 图22表示包括多个SSD3的服务器(主机计算机)100的构成例。

[0254] 作为该服务器(主机计算机)100发挥功能的信息处理装置,具备能够容纳于机箱的薄的箱形的框体301。许多SSD3可以配置在框体301内。该情况下,各SSD3可以以能够拆下的方式插入到设置在框体301的前面301A的卡槽中。

[0255] 系统板(母板)302配置在框体301内。在系统板(母板)302上,安装有包括物理CPU、物理存储器、物理NIC的各种各样的电子部件。

[0256] 如以上进行的说明,根据本实施方式,根据来自服务器(主机计算机)100的请求,

将与保存有原创数据(第1数据)的区域#1对应的LUT#1的内容复制到与区域#31对应的LUT#31。并且,每次接收到来自服务器(主机计算机)100的指定区域#31的逻辑地址的读取请求时,基于LUT#31,将第1数据的一部分从区域#1读出并返回给服务器(主机计算机)100。因此,不用将区域#1中保存的第1数据复制到区域#31,而能够经由区域#31的各个逻辑地址来读取访问区域#1中保存的第1数据。

[0257] 另外,在从服务器(主机计算机)100接收到指定区域#31的逻辑地址的写入请求的情况下,向区域#31写入应写入的第2数据,并且更新LUT#31,以使得表示已写入第2数据的区域#31内的物理存储位置的物理地址映射到由写入请求指定的逻辑地址。如此,因为LUT#31被更新,所以在第2数据被写入到区域#31之后从服务器(主机计算机)100接收到针对第2数据的读取请求的情况下,基于该更新后的LUT#31,能够将第2数据从区域#31读出并返回给服务器(主机计算机)100。如此,应写入的数据被写入到区域#31,已写入到区域#31的数据被从区域#31读出。因此,即使在区域#1的原创数据(第1数据)被多个区域(区域#31、#32,……)共用的情况下,也不会产生矛盾。另外,因为区域#1会专门用于读取,所以能够将区域#1的改写次数维持为低的值。

[0258] 另外,从分配给第1区域的区块群中检测满足刷新条件的区块,对所检测出的区块内的数据的错误进行纠正,将错误已被纠正的数据回写到与所检测出的区块相同的区块。因此,不用更新LUT#1、LUT#31,而能够执行第1区域的刷新工作。已分配给区域#1的各区块的改写次数非常少。因此,即使例如错误已被纠正的数据被回写到与所检测出的区块相同的区块,该区块的改写次数(编程/擦除循环的次数)也不会接近限制值。由此,将错误已被纠正的数据回写到与所检测出的区块相同的区块这一改良后的刷新工作,能够高效地改善已分配给区域#1的各区块的可靠性。

[0259] 此外,在本实施方式中,作为非易失性存储器而例示了NAND型闪速存储器。但是,本实施方式的功能例如也能够适用于MRAM(Magneto-resistive Random Access Memory:磁性随机存取存储器)、PRAM(Phase change Random Access Memory:相变随机存取存储器),ReRAM(Resistive Random Access Memory:可变电阻式随机存取存储器)或FeRAM(Ferroelectric Random Access Memory:铁电随机存取存储器)这样的其他各种各样的非易失性存储器。

[0260] 说明了本发明的几个实施方式,但是这些实施方式是作为例子而举出的,没有限定发明范围的意图。这些新的实施方式能够以其他的各种各样的方式来实施,在不脱离发明要旨的范围内能够进行各种省略、替换、变更。这些实施方式及其变形包含于发明的范围或要旨内,并且包含于权利要求书所记载的发明及其等同的范围内。

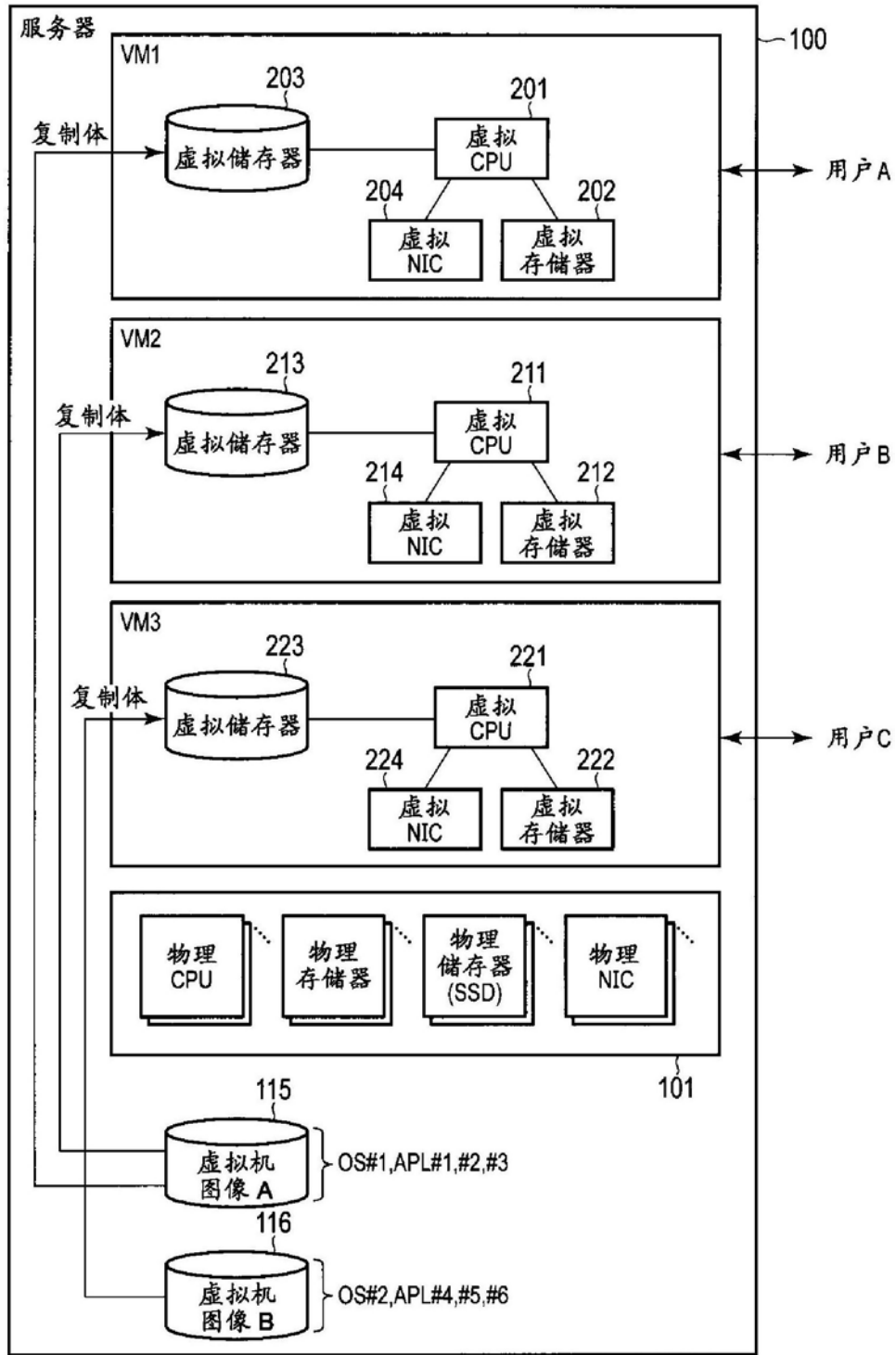


图1

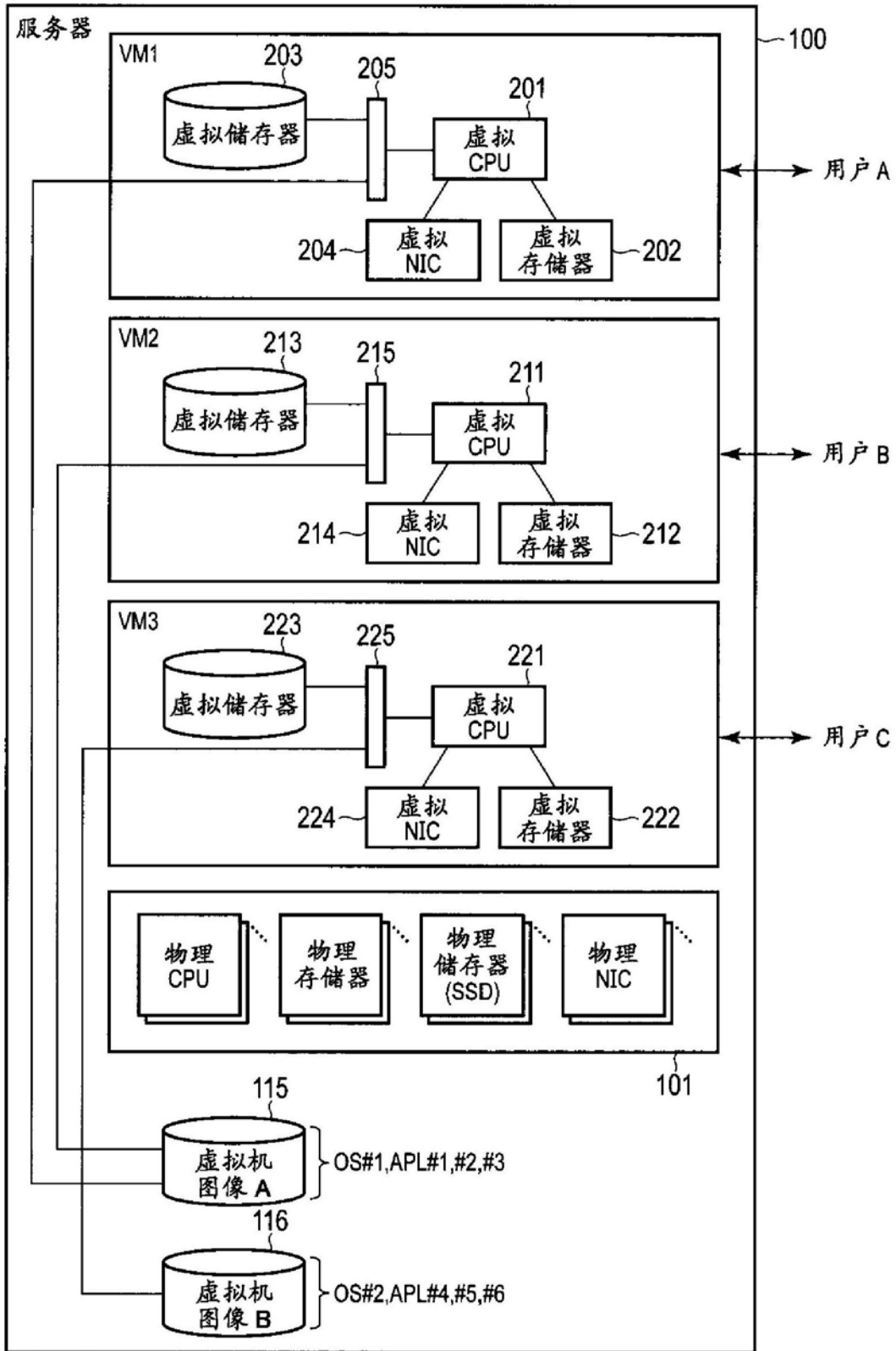


图2

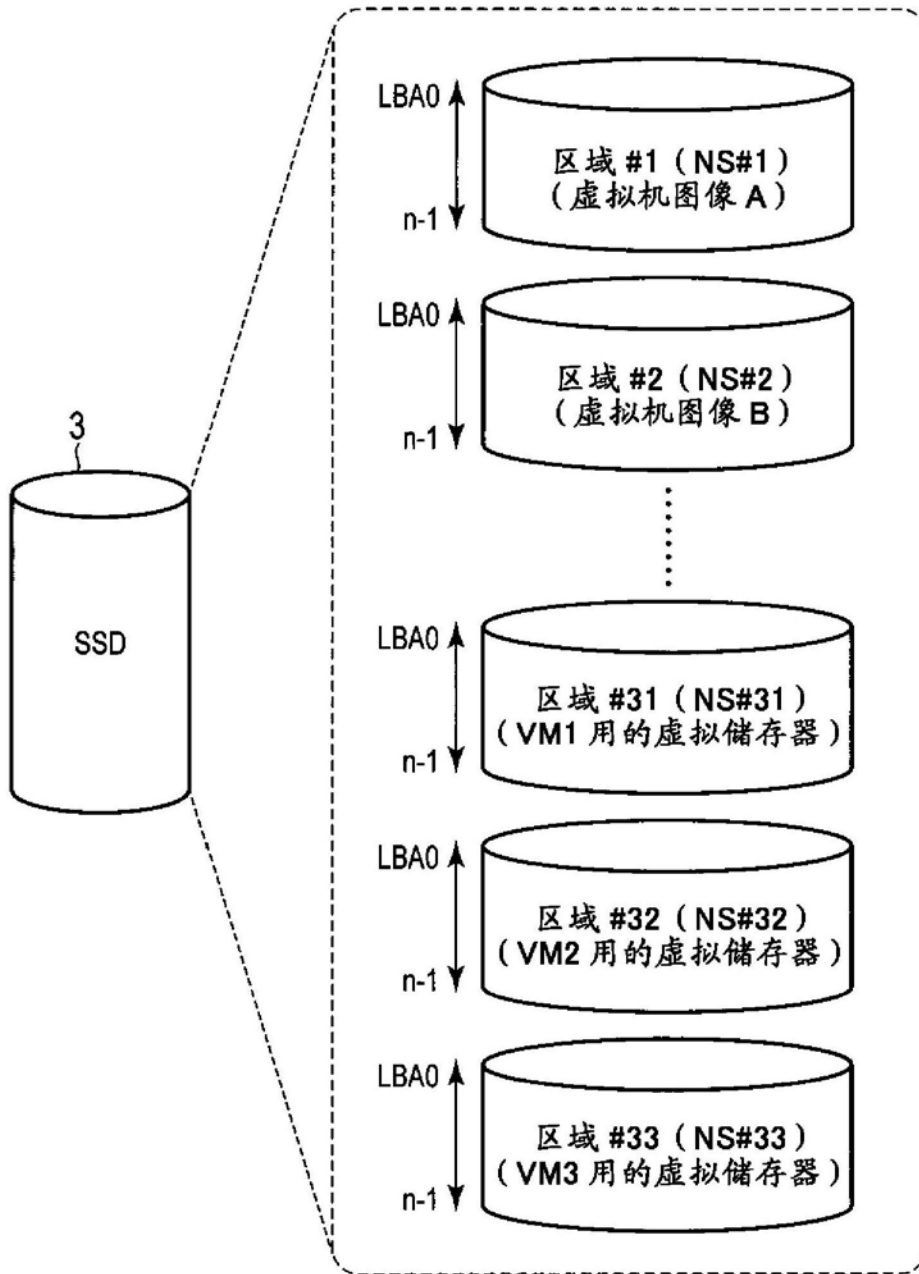


图3

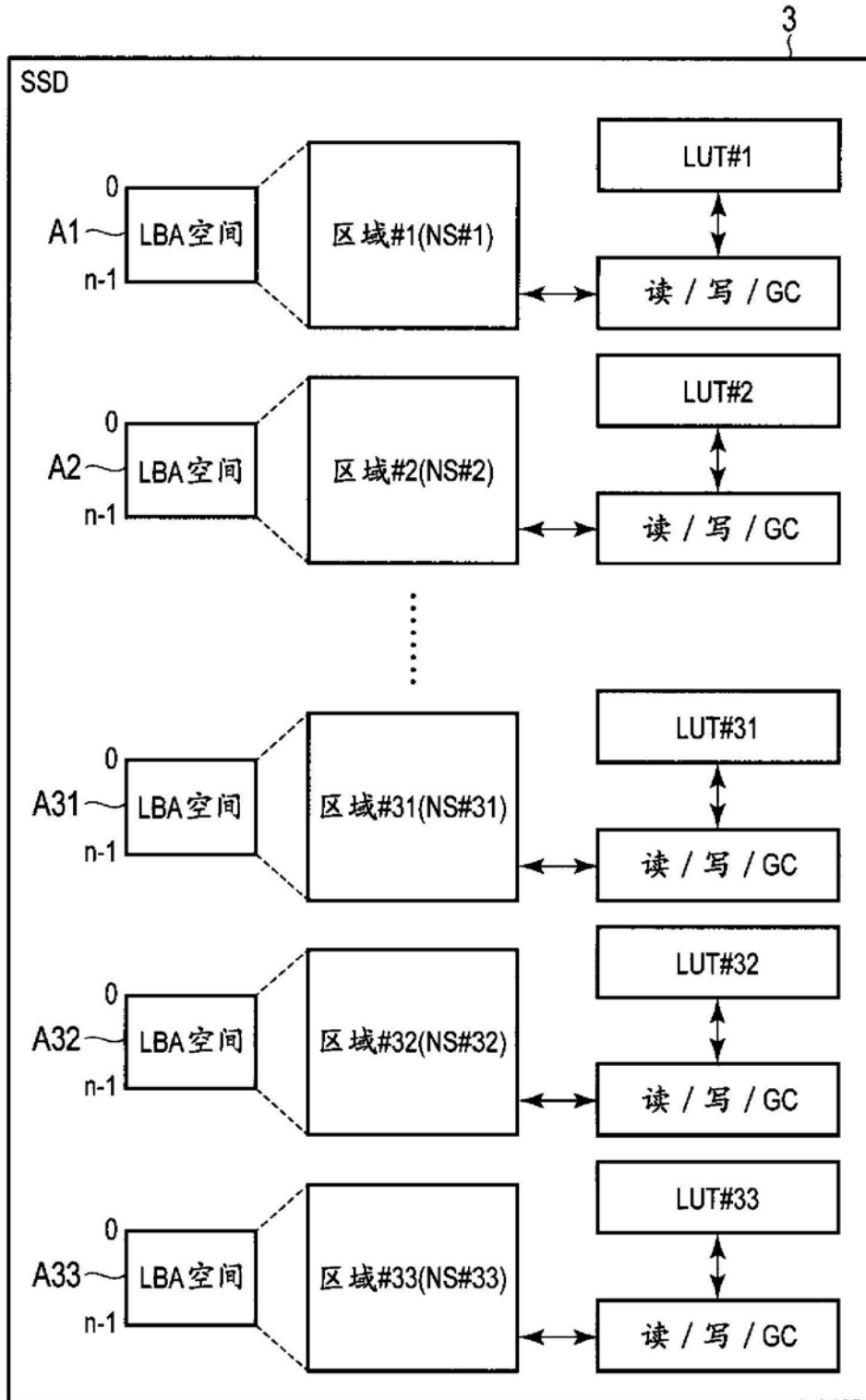


图4

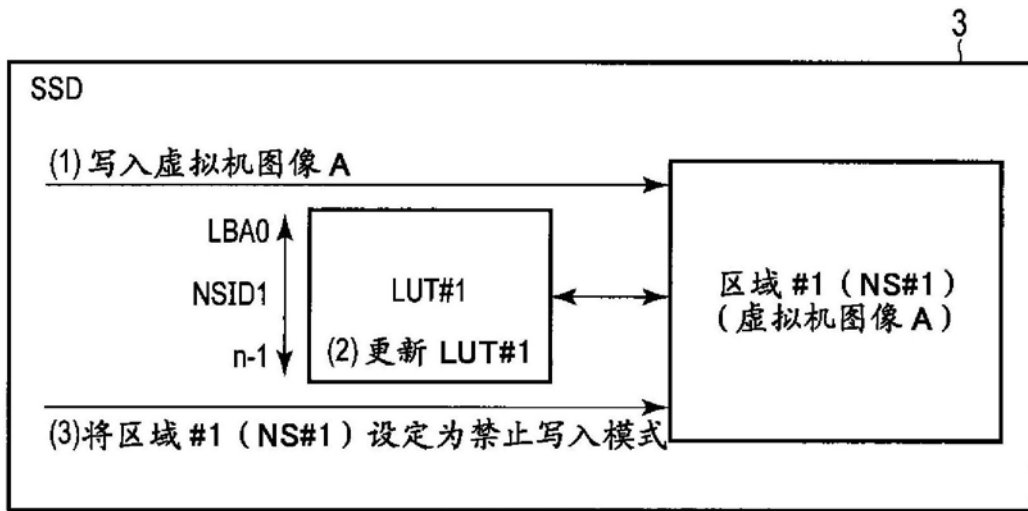


图5

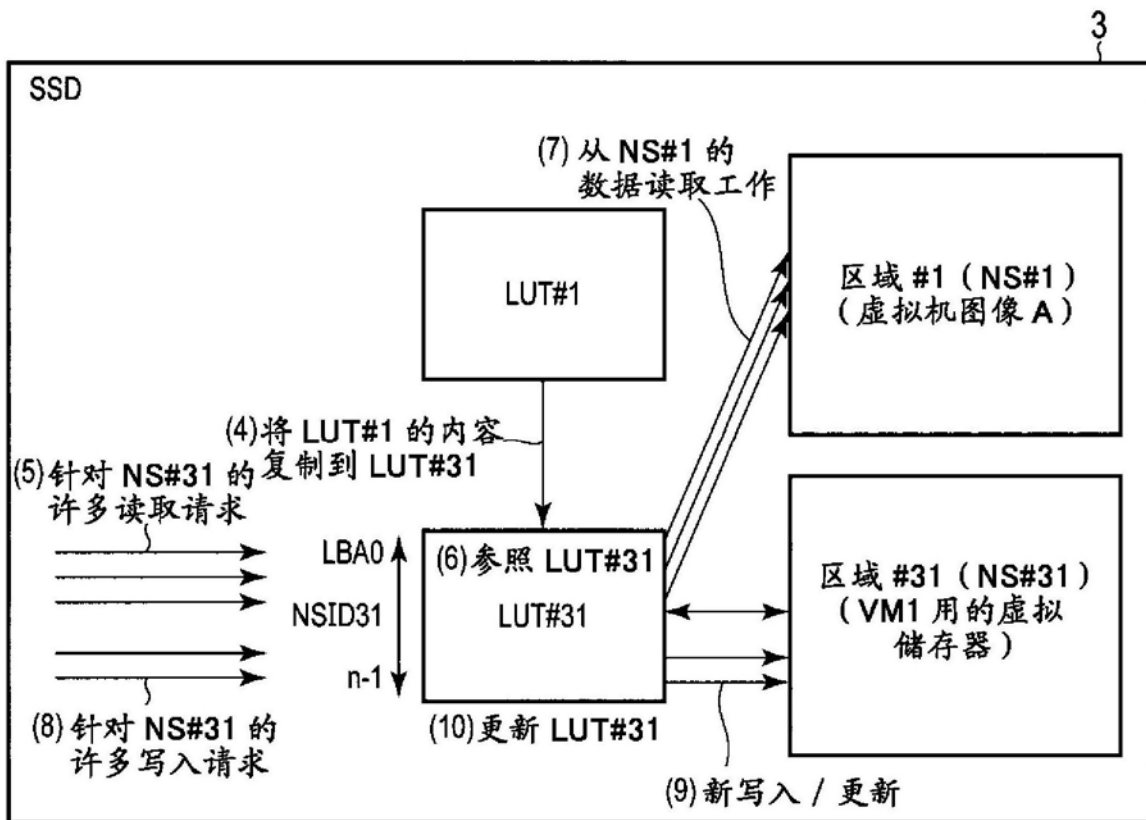


图6

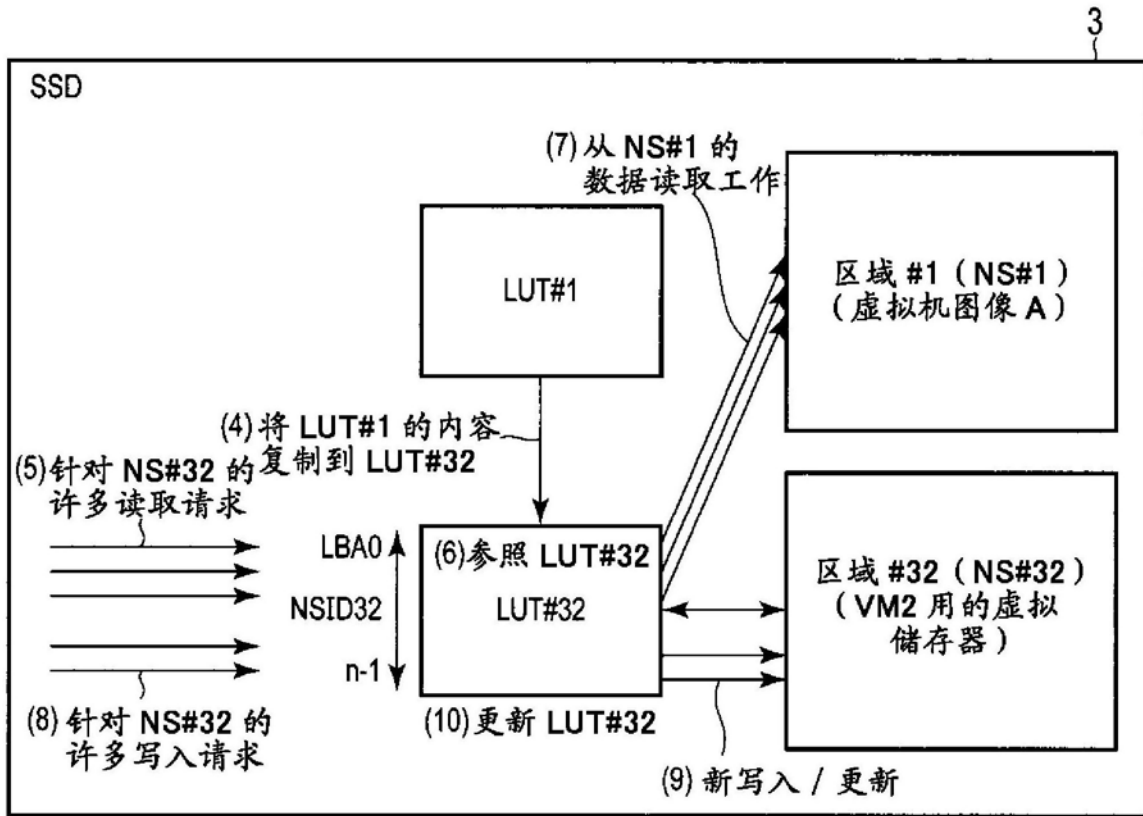


图7

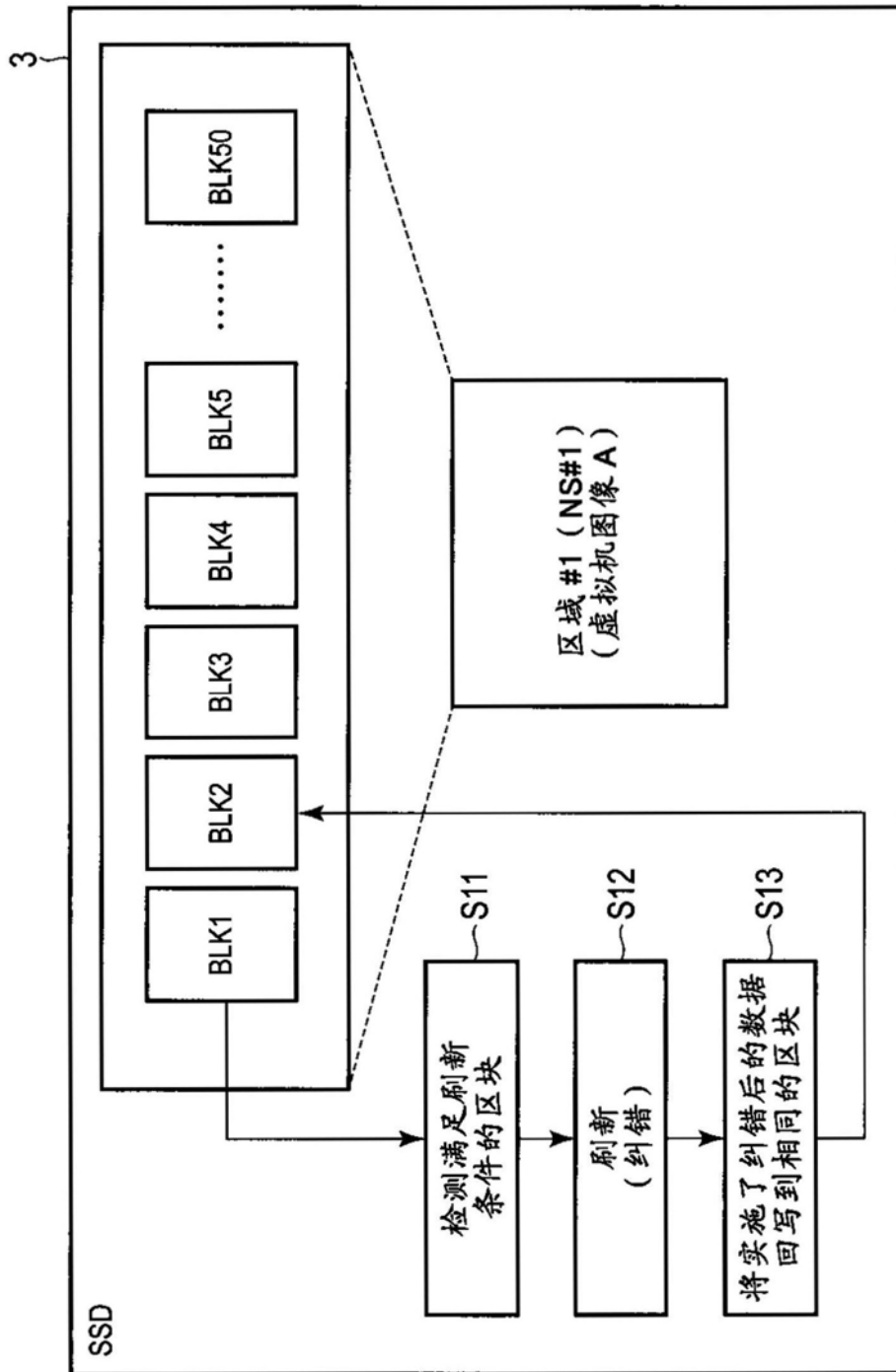


图8

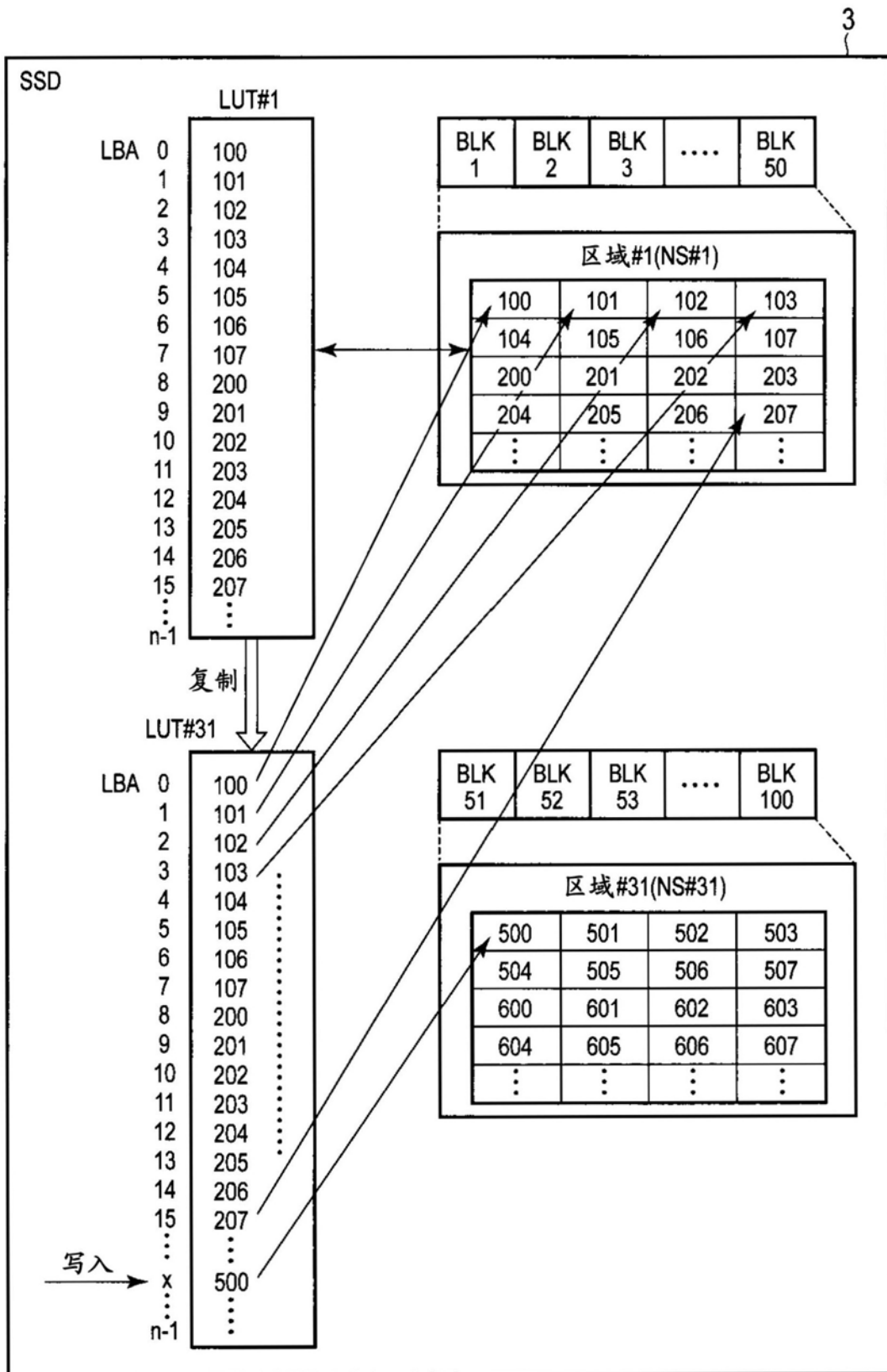


图9

3

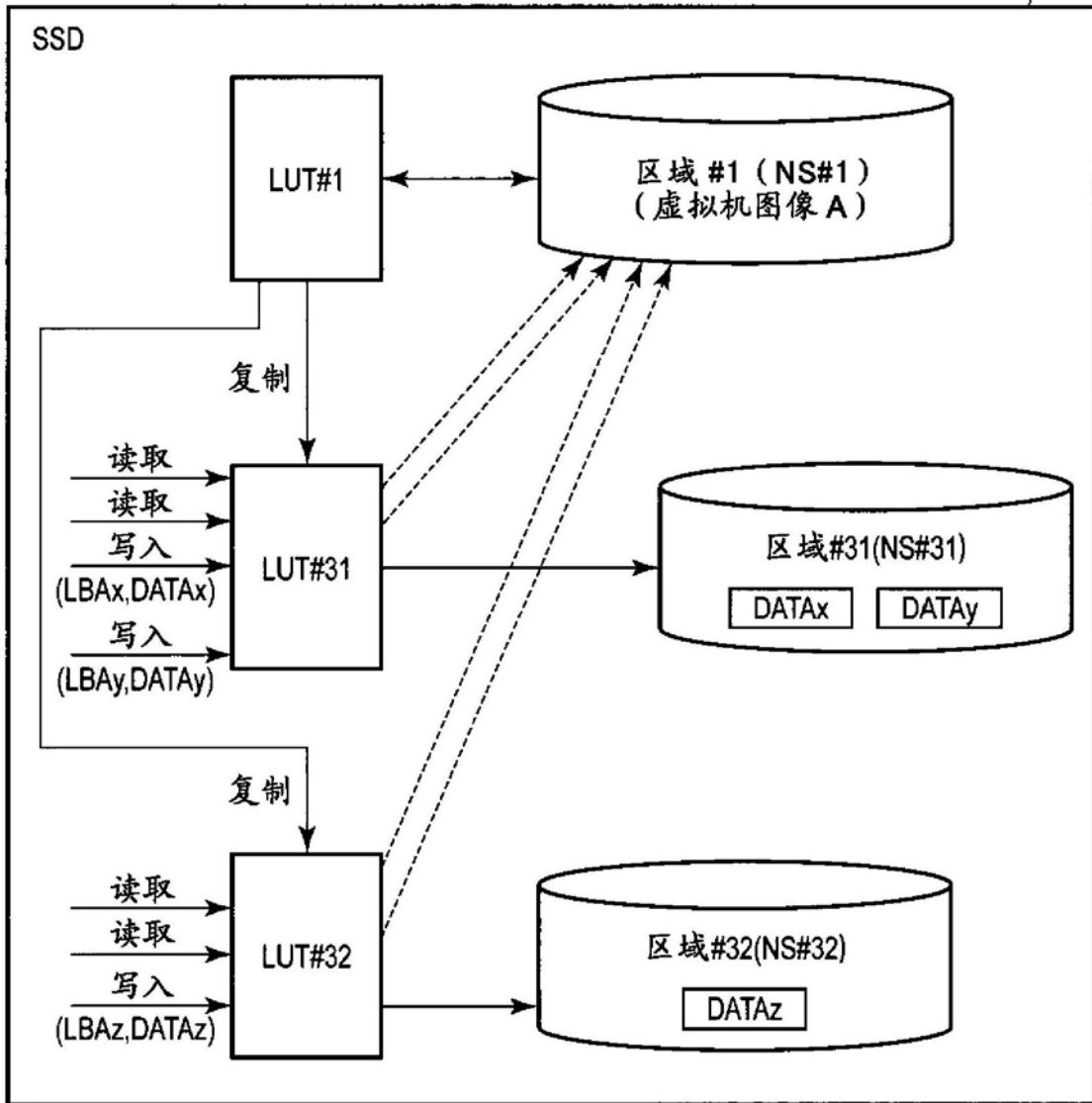


图10

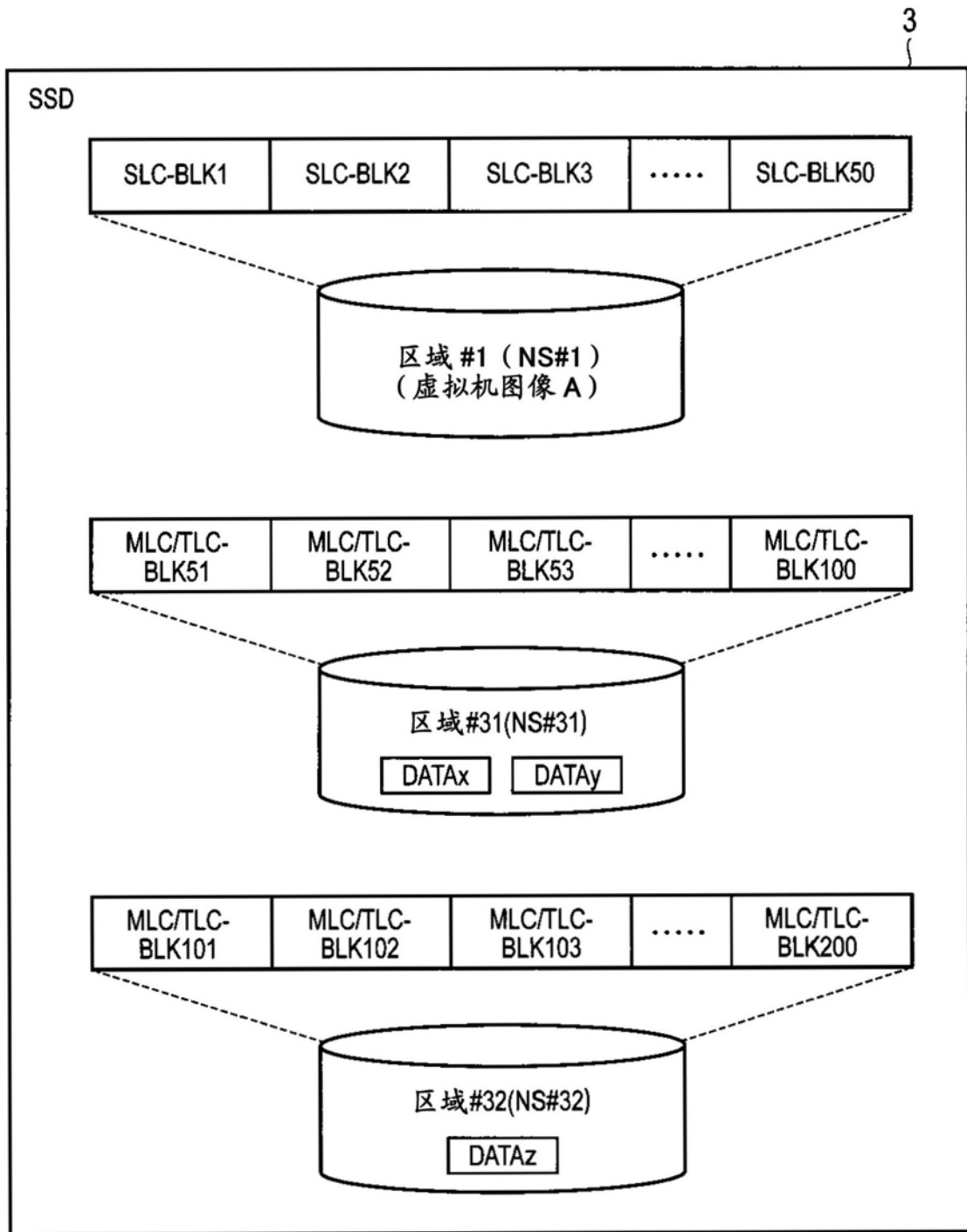


图11

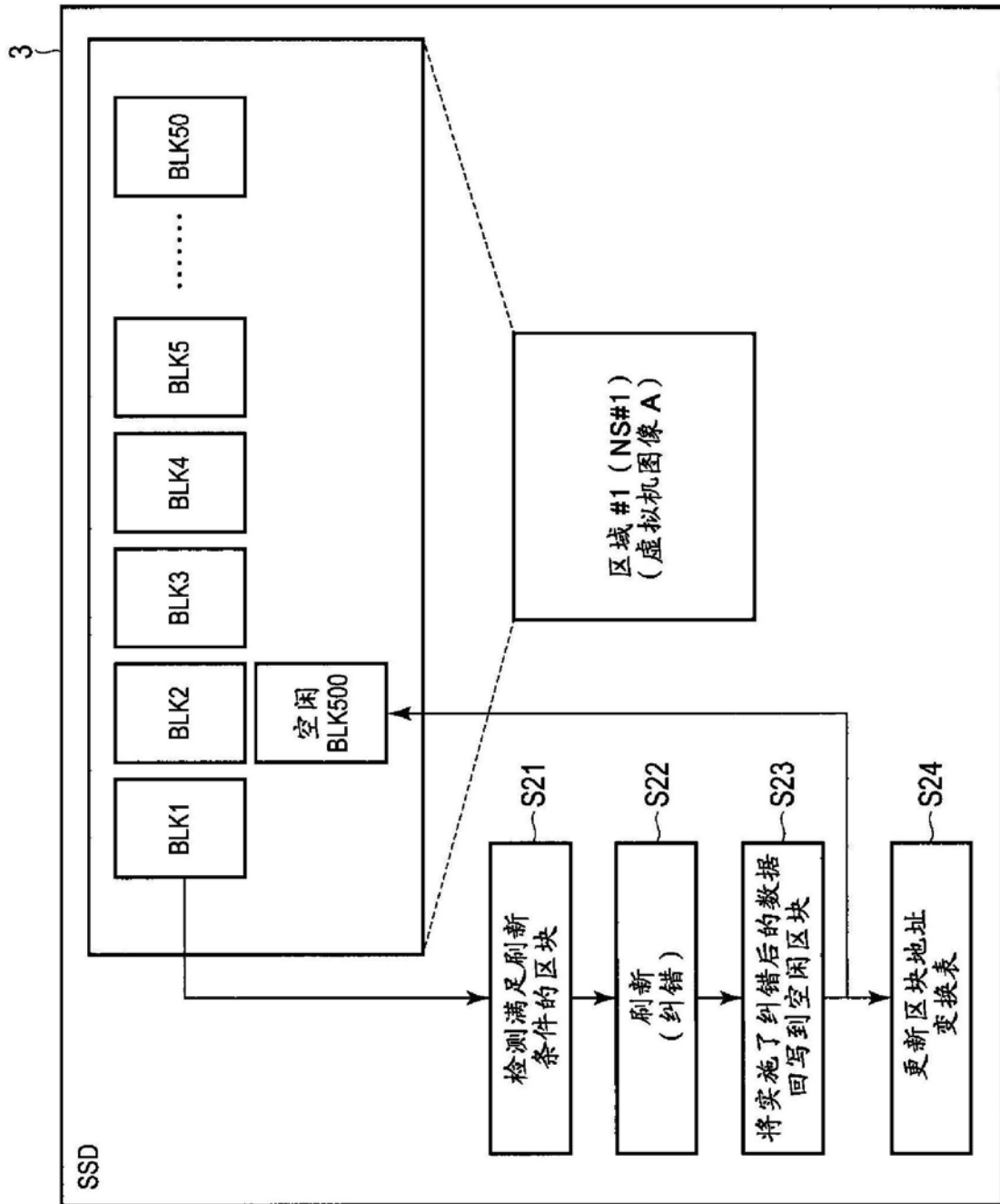


图12

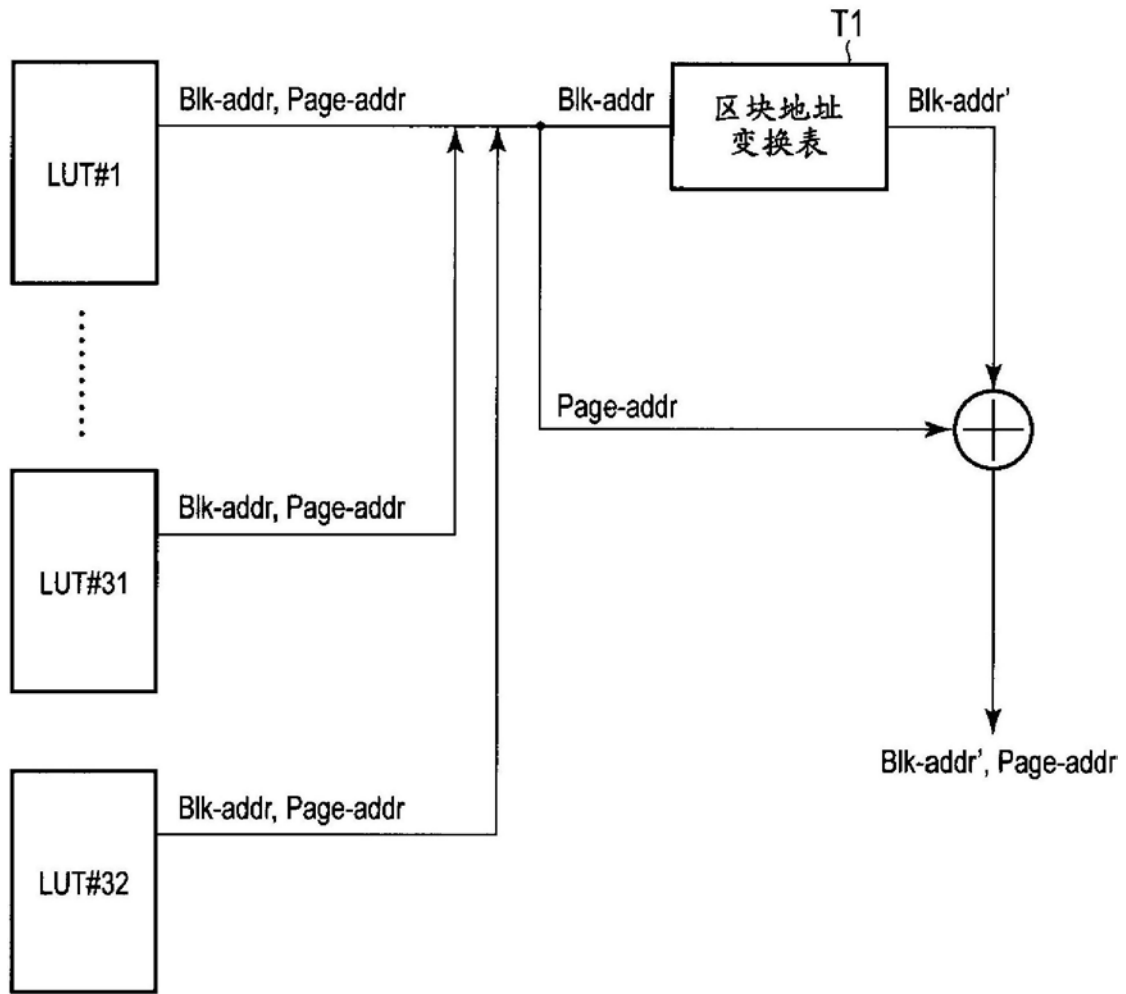


图13

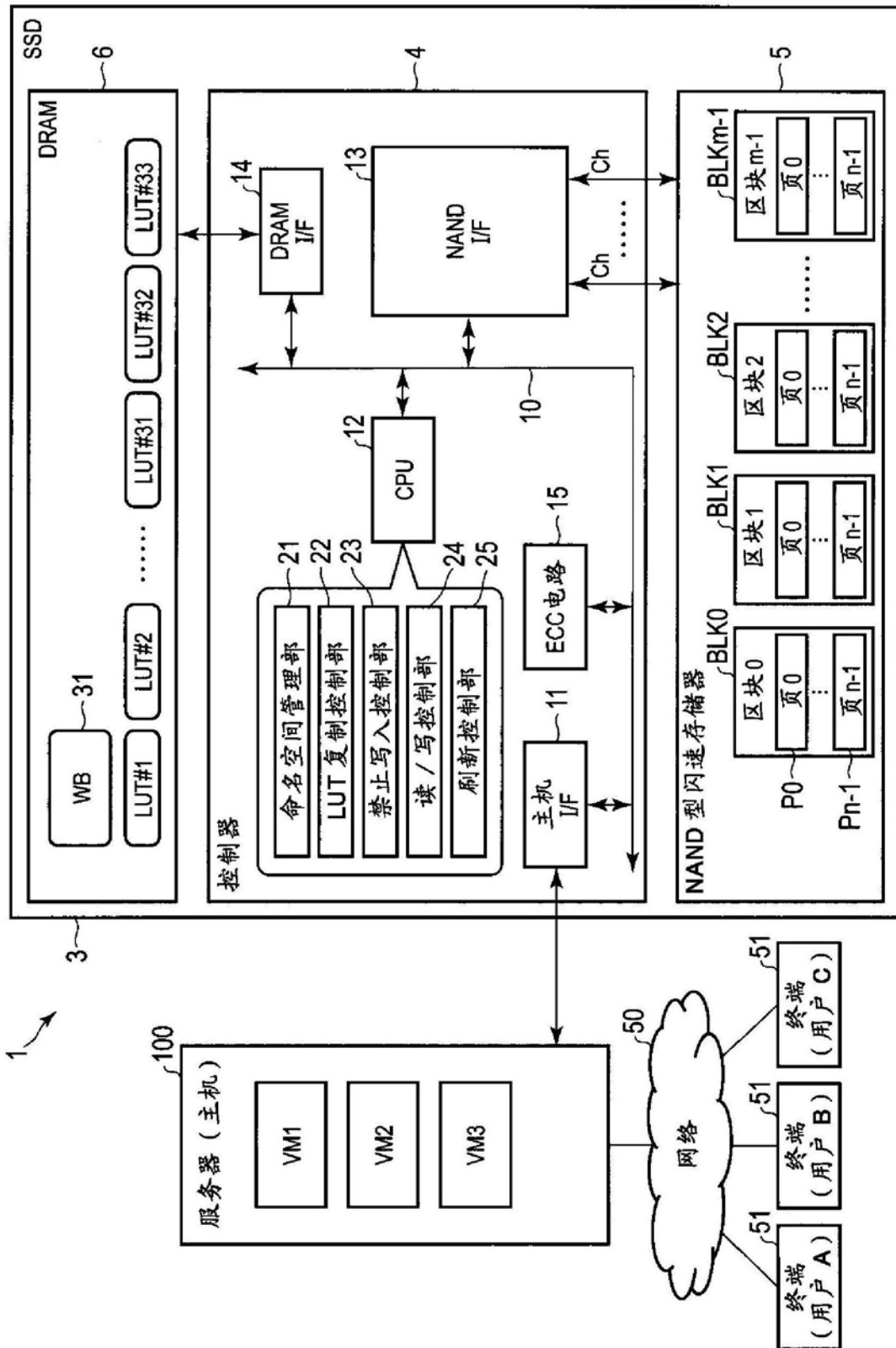


图14

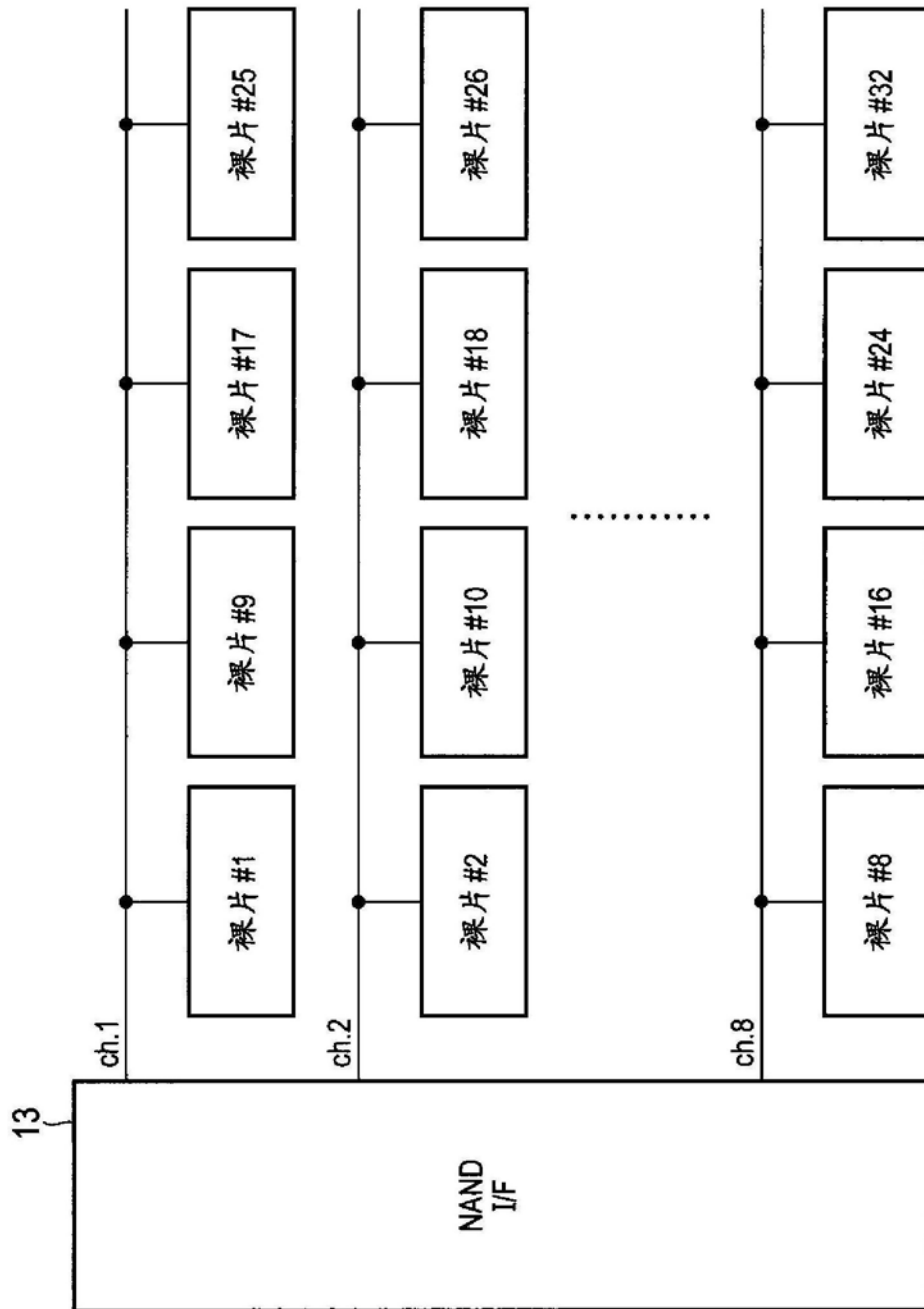


图15

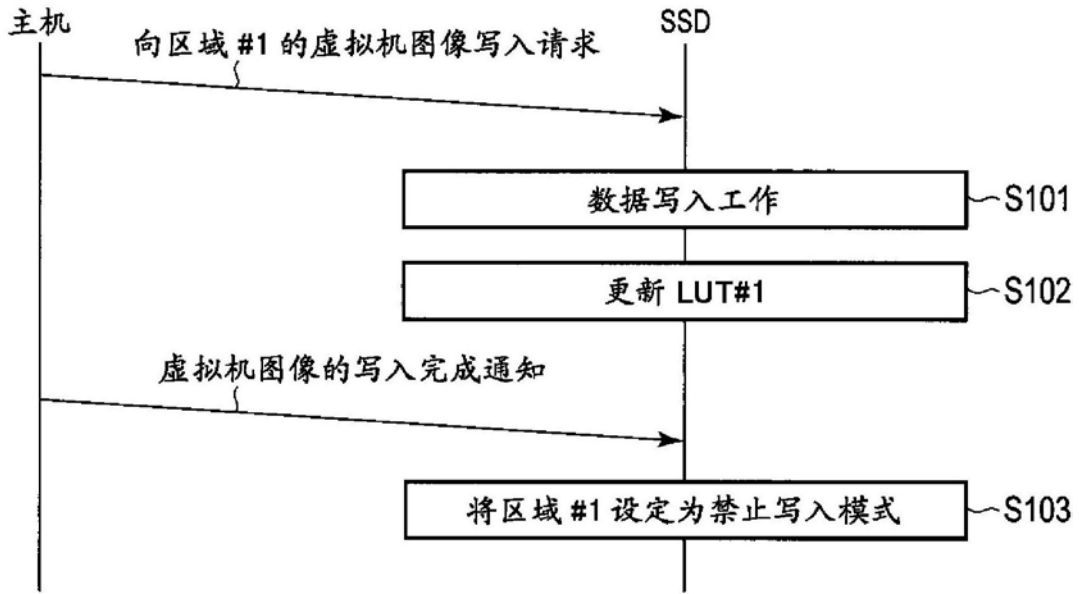


图16

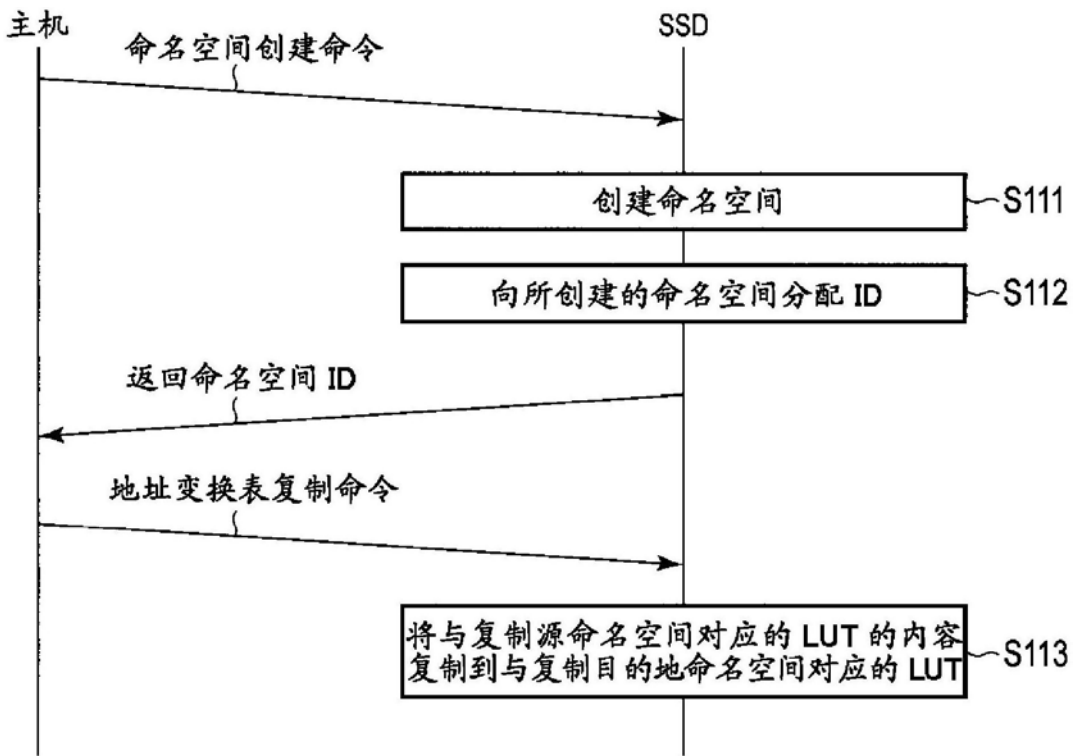


图17

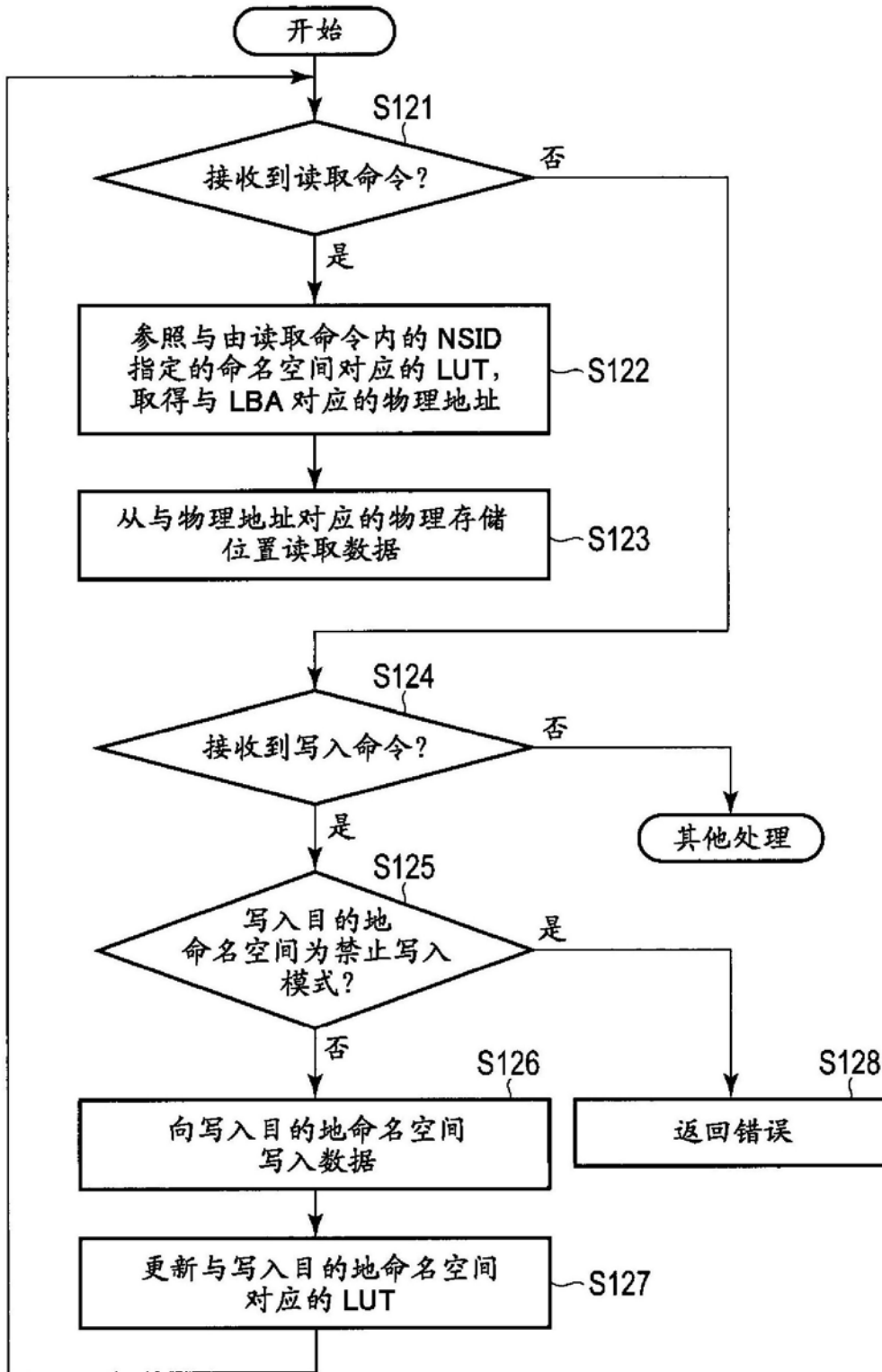


图18

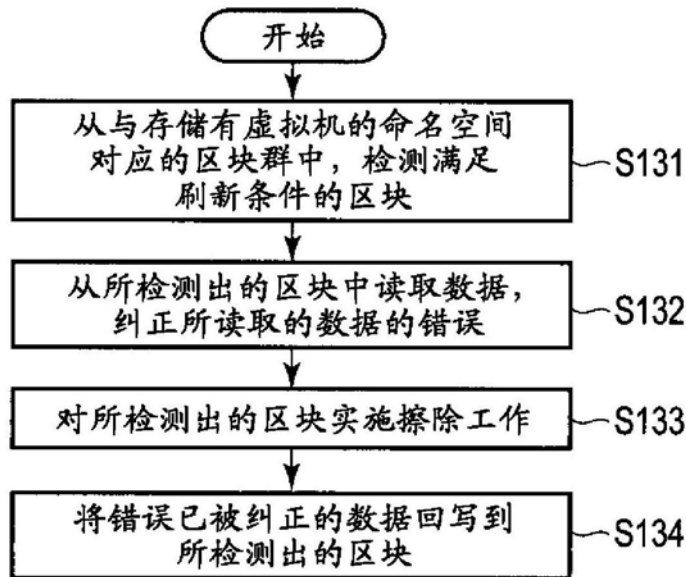


图19

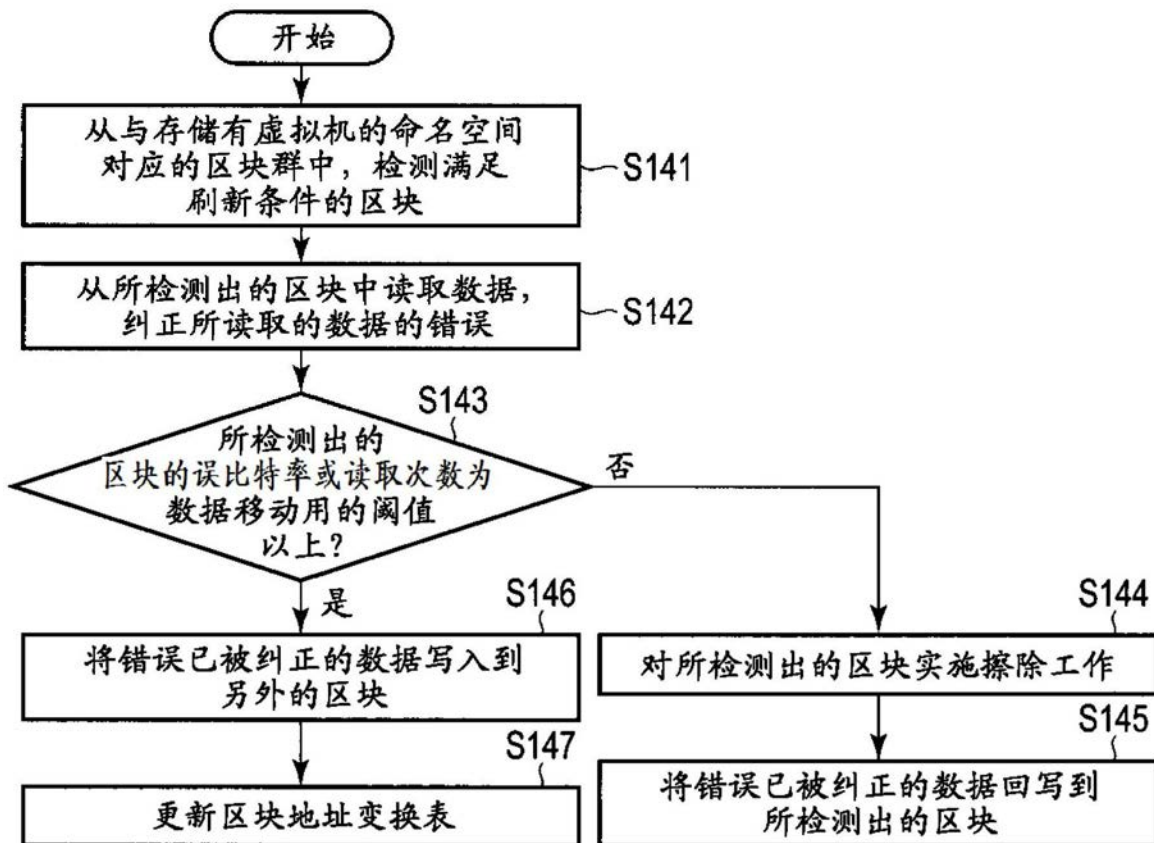


图20

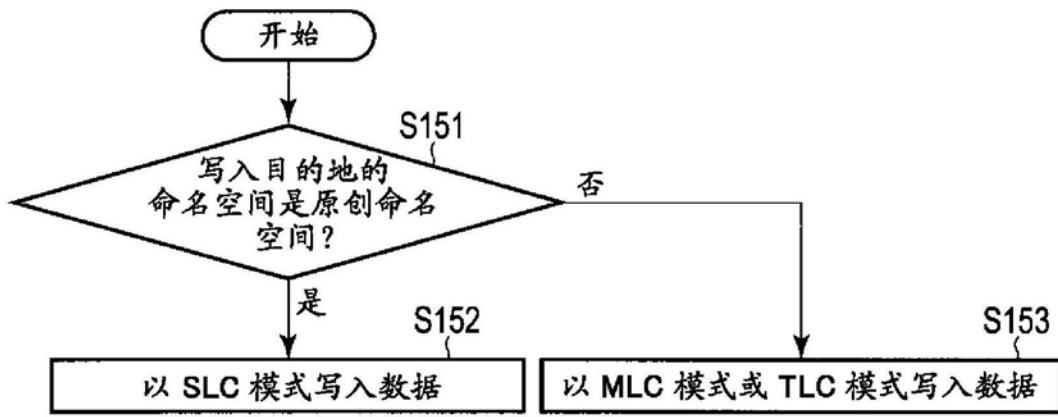


图21

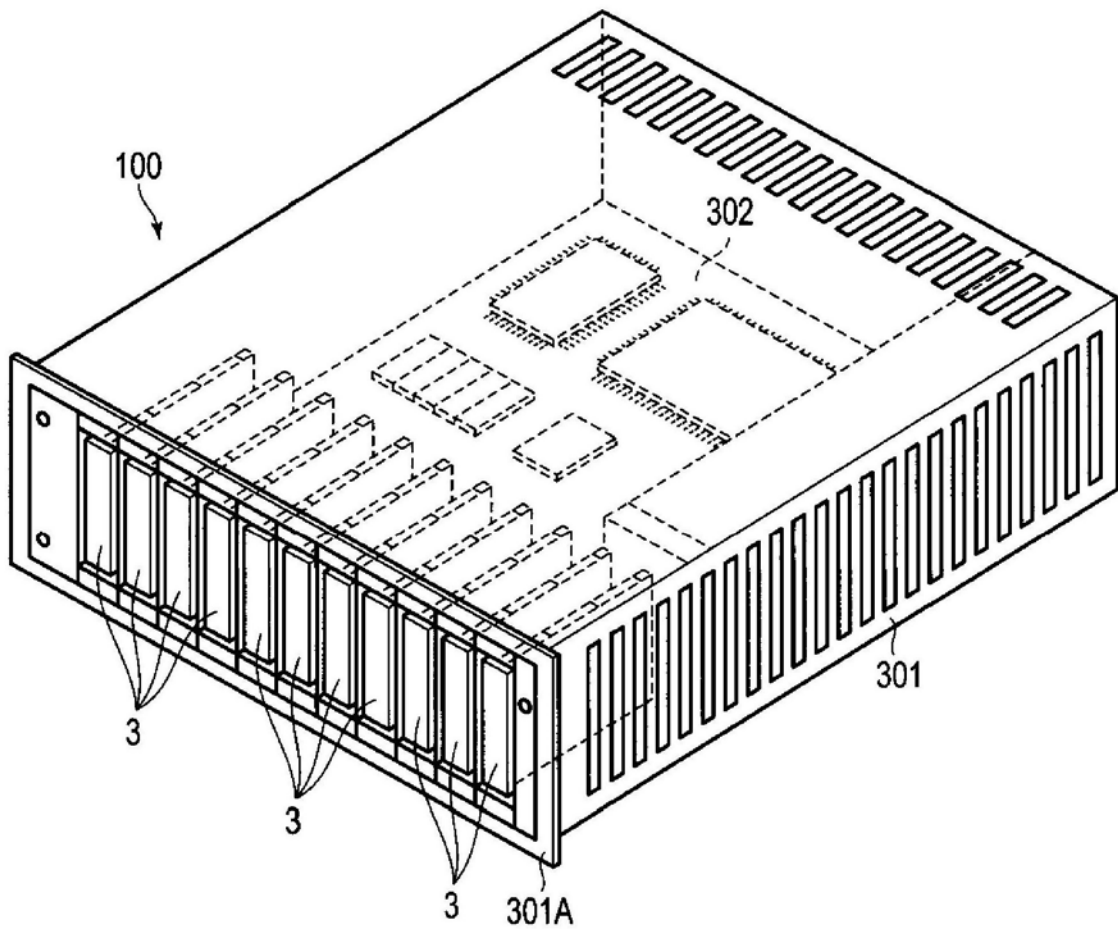


图22