



(12) 发明专利申请

(10) 申请公布号 CN 105408875 A

(43) 申请公布日 2016. 03. 16

(21) 申请号 201580000052. 1

(51) Int. Cl.

(22) 申请日 2015. 02. 21

G06F 12/02(2006. 01)

(30) 优先权数据

G06F 9/50(2006. 01)

14/187, 290 2014. 02. 23 US

G06F 12/00(2006. 01)

(85) PCT国际申请进入国家阶段日

2015. 04. 03

(86) PCT国际申请的申请数据

PCT/US2015/016985 2015. 02. 21

(87) PCT国际申请的公布数据

W02015/127327 EN 2015. 08. 27

(71) 申请人 拉姆伯斯公司

地址 美国加利福尼亚州

(72) 发明人 S·伍 R·佩雷戈 P·巴特拉

L·莱 C-M·杨

(74) 专利代理机构 北京市金杜律师事务所

11256

代理人 鄂迅

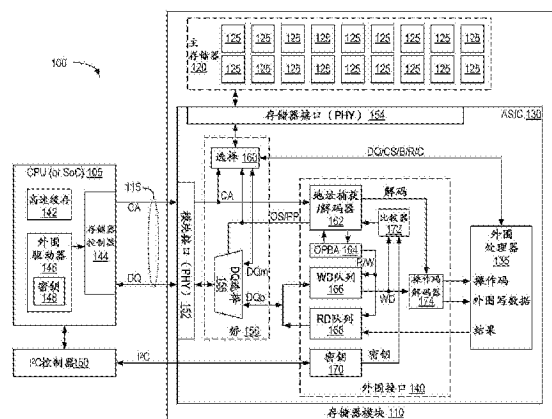
权利要求书4页 说明书23页 附图26页

(54) 发明名称

在存储器接口上的分布式过程执行和文件系统

(57) 摘要

在 DRAM 存储器通道上的存储器模块上包括非易失性存储器 (例如, 闪存、固态硬盘)。驻留在 DRAM 存储器通道上的非易失性存储器可以被集成到操作系统的现有文件系统结构中。驻留在 DRAM 存储器通道上的非易失性存储器可以被呈现为分布式文件系统的一部分或全部。请求和 / 或远程过程调用 (RPC) 请求, 或者与请求和 /RPC 相关联的信息, 可以通过 DRAM 存储器通道被路由至存储器模块, 以便于服务计算和 / 或分布式文件系统命令。



1. 一种系统,包括:

具有存储器控制器的中央处理单元 (CPU),所述存储器控制器被配置为使用并行数据通道接收来自模块存储器的模块数据的传送;以及

双列直插式存储器模块 (DIMM),所述 DIMM 被配置为经由所述存储器控制器向所述 CPU 发送所述模块数据,所述 DIMM 包括:

模块接口,所述模块接口被配置为使用并行数据通道发送来自所述 DIMM 的所述模块数据,

非易失性闪存库,以及

一个或多个集成电路,所述一个或多个集成电路被配置为使用所述并行数据通道发送所述模块数据并且使得所述模块数据能够被存储在动态随机存取存储器 (DRAM) 库中、被写入所述 DRAM 库以及从所述非易失性闪存库中被读取,其中所述一个或多个集成电路被配置为在由所述 DRAM 库采用的第一寻址方案和由所述非易失性闪存库采用的第二寻址方案之间转换,所述模块还被配置为执行模块块设备驱动器的至少一部分以服务于来自所述 CPU 的块 I/O 存储类型设备命令,所述块 I/O 存储类型设备命令使得所述模块数据从所述非易失性闪存库中被读取。

2. 根据权利要求 1 所述的系统,其中所述集成电路被配置为使得:

所述模块数据能够被写入采用所述第一寻址方案的所述 DRAM 库;

被写入采用所述第一寻址方案的所述 DRAM 库的所述模块数据能够被读取;

从非易失性闪存库读取的所述模块数据能够从由非易失性闪存库采用的所述第二寻址方案被转换为由所述 DRAM 库采用的所述第一寻址方案;以及

被转换为由所述非易失性闪存库采用的所述第二寻址方案的所述模块数据能够被写入采用所述第一寻址方案的所述 DRAM 库。

3. 根据权利要求 2 所述的系统,其中所述集成电路进一步包括缓冲器。

4. 根据权利要求 2 所述的系统,进一步包括桥接电路,所述桥接电路被配置为使得所述模块数据能够从所述非易失性闪存库中被读取或者被写入所述非易失性闪存库。

5. 根据权利要求 4 所述的系统,其中所述桥接电路被配置为执行串行化操作或反串行化操作。

6. 一种双列直插式存储器模块 (DIMM),包括:

模块接口,所述模块接口被配置为使用并行数据通道发送来自所述 DIMM 的模块数据;非易失性闪存库;以及

一个或多个集成电路,所述一个或多个集成电路被配置为使得所述模块数据能够从所述非易失性闪存库中被读取并且被写入动态随机存取存储器 (DRAM) 库,所述一个或多个集成电路还被配置为从所述 DRAM 库读取所述模块数据并经由所述并行数据通道发送从所述 DRAM 库中读取的所述模块数据,并且其中所述一个或多个集成电路被配置为在由所述 DRAM 库采用的第一寻址方案和由所述非易失性闪存库采用的第二寻址方案之间转换,所述模块执行模块块设备驱动器的至少一部分以服务于块 I/O 存储类型设备命令,所述块 I/O 存储类型设备命令使得所述模块数据从所述非易失性闪存库中被读取。

7. 根据权利要求 6 所述的 DIMM,其中所述 DRAM 库位于所述集成电路的外部。

8. 根据权利要求 6 所述的 DIMM,其中所述 DRAM 库包括动态数据率 (DDR) 存储器,并且

所述非易失性闪存库包括固态硬盘 (SSD)。

9. 根据权利要求 6 所述的 DIMM, 其中所述集成电路被配置为使得:

所述模块数据能够从采用所述第二寻址方案的所述非易失性闪存库中被读取;

从采用所述第二寻址方案的所述非易失性闪存库中读取的所述模块数据能够被写入所述 DRAM 库;

被写入所述 DRAM 库的所述模块数据能够从由所述非易失性闪存库采用的所述第二寻址方案被转换为由所述 DRAM 库采用的所述第一寻址方案; 以及

被转换为由所述 DRAM 库采用的所述第一寻址方案的所述模块数据能够从采用所述第一寻址方案的所述 DRAM 存储器库中被读取。

10. 根据权利要求 6 所述的 DIMM, 其中所述集成电路进一步包括缓冲器。

11. 根据权利要求 10 所述的 DIMM, 其中所述缓冲器进一步包括:

写缓冲器, 所述写缓冲器包括写数据队列电路; 以及

读缓冲器, 所述读缓冲器包括读数据队列电路。

12. 根据权利要求 10 所述的 DIMM, 其中所述集成电路被配置为使得:

所述模块数据能够从所述缓冲器被写入所述 DRAM 库;

所述模块能够从所述 DRAM 库被读入所述缓冲器; 以及

所述模块数据能够从所述非易失性闪存库被读入所述缓冲器。

13. 根据权利要求 6 所述的 DIMM, 其中所述集成电路进一步包括:

外围处理器, 所述外围处理器被配置为执行命令; 以及

控制器, 所述控制器被配置为根据来自所述外围处理器的指令在由所述 DRAM 库采用的所述第一寻址方案和由所述非易失性闪存库采用的所述第二寻址方案之间转换。

14. 根据权利要求 6 所述的 DIMM, 进一步包括桥接电路, 所述桥接电路被配置为使得所述模块数据能够从所述非易失性闪存库中被读取或者被写入所述非易失性闪存库。

15. 根据权利要求 14 所述的 DIMM, 其中所述桥接电路被配置为执行串行化操作或反串行化操作。

16. 根据权利要求 15 所述的 DIMM, 其中所述桥接电路位于所述集成电路内。

17. 一种双列直插式存储器模块 (DIMM), 包括:

模块接口, 所述模块接口被配置为使用并行数据通道接收进入所述 DIMM 的模块数据; 非易失性闪存库; 以及

一个或多个集成电路, 所述一个或多个集成电路被配置为使用所述并行数据通道接收所述模块数据并且使得所述模块数据能够被写入动态随机存取存储器 (DRAM) 库、从所述 DRAM 库中被读取以及被写入所述非易失性闪存库, 其中所述一个或多个集成电路被配置为在由所述 DRAM 库采用的第一寻址方案和由所述非易失性闪存库采用的第二寻址方案之间转换, 所述模块执行模块设备驱动器的至少一部分以服务于块 I/O 存储类型设备命令, 所述块 I/O 存储类型设备命令使得所述模块数据被写入所述非易失性闪存库。

18. 根据权利要求 17 所述的 DIMM, 其中所述 DRAM 库位于所述集成电路的外部。

19. 根据权利要求 17 所述的 DIMM, 其中所述 DRAM 库包括动态数据率 (DDR) 存储器, 并且所述非易失性闪存库包括固态硬盘 (SSD)。

20. 根据权利要求 17 所述的 DIMM, 其中所述集成电路被配置为使得:

所述模块数据能够被写入采用所述第一寻址方案的所述 DRAM 库；

被写入采用所述第一寻址方案的所述 DRAM 库的所述模块数据能够被读取；

从所述 DRAM 库读取的所述模块数据能够从由所述 DRAM 库采用的所述第一寻址方案被转换为由所述非易失性闪存库采用的所述第二寻址方案；以及

被转换为由所述非易失性闪存库采用的所述第二寻址方案的所述模块数据能够被写入采用所述第二寻址方案的所述非易失性闪存库。

21. 根据权利要求 17 所述的 DIMM, 其中所述集成电路进一步包括缓冲器。

22. 根据权利要求 21 所述的 DIMM, 其中所述缓冲器进一步包括：

写缓冲器, 所述写缓冲器包括写数据队列电路；以及

读缓冲器, 所述读缓冲器包括读数据队列电路。

23. 根据权利要求 21 所述的 DIMM, 其中所述集成电路被配置为使得：

所述模块数据能够从所述缓冲器被写入所述 DRAM 库；

所述模块数据能够从所述 DRAM 库被读入所述缓冲器；以及

所述模块数据能够从所述缓冲器被写入所述非易失性闪存库。

24. 根据权利要求 17 所述的 DIMM, 其中所述集成电路进一步包括：

外围处理器, 所述外围处理器被配置为执行命令；以及

控制器, 所述控制器被配置为根据来自所述外围处理器的指令在由所述 DRAM 库采用的所述第一寻址方案和由所述非易失性闪存库采用的所述第二寻址方案之间转换。

25. 根据权利要求 17 所述的 DIMM, 进一步包括桥接电路, 所述桥接电路被配置为使得所述模块数据能够从所述非易失性闪存库中被读取或者被写入所述非易失性闪存库。

26. 根据权利要求 25 所述的 DIMM, 其中所述桥接电路被配置为执行串行化操作或反串行化操作。

27. 根据权利要求 26 所述的 DIMM, 其中所述桥接电路位于所述集成电路内。

28. 一种双列直插式存储器模块 (DIMM), 包括：

模块接口, 所述模块接口被配置为使用并行数据通道接收进入所述 DIMM 的模块数据；非易失性闪存库；

用于一个或多个集成电路的装置, 所述用于一个或多个集成电路的装置被配置为使用所述并行数据通道接收所述模块数据并且使得所述模块数据能够被写入动态随机存取存储器 (DRAM) 库、从所述 DRAM 库被读取以及被写入所述非易失性闪存库, 其中所述一个或多个集成电路被配置为在由所述 DRAM 库采用的第一寻址方案和由所述非易失性闪存库采用的第二寻址方案之间转换；以及

用于所述模块的装置, 所述模块执行模块设备驱动器的至少一部分以服务于块 I/O 存储类型设备命令, 所述块 I/O 存储类型设备命令使得所述模块数据被写入所述非易失性闪存库。

29. 根据权利要求 28 所述的 DIMM, 其中所述用于一个或多个集成电路的装置被配置为使得：

所述模块数据能够被写入采用所述第一寻址方案的所述 DRAM 库；

被写入采用所述第一寻址方案的所述 DRAM 库的所述模块数据能够被读取；

从所述 DRAM 库读取的所述模块数据能够从由所述 DRAM 库采用的所述第一寻址方案被

转换为由所述非易失性闪存库采用的所述第二寻址方案；以及

被转换为由所述非易失性闪存库采用的所述第二寻址方案的所述模块数据能够被写入采用所述第二寻址方案的所述非易失性闪存库。

30. 根据权利要求 28 所述的 DIMM, 其中所述用于一个或多个集成电路的装置进一步包括缓冲器。

31. 根据权利要求 28 所述的 DIMM, 进一步包括桥接电路, 所述桥接电路被配置为使得所述模块数据能够从所述非易失性闪存库中被读取或者被写入所述非易失性闪存库。

32. 一种系统, 包括:

具有存储器控制器的中央处理单元 (CPU), 所述存储器控制器被配置为使用并行数据通道从所述 CPU 向模块存储器直接传送模块数据; 以及

双列直插式存储器模块 (DIMM), 所述 DIMM 被配置为从所述 CPU 接收所述模块数据, 所述 DIMM 包括:

模块接口, 所述模块接口被配置为使用并行数据通道接收进入所述 DIMM 的所述模块数据,

非易失性闪存库, 以及

一个或多个集成电路, 所述一个或多个集成电路被配置为使用所述并行数据通道接收所述模块数据并且使得所述模块数据能够被写入动态随机存取存储器 (DRAM) 库、从所述 DRAM 库被读取以及被写入所述非易失性闪存库, 其中所述一个或多个集成电路被配置为在由所述 DRAM 库采用的第一寻址方案和由所述非易失性闪存库采用的第二寻址方案之间转换, 所述模块还被配置为执行模块块设备驱动器的至少一部分以服务于来自所述 CPU 的块 I/O 存储类型设备命令, 所述块 I/O 存储类型设备命令使得所述模块数据被写入所述非易失性闪存库。

33. 根据权利要求 32 所述的系统, 其中所述集成电路被配置为使得:

所述模块数据能够被写入采用所述第一寻址方案的所述 DRAM 库;

被写入采用所述第一寻址方案的所述 DRAM 库的所述模块数据能够被读取;

从所述 DRAM 库读取的所述模块数据能够从由所述 DRAM 库采用的所述第一寻址方案被转换为由所述非易失性闪存库采用的所述第二寻址方案; 以及

被转换为由所述非易失性闪存库采用的所述第二寻址方案的所述模块数据能够被写入采用所述第二寻址方案的所述非易失性闪存库。

34. 根据权利要求 33 所述的系统, 其中所述集成电路进一步包括缓冲器。

35. 根据权利要求 33 所述的系统, 进一步包括桥接电路, 所述桥接电路被配置为使得所述模块数据能够从所述非易失性闪存库中被读取或者被写入所述非易失性闪存库。

36. 根据权利要求 35 所述的系统, 其中所述桥接电路被配置为执行串行化操作或反串行化操作。

## 在存储器接口上的分布式过程执行和文件系统

### [0001] 相关申请的交叉参考

[0002] 本申请要求 2014 年 2 月 23 日提交的名称为“DISTRIBUTED PROCEDURE EXECUTION AND FILE SYSTEMS ON A MEMORY INTERFACE”的美国专利申请第 14/187,290 号的权利,其内容为了所有目的通过引用并入于此。本申请还与 2013 年 3 月 15 日提交的名称为“METHODS AND SYSTEMS THAT COMMUNICATE VIA A MEMORY MODULE INTERFACE”的美国专利临时申请第 61/788,196 号有关,与 2012 年 12 月 3 日提交的名称为“METHODS AND SYSTEMS FOR SUPPORTING FUNCTIONS ON A LEGACY MEMORY INTERFACE”的美国专利临时申请第 61/732,898 号有关,与 2013 年 10 月 27 日提交的名称为“COMMUNICATION VIA A MEMORY INTERFACE”的美国专利申请第 14/064,167 号有关,与 2011 年 8 月 24 日提交的名称为“METHODS AND SYSTEMS FOR MAPPING A PERIPHERAL FUNCTION ONTO A LEGACY MEMORY INTERFACE”的美国专利临时申请第 61/526,953 号有关,与 2012 年 5 月 16 日提交的名称为“METHODS AND SYSTEMS FOR MAPPING A PERIPHERAL FUNCTION ONTO A LEGACY MEMORY INTERFACE”的美国专利临时申请第 61/647,986 号有关,与 2012 年 7 月 12 日提交的名称为“METHODS AND SYSTEMS FOR MAPPING A PERIPHERAL FUNCTION ONTO A LEGACY MEMORY INTERFACE”的美国专利临时申请第 61/670,874 号有关,与 2012 年 8 月 20 日提交的名称为“METHODS AND SYSTEMS FOR MAPPING A PERIPHERAL FUNCTION ONTO A LEGACY MEMORY INTERFACE”的美国专利临时申请第 61/691,134 号有关,与 2012 年 8 月 23 日提交的名称为“METHODS AND SYSTEMS FOR MAPPING A PERIPHERAL FUNCTION ONTO A LEGACY MEMORY INTERFACE”的国际申请 PCT/US12/52000 有关,与 2012 年 8 月 23 日提交的名称为“METHODS AND SYSTEMS FOR MAPPING A PERIPHERAL FUNCTION ONTO A LEGACY MEMORY INTERFACE”的国际申请 PCT/US12/52043 有关,与 2012 年 8 月 23 日提交的名称为“METHODS AND SYSTEMS FOR MAPPING A PERIPHERAL FUNCTION ONTO A LEGACY MEMORY INTERFACE”的国际申请 PCT/US12/52052 有关,以及与 2012 年 8 月 23 日提交的名称为“METHODS AND SYSTEMS FOR MAPPING A PERIPHERAL FUNCTION ONTO A LEGACY MEMORY INTERFACE”的国际申请 PCT/US12/52059 有关,其全部的十一个的内容为了所有目的通过引用并入于此。

### 附图说明

[0003] 在附图中通过示例示例的方式而非限制的方式示出本发明,并且其中类似的参考标号指代类似的元件。

[0004] 图 1 描绘了根据一个实施例的存储器系统 100。系统 100 包括经由主存储器接口 115 与存储器模块 110 交流数据 DQ、命令和地址 CA 的 CPU 105。

[0005] 图 2 示出了根据一个实施例的数据结构 200,外围驱动器 146 在高速缓存器 142 中汇集并存储该数据结构 200 以将命令和数据传送至 OPBA 命令端口。

[0006] 图 3 是示出根据一个实施例的图 1 中的存储器模块 110 如何管理来自 CPU 105 的对于访问主存储器 120 和来自外围处理器 135 的支持的 OS 请求的流程图。

[0007] 图 4 描绘了根据另一实施例的存储器系统 400。

[0008] 图 5 描绘了根据实施例的支持外围命令广播和可配置的数据宽度的存储器模块 500。

[0009] 图 6 示出了外围驱动器（例如，图 1 的驱动器 146）在高速缓存器中汇集和存储以将指令和数据传送至 32 个模块 500 的 OPBA 命令端口的数据结构 600，在四个存储器通道中的每个存储器通道上有 8 个模块。

[0010] 图 7 是混合闪存 /DRAM 存储器模块的示图。

[0011] 图 8 是示出混合闪存 /DRAM 计算子系统的框图。

[0012] 图 9 是混合闪存 /DRAM 存储器模块的示图。

[0013] 图 10 是示出混合闪存 /DRAM 计算子系统的框图。

[0014] 图 11 是混合盘类型非易失性存储和 DRAM 存储器模块的示图。

[0015] 图 12 是示出混合盘类型非易失性存储和 DRAM 存储器模块计算子系统的框图。

[0016] 图 13 是用于访问存储在 DRAM 存储器通道上的非易失性存储器中的文件数据的系统的示图。

[0017] 图 14 是用于远程地访问存储在 DRAM 存储器通道上的非易失性存储器中的文件数据的系统的示图。

[0018] 图 15 是用于使用块 I/O 存储类型设备命令远程地访问存储在 DRAM 存储器通道上的非易失性存储器中的文件数据的系统的示图。

[0019] 图 16 是用于服务对于存储在 DRAM 存储器通道上的非易失性存储器中的数据远程请求的系统的示图。

[0020] 图 17 是用于服务向 DRAM 存储器通道上的固态盘的远程请求的系统的示图。

[0021] 图 18 是用于分布式文件处理的系统的示图。

[0022] 图 19 是示出操作存储器模块的方法的流程图。

[0023] 图 20 是示出操作存储器模块的方法的流程图。

[0024] 图 21 是示出操作存储器模块的方法的流程图。

[0025] 图 22 是示出操作存储器模块的方法的流程图。

[0026] 图 23 是示出服务远程过程调用请求的方法的流程图。

[0027] 图 24 是示出传送和服务远程过程调用请求的方法的流程图。

[0028] 图 25 是示出执行分布式处理功能的方法的流程图。

[0029] 图 26 是示出操作存储器模块的方法的流程图。

[0030] 图 27 是计算机的框图。

### 具体实施方式

[0031] 在一个实施例中，非易失性存储器（例如，闪存、固态硬盘）被包括在 DRAM 存储器通道上的存储器模块上。这帮助由非易失性存储器提供的大容量存储驻留于更接近系统的计算机资源（即，CPU）。此外，CPU 和非易失性存储器可以使用 DRAM 存储器通道的相对高的带宽来传送命令和 / 或数据（可能经由控制器）。

[0032] 驻留在 DRAM 存储器通道上的非易失性存储器可以被集成到流行的操作系统的现有文件系统结构中，以利于采纳到流行的编程范例中。一种这样的集成方法是使驻留在 DRAM 存储器通道上的非易失性存储器呈现为分布式文件系统的一部分或全部。例如，可

以修改流行的操作系统以将驻留在 DRAM 存储器通道上的非易失性存储器呈现为网络文件系统 (NFS)、Hadoop 文件系统和 / 或其他分布式文件系统 (DFS) 的一部分。远程过程调用 (RPC) 请求和 / 或源自本地 CPU 的请求 (使用 RPC 机制 / 软件层或者另一种方式), 或者与本地请求和 / 或 RPC 相关联的信息可以通过 DRAM 存储器通道被路由至存储器模块, 以便于服务计算和 / 或 DFS 命令。

[0033] 图 1 描绘了根据一个实施例的存储器系统 100。系统 100 包括经由主存储器接口 115 与存储器模块 110 交流数据 DQ 以及命令和地址 CA 的 CPU 105。在该示例中, 存储器模块 110 包括主存储器 120, 主存储器 120 由 IC 主存储器设备 125 的组来组成。模块 110 附加地包括专用集成电路 (ASIC) 130, 该专用集成电路 (ASIC) 130 充当缓冲设备以在 CPU 105 和主存储器 120 之间中继命令和数据。ASIC 130 附加地包括嵌入式处理器 135, 该嵌入式处理器 135 共享对于支持外围功能、诸如图形或计算处理的主存储器的访问, 用于改善的总体的系统性能。外围接口 140 以将修改 CPU 105 的需求最小化或者消除的方式来促进 CPU105 和外围处理器 135 之间的外围命令和数据的通信, 从而减少对于采用具有集成处理支持的主存储器模块的实践阻碍。

[0034] 可能是片上系统 (SoC) 的 CPU105 包括高速缓存器 142 和存储器控制器 144, 并且执行软件外围驱动器 146。如本文详细描述, 驱动器 146 具有对密钥存储 148 的访问, 并且例如, 可以是提供来支持与模块 110 的通信的软件和 / 或固件驱动器。驱动器 146 可以是由制造商或消费者加载的软件, 并且可以考虑到遗留存储器系统与少量硬件修改或无硬件修改的兼容性。

[0035] I2C 控制器 150 和相关的串行总线提供可靠的标准化通道, 该标准化通道允许 CPU 105 访问来自模块 110 的特定于模块的配置信息, 该配置信息通常存储在 EEPROM (未示出) 中。该信息用于使用本领域技术人员公知的技术来初始化相对高性能的接口 115。在描绘的实施例中, 使用这些相同的资源以在外围接口 140 和密钥存储 148 之间共享密钥。控制器 150 可以与 CPU 105 的其他部件集成。

[0036] ASIC 130 包括两个物理接口 (PHY), 这两者可以是传统的物理接口。第一物理接口是模块接口 152, 该模块接口 152 从控制器 144 接收外部生成的事务, 如模块命令、主存储器地址和模块数据。第二个物理接口是存储器接口 154, 该存储器接口 154 支持 ASIC 130 和主存储器 120 之间的通信。尽管存储器接口可以是与模块接口相同的类型, 但在一些实施例中, 存储器接口可以在所使用的信令类型、数据宽度、命令格式或者要求由 ASIC 130 转换的其他方面上有所不同。桥接电路 156 包括选择逻辑 158 和 160, 它们基于使用模块命令所指定的桥接标准, 允许外围接口 140 桥接主存储器 120 与 CPU 105 或者与外围处理器 135 之间的命令、地址和数据的通信。桥接电路 156 还允许外围接口 140 捕获模块数据, 该模块数据包括操作代码 (“操作码”或“外围命令”)、地址、数据和其他用于外围处理器 135 的控制信号。模块接口 152 可以支持与主存储器接口 154 不同数量的并行数据通道, 其中桥接电路 156 可以针对在接口之间传送的存储数据执行串行化 / 反串行化操作。

[0037] 外围接口 140 功能性地驻留在桥接电路 156 和外围处理器 135 之间, 并且包括地址 - 捕获 / 命令解码器 162、操作码端口基地址 (OPBA) 寄存器 164、写数据队列 166 和读数据队列 168、密钥寄存器 170、比较器 172 以及操作码解码器 174。这些元件共同允许外围接口 140 建立操作码开口, 该操作码开口与一个或多个主存储器地址的指定的捕获范围相关



联, CPU 105 可以通过该操作码开口向外围处理器 135 传输操作码和数据。响应于这种操作码, 外围处理器 135 可以通过执行存储在主存储器 120 中或其他地方的指令来支持各种开口功能。在一些实施例中, 外围处理器 135 具有响应于操作码生成外围存储命令、地址和数据的能力。外围接口用来捕获模块数据的主存储器地址的指定的捕获范围表示在模块接口处可接收的存储器地址的子范围, 并且可以位于在用于对主存储器定址的主存储器地址的全范围之外。

[0038] 主存储器 120 可包括多个独立的可访问区块或区块组, 并且设备 125 中的一些设备可以被用于存储纠错码。在这种上下文中, “区块 (rank)” 指代用于单个存储器访问的单独可寻址存储设备的集合。在这种实施例中, 不同的区块或区块组可以支持单独的物理存储器接口, 诸如用于 CPU 105 和外围处理器 135 中的每一个的物理存储器接口。在这种实施例中, 向解码器 162 指定的桥接标准将模块命令、主存储器地址和模块数据从模块接口 152 传递至存储设备的多个区块中的至少一个区块, 并且将外围存储器命令、地址和数据从外围处理器传递至存储设备的多个区块中的至少另一个区块。这样的桥接标准可以是动态地可选的, 以支持从两个处理器对主存储器的顺序的或者同时的访问而不会丢失多个区块中的所选的一个区块中的数据, 或者在将模块命令、主存储器地址和模块数据从模块接口传递至存储器设备的多个区块中的所选的一个区块与将外围存储命令、地址和数据从外围处理器传递至多个区块中的所选的一个区块之间进行改变。

[0039] 图 2 示出了根据一个实施例的数据结构 200, 外围驱动器 146 在高速缓存 142 中汇集并存储该数据结构 200 以将指令和数据传送至 OPBA 命令端口。数据结构 200 将向 OPBA 行指定通过一个或多个写操作被存储在写数据队列 166 中的信息。信息以与写数据队列 166 相同的方式被划分为 256 个逻辑列, 并且每个逻辑列被进一步划分为 8 个 64 位字段: 指令字段、地址字段、写数据字段、写掩码字段以及预留的四个字段。这些字段都唯一地由图 2 中的“四字索引”来指定。术语“四字”代表“四倍长字”, 其中一个字是 16 位。

[0040] 指令字段的八位 OP[7:0] 用于表示将由操作码解码器 174 解码的操作码 (或者被传递至外围处理器 135 用于解码)。另外八位 BL[7:0] 存储突发长度变量, 该突发长度变量可以用于指定突然进入队列 166 的后续写访问的数量。因此, 实施例考虑到高效率的开放页面的高达 256 列的写和读数据突发。可以要求处理器 135 在执行在位 OP[7:0] 中编码的任何指令之前完成指定的突发以避免竞态条件。下一指令位 M[0] 表示是否应用在写掩码字段中指定的数据掩码。位 PSB[2:0] 是可选的外围选择位, 这样说是因为它们指定作为广播命令目标的一个或多个外围处理器。结合图 5 详细描述了使用 PSB 位来允许在多个模块处接收的命令将由它们对应的外围处理器中的任何一个外围处理器或组合执行的实施例。指令字段的剩余位 63:20 是预留的。

[0041] 地址字段允许 CPU 105 向外围处理器 135 指定存储器地址, 诸如以标识主存储器中的位置、模块 110 上的寄存器或者一些其他存储器或功能。这些地址可以映射到 ASIC 130 的存储器层次结构中的任何地方并且以驱动器 146 与 ASIC 130 之间期望的任何方式, 而对 CPU 105 或操作系统的影响很小或没有影响。

[0042] OPBA 列中的写数据字段允许操作码向 OPBA 传输多达 64 位的写数据。OPBA 的剩余 255 个列中的每一列都可以包含多达 512 个附加位的写数据。所有这些写数据都可以作为外围写数据 W<sub>Dp</sub> 被传递至外围处理器 135 用于存储和操作。例如, 这样的数据可能包括

将被存储在主存储器 120 中或其他地方用于由处理器 135 执行的处理器指令。

[0043] 如之前所指出的,数据结构 200 在高速缓存器 142 中被创建并且被写入写数据队列 166。处理器 135 可以操作或者存储这一信息并经由读数据队列 168 将这种操作的结果或者从存储器读取的数据返回至 CPU105。下面指出可能的操作和它们相关联的操作码的非穷尽的列表。

[0044] 图 3 是示出根据一个实施例的图 1 中的存储器模块 110 如何管理来自 CPU 105 的对于访问主存储器和来自外围处理器 135 的支持的 OS 请求的流程图 300。可以使用指向主存储器地址的访问命令来初始化和引导主存储器请求和外围处理器请求两者。

[0045] 在 305 处,外围接口 140 中的解码器 162 经由命令/地址总线 CA 捕获来自存储器控制器 144 的激活命令 ACT。解码器 162 参考寄存器 164 以确定命令指向的主存储器地址是否与存储的 OPBA 行相匹配(判定 310)。如果不匹配,则该命令是传统的主存储器命令。解码器 162 将该命令经由选择逻辑 160 导向主存储器 120,以激活(打开)设备 125 中的被寻址的行(315)。模块 110 随后接收指向打开的行内的列的一个或多个访问命令。在该实施例中,这种访问命令指定 OPBA 的库和区块,但是缺少行地址。如 For 循环 320A/B 所示,主存储器设备 125 解码并执行这些命令(325 和 330)。最后,模块 110 将接收预充电命令(355),并且设备 125 将关闭打开的行(340)来为下一次主存储器访问做准备。

[0046] 返回到判定 310,并且假设接收到的命令指向 OPBA 行,解码器 162 激活用于对与 OPBA 行相对应的区块(rank)和库(bank)地址的后续列访问的陷阱(trap)(345)。列命令通常不明确地标识命令所指向的打开行,但是大多数 DRAM 设备和控制器仅允许在每个区块和库中打开一行。因此,一旦 OPBA 行被“激活”,指向 OPBA 区块和库地址的任何列访问命令都可以被陷阱捕获为使用 WD 队列 166 或者 RD 队列 168 的数据传输。对指向激活的 OPBA 行的 OPBA 行和列命令的激活还可以与由解码器 162 执行的比较并行地被传递至存储器接口 154,以避免用于比较操作的潜在增加的延迟。向存储器接口 154 并行转发 OPBA 行活动可以导致在主存储器设备 125 上的活动(这种活动是不重要的),然而,如果 OPBA 行被选择,则桥接电路 156 将不向主存储器 120 或者从主存储器 120 传递被寻址的数据。解码器 162 还控制选择逻辑 158 以将数据总线 DQ 连接至一对队列 166 和 168,它们分别充当写缓冲器和读缓冲器。

[0047] 队列 166 和 168 中的每个队列存储被划分为 256 个由列地址唯一标识的 64B 分段的一个页面(16KB),例如,对主存储器 120 的页面的分段进行镜像的寻址结构。写队列 166 有效地代替主存储器中的 OPBA 行地址来用于写操作,同时读队列 168 对于读操作进行相同的处理。在该示例中队列 166 和 168 为静态随机存取存储器(SRAM),因此从 DRAM 行要求激活的意义上来说它们不要求“激活”。

[0048] 一旦解码器 162 将行激活识别为指向 OPBA 行并且因此指向“打开的”队列,解码器 162 就等待指定操作和该操作指向的列的访问命令。然后,解码器 162 将每个后续的访问命令解码为 OPBA 行地址解码(For 循环 350A/B 中的 355)。如果在该示例中写访问针对第 0 列、指派的 OPBA,则解码器 162 向操作码解码器 174 发出控制信号“解码”,使得解码器 174 从写数据队列 166 中的第 0 列地址中解码操作码(365),并将经解码的操作码 OpCode 和任何相关联的外围写数据 WOp 传递至外围处理器 135。外围处理器 135 执行操作码 OpCode 以执行一些外围功能(370),并且这样做可以经由桥接电路 156 改变主存储器的内容。

[0049] 针对 OPBA 行的访问命令可以是针对 256 列中的任意列。解码器 162 在每一种情况下对命令信息进行解码并且对队列 166 和 168 执行所命令的功能 (375)。例如, 针对 OPBA 的行地址的任意列的读命令从读队列 168 中的指定的位置子集中读取, 以及针对同一行的任意列的写命令写入写队列 166 中的对应的位置子集。最终, 模块 110 将接收针对指派给 OPBA 的区块和库的预充电命令 (380), 并且解码器 162 将释放列地址陷阱 (385)。此后, 模块 110 等待下一命令。还应该注意, 尽管流程图 300 示出了用于主存储器访问和 OPBA 行访问的单独的路径, 但典型的存储器控制器将交错多个库和 / 或区块的存储器命令, 因此在许多实例中, CPU 105 可以在指向不同区块和 / 或库的连续列访问命令中访问主存储器 120 并与外围接口 140 进行通信。

[0050] 图 4 描绘了根据另一实施例的存储器系统 400。系统 400 包括与图 1 的系统 100 共同的特征, 类似标识的元件是相同或相似的。为了简化而大幅省略了共同特征的讨论。具体地, 在图 4 所示实施例中, 主存储器 120 被划分为两个部分 120a/120b, 它们分别经由存储器接口 154a/154b 被耦合至 ASIC 405。

[0051] 系统 400 包括通过主存储器接口互连的 CPU 105 和存储器模块 410, 该主存储器接口包括命令 / 地址总线 415 和数据总线 420, 它们中的每一个都包括多个并行通道。命令 / 地址总线 415 传送片选、库、行和列 (CS/B/R/C) 地址信号, 以及数据总线 420 传送数据信号 DQ。总线 415 和 420 均被示出为包括将途中的位置切换至模块 410 的信号导线。交叉处示出了在一些系统中做出的信号路由选择, 以简化在支持 CPU 105 的印刷电路板 (例如, 主板) 上的部件布局和线路布线, 或者通过减小线路长度或分支长度来提高信号完整性。对于存储器操作, 假如将切换应用于读和写两者, 通常可以切换数据或地址的一些位的位置而不影响电路性能。一些信号线路因此可以以更加便利或有效的方式来进行布线, 而不引入逻辑错误, 只要存储器模块仅存储和读取。这种习惯做法有时被称为“搅和 (swizzling)”。

[0052] 不影响主存储器访问的搅和的数据或地址线可以仍然干扰去向外围接口 425 的命令。例如, 交换数据位可以改变嵌入到外围处理器 135 的写数据中的操作码。因此, 外围接口 425 包括可编程去搅和电路 430 和相关联的搅和寄存器 435, 其中后者可以被连接至一些串行存在检查 (SPD) 逻辑 440 或其部分。在模块 410 被置于系统 400 中之前, EEPROM 445 或其他存储器利用用于母板的搅和信息、用于寄存器 170 的密钥和其他初始化信息来编程 (EEPROM 还可以在模块第一次被连接时由系统编程, 或者搅和寄存器 435 可以在每次重置之后明确地由系统加载)。然后, 微控制器 450 加载寄存器 435 和 170, 作为例如校准模块和存储器接口的初始化程序的一部分。因此, 去搅和电路 430 响应于来自寄存器 435 的去搅和信号 DeS 通过对在模块接口处并行地接收到的模块数据重新排列来抵消板上搅和。

[0053] 传统的串行接口 455 向控制器 450 提供测试和控制访问。外围接口 425 如上面结合图 1 所描述地来操作, 因此这里省略了详细的处理。EEPROM 445 中的信息包括引导代码, 该引导代码可以由控制器 450 执行, 使得当系统 400 完成上电重置时, 模块接口 152 和存储器接口 154, 以及 480 被校准。这些校准过程可以对于 CPU 105 是透明的。

[0054] 去搅和电路 430 没有被示为影响模块接口 152 与桥接电路 156 之间的存储器流量。在备选的实施例中, 所有存储器流量都被“去搅和”, 使得存储在主存储器中的数据可以被外围处理器以一致的存储格式直接访问。

[0055] 关于接口 425 的右侧, 外围处理器 135 与外围存储器控制器 470 相连, 该外围存储

器控制器 470 以与存储器控制器 144 管理针对 CPU 105 的访问相同的方式管理针对处理器 135 的对于主存储器 120 的访问。

[0056] 在该实施例中,外围处理器 135 具有对于附加的外围本地存储器设备 475 的访问,并且经由本地物理接口 480 和控制器 485 来获得和管理访问。本地控制器 485 可以包括对在由主存储器采用的寻址方案和专用于本地存储器的寻址方案之间的地址转换的支持。外围存储器控制器 470 可以类似地包括对根据处理器 135 的需求的地址转换的支持。

[0057] 例如,外围存储设备 475 可能是高性能的但是相对于主存储器具有较小的可寻址空间以支持改善的外围性能。例如,外围处理器 135 可以执行存储在(多个)外围存储器设备 475 中的外围图形指令。不是图形处理或者除图形处理之外,外围功能可以包括网络支持、数据压缩/解压缩、加密/解密、科学计算等。不同的存储器模块能够支持相同或者不同类型的外围处理,或者仅模块的子集可以包括这种支持。用于外围功能的操作对象可以被提供为外围写数据 W<sub>Dp</sub> 或者从外围写数据 W<sub>Dp</sub> 中检索到。

[0058] 图 5 描绘了根据实施例的支持外围命令广播和可配置的数据宽度的存储器模块 500。存储器模块 500 与图 1 的模块 110 类似,类似标识的元件是相同或相似的。模块 500 与模块 110 的不同之处在于,模块 500 的外围接口 505 的实施例包括外围选择位 (PSB) 解码器 510,该外围选择位 (PSB) 解码器 510 允许外围接口 505 确定广播外围命令是否被指向常驻处理器 135。模块 500 的这一方面在本文参照的相关应用已经被详细说明。模块 500 与模块 110 的不同之处还在于,模块 500 的桥 515 是基于来自串行存在检查 (SPD) 存储器 520 的控制输入可配置为经由数据线 DQ[63:0] 的全部或者子集来传输宽度为 8、16、32 或 64 的数据。支持多种数据宽度的模块,例如可以用于实现动态点对点 (DPP) 存储器架构。简而言之,DPP 架构将点对点信令的性能优势与多点拓扑的灵活性组合到一起。模块 500 以前面详细说明的方式将对 DPP 的支持与完成外围处理的能力组合到一起。使用公知的技术,在启动时初始化 SPD 存储器 520 以选择期望的宽度。

[0059] 在这样的上下文中,命令广播指代相同命令到多个外围处理器的同时传送,该多个外围处理器被实例化在相同或不同模块上。有时期望将工作分割成多个类似的任务并且在单独的外围处理器上执行每一个。例如,在视频处理中,已知为分割帧渲染的技术在多个图形处理器中分配将对单个视频帧执行的渲染处理。处理器在帧的不同子集上并行地工作以加快渲染过程。这种并行处理可以在图形之外的许多领域中具有优势。具有模块 500 的多个实例的存储器系统可以向支持并行处理的多个模块广播相同的命令。在一些实施例中,命令可以指定一个或多个目标处理器,并由此可以被传送至多个外围处理器中的一个外围处理器或任意组合。

[0060] 图 6 示出了数据结构 600,外围驱动器(例如,图 1 的驱动器 146)在高速缓存器中汇集和存储该数据结构 600 以将指令和数据传送至 32 个模块 500 的 OPBA 命令端口,其中在四个存储器通道的每个存储器通道上有 8 个模块。在该示例中,模块是双列直插式模块 (DIMM),但是还可以使用其他模块类型。一些实施例附加地支持 PSB 位字段来以结合图 5 所详细描述的方式支持广播命令。

[0061] 图 7 是混合闪存 /DRAM 存储器模块的示图。在图 7 中,计算机系统 700 包括 CPU/存储器控制器子系统 710、I/O 控制器 730 和存储器模块 720。CPU/存储器控制器子系统 710 包括耦合至存储器控制器 714 的 CPU 712。一个或多个存储器模块 720 耦合至子系统

710 中的存储器控制器 714。每个存储器模块 720 都包括集成电路 (IC) 设备 721、CPU 存储器 723 (又称主存储器) 和非易失性存储器 728。IC 721 包括非易失性存储器控制器 725 和处理器 722。处理器 722 可以是缓冲设备的一部分。处理器 722 可以是微处理器、有限状态机或作为缓冲设备的一部分的其他逻辑电路, 或者包括微处理器、有限状态机或作为缓冲设备的一部分的其他逻辑电路。存储器 723 通常包含 CPU 712 和 / 或 IC 721 使用的指令和 / 或数据。应该理解, CPU 712 可以包括多个处理器核。CPU 712 可以包括堆叠的裸片设备, 其具有例如使用硅穿孔堆叠的一个或多个处理器和 / 或存储器。CPU 712 可以包括和 / 或作为专用处理器, 诸如举例而言, 数字信号处理器、图形处理单元 (GPU)、阵列处理器、存储管理处理器、数据分析处理器 (例如, Hadoop 分布式文件系统处理器或 MapReduce 处理器)、模式识别处理器和 / 或图像处理处理器 (即, 图像处理器)。CPU 712 可以在模块 720 之间划分和协调计算过程和任务。

[0062] IC 721 还可以被称为“运算引擎”、“计算引擎”、“图形处理器”、“渲染引擎”、“处理单元”、“加速器”、“卸载引擎”和 / 或 GPU。IC 721 可以包括和 / 或作为异构处理单元, 该异构处理单元包括 CPU、GPU、视频处理器等中的一个或多个的功能。IC 721 可以包括或者作为串行 -ATA (SATA)、串行连接的 SCSI (SAS)、eSATA、PATA、IEEE1394、USB (所有修订版本)、SCSI 超级 (SCSI Ultra)、光纤通道 (FiberChannel)、无限带宽 (Infiniband)、雷电 (Thunderbolt) 或其他工业标准 I/O 接口 (诸如 PCI 快速——PCIe)。IC 721 可以包括或者作为网络处理器单元 (NPU), 诸如 TCP 卸载引擎 (TOE)、协议转换器 (例如, 通过 SATA 的 TCP、通过 PCIe 的 TCP、加速 SCSI 互连等) 和 / 或协议包转换器。IC 721 可以包括或者作为固定功能图形处理单元、加密 / 解密加速器 / 卸载引擎 (例如, 用于实现 / 加速 SSL、AEC、DEC 等)、压缩器 / 解压器、正则表达式加速器引擎、数字信号处理器 (DSP)、信号路径处理器、傅里叶变换处理器、反傅里叶变换处理器和 / 或媒体格式编码器 / 解码器 (例如, JPEG、DVX、AVI、MP2、MP3、MP4、蓝光、HD-DVD、DVD 等)。还应该理解, 模块 720 可以被耦合至本地 SSD/HDD 和 / 或企业存储类型系统, 诸如外部盘、外部盘阵列、JBOD、RAID 阵列、磁带驱动、光学驱动等。

[0063] 存储器 723 通常包括耦合到一起以形成存储空间块的多个存储设备。存储器 723 可以是或者包括但不限于, SRAM、DDR3、DDR4、DDR5、XDR、XDR2、GDDR3、GDDR4、GDDR5、LPDDR 和 / 或 LPDDR2 以及后继存储器标准和技术。存储器 723 可以是或者包括设备的堆叠, 诸如硅穿孔 (TSV) 堆叠和 / 或混合存储立方体 (HMC)。可从 Hybrid Memory Cube Consortium (<http://hybridmemorycube.org/>) 得到关于 HMC 的进一步的信息。

[0064] 每个 IC 721 都能够执行各种存储器访问和 / 或数据处理功能。对于图 7 所示实施例来说, 存储器控制器 714 还被耦合至 I/O 控制器 730, 该 I/O 控制器 730 控制数据进出系统的流动。可选的视频输入端口 (图 7 中未示出) 可以向存储器控制器 714 提供数据。显示接口 (图 7 中未示出) 可以向一个或多个设备 (诸如显示设备或存储设备) 提供数据输出。对于支持视频输入或捕获能力的系统来说, 存储器控制器 714 上的视频输入端口是处理视频源数据的传送的一种方式。另一种向系统传送视频输入数据的方式将包括通过 I/O 控制器 730 将向存储器控制器 714 传送来自外围模块的数据。

[0065] 在图 7 的示例中, CPU / 存储器控制器子系统 710 被耦合至多个不同的存储器模块 720。每个存储器模块 720 都包括 IC 721 和非易失性存储器 728。IC 721 可以是或者包括

微处理器、有限状态机或作为缓冲设备的一部分的其他逻辑电路。每个 IC 721 都能够执行各种数据处理功能。因此,不同(或相同)存储器模块上的 IC 721 能够同时执行不同的处理功能(即,并行处理)。不同(或相同)存储器模块上的 IC 721 能够执行各种矢量并行处理,其中在(多个)IC 721 之间将功能矢量化并且划分功能。不同(或相同)存储器模块上的 IC 721 能够执行簇并行处理,其中(多个)IC 721 协作以同时(或并行)作用于相同问题。此外,每个 IC 721 能够与其他存储器模块 720 上的其他 IC 721 通信。

[0066] 每个 IC 721 能够在 CPU 712 和 / 或存储器控制器 714 的帮助下与其他存储器模块 720 上的其他 IC 721 通信。模块 720 和 / 或 IC 721 可以是异构的。换句话说,模块 720 和 / 或 IC 721 可以不是全部完全相同的。IC 721 可以包括不是全部完全相同的多个处理器核。例如,IC 721 可以包括 CPU 类型处理核和 GPU 类型处理核的混合。模块 720 和 / 或 IC 721 可以在相同时刻执行不同的功能。模块 720 和 / 或 IC 721 可以由不同的供应商来生产。可以添加、减少以及以即插即用方式互换地使用由不同供应商生产的模块 720 和 / 或 IC 721。模块 720 和 / 或 IC 721 可以类似于虚拟机的操作与运行软件、过程、应用和操作系统的独立(和非独立)附件并行地作用。

[0067] CPU 712 可以控制存储器控制器 714 以将特定的处理任务(诸如图形处理任务、Hadoop 和 / 或 MapReduce 任务)分配给 IC 721,并且自身可以执行某些处理任务。这些任务可以包括将要处理的数据和 / 或将要执行的指令。这些任务可以包括执行应用、驱动、软件层、运行时功能和 / 或操作系统代码中的全部或者部分。尽管在图 7 中示出三个存储器模块 720,但备选的系统可以包含耦合至一个或多个存储器控制器 714 的任何数量的存储器模块。添加和移除存储器模块 720 的能力可以提供可升级和可扩展的存储器和计算架构。此外,应该理解,系统 700 中的模块可以是异构的。换句话说,系统 700 中的存储器模块 720 的集合的一部分可以是混合闪存 / DRAM 类型模块,而其他部分可以是其他类型的模块(例如,完全地 DRAM、完全地存储器和 / 或完全地非易失性存储器)。系统 700 可以支持混合类型的模块。还应该理解,混合闪存 / DRAM 类型模块 720 是示出可以存在经由存储器通道访问的非易失性、大容量(即,比 DRAM 模块大)存储的示例。特定的模块 720 例如可以是全快闪模块,或者甚至可以在模块上包括或不包括 DRAM 的较新的存储类存储器。

[0068] CPU 712 可以通过从与模块 720 相关联的地址开口读取和写入与模块 720 相关联的地址开口来与 IC 721 通信。CPU 712 可以被配置为使用由任何处理器 CPU 712 支持的任何高速缓存策略以从该地址开口(或其部分)读取和写入该地址开口(或其部分)。然而,应该理解,在一个实施例中,最有用的高速缓存策略可以被限制于将被 CPU 712 处理的地址开口(或其部分)配置为不可缓存的存储器(UC)、合并写存储器或写回(WB)。在一个实施例中,这些高速缓存策略可以与某些指令(例如,屏障指令、流负载指令和 / 或流写入指令)的使用组合以实现 CPU 721 和 IC 721 之间最优的(例如,最高的)通信带宽。此外,当地址开口(或其部分)被配置为作为 WC 或 WB 进行处理,则至少最小突发尺寸可用于实现 CPU 721 和 IC 721 之间最优的(例如,最高的)通信带宽。换句话说,小突发尺寸(例如,小于某个、依赖实现的、每突发的列地址的数量)可以导致比 CPU 721 和 IC 721 之间的最优通信带宽小。较大的尺寸(例如,大于某个、依赖实现的、每突发的列地址的数量)可以接近(或近似)CPU 721 和 IC 721 之间的最优(即,最高)通信带宽。应该理解,计算机系统 700 可以利用统一的存储器架构或不统一的存储器架构。计算机系统 700 不限于异构

存储器。此外,还应该理解,计算机系统 700(尤其和 CPU 712)可以利用相干或不相干的高速缓存或缓存策略。

[0069] 还有用的是将地址开口(或其部分)配置为 UC,用于测试、易于实现和/或当 CPU 712 和 IC 721 之间的带宽不是相对高的优先级时。为了测试目的将地址开口(或其部分)配置为 WB 可能是最有用的。在一些实施例中,将地址开口(或其部分)配置为 WC 可以导致 CPU 712 和 IC 721 之间的最大带宽。

[0070] 在一个实施例中,CPU 712 通过从与模块 720 相关联的地址开口读取和写入与模块 720 相关联的地址开口来与 IC 721 通信。该地址开口可以被进一步划分为地址范围(也称端口),该地址范围被配置用于命令、状态和数据的通信。可以配置地址开口的端口并且将其尺寸定为地址(例如,列地址)的块。

[0071] 在一个实施例中,计算机 700 上运行的操作系统可以产生指向块 I/O 和/或存储类型设备的 I/O 命令。这些块 I/O 和/或存储类型设备中的一个或多个可以对应于或具有驻留在存储器模块 720 上的存储数据。响应于用于存储在存储器模块 720 上的数据的块 I/O 和/或存储命令,CPU 712 可通过从与模块 720 相关联的地址开口读取和/或写入与模块 720 相关联的地址开口来将存储器接口命令导向存储器模块 720。

[0072] 为了产生指向块 I/O 或存储类型设备的 I/O 命令,计算机 700 上运行的操作系统内核可以将块设备命令导向通用块层。响应于这些块设备命令,操作系统可以将被导向通用块层的块设备命令转换成向存储器模块 720 发送的存储器接口命令。

[0073] 处理器 722 可以执行块存储 I/O 驱动器的全部或部分,该块存储 I/O 驱动器服务于在非易失性存储器 728 中存储的数据的块 I/O 存储类型设备命令。非易失性存储器控制器 725 可以是或者包括固态硬盘控制器,该固态硬盘控制器向处理器 722 提供块 I/O 存储接口。该块 I/O 存储接口可用于去向和来自非易失性存储器 728 的数据的通信。例如,该块 I/O 存储接口可以使用 SCSI 或 ATA 命令。

[0074] 在一个实施例中,模块 720 可以被配置为使用非易失性存储器 728 服务于对于块 I/O 存储类型设备命令的 RPC 请求。这些 RPC 请求可经由 DRAM 存储器通道被模块 720 接收,该 DRAM 存储器通道操作性地将存储器控制器 714 耦合至模块 720。由模块 720 服务的 RPC 请求可以源自不直接与模块 720 连接的处理器(图 7 中未示出)(例如,网络上的远程处理器)。

[0075] 可以在模块 720 上执行经由 DRAM 存储器通道被模块 720 接收的过程调用和过程数据(例如,通过处理器 722 和/或非易失性存储器控制器 725)。这些过程调用和过程数据可以经由定址到模块 720 的存储器空间(例如,开口)的存储器写事务被传输至模块 720。这些过程调用可以包括 RPC 请求。由模块 720 执行的过程调用和过程数据可以实现一个或多个虚拟文件系统过程调用。可以使用 NFS 请求来访问非易失性存储器 728,该 NFS 请求可以经由定址到模块 720 的存储器空间的存储器写事务被传输至模块 720。可以使用 HDFS 请求来访问非易失性存储器 728,该 HDFS 请求可以经由定址到模块 720 的存储器空间的存储器写事务被传输至模块 720。

[0076] 可以在模块 720 上执行经由 DRAM 存储器通道被模块 720 接收的过程调用。这些过程调用可包括使用非易失性存储器 728 服务的块 I/O 存储类型设备命令。这些过程调用可以包括映射调用,该映射调用指定将由模块对对应的输入数据块执行的专用映射操作。映

射操作还可以产生将使用非易失性存储器 728 来存储的中间数据值。

[0077] 图 8 是示出混合闪存 /DRAM 计算子系统的框图。在图 8 中,子系统 801 包括子系统 800、本地存储器 824 和非易失性存储器 828。子系统 800 包括处理器 812、逻辑 816、存储器控制器 817、CA 输入接口 896、多路复用器 (MUX) 逻辑 819、CA0 接口 886、CA1 接口 887、HS 串行 DQ 接口 878 和非易失性存储器控制器 827。子系统 800 还包括存储器 / 高速缓存器控制器 818。存储器 / 高速缓存器控制器 818 被操作性地耦合至本地存储器 824。本地存储器 824 可以包括高速缓存存储器分区和处理器存储器分区。本地存储器 824 可以通过一个或多个存储器通道被操作性地耦合至子系统 800。应该理解,子系统 801 可以被包括在存储器模块 720 上。因此,能够看出,子系统 800 可在存储器模块 720 中用作 IC 721,并且子系统 801 可以被配置在存储器模块 720 上。

[0078] 在一个实施例中,具有子系统 801 的存储器模块(例如,存储器模块 120)可以具有被配置为与存储器控制器连接的存储器接口。子系统 800 可以是耦合至存储器接口的集成电路设备。子系统 800 包括处理器 812、CA1 接口 887、非易失性存储器控制器 827 和到非易失性存储器 828 的接口。CA1 接口 887 被配置为连接至第一类型的动态存储器集成电路设备。非易失性存储器控制器 827 被耦合至处理器 812。到非易失性存储器 828 的接口被配置为将非易失性存储器控制器 827 耦合至至少一个非易失性存储器集成电路设备。多个第一类型的动态存储器集成电路设备被耦合至 CA1 接口 887。至少一个非易失性存储器集成电路设备经由到非易失性存储器 828 的接口被耦合至子系统 800。

[0079] 子系统 801 还可以包括本地存储器 824。该本地存储器 824 可以包括至少一个第二类型的动态存储器集成电路设备。本地存储器 824 可以经由接口连接至子系统 800。该接口可以经由存储器 / 高速缓存器控制器 818 将本地存储器 824 耦合至处理器 812。本地存储器 824 可用于缓存在被耦合至 CA1 接口 887 的动态存储器集成电路设备中存储的数据。子系统 801 可以包括被耦合至处理器 812 和非易失性存储器 827 的 SSD 控制器。子系统 801 可以包括被耦合至处理器 812 和非易失性存储器控制器 827 的 SATA 控制器。非易失性存储器 828 可以存储处理器 812 可以引导和执行的操作系统。该操作系统可以是与例如 CPU 812 引导和运行的操作系统不同的操作系统。存储器模块可以经由存储器接口接收指令,该指令使得处理器 812 对数据执行操作。子系统 800 可以经由存储器接口返回这些操作的结果。

[0080] 在一个实施例中,具有子系统 801 的存储器模块(例如,存储器模块 120)可以具有存储器接口以连接至存储器通道。该存储器模块可以包括多个动态存储器集成电路。该存储器模块可以具有非易失性存储器 828,该非易失性存储器 828 包括至少一个非易失性存储器集成电路设备。存储器模块可以包括子系统 800。子系统 800 包括处理器 812、CA1 接口 887、NV 控制器 827、CA 输入接口 896 和高速 (HS) 串行 DQ 接口 878。CA1 接口 887 被耦合至多个动态存储器集成电路。NV 控制器 827 经由接口被耦合至非易失性存储器 828。

[0081] 在一个实施例中,子系统 800 从被耦合至 CA 输入接口 896 的存储器控制器接收存储器命令。响应于该存储器命令,子系统 800 在耦合至 CA1 接口 887 的多个动态存储器集成电路和 / 或耦合至 CA0 接口 886 的多个动态存储器集成电路中存储数据。响应于来自于处理器 812 的命令,子系统 800 在非易失性存储器 828 中存储数据。

[0082] 图 9 是混合闪存 /DRAM 存储器模块的示图。在图 9 中,计算机系统 900 包括 CPU/



存储器控制器子系统 910、I/O 控制器 930 和存储器模块 920。CPU/ 存储器控制器子系统 910 包括耦合至存储器控制器 914 的 CPU 912。一个或多个存储器模块 920 被耦合至子系统 910 中的存储器控制器 914。每个存储器模块 920 包括集成电路设备 921 和 CPU 存储器 923。集成电路设备 921 包括非易失性存储器 928 和处理器 922。CPU 存储器 923 通常包含由 CPU 912 使用的指令和 / 或数据。CPU 存储器 923 可以是动态存储器集成电路。集成电路设备 921 可以包括其他功能。因此,应该理解,集成电路设备 921 可以对应于存储器模块 720 上的 IC 721 和 / 或非易失性存储器 728 或子系统 800,并且模块 920 可以是存储器模块 720 的实施例。

[0083] 由于非易失性存储器 928 被包括在集成电路设备 921 中,所以可以由处理器 922 直接访问非易失性存储器 928。集成电路设备 921 可以包括 SATA 接口。处理器 922 和 / 或集成电路设备 921 与集成电路设备 921 上的 SSD 可以经由 SATA 接口通信。模块 920 可以包括用于与模块 920 外部的 SSD/HDD 通信的 SATA 连接器。

[0084] 图 10 是示出混合闪存 /DRAM 计算子系统的框图。图 10 是示出混合闪存 /DRAM 存储器模块计算子系统的框图。在图 10 中,子系统 1001 包括子系统 1000 和本地存储器 1024。子系统 1000 包括处理器 1012、逻辑 1016、存储器控制器 1017、CA 输入接口 1096、多路复用器 (MUX) 逻辑 1019、CA0 接口 1086、CA1 接口 1087、高速 (HS) 串行 DQ 接口 1087 和非易失性存储器 1028。子系统 1000 还包括存储器 / 高速缓存器控制器 1018。存储器 / 高速缓存器控制器 1018 被操作性地耦合至本地存储器 1024。本地存储器 1024 可以通过一个或多个存储器通道被操作性地耦合至子系统 1000。应该理解,子系统 1001 可以被包括在存储器模块 920 上。因此,能够看出,子系统 1000 可以用作存储器模块 720 中的 IC 721 或者用作存储器模块 920 中的 IC 921,并且子系统 1001 可以被配置在存储器模块 720 和 / 或 920 上。

[0085] 在一个实施例中,具有子系统 1001 的存储器模块 (例如,存储器模块 120 和 / 或 320) 可以具有被配置为与存储器控制器连接的存储器接口。子系统 1000 可以是耦合至存储器接口的集成电路设备。子系统 1000 包括处理器 1012、CA1 接口 1087 和非易失性存储器 1028。CA1 接口 1087 被配置为连接至第一类型的动态存储器集成电路设备。非易失性存储器 1028 被耦合至处理器 1012。多个第一类型的动态存储器集成电路设备被耦合至 CA1 接口 1087。

[0086] 子系统 1001 还可以包括本地存储器 1024。该本地存储器 1024 可以包括至少一个第二类型的动态存储器集成电路设备。本地存储器 1024 可以经由接口连接至子系统 1000。该接口可以经由存储器 / 高速缓存器控制器 1018 将本地存储器 1024 耦合至处理器 1012。本地存储器 1024 可以用于缓存存储在耦合至 CA1 接口 1087 的动态存储器集成电路设备中的数据。非易失性存储器 1028 可以存储处理器 1012 可以引导和执行的操作系统。存储器模块可以经由存储器接口接收指令,该指令使得处理器 1012 对数据执行操作。子系统 1000 可以经由存储器接口返回这些操作的结果。

[0087] 在一个实施例中,具有子系统 1001 的存储器模块 (例如,存储器模块 120 和 / 或 320) 可以具有连接至存储器通道的存储器接口。该存储器模块可以包括多个动态存储器集成电路。该存储器模块可以包括子系统 1000。子系统 1000 包括处理器 1012、CA1 接口 1087、CA 输入接口 1096、高速 (HS) 串行 DQ 接口 1078 和非易失性存储器 1028。CA1 接口 1087 被耦合至多个动态存储器集成电路。

[0088] 在一个实施例中,子系统 1000 从耦合至 CA 输入接口 1096 的存储器控制器接收存储器命令。响应于该存储器命令,子系统 1000 在耦合至 CA1 接口 1087 的多个动态存储器集成电路和 / 或耦合至 CA0 接口 1086 的多个动态存储器集成电路中存储数据。响应于来自处理器 1012 的命令,子系统 1000 在非易失性存储器 1028 中存储数据。

[0089] 图 11 是混合盘类型非易失性存储和 DRAM 存储器模块的示意图。图 11 是混合盘类型非易失性存储和 DRAM 存储器模块的示意图。在图 11 中,计算机系统 1100 包括 CPU/ 存储器控制器子系统 1110、I/O 控制器 1130 和存储器模块 1120。CPU/ 存储器控制器子系统 1110 包括耦合至存储器控制器 1114 的 CPU 1112。一个或多个存储器模块 1120 被耦合至子系统 1110 中的存储器控制器 1114。每个存储器模块 1120 都包括集成电路设备 1121、CPU 存储器 1123 和盘型存储 1128。集成电路设备 1121 包括盘接口 1125 和处理器 1122。CPU 存储器 1123 通常包含由 CPU 1112 使用的指令和 / 或数据。盘型存储 1128 通常包含由 CPU 1112 和 / 或处理器 1122 使用的指令和 / 或数据。CPU 存储器 1123 可以是动态存储器集成电路。集成电路设备 1121 可以包括其他功能。因此,应该理解,集成电路设备 1121 可以对应于存储器模块 720 中的 IC 721、存储器模块 920 中的 IC 921 或子系统 800 和 / 或 900, 并且模块 1120 可以是存储器模块 720 和 / 或 920 的实施例。

[0090] 在图 11 中示出的模块 1120 包括盘型存储 1128。盘型存储例如可以是或者包括但不限于,固态盘驱动、被控制并组织为盘型大容量存储设备(例如,USB 存储棒)的闪存或者小形状因子的磁盘驱动(例如,1.8" 或者更小的硬盘驱动)。

[0091] 盘接口 1125 管理 / 控制盘型存储 1128。因此,模块 1120 可以具有 DRAM 和盘型存储器 1128 的组合。盘接口 1125 可以包括(或者可以是)SSD 控制器(具有或不具有附加的处理器)。模块 1120 上的盘接口 1125 可以经由存储器通道被 CPU 1112 访问,该存储器通道通常具有比 I/O 总线更高的带宽。

[0092] 盘型存储 1128 可以被处理器 1122 直接访问。换句话说,盘型存储 1128 可以被处理器 1122 访问而不通过存储器控制器 1114 或依赖于 CPU1112 来传递数据 / 地址。盘接口 1125 可以是或者包括 SATA 接口。处理器 1122 和 / 或集成电路设备 1121 和盘型存储 1128 可以经由 SATA 接口通信。模块 1120 可以包括用于与模块 1120 外部的 SSD/HDD 通信的 SATA 连接器。

[0093] 图 12 是示出混合盘类型非易失性存储器和 DRAM 存储器模块计算子系统的框图。在图 12 中,子系统 1201 可以包括子系统 1200、可选的本地存储器 1224 和盘型存储 1228。子系统 1200 包括逻辑 1216、存储器控制器 1217、CA 输入接口 1296、多路复用器 (MUX) 逻辑 1219、CA0 接口 1286、CA1 接口 1287、HS 串行 DQ 接口 1278 和盘接口 1227。子系统 1200 可以可选地包括处理器 1212 和存储器 / 高速缓存器控制器 1218。如果被包括,则存储器 / 高速缓存器控制器 1218 被操作性地耦合至本地存储器 1224(如果被包括)。本地存储器 1224 可以通过一个或多个存储器通道被操作性地耦合至子系统 1200。可以通过一个或多个存储器通道访问本地存储器 1224。应该理解,子系统 1201 可以被包括在存储器模块 1120 上。因此,能够看出,子系统 1200 可以被用作存储器模块 720 中的 IC 721、存储器模块 921 中的 IC 921 和 / 或存储器模块 1120 中的 IC 1121,并且子系统 1201 可以被配置在存储器模块 720、920 和 / 或 1120 上。

[0094] 在一个实施例中,具有子系统 1201 的存储器模块(例如,存储器模块 720、920 和 /

或 1120) 可以具有被配置为与存储器控制器连接的存储器接口。子系统 1200 可以是耦合至存储器接口的集成电路设备。子系统 1200 包括 CA1 接口 1287 和盘接口 1227。CA1 接口 1287 被配置为连接至第一类型的动态存储器集成电路设备。盘接口 1227 可以被耦合至处理器 1212 (如果存在)。多个第一类型的动态存储器集成电路设备被耦合至 CA1 接口 1287。至少一个盘型存储设备 1228 经由盘接口 1227 被耦合至子系统 1201。

[0095] 子系统 1201 还可以包括本地存储器 1224。该本地存储器 1224 可以包括至少一个第二类型的动态存储器集成电路设备。本地存储器 1224 可以经由接口连接至子系统 1200。该接口可以经由存储器 / 高速缓存器控制器 1218 (如果存在) 将本地存储器 1224 (如果存在) 耦合至处理器 1212 (如果存在)。本地存储器 1224 可以用于缓存存储在耦合至 CA1 接口 1287 的动态存储器集成电路设备中的数据。盘型存储 1228 可以存储处理器 1212 (如果存在) 可以引导和执行的操作系统。存储器模块可以经由存储器接口接收指令, 该指令使得处理器 1212 (如果存在) 对数据执行操作。子系统 1200 可以经由存储器接口返回这些操作的结果。

[0096] 在一个实施例中, 具有子系统 1201 的存储器模块 (例如, 存储器模块 720、920 和 / 或 1120) 可以具有存储器接口以连接至存储器通道。该存储器模块可以包括多个动态存储器集成电路。存储器模块可以具有盘型存储 1228。存储器模块可以包括子系统 1200。子系统 1200 包括 CA1 接口 1287、盘接口 1227、CA 输入接口 1296 和高速 (HS) 串行 DQ 接口 1278。CA1 接口 1287 被耦合至多个动态存储器集成电路。盘接口 1227 经由接口被耦合至盘型存储 1228。

[0097] 在一个实施例中, 子系统 1200 从耦合至 CA 输入接口 1296 的存储器控制器接收存储器命令。响应于该存储器命令, 子系统 1200 在耦合至 CA1 接口 1287 的多个动态存储器集成电路和 / 或耦合至 CA0 接口 1286 的多个动态存储器集成电路中存储数据。响应于处理器 1212 的命令, 子系统 1200 可以在盘型存储 1228 中存储数据。

[0098] 图 13 是用于访问存储在 DRAM 存储器通道上的非易失性存储器中的文件数据的系统的示图。在图 13 中, 系统 1300 被示为包括软件层和硬件层。具体地, 系统 1300 包括虚拟文件系统 (VFS) 1302、高速缓存器 1304、映射层 1310、通用块层 1330、I/O 调度层 1340、块设备驱动器 1351、块设备驱动器 1352、模块块设备驱动器 1360、块存储设备 1370 和模块 1320。映射层 1310 包括文件系统 1311、文件系统 1312 和块设备文件 1313。模块 1320 包括非易失性存储器 1328。适合用作模块 1320 的模块包括模块 720、模块 920 和 / 或模块 1120。

[0099] 虚拟文件系统 1302 被操作性地耦合至高速缓存器 1304 和映射层 1310。高速缓存器 1304 被操作性地耦合至映射层 1310。映射层 1310 被操作性地耦合至通用块层 1330。通用块层 1330 被操作性地耦合至 I/O 调度层 1340。I/O 调度层被操作性地耦合至块设备驱动器 1351、块设备驱动器 1352 和模块块设备驱动器 1360。块设备驱动器 1351 被操作性地耦合至块存储设备 1370。模块块设备驱动器被操作性地耦合至模块 1320, 并且因此还被操作性地耦合至非易失性存储器 1328 以从非易失性存储器 1328 中读取数据 / 向非易失性存储器 1328 写入数据。

[0100] 在一个实施例中, 应用或操作系统可以访问存储在非易失性存储器 1328 中的数据, 如同数据被存储在盘驱动上的文件中。读 (或写) 服务程序调用激活 VFS 1302 功能,

该 VFS 1302 功能被传递了文件描述符和到文件内位置的偏移。VFS 1302 功能确定所请求的数据是否已经可用——例如在高速缓存器 1304 中。如果数据已经可用,则 VFS 1302 可以完成读(或写)操作。如果数据还不可用,则 VFS 1302 功能确定如何执行操作(例如,从块设备中读数据/向块设备写数据)。

[0101] 映射层 1310 确定文件系统(例如,文件系统 1311、文件系统 1312)的块大小,并根据文件块号计算目标数据的范围。映射层 1310 调用访问文件的盘索引节点的文件系统专用函数,并且根据逻辑块号确定所请求数据的位置。该位置可以被传递至通用块层 1330。

[0102] 如果该操作是针对原始块设备文件,则映射层 1310 不调用文件系统专用方法。相反,映射层 1310 可以将块设备文件中的偏移转换为存储或存储分区中的对应的位置。该位置可以被传递至通用块层 1330。

[0103] 内核使用通用块层 1330 来在块设备上发出操作。通用块层 1330 抽取内核的作为块设备(即,块存储设备 1370 和模块 1320)访问的存储的视图。通用块层 1330 向 I/O 调度层 1340 通知未决的 I/O 数据事务。I/O 调度层 1340 根据内核策略分类并向适当的块设备驱动器 1351-1352 和/或模块块设备驱动器 1360 发出未决的 I/O 数据事务。

[0104] 块设备驱动器 1351-1352 向块存储设备(例如,块存储设备 1370)发送适当的块 I/O 存储类型命令。模块块设备驱动器 1360 可以将从 I/O 调度层 1340 接收的块设备命令转换为向存储器模块 1320 发送的存储器接口命令。

[0105] 模块 1320 上的处理器可以执行模块块设备驱动器 1360 的所有或部分来服务于在非易失性存储器 1328 中存储的数据的服务 I/O 存储类型设备命令。块 I/O 存储类型设备命令可以包括例如 SCSI 或 ATA 命令。

[0106] 图 14 是用于访问存储在 DRAM 存储器通道上的非易失性存储器中的文件数据的分布式系统的示意图。在图 14 中,分布式文件系统 1400 包括客户端计算机 1410、服务器计算机 1430 和网络 1450。客户端计算机 1410 包括系统调用层 1411、虚拟文件系统层 1412、本地文件系统 1413、网络文件系统客户端 1414、块存储设备 1416、远程过程调用客户端存根 1415。服务器计算机 1430 包括系统调用层 1431、虚拟文件系统层 1432、本地文件系统 1433、网络文件系统服务器 1434、远程过程调用服务器存根 1435 和模块 1420。模块 1420 包括处理器 1422 和非易失性存储器 1428。应该理解,模块 1420 可以是、包括、对应于一个或多个模块 720、920 或 1120 或者作为其实施例,或者包括本文所描述的那些模块(例如,IC 721、IC 921、IC 1121、子系统 800、子系统 1000 和/或子系统 1200)中的一个或多个元件。

[0107] 客户端计算机 1410 的系统调用层 1411 被操作性地耦合至客户端计算机 1410 的 VFS 层 1412。VFS 层 1412 被操作性地耦合至客户端计算机 1410 的本地文件系统 1413 和 NFS 客户端 1414。本地文件系统 1413 被操作性地耦合至客户端计算机 1410 的块存储设备 1416。NFS 客户端 1414 被操作性地耦合至 RPC 客户端存根 1415。RPC 客户端存根 1415 经由网络 1450 被操作性地耦合至服务器计算机 1430 的 RPC 服务器存根 1435。

[0108] 服务器计算机 1430 的系统调用层 1431 被操作性地耦合至服务器计算机 1430 的 VFS 层 1412。服务器计算机 1430 的 RPC 服务器存根 1435 被操作性地耦合至 NFS 服务器 1434。NFS 服务器 1434 被操作性地耦合至 VFS 层 1432。VFS 层 1432 被操作性地耦合至服务器计算机 1430 的本地文件系统 1433。服务器计算机 1430 的本地文件系统 1433 被操作性地耦合至模块 1420。因此,应该理解,模块 1420 的处理器 1422 和/或非易失性存储器

1428 能够经由本地文件系统 1433、VFS 层 1432、NFS 服务器 1434、RPC 服务器存根 1435 和网络 1450 被耦合至客户端计算机 1410 的元件。具体地,模块 1420 的处理器 1422 和 / 或非易失性存储器 1428 可以经由网络 1450 被操作性地耦合至客户端计算机 1410 的 VFS 层 1412 和系统调用层 1411。因此,对读和 / 或写由系统调用层接收的数据的请求可以由服务器计算机 1430 的模块 1420 上的非易失性存储器 1428 来服务。

[0109] 图 15 是用于使用块 I/O 存储类型设备命令在 DRAM 存储器通道上远程地访问存储在非易失性存储器中的文件数据的系统的示图。在图 15 中,分布式文件系统 1500 被示为包括服务器计算机 1530 和网络 1550。在图 15 中没有示出的是客户端计算机,该客户端计算机将经由网络 1550 操作性地耦合至服务器计算机 1530。

[0110] 服务器计算机 1530 包括系统调用层 1531、虚拟文件系统层 1532、本地文件系统 1533、网络文件系统服务器 1534、远程过程调用服务器存根 1535、块设备驱动器 1560 和模块 1520。模块 1520 包括处理器 1522 和非易失性存储器 1528。应该理解,模块 1520 可以是、包括、对应于一个或多个模块 720、920 或 1120 或者作为其实施例,或者包括本文描述的那些模块(例如,IC 721、IC 921、IC 1121、子系统 800、子系统 1000 和 / 或子系统 1200)中的一个或多个元件。

[0111] 服务器计算机 1530 的系统调用层 1531 被操作性地耦合至 VFS 层 1532。服务器计算机 1530 的 RPC 服务器存根 1535 被操作性地耦合至 NFS 服务器 1534。NFS 服务器 1534 被操作性地耦合至 VFS 层 1532。VFS 层 1532 被操作性地耦合至本地文件系统 1533。本地文件系统 1533 被操作性地耦合至块设备驱动器 1560。块设备驱动器 1560 被操作性地耦合至模块 1520。因此,应该理解,模块 1520 的处理器 1522 和 / 或非易失性存储器 1528 能够经由块设备驱动器 1560、本地文件系统 1533、VFS 层 1532、NFS 服务器 1534、RPC 服务器存根 1535 和网络 1550 被操作性地耦合至客户端计算机的元件。

[0112] 在一个实施例中,本地文件系统 1533 通过从与模块 1520 相关联的地址开口读取和写入与模块 1520 相关联的地址开口来与处理器 1522 和 / 或非易失性存储器 1528 通信。该地址开口可以进一步被划分为地址范围(也称端口),它们被配置用于命令、状态和数据的通信。可以配置地址开口的端口并且将其尺寸定为地址(例如,列地址)的块。

[0113] 在一个实施例中,本地文件系统 1533 可以产生指向块设备驱动器 1560 的 I/O 命令。响应于用于存储在存储器模块 1520 的非易失性存储器 1528 中的数据的数据的 I/O 命令,块设备驱动器 1560 可以通过从与模块 1520 相关联的地址开口读取和 / 或写入与模块 1520 相关联的地址开口来将存储器接口命令导向存储器模块 1520。块设备驱动器 1560 可以将块设备驱动器 1560 接收的 I/O 命令转换为向存储器模块 1520 发送的存储器接口命令。

[0114] 图 16 是用于服务对于存储在 DRAM 存储器通道上的非易失性存储器中的数据的数据的远程请求的系统的示图。在图 16 中,分布式文件系统 1600 被示出为包括服务器计算机 1630 和网络 1650。在图 16 中没有示出的是客户端计算机,该客户端计算机可以经由网络 1650 操作性地耦合至服务器计算机 1630。

[0115] 服务器计算机 1630 包括系统调用层 1631、虚拟文件系统层 1632、本地文件系统 1633、网络文件系统服务器 1634、远程过程调用服务器存根 1635 和模块 1620。模块 1620 包括块设备驱动器 1660、处理器 1622 和非易失性存储器 1628。因此,应该明确,除了块设备驱动器的位置,服务器计算机 1630 与服务器计算机 1530 类似。在服务器计算机 1630 中,

块设备驱动器 1660 的全部或部分驻留在模块 1620 上或在模块 1620 上执行。因此,模块 1620 可以使用较高级的抽象经由 DRAM 存储器通道与本地文件系统 1633 交互。此外,服务请求所需要的一些处理由模块 1602 来处理,因此释放服务器计算机 1630 上的资源。还应该理解,模块 1620 可以是、包括、对应于一个或多个模块 720、920 或 1120 或者作为其实施例,或者包括本文所述这些模块(例如,IC 721、IC 921、IC 1121、子系统 800、子系统 1000 和 / 或子系统 1200) 的一个或多个元件。

[0116] 服务器计算机 1630 的系统调用层 1631 被操作性地耦合至 VFS 层 1632。服务器计算机 1630 的 RPC 服务器存根 1635 被操作性地耦合至 NFS 服务器 1634。NFS 服务器 1634 被操作性地耦合至 VFS 层 1632。VFS 层 1632 被操作性地耦合至本地文件系统 1633。本地文件系统 1633 被操作性地耦合至模块 1620,因此也被操作性地耦合至块设备驱动器 1660。因此,应该理解,模块 1620 的处理器 1622 和 / 或非易失性存储器 1628 能够经由块设备驱动器 1660、本地文件系统 1633、VFS 层 1632、NFS 服务器 1634、RPC 服务器根村 1635 和网络 1650 被操作性地耦合至客户端计算机的元件。

[0117] 在一个实施例中,具体地,本地文件系统 1633 通过从与模块 1620 和 / 或块设备驱动器 1660 相关联的地址开口读取和写入与模块 1620 和 / 或块设备驱动器 1660 相关联的地址开口来与模块 1620 的块设备驱动器 1660 通信。该地址开口可以进一步被划分为被配置用于命令、状态和数据的通信的地址范围(也称端口)。可以配置地址开口的端口并且将其尺寸定为地址(例如,列地址)的块。

[0118] 在一个实施例中,本地文件系统 1633 产生指向块设备驱动器 1660 的 I/O 命令。具体地,响应于用于在存储器模块 1620 的非易失性存储器 1628 中存储的数据的 I/O 命令,存储器接口命令通过从与模块 1620 和 / 或块设备驱动器 1660 相关联的地址开口读取和 / 或写入与模块 1620 和 / 或块设备驱动器 1660 相关联的地址开口被导向存储器模块 1620。块设备驱动器 1660 可以将块设备驱动器 1660 接收的 I/O 命令转换为向处理器 1622 和 / 或非易失性存储器 1628 发送的命令。

[0119] 图 17 是用于服务向 DRAM 存储器通道上的固态盘的远程请求的系统的示图。在图 17 中,所示分布式文件系统 1700 包括服务器计算机 1730 和网络 1750。图 17 中没有示出的是客户端计算机,该客户端计算机将经由网络 1750 操作性地耦合至服务器计算机 1730。

[0120] 服务器计算机 1730 包括系统调用层 1731、虚拟文件系统层 1732、本地文件系统 1733、网络文件系统服务器 1734、远程过程调用服务器存根 1735 和模块 1720。模块 1720 包括处理器 1722、固态硬盘(SSD)控制器 1725 和非易失性存储器 1728。因此,应该明确,服务器计算机 1730 与服务器计算机 1430 和服务器计算机 530 类似,除了模块 1720 明确包括 SSD 控制器 1725。SSD 控制器 1725 向处理器 1722 提供块 I/O 存储接口。该块 I/O 存储接口可以用于去向 / 来自非易失性存储器 1728 的数据的通信。该块 I/O 存储接口可以使用例如 SCSI 或 ATA 命令。处理器 1722 可以接收来自本地文件系统 1733 的命令并使用 SSD 控制器 1725 和非易失性存储器 1728 来完成这些命令。如本文所描述的,还应该理解,模块 1720 可以是、包括、对应于一个或多个模块 720、920 或 1120 或作为其实施例,或者包括本文所述这些模块(例如,IC 721、IC921、IC 1121、子系统 800、子系统 1000 和 / 或子系统 1200) 的一个或多个元件。

[0121] 服务器计算机 1730 的系统调用层 1731 被操作性地耦合至 VFS 层 1732。服务器计

算机 1730 的 RPC 服务器存根 1735 被操作性地耦合至 NFS 服务器 1734。NFS 服务器 1734 被操作性地耦合至 VFS 层 1732。VFS 层 1732 被操作性地耦合至本地文件系统 1733。本地文件系统 1733 被操作性地耦合至模块 1720，因此也被操作性地耦合至处理器 1722、SSD 控制器 1725 和 / 或非易失性存储器 1728。因此，应该理解，模块 1720 的处理器 1722、SSD 控制器 1725 和 / 或非易失性存储器 1728 能够经由处理器 1722、本地文件系统 1733、VFS 层 1732、NFS 服务器 1734、RPC 服务器存根 1735 和网络 1750 被操作性地耦合至客户端计算机的元件。

[0122] 在一个实施例中，具体地，本地文件系统 1733 通过从与模块 1720 和 / 或 SSD 的控制器 1725 相关联的地址开口读取和写入与模块 1720 和 / 或 SSD 的控制器 1725 相关联的地址开口来与模块 1720 的 SSD 控制器 1725 通信。该地址开口可以进一步被划分为配置用于命令、状态和数据的通信的地址范围（也称端口）。可以配置地址开口的端口并且将其尺寸定为地址（例如，列地址）的块。

[0123] 在一个实施例中，本地文件系统 1733 产生指向 SSD 控制器 1725 的 I/O 命令。具体地，响应于用于在存储器模块 1720 的非易失性存储器 1728 中存储的 I/O 命令，存储器接口命令通过从与模块 1720 和 / 或 SSD 控制器 1725 相关联的地址开口读取和 / 或写入与模块 1720 和 / 或 SSD 控制器 1725 相关联的地址开口被导向存储器模块 1720。处理器 1722 可以将模块 1720 接收的 I/O 命令转换为向 SSD 控制器 1725 和 / 或非易失性存储器 1728 发送的命令。

[0124] 图 18 是用于分布式文件处理的系统的示图。在图 18 中，所示分布式文件处理系统 1800 包括服务器计算机 1830 和网络 1850。图 18 中没有示出的是将经由网络 1850 操作性地耦合至服务器计算机 1830 的客户端节点、主节点或其他映射处理器计算机。

[0125] 服务器计算机 1830 包括系统调用层 1831、虚拟文件系统层 1832、本地文件系统 1833、分布式文件系统服务器 1834、远程过程调用服务器存根 1835 和模块 1820。模块 1820 包括映射功能 1829、处理器 1822 和非易失性存储器 1828。在服务器计算机 1830 中，分布式文件系统的被优化用于映射功能的全部或部分（例如，‘MapReduce’——用于利用集群上的并行的、分布式算法来处理大数据集合）驻留在模块 1820 上并且在模块 1820 上执行。还应该理解，模块 1820 可以是、包括、对应于一个或多个模块 720、920 或 1120 或者作为其实施例，或者包括本文所述这些模块（例如，IC 721、IC 921、IC 1121、子系统 800、子系统 1000 和 / 或子系统 1200）中的一个或多个元件。

[0126] 服务器计算机 1830 的系统调用层 1831 被操作性地耦合至 VFS 层 1832。服务器计算机 1830 的 RPC 服务器存根 1835 被操作性地耦合至 DFS 服务器 1834。DFS 服务器 1834 被操作性地耦合至 VFS 层 1832。VFS 层 1832 被操作性地耦合至本地文件系统 1833。本地文件系统 1833 被操作性地耦合至模块 1820，因此也被操作性地耦合至映射功能 1829。映射功能 1829 可以使用由非易失性存储器 1828 存储的数据在处理器 1822 上执行。因此，应该理解，由映射功能 1829 执行的映射函数和 / 或由模块 1820 的非易失性存储器 1828 存储的分布式文件系统数据能够经由本地文件系统 1833、VFS 层 1832、DFS 服务器 1834、RPC 服务器存根 1835 和网络 1850 被操作性地耦合至客户端计算机或主节点的元件。

[0127] 在一个实施例中，具体地，本地文件系统 1833 通过从与模块 1820 和 / 或映射功能 1829 相关联的地址开口读取和写入与模块 1820 和 / 或映射功能 1829 相关联的地址开口来

与模块 1820 的映射功能 1829 通信。该地址开口可以被进一步划分为被配置用于命令、状态和数据的通信的地址范围（也称端口）。可以配置地址开口的端口并且将其尺寸定为地址（例如，列地址）的块。

[0128] 可以在模块 1820 上执行经由 DRAM 存储器通道被模块 1820 接收到的用于映射功能 1829 的过程调用和过程数据（例如，通过处理器 1822 使用由非易失性存储器 1828 存储的数据）。这些过程调用和过程数据可以经由定址到模块 1820 的存储空间（例如，开口）的存储器写事务被传输至模块 1820。这些过程调用可以包括 RPC 请求。由模块 1820 执行的过程调用和过程数据还可以实现一个或多个虚拟文件系统过程调用。这些过程调用可以包括映射调用，该映射调用指定将由模块 1829 对使用非易失性存储器 1828 存储的对应的输入数据块执行。映射操作还可以产生将使用非易失性存储器 1828 或模块 1820 上的其他存储器存储的中间数据值。

[0129] 图 19 是示出操作存储器模块的方法的流程图。图 19 所示步骤可以由系统 100、系统 400、模块 500、系统 700、子系统 800、子系统 801、系统 900、子系统 1000、子系统 1001、系统 1100、子系统 1200、子系统 1201、系统 1300、系统 1400、系统 1500、系统 1600、系统 1700、系统 1800 和 / 或它们的部件中的一个或多个元件来执行。在集成电路设备处，接收了来自与存储器模块的存储器接口连接的存储器控制器的存储器命令；集成电路设备具有处理器、第一接口、第二接口和非易失性存储器。第一接口被连接至多个动态存储器集成电路。第二接口接收了来自存储器控制器的存储器命令（1902）。例如，集成电路设备 921 可以从存储器控制器 914（其被耦合至模块 920 的存储器接口）接收存储器命令。例如，集成电路设备 921 可以是系统 1000。系统 1000 具有可以连接至 CPU 存储器 923 的 CA0 接口 1086 和 CA1 接口 1087。系统 1000 具有 NV 存储器 1028（其可以对应于 NV 存储器 928）。系统 1000 具有 CA 输入接口 1096，该 CA 输入接口 1096 可以接收来自存储器控制器 914 的存储器命令。

[0130] 响应于存储器命令，数据被存储在多个动态存储器集成电路中（1904）。例如，响应于经由 CA 输入接口 1096 从存储器控制器 914 接收的命令，系统 1000 可以将数据存储在连接至 CA0 接口 1086 或 CA1 接口 1087 的动态存储器集成电路（例如，CPU 存储器 923）中。

[0131] 响应于来自处理器的命令，数据被存储在至少一个非易失性存储器集成电路设备中（1906）。例如，系统 1000 可以响应于处理器 1012（或处理器 922）并且在 NV 存储器 1028 中存储在 CPU 存储器 923、本地存储器 1024 和 / 或连接至 CA0 接口 1086 和 CA1 接口 1087 的动态存储器中所存储的数据。

[0132] 图 20 是示出操作存储器模块的方法的流程图。图 20 所示步骤可以由系统 100、系统 400、模块 500、系统 700、子系统 800、子系统 801、系统 900、子系统 1000、子系统 1001、系统 1100、子系统 1200、子系统 1201、系统 1300、系统 1400、系统 1500、系统 1600、系统 1700、系统 1800 和 / 或它们的部件中的一个或多个元件来执行。在集成电路设备处，接收了来自与存储器模块的存储器接口连接的存储器控制器的存储器命令；集成电路设备具有固态硬盘控制器电路、第一接口、第二接口和第三接口。第一接口被连接至多个动态存储器集成电路。第二接口被连接至至少一个非易失性存储器集成电路设备。第三接口接收了来自存储器控制器的存储器命令（2002）。例如，集成电路设备 721 可以从存储器控制器 714（其被耦合至模块 720 的存储器接口）接收存储器命令。例如，集成电路设备 721 可以是系统 800。



系统 800 具有可以连接至 CPU 存储器 723 的 CA0 接口 886 和 CA1 接口 887。系统 800 具有 NV 控制器 827 (其可以为固态硬盘控制器电路并且可以对应于 NV 控制器 725) 以向 NV 存储器 828 (其可以对应于 NV 存储器 728) 接口。系统 800 具有 CA 输入接口 896, 该 CA 输入接口 896 可以接收来自存储器控制器 714 的存储器命令。

[0133] 响应于存储器命令, 数据被存储在多个动态存储器集成电路中 (2004)。例如, 响应于经由 CA 输入接口 896 从存储器控制器 714 接收的命令, 系统 800 可以将数据存储在连接至 CA0 接口 886 或 CA1 接口 887 的动态存储器集成电路 (例如, CPU 存储器 723) 中。

[0134] 响应于来自固态硬盘控制器电路的命令, 数据被存储在至少一个非易失性存储器集成电路设备中 (2006)。例如, 系统 800 可以响应于固态硬盘控制器电路并且在 NV 存储器 828 中存储在 CPU 存储器 723、本地存储器 824 和 / 或连接至 CA0 接口 886 和 CA1 接口 887 的动态存储器中所存储的数据, 该固态硬盘控制器电路是 NV 控制器 827 或者被包括在 NV 控制器 827 中。

[0135] 图 21 是示出操作存储器模块的方法的流程图。图 21 所示步骤可以由系统 100、系统 400、模块 500、系统 700、子系统 800、子系统 801、系统 900、子系统 1000、子系统 1001、系统 1100、子系统 1200、子系统 1201、系统 1300、系统 1400、系统 1500、系统 1600、系统 1700、系统 1800 和 / 或它们的部件中的一个或多个元件来执行。在集成电路设备处, 接收了来自与存储器模块的存储器接口连接的存储器控制器的存储器命令; 集成电路设备具有处理器、第一接口、第二接口和第三接口。第一接口被连接至多个动态存储器集成电路。第二接口被连接至至少一个盘型大容量存储设备。第三接口接收了来自存储器控制器的存储器命令 (2102)。例如, 集成电路设备 1121 可以从存储器控制器 1114 (其被耦合至模块 1120 的存储器接口) 接收存储器命令。例如, 集成电路设备 1121 可以是系统 1200。系统 1200 具有可以连接至 CPU 存储器 1123 的 CA0 接口 1286 和 CA1 接口 1287。系统 1200 具有盘接口 1227 (其可以对应于盘接口 1125) 以向盘型存储 1228 (其可以对应于盘型存储 1228) 接口。系统 1200 具有 CA 输入接口 1296, 该 CA 输入接口 1296 可以接收来自存储器控制器 1114 的存储器命令。

[0136] 响应于存储器命令, 数据被存储在多个动态存储器集成电路中 (2104)。例如, 响应于经由 CA 输入接口 1296 从存储器控制器 1114 接收的命令, 系统 1200 可以将数据存储在连接至 CA0 接口 1286 或 CA1 接口 1287 的动态存储器集成电路 (例如, CPU 存储器 1123) 中。

[0137] 响应于来自处理器的命令, 数据被存储在至少一个盘型大容量存储设备中 (2106)。例如, 系统 1200 可以响应于处理器 1212 (或处理器 1122) 并且在盘型存储器 1228 中存储在 CPU 存储器 1123、本地存储器 1224 和 / 或连接至 CA0 接口 1286 和 CA1 接口 1287 的动态存储器中所存储的数据。

[0138] 图 22 是示出操作存储器模块的方法的流程图。图 22 所示步骤可以由系统 100、系统 400、模块 500、系统 700、子系统 800、子系统 801、系统 900、子系统 1000、子系统 1001、系统 1100、子系统 1200、子系统 1201、系统 1300、系统 1400、系统 1500、系统 1600、系统 1700、系统 1800 和 / 或它们的部件中的一个或多个元件来执行。在集成电路设备处, 接收了来自与存储器模块的存储器接口连接的存储器控制器的存储器命令; 集成电路设备具有处理器、第一接口、第二接口和第三接口。第一接口被连接至多个动态存储器集成电路。第二接

口被连接至至少一个非易失性存储器集成电路设备。第三接口接收了来自存储器控制器的存储器命令 (2202)。例如,集成电路设备 721 可以从存储器控制器 714(其被耦合至模块 720 的存储器接口)接收存储器命令。例如,集成电路设备 721 可以是系统 800。系统 800 具有可以连接至 CPU 存储器 723 的 CA0 接口 886 和 CA1 接口 887。系统 800 具有 NV 控制器 827(其可以对应于 NV 控制器 725)以向 NV 存储器 828(可以对应于 NV 存储器 728)接口。系统 800 具有 CA 输入接口 896,该 CA 输入接口 896 可以接收来自存储器控制器 714 的存储器命令。

[0139] 响应于存储器命令,数据被存储在多个动态存储器集成电路中(2204)。例如,响应于经由 CA 输入接口 896 从存储器控制器 714 接收的命令,系统 800 可以将数据存储在连接至 CA0 接口 886 或 CA1 接口 887 的动态存储器集成电路(例如,CPU 存储器 723)中。

[0140] 响应于来自处理器的命令,数据被存储在至少一个非易失性存储器集成电路设备中(2206)。例如,系统 800 可以响应于处理器 812(或处理器 722)并且在 NV 存储器 828 中存储在 CPU 存储器 723、本地存储器 824 和 / 或连接至 CA0 接口 886 和 CA1 接口 887 的动态存储器中所存储的数据。

[0141] 图 23 是示出操作存储器模块的方法的流程图。图 23 所示步骤可以由系统 100、系统 400、模块 500、系统 700、子系统 800、子系统 801、系统 900、子系统 1000、子系统 1001、系统 1100、子系统 1200、子系统 1201、系统 1300、系统 1400、系统 1500、系统 1600、系统 1700、系统 1800 和 / 或它们的部件中的一个或多个元件来执行。在模块处,经由 DRAM 存储器通道接收到用于块 I/O 存储类型设备命令的远程过程调用请求(2302)。例如,模块 720 可以经由耦合了模块 720 和存储器控制器 714 的 DRAM 存储器通道接收用于块 I/O 存储类型设备的远程过程调用请求(例如,SSD 的块读取)。

[0142] 使用模块上的非易失性存储来服务块 I/O 存储类型设备命令(2304)。例如,可以使用存储在非易失性存储器 728 上的数据来提供远程过程调用响应。可以经由耦合了模块 720 和存储器控制器 714 的 DRAM 通道来传输远程过程调用响应。

[0143] 图 24 是示出操作存储器模块的方法的流程图。图 24 所示步骤可以由系统 100、系统 400、模块 500、系统 700、子系统 800、子系统 801、系统 900、子系统 1000、子系统 1001、系统 1100、子系统 1200、子系统 1201、系统 1300、系统 1400、系统 1500、系统 1600、系统 1700、系统 1800 和 / 或它们的部件中的一个或多个元件来执行。在模块处,接收到通过 DRAM 存储器通道接口经由存储器写事务被传输至模块的过程调用和过程数据(2402)。例如,通过 DRAM 存储器通道接口经由存储器写事务被传输至模块的过程调用和过程数据可以被模块 720 接收,该 DRAM 存储器通道接口链接模块 720 和存储器控制器 714

[0144] 执行了过程调用并且通过 DRAM 存储器通道接口返回了返回的过程结果(2404)。例如,处理器 722 可以执行过程调用并且生成存储在非易失性存储器 728 中的过程结果。可以响应于经由 DRAM 存储器通道接口传输结果数据的读命令返回存储在非易失性存储器 728 中的过程结果,该 DRAM 存储器通道接口链接模块 720 和存储器控制器 714。

[0145] 图 25 是示出操作存储器模块的方法的流程图。图 25 所示步骤可以由系统 100、系统 400、模块 500、系统 700、子系统 800、子系统 801、系统 900、子系统 1000、子系统 1001、系统 1100、子系统 1200、子系统 1201、系统 1300、系统 1400、系统 1500、系统 1600、系统 1700、系统 1800 和 / 或它们的部件中的一个或多个元件来执行。在模块处并且经由 DRAM 存储

器通道接口,将由模块执行的过程调用经由 DRAM 存储器通道上的存储器写事务被接收,该存储器写事务被定址到模块的存储器空间 (2502)。例如,模块 720 可以从存储器控制器 714 接收定址到模块 720 的存储器空间的存储器写事务。这些存储器写事务可以传输将由模块执行的过程调用(例如,远程过程调用或者函数调用)。

[0146] 执行过程调用,该过程调用包括块 I/O 存储类型设备命令并且包括映射调用,使用非易失性存储来服务该块 I/O 存储类型设备命令,该映射调用指定将由模块对使用非易失性存储来存储的对应的输入数据块执行的专用映射操作,并且该映射操作还产生使用非易失性存储来存储的中间数据值 (2504)。例如,模块 720 可以被控制为使用利用非易失性存储器 728 存储的数据来执行“MapReduce”映射类型函数。

[0147] 图 26 是示出操作存储器模块的方法的流程图。图 26 所示步骤可以由系统 100、系统 400、模块 500、系统 700、子系统 800、子系统 801、系统 900、子系统 1000、子系统 1001、系统 1100、子系统 1200、子系统 1201、系统 1300、系统 1400、系统 1500、系统 1600、系统 1700、系统 1800 和 / 或它们的部件中的一个或多个元件来执行。在集成电路设备处,接收了来自与存储器模块的存储器接口连接的存储器控制器接收存储器命令;集成电路设备具有处理器、第一接口、第二接口和非易失性存储器。第一接口被连接至多个动态存储器集成电路。第二接口接收了来自存储器控制器的存储器命令 (2602)。例如,集成电路设备 921 可以从存储器控制器 914(其被耦合至模块 920 的存储器接口)接收存储器命令。例如,集成电路设备 921 可以是系统 1000。系统 1000 具有可以连接至 CPU 存储器 923 的 CA0 接口 1086 和 CA1 接口 1087。系统 1000 具有 NV 存储器 1028(可对应于 NV 存储器 928)。系统 1000 具有 CA 输入接口 1096,该 CA 输入接口 1096 可以接收来自存储器控制器 914 的存储器命令。

[0148] 响应于存储器命令,数据被存储在多个动态存储器集成电路中 (2604)。例如,响应于经由 CA 输入接口 1096 从存储器控制器 914 接收的命令,系统 1000 可以将数据存储在连接至 CA0 接口 1086 或 CA1 接口 1087 的动态存储器集成电路(例如,CPU 存储器 923)中。

[0149] 响应于来自处理器的命令,数据被存储在至少一个非易失性存储器集成电路设备中 (2606)。例如,系统 1000 可以响应于处理器 1012(或处理器 922)并且在 NV 存储器 1028 中存储在 CPU 存储器 923、本地存储器 1024 和 / 或连接至 CA0 接口 1086 或 CA1 接口 1087 的动态存储器中所存储的数据。

[0150] 以上描述的系统和设备可以在计算机系统、集成电路中实现或者由计算机系统存储。以上描述的系统还可以存储在非瞬态计算机可读介质上。本文所描述的设备、电路和系统可以使用本领域可用的计算机辅助设计工具来实现,并且通过包含这种电路的软件描述的计算机可读文件来具体化。图 19 所示的步骤可以由系统 100、系统 400、模块 500、系统 700、子系统 800、子系统 801、系统 900、子系统 1000、子系统 1001、系统 1100、子系统 1200、子系统 1201、系统 1300、系统 1400、系统 1500、系统 1600、系统 1700、系统 1800 和 / 或它们的部件中的一个或多个元件来执行。这些软件描述可以是:行为、寄存器传送、逻辑部件、晶体管和布局几何级描述。此外,软件描述可以存储在非暂态存储介质上或者通过载波来传输。

[0151] 可以实现这种描述的数据格式包括但不限于:支持行为语言、如 C 的格式,支持寄存器传送级 (RTL) 语言、如 Verilog 和 VHDL 的格式,支持几何描述语言、诸如 GDSII、GDSIII、GDSIV、CIF 和 MEBES 的格式,以及其他适当的格式和语言。此外,可以在互联网上

或例如经由电子邮件通过各种不同的介质电子地完成这种文件在机器可读介质上的数据传送。注意,可以在机器可读介质,诸如 4mm 磁带、8mm 磁带、3-1/2 英寸软盘介质、CD、DVD、蓝光盘等上实现物理文件。

[0152] 图 27 示出了计算机系统的框图。计算机系统 2700 包括通信接口 2720、处理系统 2730、存储系统 2740 和用户接口 2760。处理系统 2730 被操作性地耦合至存储系统 2740。存储系统 2740 存储软件 2750 和数据 2770。计算机系统 2700 可以包括系统 100、系统 400、模块 500、系统 700、子系统 800、子系统 801、系统 900、子系统 1000、子系统 1001、系统 1100、子系统 1200、子系统 1201、系统 1300、系统 1400、系统 1500、系统 1600、系统 1700、系统 1800 和 / 或它们的部件中的一个或多个,或者实现本文所述方法、功能、处理、通信、电路和 / 或系统的部件。处理系统 2730 被操作性地耦合至通信接口 2720 和用户接口 2760。计算机系统 2700 可以包括被编程的通用计算机。计算机系统 2700 可以包括微处理器。计算机系统 2700 可以包括可编程电路或专用电路。计算机系统 2700 可以被分布在多个设备、处理器、存储器和 / 或接口之中,它们一同包括元件 2720-2770。

[0153] 通信接口 2720 可以包括网络接口、调制解调器、端口、总线、链接、收发器或其他通信设备。通信接口 2720 可以被分布在多个通信设备之中。处理系统 2730 可以包括微处理器、微控制器、逻辑电路或其他处理设备。处理系统 2730 可以被分布在多个处理设备之中。用户接口 2760 可以包括键盘、鼠标、声音识别接口、麦克风和扬声器、图形显示器、触摸屏或其他类型的用户接口设备。用户接口 2760 可以被分布在多个接口设备之中。存储系统 2740 可以包括磁盘、磁带、集成电路、RAM、ROM、EEPROM、闪存、网络存储器、服务器或其他存储器功能。存储系统 2740 可以包括计算机可读介质。存储系统 2740 可以被分布在多个存储器设备之中。

[0154] 处理系统 2730 检索和执行来自存储系统 2740 的软件 2750。处理系统 2730 可以检索和存储数据 2770。处理系统 2730 还可以经由通信接口 2720 检索和存储数据。处理系统 2730 可以创建或修改软件 2750 或数据 2770 以实现有形的结果。处理系统 2730 可以控制通信接口 2720 或用户接口 2760 以实现有形的结果。处理系统 2730 可以经由通信接口 2720 检索和执行远程存储的软件。

[0155] 软件 2750 和远程存储的软件可以包括操作系统、实体、驱动器、网络软件以及通常由计算机系统执行的其他软件。软件 2750 可以包括应用程序、小应用程序、固件或通常由计算机系统执行的机器可读处理指令的其他形式。软件 2750 或远程存储的软件当被处理系统 2730 执行时,可以引导计算机系统 2700 进行操作。

[0156] 为了说明和描述的目的呈现了前面本发明的描述。其不是排他性地并不将本发明限于公开的具体形式,在上述教导的基础上可以进行其他修改和变化。选择和描述实施例以更好地解释本发明的原理及其实际应用,从而能够使本领域技术人员能够在适合于预期特定应用的各个实施例和各个修改例中使用本发明。除了被现有技术限制的范围,所附权利要求包括本发明的其他可选实施例。

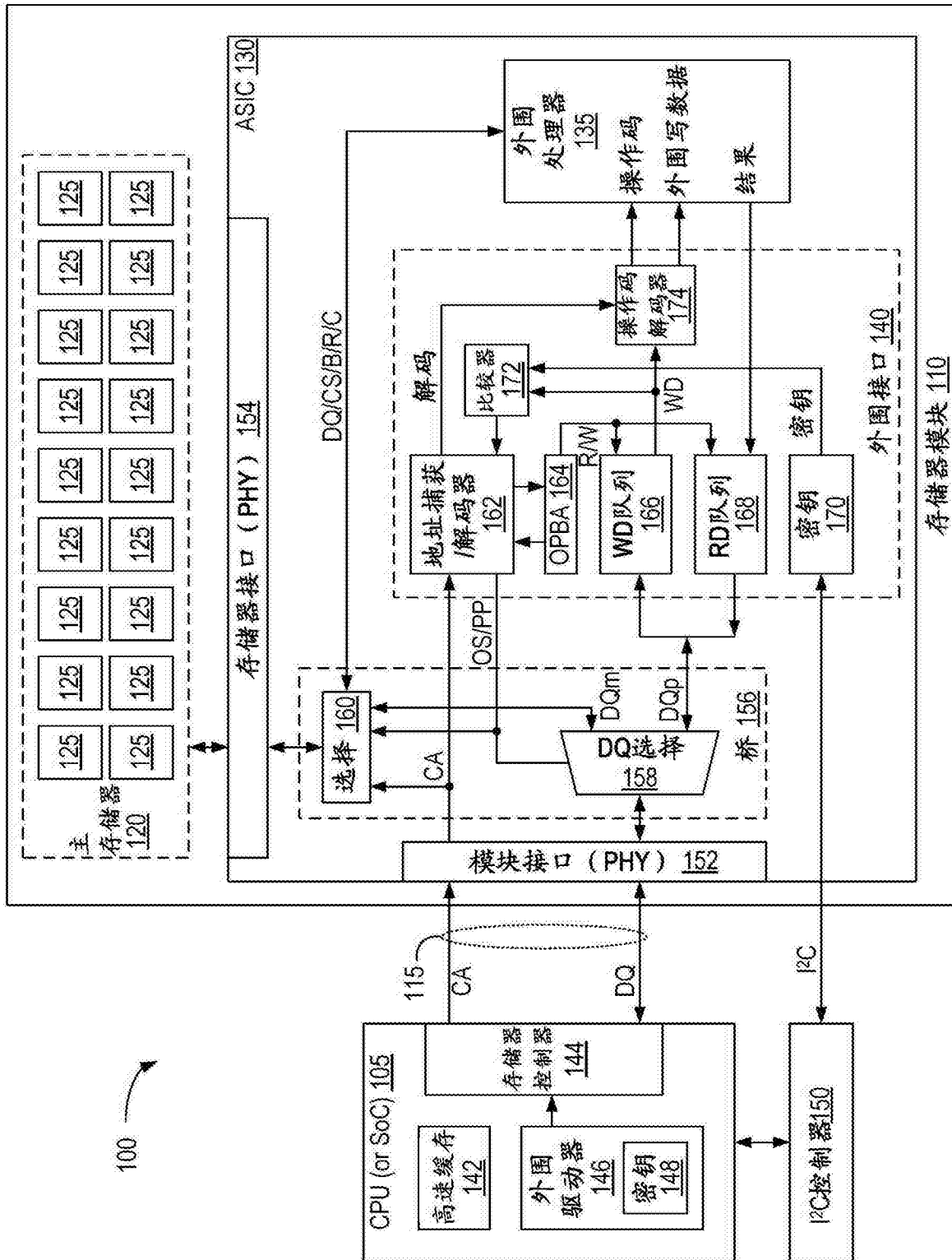


图 1

200

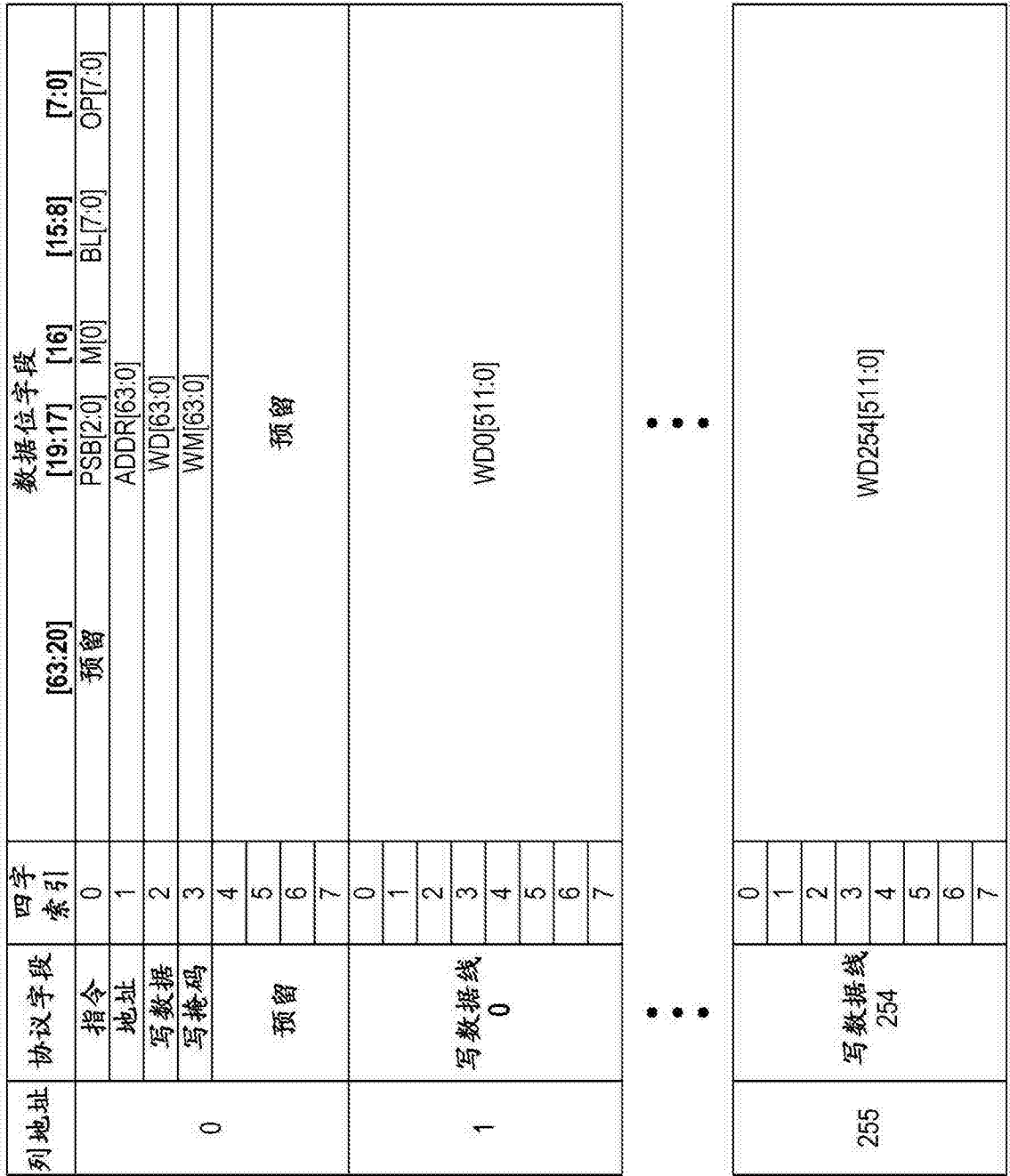


图 2

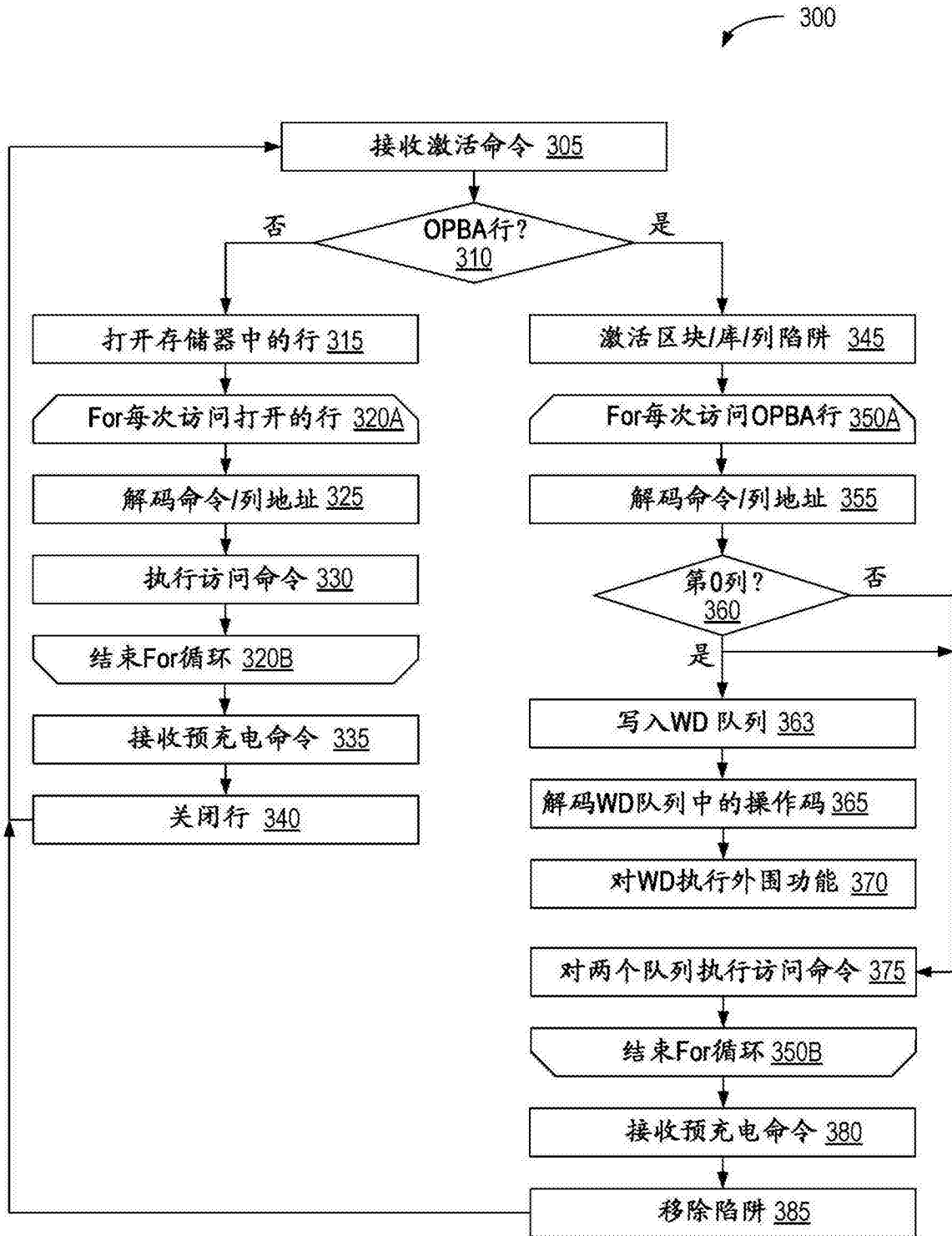


图 3

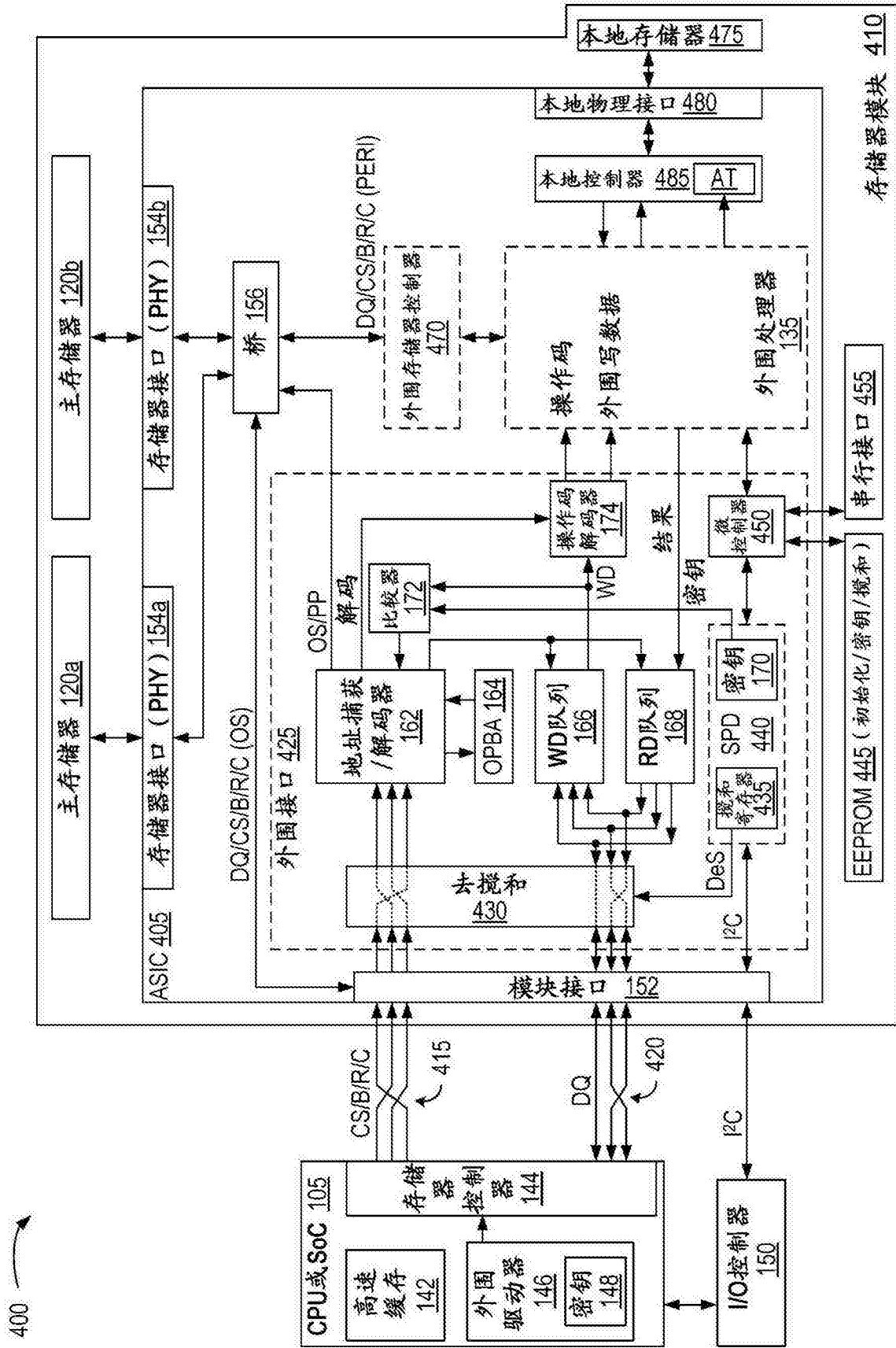


图 4



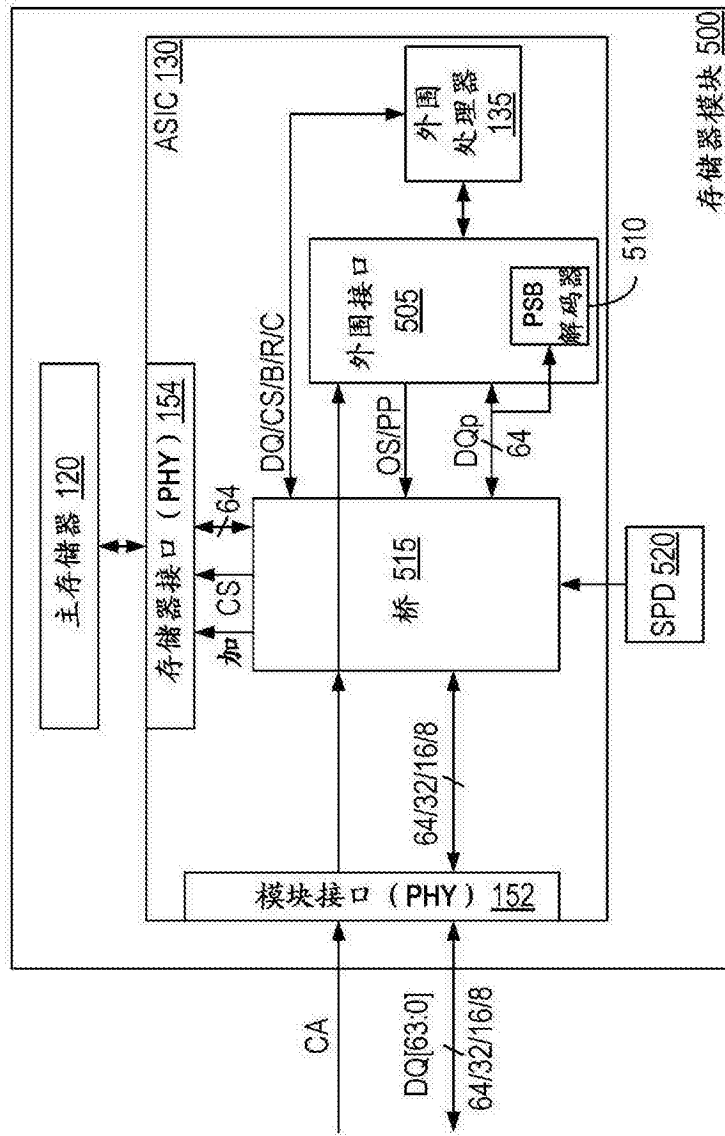


图 5

600

DIMM/ 通道	列 地址	协议字段	字节 索引	数据位字段									
				[63:56] D7	[55:48] D6	[47:40] D5	[39:32] D4	[31:24] D3	[23:16] D2	[15:8] D1	[7:0] D0		
0	0	操作码	0	OP7[7:0]	OP6[7:0]	OP5[7:0]	OP4[7:0]	OP3[7:0]	OP2[7:0]	OP1[7:0]	OP0[7:0]		
		突发长度	1	BL7[7:0]	BL6[7:0]	BL5[7:0]	BL4[7:0]	BL3[7:0]	BL2[7:0]	BL1[7:0]	BL0[7:0]		
		掩盖的WR	2	M7	M6	M5	M4	M3	M2	M1	M0		
		预留	7:3	预留	预留	预留	预留	预留	预留	预留	预留	预留	预留
1	0	操作码	0	OP7[7:0]	OP6[7:0]	OP5[7:0]	OP4[7:0]	OP3[7:0]	OP2[7:0]	OP1[7:0]	OP0[7:0]		
		突发长度	1	BL7[7:0]	BL6[7:0]	BL5[7:0]	BL4[7:0]	BL3[7:0]	BL2[7:0]	BL1[7:0]	BL0[7:0]		
		掩盖的WR	2	M7	M6	M5	M4	M3	M2	M1	M0		
		预留	7:3	预留	预留	预留	预留	预留	预留	预留	预留	预留	预留
2	0	操作码	0	OP7[7:0]	OP6[7:0]	OP5[7:0]	OP4[7:0]	OP3[7:0]	OP2[7:0]	OP1[7:0]	OP0[7:0]		
		突发长度	1	BL7[7:0]	BL6[7:0]	BL5[7:0]	BL4[7:0]	BL3[7:0]	BL2[7:0]	BL1[7:0]	BL0[7:0]		
		掩盖的WR	2	M7	M6	M5	M4	M3	M2	M1	M0		
		预留	7:3	预留	预留	预留	预留	预留	预留	预留	预留	预留	预留
3	0	操作码	0	OP7[7:0]	OP6[7:0]	OP5[7:0]	OP4[7:0]	OP3[7:0]	OP2[7:0]	OP1[7:0]	OP0[7:0]		
		突发长度	1	BL7[7:0]	BL6[7:0]	BL5[7:0]	BL4[7:0]	BL3[7:0]	BL2[7:0]	BL1[7:0]	BL0[7:0]		
		掩盖的WR	2	M7	M6	M5	M4	M3	M2	M1	M0		
		预留	7:3	预留	预留	预留	预留	预留	预留	预留	预留	预留	预留
3:0	1	地址		A7[63:0]	A6[63:0]	A5[63:0]	A4[63:0]	A3[63:0]	A2[63:0]	A1[63:0]	A0[63:0]		
3:0	2	WR数据		WD7[63:0]	WD6[63:0]	WD5[63:0]	WD4[63:0]	WD3[63:0]	WD2[63:0]	WD1[63:0]	WD0[63:0]		
3:0	3	WR掩码		WM7[63:0]	WM6[63:0]	WM5[63:0]	WM4[63:0]	WM3[63:0]	WM2[63:0]	WM1[63:0]	WM0[63:0]		
3:0	4	WR数据QW0		WD70[63:0]	WD60[63:0]	WD50[63:0]	WD40[63:0]	WD30[63:0]	WD20[63:0]	WD10[63:0]	WD00[63:0]		
3:0	255	WR数据QW251	7:0	WD7251[63:0]								WD0251[63:0]	

图 6

700 →

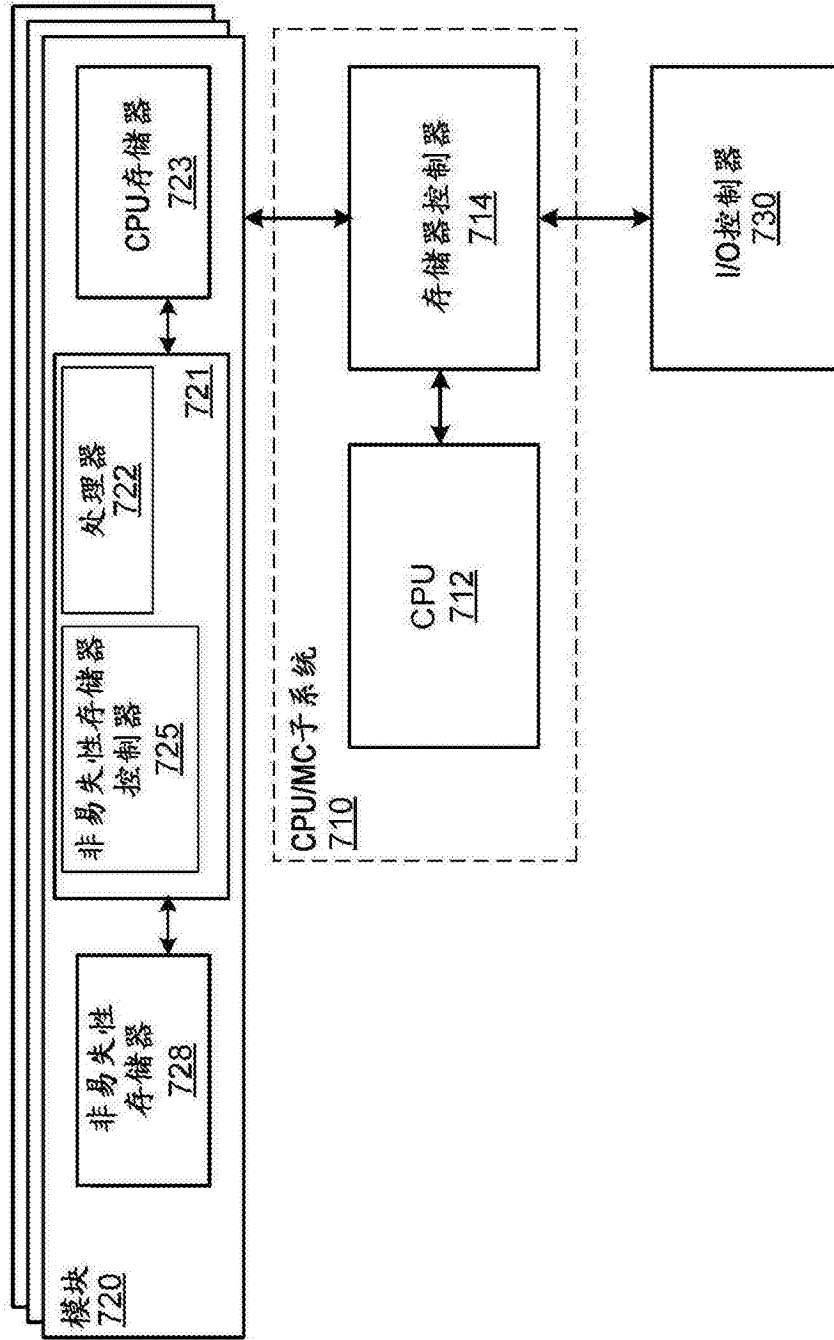


图 7

801

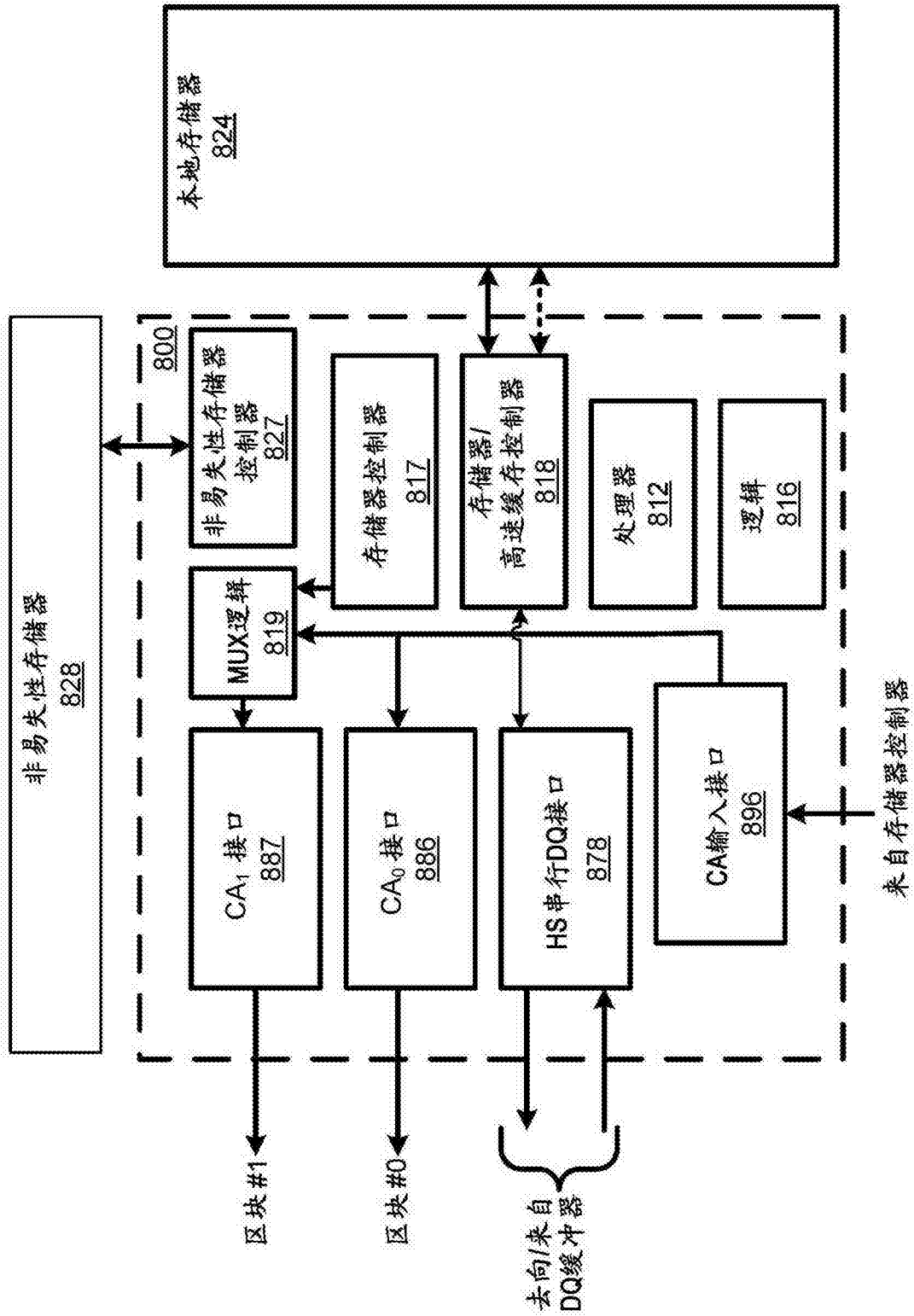


图 8

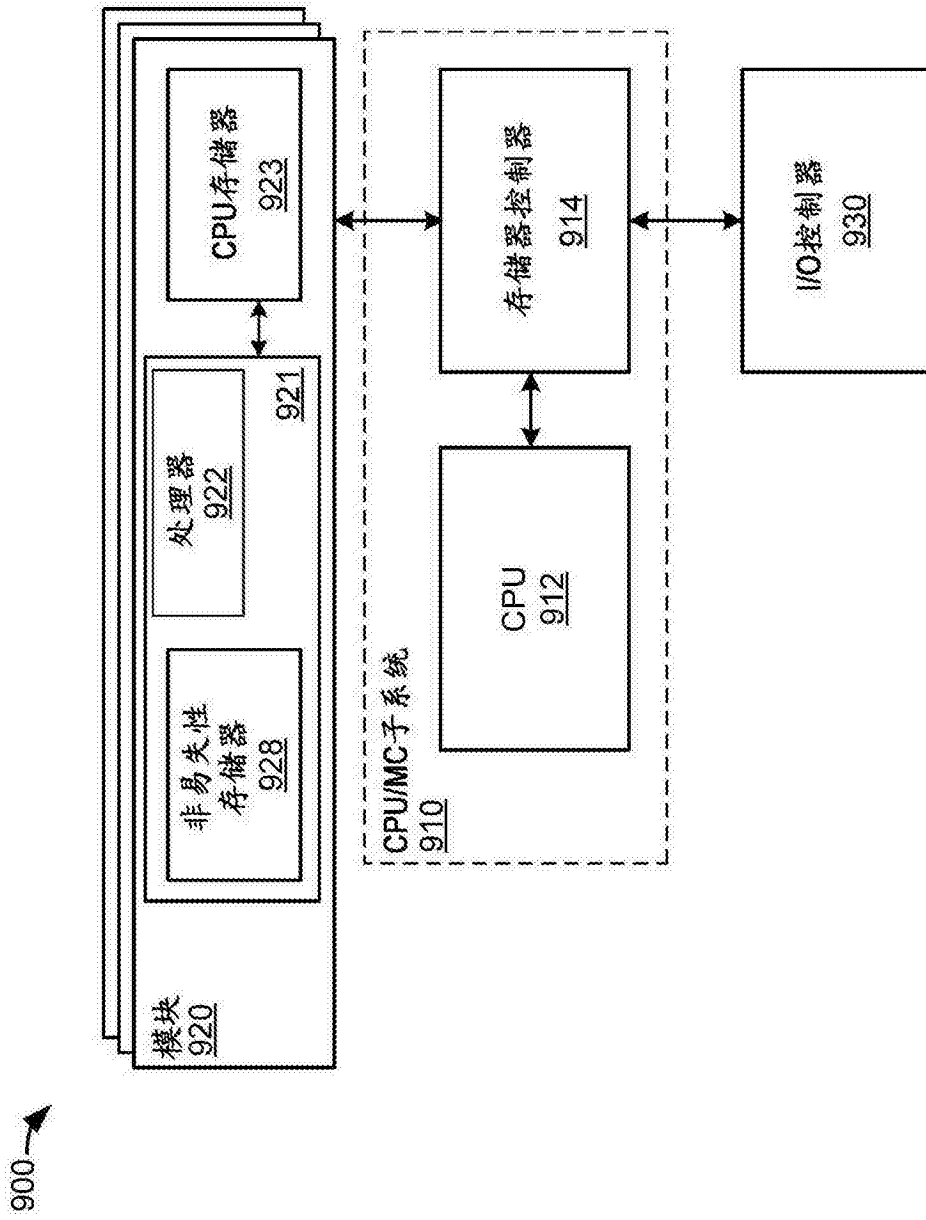


图 9

1001

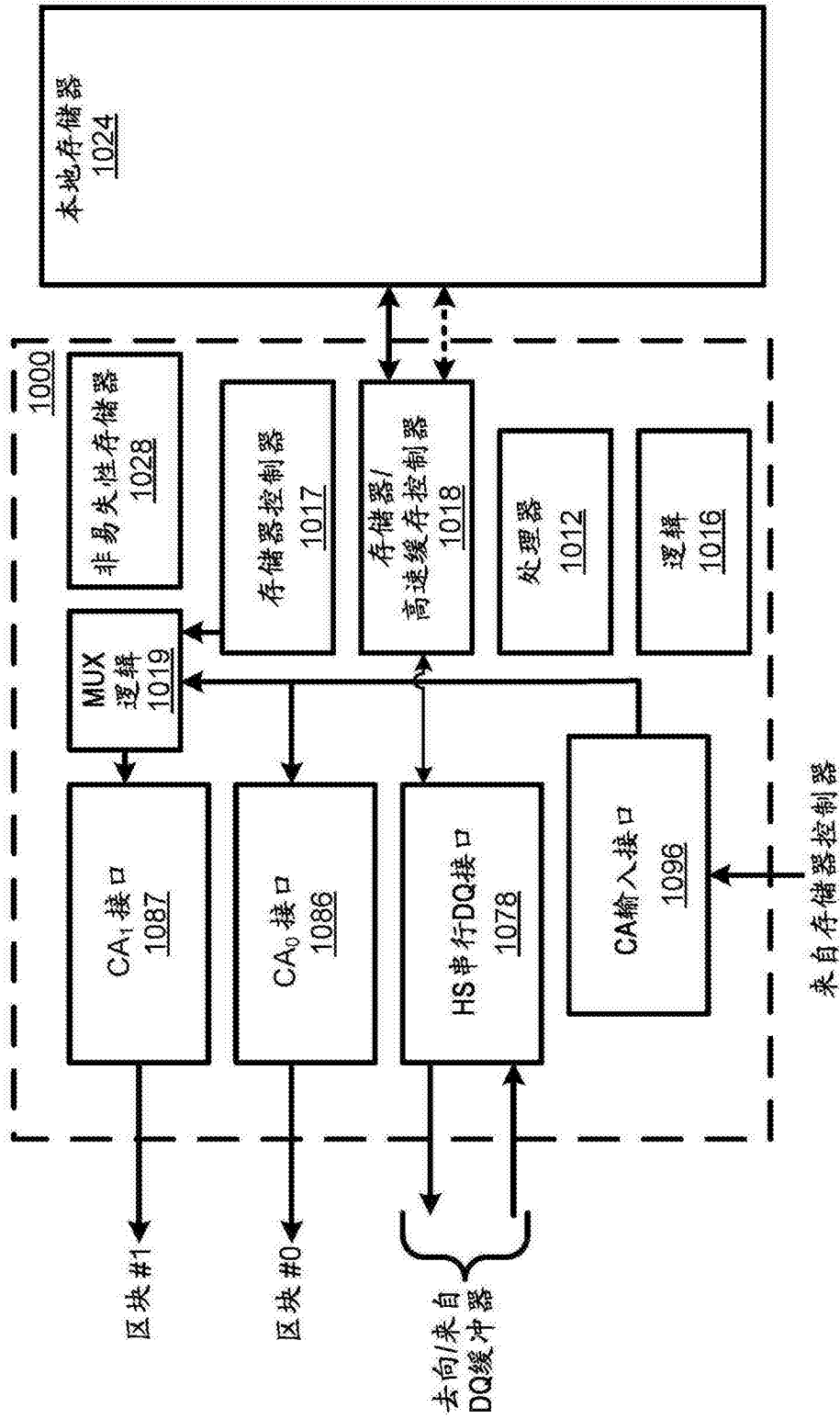


图 10

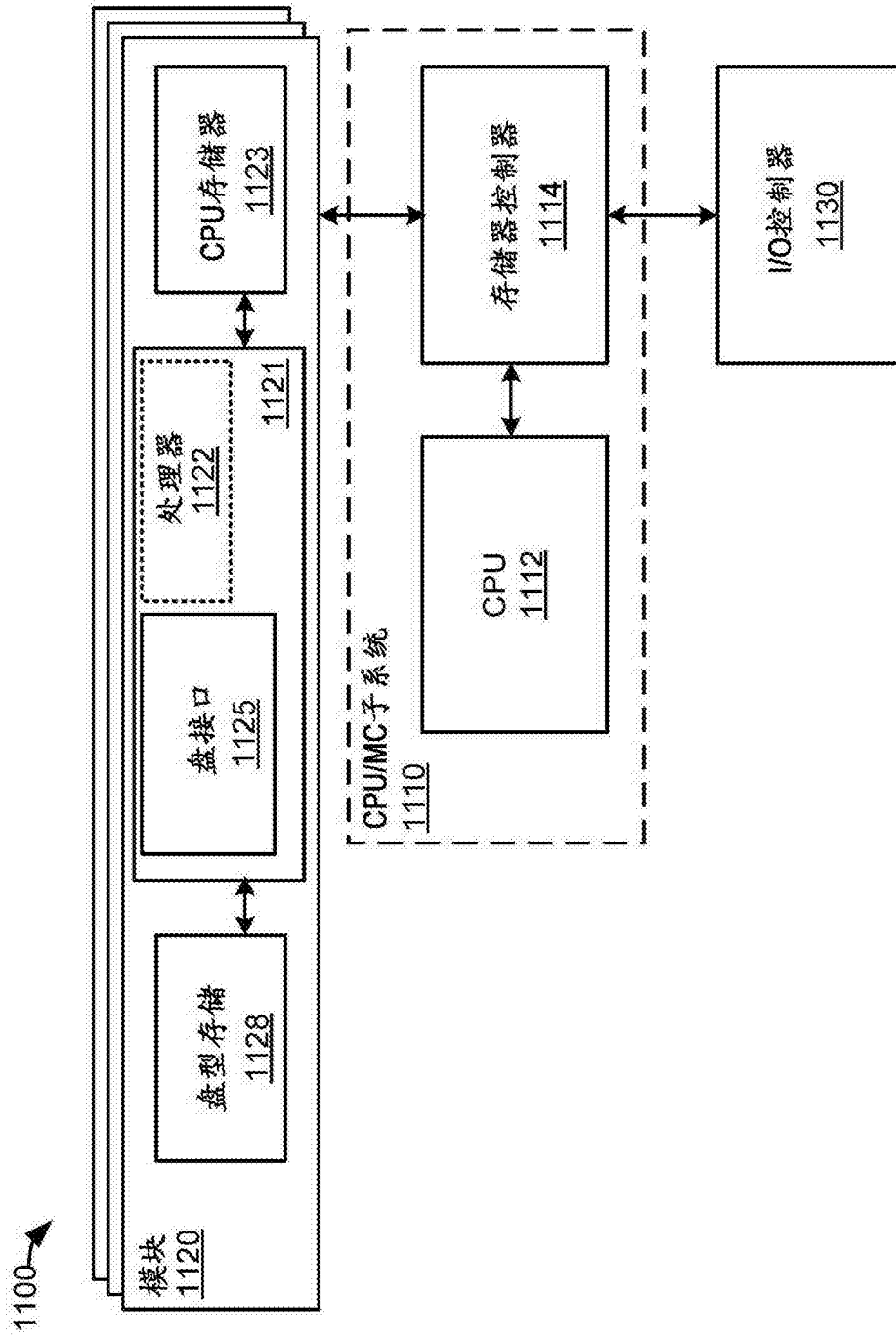


图 11

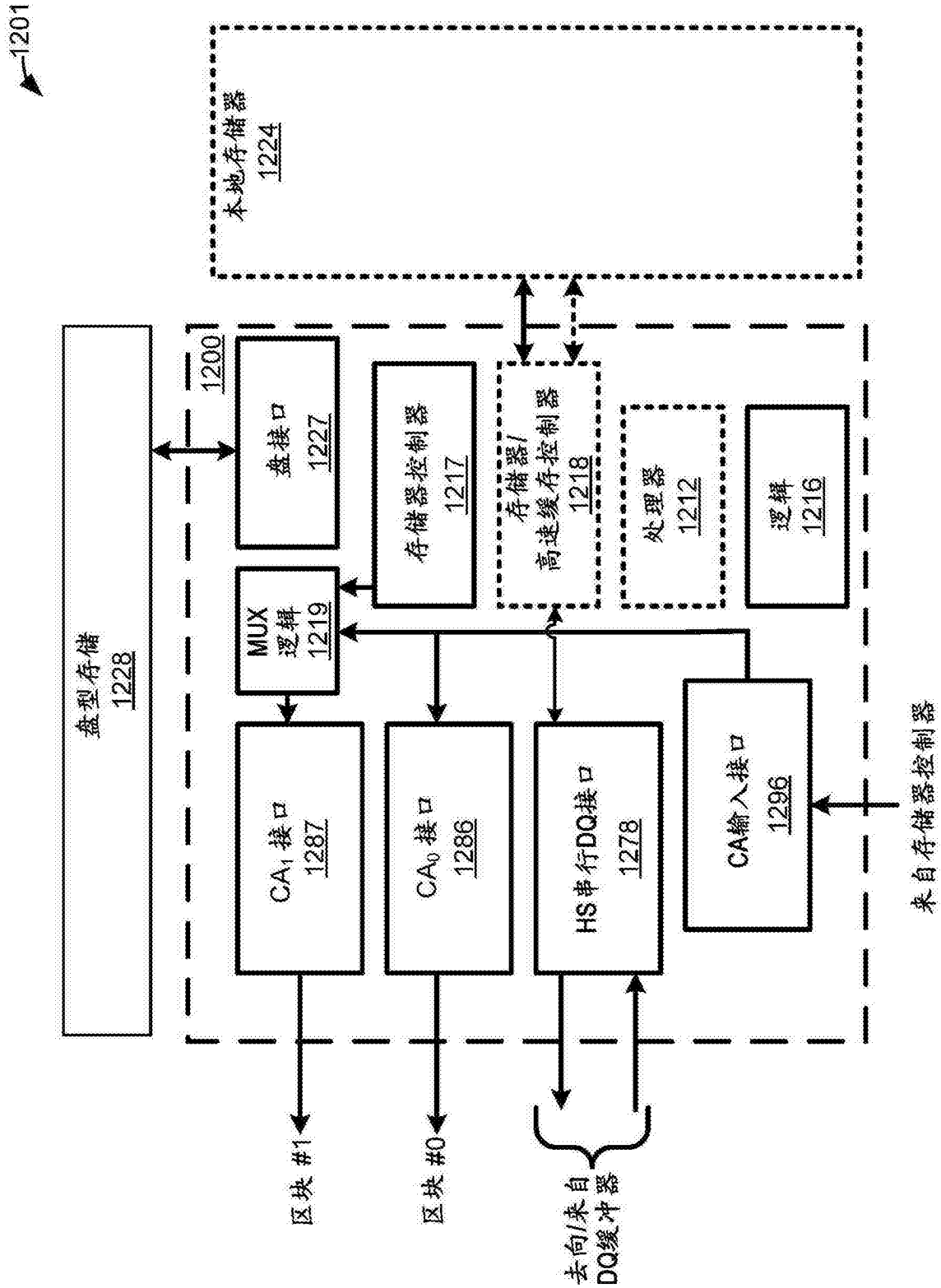


图 12



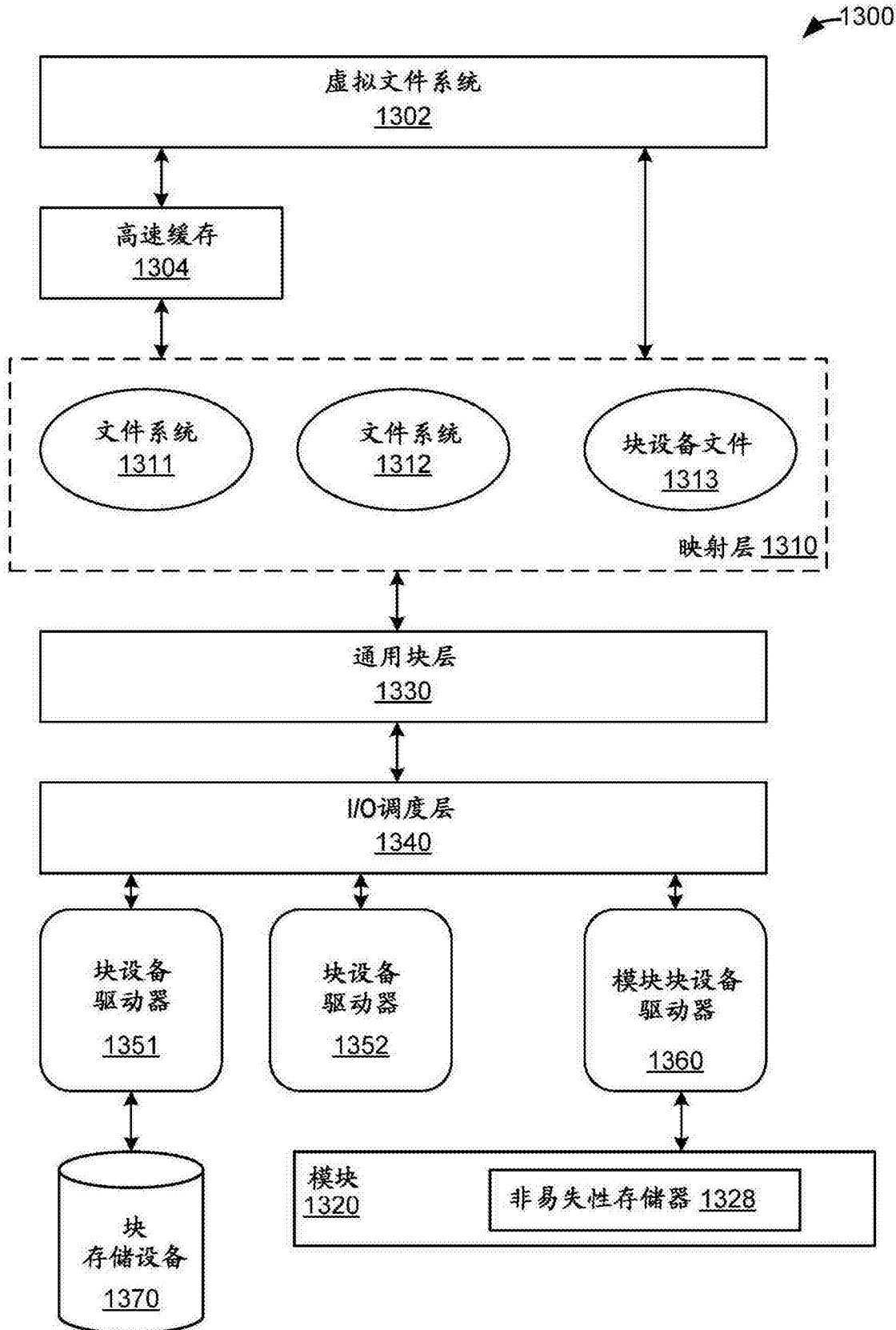


图 13

1400 →

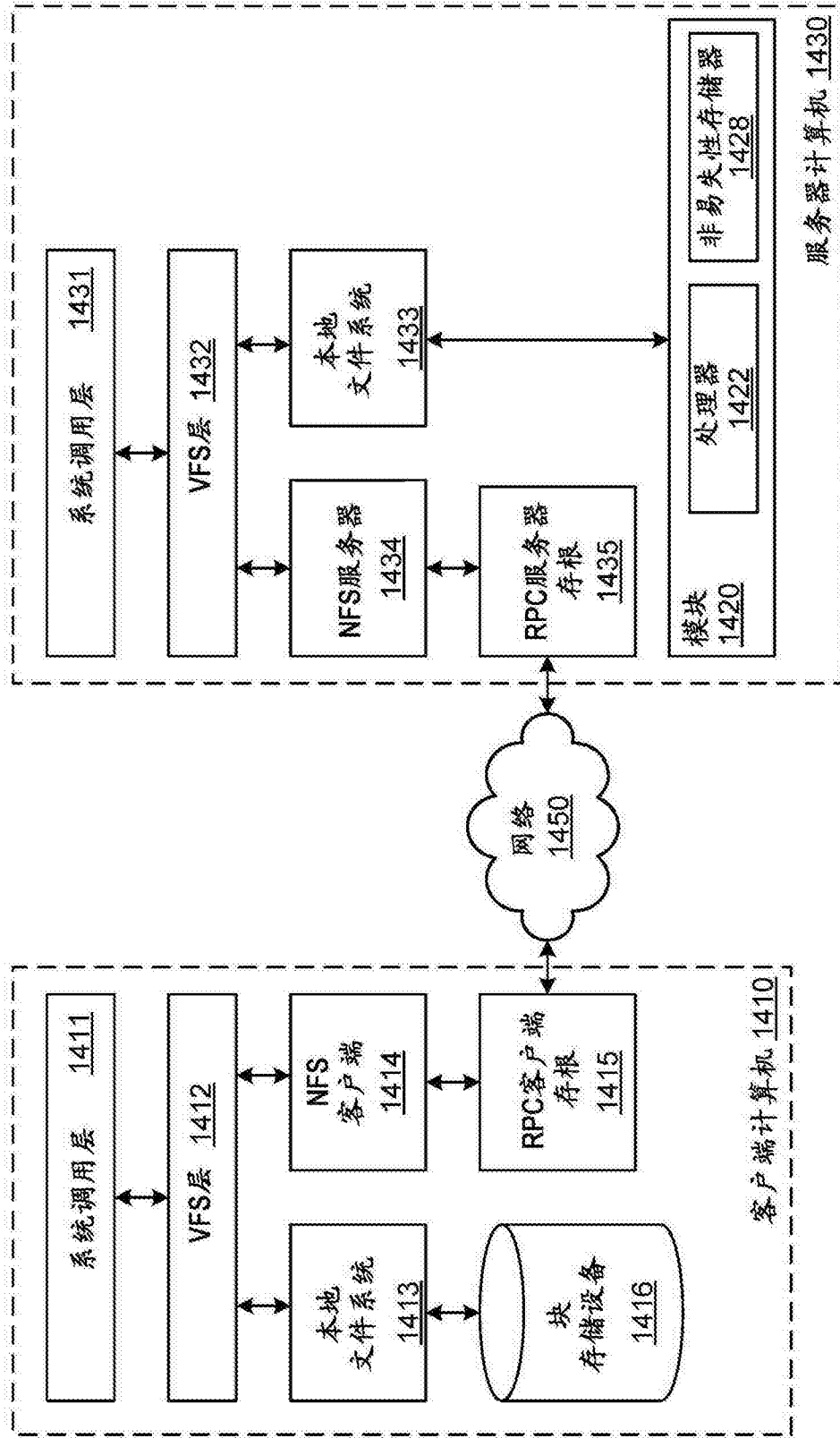


图 14

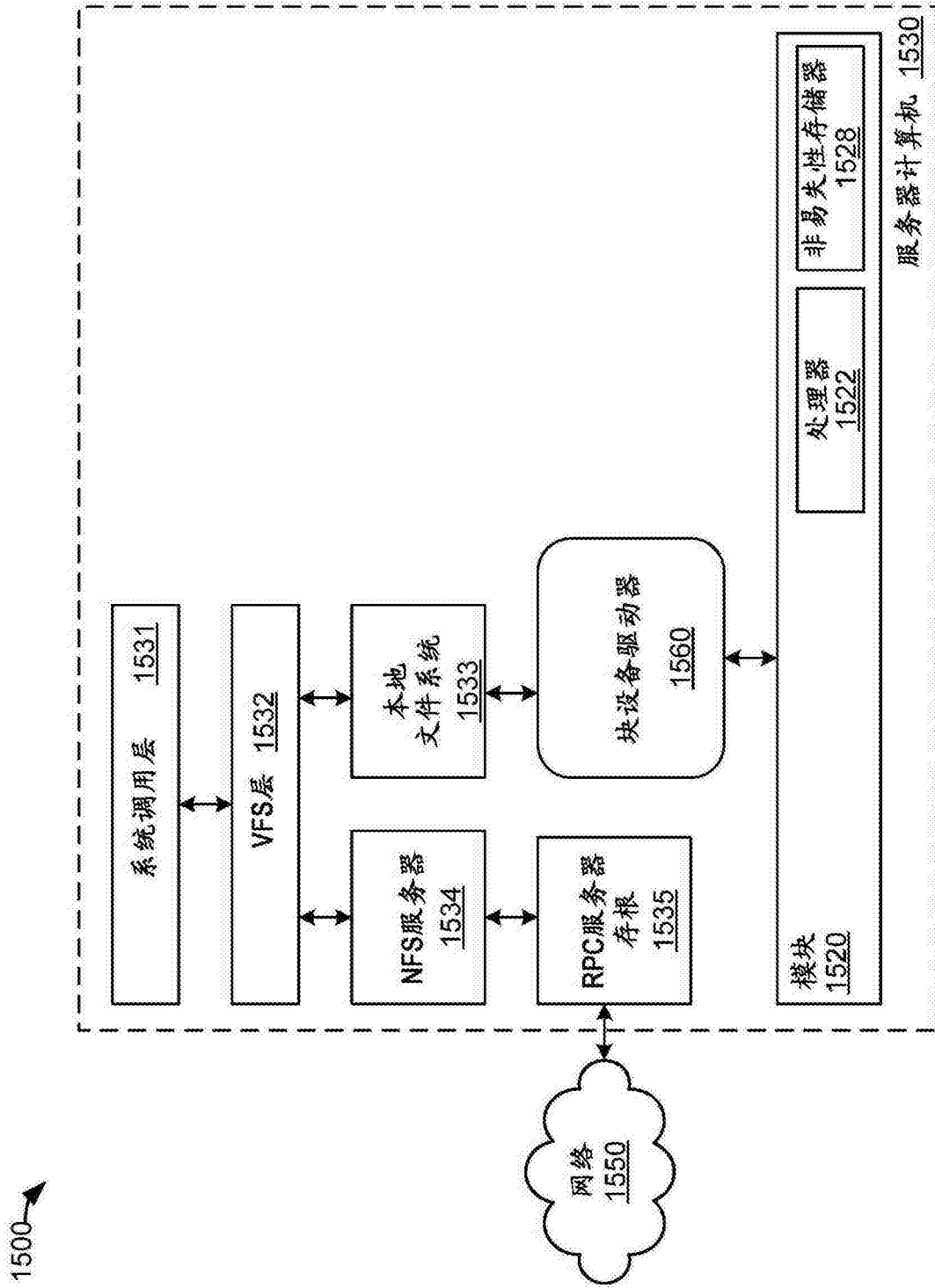


图 15

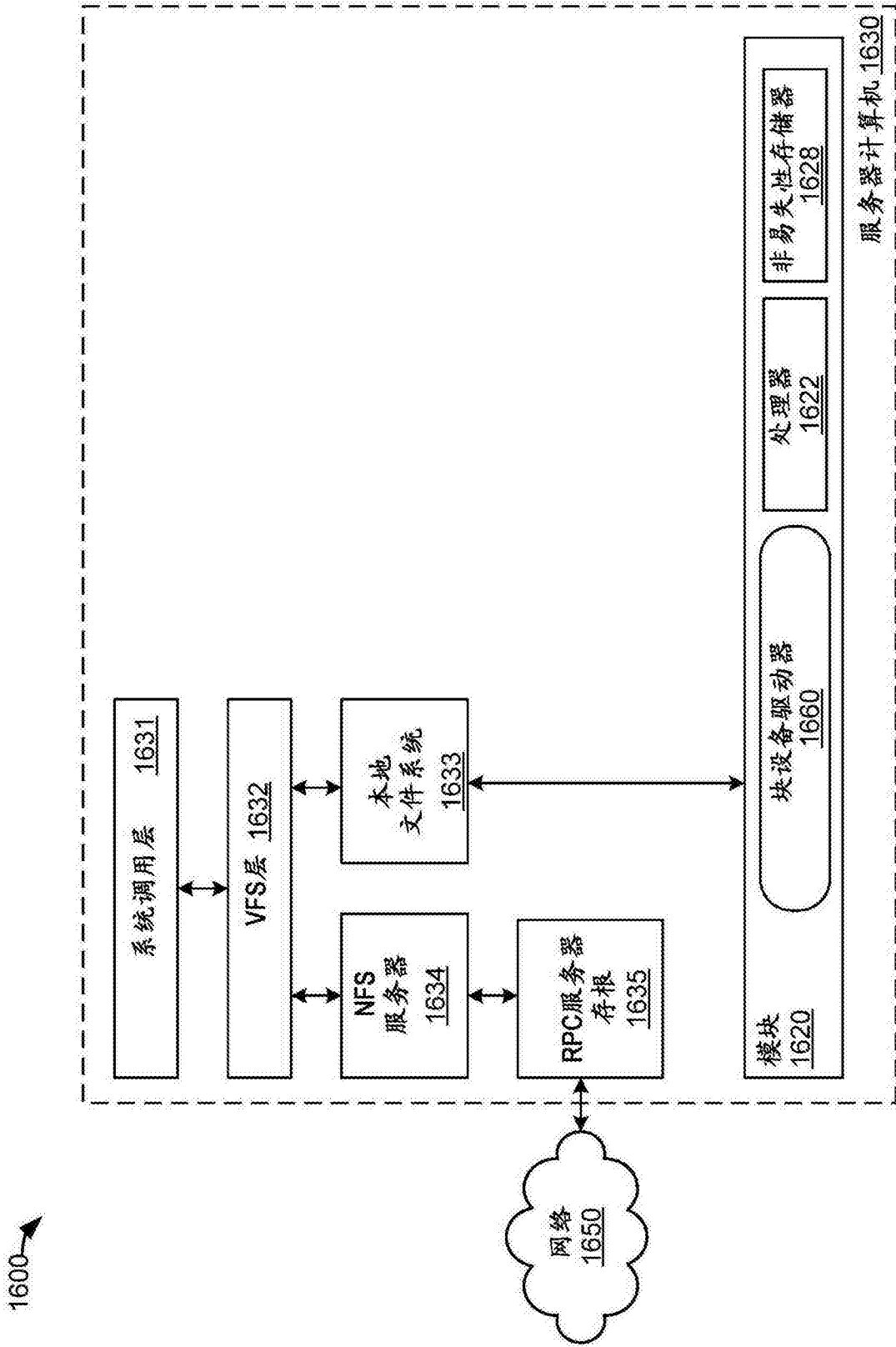


图 16

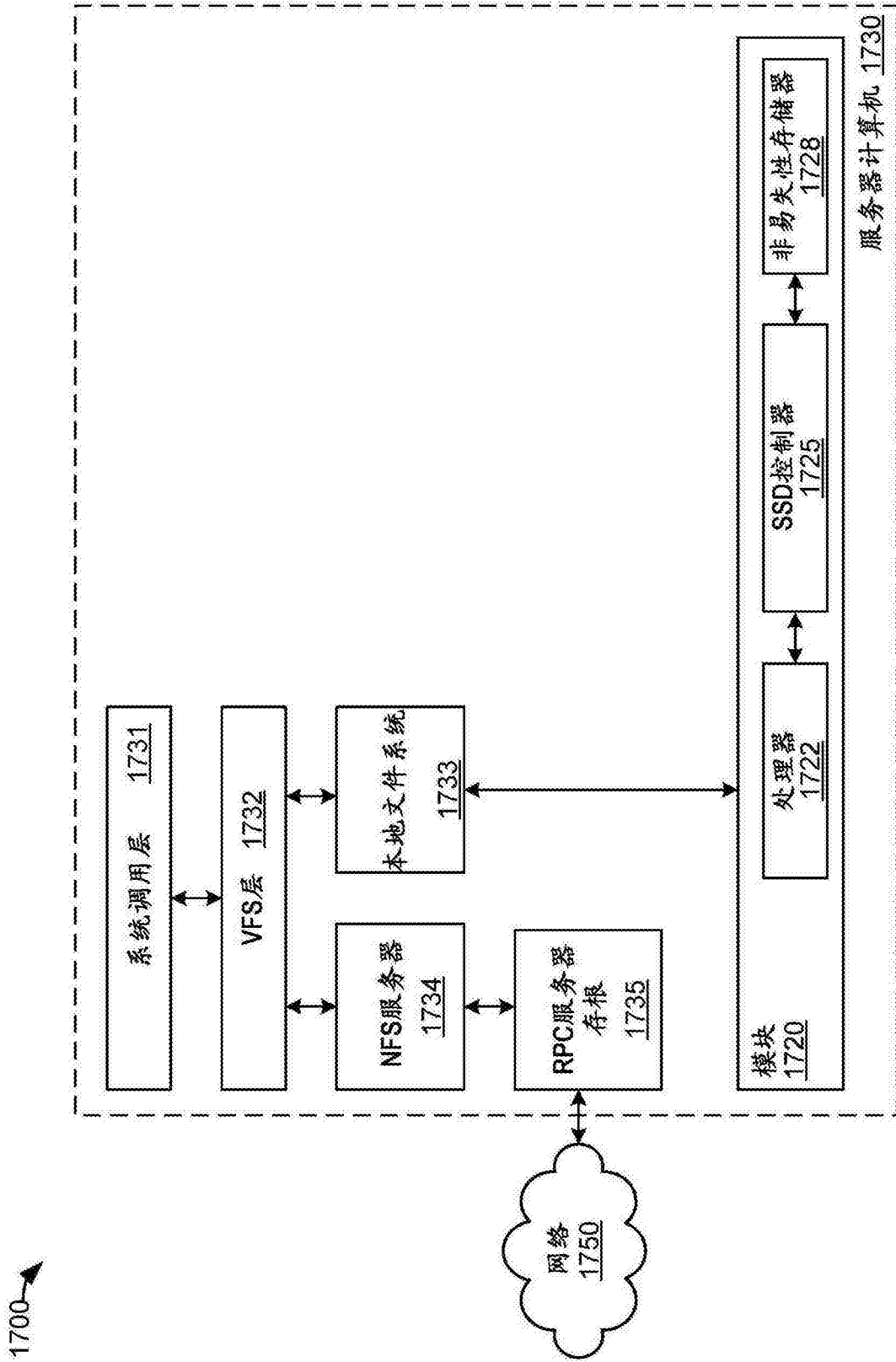


图 17

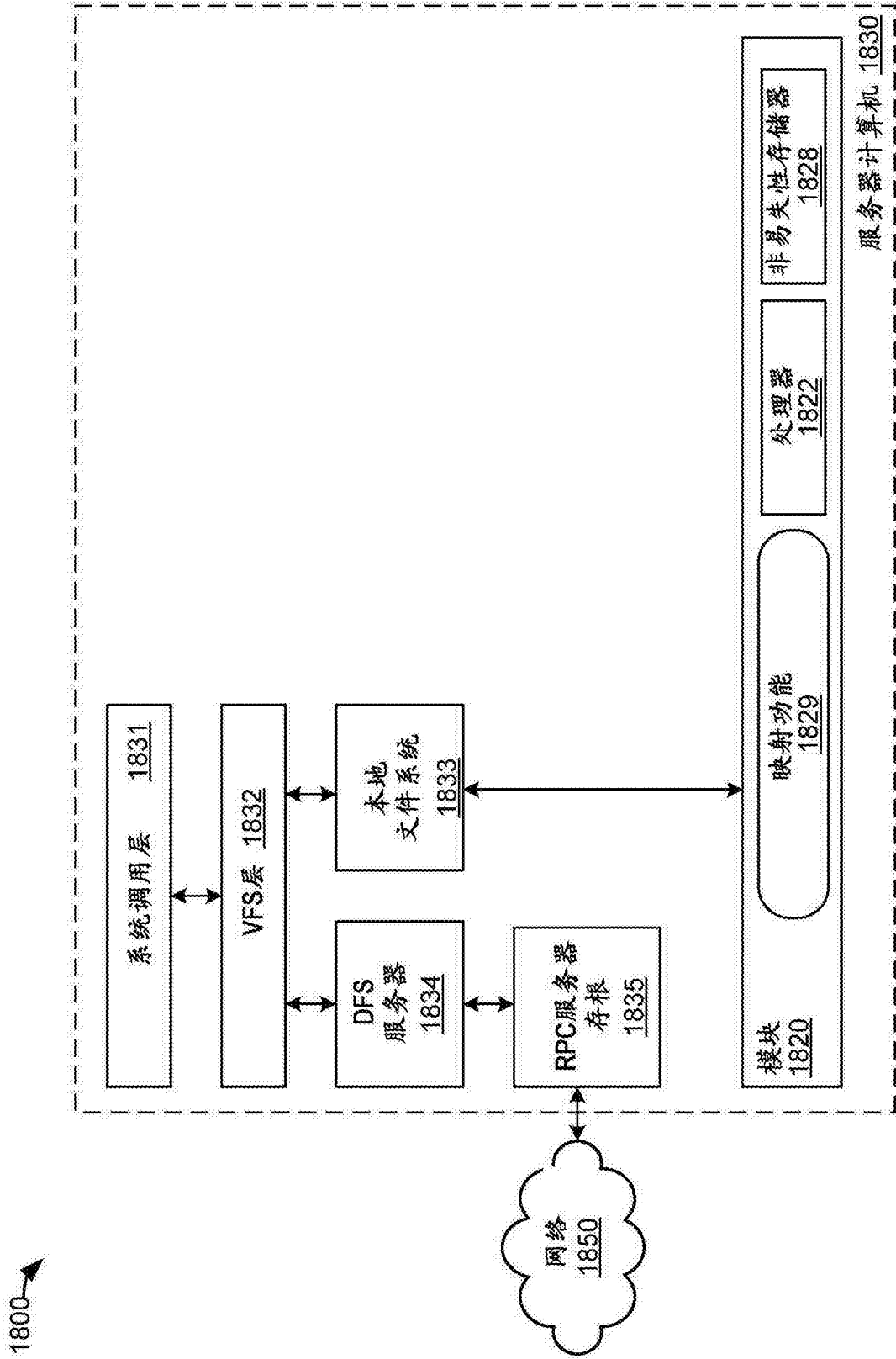


图 18

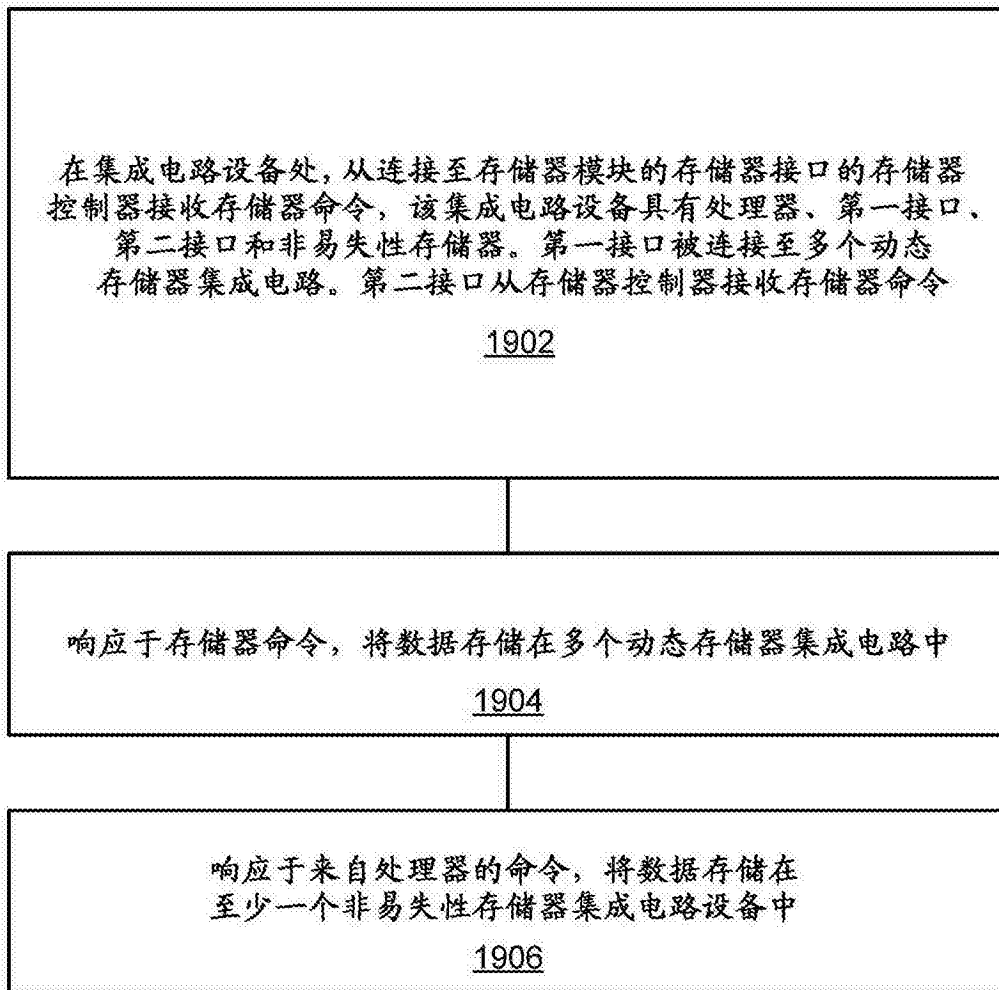


图 19

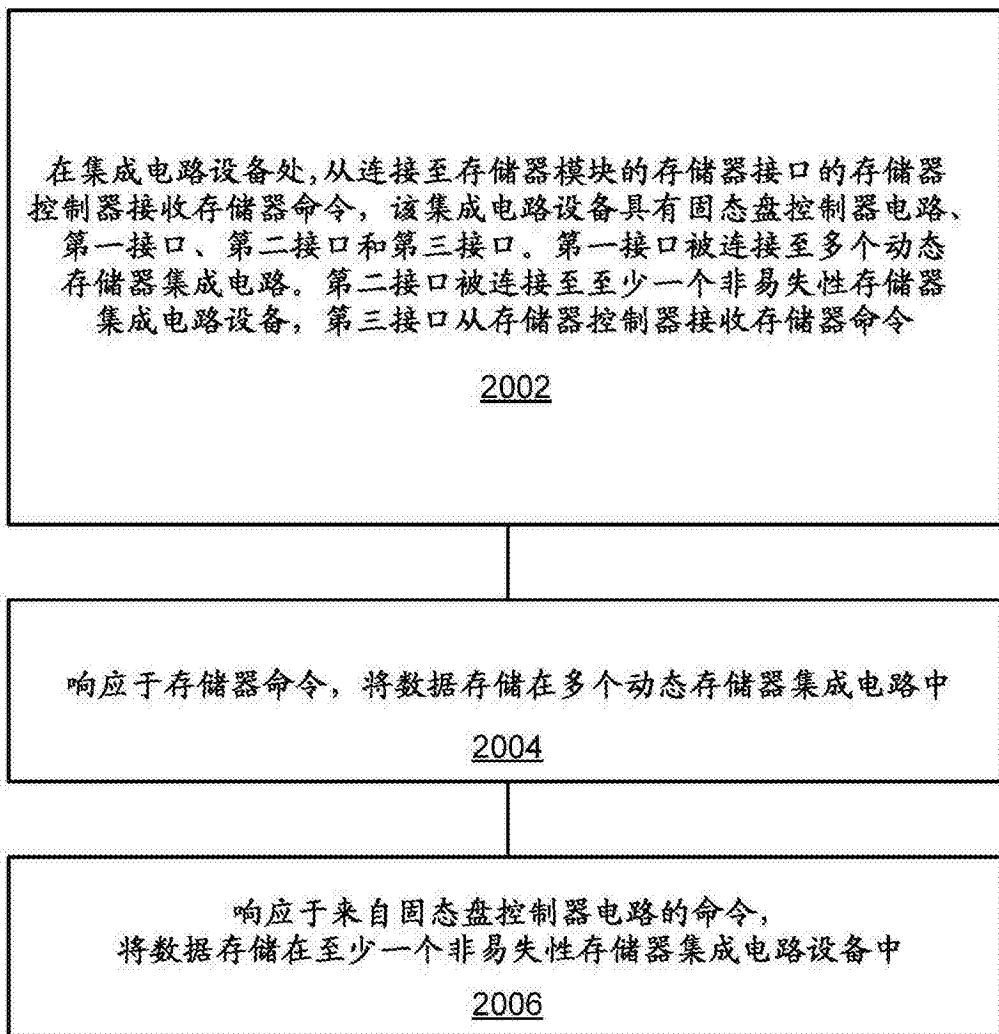


图 20



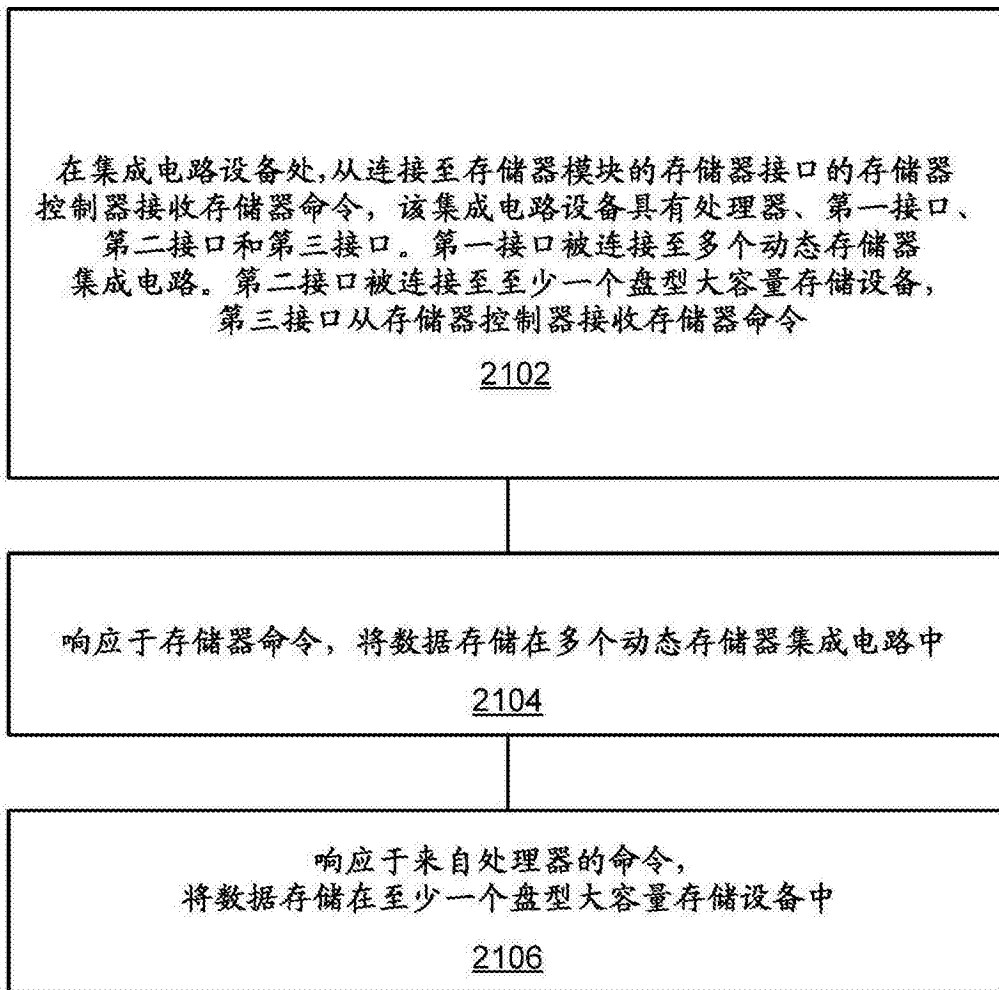


图 21

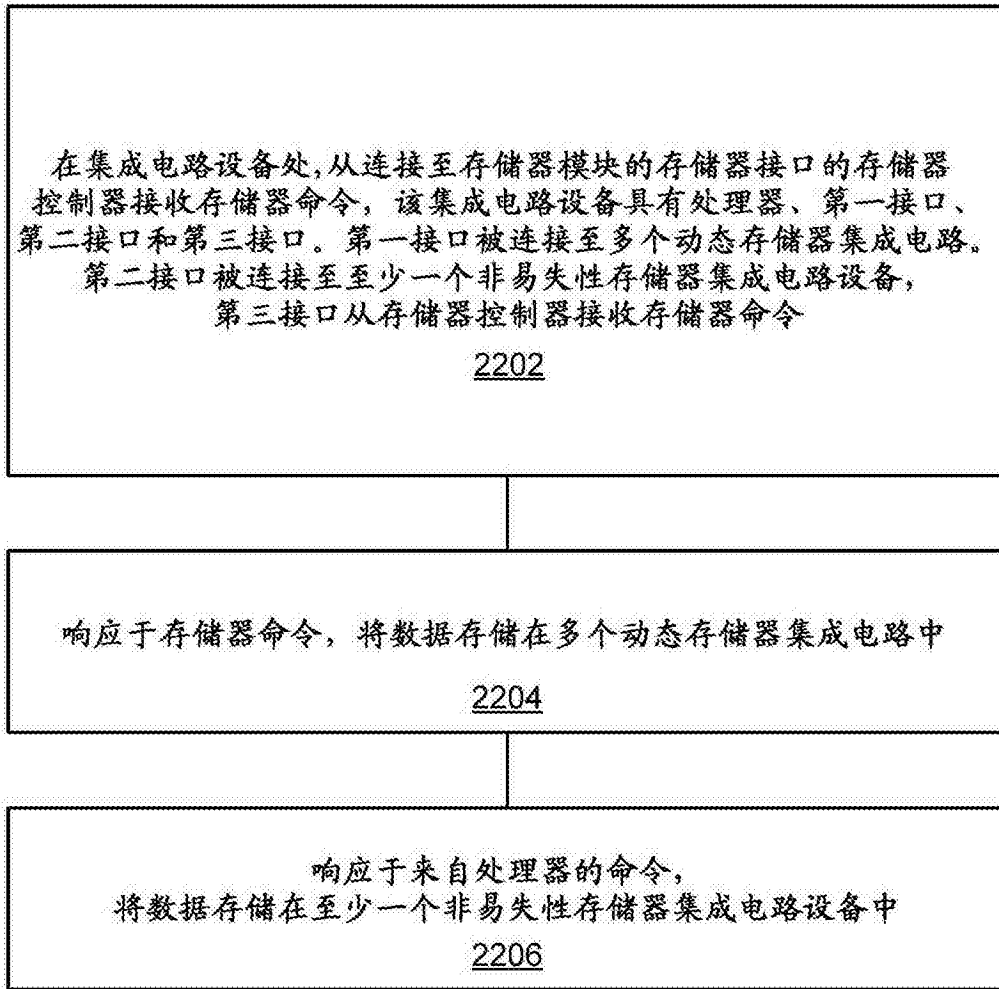


图 22

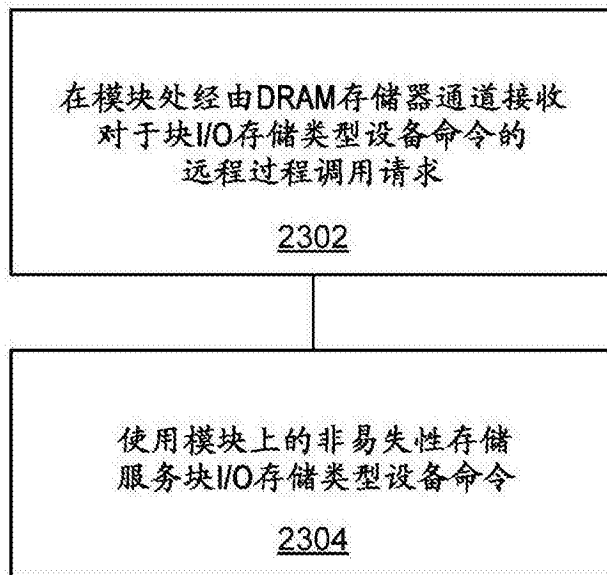


图 23

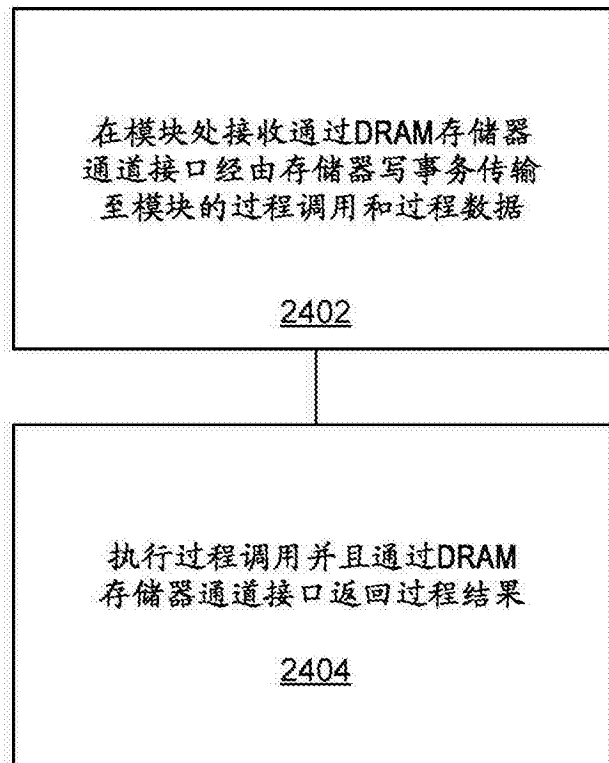


图 24

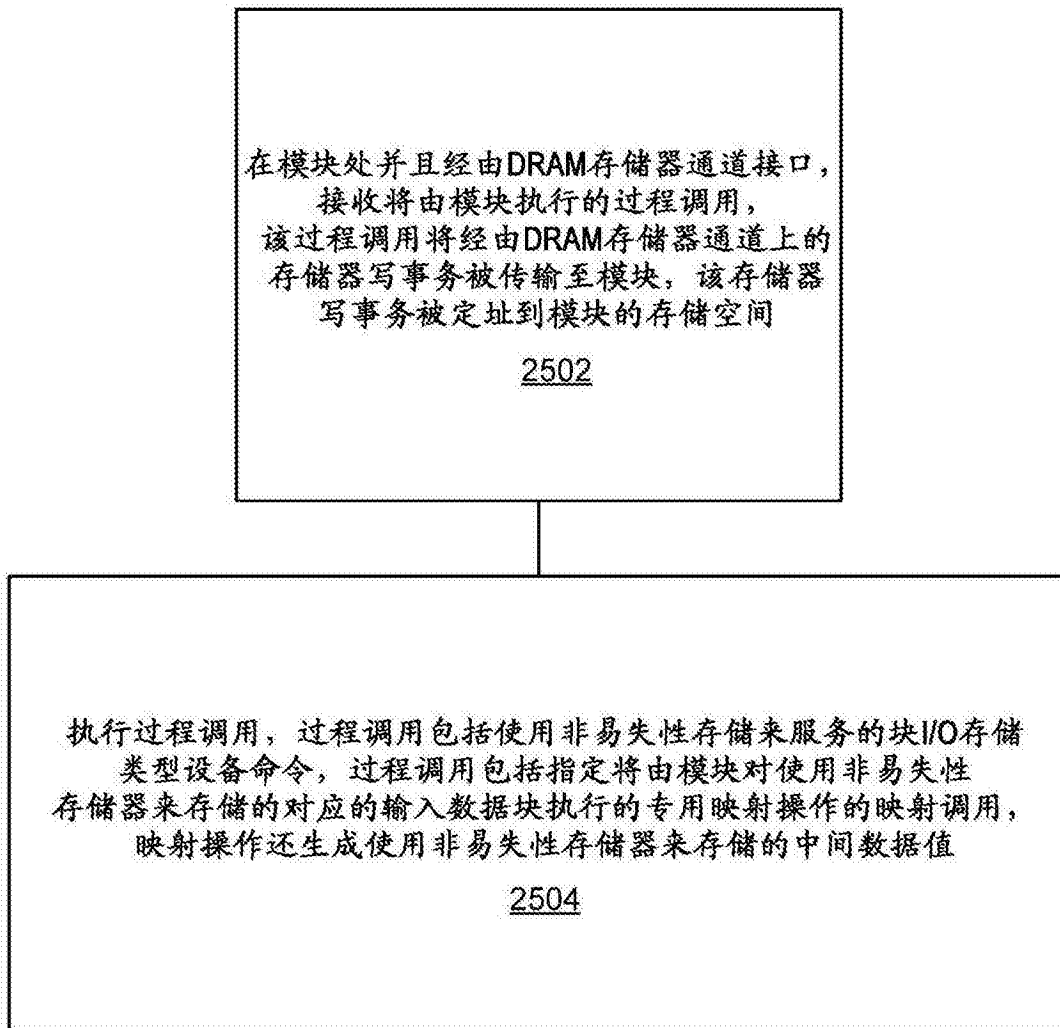


图 25

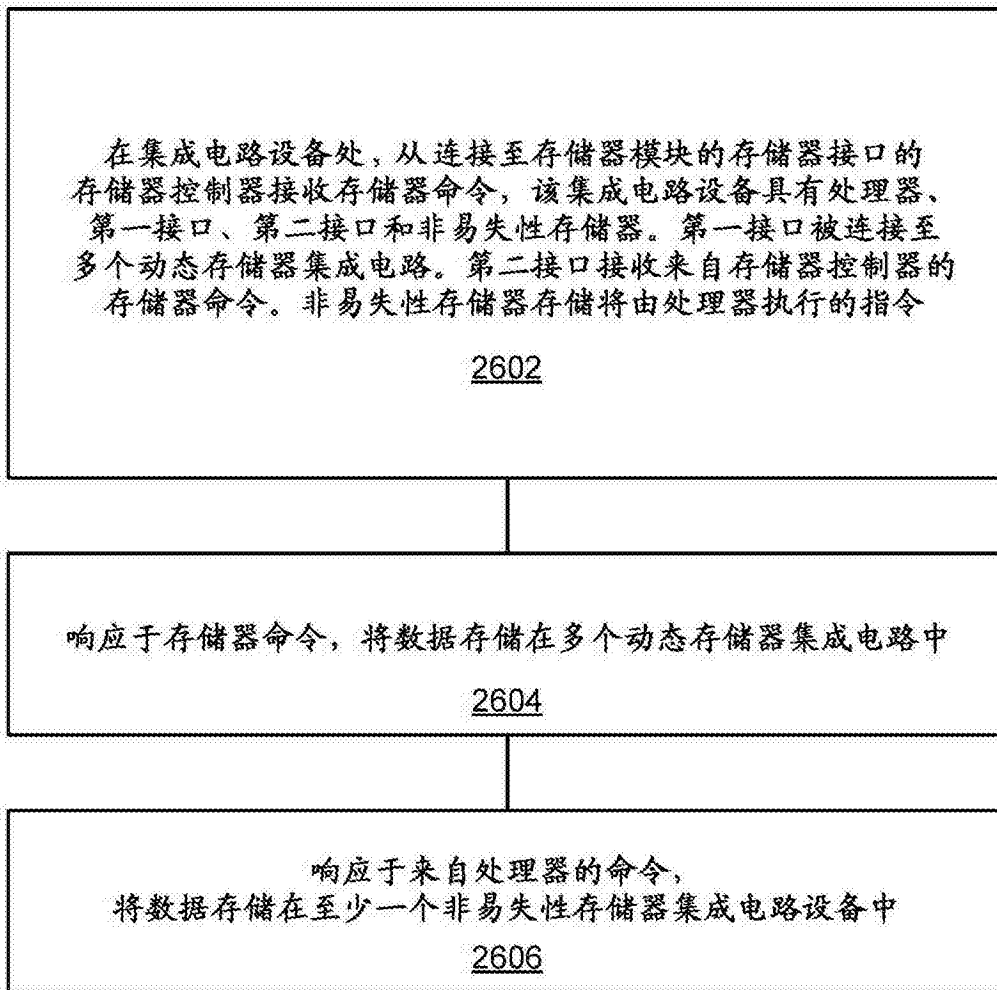


图 26

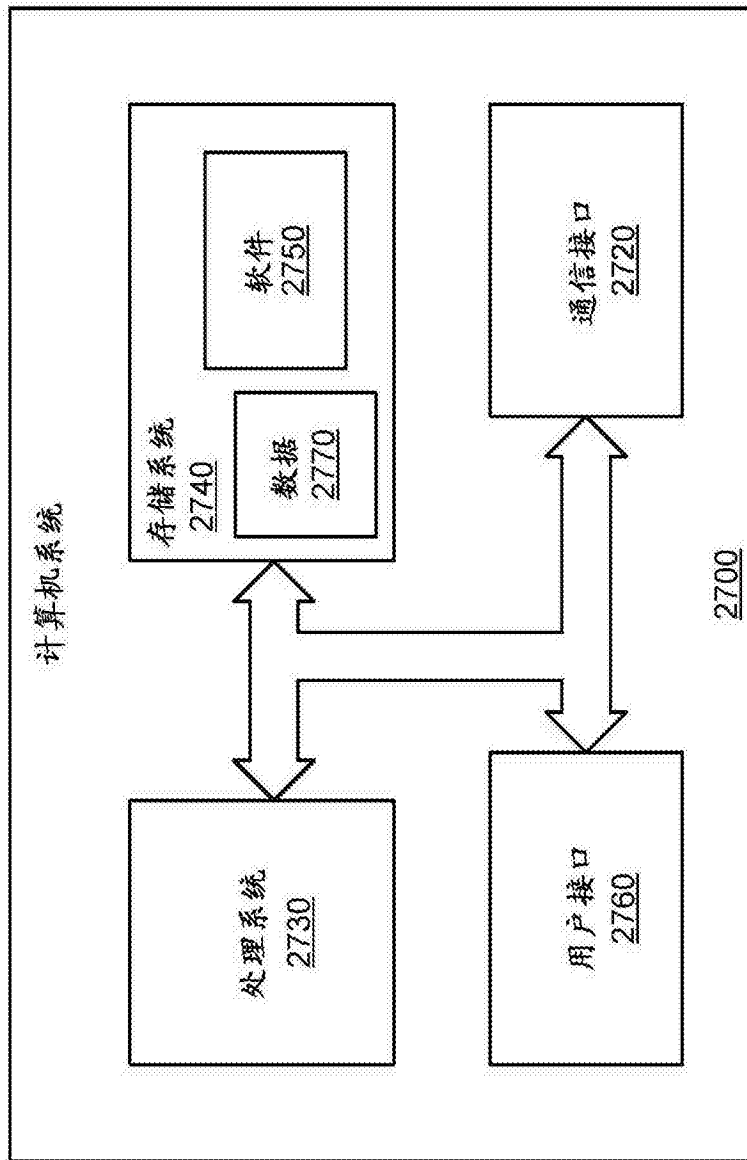


图 27