



[12] 发明专利申请公开说明书

[21] 申请号 03816773.5

[43] 公开日 2005年9月14日

[11] 公开号 CN 1669011A

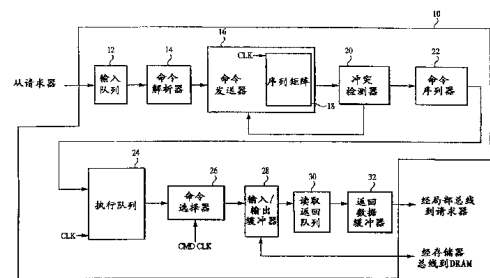
[22] 申请日 2003.5.14 [21] 申请号 03816773.5
 [30] 优先权
 [32] 2002.5.14 [33] US [31] 10/143,896
 [86] 国际申请 PCT/US2003/015184 2003.5.14
 [87] 国际公布 WO2003/098392 英 2003.11.27
 [85] 进入国家阶段日期 2005.1.14
 [71] 申请人 微米技术有限公司
 地址 美国爱达荷州
 [72] 发明人 J·M·耶德洛

[74] 专利代理机构 中国专利代理(香港)有限公司
 代理人 邹光新 王勇

权利要求书9页 说明书10页 附图7页

[54] 发明名称 无序的动态随机存取存储器序列器
 [57] 摘要

在存储控制器的存储器请求队列中顺序地接收存储器存取请求。检测在按所接收的顺序执行存储器存取请求时将发生的在暂时最邻近的请求之间的任何冲突或潜在延迟,以及重新排列该存储器存取请求的接收顺序以避免或最小化冲突或延迟,以及优化到存储器数据总线的的数据流或来自存储器数据总线的的数据流。按重新排序的顺序执行存储器存取请求,同时跟踪请求的原始接收顺序。在执行后,按原始接收读取请求的顺序将通过执行读取型存储器存取请求从存储器设备中读取的数据传送到各个请求器。



1. 一种存储控制器，包括：

输入队列，用于按原始接收的顺序从至少一个请求器中接收多个存储器存取请求；

5 序列矩阵，在其中排列多个存储器存取请求，其中所述序列矩阵按排列顺序存储多个存储器存取请求；

命令发送器，用于排列所述序列矩阵中的多个存储器存取请求的时标序列；

10 冲突检测器，用于检测在通过所述命令发送器在所述序列矩阵中排列的存储器存取请求序列中是否出现冲突或延迟；

执行队列，按不同于排列顺序的次序排列存储器存取请求以便执行，从而避免或降低在所述检测器中检测的冲突或延迟；

15 读取返回队列，用于在执行来自所述执行队列的存储器存取请求后，记录存储器存取请求的原始接收顺序，以及用于按存储器存取请求的原始接收顺序将请求数据返回到至少一个请求器；以及

返回数据缓冲器，用于保存从存储器中读取的请求数据直到将该数据返回到相关联的存取请求的请求器为止。

2. 如权利要求1所述的存储控制器，进一步包括：

20 命令解析器，用于将存储器存取请求至少解析成读取或写入命令，以及数据控制命令；

命令序列器，用于排列用于在所述执行队列中的每个存取请求的数据控制命令以及读取或写入命令。

3. 如权利要求2所述的存储控制器，进一步包括命令选择器，用于一次从所述执行队列中选择用于执行的至少一个命令。

25 4. 如权利要求3所述的存储控制器，其特征在于，一次选择的命令数量取决于输送到所述命令选择器的命令选择时钟信号。

5. 如权利要求1所述的存储控制器，其特征在于，所述检测器能检测顺序执行的存储器存取请求中的潜在的存储体忙状况。

30 6. 如权利要求1所述的存储控制器，其特征在于，所述检测器能检测顺序执行的存储器存取请求中的潜在的页冲突状况。

7. 如权利要求1所述的存储控制器，其特征在于，所述检测器能检

测顺序执行的存储器存取请求中的潜在的页丢失状况。

8. 一种存储控制器，包括：

命令发送器，用于从多个请求器中接收多个存储器存取请求信号，所述命令发送器按对应于从所述请求器中接收请求的次序接收存储器存取请求信号，其中存储器存取请求信号包括读取请求和/或写入请求；

重新排列单元，用于基于所检测的在序列中的暂时最相邻的请求之间的存储器存取冲突或延迟，重新排列存储器存取请求信号的顺序，由此将按重新排列的顺序完成存储器存取请求的执行；

10 执行单元，用于按重新排列的顺序执行存储器存取请求；以及

重新排序单元，用于将执行的读取请求重新排序成原始接收该请求的顺序，由此按重新排序的顺序将通过执行读取请求获得的数据返回到所述请求器。

9. 如权利要求8所述的存储控制器，其特征在于，所述重新排列单元重新排列存储器存取请求的序列，以避免在序列中暂时最邻近请求之间的所检测的存储体忙状况。

10. 如权利要求8所述的存储控制器，其特征在于，所述重新排列单元重新排列存储器存取请求的序列，以使得在序列中暂时最邻近请求之间的所检测的存储体忙状况期间由等待存储体变为可用于存取所产生的延迟最小化。

11. 如权利要求8所述的存储控制器，其特征在于，所述重新排列单元重新排列存储器存取请求的序列，以避免序列中暂时最邻近请求之间的所检测的页冲突状况。

12. 如权利要求8所述的存储控制器，其特征在于，所述重新排列单元重新排列存储器存取请求的序列，以使得在序列中的暂时最邻近请求之间的所检测的页冲突状况期间由等待准备存取存储体所产生的延迟最小化。

13. 如权利要求8所述的存储控制器，其特征在于，所述重新排列单元重新排列存储器存取请求的序列，以使得在序列中的暂时最邻近请求之间的所检测的页丢失状况期间由等待准备存取存储体所产生的延迟最小化。

14. 一种存储控制器，包括

命令发送器，用于从多个请求器中接收多个存储器存取请求信号，所述命令发送器按对应于从所述请求器中接收请求的秩序的次序接收存储器存取请求信号，其中存储器存取请求信号包括读取请求和/或写入请求；

5 重新排列单元，用于重新排列存储器存取请求信号的顺序，以便通过最小化存储器数据总线的空闲时间，使得用来将数据传送到存储器结构或传送来自存储器结构的数据的存储器数据总线的使用最大化，由此将按重新排列的顺序完成存储器存取请求的执行；

执行单元，用于按重新排列的顺序执行存储器存取请求；以及

10 重新排序单元，用于将执行的读取请求重新排序成原始接收该请求的顺序，由此按重新排序的顺序将通过执行读取请求获得的数据返回到所述请求器。

15. 一种存储控制器，包括：

输入电路，用于按到来顺序存储多个存储器存取请求；

15 存储器存取请求重新排列电路，用于将所存储的存储器存取请求重新排列成存储器存取请求执行顺序，所述重新排列电路基于按所存储的存储器存取请求的顺序检测到存在的存储器存取冲突和存储器存取延迟的至少一个，将所存储的存储器存取请求的顺序重新排列成执行顺序；以及

20 发送电路，用于按执行顺序输出存储器存取请求。

16. 如权利要求 15 所述的存储控制器，其特征在于，所述重新排列电路基于检测到的存储体忙冲突，重新排列所存储的存储器存取请求。

17. 如权利要求 16 所述的存储控制器，其特征在于，所述重新排列电路重新排列所存储的存储器存取请求，以避免所检测的存储体忙冲突。

18. 如权利要求 16 所述的存储控制器，其特征在于，所述重新排列电路重新排列所存储的存储器存取请求，以便使得在按所存储的存储器存取请求顺序执行存储器存取请求时，在检测的存储体忙冲突期间由等待存储体变为可用于存取所产生的延迟最小化。

19. 如权利要求 15 所述的存储控制器，其特征在于，所述重新排列电路基于按所存储的存储器存取请求的顺序检测的页冲突状况，重新排列所存储的存储器存取请求。

20. 如权利要求 19 所述的存储控制器，其特征在于，所述重新排列电路重新排列所存储的存储器存取请求，以避免所检测的页冲突。

21. 如权利要求 19 所述的存储控制器，其特征在于，所述重新排列电路重新排列所存储的存储器存取请求，以便使得在按所存储的存储器存取请求的顺序检测的页冲突状况期间由等待准备存取存储体所产生的延迟最小化。

22. 如权利要求 15 所述的存储控制器，其特征在于，所述重新排列电路基于按所存储的存储器存取请求的顺序检测的页丢失状况，重新排列所存储的存储器存取请求。

23. 如权利要求 22 所述的存储控制器，其特征在于，所述重新排列电路重新排列所存储的存储器存取请求，以便使得在按所存储的存储器存取请求的顺序检测的页丢失状况期间由等待准备存取存储体所产生的延迟最小化。

24. 如权利要求 15 所述的存储控制器，进一步包括：
执行电路，用于按执行顺序执行所存储的存储器存取请求；以及
跟踪电路，用于在执行后，记录所存储的存储器存取请求的到来顺序。

25. 如权利要求 24 所述的存储控制器，其特征在于，所述跟踪电路将标记与对应于在到来顺序中其各自位置的每个存储器存取请求相关联。

26. 如权利要求 24 所述的存储控制器，其特征在于，所述跟踪电路包括缓冲器，用于在执行存储器存取请求时组织从各个存储单元中读取的数据，由此排列在该缓冲器中的数据能按原始到来顺序从所述存储控制器中被传送出来。

27. 一种用于控制存储器存取操作的方法，包括：
按所接收的顺序接收多个存储器存取请求；
检测在序列中暂时最邻近请求之间的存储器存取冲突和延迟中的至少一个；
基于检测结果，重新排列请求的顺序；以及
按重新排列的顺序，执行存储器存取请求。

28. 如权利要求 27 所述的方法，其特征在于，所述存储器存取请求包括将数据写入指定的存储器地址的写入请求以及从指定的存储器地址

读出数据的读取请求。

29. 如权利要求 28 所述的方法, 进一步包括按接收序列中的原始顺序, 从所述存储控制器中输出通过执行读取请求从指定的存储器地址中获得的数据。

5 30. 如权利要求 28 所述的方法, 进一步包括:

在执行后, 记录存储器存取请求的接收顺序; 以及

按重新排列的顺序, 将通过执行读取操作获得的请求数据传送到各个请求器, 其中按对应于各个读取请求的接收顺序的次序将所请求的数据传送到各个请求器。

10 31. 如权利要求 30 所述的方法, 其特征在於, 记录所接收的顺序包括在重新排列请求顺序之后将标记与每个读取请求相关联以便表示原始接收读取请求的顺序。

32. 如权利要求 30 所述的方法, 其特征在於, 记录所接收的顺序包括在重新排列请求顺序之后指定将存储在执行每个各自的读取请求后获得的数据的缓冲区单元, 其中所指定的缓冲区单元表示原始接收读取请求的顺序。

33. 如权利要求 27 所述的方法, 其特征在於, 执行检测步骤以便检测在按接收顺序的当前位置中执行最新接收的存储器存取请求时是否产生冲突。

20 34. 如权利要求 33 所述的方法, 其特征在於, 执行检测步骤以便检测在按接收顺序的当前位置中执行最新接收的存储器存取请求时是否产生存储体忙冲突。

35. 如权利要求 34 所述的方法, 其特征在於, 执行重新排列步骤以避免存储体忙冲突。

25 36. 如权利要求 34 所述的方法, 其特征在於, 执行重新排列步骤以便使得在所检测的存储体忙冲突期间由等待存储体变为可用于存取所产生的延迟最小化。

37. 如权利要求 33 所述的方法, 其特征在於, 执行检测步骤以便检测在执行在所接收的序列的当前位置中的最近接收的存储器存取请求时是否产生页冲突。

30 38. 如权利要求 37 所述的方法, 其特征在於, 执行重新排列步骤以避免页冲突。

39. 如权利要求 37 所述的方法, 其特征在于, 执行重新排列步骤以便使得在检测页冲突期间由等待准备存取的存储体所产生的延迟最小化。

5 40. 如权利要求 27 所述的方法, 其特征在于, 执行检测步骤以便检测在执行在所接收的序列的当前位置中的最近接收的存储器存取请求时是否产生延迟。

41. 如权利要求 40 所述的方法, 其特征在于, 执行检测步骤以便检测在执行在所接收的序列的当前位置中的最近接收的存储器存取请求时是否产生页丢失。

10 42. 如权利要求 41 所述的方法, 其特征在于, 执行重新排列步骤以便使得在检测页丢失期间由等待准备存取的存储体所产生的延迟最小化。

43. 如权利要求 27 所述的方法, 其特征在于, 执行检测步骤以便检测在执行在所接收的序列的当前位置中的最早接收的存储器存取请求时是否产生冲突。

15 44. 如权利要求 43 所述的方法, 其特征在于, 执行检测步骤以便检测在执行在所接收的序列的当前位置中的最早接收的存储器存取请求时是否产生存储体忙冲突。

20 45. 如权利要求 44 所述的方法, 其特征在于, 执行重新排列步骤以避免存储体忙冲突。

46. 如权利要求 44 所述的方法, 其特征在于, 执行重新排列步骤以便使得在检测存储体忙冲突期间由等待存储体变为可用于存取所产生的延迟最小化。

25 47. 如权利要求 43 所述的方法, 其特征在于, 执行检测步骤以便检测在执行在所接收的序列的当前位置中的最早接收的存储器存取请求时是否产生页冲突。

48. 如权利要求 47 所述的方法, 其特征在于, 执行重新排列步骤以避免页冲突。

30 49. 如权利要求 47 所述的方法, 其特征在于, 执行重新排列步骤以便使得在所检测的页冲突期间由等待准备存取的存储体所产生的延迟最小化。

50. 如权利要求 43 所述的方法, 其特征在于, 如果检测到和所接收

的序列中的最早接收的存储器存取请求有关的冲突，则重新排列步骤包括将该最早接收的存储器存取请求移动到通常和序列中的最新接收的存储器存取请求后的位置相对应的序列的末端。

51. 如权利要求 27 所述的方法，其特征在于，执行检测步骤以便检测在执行在所接收的序列的当前位置中的最早接收的存储器存取请求时是否产生延迟。

52. 如权利要求 51 所述的方法，其特征在于，执行检测步骤以便检测在执行在所接收的序列的当前位置中的最早接收的存储器存取请求时是否产生页丢失。

53. 如权利要求 52 所述的方法，其特征在于，执行重新排列步骤以便使得在检测页丢失期间由等待准备存取的存储体所产生的延迟最小化。

54. 如权利要求 51 所述的方法，其特征在于，如果检测到和所接收的序列中的最早接收的存储器存取请求有关的冲突，则重新排列步骤包括将该最早接收的存储器存取请求移动到通常和序列中的最新接收的存储器存取请求后的位置相对应的序列的末端。

55. 如权利要求 27 所述的方法，进一步包括步骤：

将每个存储器存取请求至少解析成读取或写入命令以及数据控制命令；

在重新排列请求序列的步骤中，单独地排列用于所重新排列的序列中的每个请求的读取或写入命令以及数据控制命令；以及从重新排列的序列中一次选择至少一个命令以便执行。

56. 一种用于控制存储器存取操作的方法，包括：

按接收顺序接收包括读取请求和写入请求的多个存储器存取请求；重新排列存储器存取请求的顺序以便最大化存储器数据总线的利用；

按重新排列的顺序执行存储器存取请求。

57. 如权利要求 56 所述的方法，进一步包括步骤：

在按重新排列的顺序执行读取请求时，根据各个读取请求的原始接收顺序，排列通过执行读取请求所读取的数据；以及按排列顺序从所述存储控制器中输出读取数据。

58. 一种用于控制存储器存取操作的方法，包括：

按接收顺序接收包括读取请求和写入请求的多个存储器存取请求；
重新排列存储器存取请求的顺序以便最小化存储器存取冲突和延
迟；

按重新排列的顺序执行存储器存取请求。

5 59. 如权利要求 58 所述的方法，进一步包括步骤：

在按重新排列的顺序执行读取请求时，根据各个读取请求的原始接
收顺序，排列通过执行读取请求所读取的数据；以及
按排列顺序从所述存储控制器中输出读取数据。

60. 一种处理器系统，包括：

10 处理器；

至少一个存储器设备，用于与所述处理器交换数据；以及

存储控制器，用于管理从所述处理器到所述至少一个存储器设备的
存储器存取请求，所述存储控制器包括：

15 命令发送器，用于从多个请求器中接收多个存储器存取请求信
号，所述命令发送器按对应于从所述请求器中接收请求的顺序的次序
接收存储器存取请求信号，其中存储器存取请求信号包括读取请求和
/或写入请求；

重新排列单元，用于基于所检测的在序列中的暂时最相邻的请求
之间的存储器存取冲突或延迟，重新排列存储器存取请求信号的顺
序，由此将按重新排列的顺序完成存储器存取请求的执行；

20 执行单元，用于按重新排列的顺序执行存储器存取请求；以及

重新排序单元，用于将执行的读取请求重新排序成原始接收该请
求的顺序，由此按重新排序的顺序将通过执行读取请求获得的数据返
回到所述请求器。

25 61. 一种处理器系统，包括：

处理器；

至少一个存储器设备，用于与所述处理器交换数据；以及

存储控制器，用于管理从所述处理器到所述至少一个存储器设备的
存储器存取请求，所述存储控制器包括：

30 命令发送器，用于从多个请求器中接收多个存储器存取请求信
号，所述命令发送器按对应于从所述请求器中接收请求的顺序的次序
接收存储器存取请求信号，其中存储器存取请求信号包括读取请求和

/或写入请求;

- 重新排列单元,用于重新排列存储器存取请求信号的顺序,以便通过最小化存储器数据总线的空闲时间,使得用来将数据传送到存储器结构或传送来自存储器结构的数据的存储器数据总线的使用最大化,由此将按重新排列的顺序完成存储器存取请求的执行;
- 5

执行单元,用于按重新排列的顺序执行存储器存取请求;以及
重新排序单元,用于将执行的读取请求重新排序成原始接收请求的顺序,由此按重新排序的顺序将通过执行读取请求获得的数据返回到所述请求器。

无序的动态随机存取存储器序列器

技术领域

5 本发明涉及用于控制存储器存取操作以便实现增加有效存储器带宽的存储控制器的体系结构和操作方法。

背景技术

在大多数计算机或数据处理系统中，主有源存储器或随机存取存储器（RAM）均是动态随机存取存储器（DRAM）。DRAM的结构通常由组织
10 成多个存储体的多个存储单元组成。每个存储体对应于存储单元阵列，每个存储单元分别与唯一的存储地址有关联。特别地，存储体内的存储地址分别用行地址和列地址表示，其中每个行地址定义为存储页。因此，存储器的每页包含对应于该页内的不同列表示的多个存储单元。

当执行一系列存取请求时，如果在当前具有打开的另一页的相同存储体中找到当前请求的页，将这种状况称为“页冲突”，因此，必须首先关闭，或“预充电”先前打开的页。在预充电后，然后打开，或“激活”所请求的页，然后执行读取或写入操作。如果在不具有打开页的存储体中找到当前请求的页，出现“页丢失”，因此，执行要求激活过程。
15 当当前存储器存取请求是由于先前存储器存取请求而已经打开的页时，
20 则出现所谓“页命中”。

由于相对于页命中请求，对页冲突和页丢失存储器存取必须执行额外的处理，所以执行前者的两个处理所需的时间显著大于后者。在微型处理器技术发展的早期阶段，在先入、先出的基础上，接收和履行用于读取和写入操作的存取DRAM存储器页的请求。这种处理趋向于非常低
25 效，导致大量页丢失和冲突，因此，要求大量的专用处理器和/或存储控制器资源来预充电和激活存储页。

最近，已经开发了存储器存取基于优先级的更先进的处理方法。存取请求的优先级可以基于各种因素，诸如发送请求的设备的类型、所请求的存取类型，通过该请求存取所需的存储器地址等等。然而，严格地
30 根据优先级提供存储器存取的问题在于在不可接受的长时间周期中，会拒绝访问低优先级请求。

此外，随着每个新一代计算机发展，显著地增加存储器时钟的速度。

随着存储器时钟的速度增加，页丢失存储器操作、存储体忙冲突和其他冲突的潜在出现及其损失的时间也变得日益显著。特别地，在预充电、激活、等待存储体可用性等期间，用来将信息传送到每个被访问的存储单元和从每个被访问的存储单元传送信息的数据总线是空闲的。

- 5 因此，需要一种解决方案来降低上述缺陷。特别地，如果重新排列所接收的存储器存取请求序列的顺序以避免或减小冲突，将大大地提高存储器处理效率。通过避免或减小冲突，在减小或消除存储器数据总线的空闲时间中，更有效地利用存储器数据总线，因此，有效地增加存储器系统的存储器带宽，以及允许在比先前尽可能的更短时间量中，出现
10 更多存储器存取事务。

发明内容

本发明试图通过重新排列顺序接收的 DRAM 存取请求的顺序以最小化冲突和延伸，诸如上述，同时按与初始接收请求相同的顺序将所请求的信息返回到请求单元来解决在现有技术中识别的问题。

- 15 在本发明中，在存储控制器的输入队列中，接连地接收存储器存取请求。在请求输入队列后，排列序列矩阵，由此，由冲突检测器识别连续请求之间的冲突或潜在延迟。冲突检测器重新排序存储器核心存取请求以便优化到数据总线和来自数据总线的数据流。例如，如果在顺序接收的存储器请求中，由冲突检测器识别出存储体忙状况或其它延迟，存储
20 控制器重新排列将执行未决的存储器请求的顺序以便如果可能的话消除冲突或延迟，或者最小化延迟。

只要最早请求没有寻址冲突，就能够无序执行写入请求。

- 重新排序的序列保留在执行队列中，其中，标记每个请求以表示在原始序列中的位置，使得在存储控制器中适当地重新排序返回数据以匹
25 配输入的存储器存取请求的顺序。

命令选择器选择将由执行队列执行的一个命令或多个命令。命令选择器包含允许恒定速度 DRAM 序列器与多个时钟速度接口而没有复杂的时钟相位调整操作的接口定时特性。

- 读取返回队列跟踪从 DRAM 获得的返回数据。基于与每个返回数据有
30 关联的标记，使读取返回队列按原始顺序将读取数据返回到各自的请求器。特别地，在执行读取请求时，如果与返回数据有关联的标记对应于存储控制器中具有最长等待时间的读取请求，将返回数据返回到请求数

据的系统单元。如果标记不与具有最长等待时间的读取请求有关联，则使返回数据存储在缓冲器中直到用于具有最长等待时间的所有读取请求的返回数据返回到它们各自的请求器为止。

另外，基于所接收的请求序列，能为每个请求指定缓冲区单元。当从 DRAM 获得的返回数据放入对应于输入队列中最早接收的读取请求的缓冲区单元时，将那个数据返回到适当的请求器。否则，所返回的数据保留在缓冲器中，直到执行在输入队列中所接收的所有先前读取的请求为止。用这种方式，按原始接收它们的顺序，读取返回队列返回来自缓冲区单元的数据。

10 附图说明

从下述结合附图所述的本发明的优先实施例的详细描述，本发明的另外的特征、方面和优点将变得显而易见，其中：

图 1 是根据本发明的存储控制器的框图。

图 2 是示例说明用于按原始接收的请求顺序，将通过重新排序的读取请求执行获得的请求数据返回到它们各自的请求器的过程的流程图。

图 3 是在检测到当前定时矩阵中的冲突或延伸时，由存储控制器中的顺序加载逻辑单元的冲突重新排序过程的第一实施例。

图 4 是在检测到当前定时矩阵中的冲突或延伸时，由存储控制器中的顺序加载逻辑单元的冲突重新排序过程的第二实施例。

图 5 是在检测到当前定时矩阵中的冲突或延伸时，由存储控制器中的顺序加载逻辑单元的冲突重新排序过程的第三实施例。

图 6 是用于示例说明本发明的操作的时序图。

图 7 是可以利用本发明的存储控制器的处理系统的框图。

具体实施方式

为便于描述，结合动态随机存取存储器 (DRAM) 设备，论述本发明的优选实施例。不过，应注意到本发明不限于包含 DRAM 的应用。相反，应该强调，可以结合其他类型的随机存取存储器，诸如静态 RAM (SRAM) 和许多不同次种类的 DRAM，包括例如快页模式 DRAM (FPM DRAM)、扩展数据输出 DRAM (EDO DRAM)、突发 EDO DRAM、同步 DRAM (SDRAM)、双倍数据速率 DRAM (DDR DRAM)、Rambus DRAM (RDRAM) 等等，使用本发明的存储控制器和方法。

图 1 表示根据本发明的存储控制器 10，以及包括输入队列 12、命令

解析器 14、命令发送器 16、序列矩阵 18、冲突检测器 20、命令序列器 22、执行队列 24、命令选择器 26、输入/输出缓冲器 28、读取返回队列 30 和返回数据缓冲器 32。

存储器存取请求进入存储控制器 10 以及根据先入先出被接收在输入队列 12 中。然后，通过命令解析器 14 顺序地处理所接收的请求以便从每个请求信号中获得相关信息，诸如存储器地址 (MA) 数据、片选 (CS) 命令 (指示将存取的请求存储体)、行地址选择 (RAS) 命令和列地址选择 (CAS) 命令，以及用于表示该请求是读取还是写入操作的允许写入 (WE) 状态。

然后，将所获得的信息提供到命令发送器 16，其根据在序列矩阵 18 中接收的时钟信号将所接收的存储器存取请求放入序列矩阵 18 中。注意在启动该处理后，按在命令发送器 16 中接收它们的顺序，将存取请求加载到该矩阵中。

冲突检测器 20 监视序列矩阵 18 中的信息以及校验在按序列矩阵中的当前顺序执行矩阵中的请求序列时可能会发生的任何冲突或延迟。将由冲突检测器 20 检测到的任何冲突或延迟报告回到命令发送器 16，然后，重新排列矩阵中的请求顺序以便最小化或消除由于所识别出的冲突或延迟而使存储器数据总线为空闲的时间。可以由冲突检测器 20 检测到的冲突例如包括但不限于页冲突和存储体忙状况，其中存储体忙于执行另一读取或写入操作。由冲突检测器识别的延迟是在存储器数据总线为空闲的时间期间，要求执行准备步骤的状况，但不一定与执行另一存储器存取请求冲突。

通常，可以在序列矩阵 18 中重新排序读取和写入型存储器存取请求。然而，如果有必要的话，最好在重新排序的序列不会产生与进入输入队列中的先前接收的请求的任何寻址冲突或者干扰存储在或将存储在与矩阵中的任何其他存储器存取请求有关的相关存储单元中的数据的范围上，仅重新排列写入请求。

当另外的存储器存取请求移入序列矩阵 18 时，同时也通过命令发送器 16 重新排列以便解决冲突或者减少延迟，将在序列前面的请求移入用作执行重新排列的请求等待的发送缓冲器的执行队列 24。根据用在命令发送器 16 中的冲突判决处理，按成批多个存取请求，根据时钟信息在连续基础上，或基于请求的预定累积大小，将请求移入执行队列 24 中。最

好，用于执行队列 24 的时钟 (CLK) 是与引导加载序列矩阵 18 相同的时钟 (CLK)。

命令序列器 22 排列与将从序列矩阵 18 中传送到执行队列 24 的每个请求有关联的各种命令，当有必要插入数据控制命令和/或临时将数据控制命令与特定的存储器存取请求的读取或写入命令分开。例如，如果发送到执行队列 24 的读取或写入请求要求预充电和/或激活操作，在相关读取或写入操作前，将初始化预充电和/或激活操作的数据控制命令放入执行队列中，至少一个其它的读取或写入命令与位于预充电和/或激活命令和相关联的读取或写入命令之间的不同存取请求有关。

10 通过用这种方式将读取或写入命令与这些数据控制命令分开来实现本发明的优点。具体地，在上述例子中，能执行上述例子中的预充电和/或激活操作，同时能立即执行用于另一存储器存取请求的读取或写入命令。因此，在执行预充电和/或激活操作期间，存储器数据总线不必空闲。

15 当将每个读取请求传送到执行队列时，将标记临时增加到用于那个请求的数据控制命令，用于识别如在输入队列 12 中接收的每个请求的最初相对位置。另外，为每个读取请求指定读取缓冲器 32 中的各个缓冲区单元，这将在下面更详细地描述。

20 根据输入到命令选择器 26 中的命令选择时钟 (CMD CLK) 信号，选择来自执行队列 24 的前面的一个或多个存储存取请求，以便在适当的 DRAM 存储体中执行。例如，如果命令选择时钟信号是将请求加载到序列矩阵 18 中的时钟速度的四倍速度，那么对队列 24 的每个时钟信号，从执行队列 24 中移出四个存取请求。在这种情况下，命令选择器 26 调节请求选择过程，使得以平均间隔执行从队列 24 中一次选择的四个存取请求。如果从执行队列中选择命令用于以和新请求进入序列矩阵 18 相同的时钟速度执行，那么从存储控制器 10 中省略命令选择器 26。

I/O 缓冲器 28 是在每个存取请求中指定的读取或写入操作期间使用的过渡缓冲器。如果将执行的当前存取请求是写入操作，将写入所选择的存储单元中的数据临时写入到 I/O 缓冲器 28 中。类似地，在读取操作中，从所选择的存储单元中读取的数据临时存储在 I/O 缓冲器 28 中。

30 在执行所请求的存储器存取时，将每个读取请求放入到读取返回队列 30 中。读取返回队列 30 在执行读取请求时，管理从 DRAM 中读取的请求数据，以及按在它们被接收在输入队列中的顺序将所请求的数据返回

到各个请求器。根据与返回的请求数据有关联的标签或指定的缓冲区单元，从 DRAM 中读取的数据被直接传送到请求器或者被放入到返回数据缓冲器 32 中。

5 现在参考图 2，在执行读取请求后，将所执行的读取请求以及通过暂时保存在 I/O 缓冲器 28 中的请求所获得的数据返回到读取请求队列 30 中（步骤 100）。如果读取请求队列 30 在步骤 110 确定返回读取请求与最当前标记或指定的缓冲区单元有关联，在步骤 120 将通过那个请求获得的数据返回到请求器，以及在步骤 S130，更新当前标记/缓冲区单元信息。

10 标记或缓冲区单元在其被指定到存储控制器 10 中具有最长等待时间的读取请求或者与其有关时则为“当前”。读取返回队列 30 可以记录最当前标记或缓冲区单元，例如，通过每次将返回数据传送到其请求器时使表示当前标记或缓冲区单元的计数值递增。

15 如果返回读取请求具有不是当前的标记或指定的缓冲区单元，则将数据放入读取数据缓冲器 32 中（步骤 140），直到相关标记或缓冲区单元变成当前为止。在步骤 140 中，将返回数据放入缓冲器 32 中后，或在步骤 130 更新当前标记或缓冲区单元后，读取返回队列 30 确定是否能在返回数据缓冲器 32 中找到对应于当前标记或缓冲区单元的返回数据。如果“是”，处理返回到步骤 120，其中将当前返回数据传送到那个数据的请求器，以及在步骤 130 中再次更新当前标记或缓冲区单元。如果在缓冲器 32 中未找出对应于当前标记或缓冲区单元的返回数据，该处理返回到步骤 100 以便接收通过执行该执行队列中的下一个读取请求所获得的返回数据。由于图 2 中所示的这一过程，按请求被接收在输入队列 12 中的顺序将读取数据返回到各个请求器。

25 如果使用标记来表示读取请求的初始接收顺序，在每个请求的数据控制命令中临时插入标记，但不包括在返回到请求器的读取数据中。最好，根据对应于 DRAM 的时钟速度或其整数倍的时钟速度执行读取返回队列 30 和返回数据缓冲器 32 的功能。同样地，能分别通过更新当前标记/缓冲区单元和传送返回数据，协调在读取返回队列 30 和返回数据缓冲器 32 中接收的返回读取请求和返回数据。

30 将参考图 3 所示的流程图，描述由于所检测的冲突或延迟，可以使用命令发送器 16 重新安排存储器存取请求的冲突重新排序处理的第一实

施例。在步骤 200，将解析的最后一个存储器存取请求放入到序列矩阵 18 中。在步骤 210，校验存取所需的存储体的状态以便确定如果在序列矩阵 18 中的当前位置执行新到达的存储器存取请求，是否发生任何冲突或延迟。

5 如果命令发送器 16 根据新到来的请求检测冲突或延迟，命令发送器 16 在步骤 240 确定在其当前位置前在存储器存取请求的序列中是否能找到更适当的定时位置。特别地，命令发送器 16 在新存取请求前首先确定在序列中是否存在未解决的冲突或延迟，以及如果是的话，在没有冲突的时间期间能否执行新存取请求。如果没有未决的冲突或延迟，命令发送器 16 校验在矩阵的任何点处是否可以重新安排新存取请求，而在先前
10 安排的请求中不引起任何新的冲突或延迟。如果不是，过程被重新引到步骤 230，其中在具有未解决状况的当前定时序列的末端处留下新存取请求。如果在序列中提前发现适当的定时位置，将新到来的请求插入序列中的那个位置处（步骤 250）。然后，该过程返回到步骤 200 以便重复
15 下一个输入存储器存取请求。

另一方面，如果相对于新到来的存取请求没有检测到冲突或延迟，则冲突检测器 20 在步骤 220 接着确定在新到来的请求的当前位置前，在定时矩阵中是否存在任何未解决的冲突或延迟。如果找到未解决的冲突或延迟，处理被重引到如上所述的步骤 240。如果未找到现有冲突，命令
20 发送器 16 在步骤 230 使该请求处于序列矩阵 18 中的其当前位置，然后返回到步骤 200 以便重复用于下一个输入存储器存取请求的处理。

在该实施例中，随着命令发送器 16 的后来重复，在新存取请求到来的情况中，可以或不可以稍后解决任何未解决的冲突或延迟。如果没有适当的请求到来以降低冲突或延迟，则存储器存取请求将继续按指定的
25 顺序被处理，但由于未解决的冲突或未降低的延迟，在利用存储器数据总线方面有些低效。

在图 4 所示的流程图中，示例说明由命令发送器 16 执行的冲突重新排序过程的第二实施例。当另一存储器存取请求移出序列矩阵 18 到执行
30 排队 24 时，在步骤 300，使序列矩阵中下一行中的存取请求移入该序列矩阵中的第一单元。在步骤 310，冲突检测器 20 确定关于在序列矩阵头处的存取请求（即，矩阵请求中的最早的一个）是否存在冲突或延迟。如果未找到冲突，则在步骤 320，存取请求传送到执行队列 24 上，以及

该过程返回到步骤 300。

另一方面，如果在步骤 310 中检测到冲突或延迟，冲突检测器 20 将其注意力转变到序列矩阵中的下一个存取请求，以及在步骤 330 确定在没有任何冲突的时间位置处是否能执行那个请求。如果通过在那个时间位置调整存取请求，没有产生冲突或延迟，则在步骤 320 将该请求发送到执行队列 24，以及该过程返回到步骤 300。如果发现冲突或延迟，则该过程重复步骤 330 直到找到在那个时间位置适合执行的请求为止。

在该实施例中，如果发现存在和序列矩阵头处的请求的存储器存取有关的冲突或延迟，则任何后续请求、拒绝的请求仍然保持在序列中的它们的当前位置直到找到该请求的适当时隙为止。通过该处理，不将存储器存取发送到具有冲突或延迟状况的执行队列 24。同时，基于等待时间为每个请求指定优先级，以及在其中通过那个请求的定时未产生冲突状况的最早可能时隙执行每个请求。

参考图 4 所示的实施例的变形如图 5 所示，其中如果发现和序列矩阵头处的存取请求的当前定时位置有关的冲突或延迟，将不欢迎的请求发送到序列的后面，而不是留在序列中的其当前位置。具体地，如果在步骤 310 发现冲突，执行步骤 420，其中将具有冲突或延迟的请求发送到序列的后面。然后，冲突检测器 20 移动到行中的下一个存取请求以便确定是否发现与那个请求有关的任何冲突或延迟（步骤 430）。如果未检测到冲突或延迟，将该请求发送到执行队列 24，与图 4 所示的过程类似。如果检测到冲突或延迟，该过程返回到步骤 420。

现在，参考图 6，示出了时序图，其示例说明在操作本发明时获得的结果。在该实例中，按所列出的顺序将读取请求 RD_0 、 RD_1 和 RD_2 接收在输入队列中。假定已经检测到关于 RD_0 的页冲突状况，以及 RD_1 和 RD_2 均是其它存储体中的页命中。因为如果允许在 RD_1 和 RD_2 之前完成执行 RD_0 ，则导致延迟，所以重新排列请求以便当正执行 RD_0 的预充电操作时，执行用于 RD_1 和 RD_2 的读取存储器存取。用这种方式重新排列命令使得在等待 RD_0 的预充电操作以完成执行时最小化存储器数据总线空闲的时间，从而导致存储器数据总线的更高效利用。

在从 DRAM 中读取用于 RD_1 和 RD_2 的数据后，执行用于 RD_0 的激活操作，然后执行用于 RD_0 的读取访问命令。按执行读取操作的顺序将读取数据“1111”和“2222”传送到返回数据缓冲器。然而，在从适当的存储

单元中获得“0000”数据后，首先将该数据返回到RD₀的请求器。然后，按此顺序将数据“1111”和“2222”返回到它们各自的请求器。

图7示例说明可以利用本发明的存储控制器10的示例性处理系统900。处理系统900包括连接到局部总线904的一个或多个处理器901。
5 存储控制器10和主总线桥903均连接局部总线904。处理系统900可以包括多个存储控制器10和/或多个主总线桥903。存储控制器10和主总线桥903可以被集成为单个器件906。

存储控制器10还连接到一个或多个存储器数据总线907。每个存储器总线接受包括至少一个存储器设备902的存储元件908。存储元件908
10 形成为存储卡或存储器模块。用在该系统900中的存储器模块的例子包括单内联存储器模块(SIMMs)和双内联存储器模块(DIMMs)。存储元件908可以包括一个或多个附加器件909。例如，在SIMM或DIMM中，该附加器件909可以是配置存储器，诸如连续存在检测(SPD)存储器。

存储控制器10还可以连接到高速缓冲存储器905。高速缓冲存储器905可以仅是处理系统中的高速缓冲存储器。另外，其他的设备，例如处
15 理器901也可以包括高速缓冲存储器，其可以形成具有高速缓冲存储器905的高速缓存体系。如果处理系统900包括为总线主控制器或支持直接存储器存取(DMA)的外设或控制器，存储控制器10可以实现高速缓存相干协议。如果存储控制器10连接到多个存储器总线907，每个存储器
20 总线907可以并行操作，或可以将不同地址范围映射到不同存储器总线907。

主总线桥903连接到至少一个外围总线910。各个设备，诸如外设或另外的总线桥可以连接到外围总线910。这些设备可以包括存储控制器911、其他I/O设备914、次总线桥915、多媒体处理器918和传统设备
25 接口920。主总线桥903还可以连接到一个或多个专用高速端口922。在个人计算机中，例如，专用端口可以是加速图形接口(AGP)，用来将高性能视频卡连接到处理系统900。

存储控制器911经存储总线912将一个或多个存储设备913连接到外围总线910。例如，存储控制器911可以是SCSI控制器以及存储设备
30 913可以是SCSI盘。I/O设备914可以是任何类型的外设。例如，I/O设备914可以是局域网接口，诸如Ethernet卡。次总线桥可以用来经另一总线将另外的设备连接到处理系统。例如，次总线桥可以是通用串行

端口 (USB) 控制器, 用来将 USB 设备 917 连接到处理系统 900。多媒体处理器 918 可以是声卡、视频俘获卡或任何其它类型的媒体接口, 也可以连接到一个另外的设备, 诸如扬声器 919。传统设备接口 920 用来将传统的设备, 例如老式键盘和鼠标连接到处理系统 900。

5 图 7 中所示的处理系统 900 仅是示例性的处理系统, 通过该系统可以使用本发明。尽管图 7 示例说明特别适合于通用计算机诸如个人计算机或工作站的处理体系结构, 应意识到, 能做出众所周知的改进以便将处理系统 900 配置成更适合于用在各种应用中。例如, 使用依赖于连接到存储元件 908 和/或存储设备 902 的 CPU901 的更简单的体系结构, 10 可以实现要求处理的许多电子设备。这些电子设备可以包括, 但不限于音频/视频处理器和记录器、游戏控制台、数字电视机、有线或无线电话、导航设备 (包括基于全球定位系统 (GPS) 和/或惯性导航的系统) 以及数码相机和/或记录器。改进可以包括例如消除不必要的元件, 增加专用设备或电路, 和/或集成多个设备。

15 尽管根据特定的实施例描述了本发明, 许多其他的变形和改进以及其他用途对本领域的技术人员来说是显而易见的。因此, 可取的是, 本发明不是由在此具体公开的内容来限定, 而是由附加权利要求书来限定。

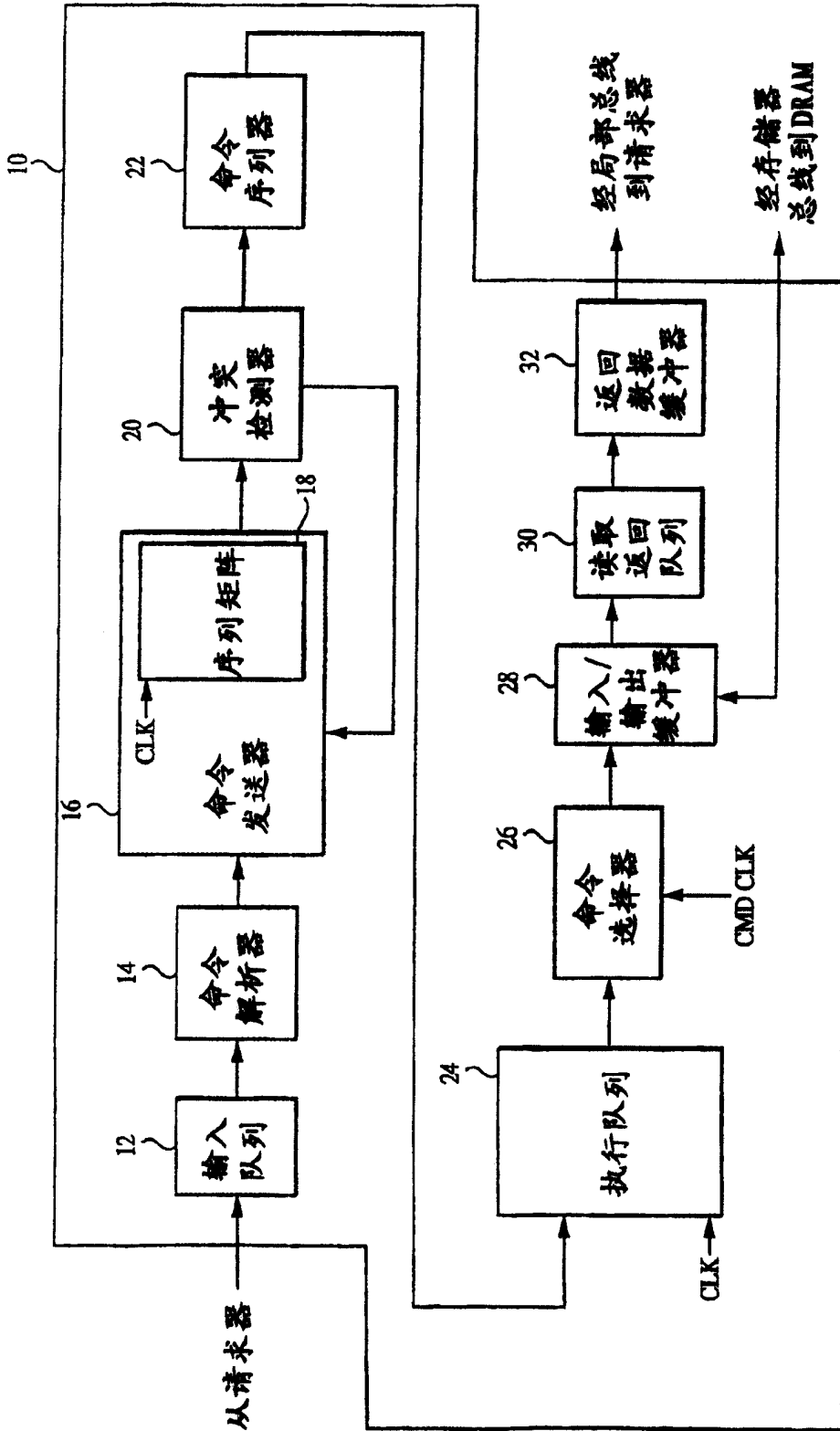


图 1

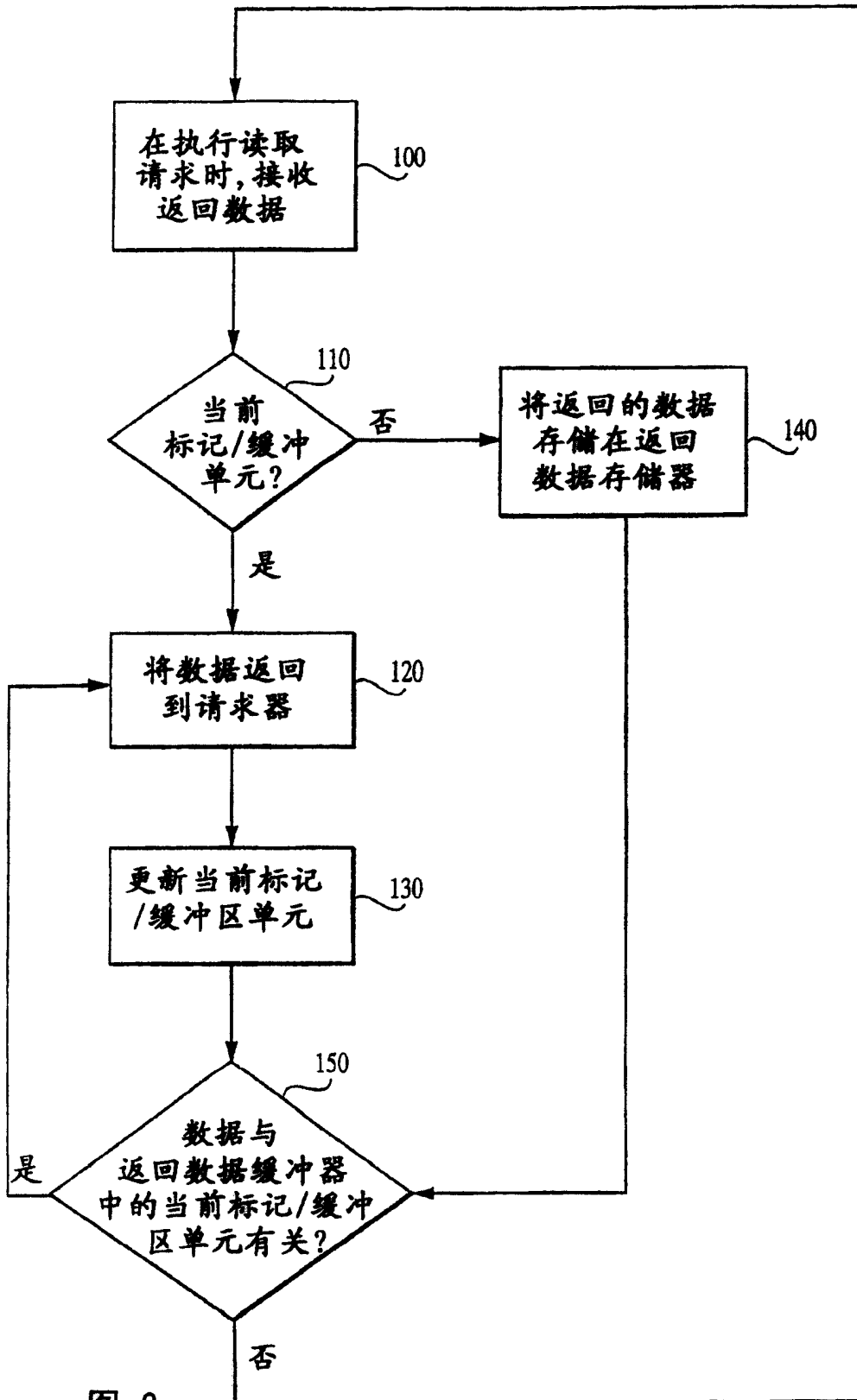


图 2

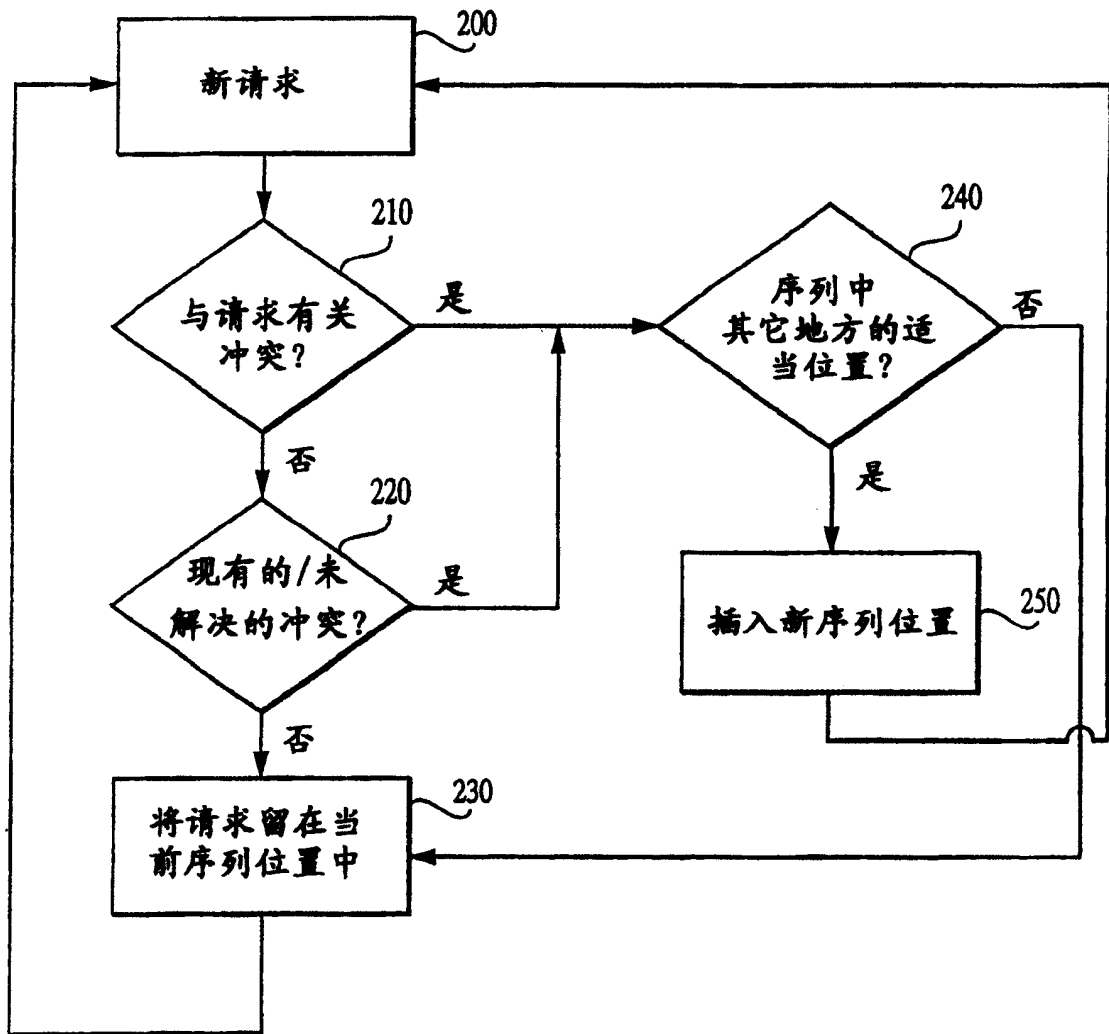


图 3

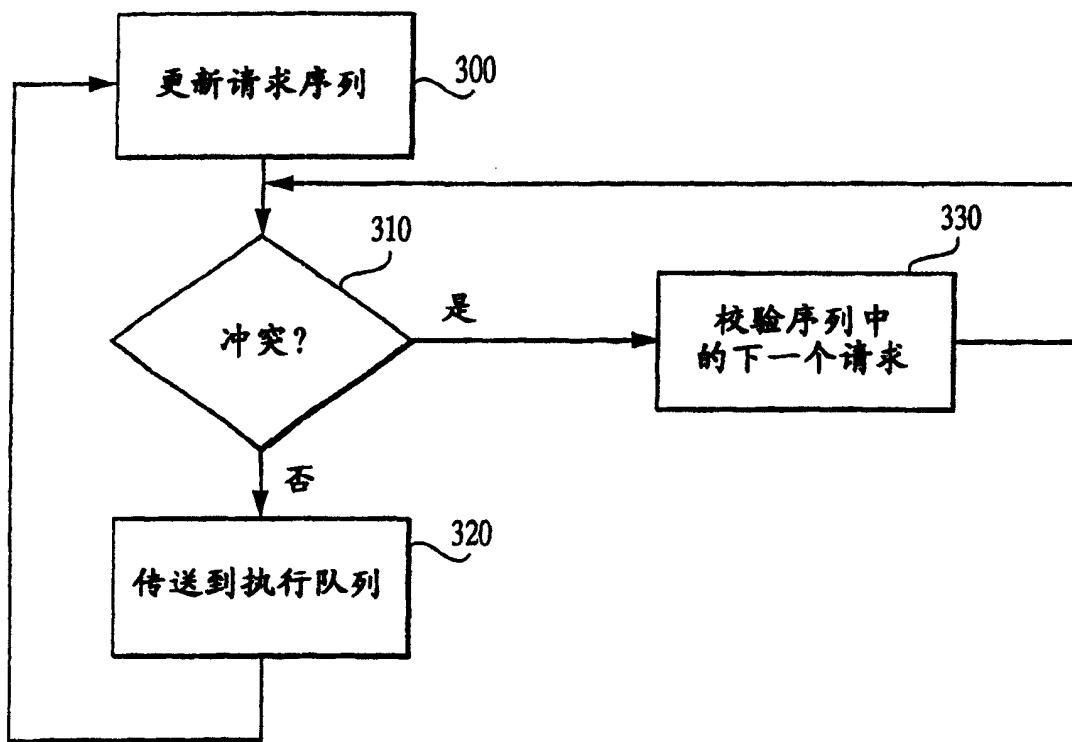


图 4

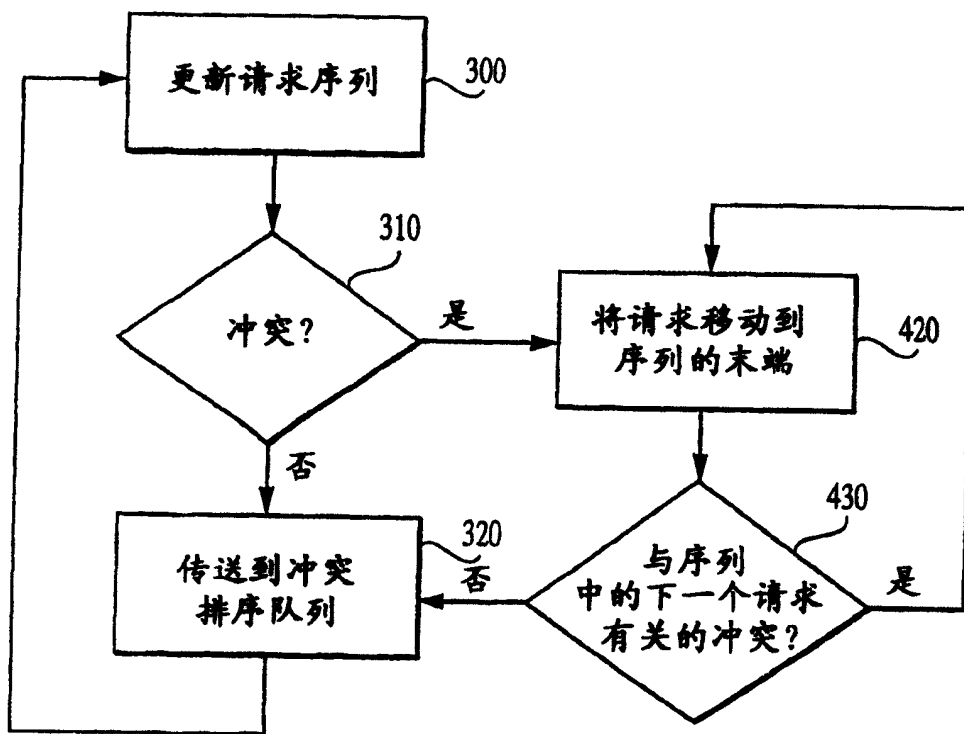


图 5

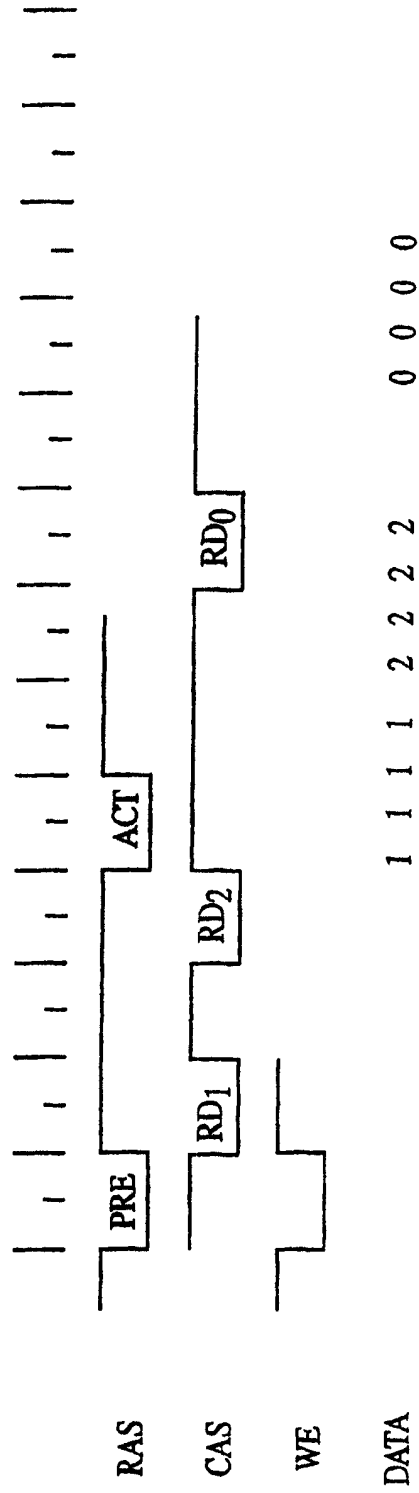


图 6

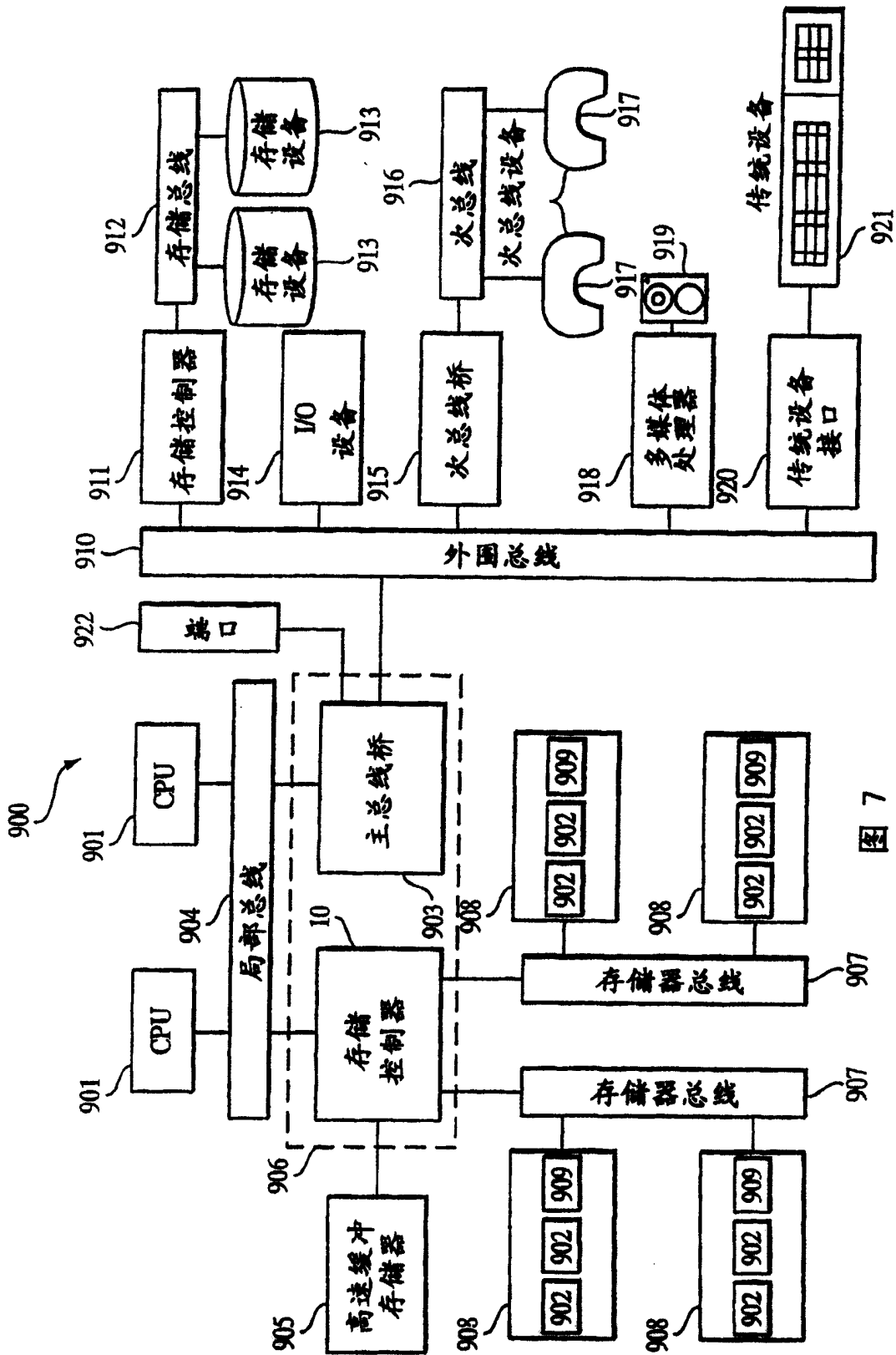


图 7