

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G11C 11/40 (2006.01)



[12] 发明专利说明书

专利号 ZL 200310103611.0

[45] 授权公告日 2010年2月17日

[11] 授权公告号 CN 100590736C

[22] 申请日 2003.10.29

[21] 申请号 200310103611.0

[30] 优先权

[32] 2002.10.29 [33] KR [31] 2002-66269

[32] 2003.2.21 [33] KR [31] 2003-11121

[73] 专利权人 海力士半导体有限公司

地址 韩国京畿道利川市

[72] 发明人 安进弘 洪祥熏 金世俊 高在范

[56] 参考文献

US6191988B1 2001.2.20

US5210723A 1993.5.11

US6445611B1 2002.9.3

US5892724A 1999.4.6

US5917745A 1999.6.29

US5555529A 1996.9.10

审查员 刘浩然

[74] 专利代理机构 北京集佳知识产权代理有限公司

代理人 杨林森 康建峰

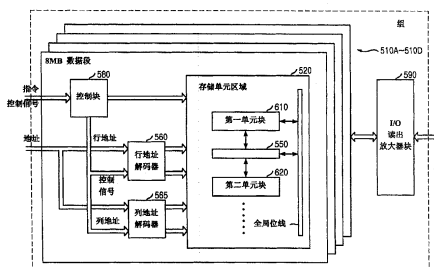
权利要求书2页 说明书47页 附图37页

[54] 发明名称

数据存取时间降低的半导体存储装置

[57] 摘要

一种存储装置，包括以下部分：至少两个单元块，该单元块与响应指令输出数据的全局位线相连；至少一个全局位线连接单元，该连接单元在一个控制块的控制下有选择的连接全局位线与每个单元块，全局位线连接单元位于两个单元块之间；如上所述的控制块，该控制块用于控制将每个单元块内的数据输出到全局位线上及将全局位线上的数据恢复到原来的单元块或另一单元块，具体的选择由响应下一指令的数据是由原来的单元块输出还是由另一单元块输出来决定。



1. 一种存储装置，包括：

至少两个单元块，所述单元块与响应指令输出数据的全局位线相连；
至少一个全局位线连接单元，用于在一控制块的控制下有选择地将该全局位线连接至每个单元块，一个全局位线连接单元被分配在所述两个单元块之间；且

所述控制块用于控制将存储在每个单元块内的数据输出到该全局位线上，及控制将该全局位线上的输出数据恢复到原始单元块或另一单元块，其由响应下一指令的数据是由该原始单元块输出还是由另一单元块输出来决定。

2. 根据权利要求1所述的存储装置，其中，每个单元块包括：

包含多个单元并输出数据的第一单元阵列；

用于放大由该第一单元阵列输出的数据并将该放大数据输出到该全局位线的第一位线读出放大器块；

第一位线读出放大器连接单元，用于在第一位线读出放大器块一读出到该数据时就立即接通或断开第一单元阵列与第一位线读出放大器块的连接。

3. 根据权利要求2所述的存储装置，其中，每个单元块包括：

第二位线读出放大器块，用于放大由该全局位线输入的数据；

第二位线读出放大器连接单元，用于接通或断开第一单元阵列与第二位线读出放大器块的连接。

4. 根据权利要求3所述的存储装置，进一步包括用于暂时锁存响应该指令提供给该全局位线的数据的锁存装置。

5. 根据权利要求4所述的存储装置，其中，所述控制块控制恢复由第一单元块中的锁存装置锁存的数据并响应于下一指令以基本同步的定时周期读取第一单元块或另一单元块中存储的另一数据。

6. 根据权利要求5所述的存储装置，其中，该控制块响应读出指令

控制输出由该锁存装置锁存的数据。

数据存取时间降低的半导体存储装置

技术领域

本项发明涉及半导体存储装置，更具体地，本发明涉及能够提供缩短的数据存取时间的半导体存储装置。

背景技术

一般说来，半导体存储装置可分为随机存取存储装置（RAM）和只读存储装置（ROM）。

随机存取存储装置（RAM）包括动态随机存取存储装置（DRAM）和静态随机存取存储装置（SRAM）。动态随机存取存储装置（DRAM）的每个单元具有 1 个晶体管 and 1 个电容器，而静态随机存取存储装置（SRAM）的每个单元则具有 4 个晶体管和 2 个负载电阻。在芯片集成和制造工艺方面存储装置 DRAM 比存储装置 SRAM 更加高效，所以前者的应用比后者更加广泛。

目前，中央处理器（CPU）的运算速度比动态随机存取存储装置（DRAM）的运算速度快得超乎人的想象。因此由于存储装置的速度比中央处理器（CPU）的速度慢，结果就出现了许多问题。为了解决这些问题，已开发出几种用于高速数据传输的存储装置中的设计方案。

图 1 示出传统存储装置的结构图。

如图所示，存储装置配置了一个指令控制器 200，一个以上的存储体 100 和一个数据输入/输出（I/O）缓冲器 300。指令控制器 200 以不同运行模式对存储体 100 进行控制，如读、写和刷新模式，对从外部输入（没有显示）的指令/RAS、/CAS、/WE、/CS、CKE、CK 等做出反应。存储体 100 包括许多片段 120A—120H，每个片段由一个行地址解码器 121 和一个列地址解码器 122 组成，这样可以将存入单元块根据输入的地址和输

入的数据而读出的单元块内的数据，写入片段 120A 的单元块内。数据输入/输出 (I/O) 缓冲器 300 为存储体 100 单元块缓冲输入或输出的数据。

比较典型的是存储装置具有例如 4 个存储体。然而，由于每个存储体具有相同的结构，所以图 1 只详细提供了一个存储体 100 的结构。为了方便起见，图 1 简化了常规存储装置的元件，但常规存储装置还可以包括另外一个元件。

一个存储体具有输入/输出 (I/O) 读出放大器块 110A 和 110B，对例如从八个片段 120A—120D 和 120E—120H 输出的读出数据进行放大，来为数据 I/O 缓冲器 300 提供放大的数据。每个片段由几个单元块组成。并且输入/输出 (I/O) 读出放大器块 110A 和 110B 向八个片段 120A—120H 提供从数据 I/O 缓冲器 300 输入的数据。

第一片段 120A 包括行地址解码器 121，列地址解码器 122 和一个存储单元区 120A_1。行地址解码器 121 对一个行地址进行解码，向存储单元 120A_1 输出经过解码的行地址。列地址解码器 122 对一个列地址进行解码，向存储单元 120A_1 输出经过解码的列地址。

存储单元区 120A_1 由许多单元块组成，每个单元块包括许多单元。单元块 124A 与一对位线读出放大器块相连，如 123A 和 123B，来读出和放大从单元块（如 124A）中输出的数据。

例如，典型的 256MB 存储装置包括 4 个 64Mb 存储体。在这种情况下，片段长度为 8MB，因为每个存储体由 8 个片段 120A—120H 组成。每个片段（如 120A）包括 8 个单元块 124A—124B，每个单元块（如 124A）包括 256 个字线和 4Kb (4×1024) 位线。这样每个单元块（如 124A）有 $256 \times 4KB$ 个单元组成。在下文中，我们假设片段长度为 8MB，每个单元块包括 256 个字线。

如上所述，如果将每个单元块（如 124a）与两个读出放大器块（如 123A 和 123B）相连，则需要具有 8 个单元块 124A—124H 的片段具有 16 个位线读出放大器块。然而，以常规存储装置的片段 120A 通过 9 个位线

读出放大器块 123A—123I 来执行。一般情况下，每个位线读出放大器块（如 123B）与 2 个相邻单元块相连（如 124A 和 124B）。也就是两个相邻的单元块（如 124A 和 124B）一般情况下控制一个位线读出放大器块（如 123B）以提供芯片集成的效率。在这种情况下，位线读出放大器块（如 123B）在指令控制器 200 的控制下可以有选择地与单元块 124A 和 124B 相连。

图 2 为图 1 所示的位线读出放大器块和单元块的电路图。也就是说，对两个单元块 124A 和 124B 及位线读出放大器块 123B 进行了说明。

如图所示，第一单元块 124A 有 $256 \times 4\text{Kb}$ 个单元，每个单元包括一个 MOS 晶体管和一个电容器，其中一个字线与 MOS 晶体管的栅极相连，每个位线 BL 和 /BL 与 MOS 晶体管的漏极相连。MOS 晶体管的源极与电容器的一个电极相连，电容器的其它电极与一个单元平板电源连接（如地电压）。为了方便起见，由于第二个单元块 124B 与第一单元块 124A 的结构相同，所以没有对第二单元块 124B 的结构进行详细说明。

位线读出放大器块 123B 配置了一个位线读出放大器 12B_4，一个预充电块 123B_3，一个平衡块 123B_2，一个数据输出块 123B_5 及第一和第二位线读出放大器连接单元 123B_1 和 123B_6。

位线读出放大器 123B_4 通过位线读出放大器启用信号 RTO 和 /S 来启用，用以对向位线线对 BL0 和 /BL0 供应的两个电压电平之间的差别进行放大，作为读出数据。根据预充电启用信号 BLEQ'，预充电块 123B_3 对位线对 BL0 和 /BL0 进行预充电达到位线预充电电压 V_{blp} 。预充电启用信号 BLEQ' 在位线读出放大器 123B_4 禁用时启用。平衡块 123B_2 通过一个平衡信号 BLEQ 来启用，对位线线对 BL0 和 BL1 上的电压电平进行平衡，该线对与第一单元块 124A 相连。数据输出块 123B_5 向数据线对 DB0 和 /DB0 传送经过位线读出放大器读出或放大的数据，这一过程是通过利用一个列地址而产生的列控制信号（如 CDO）来完成的。如果第一连接单元 123B_1 通过第一读出放大器连接信号 BISH 来启用，则位线读出放大器

123B_4 与第一单元块 124A 相连。另外，如果第二读出位线读出放大器连接单元 123B_6 通过第二连接信号 BISL 来启用，则位线读出放大器 123B_4 与第二单元块 124B 相连。

因此，第一单元块124A配置了许多单元，这些单元与N个位线线对相连(如BL0和/BL0)……，BLN-1和/BLN-1)和M个字线(如WL0……，WLM-1)。由于第一单元块124A与第一和第二位线读出放大器块123A和123B相连，每个位线读出放大器块 123A或123B中的读出放大器的数量为N/2。因而，在第一单元块123A中，与位线线对相连一些数据(如BL1和/BL1，……)被连到第二位线读出放大器块123B。

此外，第二位线读出放大器块123B可以有选择性在连接到第一和第二单元块124A和124B中的一个，对第一和第二读出放大器连接信号BISH和BISL做出反应。

图3和图4为时序图，对图1和图2中的常规存储装置的动作进行说明。

在下文中，图1至图4对常规存储装置进行了详细描述。

首先，当一个地址输入到存储装置中时，该地址被分为一个存储体地址，一个列地址和一个行地址。行地址进一步分裂为一个单元块地址和一个局部地址。存储体地址用于从四个存储体中选出一个存储体。局部地址用于从被选择的单元块中选择一个字线，列地址用于通过字线从4KB单元中选出一个单元。

对读写模式中的存储装置进行连续的描述。

首先，通过存储地址选择一个存储体(如100)，以对读出指令RDO做出反应。然后一个行地址被输入到一个行地址解码器121，该解码器包含在各个选择的存储体100片段中。

其次，行地址解码器121对单元块地址和输入的行地址中的局部地址进行解码，在存储体100中，一个被选择的片段120A中的单元块124A通过单元块地址和局部地址来选择。

其次，当位线线对BL0和/BL0……，第一单元块中的BLN-1和/BLN-1

通过预充电块123A_3和123B_3（见图2）进行预充电时，第一单元块124A中256字线（如WL0）通过从行地址解码器121中输出的局部地址激活。

然后，存储在被激活的字线WL0上的单元中的4KB每个字节被分别提供给各个位线线对BL0和/BL0……，BLN-1和/BLN-1。这时如果第一个读出放大器连接信号BISH启用，第一个读出放大器连接信号禁止，位线读出放大器块123A和123B就被连接到被选择的第一个单元块124A。

接着，位线读出放大器123A_4和123B_4对位线线对BL0和/BL0，……，BLN-1和/BLN-1处提供的数据进行读出或放大处理。

也就是说，位线读出放大器123B对向一些位线线对向BL0和/BL0，……，BLN-1和/BLN-1提供的数据进行读出或放大处理。同时，位线读出放大器123B对向其它位线线对BL1和/BL1，BL3和/BL3……，BLN-1和/BLN-1提供的数据进行读出或放大处理。当然，在由指令/RAS，/CAS，/WE，_CS，CKE和CK等决定的时序图中，两个单元块124A和124B间分配的位线读出放大器块有选择性地连接到两个相邻的单元块124A和124B，来达到存储装置高度集成和高度运算速度的目的。

于是，列地址解码器122对输入的列地址进行解码，对读出指令RDO做出反应输出列选择信号（如CD0，CD1，CD2……）。4KB数据由位线读出放大器块123A和123B进行放大，并且列选择信号选择的数据通过数据线对DB和/DB被输出到I/O读出放大器块110A中。

I/O读出放大器块（如110A）对被选择的数据信号进行放大来补偿相对较长数据线对DB和/DB中的损失，并向数据I/O缓冲器300输出被放大的数据信号。数据I/O缓冲器暂时存放和输出从各个存储体向外部（没有显示）输出的被放大的数据。

同时，经过位线读出放大器块123A和123B读出和放大锁存的选择4KB数据应在被选择的4KB单元中进行复原，该单元连接在被读出指令RDO激活的字线WL0上。

在常规存储装置中，单元在电容器的电容为几毫微法拉，电容器中

的充电量非常小。结果，在电容器中的电荷作为读出数据向位线放出后，要在由读出放大器进行放大，并应在电容器中复原。通过上述复原操作电容器能够充满其90%电容器容量

并且由于一小部分存储在电容器中的电荷可能会随着时间而失去，为了定时对电容器进行充电，存储装置需要进行刷新操作。因而，存储装置花费一些额外的时间来进行刷新或数据复原操作来防止电容器中的数据丢失，这一点是非常必要的。

图3为一个时序图，显示图1中的存储装置一个存储体中至少两个数据的连续和随后操作。此后，连续和有次序地复原一个存储体中的至少两个数据，以作为内部存储体数据存取进行参考。

如图所示，如果第一个读出指令RD0被输入到存储装置，则被输入的行地址在第一个定时段 t_0 内被分配到单元地址和局部地址。在这里，单元块地址和局部地址为分别用于选择一个单元块和一个字线（如第一单元块124A和和字线WL0。然后在第一时间段 t_0 内存储在单元内的4KB数据经过读出和放大，对由位线读出放大器块123A和123B激活的字线WL0做出反应，读出放大器被连接到第一单元块124A。

其次，在第二个定时时间段 t_1 中，被放大的4KB数据中一个被选择的数据D0被由输入的列地址产生的列选择信号CD0进行选择 and 放大，同时，被放大的4KB数据在原来的单元中进行复原，该单元被连接在激活的字线WL0上。据上所述，一个读出操作在两个时间段 t_0 和 t_1 中完成。

同样，在两个时间段 t_0 和 t_1 后，如果输入第二个读出指令RD1，由第二个读出指令RD1进行的第二次读出操作在第二和第三时间段 t_2 和 t_3 中完成。

由于一个字线被激活对目前行地址或一个指令做出反应，直到下一个字线被下一个行地址或下一个指令激活，图3中所示的标准行周期时间被定义为一个定时周期。

下文对数据写入模式进行了详细描述。

像数据读出模式一样，写入操作也包括4KB数据在一个被选择的字线被激活后由位线读出放大器块进行读出和放大的步骤。

在第一个定时周期 t_0 内，字线被行地址激活接着对激活字线做出反应的4KB数据被位线读出放大器块（如123A和123B）读出或放大。

在第二个定时周期 t_1 内，从外部输入的数据在被选择的位线读出放大器中被锁存，对列选择信号（如CDO做出反应。在这里对新数据进行存储的列选择信号CDO被从列地址中解码。然后，4KB数据通过被激活的字线存储在单元中。对输入数据进行读出和放大是非常必要的，这些数据在位线读出放大器中被锁存，这是因为输入数据的电压电平足以被读出和存储。如上所述，写入操作也在两个定时周期 t_0+t_1 中完成。

据上所述，具有电容器的存储装置需要两个定时周期来完成数据读出或写入操作，电容器作为其一个存储元件。首先，存储在单元内的数据被读出或放大，或从外部输入的数据在第一个定时周期内被读出；其次，被放大的数据被输出到外部，或从外部输入的数据在第二个定时周期中被存储在各个单元中。

在一个高度集成的存储装置中，作为数据信号存储的存储电荷数量非常小，这是因为电容器的尺寸非常小的缘故。结果，如果电容器中的电荷不定时放大，电容器中存储的电荷就可能损失而使存储的数据产生问题。也就是包括数据复原操作的刷新操作应定期进行，在所述读出或写入操作模式后保存单元的数据。

在常规存储装置中，数据被一个指令存取后，下一个数据不能立即进行存取，这是因为最初存取的数据应复原到最初的单元中。

另一方面，常规存储装置使用一个存储体交叉模式来降低在内部存储体中的数据复原时间，也就是提高存储装置的运算速度。内部存储体数据存取表示存储在不同的存储体中，至少两个数据为连续或有次序地存取。

在存储体交叉模式中，常规存储装置可以连续向外部输出数据，对

输入的指令做出反应，而不需要任何额外的数据存取时间进行复原操作。也就是说，在目前数据存储在最初的存储体期间，下一个数据可以从相邻存储体中存取或输出。

图4是为时序图，描述图1中所示的常规存储装置的存储体交叉模式。

如图所示，在存储装置存储体交叉模式中，第一个存储体（如W_{L0}）中的第一个字线在第一个定时周期 t_0 内由第一个读出指令RD₀激活，对第一个激活字线W_{L0}做出反应的第一个数据D₀在第二个定时周期 t_1 中被输出和复原。同时，在第二个定时周期 t_1 中，下一个读出指令RD₁被连续输入到第一个存储体B₁中。第二个存储体（如B₁中的W_{L0}）的第二个字线被激活，被第二个激活的字线W_{L0}做出反应的第二个数据D₁在第二个定时周期 t_1 中被输出。这样，在存储体交叉模式中，第一至第三个数据D₀、D₁、D₂被连续输出，对第一至第三个读出指令RD₀、RD₁、RD₂做出反应。

如上所述，当前数据存储在未来的存储体中时，如果从相邻存储体中输出的数据被下一个输入的相邻存储体内的指令进行读出或放大，数据存取时间就可以减少。

然而，如上所述，在常规存储装置的内部存储体数据存取中很难应用存储体交叉模式在常规存储装置获得较高的数据存储时间。也就是说，在常规存储装置外部存储体数据存储中，常规存储装置能够在当前输出的数据存储最初的单元中存储后接收下一个指令。

因而，即使常规存储装置可以运用存储体交叉模式，存储装置数据存取速度大大依赖于数据存取模式，也就是存储体间的数据存取或存储体内的数据存取。

如果一个系统使用常规存储装置，这种存储装置的数据存取速度严重依赖于数据存储模式，系统操作可靠性就会出现问題，这是因为系统的运算速度不稳定。

因而，研制一种能够高速存取对数据存储模式要求不是很高的存储装置是很必要的。

发明内容

因而，目前发明的方向就是提供一种存储装置方法或装置来高速对数据进行存取，这样数据复原时间不会对数据存取时间产生严重影响。

另外，还提供了存储装置的一个方案和一种装置用于高速数据存取，与数据存取模式不相关，这是通过存储体间或内部单元块交叉模式来实现的，这种模式可以使存储装置高度动作，而不需要当一个存储体中顺序地出现至少两个数据存取时减少数据存取时间。

根据当前发明的一部分，还设计了一种存储装置，包括至少两个单元块与共用位线，根据指令对数据进行输出至少有一个共用位线连接单元可以在控制模块的控制下有选择地连接在各个单元块上，一个共用位线单元被分配在两个单元块之间。上述控制模块对存储在各个单元块的数据进行输出控制，和对共用位线的输出的数据进行复原，这取决于数据是否对下一个指令做出反应，从最初单元块或另一个单元块中输出。

根据当前发明的另一部分技术，设计了一种存储装置，包括许多单元块，每个单元块具有许多单元，第一个局部位线读出放大器块对来自单元的数据进行读出或放大操作，第二个当地位线读出放大器块对另一外单元的数据进行读出或放大；第一个共用位线读出放大器块对从第一个局部位线读出放大器读出或放大的数据进行锁存；第二个位线读出放大器块对从第二个局部位线读出放大器块读出或放大的数据进行锁存。第一个共用位线连接单元，向第一个控制信号做出反应，可以连接把第一个单元块中的第二个局部位线读出放大器块连接到第二个共用位线读出放大器块；第二个共用位线连接单元，对第二个控制信号做出反应，可以有选择地把第二个单元块中的第二个局部位线读出放大器块连接到第二个共用位线读出放大器块或把第四个中的第一个局部位线读出放大器连接到第一个共用位线读出放大器块上；一个控制单元产生控制信号来控制第一或第二共用位线读出放大器块中锁存的数据存储操作。

根据当前发明的另一部分技术，设计了一种存储装置，包括许多第一单元块，第一单元块具有很多单元，第一个局部位线读出放大器块对来自一个单元的数据进行读出和放大操作，第二个局部位线读出放大器块对另一个单元中的数据进行读出或放大；第一个共用位线读出放大器块通过第一个局部位线读出放大器块读出或放大的数据进行锁存操作；第二个共用位线读出放大器块对由第二个局部位线读出放大器块读出和放大的数据进行锁存操作；一个控制单元对在第一个和第二个共用位线中锁存的数据进行复原操作。

根据当前发明的另外一部分，设计了一种存储装置，包括多个单元块，每个单元块具有许多单元单元并具有一个局部位线读出放大器块对许多单元中存储的数据进行读出和放大；一个共用位线读出放大器块对由局部位线读出放大器块读出或放大的数据进行锁存；一个控制单元对复原操作中的被锁存的数据的应用进行控制。

根据当前发明的另一部分技术，设计了一种存储装置，包括具有 $N+1$ 个单元块的存储单元区，每个单元区有 M 个字线对输入的行地址做出反应；一个控制单元对存储单元区进行控制，对存储在第一个单元中的数据进行复原，这些单元块从第一单元块或第二个单元块中的 $N+1$ 个单元块中选出。

根据当前发明的另一部分技术，设计了一种存储装置，包括一个具有 $N+1$ 个单元块的存储单元区，每个存储单元区包括 M 个字线对输入的行地址做出反应；一个预定的单元块表对信息进行复原，在这里， $(N+1) \times M$ 个字线中间至少有一个以上的字线被指定为定的可复原的字线，这是通过使用信息来完成的；一个标签块对输入的逻辑单元块地址进行读出操作来指定一个待复原的单元块；一个控制单元对标签块和定的单元块表进行控制来对从物理单元块地址中选出的单元块中的一个字线进行激活。

根据当前发明的另一部分技术，设计了一种存储装置，包括一个具

有 $N+1$ 个单元块的存储单元区，分别添加 N 个单元块，这些单元块有 M 个字线，对输入的行地址和具有 M 个字线的额外的单元块做出反应；一个数据存取控制单元对操作进行控制，存取在从 $N+1$ 个单元块中选择的第一个单元块中的数据在第一个单元块或第二个单元块中进行复原，一个指令控制单元对操作进行控制，该操作可以对在下一个时序中的第二个指令做出反应，以及使一个对第一个指令做出反应的位线在存储数据期间进行预充电，以便对第一个目前激活的指令做出反应。

根据当前发明的另一部分技术，设计了一种方案对存储装置进行操作，该存储装置具有第一和第二单元块，每个单元块有许多单元，包括下列步骤：(A) 在第一个单元块中读出或放大第一个数据；(B) 在第二个单元块中复原第一个数据；(C) 在第一个单元块中读出或放大第二个数据；在这里，步骤(B)和(C)以基本相同的定时周期进行。

根据当前发明的另一部分技术，设计了一种存储装置，包括一个具有 $N+1$ 个单元块的存储装置，即 N 个单元块和一附加的单元块，两者各对应于一输入行地址，且具有 M 个字线，其中 N 和 M 为正整数，包括以下步骤：(A) 读出并放大从 $N+1$ 个单元块中选出的第一个单元块中的第一数据；(B) 恢复从该 $N+1$ 个单元块中选出的第二个单元块中的第一数据；(C) 读出并放大第一个单元块中的第二数据；其中步骤(B)和步骤(C)两步骤以基本同步的定时周期来执行。

根据当前发明的另一部分技术，提供了一种方案，对存储装置进行操作，该存储装置包括 $N+1$ 个单元块，即 N 个单元块和一个附加单元块，两者对一个输入的行地址做出反应，具有 M 个字线， N 个和 M 为正整数，由下列步骤组成：(A) 激活从 $N+1$ 个单元块中选出的第一个单元块中的第一个字线；(B) 读出或放大 K 个数据对第一个字线做出反应；(C) 把经过读出并放大的 K 个数据移动到一个单元块中，该单元块中有一个预定的字线对第一个字线进行复原；(D) 激活第一个单元块中的第二个字线；(E) 对 K 个数据进行读出和放大，对第二个字线做出反应，在这些步骤中，(C)

和(E)以基本相同的定时周期进行。

根据当前发明的另一部分技术,设计了一种方案对存储装置进行操作,具有一个存储单元区,该单元区包括 $N+1$ 个单元块,即 N 个单元块和一个附加单元块,两者都与一个输入的行地址相一致,有 M 个字线, N 和 M 个正整数,包括下列步骤:(A)通过一个逻辑单元块对数据进行读出操作来选取 N 个逻辑单元块,通过一个局部地址对数据进行复原从 M 个包含在选择的单元块中的 M 个字线中选择一个;(B)把逻辑单元块地址转换到一个物理单元块地址中来从 $N+1$ 个物理单元块地址中选择一个;(C)激活第一个字线,对从第一个单元块中的局部地址做出反应,第一个单元块对被转换的物理单元块地址做出反应;(D)对数据进行读出和放大操作,对第一个字线做出反应;(E)第一个数据移动到第二个单元块中,一个预定字线被指定对第一个字线做出反应;(F)激活第二个单元块对从下一个指令中输入的局部地址做出反应;(G)对第二个数据进行读出或放大,对第二个字线做出反应。这些步骤中(E)和(G)以基本相同的定时周期进行。

根据当前发明的另一部分技术,设计了一种方案,对存储装置进行操作,包括一个存储单元区,该单元区具有 $N+1$ 个单元块,即 N 个单元块和一个附加单元块,分别对一个输入的逻辑单元块地址做出反应,具有 N 个字线, N 和 M 为正整数,包括下列步骤:(A)接收第一个逻辑单元块地址对第一个指令做出反应,转换到第一个物理单元块地址从 $N+1$ 个物理单元地址选择一个;(B)对第一单元块中的第一个数据进行读出和放大,对第一个物理单元块地址做出反应;(C)移动第一个数据并锁存;(D)对第一个单元块中读出或放大的第一个数据进行预充电;(E)收到第二个单元块地址后转成第二个物理单元块地址;(F)把第一个经过锁存的数据移动到从 $N+1$ 个单元块中选出的第二个单元块中,并进行复原;(G)对第二个单元块中的第二个数据进行读出或放大,对第二个物理单元地址做出反应,这些步骤中F和G以基本相同的定时周期进行。

附图说明

结合附图通过以下对所述实施例的说明可以明确本发明的以上和其他功能特点，附图包括：

图 1 所示为传统存储装置的框图；

图 2 所示为图 1 所示的位线读出放大器块和单元块的电路原理图；

图 3 和图 4 所示为用于说明图 1 所示的传统存储装置的数据存取操作的时序图；

图 5 所示为根据本发明的优选实施例所述的半导体存储装置的框图；

图 6 所示为根据本发明的优选实施例所述的半导体存储装置的片段的框图；

图 7 所示为根据本发明的另外一个实施例所述的半导体存储装置内的片段的框图；

图 8 和图 9 所示为根据数据存取模式对半存储装置运行进行说明的时序图；

图 10 所示为用于说明根据本发明的另一种实施例所述的存储装置中的片段的框图；

图 11 所示为说明图 10 所示的存储单元区的电路原理图；

图 12 所示为说明图 11 所示的存储单元区的电路原理图；

图 13A 至图 13D 为用于说明图 11 所示的存储单元区操作的示意图；

图 14 所示为根据本发明的另一种实施例所述的存储装置的片段的框图；

图 15 所示为说明图 14 所示的存储体的单元块的电路原理图；

图 16 所示为根据本发明的另一种实施例所述的存储装置的片段的框图；

图 17 所示为对图 16 所示的存储装置的操作进行说明的时序图；

图 18 所示为根据本发明的另一种实施例所述的发明的存储装置的片

段的框图；

图 19 所示为说明图 18 中所示的预定的单元块表的示意性的框图；

图 20 所示为说明图 18 所示的标签块的示意性框图；

图 21 所示为图 18 所示的控制块的部件的框图；

图 22 和图 23 所示为图 20 所示的标签块的详细示意图存储装置；

图 24 所示为说明图 18 所示的存储装置的操作的时序图；

图 25 所示为说明使用图 18 中所示的标签块的存储装置的操作的时序图；

图 26 所示为根据本发明的另一种优选实施例所述的半导体存储装置的框图；

图 27 所示为说明图 26 所示的存储装置内的单元块间交叉存取模式的时序图；

图 28 所示为说明图 26 所示的读出模式下的存储装置的操作的时序图；

图 29 所示为说明图 26 所示的写入模式下的存储装置的操作的时序图；

图 30 所示说明图 26 所示的读出模式下的存储装置的操作的模拟波形图；

图 31 所示为说明根据图 24 所示的电路产生的信号而进行的操作的电路原理图；

图 32 所示为根据图 31 所示的电路产生的信号而进行的存储装置操作的时序图；

图 33 所示为具有减少的行循环时间的存储装置的框图；

图 34 所示为详细说明图 33 所示的存储装置的框图；

图 35 所示为说明对响应连续输入读出指令而进行的图 34 所示的存储装置操作的时序图；

图 36 所示为对比图 34 和图 18 所示的存储装置的操作的时序图。

具体实施方式

下文将结合附图对本发明的半导体存储装置加以详细说明。

参见图 5，所示为根据本发明的优选实施例所述的半导体存储装置。

如图所示，半导体存储装置包括许多存储体，每个存储体具有至少一个片段 510A-510D，一个输入/输出读出放大器块 590。片段（如 510A）包括一个存储单元区 520，一个控制块 580，一个行地址解码器 560 和一个列地址解码器 565。其中，由于行地址解码器和列地址解码器 560 和 565 与图 1 所示的行和列地址解码器存储装置是相似的，为了方便起见略去对行地址解码器和列地址解码器 560 和 565 进行详细说明。

参见图 6，存储单元区 520 包括至少两个单元块（如 610 和 620），至少一个共用位线连接单元（如 550）。单元块（如 610）通过使用一个共用位线 505 根据指令输出数据。共用位线连接分配在 610 和 620 之间的单元 550，用以有选择地把共用位线 505 连接到每个单元块（如 610 或 620）。为了方便起见，对于与图 5 所示相似的行地址解码器和列地址解码器，所有下文没有对片段实施例进行详细说明。

存储装置能够支持“单元块内交叉存取模式”。在下文中，将单元块间交叉存取模式定义为一种操作，在根据当前指令将当前数据在原始单元块或另一个单元块中恢复的过程中，根据后续指令将下一个数据同步地从同一个单元块中输出。

为了执行单元块内交叉存取模式，控制块 580 以某种方式对存储单元区进行控制，在这种方式中，在单元块（如 610）根据指令向共用位线 505 中输出第一数据后，第一数据复原到原始单元块（如 610），根据下一个指令第二数据同时地从同一个单元块（如 610）中输出单元块。

具体地说，第一单元块 610 包括第一单元阵列 616，第一局部读出放大器单元 612A 和第一局部读出放大器连接单元 614A。第一个单元块 610 还包括第二局部读出放大器单元 612B 和第二局部读出放大器连接单元

614B。

第一单元阵列 616 有多个单元，通过第一和第二局部读出放大器连接单元 614A 和 614B 连接在第一和第二局部读出放大器单元 612A 和 612B 上。第一局部读出放大器块 612A 放大从第一单元阵列 616 输出的第一数据，并将其输出到共用位线 505。第一局部读出放大器连接单元 614A 可以有选择地连接第一单元阵列 616 到第一局部读出放大器块 612A。当将来自第一单元阵列 616 的第一数据供应到第一局部读出放大器单元 616 后，第一局部读出放大器连接单元 614A 随即将第一单元阵列 616 与第一局部读出放大器块 612A 中隔离。

在第一数据连接到 I/O 读出放大器块 590 后，第一数据应复原。这样共用位线连接单元 550 把共用位线 505 连接到第二局部读出放大器单元 612B 上。然后，第一数据通过第二局部读出放大器单元 612B 进行放大；经过放大的第一数据在原始单元中被复原。在经过局部读出放大器单元（如 612A 或 622A）放大后，同时根据下一个指令第二数据能够从同一个单元阵列 616 或另一个单元阵列（如 626）中输出。

这样，发明的存储装置的数据存取时间不依赖于在单元块内数据存取中第一和第二数据是否在同一个单元块或每个不同的单元块进行存取。

图 7 所示为根据本发明的另一种实施例所述的半导体存储装置中的片段的框图。

如图所示，半导体存储装置中的片段包括第一单元块 710，第二单元块 720，一个连接单元 790 和一个控制块。片段还包括一个锁存器 770，存储单元区连接在一个控制块 780 上，该模块与图 6 中所示的控制块相似。

第一和第二单元块 710 和 720 中，每个都有许多单位单元和一个局部读出放大器单元 712 或 722。连接单元 790 用于有选择性地将第一和第二单元块 710 和 720 的二者择一的连接到共用位线 705 或断开与其的连

接单元块。锁存器 770 暂时将提供的数据锁存在共用位线 705 中。控制块 780 支持单元块内交叉存取模式。

在第一单元块（如 710）向共用位线 705 输出第一数据后，为了支持单元块间交叉存取方式，控制块 780 以某种方式对存储单元区进行控制，在这种方式中，当第一单元块（如 710）根据指令向共用位线 705 输出第一数据之后，来自第一单元块 710 的第一数据被复原到另一个单元块中（如 720），根据下一指令，同时从同一个单元块（如 710）中输出第二数据。

具体地说，如果根据当前指令在相同单元块 710 中存取第一数据后，根据下一个指令随后在第一单元块 710 中存取第二数据，则将第一数据复原到第二单元块 720 中，而不是复原到原始单元块 710 中。这时，锁存器 770 临时用于对向共用位线 705 提供的第一数据进行锁存。

也就是，首先，连接单元 790 把第一单元块 710 连接到共用位线 705 中来输出第一数据。第一数据输出到外部电路中并在锁存器 770 中进行锁存。然后，当第二数据通过第一单元块 710 的局部读出放大器单元 712 存取和放大后连接单元 790 把第二单元块 720 连接到共用位线 705 上来对经过锁存的第一数据进行复原。

如果第二数据从第二单元块中存取和输出，当第二数据经过第一单元块 720 中的局部读出放大器单元 722 进行放大时，经过锁存的第一数据在原单元块 710 中进行复原。

这样本发明的创新的存储装置的数据存取时间不依赖于第一和第一数据是否从同一个单元块或一个不同的单元块中进行输出。也就是创新的存储装置能够保证一个对数据存取方式具有更少敏感度的稳定的运算速度。

图 8 和图 9 所示为用于说明根据数据存取方式半导体存储装置的操作的时序图。

如图 7 至 9 所示，本文对根据本发明所述的存储装置的操作进行了

说明。首先，参见图 8，对以单元块内数据存取方式的存储装置的操作进行了说明。

如图所示，通过按顺序输入的指令 CD0 至 CD2，第一数据在第一定时周期 t_0 在第一单元模块 710 中被读出和放大。其中，在第二定时周期 t_1 内，第一数据在第二单元块中进行复原，第二数据在同一个单元块 710 中被读出和放大。

其中，在第三个定时周期 t_2 本发明内，第二数据在第二单元块 720 中被复原，第三个数据在相同的第一单元块 710 中被读出并放大。接着，在第四定时周期 t_3 ，第三个数据在第二单元块 720 中被复原。

在第一数据由第一单元块 710 中的局部读出放大器 712 读出和放大后，经过锁存器 770 锁存的第一数据在第一数据的复原时间内移到第二单元块 720 中。

锁存器 770 和第一或第二单元块 710 或 720 通过共用位线 705 进行数据交换。共用位线 705 可以通过连接单元 790 有选择地连接在第一或第二单元块 710 或 720 上。

参见图 9，在一个存储体间单元块数据存取中，在第一数据存取在第一单元块 710 后，第二数据被存取在第二单元块 720 中。

当根据连续输入的指令 CD0 至 CD2，第一至第三个数据在第一和第二单元块 710 和 720 中进行检索时，第一数据在第一定时周期 t_0 在第一单元模块 710 中进行读出和放大；然后，在第二定时周期 t_1 内，第一数据在第一单元块 710 中被复原，同时第二数据在第二单元块 720 中被读出和放大。

其次，在第三定时周期 t_2 ，第二数据在第二单元块 720 中进行复原；同时第三个数据在第一单元块 710 中被读出和放大。然后，在第四时序周期 t_3 ，第三个数据在第一单元块 710 中进行复原。

例如，在第一数据被读出和放大后，经过锁存器 770 锁存的第一数据在数据复原时间中移到第一或第二单元块 710 或 720 中。也就是说，

当数据从第一和第二单元块 710 和 720 中以单元块间单元块数据存取方式被检索时，单元块间交叉存取模式以某种模式进行，在这种模式中，由下一个指令存取的另一个数据在第一数据在原单元块 770 中复原期间被读出和放大。

当输入的指令为读出指令时，在锁存器 770 中锁存的数据被输出到外部；否则，在写入模式中，来自外部的数据被经过锁存的数据进行替换。

由于在本发明的存储装置中，数据存取操作与复原操作在同一个行循环时间内同时进行，数据存取时间能够大大减少。

图 10 为根据本发明的另一种实施例所述的存储装置的片段的框图。

如图所示，存储装置的片段包括 $N+1$ 个单元块（如 1010 至 1050）， $N/2+1$ 共用位线连接单元（如 1090A 至 1090C），第一和第二共用位线读出放大器单元 1075A 和 1075B，和一个控制块 1080。

每个单元块（如 1010）包括一个单元阵列 1016，该阵列有许多单元，一个第一局部位线读出放大器块（如 1012A）和一个第二局部位线读出放大器块（如 1012B）。第一共用位线读出放大器单元 1075A 用来对由第一局部位线读出放大器块（如 1012A）读出和放大的数据进行锁存。第二共用位线读出放大器单元 1075B 用于对由第二局部位线读出放大器块（如 1012B）读出并放大的数据进行锁存。各个共用位线连接单元（如 1090）用于有选择性地使共用位线 1005 连接到第二局部位线读出放大器单元（如一个单元块（如 1010）中的 1212B）和第一局部位线读出放大器块（如其它单元块（如 1020）中的 1022A。控制块 1080 用于对来自每个单元模块的数据进行输入和输出操作，在原单元或其它单元中对输出的数据进行复原。

控制块 80 支持下列操作，在单元块内数据存取中，当第二数据在第一单元模块 1010 中存取时，第一数据复原在另一个单元块（如 1050）中而不是复原在第一单元块 1010 中。

另外，控制块 80 还支持这个操作，在单元块间数据存取中，当第二数据在另一个单元块（如 1020）中进行存取时，第一数据在第一单元块 1010 中进行复原。

图 11 为图 10 所示的存储单元区 1020 的电路图。

如图所示，存储单元区 1020 包括第一至第五单元块 1180A-1180E，每个单元块有许多单元，第一和第二共用位线读出放大器块 1175A 和 1175B 通过共用位线 1005 与各个单元块进行连接。

每个第一和第二共用位线读出放大器单元 1175A 和 1175B 有许多共用位线读出放大器 GSA。第一共用位线读出放大器单元 1075A 中的共用位线读出放大器 1170A 通过第一共用位线线对 GBLO 和/GBLO 连接在共用位线读出放大器 1170 上。被第一和第二共用位线读出放大器单元 1075A 和 1075B 放大的数据通过共用读出放大器线 1105A 和 1105B 进行输出。

共用位线读出放大器 GSA 对来自局部位线读出放大器 LSA 中的数据进行锁存和放大。共用位线读出放大器 GSA 的配置可以与局部位线读出放大器 LSA 的配置相同。

共用位线连接单元（如 1090A）可以有选择性在把共用位线 1005 连接到两个相邻的局部位线读出放大器（如 1212B 和 1022B）上。每个共用位线连接单元如（1090A）配置在两个单元块 1010 和 1020 之间。如果单元块的个数为 5（如图 11 所示），则可以提供 3 个共用位线连接单元。如果数量为 9，则提供 5 个共用位线连接单元。

如上所述，每个单元块（如 1010）包括单元阵列（如 1216）和第一和第二局部位线读出放大器（如 1012A 和 1012B）。第一和第二局部位线读出放大器有许多位线读出放大器。每个第一局部位线读出放大器（如 1012A）通过一个开关晶体管连接到共用位线 1005 上。

第一至第五单元块控制块 1180A 至 1180E 可以分别输出一个开关控制信号（如 GBIS12 或 GBIS34），通过共用位线连接单元（如 1090A、1090B、1090C）来有选择性地将包含在每个单元块（如 1010）局部位线读出放大

器 LSA（如 1212A 或 1212B）连接到共用位线 1005。

通过连接到第一和第二共用位线读出放大器 1070A 和 1070B 上的共用位线 1005，在第一和第二共用位线读出放大器块 1070A 和 1070B 中经过锁存的数据可以输出到外部，或者从外部输入的数据可以提供到第一和第二共用位线读出放大器块 1070A 和 1070B。

图 12 为图 10 中所示的存储单元区的单元块的部分电路的电路原理图。

包含在每个第二单元块（如 1010）中的每个单元阵列（如 1016）有许多字线 WL_0, WL_1, \dots ，和 WLN ，许多位线线对（如 DB_0 和 $/BL_0$ ） \dots BL_N 和 $/BL_N$ ，许多单元。在这里，单元包括一个电容器和一个 MOS 晶体管，相应地连接地每个字线和每个位线线对。

如图所示，第一单元块 1010 中的第二局部位线读出放大器块 1012B 包括一个位线读出放大器 1230B，一个预充电块 1210B，一个位线读出放大器连接块 1040B 和一个平衡块 1220B。由于这些块 1210B, 1220B, 1040B 和 1230B 与图 2 中所示大体相同，为了方便起见，没有对其进行详细说明。然而，特别是防止在每个共用位线对 GBL_0 和 $/GBL_0$ 中的数据冲突，与一单元阵列相关的第一局部位线读出放大器 LSA 被连接到共用位线读出放大器对 GSA 的一个，并且与同一单元阵列相关的第二局部位线读出放大器 LSA 被连接到共用位线读出放大器对 GSA 的另一个。

在单元块控制器（如 1180A）的控制下，每个位线读出放大器（如 1012A 和 1012B）有选择地连接到共用位线 1005 或共用位线连接单元（如 1090A）上。

图 13A 至 13D 为用于单元块中的数据存取方法和向共用读出放大器块输出存取数据的方法的电路图。

图 13A 至 13D 对局部位线读出放大器 LSA 和共用位线读出放大器 GSA 之间的数据传输进行了说明。特别是分别从第一和第二局部位线读出放大器块输出的两个数据通过共用位线对 GBL_0 和 $/GBL_0$ 向共用位线读出放

大器 1170A 和 1170B 中传输。

在下文中，参见图 10 至 12 和图 13A 至 13D，对图 10 中所示的存储装置进行了详细的说明。

参见图 13A 和 13C，在读出模式下，对存储在单元块（如 1020 和 1040）的数据读出操作进行了说明。指令输入到存储装置后，响应于指令存储在单元块（如 1020）的数据通过输入的地址进行选择。然后，被选择的数据通过两个局部位线读出放大器（如 1320A 和 1320B）进行读出和放大。被读出和放大的数据被移动，接着被锁存到共用位线读出放大器 1170A 和 1170B（如图 13A 至 13C 所示，由粗线标明的部分）。

为了方便起见，一位数据传输在图 13A 至 13D 中进行说明。然而，根据对激活的字线（如 WLO），几个位的数据（如 4KB 数据）可以在同一时间中输出多位数据。

参见图 13B 和 13D，对存储在单元块（如 1010 和 1050）的数据的写入操作进行了说明。在指令输入到存储装置后，从外部输入的数据在共用位线读出放大器 1170A 和 1170B 中被锁存。然后，被锁存的数据移到两个局部位线读出放大器（如 1350A 和 1350B）中。同样写入操作在共用位线读出放大器 1170A 和 1170B 及其它单元块（如 1010、1020、1030 和 1040）之间进行（如图 13B 至 13D 所示，由粗线标明的部分）。

另外，在发明的存储装置中，每个单元块有两个局部位线读出放大器对数据进行读出或放大。也就是当根据字线 4KB 数据被输出后，一些数据（如 2KB 数据）被第一局部位线读出放大器块读出和放大。结果，存储体应有两个共用位线读出放大器块（如 1170A 和 1170B）。

图 14 为根据本发明的另一种实施例所述的存储装置中的片段的框图，图 15 为图 14 中所示的存储体的单元块的电路原理图。

如果片段单元块（如 1410）有一个局部位线读出放大器块（如 1412），所有对被激活的字线做出的数据（如 4KB 数据）被局部位线读出放大器 1412 读出和放大反应，则不需要共用位线连接单元，仅需要一个共用

位线读出放大器块。

因而，图 14 和图 15 只对单元块（如 1410）有一个局部位线读出放大器块（如 1412）的情况下的存储体进行了说明。在此，为了方便起见，因为局部位线读出放大器块（如 1412）中的每个块（如 1510、1520、1530 和 1540）对于每个块（如 1210B、1220B、1230B 和 1240B）（见图 12）是相同的，所以对单元块（如 1410）的操作说明就省略了。

如图 15 所示，字线（如 WL0）选择的数据被位线读出放大器 1412 读出。然后被读出和放大的数据通过一个位线读出放大器 1412 的连接单元 1540 连接到共用位线线对 GBL0 和 /GBL0。在这里，位线读出放大器包括一个平衡单元 1510，一个预充电单元 1520，一个读出放大器 1530 和连接单元 1540。

图 16 为根据本发明的另一种实施例创新的存储装置的存储体的框图。

如图所示，存储体包括一个存储单元区 1620 和一个控制块 1680。存储单元区 1620 有 $N+1$ 个单元块（如 1630_1……1630_ $N+1$ ），一个数据锁存器 1670。在每个单元块中， M 个字线被连接到许多单元上。 N 和 M 为正整数。在这里，存储装置的大小，也就是存储能力计算不含附加单元块。也就是说，在图 16 中，存储体大小为 M （字线数） $\times N$ （单元块数） \times （位线数）。

控制块 1680 支持内单元块交叉模式。在根据当前指令第一数据从第一单元块 1630_1 中输出后，第一数据在数据锁存器 1670 中被锁存。然后，如果根据下一个指令第二数据从第一单元块 1630_1 中输出，第一数据在另一个单元块（如 1630_ $N+1$ ）中进行复原。这时，由于存储装置能够使用一个附加的单元块和数据锁存器，第一数据的行地址没有改变。

图 17 为时序图说明图 16 中所述的存储装置的操作的时序图；特别是当第一和第二数据按顺序被存取在 $N+1$ 个图 16 所示的单元块当中的同一个单元块时的单元块内交叉存取模式进行了说明。

下文中，参见图 16 和 17，对图 16 所示的根据本发明的存储装置的操作进行了说明。

附加单元块用于当第二数据在同一个单元块中被存取时对第一数据进行复原。图 17 所示，第一数据被存储在与第一字线 WL0 相连的单元上，第二数据在与同个单元块的第二字线 WL1 相连的单元中存储。

首先，在第一定时周期 t_0 内，第一字线 WL0 被激活对第一指令 CD0 做出反应，然后根据第一字线 WL0 第一数据被读出和放大。被放大的第一数据被移动到数据锁存器 1670 中。

在第二定时周期 t_1 内，第一单元块 1630_1 的第二字线 WL1 被激活对第二指令 CD1 做出反应，同时第三个单元块 1630_3 的第一字线 WL0 被激活。然后根据第二字线 WL1 第二数据被读出和放大，同时，第一数据被复原到单元中，对第三单元块 1630_1 的第一字线 WL0 做出反应。

如上所述，依照本发明设计的存储装置的数据存取时间可以排除数据复原时间，因为第二数据可以在根据当前指令第一数据被复原的期间由下一个指令进行读出和放大。这样，数据存取时间大大减少，从而实现了高速存取。另外，在本发明的存储装置中，数据复原操作可以通过改变数据单元地址就可以大大简化。

图 18 为根据本发明的另一种实施例的创新的存储装置的片段的框图。图 18 中的片段的配置结构与图 16 中所示的片段的配置结构非常相似，只是预定的单元块表 1810 和标签块 1830 被连接在一个控制块 1880 上。

预定单元块表 1810 包括许多寄存器对预定复原单元块地址信息进行存储。标签块 1830 基于预定复原单元块地址信息产生一个目标复原单元块地址，其对应的存取单元块地址相应。控制块 1880 对预定单元块表 1810，存储单元区 1820 和标签块 1830 进行控制，来实现对内单元块交叉模式的支持。

图 19 为图 18 所示的预定单元块表 1810 的框图。

如图所示，预定单元块表 1810 包括 $N \times (N+1)$ 个寄存器，每个寄存器都有预定复原单元块地址信息。预定复原单元块地址信息包括一个预定复原单元块地址和状况信息。预定复原单元块地址包括 3 位及如包括 1 位的其状况信息。预定复原单元块地址信息表示与被存取的单元块的字线相应的需要复原的目标单元块，其状况信息表示要复原的目标单元块是否有效。

预定复原单元块表 1810 接收一个当前行地址 CUR_RA，一个前行地址 PRE_RA 和一个前复原物理单元块地址 PRE_PBA，预定复原单元块表 1810 向标签块 1830 输出一个复原单元块地址 EXTRA_BA。

图 20 为图 18 中所示的标签块 1830 的框图。

如图所示，标签块 1830 包括 $N+1$ 个单元标签表 1910 至 1910（在这里，如 N 等于 8）， $N+1$ 个比较器 1912A 至 1912B（在这里，如 N 等于 8），一个单元块地址解码器 1832，一个单元块地址编码器 1834 和一个标签控制块 1838。

单元块地址解码器 1832 接收从预定复原单元块表 1810 中输出的复原单元块地址 EXTRA_BA，在单元标识表中选择一个，对有效复原单元块地址 EXTRA_BA 做出反应。然后，比较器（如 1912A）对基于选择单元标识表转化现行地址 CUR_PA 的地址和逻辑单元块地址 CUR_LBA 进行比较。最后，单元块地址解码器 1934 输出一个现复原物理单元块地址 CUR_PBA，该地址被编码对比较结果做出反应。在这里，每个单元标识表 19A0 至 1910 存储一个与 $N \times 256$ 字线相应的复原单元块地址信息。

标签块 1830 还包括一个延迟元件 1836，用于将当前复原的物理单元块 CUR_PBA 延迟一个时钟周期以存取一个复原单元块。

其中，每个单元标识表 19A0 至 1910 有 256 个寄存器，一个寄存器 LBA 由三个位组成。例如，第一单元标识表 19A0 存储逻辑单元块与第一单元块中的每个字线相对应的信息，第二单元标识表 432B 存储逻辑单元块与第二单元块中的每个字线相对应的信息上。

另外,在每个标识表 19A0 至 19I0 中,根据每个单元块的字线“WL0”第一寄存器 0 存储单元块逻辑单元块地址,根据每个单元块的字线“WL1”第二寄存器 1 存储单元块逻辑单元块地址。根据每个单元块的字线 WL255 第 256 个寄存器 255 存储单元块逻辑单元块地址。

图 21 为图 18 中所示的控制块 420 的部分框图。

如图所示,控制块 1880 有信号输入块 2110 接收指令 EXT_CMD,一个逻辑单元块地址 EXT_LBA 和一个局部地址 EXT_RA,来输出一个当前指令 CUR_CMD,一个当前逻辑单元块地址 CUR_LBA 和当前行地址 CUR_RA,前一指令 PRE_CMD,前一逻辑单元块地址 PRE_LBA,前一行地址 PRE_RA。

另外,控制块 1880 中的地址转换控制块 2120 接收指令 CUR_CMD,逻辑单元块地址 CUR_LBA 和局部地址 CUR_RA 和指令 PRE_CMD,逻辑单元块地址 PRE_LBA 和局部地址 PRE_RA。然后,地址转换控制块 2120 输出预定单元块更新信号 (EBT_UPDATE,预充电有源信号 PC_ACT[0:8],指令删除符号_CMD_KILL 和一个内部复原信号 INTW[0:8])。

当复原物理单元块地址 CUR_PBA 与前一工作时钟的 PRE_PBA 同相同时,预定复原单元块更新信号 EBT_UPDATE 用于更新存储在预定复原单元块表 1810 中的信息。

另外,当单元块的字线被连续存取时,指令删除信号 CMD_KILL 终止下一个指令的操作。当单元块的字线被连续存取时,如果标识信息在与物理单元块地址的信息没有对前一个指令做出正确反应的情况下,就会出现错误。

预充电有源信号 PC_ACT[0:8]用于选择物理单元块地址对输入的指令做出反应。在当前执行时钟的物理单元块地址 Cur_PBA 与前一个执行时钟的 Pre_PBA 相同时,内部复原信号 INTW[0:8]用于传送通过当前有效指令存储在单元块中从而在数据锁存器中被锁存到单元块中的数据,该单元块中的字线被指定,从而对数据进行复原。

时钟延迟块 2121 延迟内部复原信号 INTW[0:8],使内部复原信号

INTW[0:8]比预充电有源信号 PC_AC[0:8]时间推迟一个时钟。预充电信号 PC_ACT[0:8]选择单元块后，用内部复原信号 INTW[0:8]复原数据时，在被选择的单元块中选择一个字线，或读出或放大被选择的字线数据时，需要这一步骤。

图 22 为示出图 19 中所示的各个单元标识表 19A0 至 19I0 的第一至第三位的电路图，图 23 为图 19 中所示的各个标识表 19A0 至 19I0 的第四位的电路图。

如图所示，第一至第三位为动态单元，第四字节为静态单元，这是为了满足保持操作速度的需要。静态单元用于第四位，这是因为第四位比第一至第三位相对更新较快。

图 24 为用于说明图 18 中所示的创新的存储装置的存储体操作的时序图。

下文中，请参见图 18 至 24，对根据本发明设计的存储装置的操作进行了详细说明。

首先，指令 CD0 在第一定时周期被输入，要运行一个指令（如 CD0）需要两个定时周期。例如，在第一和第二定时周期 t_0 和 t_1 中进行第一指令 CD0 的运行。每个定时周期（如 t_0 、 t_1 ……）被分为两个时序集（如 a_0 和 b_0 ）。在时序集 a_0 中，运行与标识相关的块；在另一时序集中，运行存储单元区 1820 的块。

首先要详细说明 a_0 的时序集。

在 a_{0-1} 的时序集中，控制块 1480 要接收到逻辑单元区地址以从包含在单元区域 1820 中的 8 组单元区中选出一组，还要接收局部地址从 256 条字线中选出一条来。

然后，在 a_{0-2} 的时序集中，标签区 1830 把逻辑单元区地址转化为物理单元区地址

在 a_{0-3} 的时序集中，预定的分区表 1810 会读出响应当前的输入的局部地址 Cur_RA 的字线 WLO 含有的预定字线的一组单元的区地址。于是就

把含有预定的字线的单元区地址 Extra_BA 输出到控制区 420 和标志区 430 来作为对比较结果的响应。

然后，在 a_{0-4} 的时序集中，就把用第一条指令 CD0 选中的这组单元区与用先前的指令选中的那组单元区比较的结果来更新预设的单元区 410。

接着，就说明响应第一定时周期 t_0 的第一条指令 CD0 的单元区域 1820 的操作。

在 b_{0-1} 的时序集当中，在响应转换过来的物理单元区单元地址 Cur_PBA 的第一组单元区中，响应局部地址的第一条字线被激活。

然后，在 b_{0-2} 的时序集当中，响应第一条字线的 K 位位第一数据被读出并被放大。

在 b_{0-3} 的时序集当中，K 位的第一数据就被锁存在数据锁存区 1870 中。

于是，在读模式下，即第一条指令 CD0 是一条“读”指令的模式下，当输入第二条指令 CD1 时，就输出用锁存在数据锁存器 1870 中的 K 位数据列地址选中的数据。然而，在写模式下，即第一条指令 CD0 时一个“写”指令的模式下，从外部单元输入的数据就会覆盖用锁存在数据锁存器 1870 中的 K 位数据列地址选中的数据。

在第二定时周期 t_1 中，即当响应第一条指令 CD0 的 K 位第一数据确实被输出到存储设施的外部单元的时序集中，可以在另一个响应第二条指令 CD1 的 4K 位数据被锁存在数据锁存区 1870 之前输出 K 位第一数据。

接着，当与标志有关的区在第二定时周期 t_1 操作时，就会在每个时序集中，例如， a_0, a_1, \dots 中说明。

在 a_{1-1} 时序集中，响应第二条指令 CD1 而输入的行地址 CD1 就被分割成局部地址和全局单元区地址。

在 a_{1-2} 时序集中，标志区 430 把逻辑单元区地址 Cur_LBA 转化为物理单元区地址 Cur_PBA。

在 a_{1-3} 时序集中，预定的分区表 1810 读出响应由第二条指令 CD1 输

入的局部地址 Cur_RA 的第二条字线 WL3，从而把含有预定字线的单元区地址 Extra_BA 输出到控制区 1880 和标志区 1830。在这里，选中的寄存器就被更新了。

然后，在 a_{1-4} 时序集中，由第二条指令 CD1 选中的这组单元区地址就与由第一条指令 CD0 选中的那组单元区地址进行比较。

作为比较的一个结果，如果用第一条指令 CD0 和第二条指令 CD1 连续地得到的是同一组单元区，由第一条指令 CD0 选中的第一条字线的预定的这组单元区就要被响应第一条指令 CD0 的那组单元区代替。在这里，代替预定的字线是指更新存储在预定的单元分区表 1810 中的信息。

例如，响应第一条指令 CD0 的字线“WL1”的这组预定的单元区被指定为输入第二条指令的计时点开始的第四组单元区；并且在更新了预定的单元分区表 1810 之后，响应第一条指令 CD0 的字线“WL1”的预定的那组单元区就从第四组单元区改变到第一组单元区。也就是说，在预定的单元分区表 1810 中，响应字线“WL1”的寄存器是“011”。而在更新预定的单元分区表 1810 之后，该寄存器就被改为“000”了。

接着，就要说明响应第二定时周期 t_1 的第二条指令 CD1 的单元区 1820 的操作了。

在 b_{1-1} 时序集中，响应由第二条指令 CD1 输入的局部地址的第二条字线被激活。同时，响应在第一定时周期 t_0 的 a_{0-3} 时序集内的由第一条指令 CD0 选中的第一条字线的预定的那组单元区被激活。

然后，在 b_{1-2} 的时序集中，K 位寄存在数据锁存区 1870 的第一数据被存储在响应第一条字线的预定的那组单元区中。响应第二条激活的字线的 K 位的第二位数据也同时被读出并被放大。

在 b_{1-3} 的时序集中，响应第二条字线并被读出和放大的 K 位第二位数据被锁存在数据寄存器 1870 中。

然后，在读模式下，即第二条指令 CD1 是一条“读”指令的模式下，当输入第二条指令 CD1 时，就输出由锁存在数据锁存区 1870 中的 K 位数

据中的列地址选中的数据。同时，在写模式下，即第二条指令 CD1 是一条“写”指令的模式下，从外部单元输入的数据就会覆盖由寄存在数据锁存区 1870 中的 K 位数据中的列地址所选中的数据。

在单元块内数据存取模式下，本发明的存储装置并不存储第一数据但却可以存取本组单元区中的第二数据，并且第一数据的恢复操作是由预定的那组单元区来完成的。

同时，还要说明在单元块间的数据存取模式下的存储装置操作。

首先，第一数据被读出并且在第一组单元区被放大，储存在数据锁存区 1870 里面。在这里，第一数据被存储在第一组单元区中，而实际上在同一个定时周期内，第二数据被读出并且在另外一组单元区中放大后存储在数据锁存区中。

因此，本发明的存储装置在单元区内部存取模式下支持单元区间交错模式，在单元区间存取模式下支持单元区内存取模式。结果存储装置就可以稳定的高速存取数据而不必依赖数据存取的模式。

图25就是一个说明图18中的存储装置操作的时序图。

如图所示，在与标志有关的区的操作时序中，存储装置的操作包括下列步骤：输入局部地址的步骤和由此读出预定的字线A；把逻辑单元区地址转化为带有标志块430B的物理单元区地址；然后比较两组可以被当前起作用的指令和前一个指令F存取的单元；用比较他们后的结果G来更新预定的单元区。这里，单元区可以激活选中的字线，等等，来更新预定的单元区G。

图26展示了一个与当前发明的另一个优选实施例相一致的半导体存储装置。

与当前实施例相一致的存储装置有很多的存储体，但是图26中仅说明了其中的一个存储体2600，这是因为每个都有相同的结构和配置。

如图所示，一个存储体2650总共有从2640A到2640H的8个段，和用于放大和传输从2640A到2640H输出的数据或将从数据输入/输出缓冲器

2690输出的数据传输到从2640A到2640H的段的输入/输出读出放大器区的2580A和2680B。

另外，数据段2640A有一个低位地址解码器2660来解码从控制块2700输出的地址并将它输出到单元区域2650，还有一个列地址解码器2670，用于把输入的列地址输出到单元区域2650。

在图26中显示的存储装置的大小是256Mb。如果存储装置由四个存储体组成，每个存储体就有64Mb的单位单元。如果一个存储体含有8段，每段的大小就是8Mb。

包括在每个段中的单元区域2650包含从2710到2790的9个单位单元区，每一个单位单元区有256条字线，4Kb (4×1024) 字节线，和 $256 \times 4Kb$ 单位单元。

控制块2700包含了一个把从2710到2790的9个单位单元区的逻辑单元区地址转化为物理单元块地址的标志块2730，一个用来储存关于256条预定字线的信息的预定单元分区表，一个用来控制标志块2830和预定单元块2810的标志控制器2820。

由于除了响应输入的单元区地址区的单位单元区外，单元区域2650还有另一个单位单元区，输入的单元区地址就被当作逻辑单元区地址，而控制块2700的标志区2830则将逻辑单元区地址转化位物理单元区地址。控制块2700也搜索那些响应由输入行地址选中的字线的单位单元区。

另外，包含在单元区域2650中的9个单位单元区中的任何一个，例如2710，有局部位线读出放大器块，例如2712A和2712B，一个局部位线读出放大区2712A读出和放大响应选中的字线的K个数据中的一部分，而另一个局部位线读出放大器块2712B读出和放大K位数据中的其他部分。

单元区域2650有一个全局位线读出放大器块2705A和另一个全局位线读出放大器块2705B用于锁存从包含在从2710到2790的每一个单位单元中的局部位线读出放大区输出的数据；一个全局位线2707，用于在每一个单位单元区，例如2710和第一个，第二个全局位线读出放大区2705A

和2705B之间传输数据；还有从2718到2758的五个全局位线连接设备，用于把全局位线2707连接到从2710到2790的九个设备单元区。

第一个和第二个全局位线读出放大器块2705A和2705B主要是提供给局部位线读出放大器和锁存从局部位线读出放大器区输出的数据以及放大通过全局位线传输时减弱的数据信号的读出放大器。

图27是一个说明图26中展示的存储装置在单元块间交叉数据存取中，从单元区间存取数据的一个的单元区的时序图。图28是一个说明在读模式下从单元区内部存储装置操作的时序图，而图29则是一个说明图26中的存储装置在写模式下，从单元区内存取数据的一个的单元区的时序图

首先，参考图27，在那里说明了单元块间交叉数据存取中与当前实施例相一致的存储装置。

响应一条指令输入的行地址具有：要指定存储体、段，和段内的单位单元块的单元块地址；用于在选中的单位单元区中选中一条字线的局部地址。作为对指令的响应，要输出用于从包含在一个单位单元区中的4K条位线里面选出一条的列地址。

在第一个定时周期 t_0 ，响应第一条读指令RD0的逻辑单元区地址被转化为物理单元区地址，并且第一个单位单元区BL0被转换后的物理单元区地址选中。

首先，第一个单元区BL0的字线WL0被响应第一条读操作指令RD0的局部地址激活。这储存在响应激活的字线的4K个单位单元中的4K位数据被局部位线读出放大器区，例如2712A和2712B读出并放大。

这4K位读出并放大的数据就被移动到第一个和第二个全局位线读出放大器区2705A和2705B，并且在那里锁存。于是，用响应第一条读指令RD0的列地址从锁存在第一个和第二个全局位线读出放大器区2705A和2705B中的4Kb数据选出的一个数据读取指令D0被输出到输入/输出读出放大器区2680A。在这里，当实际数据D0被输出到存储装置的外围电路的时

间是第二个定时周期 t_1 。

在第二个定时周期 t_1 内，这4Kb的锁存在响应第一条读指令RDO的第一条和第二条全局位线读出放大器区2705A和2705B里面的数据就被移动到响应第一个单位单元区BL0的字线WL0的4K个单位单元中。

换句话说，在响应第一条读操作RDO的数据存储过程中，作为对第二条读指令RD1响应的选中的第二个单位单元区BL1在这里被激活，而作为对激活的字线WL0的响应的4Kb数据被读出，并且被包含在第二个单位单元区（BL1）2720中的局部位线读出放大区2722A和2722B读出和放大。

然后，这4Kb响应第二条读操作指令RD1的被读出并放大的数据就通过全局位线2707被移动到第一个和第二个全局位线读出放大器区2705A和2705B并且从此锁存在那里。

被列地址选中的，锁存在第一个和第二个全局位线读出放大器区171和173中的作为对第二条读指令RD1响应的4Kb数据就被输出到输入/输出读出放大器区180A。在这里，当实际数据D1被输出到存储装置的外围电路时的定时周期是第三个定时周期 t_2 。

于是，在第三个定时周期 t_3 内，这寄存在第一个和第二个全局位线读出放大器区2705A和2705B中作为对第二条读指令RDO的响应的4Kb数据就被移动到响应第二个单位单元区（BL1）152的字线WL0的4Kb的单位单元中。

因此，在单元间块数据读取模式中，与当前实施例相一致的存储装置读出并放大储存在另一个单位单元区中的其他数据。

在单元间块数据存取模式中，读数据的时间被减少下来留给了通过在交叉存取模式下的数据恢复时间，并且因此存储装置就可以高速执行。图27中所示的单位单元区交叉存取的行周期时间称作通过输入一条指令而激活的字线之后激活响应下一条指令的字线的时间集。

另外，没有说明数据在单位单元块间依次写入的执行，因为它和读执行在除以下之外是相同的：以响应写指令而输入的数据覆盖响应写指

令而在锁存在第一和第二全局位线读出放大器块2705A和2705B中的4Kb数据中选择的数据。

接下来，参考图28，其说明了在单元区内存取模式下的存储装置操作。

假定如图28所示输入从RD0到RD7的8条读指令，每一条读指令标记过的内容就是在读数据的地方的单位单元区的字线。这里就是假定响应从RD0到RD7的8条读指令的字线的预定字线在第二条单位单元区（BL1）2720中。

在第一个定时周期 t_0 ，响应第一个输入的读指令RD0的逻辑单元区地址被转换为物理单元区地址。每一个定时周期的斜线部分就是当逻辑单元区被转换为物理单元区地址时的时间集。在这里，假定被转换后的物理单元区地址选中的单位单元区和用逻辑单元区地址选中的相同。

第一个单位单元区BL0的字线WL0被响应第一个读指令RD0而转换的物理单元区地址激活。然后，储存在响应激活的字线WL0的4K个单位单元中的数据被包含在第一个单位单元区BL0中的局部读出放大器区读出并放大。

然后，这4Kb被局部读出放大器读出并放大的数据就要移动到第一个和第二个全局位线读出放大器区2705A和2705B并因此通过全局位线2707储存。从4K位储存在作为对第一条读指令CD0的响应而输入的列地址所选的第一个和第二个全局位线读出放大器区2705A和2705B中的数据将输出到输入/输出读出放大器区2680A。在这里，数据D0在第二个定时周期 t_1 内通过输入/输出读出放大器区2680A和数据输入/输出缓冲区2690输出到存储装置的外围电路

在第一个定时周期 t_0 内，第二个单位单元区BL1的字线WL0也被读出为字线WL0的预定的字线。

然后，局部位线读出放大器区和位线在第二个定时周期 t_1 内也被强迫预充电，在作为对第一条读指令的响应被读出并放大后，储存在第一

个单位单元区BL0的局部位线读出放大器区中的4Kb数据就消失了。

于是，第一个单位单元区BL0的字线WL1就被作为对第二条读指令RD1响应的地址激活了。储存在作为对激活的字线WL1的响应的4Kb单位单元就本地被包含在第一个单位单元区BL0的位线读出放大器区中读出并放大。

换句话说，在此定时周期 t_1 ，假定这4K位数据是由第一条读指令RD0所储存的，这4K不锁存在第一个和第二个全局读出放大器区2705A和2705B中的数据，就被移动到并且储存在作为对指定为字线WL0的预定字线的第二个单位单元区BL1的响应的4Kb单位单元中。数据就通过全局位线2705B移动，并且在图28中所示的“INTWO”就表示了前述的过程。

也就是说，当这4Kb作为对第二条读指令RD1响应的数据在第一个单位单元区BL0被读出并且放大的时候，这4Kb锁存在作为对第一条读指令RD0响应的第一个和第二个全局位线读出放大器2705A和2705B中的数据就被移动到第二个单位单元区BL1的局部位线读出放大器区，并且响应字线WL0的4Kb单位单元被储存。

偶尔，第二个单位单元区BL1的字线WL0就是逻辑第一个单位单元区BL0，而第一个单位单元区BL0的字线WL0被指定为字线WL0的预定字线。这里就单独更新在预定的单元分区表2710和标志区2730的相关信息，并且在图26中显示“EBT_UP1”来说明该更新操作。

然后，作为对第二条读指令RD1的响应，这4Kb被读出并且放大的数据就被移动并且寄存在第一个和第二个全局位线读出放大器区2705A和2705B中，而从寄存在作为对第二个读指令RD1决定的列地址响应而选出来的数据D1则被输出到输入/输出读出放大器区2680A上。

于是，在第三个定时周期 t_2 ，作为对第三条读指令RD2的响应，第一个单位单元区BL0的字线WL2被激活，这4Kb响应激活的字线WL2而移动到第一个和第二个位线读出放大器区2705A和2705B的数据也因此在他们被读出和放大后锁存在这里。

换句话说，关于第二个读指令RD1的恢复的执行发生在响应第二个单位区BL1的字线WL1的4K个单位单元中，并且在图28中以“INTW1”来表示。这时就在这里更新预定的单元分区表2710和标志区2730，而“EBT_UP2”则表示该更新的执行。

在第四个定时周期t3，关于第三个读指令RD2的恢复执行发生在响应第二个单位区BL1的字线WL1的4K个单位单元中，并且在图26中以“INTW2”来表示。这时就在这里更新预定的单元分区表2710和标志区2730，而“EBT_UP3”则表示该更新的执行。

换句话说，作为对第四条读指令RD3响应的第一个单位单元区BL0的字线WL3就被激活了。而这4Kb的响应已激活的字线WL3的数据就被读出并且放大。

然后，在第五个定时周期t4，关于第四个读指令RD3的恢复执行并不把数据移动到已经有了字线WL3的预定字线的第二个单位单元区BL1，但是发生在响应第一个单位区BL0的字线WL3的4K个单位单元中。

关于为何响应第五条读指令RD4的字线WL0马上被激活，有一个原因。那就是关于第五个读操作RD4作为下一个指令的执行发生在第一个单位单元区BL0。因此，必须更新预定的单元分区表2710，因为这次字线WL3的预定字线没有改变，并且用更新信号“EBT_UPN”来表示。

为了参考的连续性，如果在每个定时周期t4和t5内输入第五和第六个读指令RD4和RD5，如前所述，响应读指令的4Kb数据就在第一个定时周期被读出和放大，然后从这4Kb锁存的数据中选出一位来输出。然后，在第二个定时周期内，已激活的字线的预定的字线使得数据移动到指定的单位单元，并且也执行恢复操作。

参考图28， 存储装置以单元区内数据存取法操作。

尽管在每个定时周期内的虚线部件中，读指令实际上已经执行了，数据被移动并且储存在预定的恢复单位单元区内，这里还要在时间集内更新预定的单元分区表2710和标志表2730。更新信号“EBT_UP1”到

“EBT_UP5”表示标志区2730的更新，而更新信号“EBT_UPN1”到“EBT_UPN5”表示预定的标志表没有更新。

另外，斜线部分，例如“X”时间集，表示当有强制性预充电时通过连续用同样的单位单元区来读数据，并且把逻辑单元区地址转化成为物理单元区地址时，锁存在与前一指令相一致的单位单元区局部位线读出放大器区的时间集。

总之，当连续用同样的单位单元区读的时候，在第一个定时周期，与当前的实施例相一致的存储装置放大做为输入的读指令的响应的数据，在第二个定时周期移动和锁存到全局位线读出放大区，例如2705A和2705B，以及锁存在全局位线读出放大器区被移动到的那些字线中的预定的字线被读指令激活并锁存的单元区。

于是，当数据用同一单位单元区连续存取时，数据可以以高速来存取，因为下列的读指令没有经过数据恢复操作。

参考图29，这里说明了在一个单位单元区连续写数据的执行。

在第一个定时周期 t_0 ，响应第一个写指令WR0的输入的逻辑单元区地址被转化为物理单元区地址，并且第二个单位单元区BL1的字线WL0被激活。

在这里，尽管作为对第一个写指令WR0的响应的单元区地址选择第一个逻辑单位单元区BL0的字线WL0，实际上作为对第一个逻辑单位单元区BL0的响应的数据被储存在作为第二个物理单位单元去卡BL1的字线WL0的单位单元中。在图29中所示的“指令操作时间集”中，说明了在转换单元区地址后选中作为对写指令的响应的由单元区地址转换操作输入的的逻辑单位单元区字线和物理单位单元区的信息。

然后，作为对被激活的字线WL0的响应的4Kb数据被读出并被放大，移动到第一个和第二个全局位线读出放大器区2705A和2705B，并且在此寄存。

接着，响应被作为第一条写指令WR0的响应而输入的列地址所激活的

字线WL0的4Kb数据中选择的一位数据就被删除了，而从外部电路输入的作为对写指令WR0的响应的数据D0就代替出去的数据被锁存起来。

在第二个定时周期t1中，被第一个写指令WR0读出并且放大的4Kb数据在响应第二个单位单元区BL0的字线WL0的4K个单位单元中复原。因此，除了在读出放大和寄存数据后用外部电路输入的数据代替锁存的数据外，写操作和读操作基本上是相同的。

换句话说，在这个定时周期t1中，逻辑单元区地址被转换为物理单元区地址以响应第二个写指令WR1，而为了与转换的物理单元区地址相一致，第三个单位单元区BL2的字线WL1被激活。于是，这4Kb响应激活的字线WL1的数据就被读出，并放大和寄存在第一个和第二个全局位线读出放大器区710和730中。

接着，响应第二个写指令WR1而输入的列地址所选中的数据就被清除，而作为对第二个写指令WR1的响应而从外部电路输入的数据D1就代替清除的数据而被锁存起来。

在第三个定时周期t2，这4Kb锁存在第一个和第二个全局位线读出放大器区2705A和2705B中的响应第二个写指令WR1的数据就被存储在4K个与第三个单位单元区BL2结合的单位单元中。

换句话说，为了响应第三个写指令WR2，要激活第二个单位单元区BL1的字线WL2、那么这4Kb响应激活的字线WL2的数据就被读出并放大，然后寄存在第一个和第二个全局位线读出放大器区2705A和2705B中。

接下来，就要清除由响应第三个写指令WR2而输入的列地址选中的一个数据，并把为了响应第三个写指令WR2而从外部电路输入的数据D0代替那个清除的数据而寄存起来。

在第四个定时周期t3，这4Kb寄存在第一个和第二个全局位线读出放大器区2705A和2705B中的响应第三个写指令WR2的数据就不存储在第二个单位单元区BL1中，而是存储在4K个响应指定为字线WL2的预定字线的第一个单位单元区BL0的单位单元中。

这时候，字线WL2的预定字线就把第一个单位单元区BL0变成了第二个单位单元区BL1的字线WL2，而关于此的信息就在预定的单元分区表2710中更新。标志表2730也会更新，因为关于逻辑单元区地址的更改信息和这个都在“EBT_UP1”中做了说明。

这次之后，字线WL2的预定字线就是第二个单位单元区BL0的字线WL2。第二个单位单元区BL1的字线WL2就是上一步中的逻辑单元区地址“BL0”的字线WL2，而第一个单位单元区BL0的字线WL2就是这次之后的逻辑单元区地址”BL0“。

此后，关于第四个写指令WR3的操作就跟关于第一个写指令WR0的操作相同。

那么，当第五条写指令WR4和第六条写指令WR5被执行的时候，与上面操作类似，由于必须在第一单位单元块 BL0中连续写入数据，这4Kb的响应第五条写指令WR4的锁存数据在第六个定时周期 t_5 就恢复到响应指定为字线WL0的预定字线的第二个单位单元区BL1的字线WL0的4Kb单位单元中，图29中所示的“INTW1”就说明了这个过程。

如图29所示虚线部分的激活时间集说明了当响应先前指令的数据被移到那些字线指定为预定的字线的单位单元区中，并且在此复原。因为响应下一条指令的数据要写入同一个单位单元区中。

总之，当数据被写入每一个单位单元区中时，本实施例的存储装置使用单元块交叉存取模式写入数据，并且在当数据被连续地写入同一个单位单元区中时，通过把数据移动到那些字线被指定为预定的字线的单位单元区恢复数据。因为存储装置有256个指定字线，所以通过包含除了响应输入的单元区地址的单位单元区以外的另一个单位单元区就是可能的。

因此，在写模式下，数据可以以高速写入，因为下一条写指令不用依赖当前数据的恢复时间就可操作。

另外，也可以高速的写数据，因为数据恢复时间总是被减少到不用

依靠写数据模式，不管数据是被连续的写入同一个单元单元区还是写入每一个单元单元区中。

图30是一个显示了在读模式下的单元区内的数据存取操作的模拟波形。

如图所示，由于当读指令RD0到RD2和RD4到RD5被激活时，后面的读指令连续在同一个单元单元区内读数据，所以关于读数据的恢复操作在其字线被指定为预定字线的单元单元区中发生，并且总共发生了5次的数据传送和恢复操作，例如图30中所给的“INTW”。也由于当从读指令RD3，RD6和RD7执行时下一条指令存取其他的单元单元区，如图30中所示的“ACT”数据恢复操作，就会发生在一个选中的单元单元区中。

图31是说明一个产生用于全局位线连接单元，例如图26中的2718，2728，...和2758上，产生连接控制信号的原理电路图。

由于图29中描述的是一个用于控制全局位线连接单元，例如，2718的电路，第一个或非门NOR1接收信号PCG-ACT[2n] 和PCG-ACT[2n+1]来读出和放大响应指令选中的字线的数据，然后另一个或非门NOR2接收信号INTW[2n] 和 INTW[2n+1]，以便在同一个单元单元区被连续存取时，来移动和恢复数据，并且输出一个全局位线控制信号GBIS来控制全局位线连接单元。 τ_1 ， τ_2 ， τ_3 表示被从Delay1到Delay3的每一个延迟块所延迟的时间。

图32是一个表示响应从图31中的电路图所产生的信号的存储装置操作时序图

图32中显示的斜线部件表示当在每个定时周期内的逻辑单元区地址被转换成物理单元区地址时的时间。图32中所示的延迟时间 τ_1 ， τ_2 ，and τ_3 被图31中的延迟块Delay1到Delay3确定。

图32显示的“LSA_EN”是一个局部位线读出放大器的使能信号，“GBIS”为一个全局位线读出放大器的使能信号，“GSA_EN”是一个全局读出放大器的使能信号

当下一个激活的第二指令在同一个单位单元块内存取被第一指令激活的字线时“ Δ ”被强制失效，并且“INTW”表示响应第一个指令预定的字线就移动数据到那些位线被指定为的预定字线并且在那里锁存数据的单位单元区。在如图32所示的在本发明的存储装置中，可以理解单元区地址转换时间和强制性预充电时间占一个时序周期的40%。

参考上述说法，为了实施根据本发明的存储装置，在段内的每个单元区域，存储装置必须包含一个单位单元区，预定的单元分区表，标志区，等等。另外，因为全局位线读出放大器和全局位线被进一步包含，除了典型的存储装置以外还有很多附加的区域。

但是，当前制造的存储装置可以高速存取比先前的存储装置多的东西，因为存储装置的执行时间为了数据的恢复而通过增加上面规定的区域获得大幅度降低。

假定存储装置的行周期是 t_{RC} ，根据本发明的行周期是 $\text{MAX}\{0.5 \times (t_{BAT} + t_{RP} + t_{RC}), t_{INTW}\}$ 。也就是说，在“ $0.5 \times (t_{BAT} + t_{RP} + t_{RC})$ ”和“ t_{INTW} ”中，哪一个的时间更长就决定了存储装置的行周期时间。

在这里，“ t_{BAT} ”就是单元区地址的转换时间，“ t_{RP} ”是预充电时间，“ t_{RC} ”是前一个行周期时间，“ t_{INTW} ”则是通过将内部寄存数据移动到其字线被指定为预定字线的单元块的数据复原时间。

在这里，为了实施优选实施例，由于“ t_{RP} ”是强制预先充电，所以当每一个指令被操作时就会发生。为了在数据在同一个段内为单元区存取时，马上可以存取下一个数据。这意味着，在前一个时间读出和放大数据，禁止局部位线读出放大器，并给局部的位线预充电。

根据本发明的存储装置的行周期时间是由哪一个时间更长决定的存储装置的行周期时间而确定的，并且典型的，一般会确定为“ $0.5 \times (t_{BAT} + t_{RP} + t_{RC})$ ”，因为 $0.5 \times (t_{BAT} + t_{RP} + t_{RC})$ 大于“ t_{INTW} ”。

如果存储装置的行周期时间是“ $0.5 \times (t_{BAT} + t_{RP} + t_{RC})$ ”，因为数据

恢复时间的缩小尽管进一步增加了强制性预充电时间和当逻辑单元区地址被转换位物理单元区地址时，行周期时间就被减少了大约30%到40%，参考图28和图29中给出本发明的行周期时间与传统的存储装置的常规行周期时间相比的情况。

例如，如果一个指令执行了15纳秒，在常规的存储装置中，行周期时间是两个定时周期，也就是30纳秒。然而，在本发明中，假定单元区转化时间时3纳秒，强制预充电时间为3纳秒，行周期时间是21纳秒，其是一个定时周期15纳秒加上6纳秒的附加时间。

换句话说，如果被下一条指令存取的单位单元区与前一条指令存取的单位单元区等同的，那么根据优选实施例的存储装置的单位单元就移动数据，并且在数据移动时更新标志。

但是，为了实施本发明的存储装置，激活在当前时序存取的单位单元区的字线和移动到包括所选字线的预定字线的单位单元区，并且二者可以在存储装置上同时发生。在这种情况下，数据就总是储存在两个单位单元区中，储存实际数据的单位单元区由标志表决定。在这种情况下，如果在单位单元区中读出并被放大的数据就寄存在全局位线读出放大器区上无论何时只要指令被执行了，存储装置就被强制实施对寄存在单位单元区的局部位线读出放大器区预充电。

另外，在优选的实施例中，存储装置在第一时间读出和放大接收数据之后的寄存时间输出数据，但是不可能实施存储装置的响应数据恢复的指令操作，当指令有效时，已经是第二时序了。

从上述方法中适当地根据存储装置的设计技巧选择一种，数据就可以高速存取访问但却与使用上面地存储装置方法中哪一个来实施数据恢复时间无关。

根据优选实施例的存储装置以高速执行是可以的。因为数据存取时间地减少，但是增加了 t_{BAT} 转换逻辑单元块地址为物理单元快速之，以及先前的存储装置中所没有的强制预充电时间 t_{RP} 。

为了在本发明下高速执行，提供的条件是具有行周期的存储装置要减少，以用于强制预充电时间 t_{PR} 以及来自图26中所示的存储装置的单元块地址的转换时间 t_{BAT} 。

图33是一个说明一个减小的行周期时间的存储装置的方框图。

如上所示，本发明的存储装置直接从控制块3380接收指令CD，并且产生第二条指令CD_D，延迟指令CD以满足强制充电时间 t_{RP} 和从延迟块3382输出地单元块地址转换时间 t_{BAT} 。控制块3380把逻辑单元块地址转换为物理单元块地址并且执行强制性预充电，延迟地指令CD_D控制单元区域3320来执行其他在数据存取过程中需要地操作。

图34是一个详细说明在图33中的存储装置的方框图。

如图所示，存储装置包括一个单元区域包括了 $N+1$ 个单位单元块，即 N 个单位单元区，有 M 个字线相应输入的行地址和有 M 个字线地辅助单位单元区，以及一附加的单位单元块，也有 M 个字线；一个数据存取控制块来控制被储存在一个在 $N+1$ 个单位单元区地第一个单位单元区或者另外的单位单元块中；一个指令控制块，控制以下操作：响应处于有效的下一时序周期的第二指令来转换单元块地址，在响应第一当前有效指令存取数据期间对响应第一指令的位线强制性预充电。当在第一单位单元块中连续存取第一和第二数据时，存取控制块控制如下操作：不在第一单位单元块中复原第一数据而是在第一单位单元块中存取第二数据并且在第一单位单元块中复原第一数据。

当一个数据可以存贮在本地的单位单元区中，数据存取控制块控制那些在多余的数据操作单位单元区内可以存取多余的数据的操作。当第一和第二个数据可以在第一和第二单位单元区间依次操作。

图35是一个时序图，它说明了在图34中列出来的可以响应不断输入的读操作的存储装置。在此，请参考图33和图35，里面有存储装置操作的说明。

首先，该存储装置将响应第一个输入的读指令RD0的逻辑单元地址转

换为物理单元，并且执行强制性预充电操作。这里，如果响应先前指令存取的单位单元区和响应现在生效的读指令RD0一样，预充电操作就不执行恢复先前读指令的操作，但是操作强制预充电操作的执行。然而，像这样，也有在存储装置的第一个读指令执行实际的强制性预充电操作，因为没有先前的操作。

于是，当本发明的存储装置执行第一条指令时，存在一个等待时间，例如，图35页的“X”时间集，以用于单元区地址的强制预充电时间 t_{PR} 和转变时间 t_{BAT} 。

换句话说，指令时间控制块3450的延迟块3382产生第一个为了强制性的预充电时间 t_{PR} 和转变时间 t_{BAT} 而延迟的指令RD0_D。

然后，在第一定时周期 t_0 ，数据存取控制块3380被控之来激活第一个单位单元区BL0的字线WL0来响应从指令控制块3420输出的第一个延迟指令RD0_D。响应激活字线WL0 4Kb的数据就被局部的位线读出放大器读出并放大。

那么，这4K被读出和放大的数据就被移动并锁存到数据寄存区3470。从4K位读出并且放大的数据中选出的一个数据D0就被输入的列地址以响应第一个读指令CD0，该数据还将输出到输入/输出读出放大器区，参见表26的2600。这里，如前所述，输出数据D0通过输入/输出缓冲器等输出到存储装置的外部电路。

另一方面，在第一个时间 t_0 内，指令控制区3420接受到第二条读指令RD1并把逻辑单元区地址转化为物理单元区地址来响应第二个读指令。然后，指令控制块3420比较第一条读指令RD0和被第二条读指令RD1存取的单位单元分区的数据。这里，因为数据在同一个单位单元区被第一条读指令RD0和第二条读指令RD1读出，所以在被读出和放大以后，作为响应第一条读指令RD0，需要强制性的预充电这4Kb的寄存在局部位线读出和放大器分区上的数据，例如，在图35的“Y”时序集。

另外，在“Y”时序集，指令时间控制区3450产生第二条延迟指令

RD1_D, 延迟第二条读指令RD1来强制预充电时间 t_{PR} 和转变单元分区地址的时间 t_{BAT} 。

那么, 在第二定时周期 t_1 内, 第一个单位单元分区BL0的字线WL1就作为对第二个延时读指令RD1_D的响应而被激活。而这4Kb响应生效的字线WL1的数据就被局部的位线读出和放大, 然后移动到数据寄存区3470。然后从这4Kb寄存在数据寄存区3470中的响应第二个读指令RD1的数据中选择一个D1并将其输出到输入/输出读出放大器。

换句话说, 在这个定时周期 t_1 中, 要激活第二个单位单元区BL1的字线WL0, 该字线被指定为响应第一个读指令而激活的字线WL0的预定的字线。锁存在数据锁存区3470的数据就移动到响应第二个单位单元区BL1的4K个单位单元中, 并寄存进去。“INTWO“说明了该操作。

然后, 在第二个定时周期 t_1 , 第三个读指令RD2被输入到指令控制区3420, 响应第三个读指令的逻辑单元区地址被转换成为物理单元区地址。并且, 指令的时间控制区3450把第二个读指令RD1和这组有数据可以被第三个指令RD2存取的单元区进行比较。这里, 因为数据被第二条读指令RD1和第三条读指令RD2在同一个单元区读出, 这4Kb的锁存在局部位线读出放大器分区上的数据在被读出和放大后响应第二条读指令RD1被强制性的预充电, 例如, 图35中的“Z”时间集。另外, 在“Z”时间集中, 指令时间控制块3450产生第三个延迟指令RD2_D延迟第三个读指令RD2以用于强制性预时间 t_{RP} 和单元区地址的转换时间 t_{BAT} 。

在第三个定时周期 t_2 , 第一组单元区BL0的字线WL2就被第三个延迟的读指令RD2_D激活, 该指令被延迟以用于强制充电的时间 t_{RP} 和延迟区440开始的单元区地址的转换时间 t_{BAT} 。

响应激活字线WL2的4Kb数据就被局部的位线读出放大器读出并且被放大, 然后移到数据寄存区3470。这时候就由响应第三个读指令RD2的列地址从这4Kb寄存在数据锁存区3470的数据中选出一个数据D2来输出到输入/输出的读出放大器上。

换句话说，在这个定时周期 t_2 中的第二个单元区BL1上有一个激活的字线WL1，这个字线已经被指定为响应第二条读指令RD1有效的预定的字线。而寄存在数据寄存器3470中的数据就要移动到4Kb个响应第二个单元区BL1上的字线WL1的单元区，并在此锁存。“INTW1”说明了该操作。

如前所述，图34中所显示的存储装置的操作和图26中相似。但是除了在存储装置使用输入的读指令来转化单元区地址，完成激活字线相关的操作，在单位上的读出和放大，储存等等有所不同以外，两者基本类似。而在执行了强制性预充电以后也因此延迟了读指令RD_D为强制性预充电时间 t_{RP} 和单元区地址的转化时间 t_{BAT} 。

这样，在存储装置的初始化操作时，行周期时间就有了具有强制性预充电时间以及响应第一激活的读指令的等待反映时间，例如，图35中的“X”，但是由于第二读指令被激活，该等待时间减少以用于强制性的预充电时间 t_{RP} 和单元块地址转换时间 t_{BAT} 。

有个原因，那就是行周期时间不受影响，因为响应下列读指令的操作，包括强制性充电，和单元区地址转换，会在数据读出和放大已响应先前的读指令。

在上面的说明中，说明了图34中的读操作，关于写操作被忽略的原因是因为写操作和读操作很接近。也就是说，在实施写的过程中，不用从锁存在数据寄存区3470 中选出要输出的数据，因为对预存储装置来说，用从外部电路输入的数据覆盖锁存在寄存区3470中的数据是不同的。

图36是一个时序图，它说明了显示在图39中的存储装置，同在图18中显示的存储装置设备的操作进行比较，为了易于理解根据本发明的存储装置的操作。

如图所示，第一个指令CD0在a0时序集中操作来转换单元区地址和强制性预充电，将第一指令CD0延迟‘ $t_{RP} + t_{BAT}$ ’ 的第一延迟的指令CD0_D操作，以在b0时序集中激活字线并读出/放大响应激活字线的的数据。并且下一个时序操作的第二指令CD1被输入，以在b0周期的a1时序集

中转换单元区地址，然后实施强制性预充电。在b1时间集中，将第一个指令CD0延迟“ $t_{RP} + t_{BA}$ ”的第二个延迟的指令CD1_D操作，以激活字线以及读出/放大数据来响应对在b0时序集中激活的字线。

因此，在图34中显示的行周期时间为 $\text{MAX}\{0.5 \times t_{RC}, t_{INTW}\}$ 。这一次，“ t_{RC} ”时间前一个行周期b0+b1，而“ t_{INTW} ”是执行以下操作的时间，响应一个激活一个字线的预定的字线来作为对操作指令的，并把数据移动到那些字线都从数据锁存器指定了预定的字线的单元区，并在那里复原。

如前所述，在图34中所示的存储装置的行周期时间 $\text{MAX}\{0.5 \times t_{RC}, t_{INTW}\}$ ，消除了前述中描述的存储装置的 $\text{MAX}\{0.5 \times (t_{RC}, +t_{RP}+t_{RC}), t_{INTW}\}$ 中的用于转换单元块地址的‘ t_{BAT} ’和用于强制性预充电的‘ t_{RP} ’。

因为“ t_{INTW} ”典型的比“ t_{RP} ”要长很多，行周期时间大体上就是 $0.5 \times t_{RC}$ ，于是如图34中所示，把存储装置的行周期时间缩短到最大值的一半也是可能的。也就是说如图36所示，现有技术的存储装置的行周期时间是b0+b1，但是根据本发明的是b0。

在本发明当中，实现了以高速存取数据的存储装置，其免除肯定会占用使用电容器作为存储器单元元件的存储装置的大部分数据存取时间的数据复原时间。

另外，还要提供存储装置的方法和设备以便高速数据存取，而与数据在一个单元区内连续存取还是在每个单元区内轮流存取这样的数据存取模式无关。所以，在包括本发明的存储装置的系统，系统总的速度由于与存取模式无关地获得大幅提升。

本发明已经描述了特殊的实施方式，很显然对于那些熟悉本领域技术的人员来说，可以在不违背本发明权利要求的精神和范围的情况下，作出各种修改和更正。

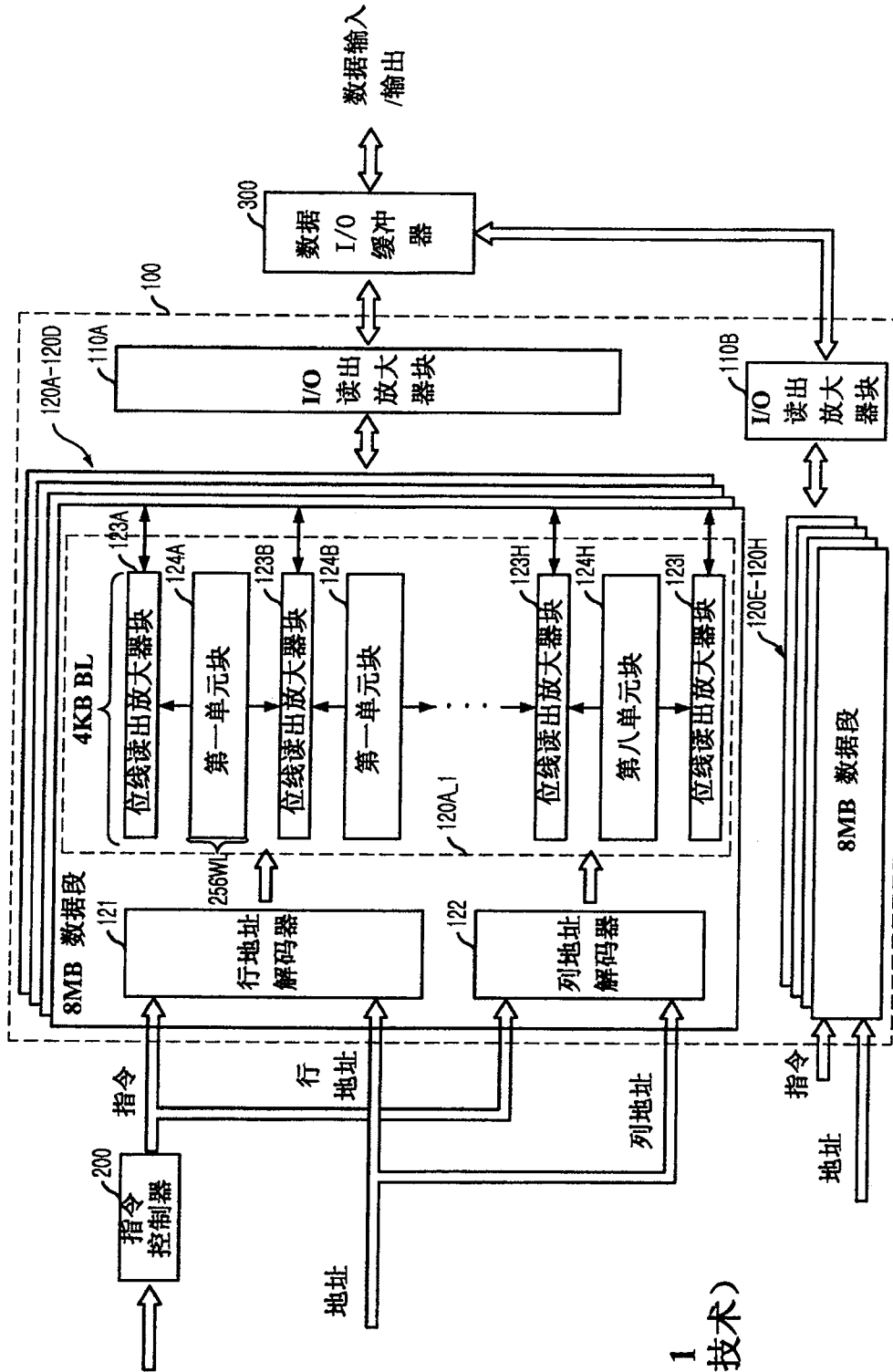


图 1
(现有技术)

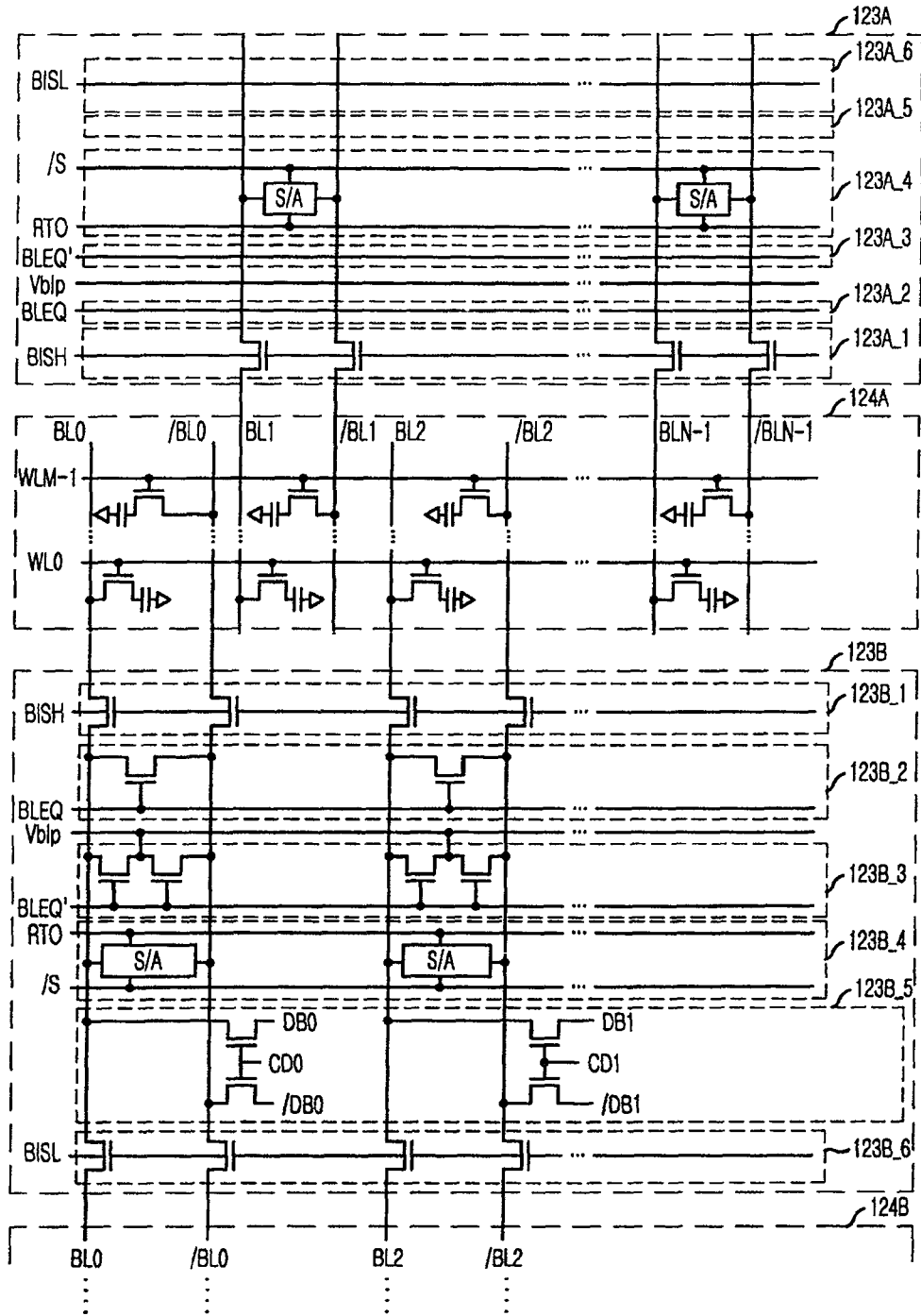


图 2
(现有技术)

图3
现有技术

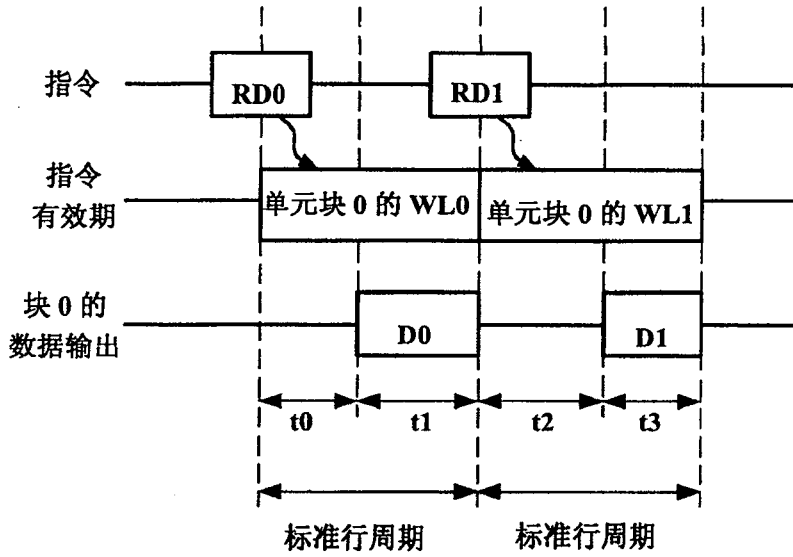


图4
现有技术

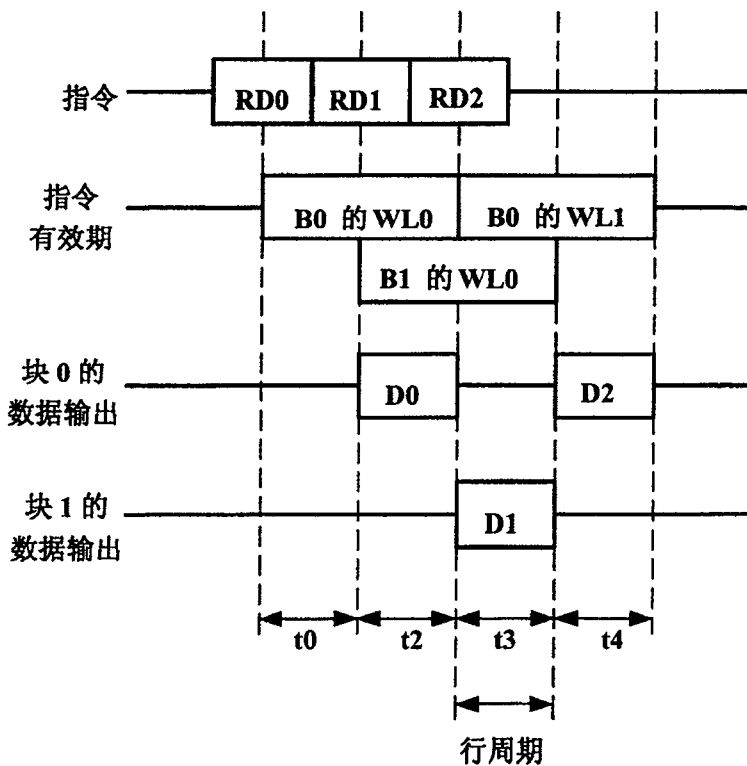


图 5

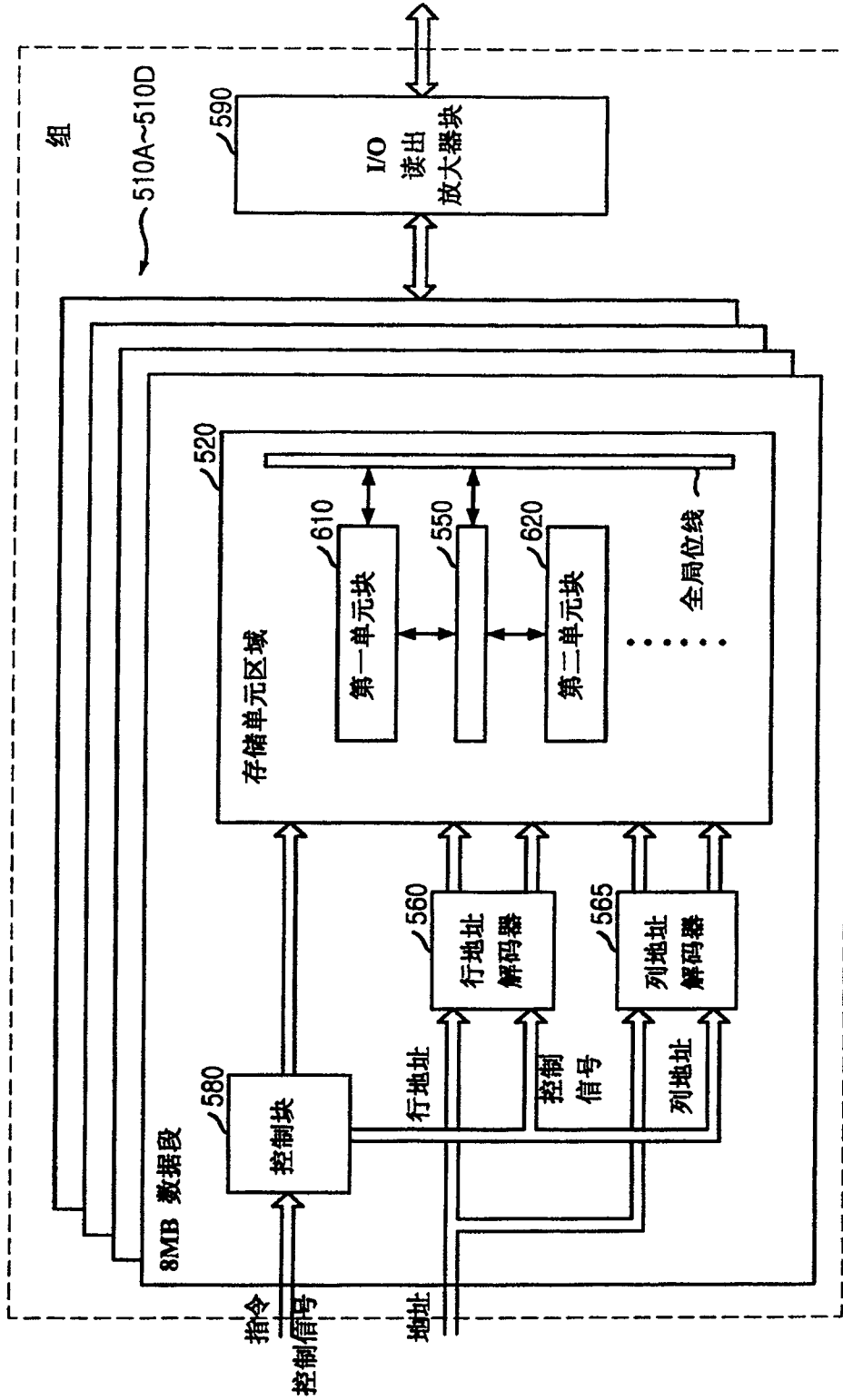


图 6

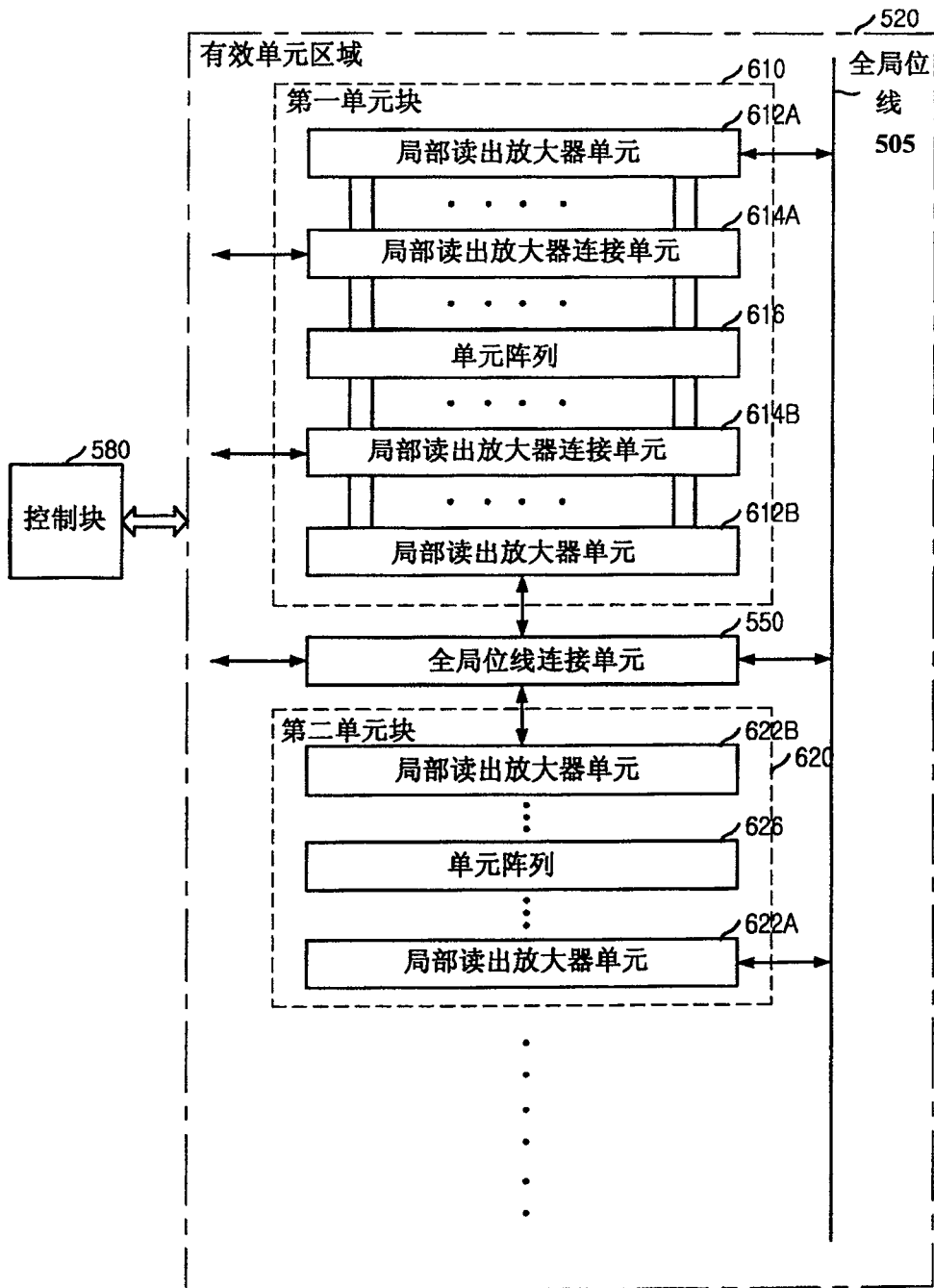
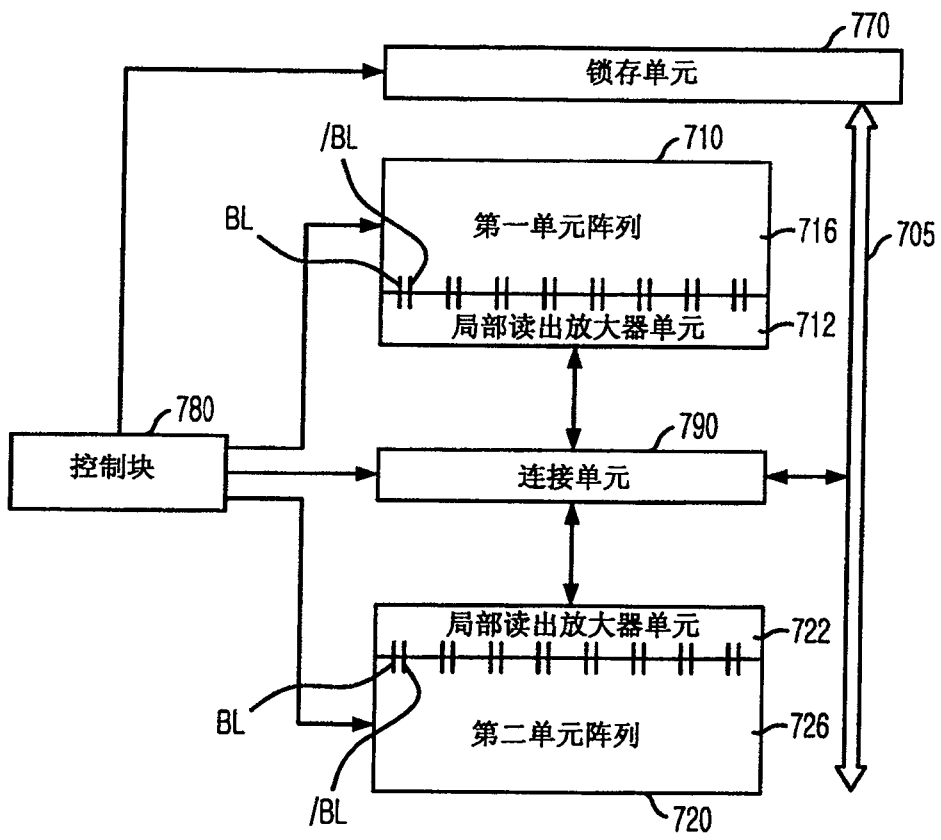
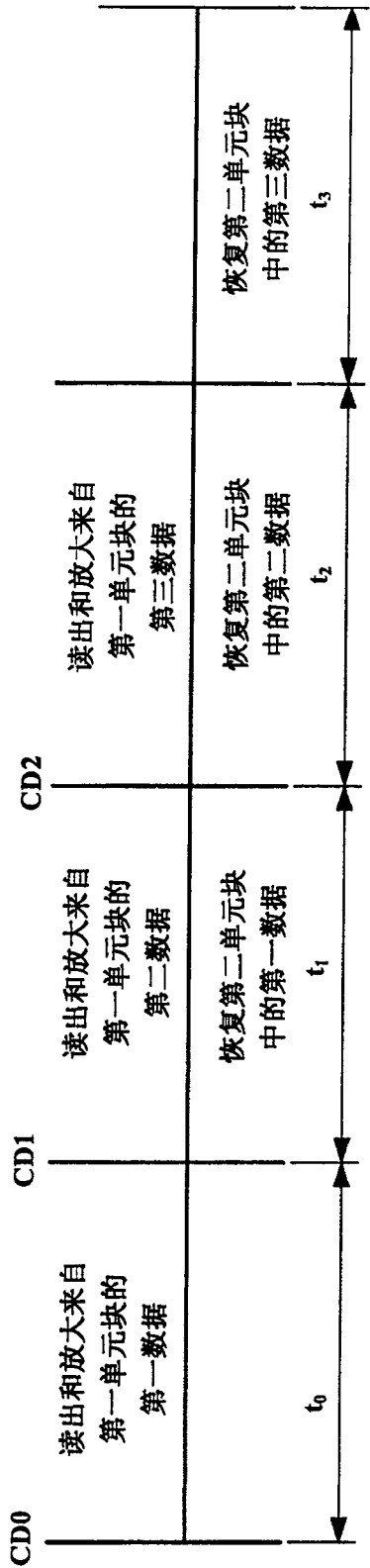


图 7





按照内单元块交织模式的行周期

<内单元块数据存取>

图 8

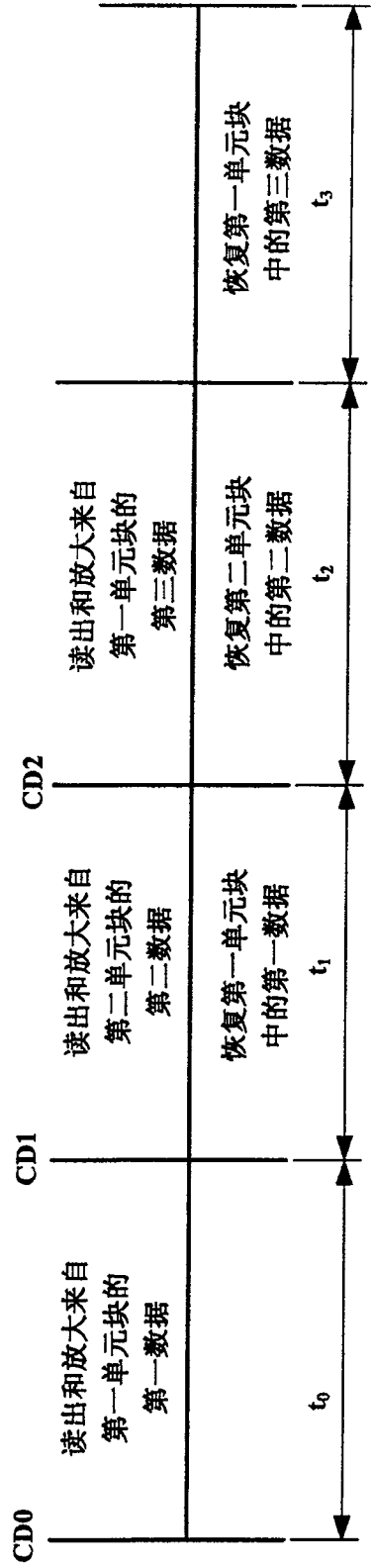


图 9

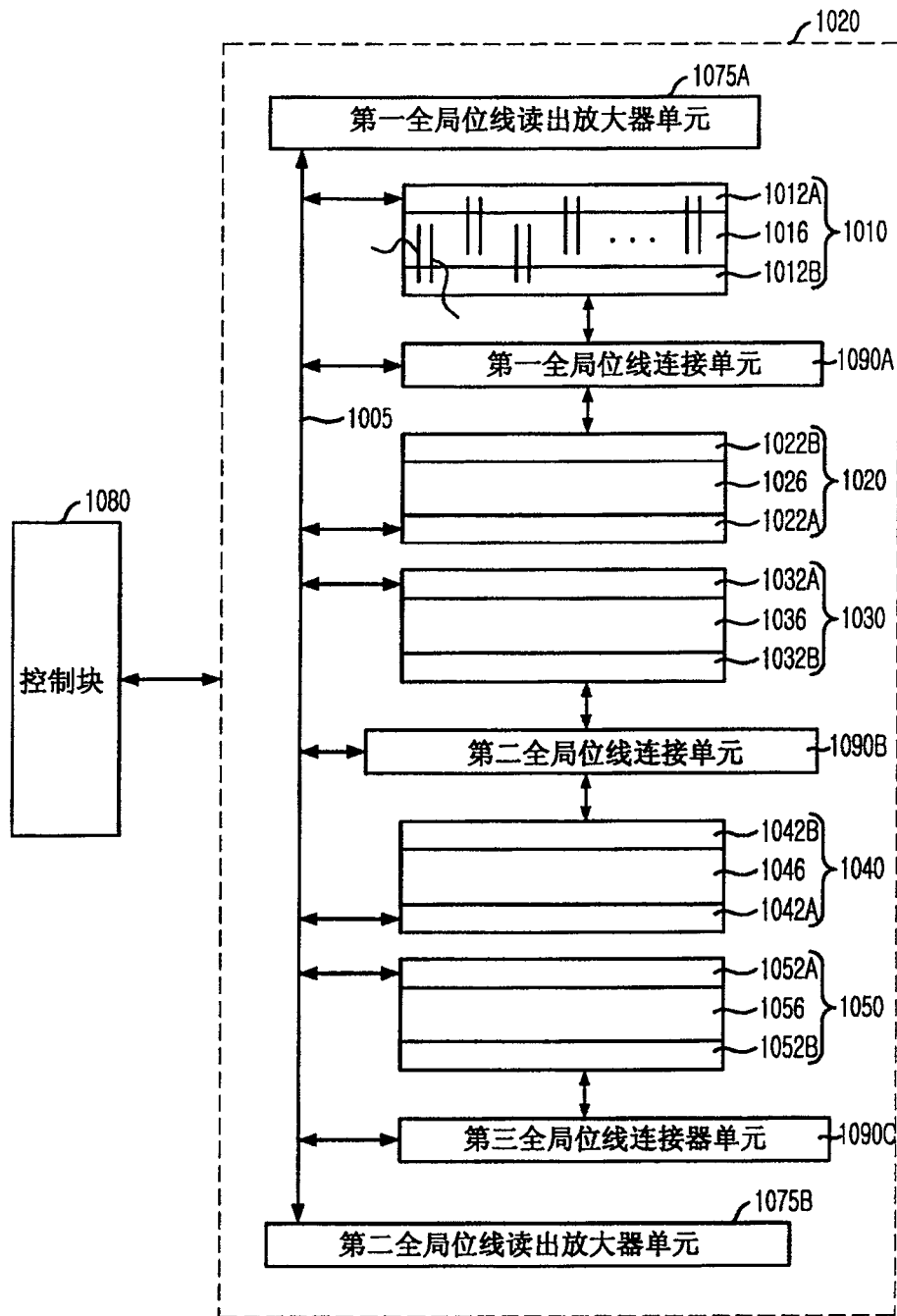
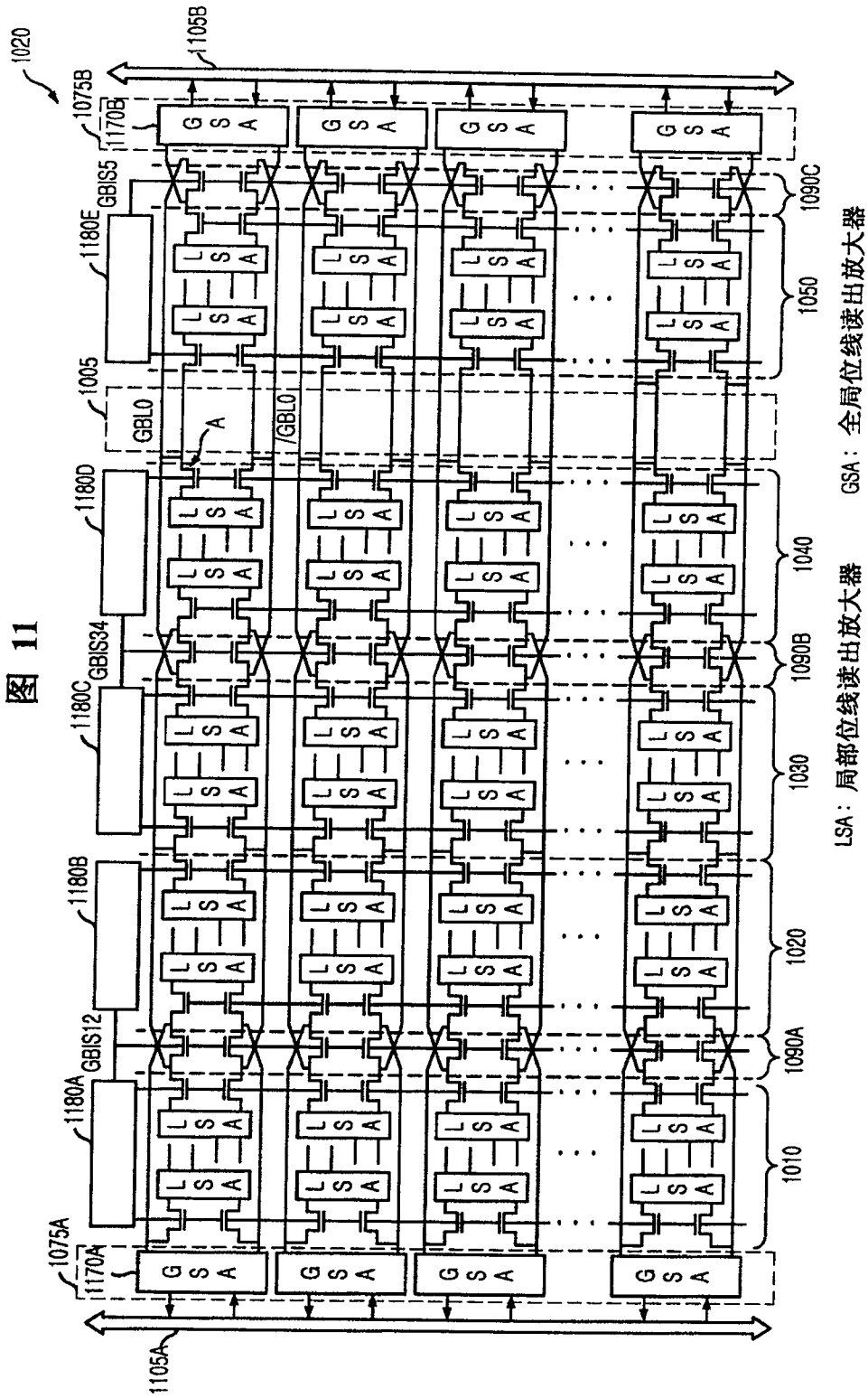


图 10



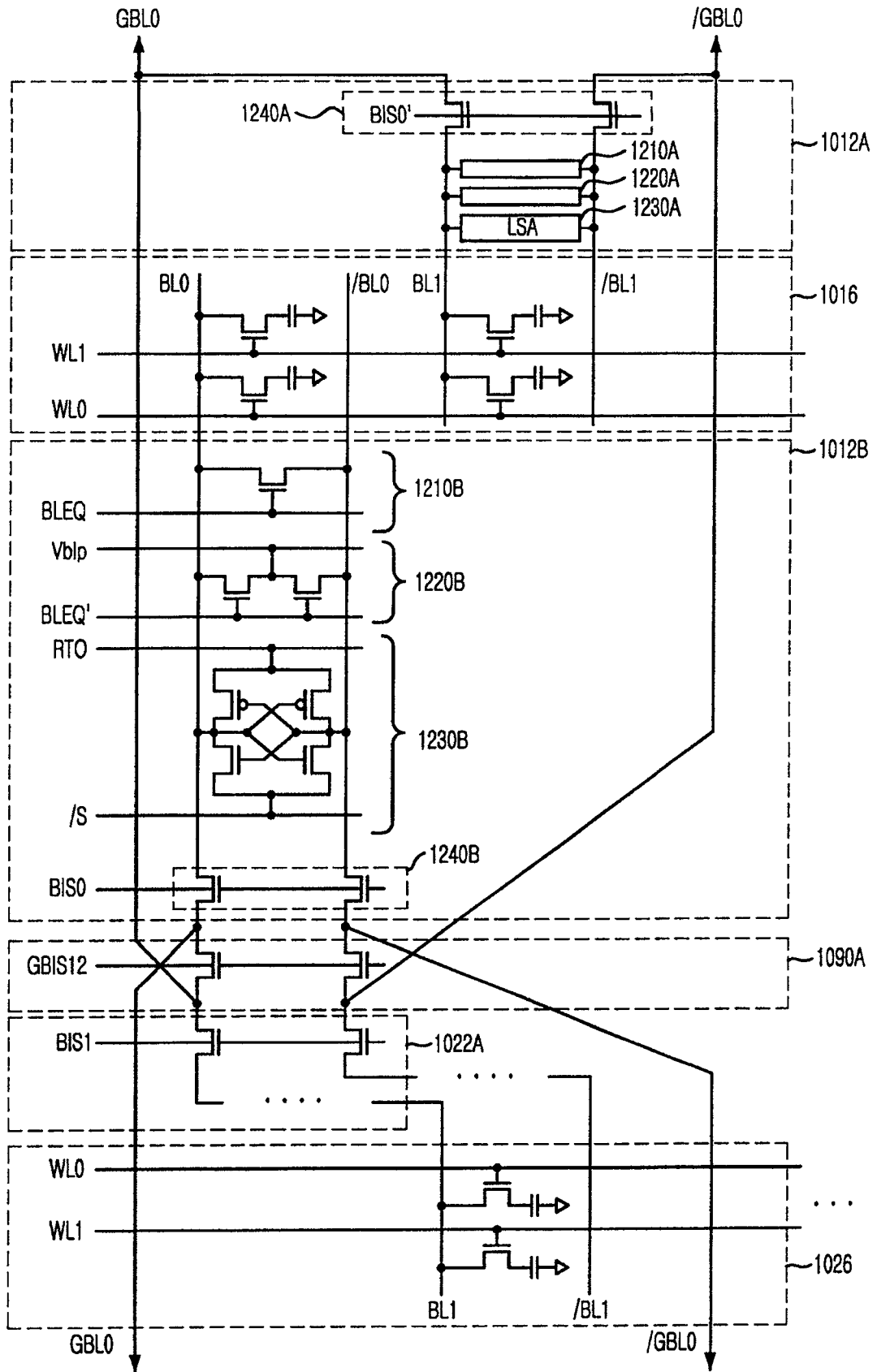


图 12

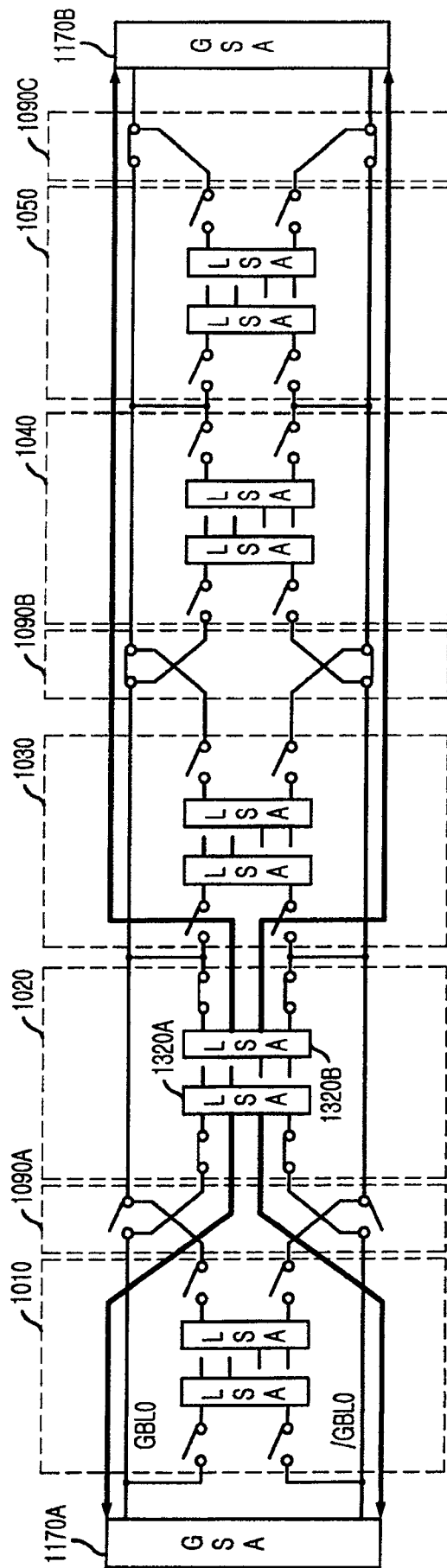


图 13A

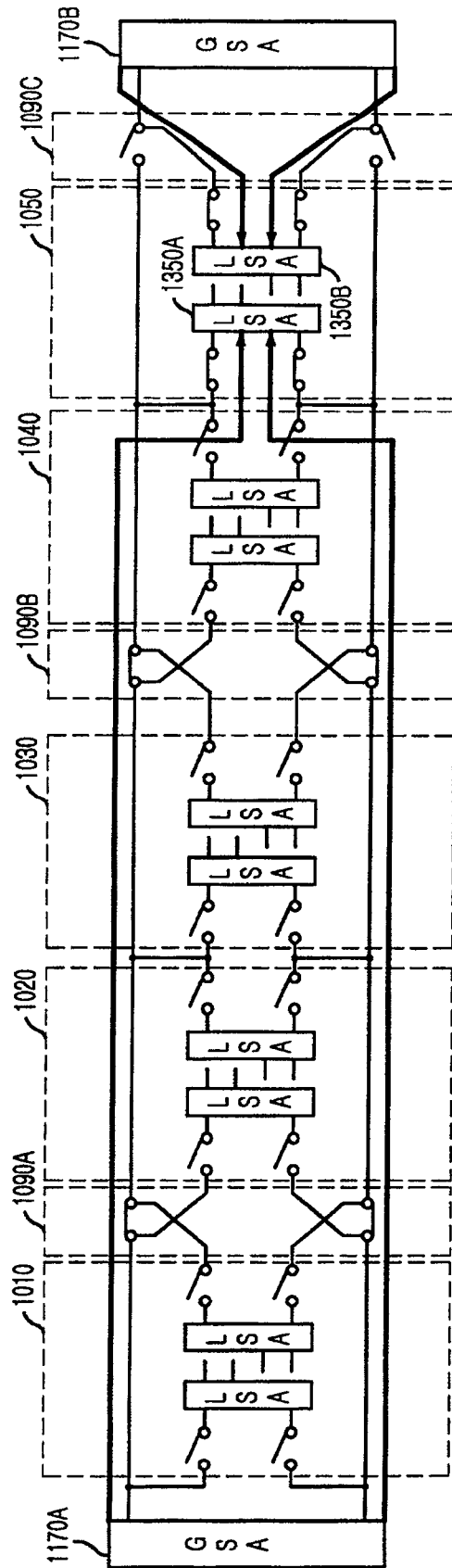


图 13B

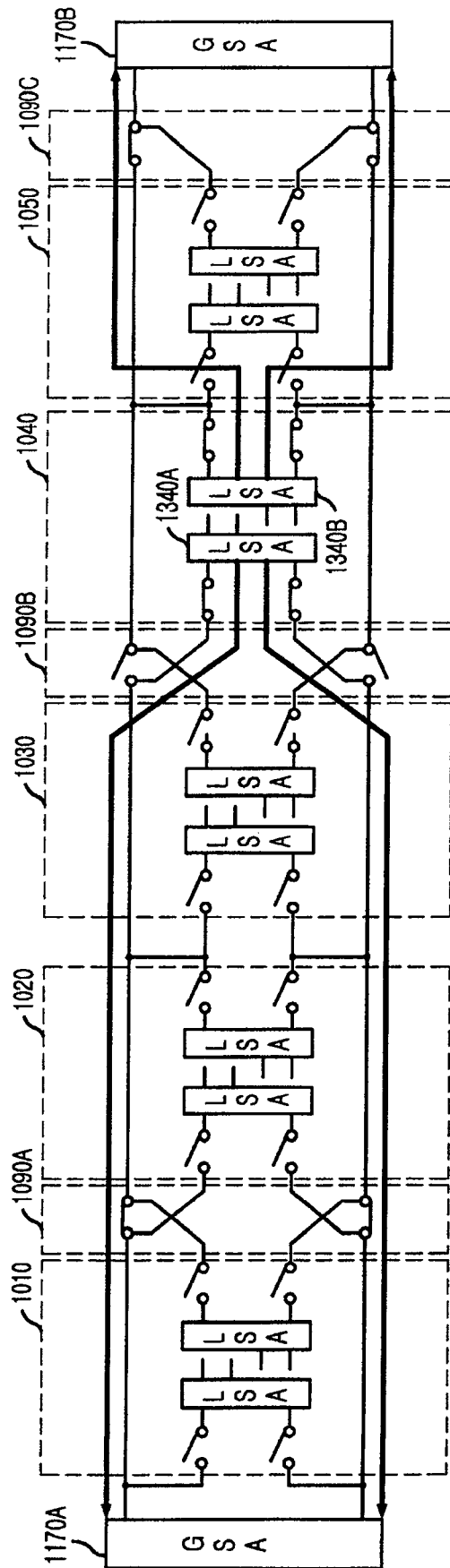


图 13C

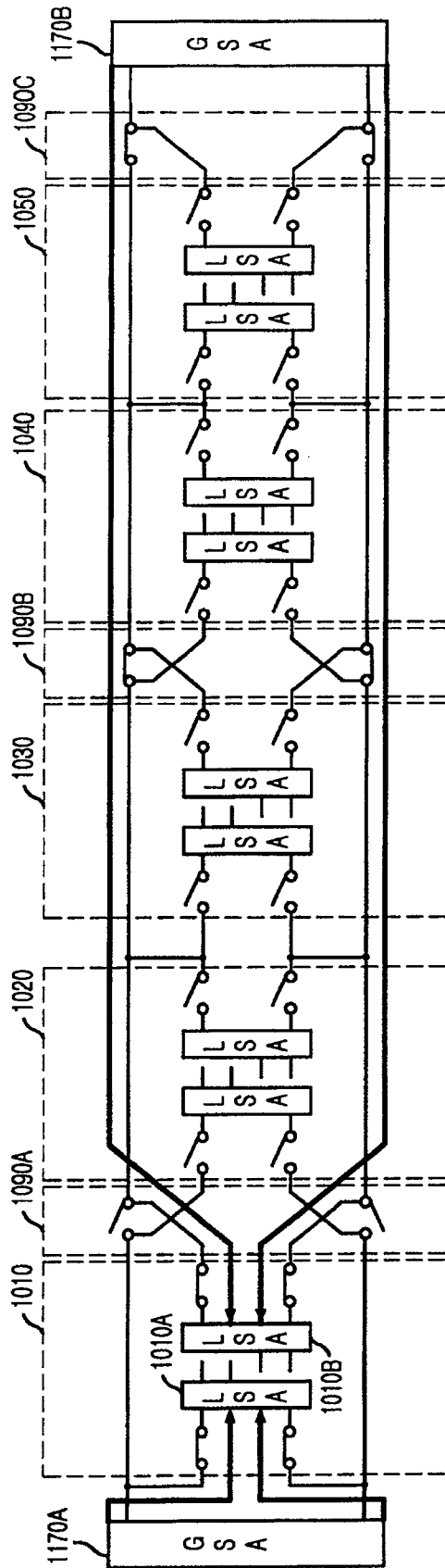


图 13D

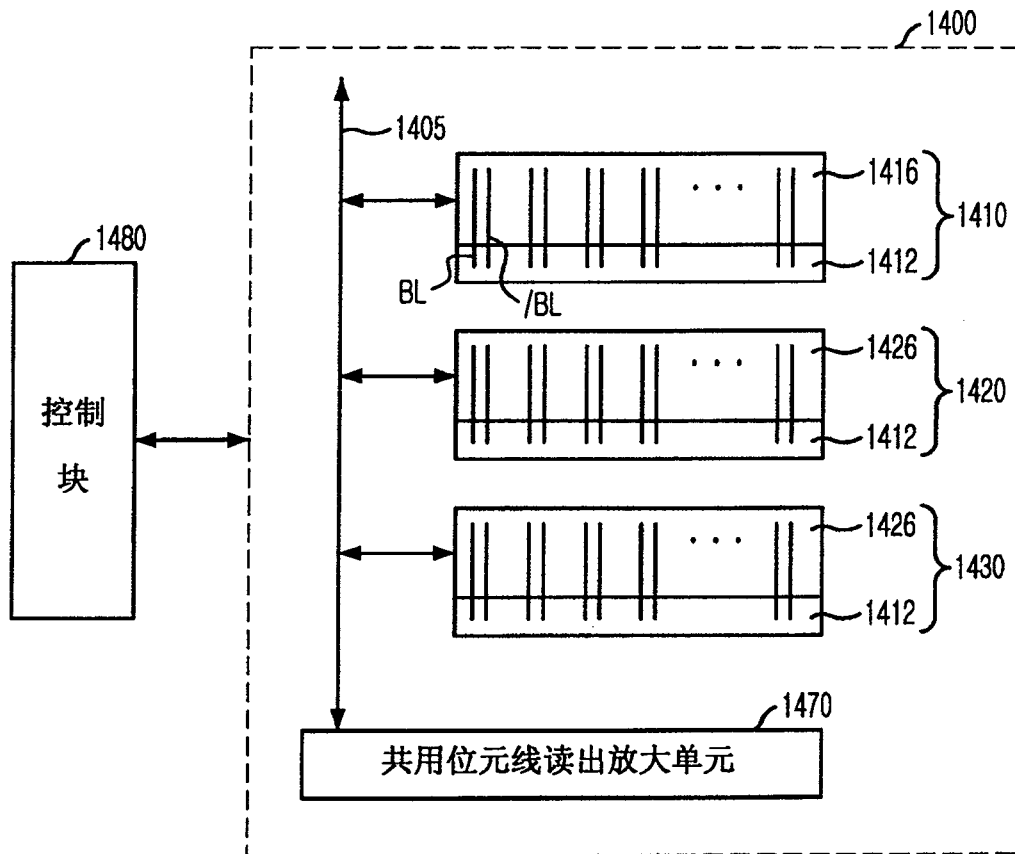


图 14

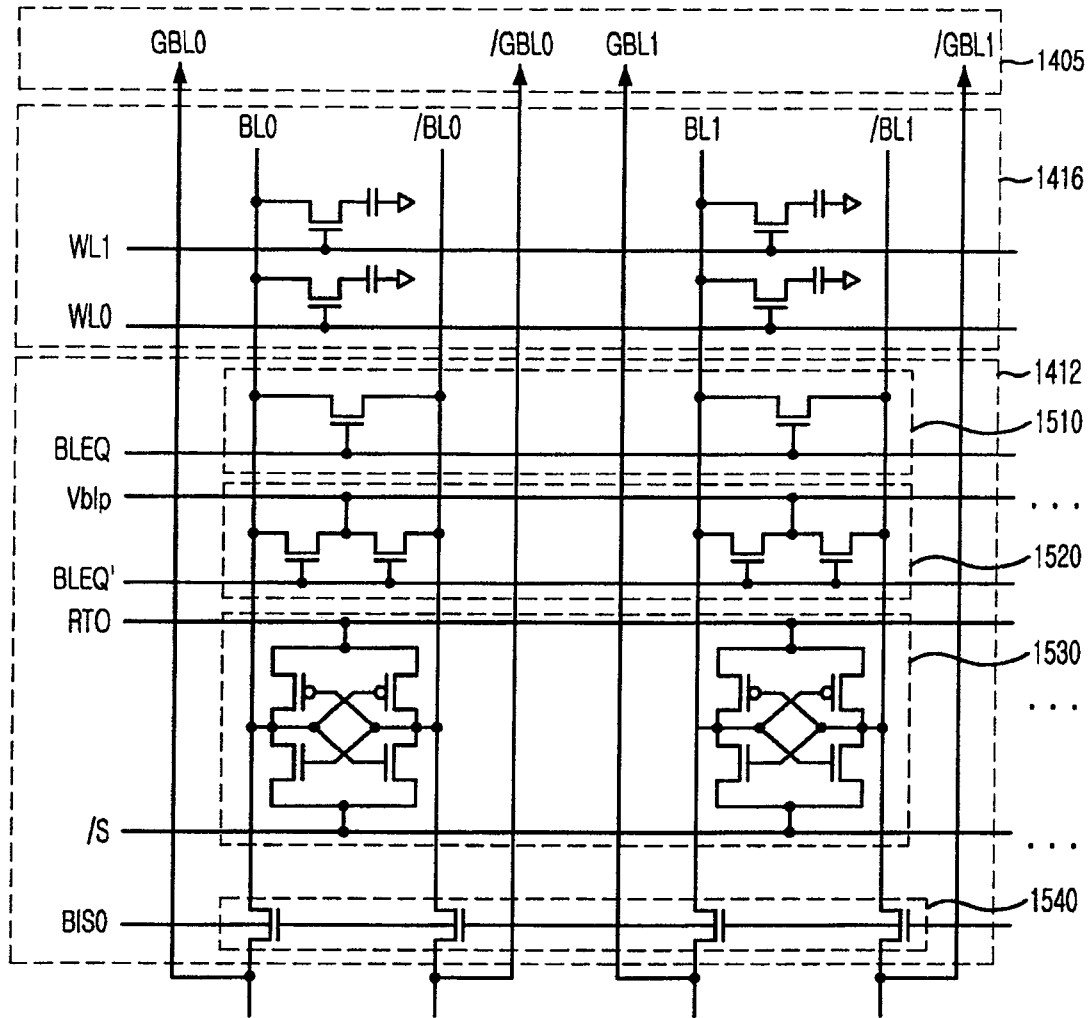


图 15

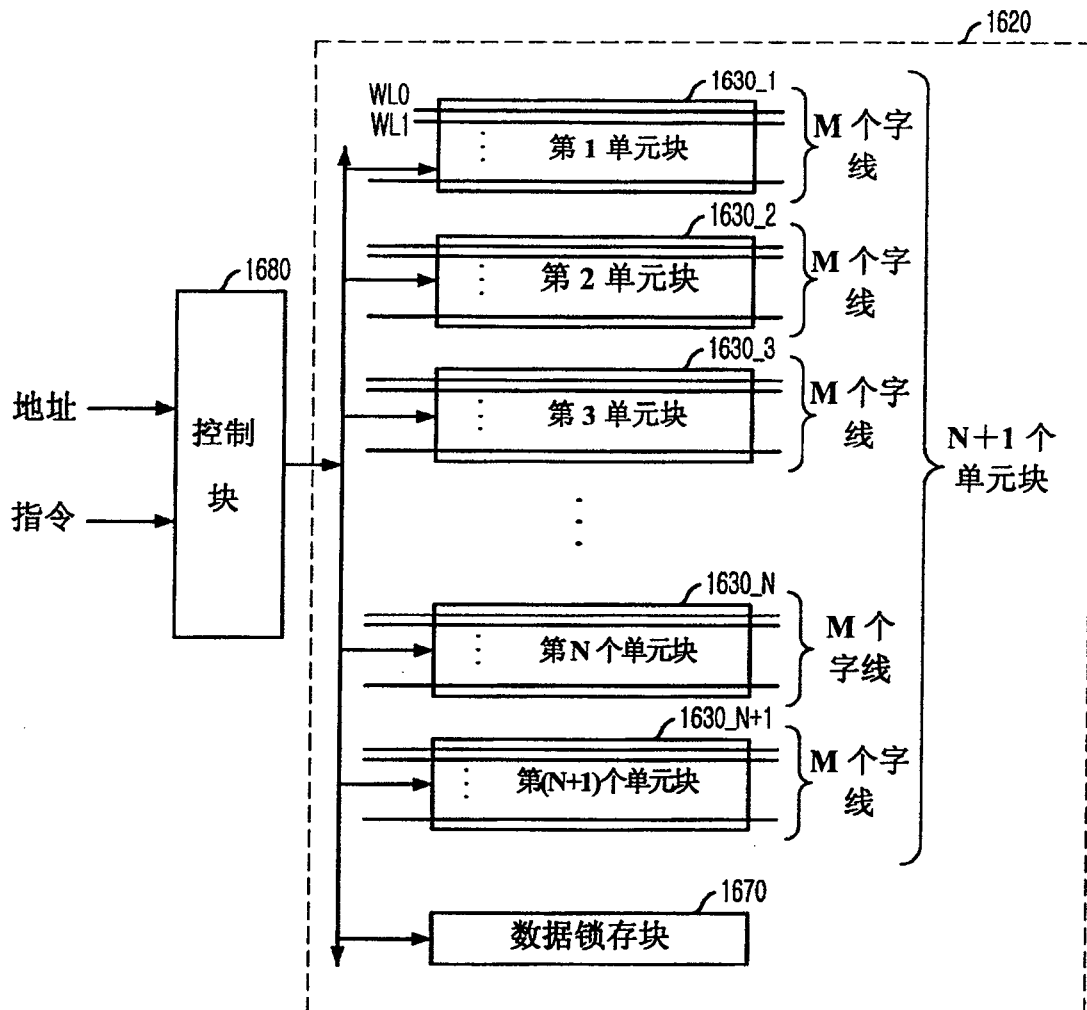


图 16

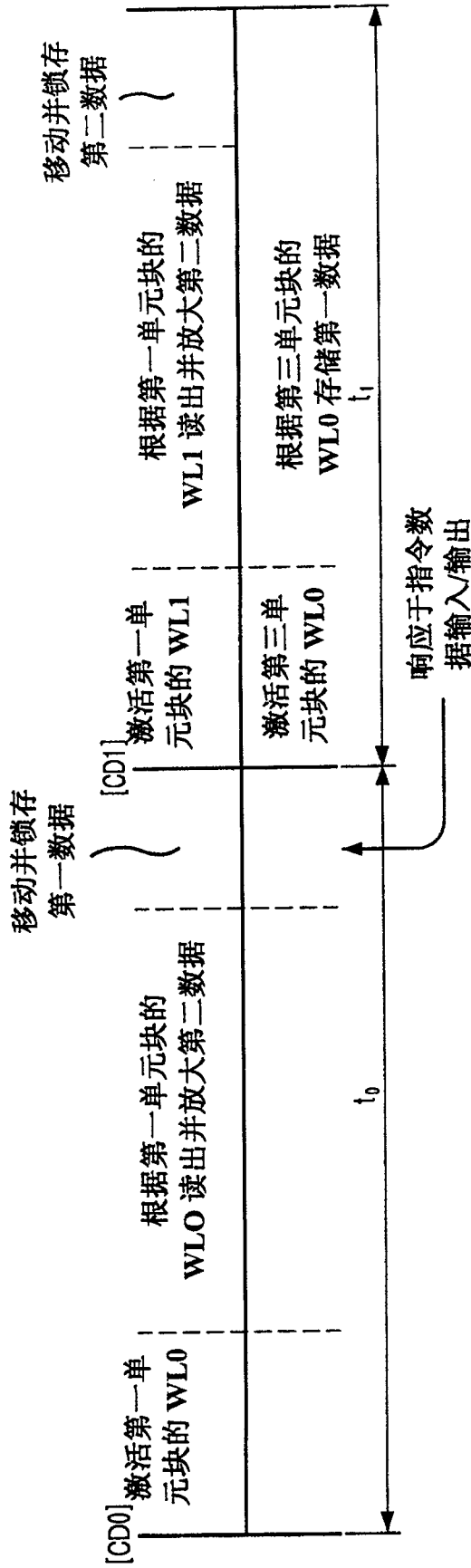


图 17

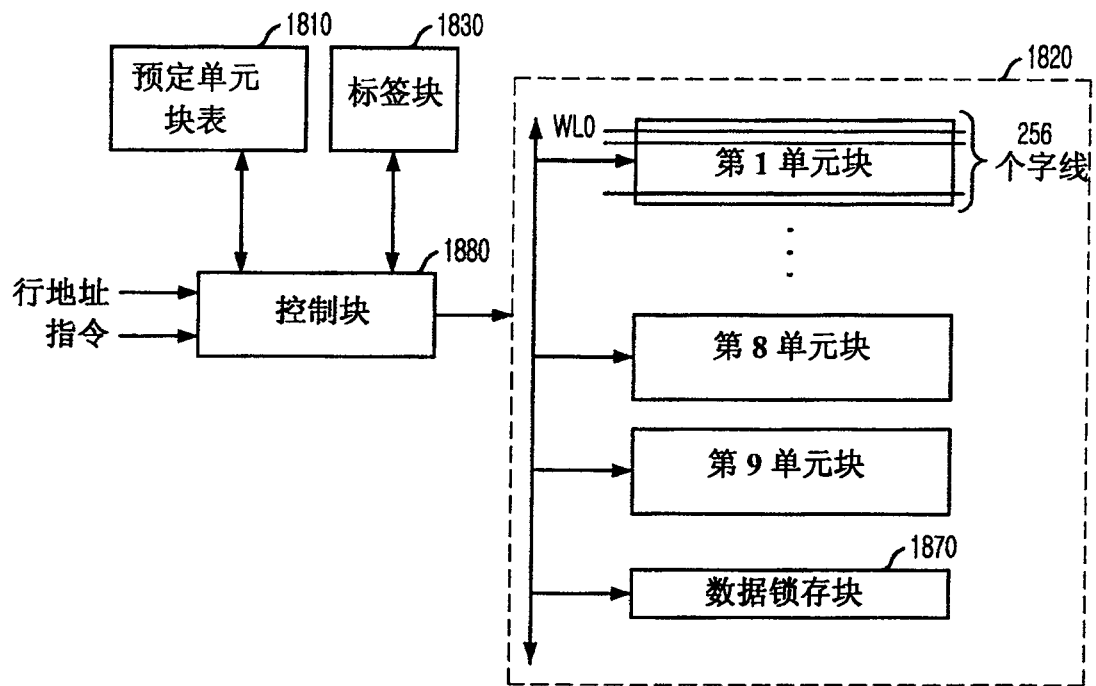


图 18

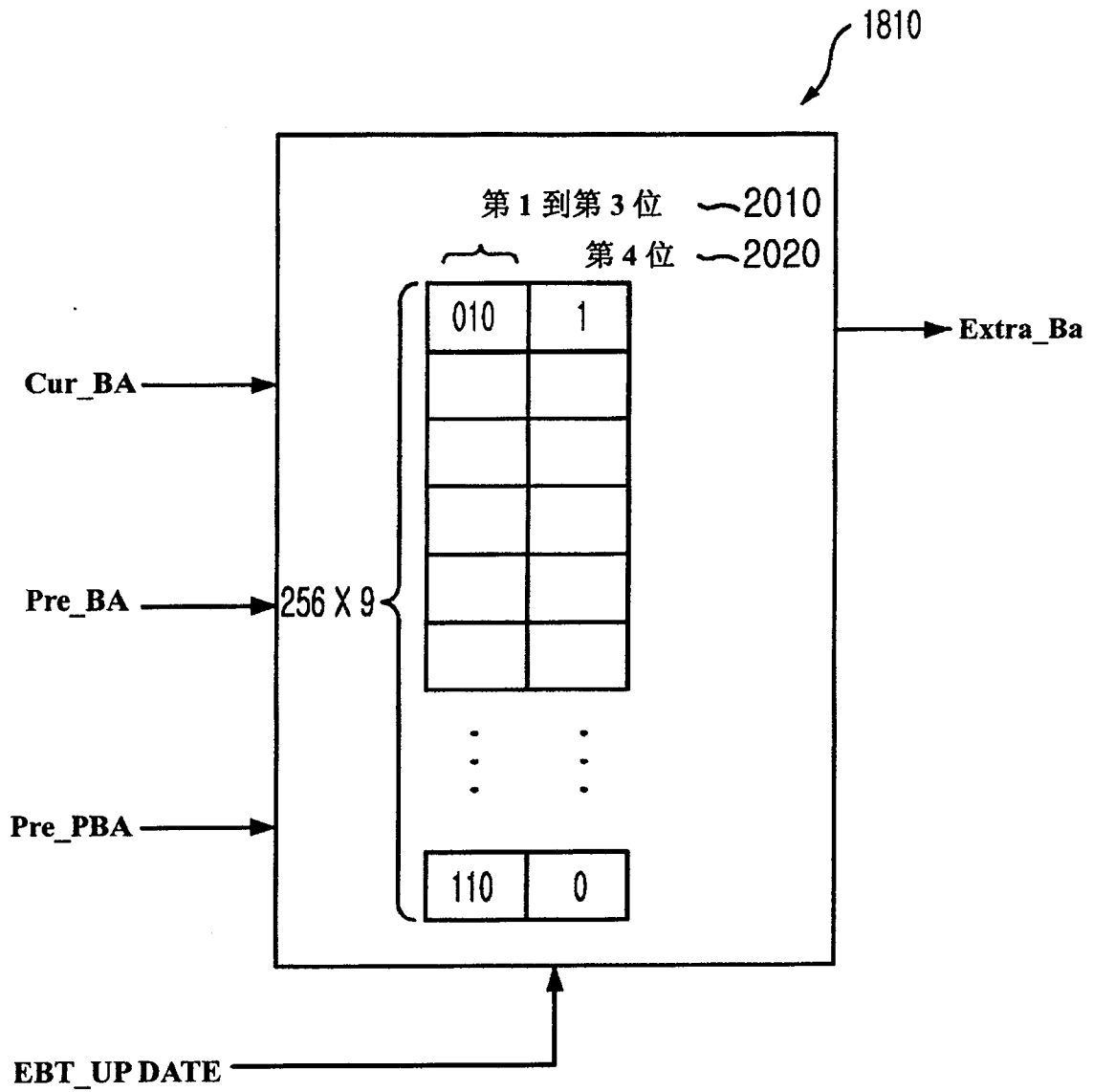


图 19

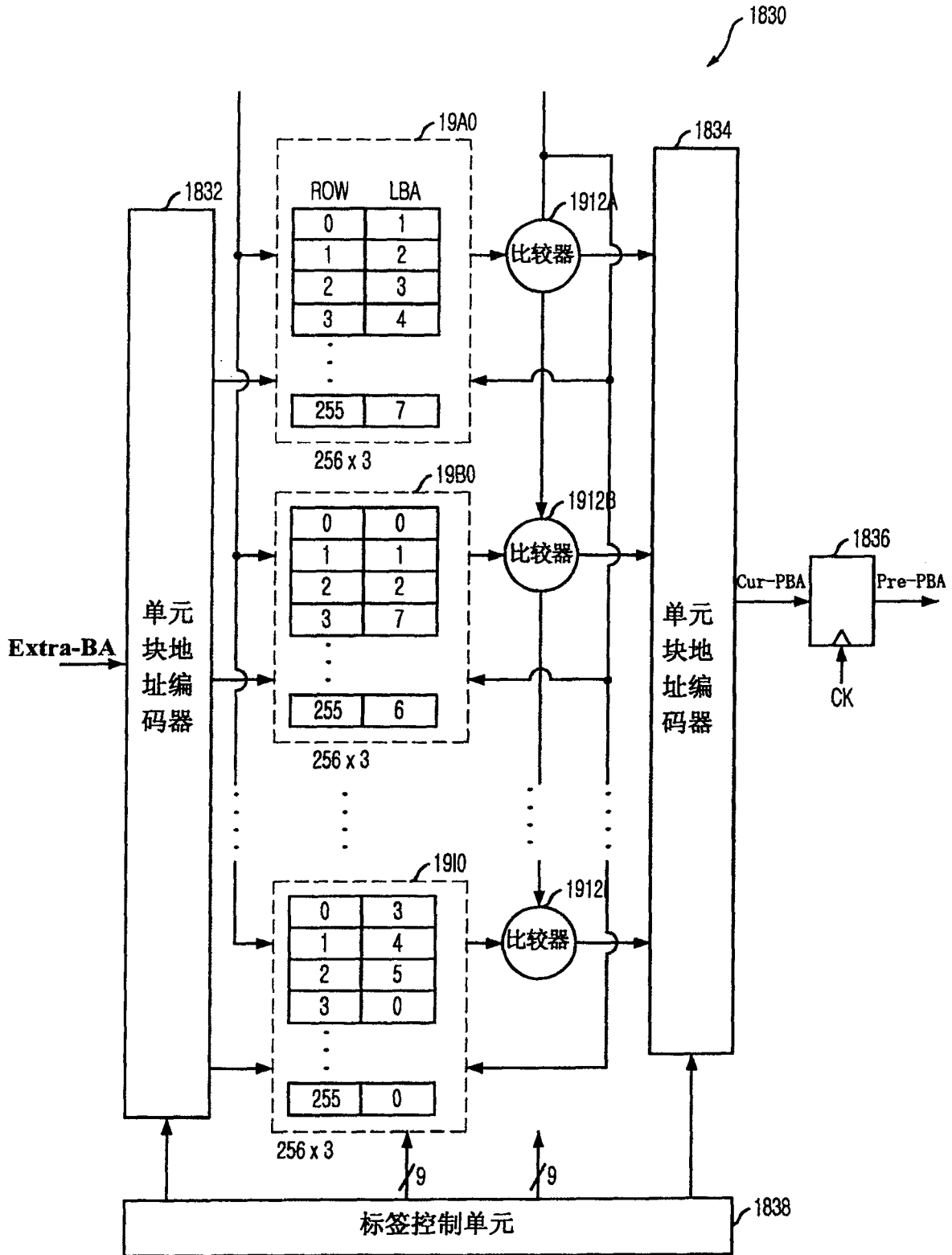


图 20

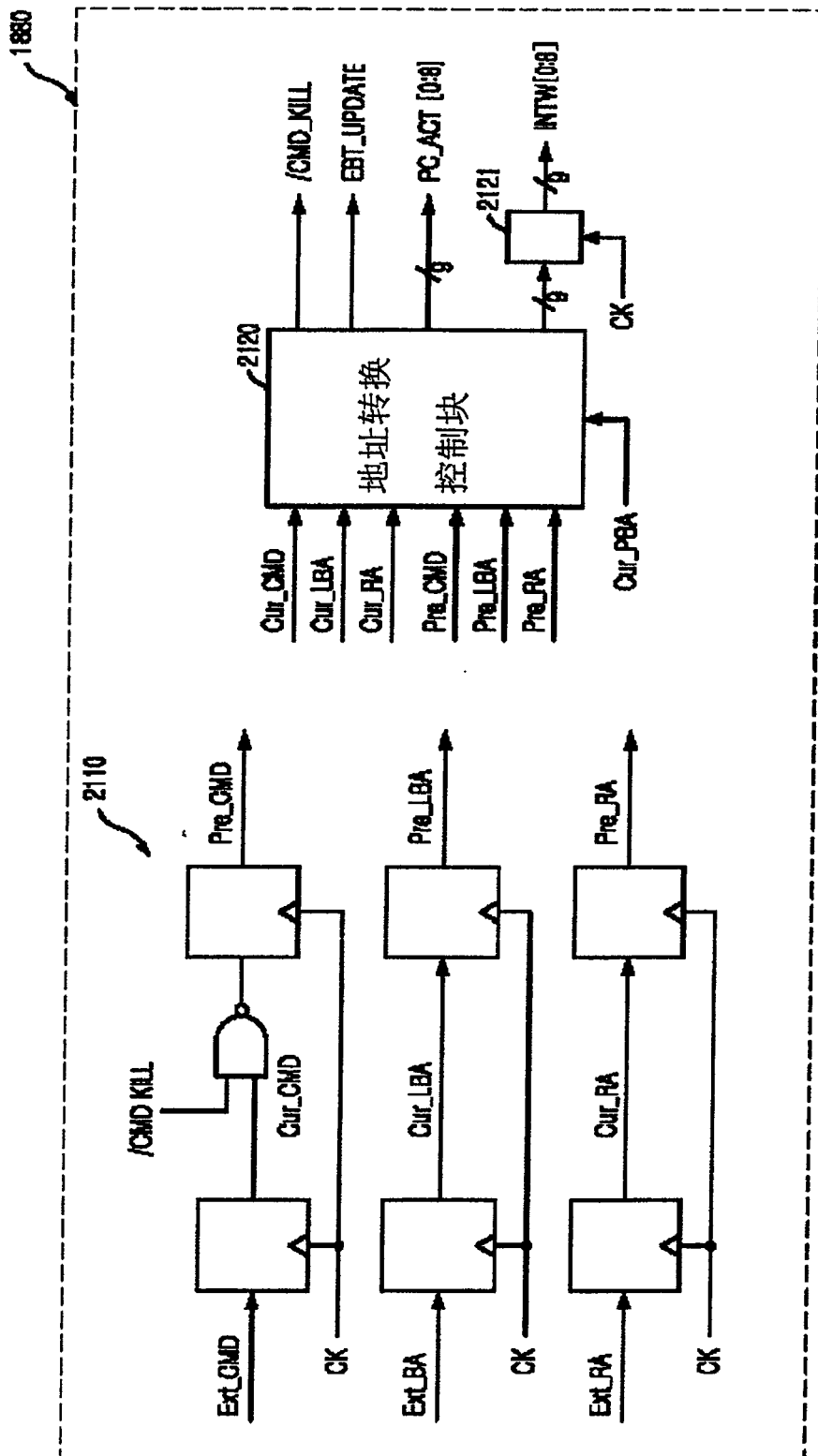


图 21

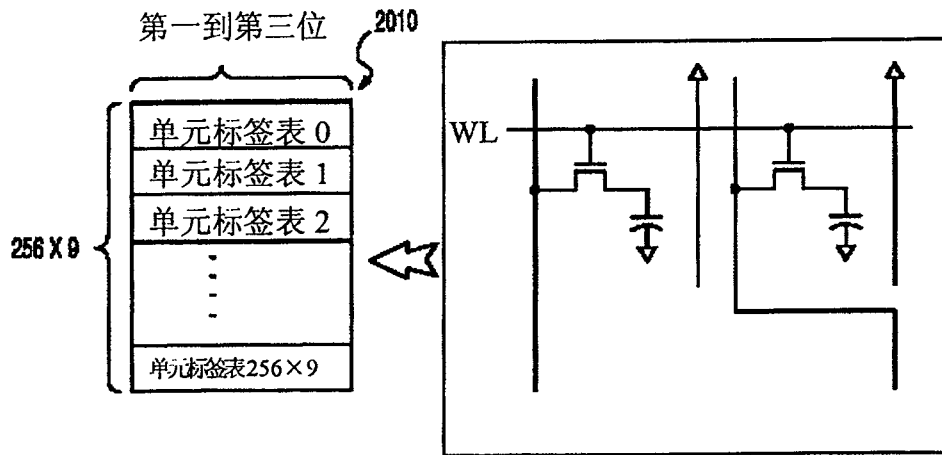


图 22

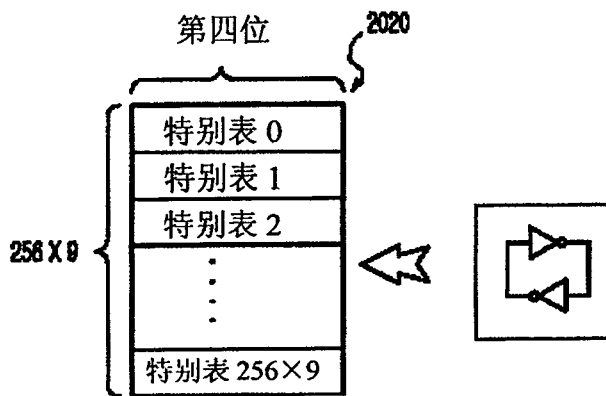


图 23

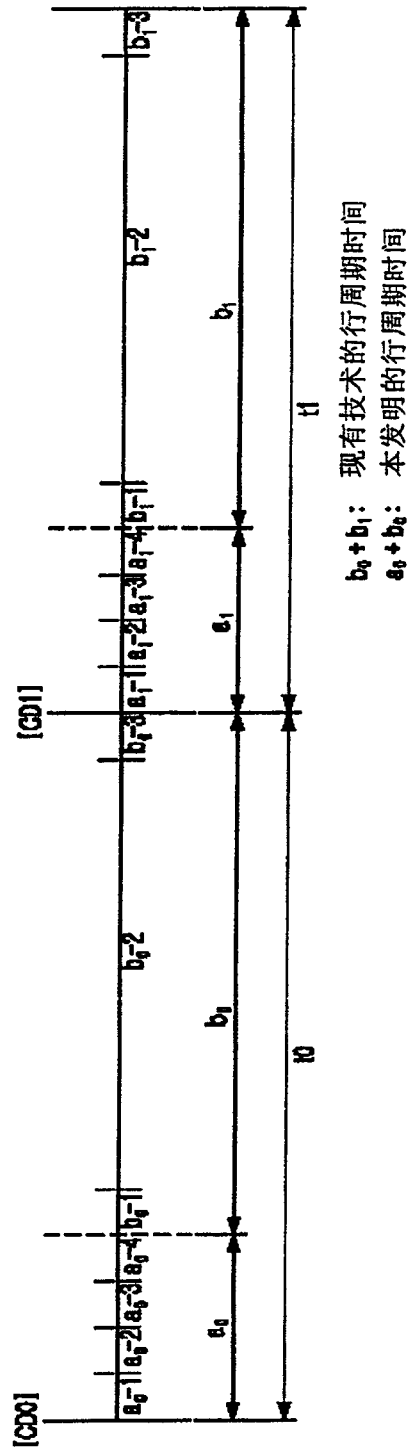


图 24

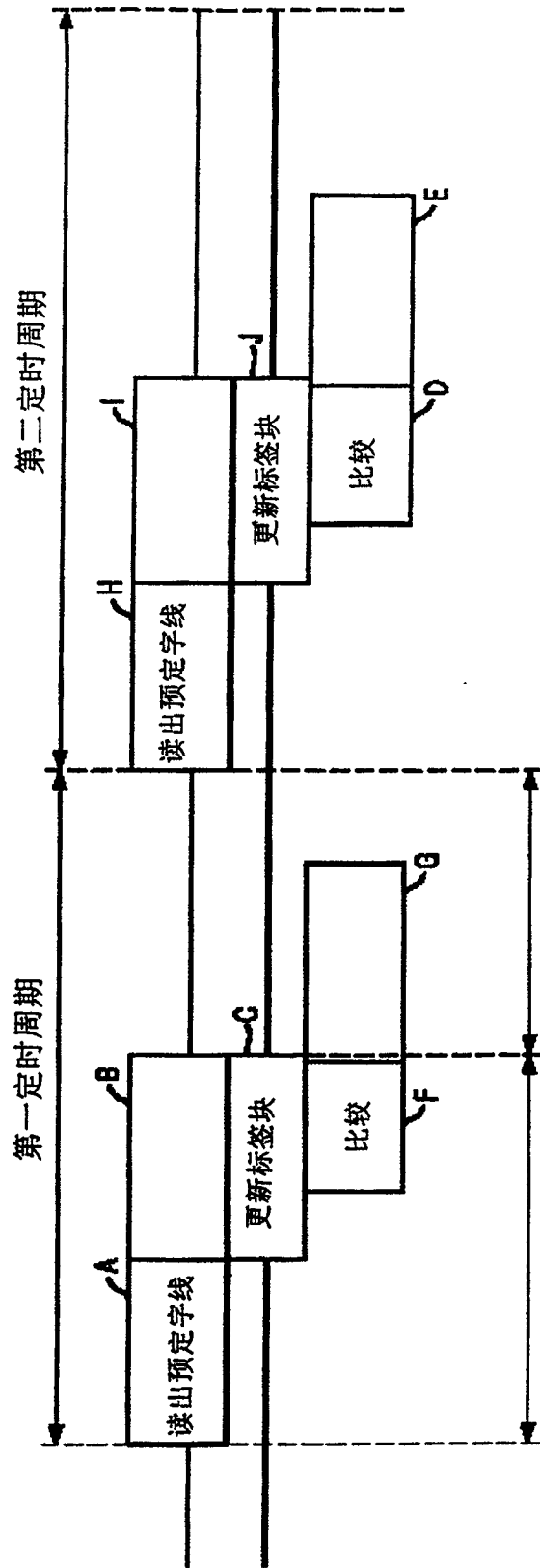
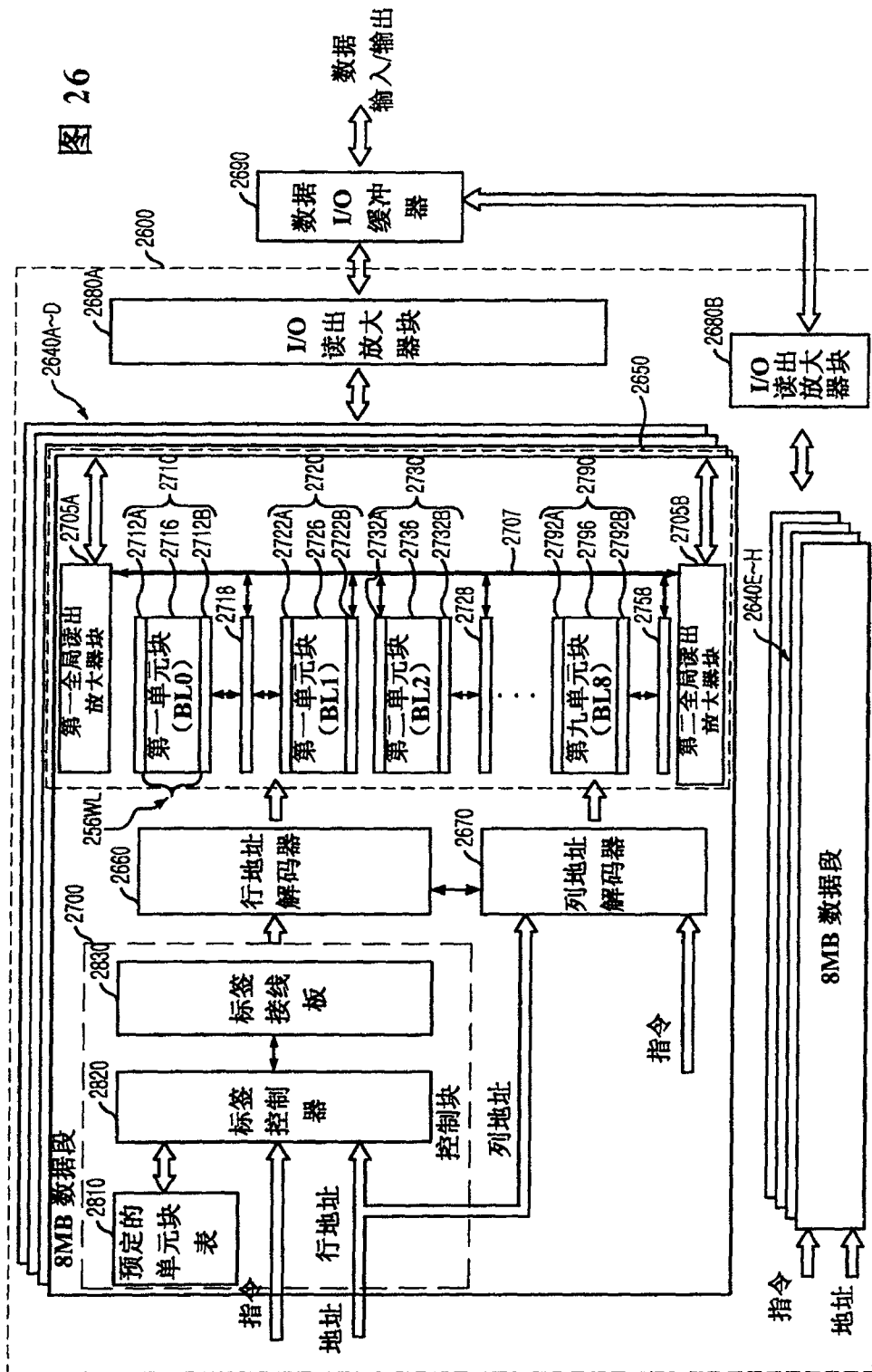


图 25

图 26



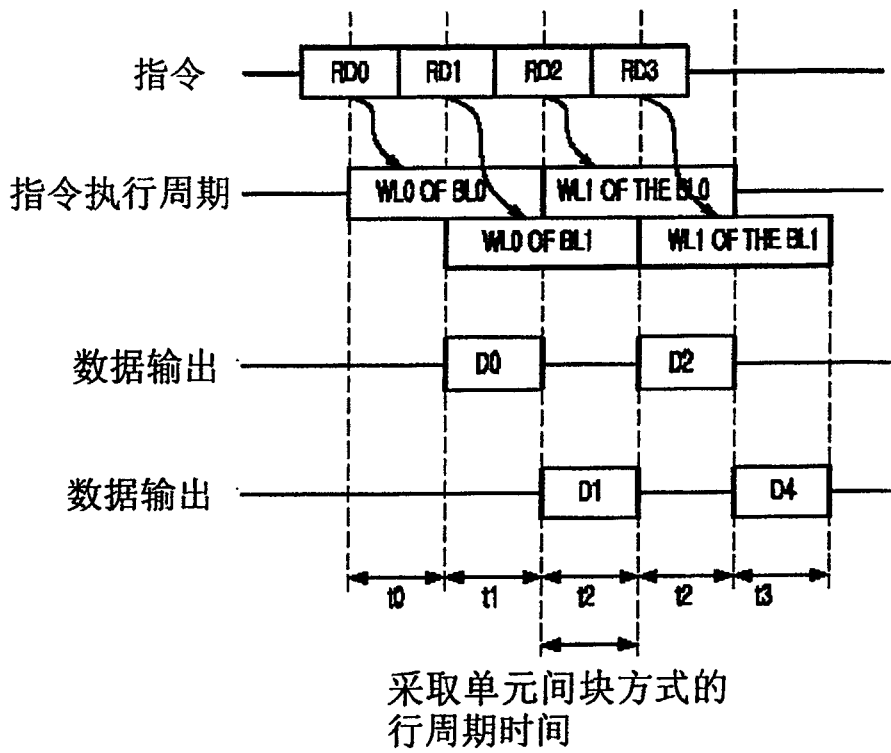


图 27

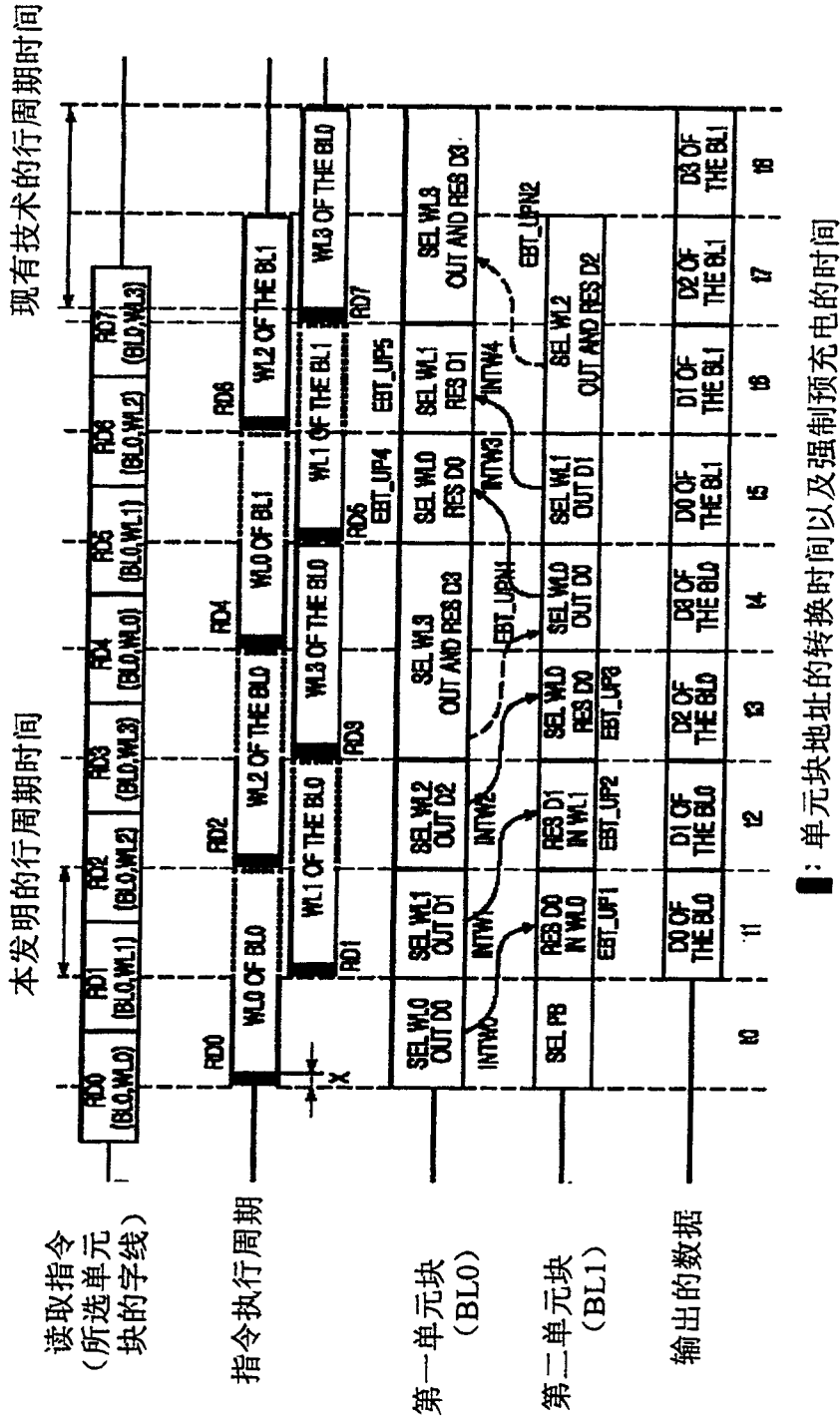


图 28

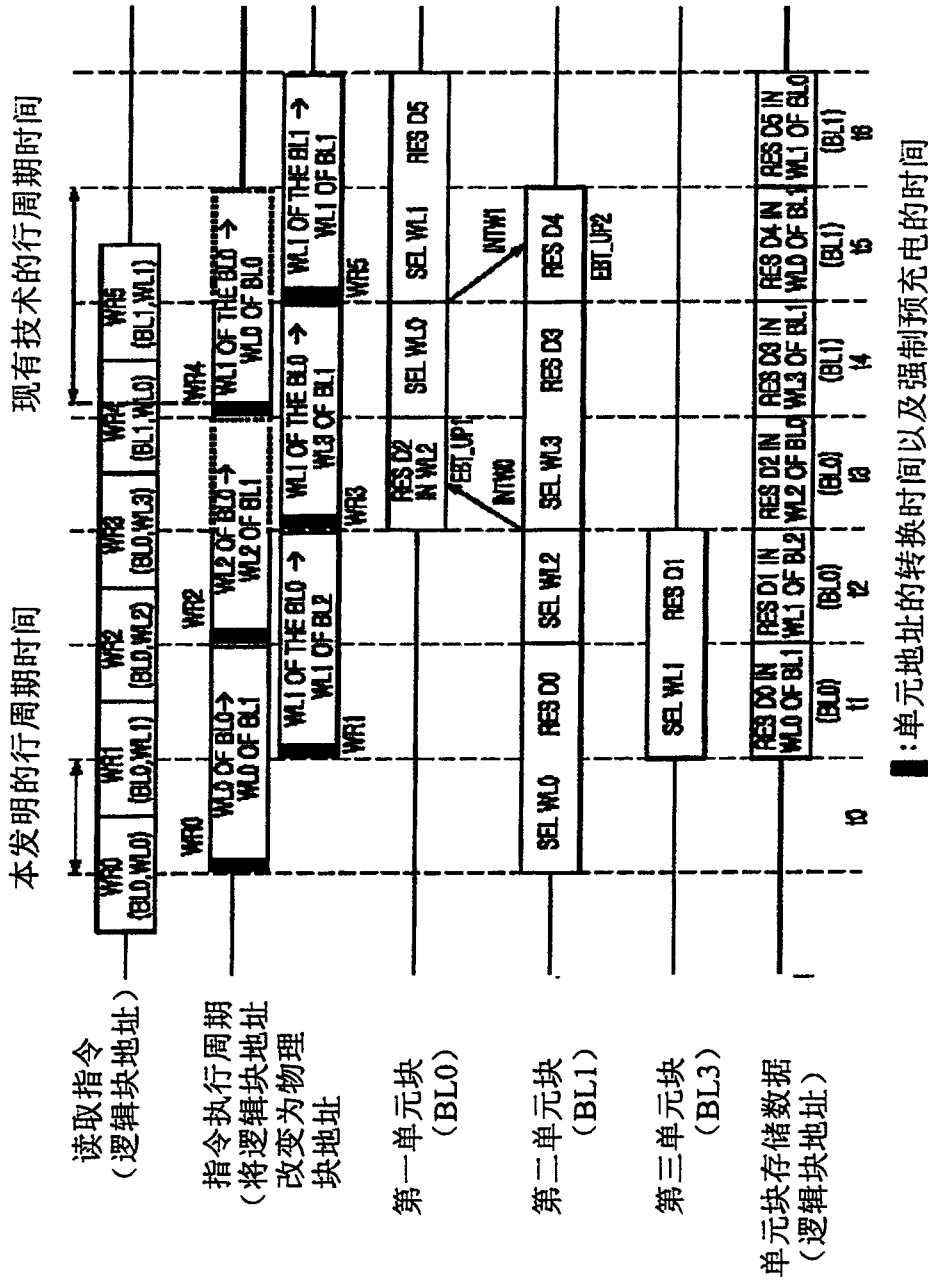


图 29

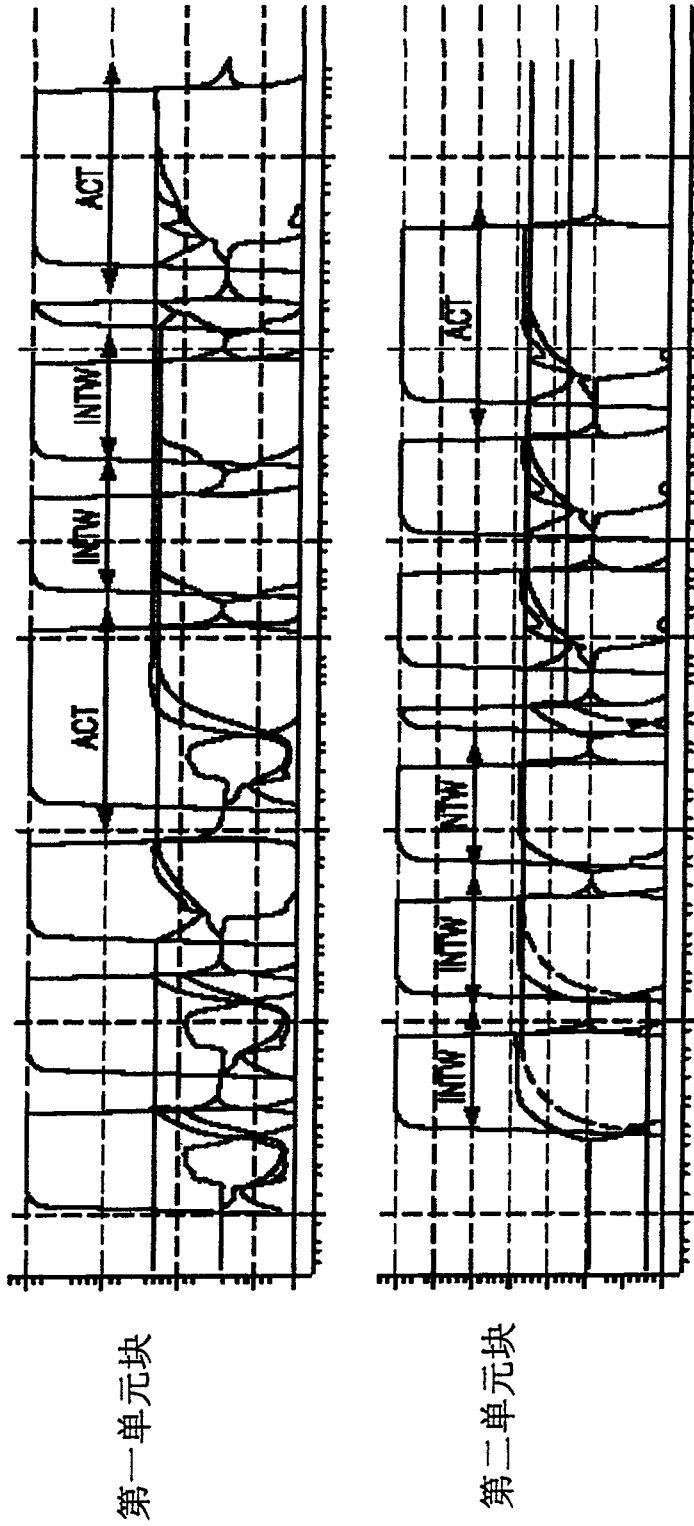


图 30

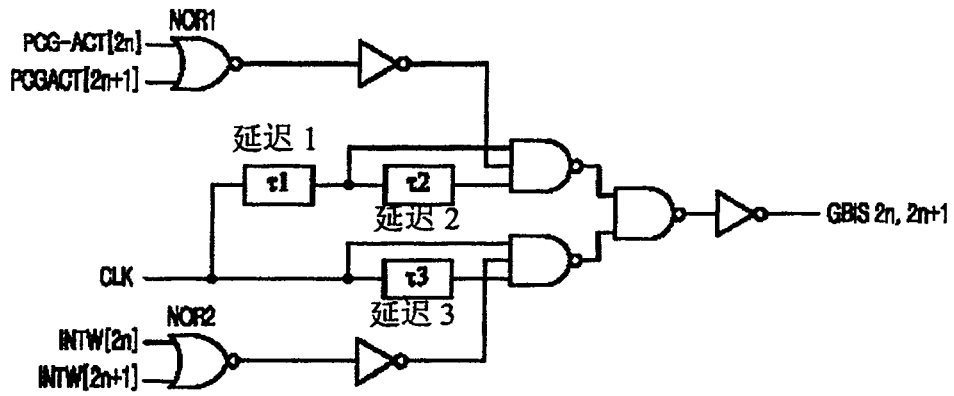


图 31

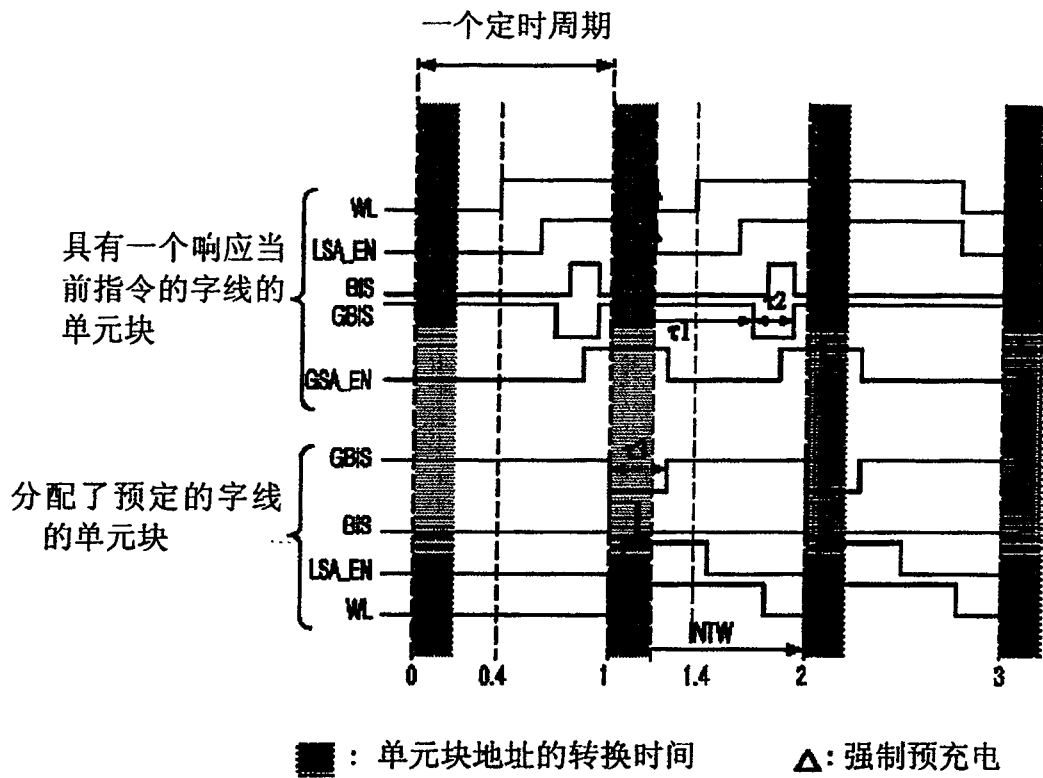


图 32

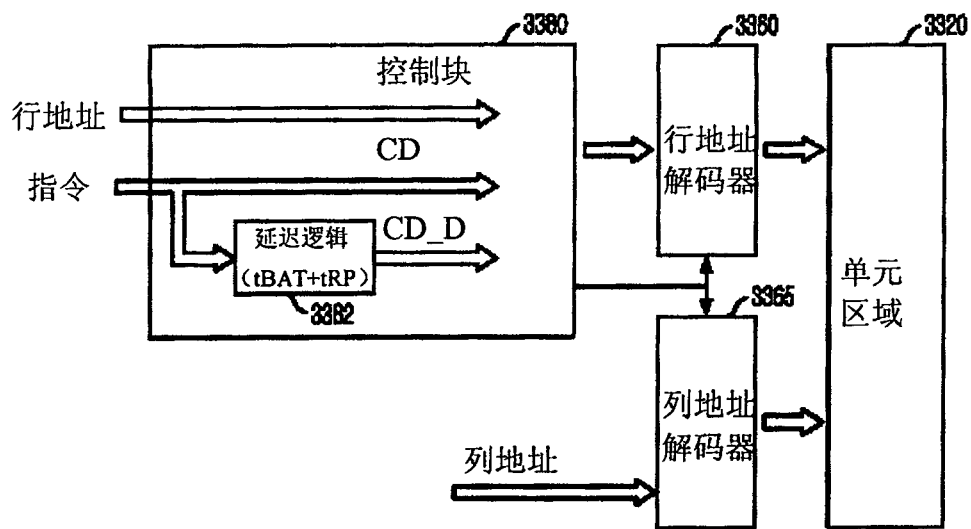


图 33

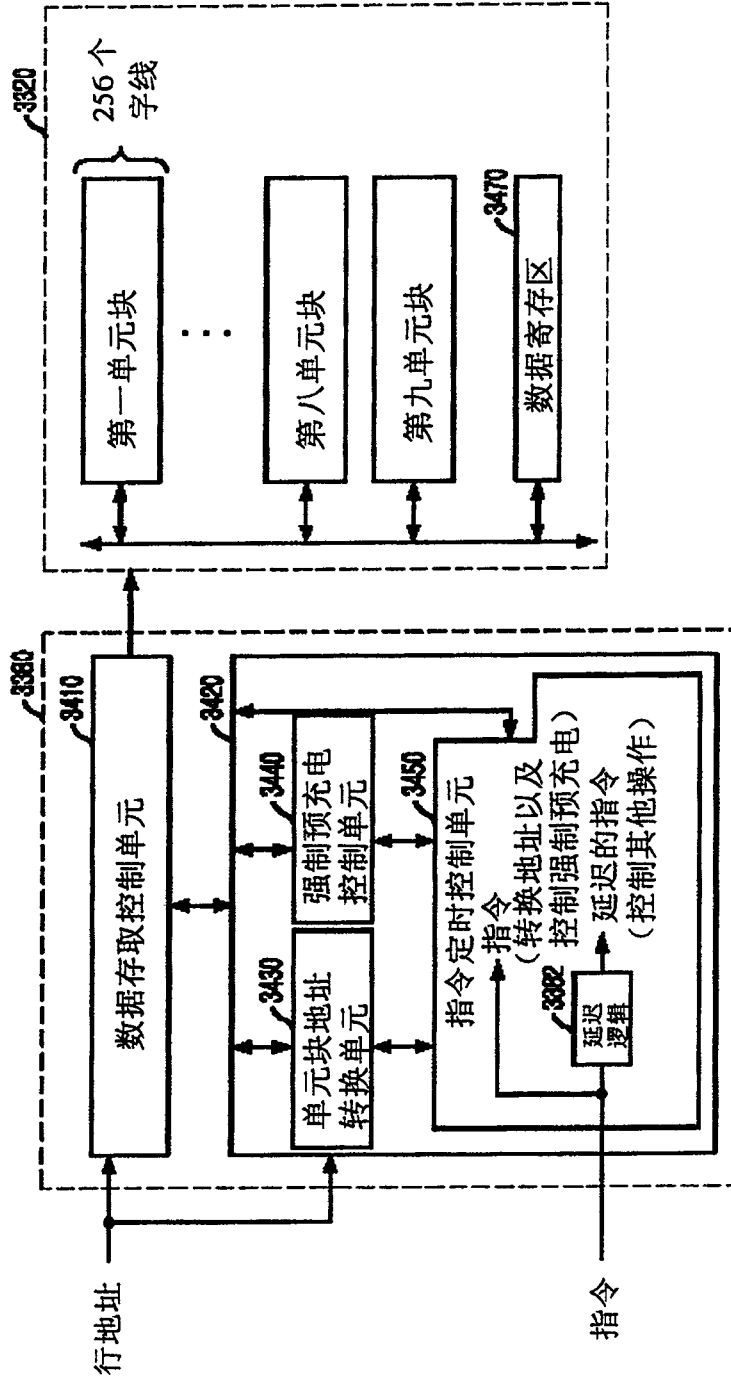


图 34

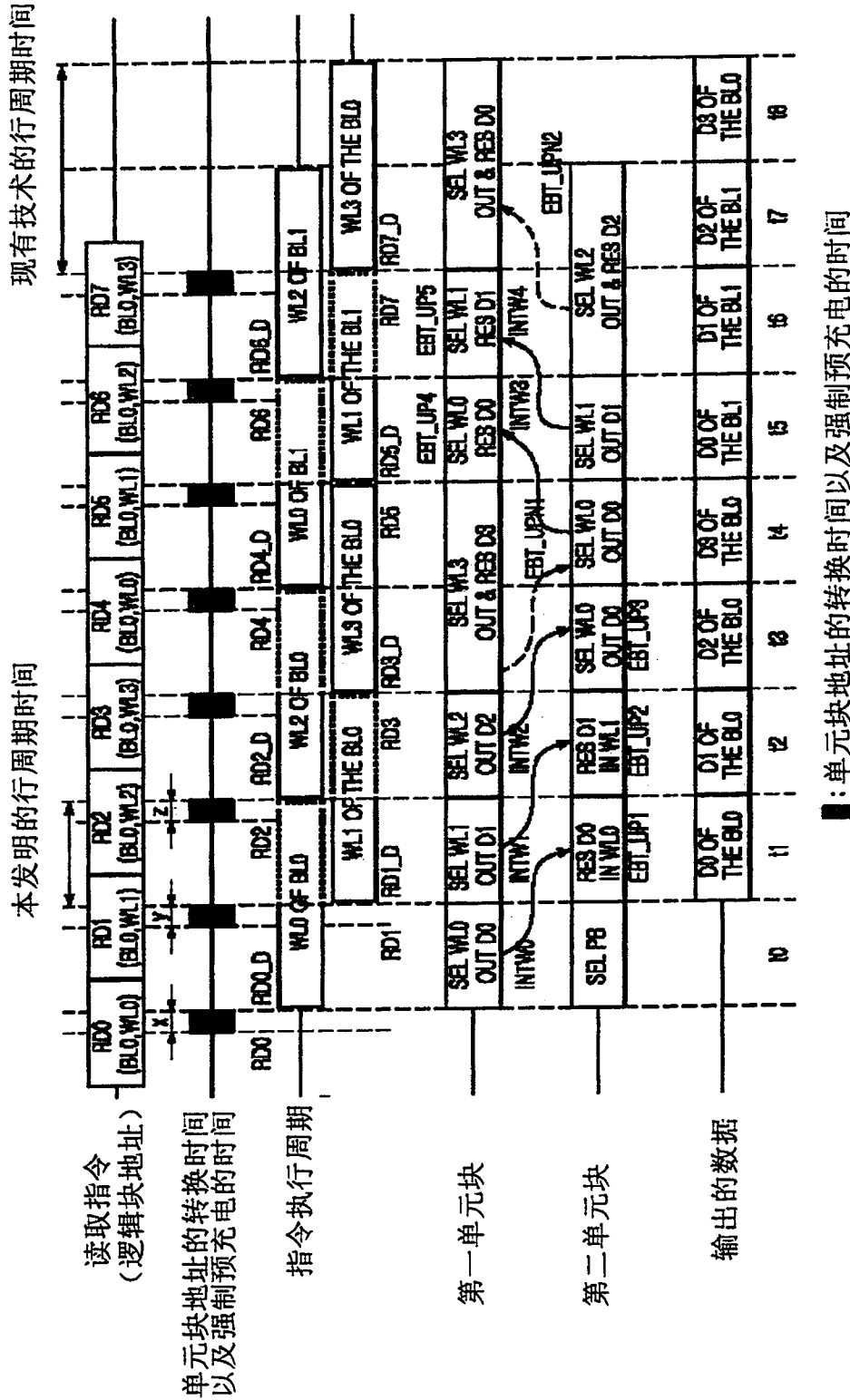
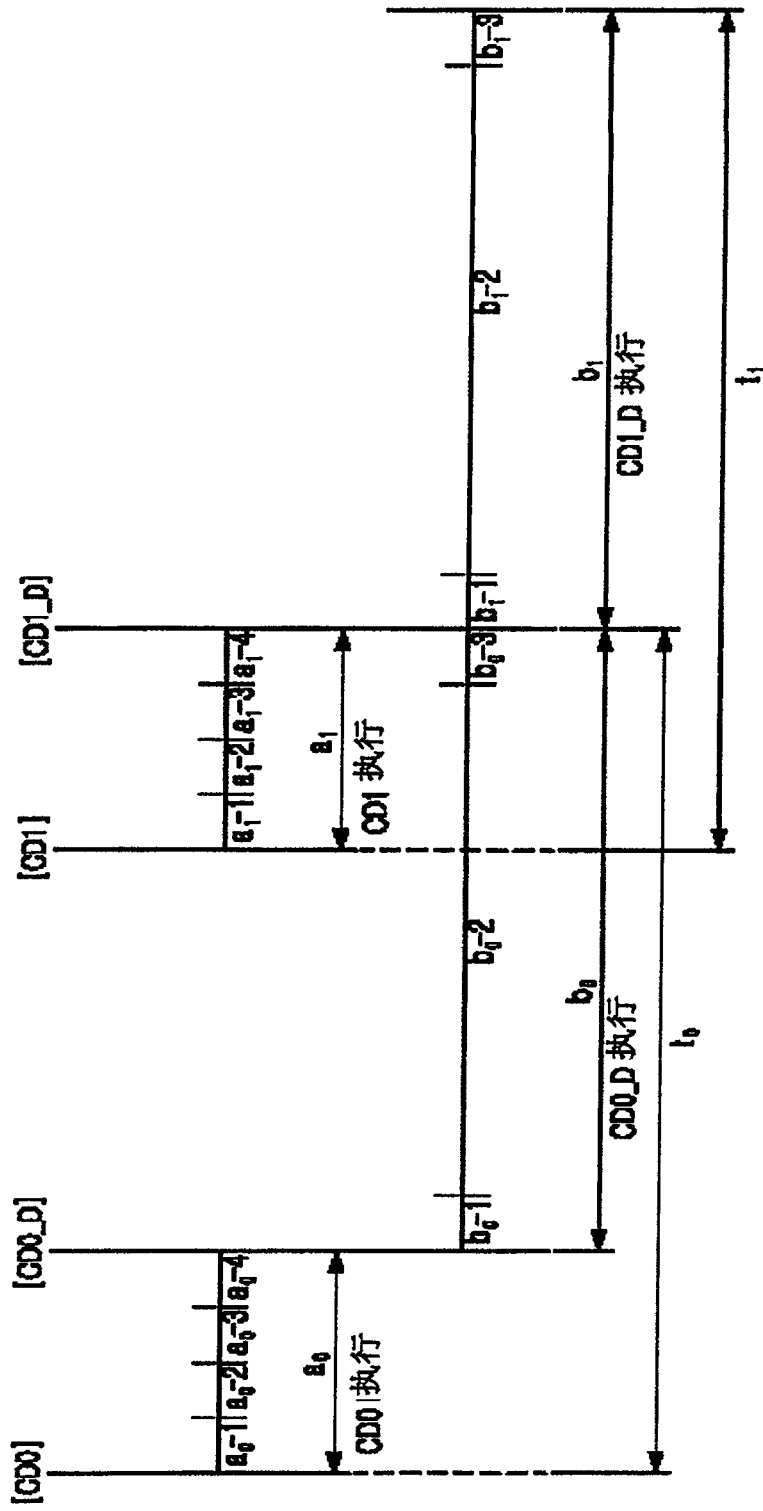


图 35



$b_0 + b_1$ (tRC) : 现有技术的行周期时间
 $b_0 / 2$ (tRC) : 本发明的行周期时间

图 36