[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G11C 11/406 (2006.01)

[12] 发明专利申请公开说明书

[21] 申请号 200480031221.X

[43] 公开日 2006 年 11 月 29 日

[11] 公开号 CN 1871663A

[22] 申请日 2004.10.21
[21] 申请号 200480031221.X
[30] 优先权
[32] 2003.10.24 [33] JP [31] 365168/2003
[86] 国际申请 PCT/JP2004/015589 2004.10.21
[87] 国际公布 WO2005/041201 日 2005.5.6
[85] 进入国家阶段日期 2006.4.21
[71] 申请人 国际商业机器公司
地址 美国纽约
[72] 发明人 砂永登志男 宫武久忠 细川浩二
[74] 专利代理机构 中国国际贸易促进委员会专利商标事务所
代理人 曲 瑞

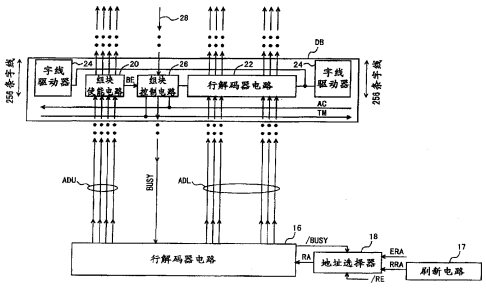
权利要求书 3 页 说明书 15 页 附图 10 页

[54] 发明名称

半导体存储器件及其刷新方法

[57] 摘要

为了提供一种在常规存取操作期间能够插入刷新操作并且能够设定内部循环时间长于外部循环时间的一半的 DRAM，本发明提供一种半导体存储器件及其刷新方法。地址选择器(18)选择存取行地址信号 ERA 或刷新行地址信号 RRA。行解码器控制电路(16)响应所选的行地址信号 RA 选择分割存储单元阵列后得到的组块之一，并通过行解码器电路 22 选择字线。当对该一个组块开始操作时，激活忙信号/BUSY 以禁止由地址选择器 18 执行选择。当操作结束时，使忙信号/BUSY 无效以取消对地址选择器 18 的选择的禁止。因此，优先执行行地址信号 ERA 或 RRA 中较早输入的一个，并使随行地址信号 ERA 或 RRA 中后输入的一个等待，直到在先操作结束为止。



1. 一种半导体存储器件，包括:

存储单元阵列，包括多条字线;

刷新装置，用于产生刷新请求并随之生成刷新地址;

地址选择装置，用于响应存取请求而选择存取地址，所述地址选择装置响应所述刷新请求而从所述多个刷新地址中选择刷新地址;

字线选择装置，用于响应由所述地址选择装置选择的地址而从所述多条字线中选择字线；和

选择停止装置，用于在所述存储单元阵列中正在进行存取或刷新的同时，停止由所述地址选择装置执行的地址选择。

2. 如权利要求 1 所述的半导体存储器件，其中将所述存储单元阵列分成多个组块，所述半导体存储器件还包括用于响应由所述地址选择装置选择的地址而从所述多个组块中选择组块的组块选择装置，所述选择停止装置在对由所述组块选择装置选择的组块执行存取或刷新的同时停止所述地址选择装置执行地址选择。

3. 如权利要求 2 所述的半导体存储器件，其中所述字线选择装置响应由所述地址选择装置选择的刷新地址而针对所述组块中的每一个连续地选择所有字线。

4. 如权利要求 2 或 3 所述的半导体存储器件，其中所述选择停止装置包括:

忙信号发生装置，用于响应所述存取请求或刷新请求来激活忙信号，并在完成了对由所述组块选择装置选择的组块的存取或刷新之后使所述忙信号无效，

所述地址选择装置包括:

输入装置，用于响应所述存取请求而输入所述存取地址，以及响应所述刷新请求而输入所述刷新地址；和

锁存装置，用于在忙信号被无效之后接收并锁存所输入的地址。

5. 如权利要求 4 所述的半导体存储器件，其中所述忙信号发生

装置包括:

通过所述多个组块以共有方式被提供的忙信号线;

充电装置,用于响应所述刷新请求而对所述忙信号线进行充电;和

对应于所述多个组块中的每一个而提供的放电装置,所述放电装置在完成了对相应的组块的存取或刷新之后对所述忙信号线进行放电。

6. 一种用于半导体存储器件的刷新方法,所述半导体存储器件具有包括多条字线的存储单元阵列,所述方法包括以下步骤:

产生刷新请求并随之生成刷新地址;

响应存取请求而选择存取地址,以及响应所述刷新请求而从所述多个刷新地址中选择刷新地址;

响应所选的地址而从多条字线中选择字线;以及

在存储单元阵列中执行了存取或刷新之后,停止选择所述存取地址和所述刷新地址。

7. 如权利要求 6 所述的用于半导体存储器件的刷新方法,其中将所述存储单元阵列分成多个组块,所述刷新方法还包括响应所选的地址而从多个组块中选择组块的步骤,所述停止步骤还包括在对所选的组块执行了存取或刷新之后停止选择所述存取地址和所述刷新地址的步骤。

8. 如权利要求 7 所述的用于半导体存储器件的刷新方法,其中所述选择步骤包括响应所述刷新地址而针对每一个组块连续地选择所有字线的步骤。

9. 如权利要求 7 或 8 所述的用于半导体存储器件的刷新方法,其中所述停止步骤包括:

忙信号发生步骤,响应所述存取请求或刷新请求来激活忙信号,并在完成了对所选组块的存取或刷新之后使所述忙信号无效,

所述地址选择步骤包括如下步骤:

响应所述存取请求而输入所述存取地址;

响应所述刷新请求而输入所述刷新地址；和

在所述忙信号被无效之后接收并锁存所输入的地址。

10. 如权利要求 9 所述的用于半导体存储器件的刷新方法，其中半导体存储器件还具有通过多个组块以共有方式被提供的忙信号线，所述忙信号发生步骤包括如下步骤：

响应所述存取请求或刷新请求而对所述忙信号线进行充电；和

在完成了对相应组块的存取或刷新之后，对所述忙信号线进行放电。

半导体存储器件及其刷新方法

技术领域

本发明涉及一种半导体存储器件以及刷新该半导体存储器件的方法。具体而言，本发明涉及一种能够在常规的存取操作期间插入刷新操作的 DRAM（动态随机存取存储器）以及刷新该 DRAM 的改进方法。

背景技术

近来，在低功耗应用中，由 DRAM 替代 SRAM（静态随机存取存储器）已经相当普遍，因为 DRAM 每单位面积的存储器容量远远大于 SRAM 每单位面积的存储容量。但是，DRAM 需要刷新，而 SRAM 则不是必须刷新。因此，存在这样一种需求，要求能够以如下方式来使 DRAM 可被用于与使用 SRAM 的方法相同的方法，即以使用 DRAM 中的内部电路来执行自动刷新代替使用外部电路（例如刷新控制器）的刷新。

下面示出的专利文献 1 公开了一种 DRAM，该 DRAM 使用了在一个循环时间（以下称作“外部循环时间”）内插入了常规读出操作或写入操作（以下称作“常规存取操作”或简称为“存取操作”）以及刷新操作的系统。在此系统中，由于在一个外部循环时间内确保了用于存取的时间和用于刷新的时间，所以能够在任何时间执行刷新而不必延迟常规的存取。用于存取的时间和用于刷新的时间基本上彼此相等，因此下面将它们统称为“内部循环时间”。

此 DRAM 的外部循环时间是实际的循环时间，其确定了操作速度。因此，为了增加此 DRAM 的操作速度而必需缩减外部循环时间。为了实现外部循环时间的缩减，必需将内部循环时间缩减到等于或小于外部循环时间的一半的时间长度。降低外部循环时间是很困难的。

此 DRAM 被设计成通过在每个外部循环时间内确保用于刷新的内部循环时间，而能够在任何时间执行刷新。因此，仅仅使用了此 DRAM 的实际能力的一半，并且此 DRAM 的加速是困难的。

『专利文献 1』

日本专利公开 No.2002-298574

发明内容

『本发明所要解决的问题』

本发明的主要目的是提供一种能够在常规存取操作期间插入刷新并能够实现高速化的半导体存储器件以及刷新该存储器件的方法。

『发明概述』

依据本发明的一种半导体存储器件具有存储单元阵列、刷新装置、地址选择装置、字线选择装置和选择停止装置。存储单元阵列包括多条字线。刷新装置产生刷新请求并随之生成刷新地址。地址选择装置在产生存取请求时选择存取地址，在产生刷新请求时选择刷新地址。字线选择装置按照地址选择装置所选的地址来选择字线。在存储单元阵列中正在进行存取或刷新的同时，选择停止装置停止由地址选择装置执行的地址选择。

依据本发明的一种刷新方法具有产生刷新请求并随之生成刷新地址的步骤；在产生存取请求时选择存取地址以及在产生刷新请求时选择刷新地址的地址选择步骤；按照所选的地址来选择字线的字线选择步骤；以及在存储单元阵列中正在进行存取或刷新的同时停止选择存取地址和刷新地址的选择停止步骤。

依据本发明，当产生存取请求时选择存取地址，当产生刷新请求时选择刷新地址，并且按照由此选择的地址来选择字线。因此，能够在常规的存取期间插入刷新。在存储单元阵列中执行存取或刷新的同时，停止上述地址选择。因此，在产生存取请求之前产生了刷新请求的情况下，优先执行刷新，并将随后请求的存取延迟到先前启动的刷新完成为止。相反地，在产生刷新请求之前产生了存取请求的情况下，

优先执行存取，并将随后请求的刷新延迟到先前启动的存取完成为止。因此，内部循环时间相对于外部循环时间而增加，由此外部循环时间被缩减以增加操作速度。

优选地，将存储单元阵列分成多个组块。上述的半导体存储器件还具有用于响应由地址选择装置选择的地址来选择组块的组块选择装置。在对组块选择装置所选的组块执行存取或刷新的同时，选择停止装置使地址选择装置停止执行地址选择。上述的刷新方法还包括响应所选的地址来选择组块的步骤。选择停止步骤包括在对所选的组块执行存取或刷新的同时停止选择存取地址和刷新地址。

此外，优选地，在上述半导体存储器件中，字线选择装置响应刷新地址而针对每一个组块连续地选择所有字线。在上述的刷新方法中，字线选择步骤包括响应刷新地址而针对每一个组块连续地选择所有字线。

由于在此情况下，以组块为单位来执行所谓的集中式刷新（burst refresh），所以刷新中的延迟（如果有）能够在对选定组块的操作过程中被吸收，并且不会延续至任何其它组块。

附图说明

图 1 是示出了表示本发明实施例的 DRAM 的整体构造的功能方框图；

图 2 是示出了图 1 中所示的解码器组块和解码器控制电路的构造的功能方框图；

图 3 是示出了图 2 中所示的地址选择器和刷新电路的构造的功能方框图；

图 4 是示出了图 1 至 3 中所示的 DRAM 的读出和刷新操作的时序图；

图 5 是示出了图 2 中所示的组块控制电路的构造的功能方框图；

图 6 是示出了图 2 和 3 中所示的地址选择器的构造的功能方框图；

图 7 是示出了图 6 中所示的地址选择器的操作的时序图；

图 8 是示出了图 1 至 3 中所示的 DRAM 的集中式刷新操作的时序图；

图 9 是示出了如图 8 中所示的集中式刷新操作的时序图，特别是示出了在刷新操作之后插入了被设为 N 个的、不同数量的常规存取操作时的操作；和

图 10 是示出了对应于图 9（E）中所示情况的、当 N = 5 时的操作情况的时序图：情况（A）为仅执行存取操作；情况（B）为混合了刷新操作和存取操作；情况（C）为仅执行刷新操作。

『符号说明』

12 ... 存储单元阵列

14 ... 行解码器

16 ... 行解码器控制电路

17 ... 刷新电路

18 ... 地址选择器

20 ... 组块使能电路

22 ... 行解码器电路

24 ... 字线驱动器

26 ... 组块控制电路

28 ... 忙信号线

30 ... 刷新计时器

32 ... 地址计数器

34 ... 刷新使能电路

40、42 ... 晶体管

46 至 49 ...NANAD 电路

54 ... 锁存电路

/AE ... 阵列使能信号

BUSY、/BUSY ... 忙信号

CD、/CE ... 芯片使能信号

RE、/RE ... 刷新使能信号

/RT ... 刷新计时器信号

A1、A2 ... 存取指令（常规存取操作）

BE ... 组块使能信号

BK ... 存取阵列组块

BL ... 位线对

BLEQ ... 位线均衡信号

DB ... 解码器组块

ERA ... 存取行地址信号

LT ... 锁存信号

MC ... 存储单元

R1、R2、R3 和 R4 ... 刷新指令（刷新操作）

RRA ... 刷新行地址信号

Tac ... 存取时间

Tec ... 外部循环时间

Tic ... 内部循环时间

WL ... 字线

具体实施方式

将参照附图详细说明本发明的实施例。相同的参考标记表示相同或相应的部分并且将不重复进行相同的说明。

参照图 1，表示本发明一实施例的 DRAM 10 具有存储单元阵列 12，该存储单元阵列包括 64M（$=64\times2^{20}$）个存储单元 MC 和 4K（$=4\times2^{10}$）条字线 WL。

将存储单元阵列 12 分成 16 个存取阵列组块(以下简称为“组块”) BK。每个组块 BK 包括 256 条字线 WL、与字线交叉的 16K 条（$=16\times2^{10}$)位线 BL、和连接到位线 BL 的 16K 个读出放大器(图中未示)。每个存储单元 MC 被连接到相应的字线 WL 和位线 BL。

DRAM 10 还具有行解码器 14 和用于控制行解码器 14 的行解码器控制电路 16。行解码器 14 响应行地址信号而从字线 WL 中进行选

择。按照存储单元阵列 12 的样子，将行解码器 14 分成 16 个解码器组块 DB。

图 2 示出了行解码器控制电路 16 和一个解码器组块 DB 的细节。参照图 2，DRAM 10 还具有刷新电路 17 和地址选择器 18。刷新电路 17 产生刷新使能信号/RE 并随之生成刷新行地址信号 RRA。地址选择器 18 选择外部施加的存取行地址信号 ERA 或刷新行地址信号 RRA，并将所选的信号作为行地址信号 RA 提供给行解码器控制电路 16。行解码器控制电路 16 解码所提供的行地址信号 RA，以生成行地址解码信号 ADU 和 ADL，并将这些信号提供给行解码器 14。

图 3 示出了地址选择器 18 和刷新电路 17 的细节。参照图 3，刷新电路 17 包括刷新计时器 30、地址计数器 32 和刷新使能电路 34。刷新计时器 30 按预定周期产生刷新计时器信号/RT。地址计数器 32 响应该刷新计时器信号/RT 而增加该刷新行地址，以产生刷新行地址信号 RRA。刷新使能电路 34 响应芯片使能信号/CE 和刷新计时器信号/RT 而产生刷新使能信号/RE。

参照图 4，芯片使能信号/CE 按外部循环时间 Tec 的每个周期被激活到 L（逻辑低）电平。芯片使能信号/CE 的激活对应于存取指令的发布。当芯片使能信号/CE 被激活时，地址选择器 18 接收外部施加的存取行地址信号 ERA，并依据此信号从存储单元 MC 中读出数据。

如果将读出或刷新操作所需的内部循环时间 Tic 设为外部循环时间 Tec 的一半，则即使在读出操作期间也能够可靠地插入刷新操作。如果存储单元 MC 的保持时间是 64ms，则必须以 16μs（ = 64ms÷4K）的间隔连续选择 4K 条字线 WL，以便在此时间内刷新所有的存储单元 MC。通过如上所述的以恒定周期均匀连续地选择所有字线 WL 的刷新被称作“分布式刷新”。

在分布式刷新的情况下，通过与芯片使能信号/CE 无关地以 16μs 的周期将刷新计时器信号/RT 激活到 L 电平。当在激活刷新计时器信号/RT 之后激活芯片使能信号/CE 时，将刷新使能信号/RE 激活到 L 电平。该刷新使能信号/RE 的激活对应于刷新指令的发布。当在激活

了刷新使能信号/RE之后又经过了预定时间之后，刷新计时器30被复位，并且刷新计时器信号/RT返回到H（逻辑高）电平。当激活了刷新使能信号/RE时，地址选择器18接收由地址计数器32生成的刷新行地址信号RRA，并依据此信号来刷新存储单元MC。

如果如上所述将内部循环时间Tic设为外部循环时间Tec的一半，则刷新指令不可能与存取指令（在本说明书中为读出指令）竞争，并且能够在任何时间进行刷新。在此实施例中，即使刷新指令在内部循环时间Tic被设置为长于外部循环时间Tec一半的情况下与存取指令竞争，也能够仲裁其间的竞争以便在常规存取操作期间插入刷新操作。

再次参照图2，每个解码器组块DB包括组块使能电路20、行解码器电路22、字线驱动器24和组块控制电路26。高位行地址解码信号ADU被提供给组块使能电路20，而低位行地址解码信号ADL被提供给行解码器电路22。每个组块使能电路20响应该行地址解码信号ADU而生成组块使能信号BE，以选择相应的解码器组块DB。每个行解码器电路22响应行地址解码信号ADL而选择相应的256条字线WL中的一条。字线驱动器24驱动所选的字线WL。在此实施例中，提供了12位行地址信号RA，其中的4位信号被用来选择组块BK，另8位信号用来选择字线WL。

响应组块使能信号BE来激活组块控制电路26，组块控制电路26从对应的组块BK接收到时序监视信号TM，并向对应的组块BK提供阵列控制信号AC。时序监视信号TM是在对应的组块BK中生成的。阵列控制信号AC是用于针对相应的组块BK来控制读出放大器的激活、复位之后的位线预充电等的信号。也就是说，每个组块控制电路26控制对应的组块BK，以使操作序列按自完成方式被完成。

此实施例的特征在于DRAM 12还具有一条用于产生忙信号BUSY的忙信号线28。忙信号线28为16个组块BK所共用，并以平行于位线对BL的方式在行解码器14中延伸。

图5示出了用于产生忙信号/BUSY的电路。参照图5，每个组块

控制电路 26 包括阵列存取时序控制电路 36、延迟电路 38 和 n 沟道 MOS 晶体管 40。阵列存取时序控制电路 36 向对应的组块 BK 提供包括位线均衡信号 BLEQ 在内的各种阵列控制信号 AC。延迟电路 38 将该位线均衡信号 BLEQ 延迟预定的时间。晶体管 40 响应延迟后的位线均衡信号 BLEQ 而导通，以将忙信号线 28 上的电压下拉至接地电压 GND。

行解码器控制电路 16 包括 p 沟道 MOS 晶体管 42 和反相器 44。晶体管 42 响应阵列使能信号/AE 而导通，以便将忙信号线 28 上的电压上拉至电源电压 VDD。阵列使能信号/AE 是响应芯片使能信号/CE 或刷新使能信号/RE 而临时产生的脉冲信号。

当对一个组块 BK 开始常规存取操作或刷新操作时，将阵列使能信号/AE 的脉冲施加到晶体管 42 的栅极。由此上拉忙信号线 28 的电压，以将忙信号/BUSY 预充电至 H 电平。通过反相器 44 将忙信号/BUSY 设为 L 电平，以指示正在操作该一个组块 BK，由此禁止启动下一个常规存取操作或刷新操作。

在完成了对该所选组块的操作序列之后，并在从输出位线均衡信号 BLEQ 的时间起经过了预定时间之后，晶体管 40 导通。由此下拉忙信号线 28 的电压以使忙信号/BUSY 返回到 L 电平。通过反相器 44 使忙信号/BUSY 返回到 H 电平，以指示完成了对该组块 BK 的操作。由此取消对下一个操作的禁止。

如上所述，当没有组块 BK 被选择时，忙信号/BUSY 保持在 H 电平，而当选择了一个组块 BK 时忙信号/BUSY 被设为 L 电平。在完成对该所选组块 BK 的操作序列之前，忙信号/BUSY 保持在 L 电平。将忙信号/BUSY 从行解码器控制电路 16 提供至地址选择器 18。也就是说，晶体管 42 依据存取指令或刷新指令对忙信号线 28 进行充电，并在完成了对相应的组块 BK 的存取操作或刷新操作时，对忙信号线 28 进行放电。忙信号线 28、晶体管 42 和对应于 16 个组块 BK 而提供的 16 个晶体管 40 是用于响应存取指令或刷新指令来激活忙信号/BUSY 以及在完成了对由组块使能电路 20 选择的组块 BK 的常规存

取操作或刷新操作时使忙信号/BUSY 无效的装置。

当忙信号 BUSY 是 L 电平时，没有组块 BK 被选择并且行解码器控制电路 16 因此被激活，以将行地址解码信号 ADU 和 ADL 提供给行解码器 14。一旦选择了一个组块 BK，忙信号 BUSY 就被激活到 H 电平，但是行地址解码信号 ADU 和 ADL 仍维持在相同的状态。不管行地址信号 RA 如何变化，在完成对前述组块 BK 的操作而使忙信号 BUSY 返回到 L 电平之前，行地址解码信号 ADU 和 ADL 都没有改变。

图 6 示出了地址选择器 18 的构造。参照图 6，地址选择器 18 包括 NAND（与非）电路 46 至 49、反相器 50 和 51、NOR（或非）电路 52 和 D 型锁存电路 54。提供了各自包含 N 个电路的 NAND 电路 46 至 48，以及 N 个 D 型锁存电路 54。在此实施例中，由于行地址信号 ERA、RRA 和 RA 是 12 位的信号，所以 N = 12。当芯片使能信号/CE 是 L 电平时，12 个 NAND 电路 46 输入 12 位的存取行地址信号 ERA。当刷新使能信号/RE 是 L 电平时，12 个 NAND 电路 47 输入 12 位的刷新行地址信号 RRA。12 个 NAND 电路 48 输出所输入的 12 位存取行地址信号 ERA 或 12 位刷新行地址信号 RRA。

当忙信号/BUSY 是 H 电平时，NAND 电路 49 用作反相器。因此，当芯片使能信号/CE 或者刷新使能信号/RE 变成 L 电平时，从 NAND 电路 49 提供给 12 个锁存电路 54 的锁存信号 LT 变成 H 电平。当锁存信号 LT 变成 H 电平时，12 个锁存电路 54 接收并锁存从 12 个 NAND 电路 48 输出的 12 位存取行地址信号 ERA 或刷新行地址信号 RRA，并输出被锁存的信号作为 12 位行地址信号 RA。简言之，如果忙信号/BUSY 是 H 电平，则地址选择器 18 在芯片使能信号/CE 为 L 电平时选择存取行地址信号 ERA，在刷新使能信号/RE 为 L 电平时选择刷新行地址信号 RRA。

另一方面，当忙信号/BUSY 是 L 电平时，锁存信号 LT 被固定在 H 电平。只要忙信号/BUSY 是 L 电平，即使在芯片使能信号/CE 或者刷新使能信号/RE 变成了 L 电平以及输入了下一个新的存取行地址信号 ERA 或刷新行地址信号 RRA 时，锁存电路 54 仍继续锁存旧的存

取行地址信号 ERA 或刷新行地址信号 RRA，而不接收下一个新的存取行地址信号 ERA 或刷新行地址信号 RRA。换言之，在忙信号/BUSY 是 L 电平时，地址选择器 18 的操作为：即使芯片使能信号/CE 或者刷新使能信号/RE 变成了 L 电平，地址选择器 18 也忽略随后提供的存取行地址信号 ERA 或刷新行地址信号 RRA，继续输出上次选择的存取行地址信号 ERA 或刷新行地址信号 RRA 而不选择随后提供的信号。

参照图 7，当芯片使能信号 CE 被激活时，启动对所选组块 BK 的存取操作，并将忙信号/BUSY 激活到 L 电平。当完成了该存取操作时，忙信号/BUSY 返回到 H 电平。另一方面，当刷新使能信号 RE 被激活时，启动对所选组块 BK 的刷新操作，并将忙信号/BUSY 激活到 L 电平。当完成了刷新操作时，忙信号/BUSY 返回到 H 电平。

如上所述，当忙信号/BUSY 返回到 H 电平时，DRAM 10 依据存取指令和刷新指令中先到的一个指令来确定随后将执行的操作。这样，外部施加的存取行地址信号 ERA 与内部产生的刷新行地址信号 RRA 彼此没有区别，并且在完成对先前组块 BK 的操作之前，依据新的行地址信号 RA 的操作被推迟。也就是说，DRAM 10 优先执行依据在另一个指令之前到来的指令的操作，并在完成在先操作之前，推迟依据后续指令的操作。

在通过设定内部循环时间 Tic 长于外部循环时间 Tec 的一半来执行分布式刷新的情况下，存在刷新指令与存取指令竞争的趋势，并且在出现竞争时必须推迟刷新。因此，在此实施例中，最好是按照在最短的时间内通过所有 256 条字线 WL 连续地对每个组块 BK 进行集中式刷新的方式，以组块为单位来执行集中式刷新。

为了以 64ms 的间隔刷新每个存储单元 MC，以 4ms（ = 64ms÷16）的间隔将集中式刷新开始信号提供给 16 个组块 BK 中的每一个组块，并且通过 256 条字线 WL 在每个组块 BK 中连续地执行集中式刷新。因此，在每个组块 BK 中，以 4ms 的周期执行 256 次刷新。实际上，即使在执行一次刷新所需的时间是 50ns 的情况下，集中式刷新所需的

时间是 12.8μs（ =256×50ns），与 4ms 相比是极短的。因此，集中式刷新在 4ms 周期中最初很短的时间内被完成。当常规的存取指令在集中式刷新期间到来时，刷新被推迟。但是，在以组块为单位进行集中式刷新的情况下，刷新中的延迟在对每个组块 BK 的操作过程中被吸收，并且不会延续至任何其它的组块 BK，如下面详细的说明。

图 8 示出了在存取指令 A1 和 A2 按最小外部循环时间 Tec 的每个周期连续到来的情况下的集中式刷新操作。图 8（A）示出了如现有技术中的内部循环时间 Tic 为外部循环时间 Tec 的一半的情况，而图 8（B）示出了内部循环时间 Tic 长于外部循环时间 Tec 的一半的情况。下面将针对如下情况进行说明:刷新指令 R1 在存取指令 A1 紧前到来，并且由此启动了刷新操作 R1（用与对应的指令相同的参考符号来表示），从而对常规存取操作 A1 的循环时间和存取时间而言，都导致最差的条件。

参照图 8（A），当刷新指令 R1 在存取指令 A1 紧前到来时，首先启动刷新操作 R1。该刷新操作 R1 在经过内部循环时间 Tic 之后被完成。由于此刷新为集中式刷新，所以每在完成在先的常规存取操作或刷新操作后就发布刷新指令。因此当完成了刷新操作 R1 时，另一个刷新指令 R2 到来。但是此时，由于存取指令 A1 在刷新指令 R2 到来之前的时刻 T0 到来，所以依据该存取指令 A1 而启动常规存取操作 A1。在经过内部循环时间 Tic 之后常规存取操作 A1 也被完成。重复此操作序列，按集中式刷新方式的刷新操作 R1 和 R2 以及常规存取操作 A1 和 A2 被交替执行。下面将更具体的说明此过程。

地址选择器 18 响应处于 L 电平的刷新使能信号/RE 而锁存刷新行地址信号 RRA，并将锁存的刷新行地址信号 RRA 提供给行解码器控制电路 16。行解码器控制电路 16 将忙信号/BUSY 激活至 L 电平，并响应刷新行地址信号 RRA 将行地址解码信号 ADU 和 ADL 提供给行解码器 14。响应该行地址解码信号 ADU 而选择一个组块 BK，并在组块 BK 中响应该行地址解码信号 ADL 而激活一条字线 WL，以刷新连接到该字线 WL 的所有存储单元 MC。

在此刷新操作 R1 期间，将芯片使能信号/CE 激活至 L 电平，以将存取行地址信号 ERA 提供给地址选择器 18。但是，由于忙信号/BUSY 已经被激活，所以地址选择器 18 不锁存该存取行地址信号 ERA，而继续锁存上次被锁存的刷新行地址信号 RRA。

当在所选组块 BK 中完成了刷新操作 R1 时，使忙信号/BUSY 被无效为 H 电平。由此，地址选择器 18 锁存已给出的存取行地址信号 ERA，并将此信号提供给行解码器控制电路 16。因此，在所选的组块 BK 中执行常规的存取操作 A1。

在情况（A）下，由于内部循环时间 Tic 为外部循环时间 Tec 的一半，所以在外部循环时间 Tec 内完成每个常规存取操作。图中的箭头表示从输入的存取指令起到完成常规存取操作。如 SRAM 的情况一样，箭头所表示的存取时间在外部循环时间 Tec 以内。

在情况（B）下，虽然可以跳过刷新指令，但是每个存储单元 MC 在被刷新的同时按外部循环时间 Tec 的每个周期被存取。

将参照图 9 来说明在内部循环时间 Tic 被设置为长于外部循环时间 Tec 的一半的情况下内部循环时间 Tic 能够被延长的程度。

内部循环时间 Tic 相对于外部循环时间 Tec 的一半越长，插入刷新操作的频率就越小。因此，就需要用于在一定数量的常规存取操作之后允许可靠地插入至少一个刷新操作的条件。在用于第一个刷新操作的内部循环时间（1×Tic）之后插入 N 个常规存取操作。如果 N 个常规存取操作所需的时间（N×Tic）在 N 个外部循环时间构成的时间段（N×Tec）内，则刷新指令在第（N + 1）个常规存取指令之前到来，以启动刷新操作。因此，下面的表达式（1）给出了刷新操作的插入条件：

$$Tic + N \times Tic < N \times Tec \quad \ldots \quad (1)$$

修改表达式（1）以获得下面的表达式（2）：

$$Tic < N/(N + 1) \times Tec \quad \ldots \quad (2)$$

表达式（2）示出了如果内部循环时间 Tic 在外部循环时间 Tec 的 N/（N + 1）倍之内，则在第（N + 1）个常规存取操作之前插入刷

新操作。例如，在 N = 1 的情况下，如果内部循环时间 Tic 短于外部循环时间 Tec 的一半，则每隔一个周期插入一次刷新操作，如图 9（A）所示。

从表达式（2）明显可知，如果 N 被增加，则内部循环时间 Tic 变得更接近于外部循环时间 Tec。也就是说，如果在刷新操作的插入频率相当小时也没有问题，则内部循环时间 Tic 可以被设置为基本上接近于外部循环时间 Tec。

如果 N 是如图 9（A）至 9（E）所示的有穷数，则刷新指令被跳过 N 次。如果 N 是无穷数，则内部循环时间 Tic 与外部循环时间 Tec 相同，刷新指令被跳过无穷次，且没有刷新操作被插入，如图 9（F）所示。即使刷新指令在第一存取指令紧前到来以插入刷新操作，存取指令也必须在完成在先的存取操作之前的一个循环到来，因此在第一存取操作之后不插入刷新操作。如果 N 不是无穷的并且内部循环时间 Tic 稍短于外部循环时间 Tec，则必然插入刷新操作。

然后获得可毫无例外地插入刷新操作的上限值的设定。如果每个组块 BK 的字线数是 Nwlb，则满足使通过以 N×Tec 乘以此数值而获得的值被设定为小于用保持时间 Tr 除以组块的数量 Nb 而获得的值即可。因此获得下面的表达式（3）。

$$N \times Tec \times Nwlb < Tr/Nb \quad \ldots \quad (3)$$

由于 Nwlb ×Nb 是字线的总数 Ntwl，所以获得使用此数值修改表达式（3）后的如下表达式（4）。

$$N < Tr/(Tec \times Ntwl) \quad \ldots \quad (4)$$

如果保持时间是 64ms 的典型值、在此实施例中字线总数 Ntwl 为 4K、以及外部循环时间是 50ns，则 N 的上限值基本上是约为 312 的较大数值。

如果将 N = 312 代入表达式（2），则即使内部循环时间 Tic 是 49.85μs，与外部循环时间 Tec 之比为 0.997（ = 312/313）/1、即为外部循环时间 Tec 的 99.7%，也必须在 312 个循环之后至少插入一次刷新操作，以便能够必然执行通过所有字线的刷新，同时按外部循环时

间 Tec 连续插入常规存取操作。

但是，即使在 N 不是这么大的数值的情况下，内部循环时间 Tic 也基本上接近于外部循环时间 Tec。例如，当 N = 4 时，内部循环时间 Tic 可被增加到外部循环时间 Tec 的 4/5（80％），也就是说，刷新操作以 1 比 4 个常规存取操作的比例被插入，如图 9（D）所示。就插入刷新操作的频率而言，即使外部循环时间是 50ns，执行 256 次集中式刷新所需的时间也为 64μs（ = 5×50ns×256）。在此情况下，通过第 256 条字线的刷新被最大地延迟。但是该刷新延迟仅为 51.2μs（ = 64μs－（50ns×256））。这个值仅为保持时间 64ms 的 0.08%，能够被完全忽略。

由于刷新是以组块为单位、按集中式刷新而被执行的，所以刷新延迟当然能够在对该组块的操作过程中被吸收，并且不会延续至任何其它的组块，也不会被累积。51.2μs 的延迟是通过所有字线的最大延迟。因此，依据本实施例，基本上不存在由刷新延迟导致的问题并且内部循环时间能被增加 Tic，到接近于外部循环时间 Tec。相反地，通过利用可按内部循环时间 Tic 工作的 DRAM 10 的几乎所有实际能力，能够实现高速化。因此，能够提供一种可在内部执行刷新的 SRAM 兼容型 DRAM，并且能够实现接近于已有 DRAM 的外部循环时间的一半的外部循环时间 Tec。

因此，就“循环时间”而言，可以说只要 N 是有穷数，即使在内部循环时间 Tic 长于外部循环时间 Tec 的一半时，也能够毫无问题地在外部循环时间 Tec 内执行常规存取操作和刷新操作。但是，就常规存取操作的“存取时间”而言，仍然存在问题。即，在 SRAM 的常规情况下，通常循环时间和存取时间彼此相等。因此，还期望在此 DRAM 10 中，数据读出在外部循环时间 Tec 内是有效的。但是，如图 8（B）所示，在外部循环时间 Tec 内最先读出的数据（表示存取时间的箭头的尖端）不是有效的，并且存取时间 Tac 不满足常规的 SRAM 标准。从附图明显可见，为了使存取时间 Tac 满足该标准，必须将用于刷新操作的内部循环时间 Tic 与存取时间 Tac 之和设置在外部循环时间 Tec 以内。在上述实施例中，用于刷新操作的内部循环时间 Tic 与用于常

规存取操作的内部循环时间 Tic 彼此相等。但是，在常规存取操作的情况下，尽管第一数据存取时间没有改变，作为出于某些原因、例如页或突发读出（burst readout）而导致没有立即启动预充电的结果，在某些 DRAM 中也可以增加循环时间。在这样的情况下，即使用于常规存取操作的内部循环时间 Tic 很长，也不必增加外部循环时间 Tec 和存取时间。

此外，如图 8（B）所示，紧随在刷新操作之后的存取时间 Tac 与后续的常规存取操作之后的存取时间 Tac 彼此不同。因此，由于此问题而令用户难以使用该 DRAM。此时，可以使用一种方法，如图 10 所示，此方法在标准中有意设置了存取等待时间 Tlt，以便将用于刷新操作的内部循环时间 Tic 与用于常规存取操作的内部循环时间 Tic 之和设为明显的（apparent）存取时间，以延迟数据在连续的常规存取操作之后变为有效的时间。当然，存取时间 Tac 很长，但是能够缩短循环时间。此操作类似于 Digest of Technical Papers（ISSC91，p. 50，Feb. 1991）中所公开的流水线突发式 SRAM 中的操作。

图 10 示出了当 N = 5 时的操作情况，即在情况（A）下，只有常规存取指令到来，使得有意将存取时间 Tac 描述为在规范方面增加的时间，并且其长于外部循环时间 Tec；在情况（B）下，在常规存取指令按外部循环时间 Tec 到来时启动集中式刷新；以及在情况（C）下，只有刷新指令到来。在情况（A）和（B）下，与图 9（E）中 N = 5 时所示的情况不同，存取时间 Tac 相对于存取指令的输入总是相同的。即使在存取时间 Tac 长于外部循环时间 Tec 时，也以与外部循环时间 Tec 相同的周期连续地使数据有效。如果以此方式持续地存取数据，则能够增加带宽。

已经针对本发明的实施例说明了本发明。但是，上述实施例仅作为本发明实施例的示例，本发明并不限于上述的实施例。在不脱离本发明主旨的情况下，能够通过适当地修改上述实施例来实施本发明。

本发明的半导体存储器件特别在低功耗应用中能够用作代替 SRAM 的 DRAM。

图1

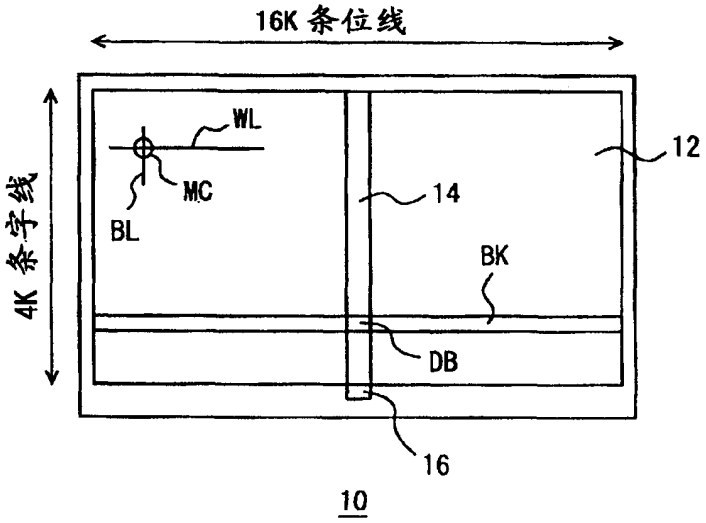


图 2

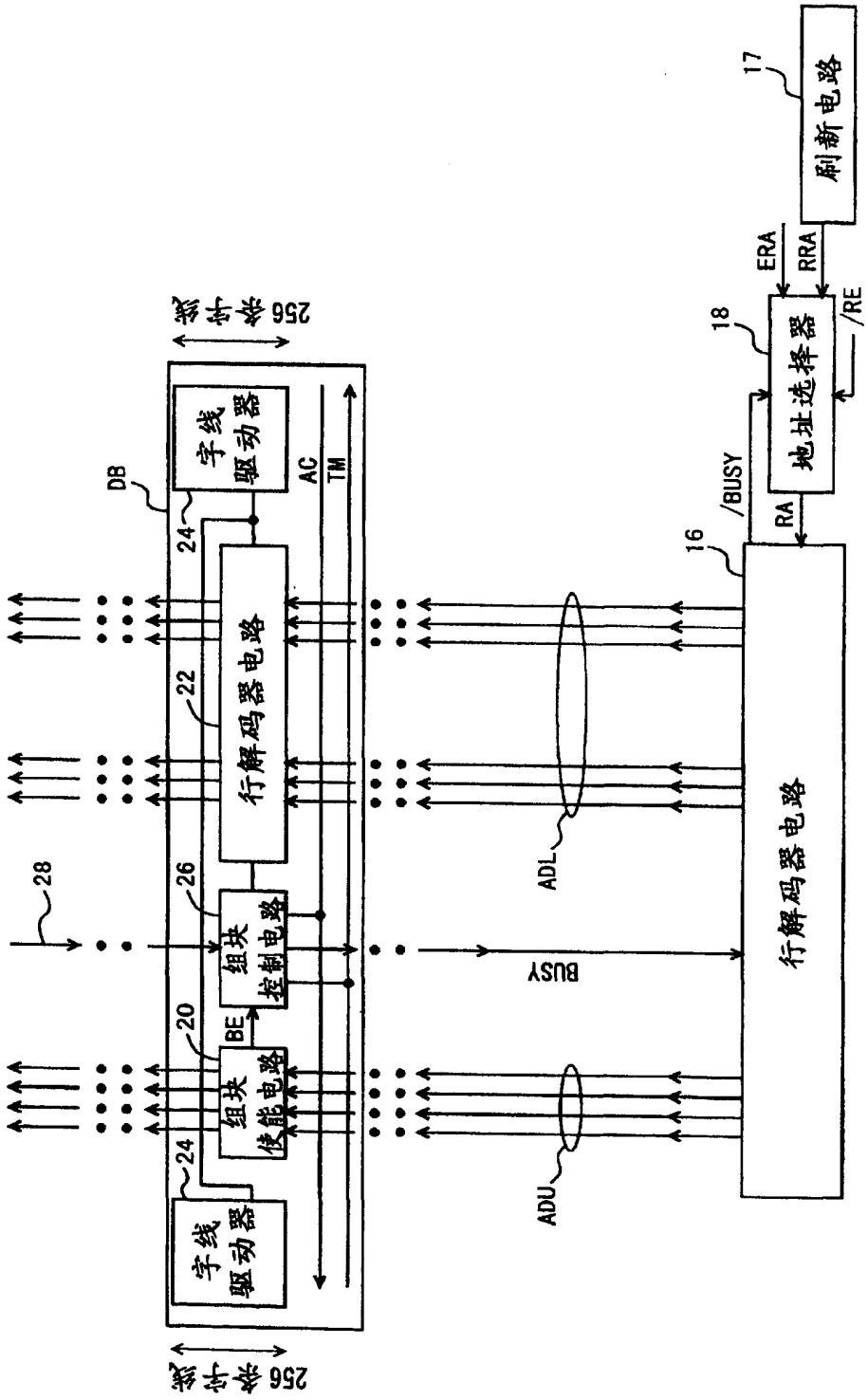

刷新电路
17
ERA
RRA
/RE
地址选择器
18
/BUSY
RA
16
行解码器电路
ADL
ADU
BUSY
256 条 字 线
字线驱动器
DB
24
AC
TM
行解码器电路
22
28
26
组块控制电路
BE
20
组块使能电路
24
字线驱动器
256 条 字 线

图 3

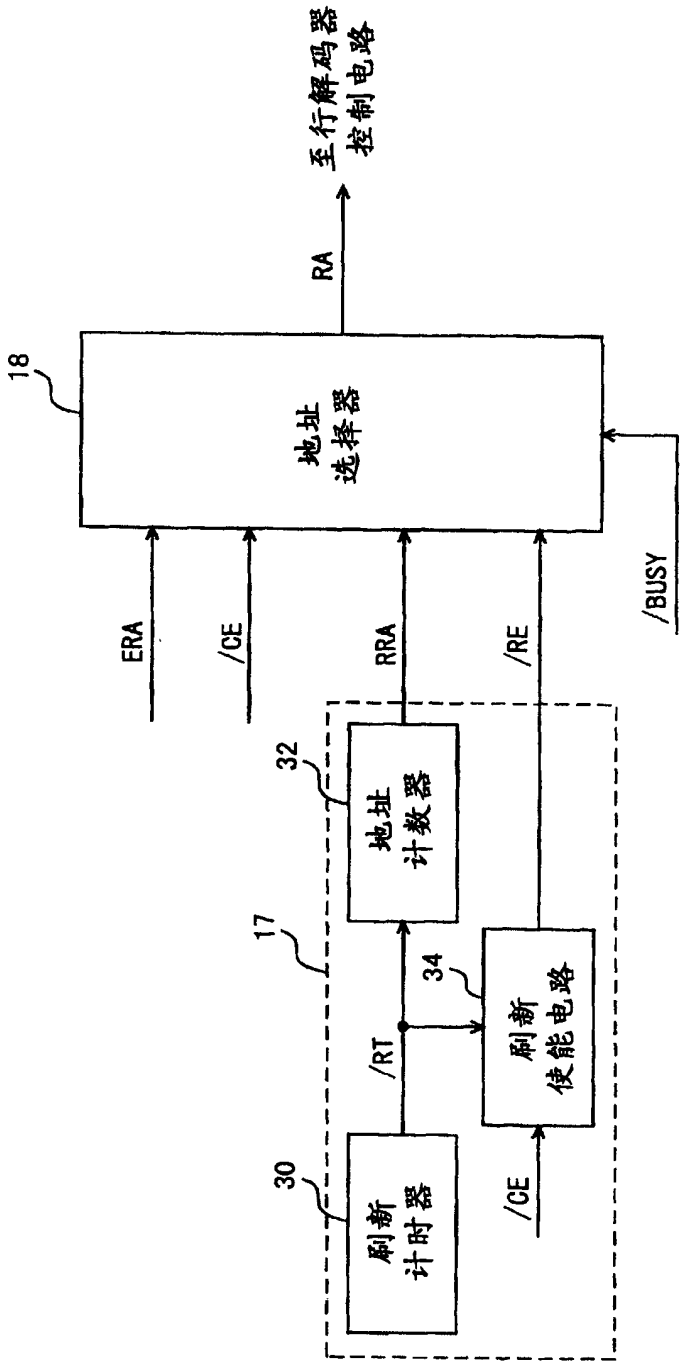

至行解码器
控制电路
RA
18
地址
选择器
ERA
/CE
RRA
/RE
/BUSY
32
地址
计数器
17
34
/RT
刷新
使能电路
30
刷新
计时器
/CE

图4

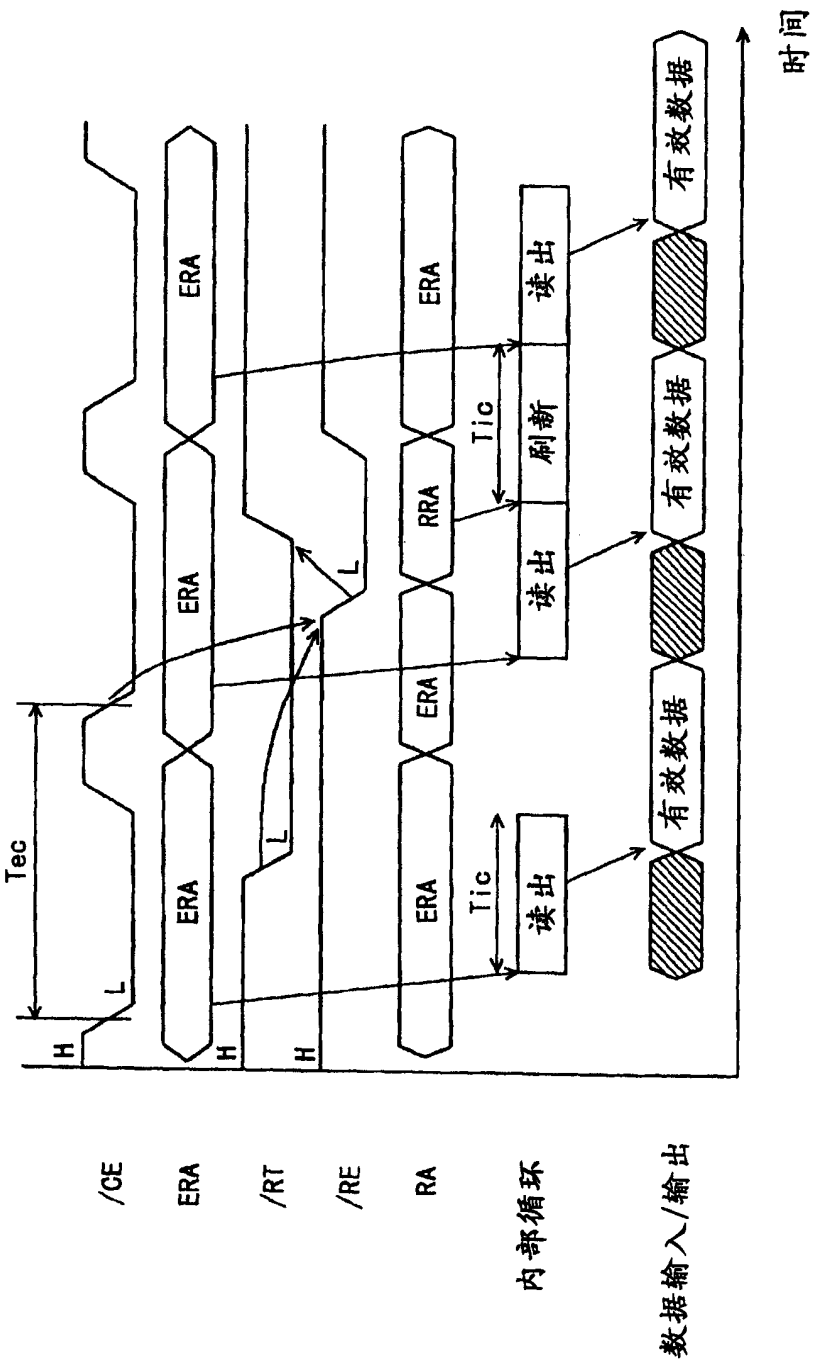


图5

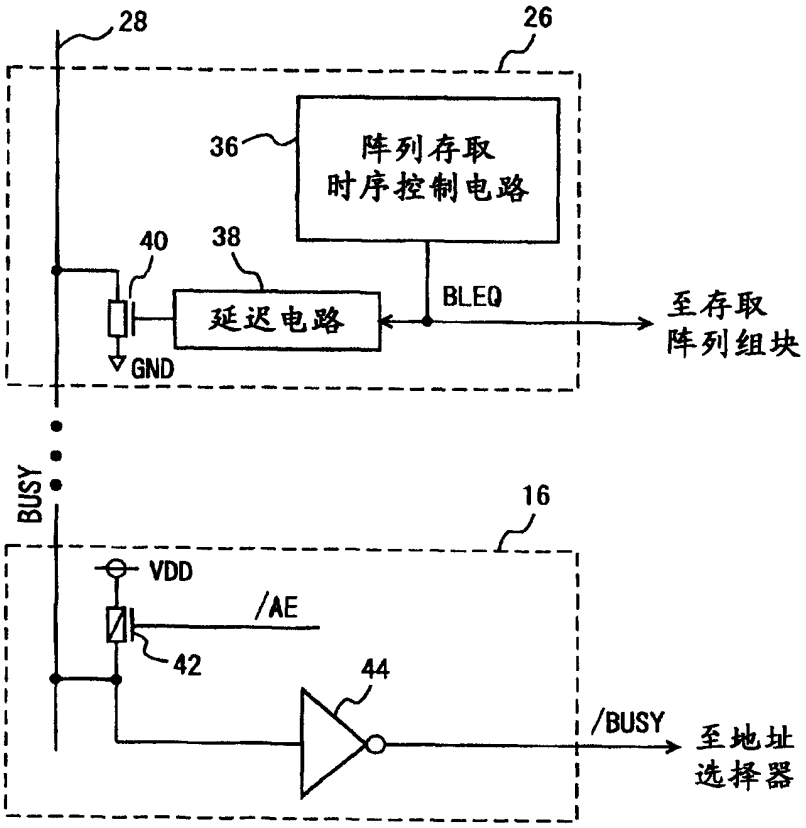

28
26
36
阵列存取
时序控制电路
40
38
延迟电路
BLEQ
至存取
阵列组块
GND
BUSY
16
VDD
/AE
42
44
/BUSY
至地址
选择器

图6

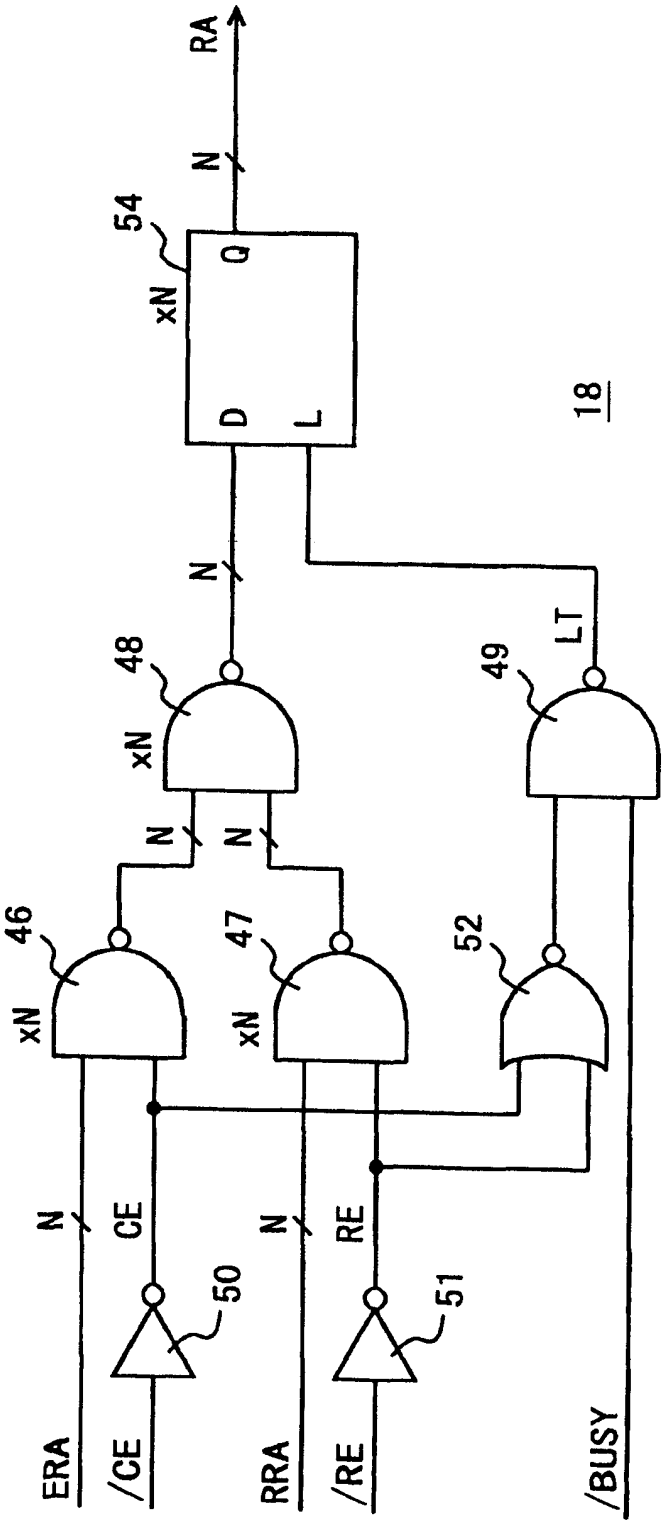

RA
N
54
xN
Q
D
L
18
N
48
xN
LT
49
N
N
46
xN
47
xN
52
N
CE
50
N
RE
51
ERA
/CE
RRA
/RE
/BUSY

图7

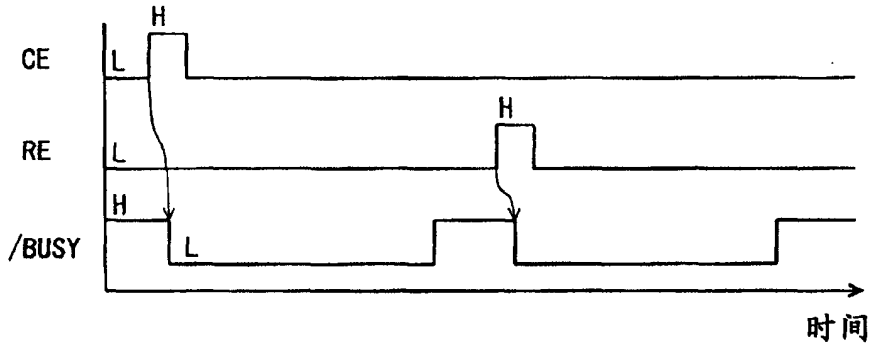


图8

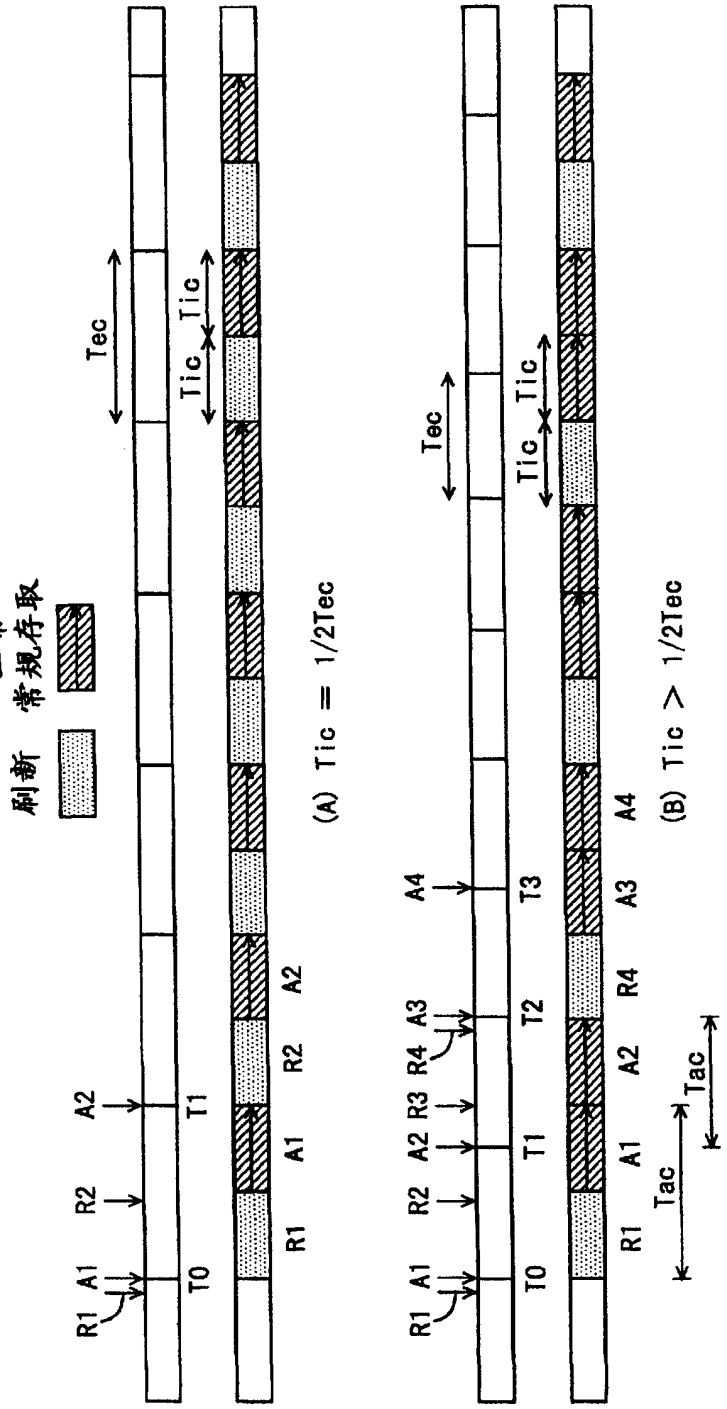
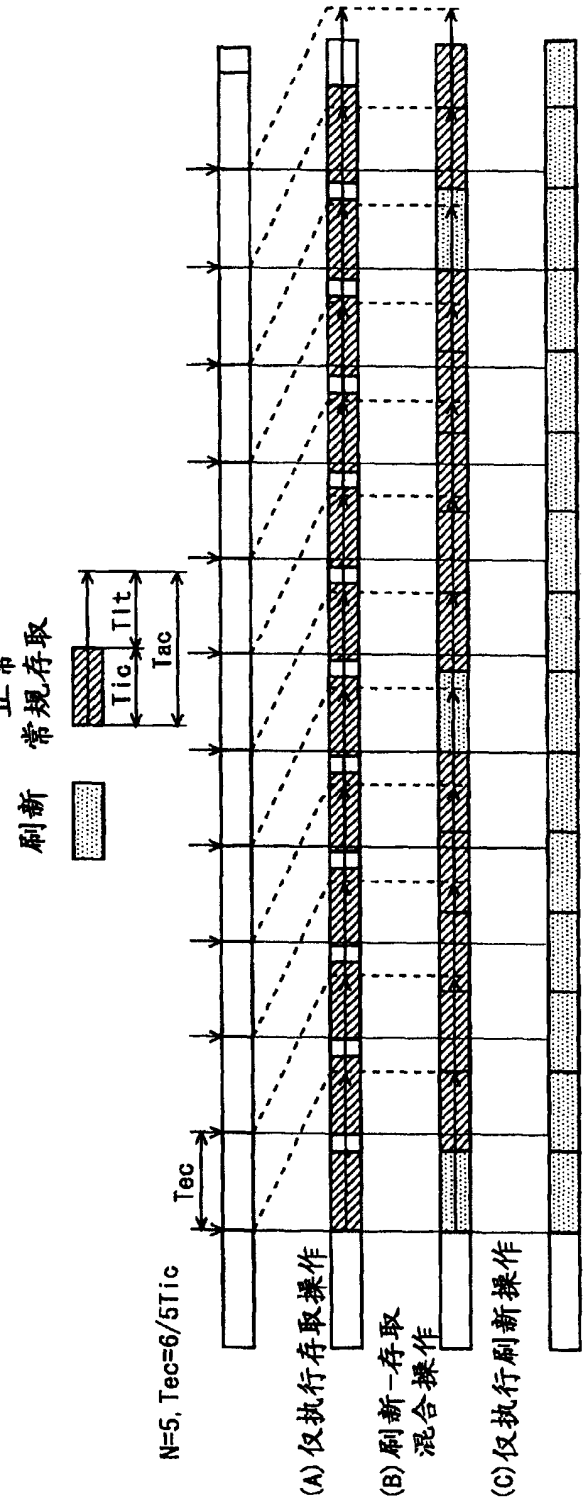

正常
刷新 常规存取
Tec
Tic
Tic
R1 A1 R2 A2
T0 T1
R1 A1 R2 A2
(A) Tic = 1/2Tec
Tec
Tic
Tic
R1 A1 R2 A2 R3 A3 R4 A4
T0 T1 T2 T3
R1 A1 A2 R4 A3 A4
Tac
Tac
(B) Tic > 1/2Tec

图9


刷新
正常
常规存取
Tec
(A) N=1, Tec=2Tic
(B) N=2, Tec=3/2Tic
(C) N=3, Tec=4/3Tic
(D) N=4, Tec=5/4Tic
(E) N=5, Tec=6/5Tic
(F) N=∞, Tec=Tic

图10


正常
常规存取
刷新
Tic
Tlt
Tac
Tec
N=5, Tec=6/5Tic
(A)仅执行存取操作
(B)刷新-存取
混合操作
(C)仅执行刷新操作