



(12) 发明专利申请

(10) 申请公布号 CN 104347118 A

(43) 申请公布日 2015.02.11

(21) 申请号 201410375592.5

G11C 16/06(2006.01)

(22) 申请日 2014.08.01

(30) 优先权数据

13/957604 2013.08.02 US

(71) 申请人 英飞凌科技股份有限公司

地址 德国瑙伊比贝尔格市坎茨昂 1 - 12 号

(72) 发明人 U. 巴克豪森 毕向婷 T. 柯恩

T. 尼尔施尔 E. 帕帕里斯托

J. 罗森布施

(74) 专利代理机构 中国专利代理(香港)有限公司

72001

代理人 申屠伟进 胡莉莉

(51) Int. Cl.

G11C 16/14(2006.01)

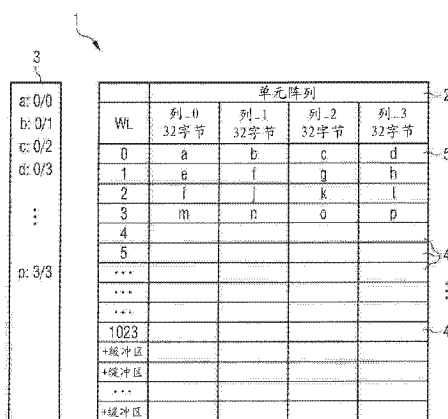
权利要求书2页 说明书8页 附图8页

(54) 发明名称

用于仿真电可擦除可编程只读存储器的系统及方法

(57) 摘要

本发明涉及用于仿真电可擦除可编程只读存储器的系统及方法。本公开涉及电子存储器系统，并且更具体地涉及用于仿真电可擦除可编程只读存储器的系统以及用于仿真电可擦除可编程只读存储器的方法。根据本公开的实施例，提供用于仿真电可擦除可编程只读存储器的系统，该系统包含第一存储器部件和第二存储器部件，其中该第一存储器部件包括多个存储位置，所述多个存储位置被配置成存储划分成多个数据片段的数据，并且其中该第二存储器部件被配置成存储把在第一存储器部件中存储的数据片段的物理地址映射到该数据片段的逻辑地址的信息。



1. 一种用于仿真电可擦除可编程只读存储器的系统,该系统包括:  
第一存储器部件;以及  
第二存储器部件,  
其中所述第一存储器部件包括多个存储位置,所述存储位置被配置成存储划分成多个数据片段的数据,并且其中第二存储器部件被配置成存储把在第一存储器部件中存储的数据片段的物理地址映射到所述数据片段的逻辑地址的信息。
2. 权利要求 1 的所述系统,其中所述第一存储器部件包括非易失性存储器阵列。
3. 权利要求 2 的所述系统,其中所述第二存储器部件包括随机访问存储器(RAM)。
4. 权利要求 2 的所述系统,其中所述第一存储器部件被配置成通过单个访问路径被访问以读取、写入和擦除数据片段。
5. 权利要求 2 的所述系统,其中所述系统被配置成对于所述非易失性存储器阵列的每个存储位置来监控为了写入或擦除而已访问数据片段的次数。
6. 权利要求 2 的所述系统,其中所述系统被配置成对于存储在所述非易失性存储器阵列中的每个数据片段来监控该数据片段已接收的干扰的数目。
7. 权利要求 6 的所述系统,其中所述系统被配置成当所述数据片段已接收的干扰的数目超过干扰阈值时移动数据片段到在所述非易失性存储器阵列中新的存储位置。
8. 权利要求 1 的所述系统,其中所述第二存储器部件被配置成用纠错码来保护在所述第二存储器部件中存储的数据。
9. 权利要求 2 的所述系统,其中所述系统被配置成标记在所述非易失性存储器阵列中的数据片段,所述数据片段将因被标记用于擦除而被擦除。
10. 权利要求 9 的所述系统,其中在所述非易失性存储器阵列中的所述存储位置被布置在行和列的阵列中,并且其中所述系统被配置成每当在行或列中为了擦除而标记的数据片段的数目达到擦除阈值时同时擦除所述行或所述列的为了擦除而标记的所有数据片段。
11. 权利要求 10 的所述系统,其中每列被布置在分离的井上。
12. 一种用于仿真电可擦除可编程只读存储器的方法,包括:  
在包括多个存储位置的第一存储器部件中存储多个数据片段;  
对于在所述第一存储器部件中存储的每个数据片段:在第二存储器部件中存储把在第一存储器部件中的所述数据片段的存储位置的物理地址映射到逻辑地址的映射信息;以及  
通过数据片段的所述逻辑地址访问所述数据片段。
13. 权利要求 12 的所述方法,进一步包括:  
对于所述第一存储器部件的每个存储位置,监控为了写入或擦除而已访问存储位置的次数。
14. 权利要求 13 的所述方法,进一步包括:  
当在第一存储器部件中写入数据片段到存储位置时选择如下存储位置:为了写入或擦除而已访问所述存储位置的次数不超过预设的阈值。
15. 权利要求 12 的所述方法,进一步包括:  
对于在所述第一存储器部件中存储的每个数据片段,监控所述数据片段已接收的干扰的数目。
16. 权利要求 15 的所述方法,进一步包括:

当数据片段已接收的干扰的数目超过干扰阈值时,移动数据片段到在第一存储器部件中新的存储位置。

17. 一种用于在存储器系统中仿真电可擦除可编程只读存储器的方法,所述存储器系统包括包含多个在行与列的阵列中布置的存储位置的非易失性存储器阵列以及随机访问存储器(RAM),所述方法包括:

在所述非易失性存储器阵列中存储多个数据片段;

对于在非易失性存储器阵列中存储的每个数据片段:在RAM中存储映射信息,所述映射信息映射在所述非易失性存储器阵列中的所述数据片段的存储位置的物理地址到逻辑地址;以及

通过数据片段的所述逻辑地址来访问所述数据片段以进行读取、擦除和编程。

18. 权利要求17的所述方法,进一步包括:

当为了擦除而访问数据片段时,标记为了擦除而标记的所述数据片段。

19. 权利要求18的所述方法,进一步包括:

对于所述非易失性存储器阵列的每行和每列,监控在该行或该列中为了擦除而标记的数据片段的数目;以及

如果在行或列中为了擦除而标记的数据片段的数目达到擦除阈值时:擦除在所述行或所述列中为了擦除而标记的所述数据片段。

20. 权利要求19的所述方法,进一步包括:

对于在所述非易失性存储器阵列中存储的每个数据片段,监控所述数据片段已接收的干扰的数目;以及

当数据片段已接收的干扰的数目超过干扰阈值时:移动数据片段到在所述非易失性存储器阵列中新的存储位置。

## 用于仿真电可擦除可编程只读存储器的系统及方法

### 技术领域

[0001] 本公开大体涉及电子存储器系统,并且更具体地涉及用于在非易失性存储器器件中仿真电可擦除可编程只读存储器的系统,以及在非易失性存储器器件中仿真电可擦除可编程只读存储器的方法。

### 背景技术

[0002] 存在多种多样的为应用的各种领域而优化的常规的非易失性存储器系统。基于应用的预期领域,可以选择适当的存储器系统。这些常规的存储器系统的性质在许多方面变化。这些方面中的两个是保持力和耐久力。两者都受各种因素诸如存储器系统的个别存储器单元的特性和受存储器系统的体系结构影响。

[0003] 保持力涉及存储器单元维持它已被编程至的状态即使在存储器单元中存储的数据保持完好无损的能力。具有高保持力的存储器系统能够在长时间内存储数据,而不需要刷新该数据。在具有低保持力的存储器系统中,在存储器系统中存储的数据不得不频繁地被刷新。

[0004] 耐久力涉及存储器单元能够被擦除或被编程的次数。在具有高耐久力的存储器系统中,存储器系统的每个存储器单元能够被擦除或被编程许多次,而不受损害。与此相反,在具有低耐久力的存储器系统中,每个存储器单元能够被擦除或被编程仅有限的次数。频繁地擦除或编程具有低耐久力的存储器系统的存储器单元将会损害存储器单元。

[0005] 常规的存储器系统大体显出高保持力或高耐久力。当存储器系统的保持力为高时,该存储器系统的耐久力大体为低。类似地,当存储器系统的耐久力为高时,存储器系统的保持力大体为低。所以,在采用存储器系统来存储只需要很少被更新的数据的情形中,通常采用具有高保持力的存储器系统。相应地,在采用存储器系统来存储频繁被更新的数据的情形中,通常采用具有高耐久力的存储器系统。从而,对于用来存储程序数据的程序存储器来说,使用显出高保持力(和从而低耐久力)的存储器系统是适宜的,然而,对于用来存储应用数据的数据存储器来说,使用显出高耐久力(和从而低保持力)的存储器系统是适宜的。这样的具有高耐久力和低保持力的存储器系统的示例是电可擦除可编程只读存储器(EEPROM)。

[0006] 然而,对于许多包括存储器系统的器件来说,在设计该器件的时候并不知道在存储器系统中将随后存储什么数据类型。所以,在这些情况下,不可能提供具有适当类型的存储器系统的器件。例如,包括具有高保持力和低耐久力的存储器系统的器件被大体优化来存储程序数据。然而,对于器件的具体应用来说,可能必要的是在存储器系统中存储应用数据。在这种情况下,需要增加存储器系统的耐久力。这能够通过使用另一种类型的存储器系统仿真具体类型的存储器系统来实现。

[0007] 常规的仿真方案遇到大量的缺陷。这些方案中的一些,例如,要求存储器空间量远大于要在存储器系统中存储的数据量。用这样的方案,不可能建立小型和廉价的存储器系统。在其他的方案中,存储器系统必须提供几个路径来访问该存储器用于读取和编程。这

再次增加该存储器系统的空间需求。其他的方案允许数据只在大的块中编程和擦除,并且要求大量的擦除操作。在大多数的实现中,这花费很长的时间。

[0008] 为了这些或其他原因,需要用于仿真电可擦除可编程只读存储器的改进系统和方法。

### 附图说明

[0009] 附图被包含来提供对本公开的进一步理解并且被合并在本说明书中并且构成本说明书的一部分。附图图示本公开的实施例并且与描述一起用来解释本公开的原理。将容易认识到本公开的其他实施例和本公开的许多预期优点,因为通过参考下面的详细描述它们变得更好理解。

[0010] 图 1a 描绘依照本公开的实施例在编程的第一状态下存储器系统的示例性表示;

图 1b 描绘在编程的第二状态下图 1a 的存储器系统的示例性表示;

图 1c 描绘在编程的第三状态下图 1a 和 1b 的存储器系统的示例性表示;

图 2 描绘依照本公开的另一个实施例包括多个存储位置的存储器系统的示例性表示,该多个存储位置被布置在包括 4 列和 1024 行的阵列中;

图 3 描绘依照本公开的另一个实施例包括多个存储位置的存储器系统的示例性表示,该多个存储位置被布置在包括 8 列和 512 行的阵列中;

图 4 描绘依照本公开的另一个实施例包括多个布置在包括 8 列和 512 行的阵列中的存储位置的存储器系统的示例性表示,其中字线驱动器布置在每行的中间;

图 5 描绘依照本公开的另一个实施例的存储器系统和与这个存储器系统关联的数据结构的示例性表示;

图 6 描绘依照本公开的另一个实施例用于仿真电可擦除可编程只读存储器的方法的时序图;以及

图 7 描绘依照本公开的另一个实施例用于仿真电可擦除可编程只读存储器的方法中执行的步骤。

### 具体实施方式

[0011] 在下面的详细描述中,参考附图,附图形成描述的一部分,并且在附图中通过图解的方式示出其中可以实施本发明的具体实施例。要理解的是,在不脱离本公开的范围情况下,可以利用其他实施例并且可以做出结构的或其他的改变。所以,下面具体实施方式不要以限制的意义理解,并且本公开的范围由所附的权利要求书来限定。

[0012] 图 1a 示出用于仿真 EEPROM 的系统 1 的示例性表示。该系统 1 包括第一存储器部件 2 和第二存储器部件 3。在本公开的一个实施例中,第一存储器部件 2 包括非易失性存储器阵列并且第二存储器部件 3 包括随机访问存储器(RAM)。例如,非易失性存储器阵列包括闪速存储器。有利地,第二存储器部件 3 被配置成用纠错码(ECC)来保护在第二存储器部件 3 中存储的数据。该第一存储器部件 2 包括单元阵列,该单元阵列包括多个在多个行与列中布置的存储位置 4。每个存储位置包括多个存储器单元。在图 1a 的实施例中,第一存储器部件 2 的每个存储位置 4 适合于存储 32 字节的数据。在本公开的另一个实施例中,每个存储位置适合于存储 16 字节的数据。在本公开的又一个实施例中,每个存储位置适合

于存储 64 字节的数据。该单元阵列包括 4 列和 1024 行的存储位置,即每个 32 字节的 4096 个存储位置。所以,该单元阵列具有 128kB 的容量。在一个示例中,图 1a 示出在单元阵列中存储的 16 个数据片段 5("a" - "p")。第一个数据片段 "a" 被存储在第一行和第一列中所提供的存储位置中,第二个数据片段 "b" 被存储在第一行和第二列中所提供的存储位置中,并且最后一个数据片段 "p" 被存储在第四行和第四列中所提供的存储位置中。

[0013] 而且,第一存储器部件 2 包括多个字线。连接单元阵列的存储位置 4 的每行到这些字线中的一个,即连接存储位置的第一行到字线 "0",连接存储位置的最后一行到字线 "1023"。此外,在图 1a 中第一存储器部件 2 包括多个指定为 "+缓冲区" 的存储位置。在一个实施例中,这些存储位置为特别目的而保留并且在正常的 EEPROM 仿真期间不被使用。

[0014] 用于仿真 EEPROM 的系统 1 的第二存储器部件 3 被配置成存储将在第一存储器部件 2 中存储的数据片段的物理地址映射到该数据片段的逻辑地址的信息。如在图 1a 中所示出,在第二存储器部件 3 中存储的映射信息例如指示:数据片段 "a" 被存储在第一存储器部件 2 的单元阵列的第一行和第一列("0/0")中,数据片段 "b" 被存储在第一存储器部件 2 的单元阵列的第一行和第二列("0/1")中,数据片段 "d" 被存储在第一存储器部件 2 的单元阵列的第一行和第四列("0/3")中,并且数据片段 "p" 被存储在第一存储器部件 2 的单元阵列的第四行和第四列("3/3")中。

[0015] 向用于仿真 EEPROM 的系统 1 提供第二存储器部件 3(该第二存储器部件 3 被配置成存储将在第一存储器部件 2 中存储的数据片段的物理地址映射到该数据片段的逻辑地址的信息)允许在第一存储器部件 2 中存储的数据片段通过它们的逻辑地址而被访问。数据片段的逻辑地址独立于在第一存储器部件中该数据片段的实际存储位置,即独立于该数据片段的物理地址。为了访问在第一存储器部件 2 中的数据片段,知道该数据片段的逻辑地址就足够,而不需要知道该数据片段的物理地址。为在第一存储器部件 2 中存储的每个数据片段 5 引入逻辑地址允许在仿真期间改变数据片段 5 的物理地址。这个改变不需要传达给系统的用户,因为通过它的逻辑地址(可能保持不变)仍能够访问该数据片段 5。

[0016] 在一个实施例中,当在第一存储器部件 2 中存储的数据片段 5 被更新时,该数据片段 5 的更新版本被写入到新的存储位置 4。通过将数据片段 5 的更新写入到新的存储位置 4,能够增强该存储器系统的耐久力。这是由于如下事实:更新操作将被展开到几个存储位置上,因此减少了为了更新而访问单个存储位置的次数。

[0017] 此外,在一个实施例中,每个数据片段 5 包括指示该数据片段 5 的版本的指示器。例如,这是在图 1b 中所示出的,该图 1b 示出在第一存储器部件 2 中存储的该数据片段 5 中的一些已被更新并且附加的数据片段 5 已被写入到第一存储器部件 2 之后的图 1a 的存储器系统 1。特别地,在图 1b 中所示出的存储器器件 1 的配置中,数据片段 "a" 已被更新了 4 次。第一次更新被标记为 "a1" 并且被写入到在第一存储器部件 2 的单元阵列的第五行和第一列("4/0")中所提供的存储位置。第二次更新被标记为 "a2" 并且被写入到在第一存储器部件 2 的单元阵列的第五行和第二列("4/1")中所提供的存储位置。第三次更新被标记为 "a3" 并且被写入到在第一存储器部件 2 的单元阵列的第五行和第三列("4/2")中所提供的存储位置。第四次更新被标记为 "a4" 并且被写入到在第一存储器部件 2 的单元阵列的第五行和第四列("4/3")中所提供的存储位置。当数据片段的更新版本被写入到第一

存储器部件 2 时,这个数据片段的先前版本大体能够被擦除。然而,在图 1a — 1c 的实施例中,弃用的数据片段不会立刻被擦除,而是改为被标记用于擦除。在图 1b 中,这是通过对数据片段加阴影来指示的。因为数据片段“a”的当前版本被标记为“a4”,这个数据片段在图 1b 中没有加阴影。这个数据片段的先前版本即标记为“a”、“a1”、“a2”和“a3”的数据片段在图 1b 中被标记用于擦除并且所以被示出加阴影。

[0018] 数据片段“b”、“c”、“e”、和“i”已被更新了一次。所以,这些数据片段的当前版本被标记“b1”、“c1”、“e1”、和“i1”并且弃用的数据片段已为了擦除而标记。而且,几个新的数据片段即数据片段“q”、“r”、…、“z”已被写入到第一存储器部件 2。

[0019] 在第一存储器部件 2 中存储的数据的更新被反映在第二存储器部件 3 中存储的映射信息中。如图 1b 所示出,这个映射信息现在例如指示:数据片段“a”的当前版本被存储在第一存储器部件 2 的单元阵列的行“4”和列“3”中,数据片段“b”的当前版本被存储在第一存储器部件 2 的单元阵列的行“5”和列“0”中,并且数据片段“c”的当前版本被存储在第一存储器部件 2 的单元阵列的行“5”和列“3”中。

[0020] 本公开的存储器系统被配置成对于第一存储器部件的单元阵列的每行和每列来监控在分别的行或列中的为了擦除而标记的数据片段的数目。而且,在一个实施例中,存储器系统被配置成每当在行或列中的为了擦除而标记的数据片段的数目达到擦除阈值时同时擦除为了擦除而标记的该行或该列中的所有数据片段。通过每当在行或列的弃用的数据片段的数目达到擦除阈值时同时擦除这行或这列的所有弃用的数据片段,能够减少在 EEPROM 的仿真期间必须执行的擦除操作的数目。如果例如设定擦除阈值为 4,则在每次擦除操作中 4 个数据片段将同时被擦除。所以,与在其中每个数据片段一旦它变成过时就被擦除的仿真方案相比,用本公开的仿真方案,擦除操作的数目能够被减少 75%。这是尤其有利的,因为在大多数存储器系统中执行擦除操作花费相当长的一段时间。设定擦除阈值到更高的值甚至进一步减少擦除操作的数目。然而,擦除阈值越高,为存储给定的数据量所要求的在第一存储器部件的单元阵列中的存储位置的数目就越高。

[0021] 在图 1b 中所示出的存储器系统 1 的配置中,监控在每行和每列中为了擦除而标记的数据片段的数目指示:在行“0”中为了擦除而标记 3 个数据片段;在行“1”中为了擦除而标记 1 个数据片段;在行“2”中为了擦除而标记 1 个数据片段;在行“4”中为了擦除而标记 3 个数据片段;在列“0”中为了擦除而标记 4 个数据片段;在列“1”中为了擦除而标记 2 个数据片段;并且在列“2”中为了擦除而标记 2 个数据片段。当该擦除阈值设定为 4 时,系统 1 将认出在列“0”中为了擦除而标记的数据片段的数目已达到擦除阈值。所以,系统 1 将现在擦除在列“0”中的为了擦除而标记的所有数据片段,即在行“0”、“1”、“2”、和“4”中的数据片段。在这些数据片段已被擦除后,相应的存储位置再次可用于存储新的数据。在图 1c 中所示出的这个情形指示行“0”的 4 个数据片段已被擦除。当为了擦除而标记的数据片段正被擦除时,在第二存储器部件 3 中存储的映射信息不需要被更新。这个映射信息只涉及有效的数据片段并且所以不受擦除弃用的数据片段的影响。所以,在图 1c 中所示出的存储器系统 1 的配置中的第二存储器部件 3 中存储的映射信息与在图 1b 中所示出的配置中的相同。

[0022] 在图 1a — 1c 中所示出的存储器系统 1 的实施例中,第一存储器部件 2 的单元阵列的每列位于分离的井上,即第一存储器部件 2 包括 4 个井,其中该单元阵列的列中的每个位

于这些井中的一个井上。通过为存储位置的每列提供分离的井,它能够保证在没有擦除在其他列中的数据片段的情况下,能够擦除在该列的一列中为了擦除而标记的数据片段。在一个实施例中,大体通过施加第一电压到连接至该数据片段的存储位置的字线和施加第二电压到在里面布置存储位置的井来擦除数据片段。所以,在图 1b 中所示出的存储器系统 1 的配置中,通过施加第一电压到字线“0”、“1”、“2”和“4”和施加第二电压到列“0”的井来擦除在列“0”中为了擦除而标记的数据片段。

[0023] 为了在特别的行中擦除所有为了擦除而标记的数据片段,第一电压被施加到连接至这些数据片段的存储位置的字线和到在里面布置存储位置的井。所以,为了在行“4”中擦除所有为了擦除而标记的数据片段,例如,在图 1b 中所示出的存储器系统 1 的配置中,第一电压被施加到字线“4”并且第二电压被施加到列“0”、“1”和“2”的井。

[0024] 在图 1a — 1c 中所示出的系统 1 的第一存储器部件 2 的单元阵列包括 4 列和 1024 行。所以,如果擦除阈值设定为 4,该单元阵列在达到擦除阈值之前能够包括最多 12 个为了擦除而标记的数据片段。4 个列中的每列能够含有 3 个为了擦除而标记的数据片段 ( $4 \times 3 = 12$ )。在这个配置中,不会达到擦除阈值,除非这些数据片段的 4 个位于单个行中。在下一个数据片段为了擦除而被标记的情况下,为了擦除而标记的数据片段的数目在下一个数据片段的列中将达到 4。13 个数据片段在那时将为了擦除而标记。接着,这些数据片段的 4 个将被擦除。该单元阵列然后将含有 9 个为了擦除而标记的数据片段。所以,在本公开的这个示例实施例中,为了在第一存储器部件中存储一定的数据量,第一存储器部件的单元阵列必须含有大量的存储位置,总计为这个数据量加上至少 12 个附加的存储位置。

[0025] 如果在图 1a — 1c 中所示出的本公开的实施例中,擦除阈值设定为 8,则该单元阵列在达到擦除阈值之前能够包括最多 28 个为了擦除而标记的数据片段。4 个列中的每列能够含有 7 个为了擦除而标记的数据片段 ( $4 \times 7 = 28$ )。下一个为了擦除而被标记的数据片段将会使为了擦除而标记的数据片段的数目在这些列的一列中为 8。所以,在这种情况下,在该单元阵列中必须提供至少 28 个附加的存储位置。然而,如果擦除阈值设定为 8 而不是 4,能够进一步减少擦除操作的数目,因为在每个擦除操作中 8 个数据片段将同时被擦除,然而在擦除阈值设定为 4 的情况下只有 4 个数据片段将同时被擦除。

[0026] 根据本公开的另一个实施例,用于仿真 EEPROM 的系统被配置成对于非易失性存储器阵列的每个存储位置来监控为了写入或擦除而已访问存储位置的次数。有利地,当写入数据片段到在第一存储器部件中的存储位置时,该系统然后选择这样的存储位置:为了写入或擦除而已访问所述存储位置的次数不超过预设的阈值。在这种方式下,能够保证写入操作和擦除操作横跨单元阵列均匀地分布。因为,如上面所提到的,在非易失性存储器系统中,尤其在闪速存储器系统中,为了写入或擦除而只能够访问每个存储位置有限的次数,所以横跨存储器系统均匀地分布写入操作和擦除操作是尤其有利的。

[0027] 如果在图 1c 中所示出的存储器系统 1 中,新的数据片段“zz”需要被写入到第一存储器部件 2,基本上能够使用任何空的存储位置,即要么是已被先前擦除的存储位置(诸如存储位置“0/0”),要么是根本没被写入的存储位置(诸如存储位置“1023/0”)。在系统选择存储位置“0/0”以将数据片段“zz”写入到单元阵列之前,该系统检查为了写入或擦除而已访问存储位置“0/0”的次数。如果这个数目没有超过预设的阈值,该系统将继续写入数据片段“zz”到存储位置“0/0”。然而,如果这个数目确实超过预设的阈值,该系统将选择另



一个存储位置,例如存储位置“1023/0”。有利地,该预设的阈值在存储器系统 1 的操作期间逐渐增加。

[0028] 根据本公开的另一个实施例,系统 1 维持一个访问表,该访问表对于第一存储器部件的每个存储位置,记录为了写入或擦除而已访问存储位置的次数和指示该存储位置是否准备好接收数据的状态信息。在一个实施例中,当数据片段需要被写入到第一存储器部件时,该系统然后从访问表中选择具有写入-/擦除-访问的最低数目的存储位置。如果这个存储位置准备好接收数据,该数据片段将被写入到这个存储位置。如果存储位置没有准备好接收数据,该系统将选择具有写入-/擦除-访问的第二个最低数目的存储位置,并且将再次检查该存储位置是否准备好接收数据。这个过程将继续直到已发现准备好接收数据的存储位置。然后,该数据片段将会被写入到这个存储位置。

[0029] 在非易失性存储器系统中,大体通过施加电压到存储器系统的各种元件诸如存储器系统的字线、位线和井来读取、写入、或擦除数据。由于常规的非易失性存储器系统的布局,在读取操作、写入操作、或擦除操作期间,这些电压通常不但被施加到为了读取、写入、或擦除而访问的存储器单元,而且被施加到非易失性存储器系统的其他存储器单元。结果,非易失性存储器系统的存储器单元在读取操作、写入操作、和擦除操作期间接收电压应力。这个电压应力一般被称为“干扰”。存在几种类型的干扰,诸如“读取干扰”、“写入干扰”、或“擦除干扰”,它们还被称为“栅干扰”或“井干扰”。接收许多干扰的存储器单元可能会遭受损坏,即在存储器单元中存储的数据值可能被错误地改变。

[0030] 在本公开的一个实施例中,用于仿真 EEPROM 的系统被配置成对于在非易失性存储器阵列中存储的每个数据片段来监控该数据片段已接收的干扰的数目。在一个实施例中,该系统被配置成监控几种类型的干扰。可替代地,该系统被配置成监控干扰的单一类型。在一个实施例中,这个干扰的单一类型是对于依照本公开的存储器系统的特别的体系结构和布局来说具有危害数据完整性的最高潜力的干扰的类型。在本公开的一个实施例中,该系统被配置成监控井干扰。

[0031] 通过监控在非易失性存储器阵列中存储的数据片段接收的干扰的数目,依照本公开的系统能够防止数据片段由于接收过多数目的干扰而遭受损坏。在一个实施例中,该系统被配置成当数据片段已接收的干扰的数目超过干扰阈值时移动数据片段到在非易失性存储器阵列中新的存储位置。这将保证数据片段在它接收过多数目的干扰之前被移动到新的存储位置并且所以有助于维持数据完整性。

[0032] 图 2 示出依照本公开的另一个实施例的存储器系统的第一存储器部件 2 的示例性表示。在这个实施例中第一存储器部件 2 非常类似于图 1a - 1c 的第一存储器部件,并且再次包括非易失性存储器阵列,该非易失性存储器阵列包括多个在行和列中布置的存储位置。该非易失性存储器阵列包括 4 列和 1024 行的存储位置,其中每个存储位置适合于存储 32 字节的数据。此外,第一存储器部件 2 包括一列存储位置 6,该存储位置 6 适合于为非易失性存储器阵列的每行存储多个状态位。在一个实施例中,这些状态位被用来存储计数器,诸如干扰计数器,尤其是井干扰计数器。此外,该状态位可以被用来记录指示该集群是否已被正确擦除的状态信息。

[0033] 而且,在图 2 中所示出的第一存储器部件 2 包括多个字线驱动器 7。字线驱动器被耦接到非易失性存储器阵列的每个字线。因为该非易失性存储器阵列包括 1024 行的存储

位置,所以该第一存储器部件 2 包括 1024 个字线驱动器。在一个实施例中,该字线驱动器 7 位于第一存储器部件 2 的最左边的位置。

[0034] 图 3 示出依照本公开的另一个实施例的存储器系统的第一存储器部件 2 的示例性表示。在这个实施例中,该非易失性存储器阵列包括 8 列和 512 行的存储位置,其中每个存储位置适合于存储 32 字节的数据,即图 3 的非易失性存储器阵列的总容量与图 2 的非易失性存储器阵列的总容量相同。该字线驱动器 7 再次位于第一存储器部件 2 的最左边的位置。在非易失性存储器系统中,字线和位线的最佳长度大体依赖于各种参数,诸如存储器单元的宽度和长度或字线和位线的电阻或电容。图 2 和 3 图示依照本公开的存储器系统的布局能够灵活地适合于存储器阵列的物理尺寸。

[0035] 图 4 示出依照本公开的另一个实施例的存储器系统的第一存储器部件 2 的示例性表示。在这个实施例中,该非易失性存储器阵列再次包括 8 列和 512 行的存储位置,其中每个存储位置适合于存储 32 字节的数据。然而,该字线驱动器 7 居中地位于非易失性存储器阵列中。尤其是,该字线驱动器 7 被安放在列“3”和列“4”之间。结果,每个字线分成两个片段,其中单个字线驱动器耦接到两个片段。通过将每个字线分成两个片段,该字线的长度就被减少。所以,该字线驱动器 7 的这种放置改进了涉及通过字线来施加电压的操作的时间效率。

[0036] 图 5 示出依照本公开的另一个实施例的存储器系统的第一存储器部件 2 和与这个存储器系统关联的数据结构的示例性表示。该第一存储器部件 2 类似于图 1a—1c 和 2 的第一存储器部件,并且包括多个布置在行和列的阵列中的存储位置。该存储器阵列包括 4 列,其中每列被布置在分离的井 9 上。所以,第一存储器部件 2 包括 4 个井 9。而且,第一存储器部件 2 包括多个字线 8。每个字线 8 被耦接到存储位置的一行。

[0037] 每个存储位置被配置成存储 32 字节的用户数据。图 5 的下面部分示出依照本公开的实施例用于存储这些 32 字节的用户数据的一个示例数据结构。该 32 字节的用户数据被划分成 4 个数据片段:第一数据片段 10a、第二数据片段 10b、第三数据片段 10c、和第四数据片段 10d。这些数据片段中的每个包括 64 位(4×16 位)的用户数据,并且通过包括 24 位的关联的 ECC 11a—11d 来保护这些数据片段中的每个。所以,四个数据片段 10a—10d 中的每个数据片段与关联的 ECC 11a—11d 一起要求 88 位的存储空间。提供另一个 88 位的块来存储标记位 12 (8 位)、状态位 13 (3×16 位=48 位)、和另一个 ECC 14 (24 位)。这个块中的 8 位没有被使用。用来存储 32 字节的用户数据的所得到的数据结构包括 440 位(5×88 位=55 字节)。通过在 55 字节的数据结构中存储 32 字节的用户数据,依照本公开能够非常高效地将用户数据存储存储在存储器系统中。该存储效率比使用例如 88 字节来存储 32 字节的用户数据的典型常规系统的效率好得多。

[0038] 在本公开的一个实施例中,用于仿真 EEPROM 的系统被配置成:当它接收编程数据到存储器系统的请求时中断正在进行的擦除操作,并且在该编程操作已被完成之后自动恢复该中断的擦除操作。这被示出在图 6 的 3 个时序图中的一个示例中。第一个图示出该系统执行短的编程操作 15,所述短的编程操作 15 后面是擦除操作 16。尽管在图 6 中的时间规范只是象征性的,该图示出擦除操作 16 比编程操作 15 花费更长得多。

[0039] 在第二个图中,执行两个编程操作。该系统再次以第一个编程操作 15a 开始,所述第一个编程操作 15a 后面是擦除操作 16。在该擦除操作 16 已被完成之前,该系统接收到

执行第二个编程操作 15b 的请求。所以,中断正在进行的擦除操作 16 并且执行第二个编程操作 15b。在该第二个编程操作 15b 已被完成之后,该擦除操作 16 就被恢复并且最后被完成。因为该擦除操作 16 被中断,它会比在第一个图的场景中更晚完成。

[0040] 在第三个图中,执行四个编程操作。该系统再次以第一个编程操作 15a 开始,所述第一个编程操作 15 后面是擦除操作 16。在该擦除操作 16 已被完成之前,该系统接收到执行第二个编程操作 15b 的请求。所以,中断正在进行的擦除操作 16 并且执行第二个编程操作 15b。在该第二个编程操作 15b 已被完成之后,该擦除操作 16 就被恢复。此后,该擦除操作 16 再次被中断来执行第三个编程操作 15c,然后再次被恢复,再次被中断来执行第四个编程操作 15d,并且再次被恢复以最终被完成。因为该擦除操作 16 被中断三次,它会在第一个或第二个图的场景中更晚完成。

[0041] 在本公开的一个实施例中,用于仿真 EEPROM 的系统被配置成提供至少 3 种操作模式。在可能被称为“标准模式”的第一种操作模式下,该系统允许数据片段被读取或更新。在这种操作模式下,在读取执行之前可能需要某种等待时间,因为已经运行的擦除需要被暂停。在可能被称为“快速读取模式”的第二种操作模式下,该系统提供具有尤其短的访问时间的特殊读取功能。通过在第二种操作模式下抑制更新、编程、或擦除操作来实现这个短的读取访问时间。所以,在第二种操作模式下不能够执行更新、编程、或擦除操作。因为在第二种操作模式下不允许编程或擦除操作,所以在能够执行读取操作之前不必要暂停编程或擦除。这个操作模式对于读取突发来说尤其合适。在可能被称为“快速更新模式”的第三种操作模式下,该系统允许数据片段被读取或被更新。在这种操作模式下,旧的数据片段不被擦除。所以,在这种操作模式下,能够实现非常快速的编程速率,因为新的数据片段在没有擦除旧的数据片段的情况下直接被写入到存储器系统。依赖于存储器系统的体系结构,这种操作模式可以被限制到某个小的缓冲区大小例如 8KB。在图 1a - 1c 的存储器系统中,在这种操作模式下,新的数据片段优选地被写入到标识为“+缓冲区”的存储位置,即第一存储器部件 2 的单元阵列的最后行。

[0042] 图 7 示出在依照本公开的实施例的用于仿真电可擦除可编程只读存储器的方法中执行的步骤。该方法包括以下动作:对于非易失性存储器阵列的每行和每列:监控在该行或该列中为了擦除而标记的数据片段的数目(动作 30);对于每行和每列:确定在该行或该列中为了擦除而标记的数据片段的数目是否超过擦除阈值(动作 35)。如果对动作 35 的响应为否,重复动作 30;如果在 35 处是,则擦除在该行或该列中为了擦除而标记的数据片段(动作 40)。然后该方法返回到动作 30。通过执行这些动作,能够减少在 EEPROM 仿真期间执行的擦除操作的数目,因为在每次擦除操作中同时擦除几个数据片段。结果,增强了 EEPROM 仿真的性能。

[0043] 尽管在这里已图示和描述具体实施例,对本领域那些普通技术人员来说将认识到的是,在没有背离本公开范围的情况下,可以用多种替代的和/或等价的实现取代示出和描述的具体实施例。本申请旨在覆盖在这里讨论的具体实施例的任何改编或变型。所以,旨在本公开仅由权利要求书及其等价物来限制。

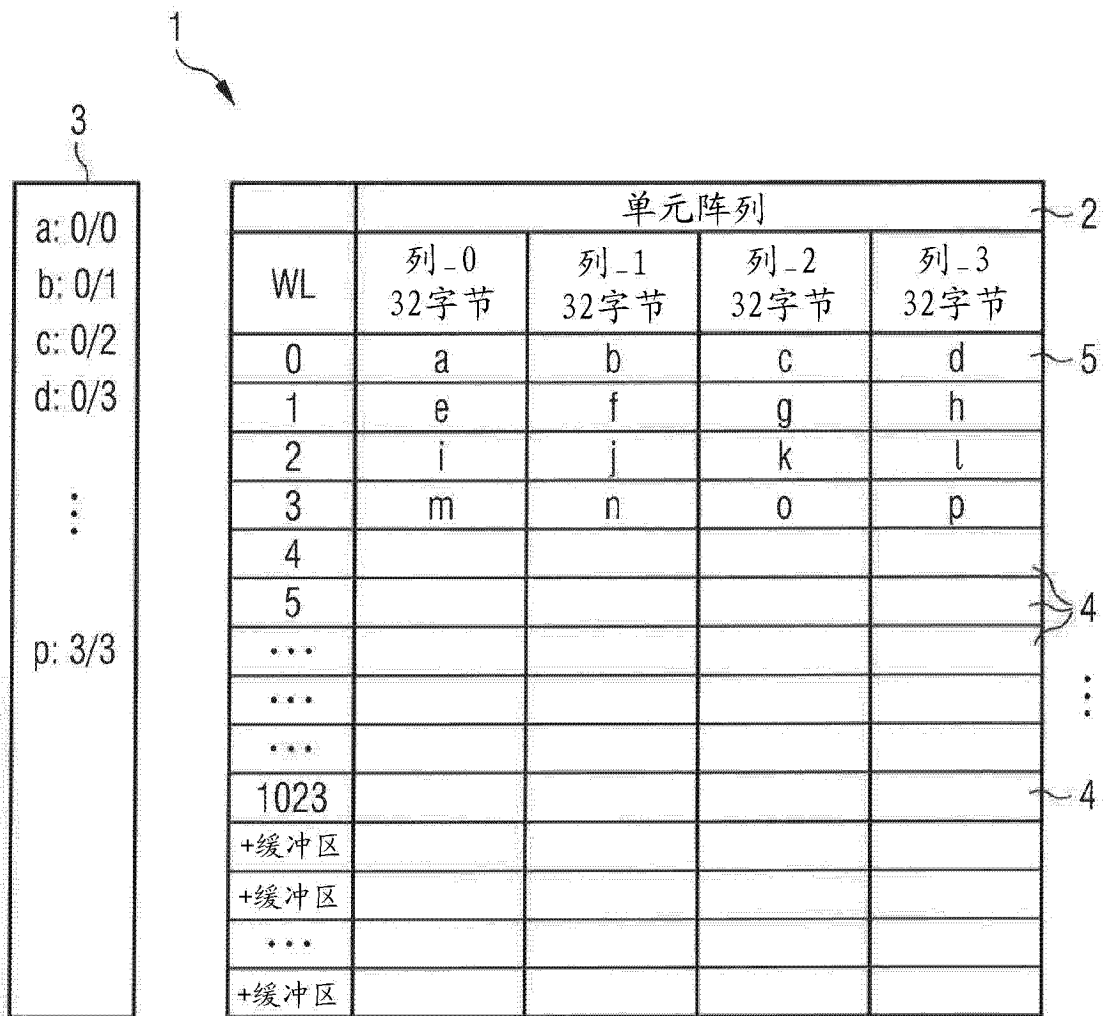


图 1A

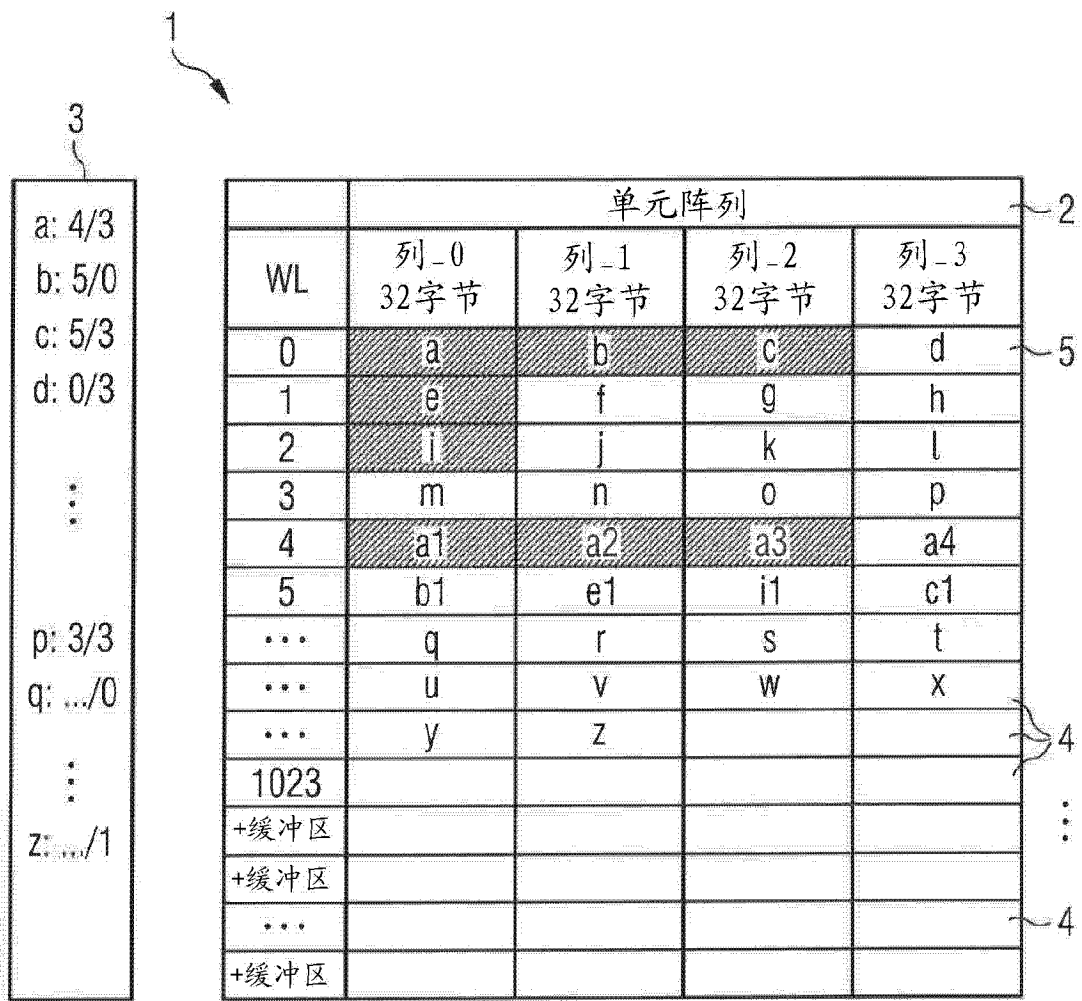


图 1B

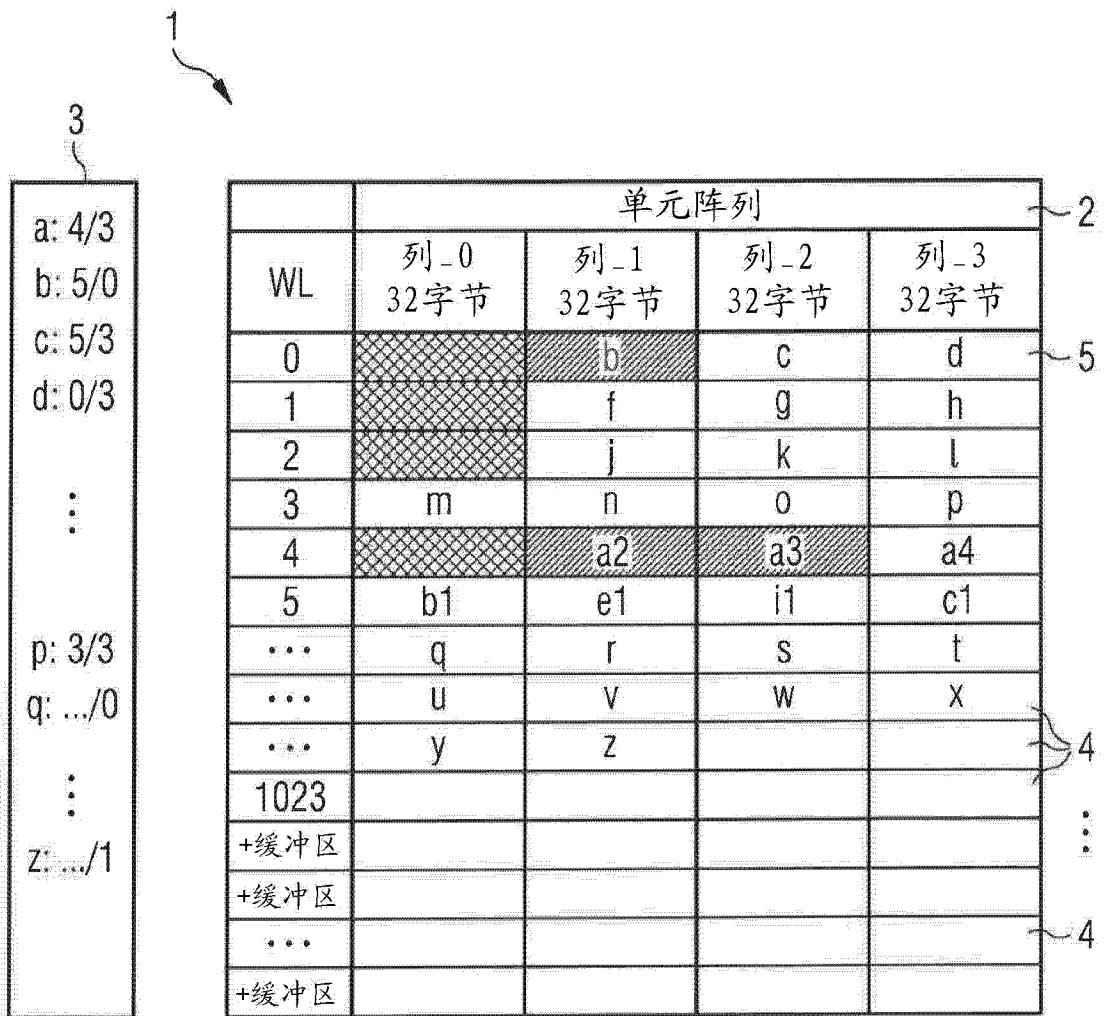


图 1C

The diagram shows a memory array structure. A bracket labeled '2' spans the top of the array. A bracket labeled '7' is positioned above the first column. A bracket labeled '6' is positioned above the second column. A bracket labeled '5' is on the right side of the first three rows. A bracket labeled '4' is on the right side of rows 4, 5, and 6. The array is a grid with 1024 rows and 6 columns.

WL 驱动器	状态位	列-0 32字节	列-1 32字节	列-2 32字节	列-3 32字节
0		D_00	D_01	D_02	D_03
1		D_10	D_11	D_12	D_13
2		D_20	D_21	D_22	D_23
3					
4					
5					
...					
...					
...					
1023					

图 2

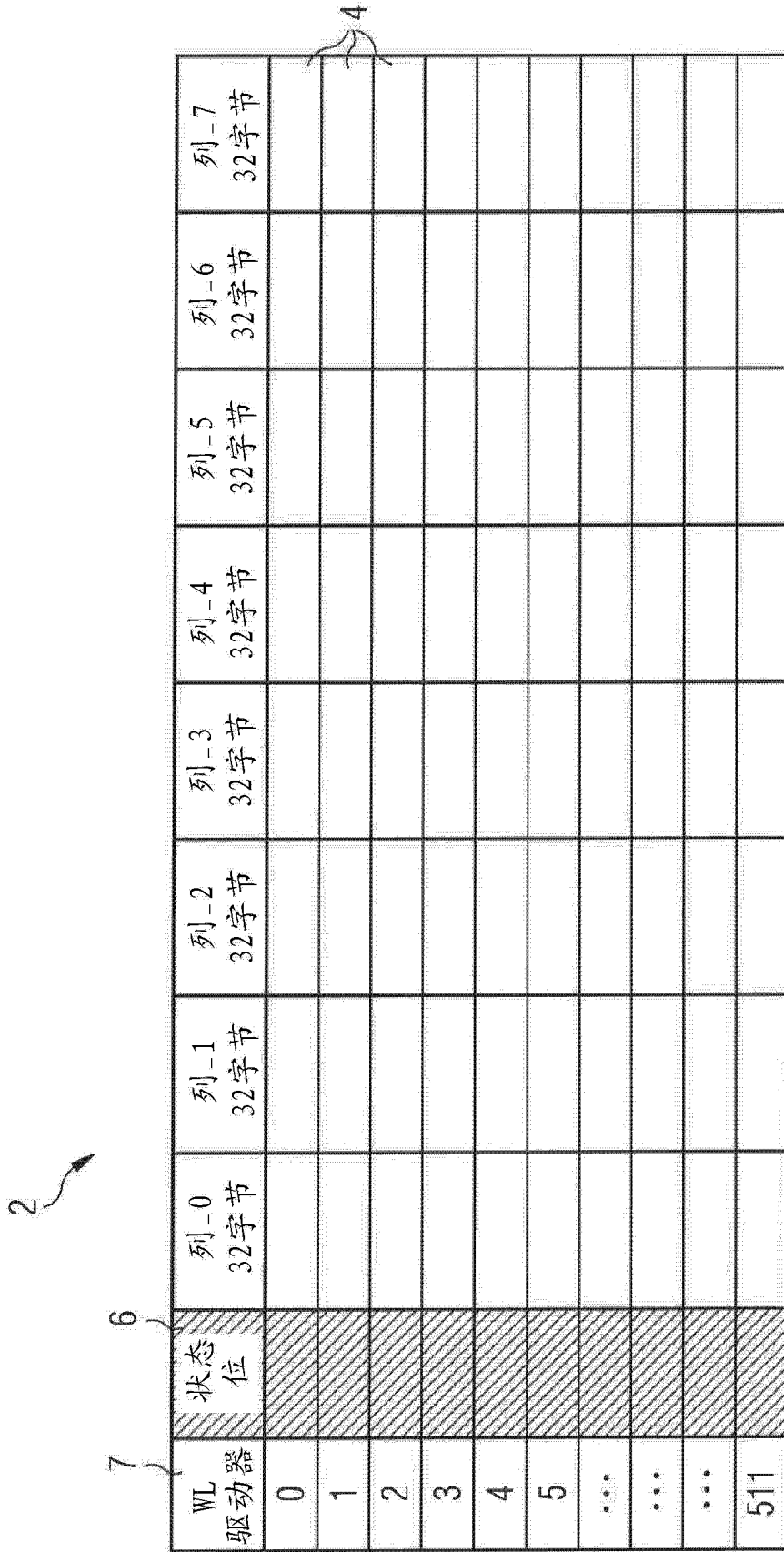


图 3



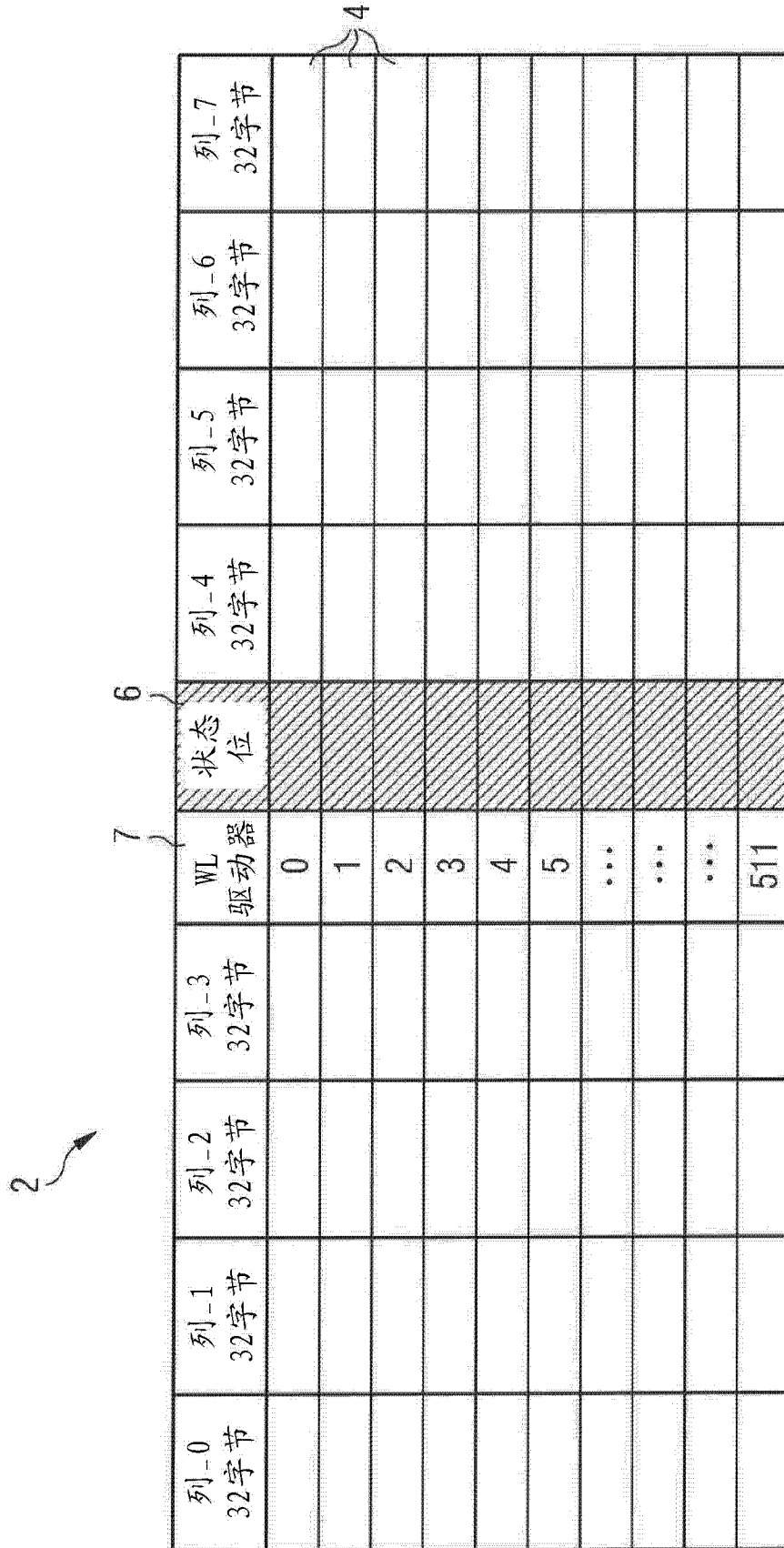


图 4



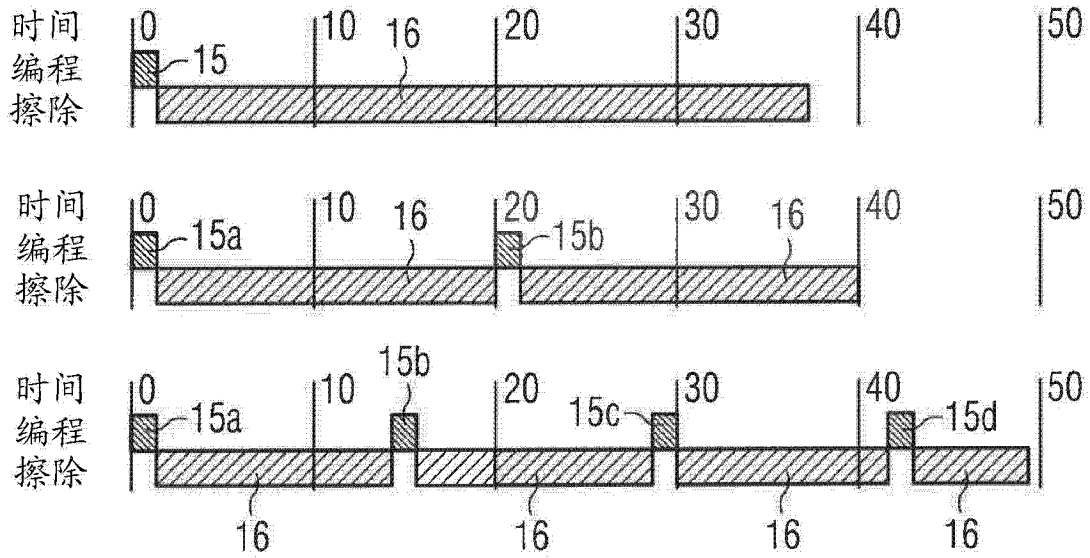


图 6

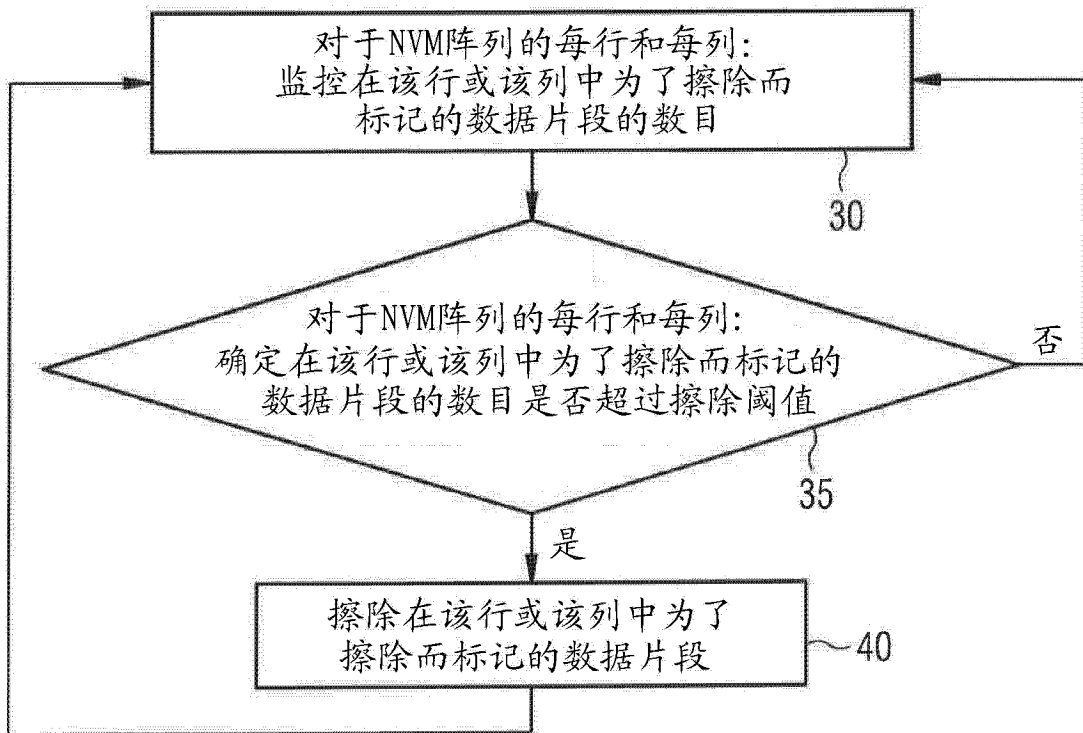


图 7