



(12) 发明专利申请

(10) 申请公布号 CN 103155044 A

(43) 申请公布日 2013.06.12

(21) 申请号 201180035507.5

代理人 王勇

(22) 申请日 2011.07.20

(51) Int. Cl.

(30) 优先权数据

G11C 16/10(2006.01)

61/366375 2010.07.21 US

G11C 16/24(2006.01)

(85) PCT申请进入国家阶段日

2013.01.18

(86) PCT申请的申请数据

PCT/CA2011/050442 2011.07.20

(87) PCT申请的公布数据

W02012/009812 EN 2012.01.26

(71) 申请人 莫塞德技术公司

地址 加拿大安大略省

(72) 发明人 金镇祺

(74) 专利代理机构 北京泛华伟业知识产权代理

有限公司 11280

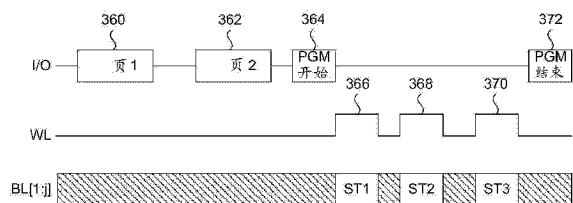
权利要求书3页 说明书23页 附图22页

(54) 发明名称

用于闪存存储器的多页编程方案

(57) 摘要

一种电路和方法,用于在单个编程操作周期中将多个比特的数据编程到闪存存储器单元。要编程到闪存存储阵列的一个物理页内的多个数据页被存储在存储器装置上的页缓冲器或其他存储部件中。在不同的时间间隔使用预定编程配置来驱动连接到要编程的单元的所选择的字线,其中,每一个预定编程配置被配置来用于将擦除阈值电压偏移到与特定的逻辑状态相对应的特定阈值电压。多页位线控制器响应于与那个相应的位线相关联的属于每个数据页的比特的特定逻辑状态的组合而在时间间隔的每一个期间偏置每个位线以启用或禁止编程。



1. 一种用于 NAND 闪存存储器装置的多页编程方法,所述方法包括:
在所述 NAND 闪存存储器装置中存储 M 个数据页,其中, M 是大于 1 的整数值;以及
在所述 NAND 闪存存储器装置中启动多页编程操作,以在所述 NAND 闪存存储器装置的存储器单元中存储多达 2^M 个状态。
2. 根据权利要求 1 所述的方法,还包括:检查所述多页编程操作的完成状态。
3. 根据权利要求 2 所述的方法,还包括:当所述完成状态对应于完成的多页编程操作时,判断所述多页编程操作是否成功。
4. 根据权利要求 1 所述的方法,其中,存储包括:迭代地将所述 M 个数据页中的每一个加载到所述 NAND 闪存存储器装置的相应的 M 个页缓冲器中。
5. 根据权利要求 4 所述的方法,其中,加载所述 M 个数据页中的每一个包括在所述 NAND 闪存存储器装置处接收后面跟随输入数据的数据加载命令。
6. 根据权利要求 5 所述的方法,其中,所述输入数据包括所述数据和地址信息。
7. 根据权利要求 5 所述的方法,其中,所述数据加载命令是第一数据加载命令,并且在接收到与所述第一数据加载命令相对应的所述输入数据后接收第二数据加载命令。
8. 根据权利要求 7 所述的方法,其中,在接收到与所述第一数据加载命令相对应的所述输入数据后,并且在接收到所述第二数据加载命令前,接收数据结束命令。
9. 根据权利要求 7 所述的方法,其中,在接收到与所述 M 个数据页的最后页相对应的输入数据后,接收用于所述 M 个数据页的最后页的数据结束命令。
10. 根据权利要求 9 所述的方法,其中,所述数据结束命令包括多页编程命令。
11. 根据权利要求 1 所述的方法,其中,所述多页编程操作包括:执行 2^M-1 次编程迭代以编程多达 2^M-1 个状态,并且所述 2^M 个状态之一是擦除状态。
12. 根据权利要求 11 所述的方法,其中,每一次编程迭代包括:响应于所述 M 个数据页的与每一个位线相对应的比特的组合而将所述位线的每一个偏置以启用或禁止编程。
13. 根据权利要求 12 所述的方法,其中,每一次编程迭代包括:在偏置所述位线以启用或禁止编程时,使用对于每一次编程迭代特定的编程配置来驱动所选择的字线。
14. 一种闪存存储器装置,包括:
存储阵列,其具有连接到字线并且耦合到位线的闪存存储器单元;
位线访问电路,用于存储 M 个数据页,并且响应于来自所述 M 个数据页的与每一个位线相对应的比特的组合将所述位线偏置,以对于 2^M-1 次编程迭代的每一次启用或禁止编程,其中, M 是至少为 2 的整数;以及
行电路,用于在偏置所述位线以启用或禁止编程时,使用用于所述 2^M-1 次编程迭代的每一次的编程配置来驱动所选择的字线。
15. 根据权利要求 14 所述的闪存存储器装置,其中,所述位线访问电路包括 M 个数据缓冲器,其中每一个数据缓冲器用于存储所述 M 个数据页之一。
16. 根据权利要求 15 所述的闪存存储器装置,其中,所述位线访问电路包括位线偏置电路,该位线偏置电路用于响应于在所述 M 个数据缓冲器中存储的所述 M 个数据页的与每一个位线相对应的比特的组合来偏置所述位线的每一个以启用或禁止编程。
17. 根据权利要求 16 所述的闪存存储器装置,其中,所述 M 个数据缓冲器的每一个数据缓冲器包括数据存储电路,用于存储数据页的一个比特。

18. 根据权利要求 17 所述的闪速存储器装置,其中,所述 M 个数据缓冲器的每一个比特位置包括:

数据验证解码器,用于接收在所述 M 个数据缓冲器中存储的所述 M 个数据页的与每一个位线相对应的比特,以及

反转电路,用于响应于所述数据解码器的所选择的输出来反转所述比特。

19. 一种闪速存储器装置,包括:

存储阵列,其具有连接到字线并且耦合到位线的闪速存储器单元,其中,每一个闪速存储器单元能够被编程来具有与擦除状态、第一状态、第二状态和第三状态之一相对应的阈值电压;

位线访问电路,用于偏置所述位线来以预定顺序编程所述第一状态、所述第二状态和所述第三状态,所述位线访问电路响应于来自两个数据页的比特的特定组合而偏置所述位线以对于所述第一状态、所述第二状态和所述第三状态中的每一个启用或禁止编程;以及

行电路,用于在偏置所述位线以分别编程所述第一状态、所述第二状态和所述第三状态时,使用与所述第一状态、所述第二状态和所述第三状态相对应的编程配置来驱动所选择的字线。

20. 一种用于对闪速存储器装置进行编程的方法,包括:

向所述闪速存储器装置的页缓冲器中加载至少两个数据页;

响应于所述至少两个数据页的数据位的逻辑状态,将所述闪速存储器装置的位线偏置到对禁止编程和启用编程之一有效的电压电平,其中所述编程用于对耦合到所述位线的闪速存储器单元编程不同的阈值电压;以及

将所述闪速存储器装置的字线驱动一个时间段,以用于并行地对连接到所述字线的所述闪速存储器单元编程所述不同的阈值电压。

21. 一种用于并行地对多个闪速存储器单元编程不同状态的方法,包括:

接收针对所述闪速存储器单元的每一个的 M 比特的数据,其中, M 是至少为 2 的整数;

响应于所述 M 比特的数据来确定要在所述多个闪速存储器单元的每一个中存储的 2^M 个可能状态中的一个状态;

使用预定电压电平来偏置连接到所述多个闪速存储器单元的位线,其中,每一个预定电压电平对应于所述 2^M 个可能状态的每一个;以及

驱动所选择的字线以并行地在所述多个闪速存储器单元中编程多达 2^M 个不同状态。

22. 一种位线调制器电路,用于将多个数据页多态编程到闪速存储器单元的物理页,所述位线调制器电路包括:

编程数据解码器,用于接收 M 比特的数据,并且用于响应于所述 M 比特的数据的逻辑组合来提供解码后的选择信号,其中, M 是至少为 2 的整数,并且所述 M 比特的数据的每一个与所述数据页之一相关联;以及

位线偏置电路,用于响应于所述解码后的选择信号将位线偏置到 2^M 个电压电平之一。

23. 一种闪速存储器装置,包括:

存储阵列,其具有连接到字线并且耦合到位线的闪速存储器单元;

M 个页缓冲器,用于存储 M 个数据页,其中, M 是至少为 2 的整数;

位线调制器,其耦合到所述位线的每一个并且耦合到所述 M 个页缓冲器,所述位线调制器的每一个响应于来自所述 M 个数据页的每一个的数据位的组合向对应的位线施加 2M 个电压电平之一;以及

行电路,用于使用编程配置来驱动所选择的字线,以将所述 M 个数据页编程到一行闪存存储器单元。

用于闪速存储器的多页编程方案

技术领域

[0001] 本发明总体上涉及半导体装置,并且具体地涉及将多个比特的数据编程到单个闪速存储器单元。

背景技术

[0002] 闪速存储器是常用的一种非易失性存储器,其被广泛用作诸如数字照相机或便携数字音乐播放器之类的消费电子产品的大容量存储器。当前可获得的闪速存储器芯片的密度可以在大小上高达几个G字节(GB),这适合于在通用USB闪速驱动器中使用,因为一个闪速存储器芯片的尺寸很小。闪速存储器的另一种出现的应用是固态驱动器,用于替换在膝上型和台式计算机中使用的传统硬盘驱动器。

[0003] 图1是在本领域中已知的典型的闪速存储器装置的一般框图。闪速存储器装置10包括输入/输出接口电路、控制电路、存储器电路和存储阵列。闪速存储器装置10的输入/输出接口电路包括就绪/忙碌信号缓冲器12、控制信号缓冲器14和全局数据缓冲器16。就绪/忙碌信号缓冲器12是输出缓冲器,其经由相应的引脚或端口来驱动就绪/忙碌信号R/B#。在本示例中,控制信号缓冲器14是输入缓冲器,该输入缓冲器从相应的引脚或端口接收闪速存储器控制信号CE#、CLE、ALE、WE#、RE#和WP#。下文中,以“#”结尾的信号名称应当被理解为低态有效电平信号,其中,低态有效信号对应于例如“0”逻辑电平或VSS电压电平。相反,高态有效逻辑电平信号对应于例如“1”逻辑电平或VDD或VCC电压电平。下面是前述的控制信号的简短说明。

[0004] • 命令锁存启用(CLE):CLE输入信号用于控制操作模式命令向内部命令寄存器内的加载。该命令在CLE为高时在WE#信号的上升沿从I/O端口被锁存到命令寄存器中。

[0005] • 地址锁存启用(ALE):ALE信号用于控制向内部地址寄存器内加载地址信息。地址信息在ALE为高时在WE#信号的上升沿从I/O端口被锁存到地址寄存器中。

[0006] • 芯片启用(CE#):在装置处于就绪状态期间,当CE#变高时,装置进入低功率待机模式。当装置处于忙碌状态(R/B#=L)时,诸如在编程或擦除或读取操作期间,CE#信号被忽略,并且即使CE#输入变高,装置也不进入待机模式。

[0007] • 写入启用(WE#):WE#信号用于控制从I/O端口获取数据。

[0008] • 读取启用(RE#):RE信号控制串行数据输出。在RE#的下降沿后可获得数据。也在这个下降沿上递增内部列地址计数器(地址=地址+1)。

[0009] • I/O端口(I/00至7):I/00至7引脚被用作用于向和从装置传送地址、命令和输入/输出数据的端口。

[0010] • 写入保护(WP#):WP#信号用于保护装置以防意外编程或擦除。当WP#低时复位内部调压器(高压发生器)。这个信号通常用于当输入信号无效时在通电/断电序列期间保护数据。

[0011] • 就绪/忙碌(R/B#):R/B#是开漏引脚,并且输出信号用于指示装置的操作条件。R/B#信号在编程、擦除和读取操作期间在忙碌状态中(R/B#=L),并且在操作结束后返回到

就绪状态(R/B#=H)。

[0012] 在图 1 的本示例中的全局数据缓冲器 16 是双向缓冲器,其在相应的输入/输出(I/O)引脚或端口上接收写入数据,并且提供读取数据。闪速存储器装置 10 被示出具有 8 个这样的端口 I/00 至 I/07,但是可以在替代的数据宽度配置中具有大于或小于 8 个的端口。这些 I/O 端口也用于接收地址和命令信息。

[0013] 闪速存储器装置 10 的控制电路包括控制器 18、命令寄存器 20、地址寄存器 22 和状态寄存器 24。控制器 18 控制在闪速存储器装置 10 中的其他电路的各种功能,其中,这样的功能包括读取、编程和编程验证。虽然未示出,但是控制器 18 可以包括命令解码器,用于响应于所接收到的命令来执行功能。寄存器存储由闪速存储器装置 10 接收或要提供到包括例如存储器控制器的外部主机系统的各种类型的信息。所述寄存器不意欲是穷举的,并且可以包括其他寄存器,诸如数据寄存器。

[0014] 由控制器控制的主要电路是存储器电路,该存储器电路包括行和列预解码器 26 和 28、行和列解码器 30 和 32、感测放大器和页缓冲块 34、以及高压发生器 36。闪速存储器装置 10 的存储阵列 38 由连接到位线的 NAND 单元串构成,其中,NAND 单元串的每一个存储器单元连接到字线。后面在图 3 中示出了 NAND 单元串的另外的细节。在编程操作中控制行预解码器 26、行解码器 30 和高压发生器 36,以将所选择的字线驱动到高压,该高压对于将所连接的存储器单元的阈值电压从默认擦除阈值电压偏移到期望的电压电平是有效的。可以注意到,通过将存储器单元的阈值电压偏移到默认值,可以在存储阵列 38 中使用高压以擦除存储器单元。高压和高压施加时间的不同组合可以用于为闪速存储器单元设置特定阈值电压。用于编程特定阈值电压的组合可以被称为编程配置(profile)。页缓冲器 34 存储要被编程到连接到所选择的字线的单元的数据页。通常,根据位线被偏置或被设置到的逻辑电平,连接到该位线的存储器单元经由所选择的字线被禁止编程或被启用编程。在页缓冲器中存储的数据位用于偏置位线。

[0015] 图 2 是用于单比特闪速存储器单元的阈值电压分布图,该单比特闪速存储器单元存储两个可能电平或状态之一:状态 0 或状态 1。其也被称为单电平单元(SLC)存储器。在图 2 中,状态 0 是存储阵列 38 的所有闪速存储器单元的默认擦除状态,其在这个示例中是负阈值电压。相反,状态 1 对应于正阈值电压。因此,当被擦除时,闪速存储器单元针对所存储的数据的单个比特存储“1”逻辑状态。然后,如果要在所选择的单元中存储的数据对应于“0”,则执行编程以将擦除阈值电压偏移到状态 1 的正电压,这如上所述是通过使用特定的编程配置来驱动所选择的连接到单元的字线而进行的。

[0016] 图 3 是示出图 1 的存储阵列 38 的一部分的电路示意图,并且具体地说示出了连接到两个位线 BL1 和 BLj 的 NAND 单元串。每一个 NAND 单元串具有串联耦合地布置并且彼此电耦合的闪速存储器单元 50。因此,字线 WL0 至 WLi 耦合到存储器单元串中的各闪速存储器单元 50 的栅极。补位数字“i”和“j”是整数值,用于分别指定最后的字线和位线。耦合到信号 SSL(串选择线)的串选择装置 52 选择性地将存储器单元串连接到位线(BL1 或 BLj),而耦合到信号 GSL(接地选择线)的接地选择装置 54 选择性地将存储器单元串连接到源极线,诸如 VSS。串选择装置 52 和接地选择装置 54 是 n 沟道晶体管。连接到相同的字线的在图 3 中所示的两个 NAND 单元串、SSL 和 GSL 是一个存储器块 56 的一部分。因此,多个存储器块可以与图 1 所示的存储器块并行地连接到相同的位线 BL1 至 BLj。

[0017] 如果闪速存储器单元 50 存储单个比特的数据,则连接到同一字线的所有单元 50 存储一个数据页。在图 3 中,例如,连接到 WL0 的单元被统称为物理页 0,而连接到 WLj 的单元被统称为物理页 i,其中,每个物理页存储一个数据页。

[0018] 页缓冲器 58 耦合到各个位线 BL1 至 BLj,用于存储要被编程到闪速存储器单元的一个页内的一个数据页。页缓冲器 58 还包括感测电路,用于感测从闪速存储器单元的一个页读取的数据。在编程操作期间,数据寄存器执行编程验证操作,以保证数据已经被正确地编程到耦合到所选择的字线的闪速存储器单元内。因此,每行单元存储一个数据页。为了实现改善的密度,每一个闪速存储器单元可以存储至少两个比特的数据,并且一般被称为多比特单元(MBC)。在每一个单元存储至少两个比特的数据的情况下,存储阵列的存储密度相对于具有仅存储一个比特的数据的单元的同存储阵列至少被加倍。

[0019] 图 4 是两比特闪速存储器单元的阈值电压分布图。其也被称为每单元多电平(MLC)存储器。图 4 示出四个可能状态,状态 0、状态 1、状态 2 和状态 3,它们可以被两比特闪速存储器单元存储。如图 4 中通过示例示出,每一个状态对应于两个比特的数据的组合。状态 0 可以表示二进制值“11”,状态 1 可以表示二进制值“10”,状态 2 可以表示二进制值“00”,以及状态 3 可以表示二进制值“01”。在图 4 中所示的二进制值是示例组合,并且,用于每一个状态的二进制值组合的分配可以与当前在图 4 中所示的那些不同。因此,连接到同一字线的一行闪速存储器单元现在可以存储两个数据页,而不是对于单比特闪速存储器单元的单个数据页。现在描述与被称为物理页的连接到公共字线的单元相关联的、被称为逻辑页的这两个数据页的逻辑组织。

[0020] 图 5 是示出在五行两比特闪速存储器单元中存储的下数据页和上数据页的存储的图示。在图 5 中,闪速存储器单元 70 和 72 是第一 NAND 单元串的一部分,闪速存储器单元 74 和 76 是第二 NAND 单元串的一部分,闪速存储器单元 78 和 80 是最后 NAND 单元串的一部分,最后 NAND 单元串用于表示在存储器块中的最后的 NAND 单元串。根据块的配置,在第二 NAND 单元串和最后 NAND 单元串之间可以有多个中间 NAND 单元串。闪速存储器单元的每一个具有在图 4 中所示的四个可能逻辑状态之一,其具有对应的两比特二进制值分配。在图 5 的示例中,单元 70 存储状态 3,单元 72 存储状态 1,单元 74 存储状态 1,单元 76 存储状态 3,单元 78 存储状态 2,并且单元 80 存储状态 0。在每一个单元中示出二进制值,其中,上比特对应于在图 4 的对应的状态中的最左边的比特,并且下比特对应于最右边的比特。连接到字线 WL0 的闪速存储器单元 72、76 和 80 对应于以下物理页:闪速存储器单元 72、76 和 80 的下比特对应于逻辑页 0,而上比特对应于逻辑页 1。类似地,连接到字线 WL_i 的闪速存储器单元 70、74 和 78 对应于另一个物理页:下比特对应于逻辑页 k-1,而上比特对应于最后的逻辑页 k。因此,可以看出每一个物理页如何存储两个数据页。

[0021] 通过扩展,如果每一个闪速存储器单元可以存储 3 比特的数据,则每一个单元可以存储 $2^3=8$ 个状态,这意味着每一个物理页可以存储多达 8 个数据页。如果可以编程所有的不同状态并在读取操作期间进行感测,则在存储密度上的增大而不需要增大物理的存储阵列大小提供了用于大容量存储应用的显著优点。不幸的是,用于对每一个物理页编程两个数据页(每单元两个比特)所需的时间相对于对每一个物理页编程单个数据页所需的时间显著增加。如果使用每一个单元存储三个或更多的比特,则编程时间进一步增加。

[0022] 虽然偶尔地将几个小数据文件复制到每个单元能够存储多个比特的 USB 驱动器

所需的时间可能看起来对于用户来说没有不方便,但是大量的小数据文件的频繁写入将显著地需要更长的时间。较小的数据文件的频繁写入在固态硬盘驱动器中是常见的,这可能对它被用在的计算机系统施加性能限制。诸如音乐和视频文件的较大的数据文件向每个单元能够存储多个比特的闪速存储器的存储将需要很显著的持续时间来编程。

[0023] 因为闪速存储器广泛地用在诸如 Sod 和闪速高速缓存的计算应用中,所以期望更高性能的闪速存储器。应当注意,可以使用较高的 I/O 带宽、较高的读取和写入吞吐量以及较高的操作灵活性中的一个或多个来获得高性能操作。而且,用于计算应用的闪速存储系统需要较高的存储器容量,同时不使得性能变差。

[0024] 因此期望减少用于编程存储两个或多个比特的数据的单元所需的时间,由此改善它被用在的系统的整体性能。

发明内容

[0025] 根据第一方面,提供了一种用于 NAND 闪速存储器装置的多页编程方法。所述方法包括:在所述 NAND 闪速存储器装置中存储 M 个数据页,其中, M 是大于 1 的整数;以及,在所述 NAND 闪速存储器装置中启动多页编程操作,以在所述 NAND 闪速存储器装置的存储器单元中存储多达 2^M 个状态。根据本方面的一个实施例,所述方法还可以包括:检查所述多页编程操作的完成状态,并且当所述完成状态对应于完成的多页编程操作时判断所述多页编程操作是否成功。根据本方面的下面一个实施例,存储可以包括:迭代地将所述 M 个数据页的每一个加载到所述 NAND 闪速存储器装置的相应的 M 个页缓冲器中,其中,加载所述 M 个数据页的每一个可以包括在所述 NAND 闪速存储器装置处接收后面跟随输入数据的数据加载命令。在本实施例中,所述输入数据包括数据和地址信息。而且,在本实施例中,所述数据加载命令是第一数据加载命令,并且在接收到与所述第一数据加载命令相对应的所述输入数据后接收第二数据加载命令。在这个实施例中,在接收到与所述第一数据加载命令相对应的所述输入数据后,并且在接收到所述第二数据加载命令前,接收数据结束命令。可替代地,在接收到与所述 M 个数据页的最后页相对应的输入数据后,接收用于所述 M 个数据页的最后页的数据结束命令,其中,所述数据结束命令包括多页编程命令。

[0026] 在另一个实施例中,所述多页编程操作包括:执行 2^M-1 次编程迭代以编程多达 2^M-1 个状态,并且所述 2^M 个状态之一是擦除状态。每一次编程迭代包括:响应于所述 M 个数据页的与每个位线相对应的比特的组合而将所述位线的每一个偏置以启用或禁止编程。而且,每一次编程迭代包括:在偏置所述位线以启用或禁止编程时,使用对于每一次编程迭代特定的编程配置来驱动所选择的字线。

[0027] 在第二方面,提供了一种闪速存储器装置。所述闪速存储器装置包括存储阵列、位线访问电路和行电路。所述存储阵列具有连接到字线并且耦合到位线的闪速存储器单元。所述位线访问电路存储 M 个数据页,并且响应于来自所述 M 个数据页的与每一个位线相对应的比特的组合将所述位线偏置以对于 2^M-1 次编程迭代的每一次启用或禁止编程,其中, M 是至少为 2 的整数。所述行电路在所述位线被偏置以启用或禁止编程时使用用于所述 2^M-1 次编程迭代的每一次的编程配置来驱动所选择的字线。根据第二方面的实施例,所述位线访问电路包括: M 个数据缓冲器,其中每一个数据缓冲器用于存储所述 M 个数据页之一;以及位线偏置电路,用于响应于在 M 个数据缓冲器中存储的所述 M 个数据页的与每一个位线

相对应的比特的组合来偏置所述位线的每一个以启用或禁止编程。在这个实施例中,所述 M 个数据缓冲器的每一个数据缓冲器包括数据存储电路,用于存储数据页的一个比特,并且所述 M 个数据缓冲器的每一个比特位置包括数据验证解码器和反转电路。所述数据验证解码器接收在所述 M 个数据缓冲器中存储的所述 M 个数据页的与每一个位线相对应的比特。所述反转电路响应于所述数据解码器的所选择的输出来反转所述比特。

[0028] 在第三方面,提供了一种闪速存储器装置。所述闪速存储器装置包括存储阵列、位线访问电路和行电路。所述存储阵列具有连接到字线并且耦合到位线的闪速存储器单元,其中,每一个闪速存储器单元能够被编程来具有与擦除状态、第一状态、第二状态和第三状态之一相对应的阈值电压。所述位线访问电路偏置所述位线来以预定顺序编程所述第一状态、所述第二状态和所述第三状态。所述位线访问电路响应于来自两个数据页的比特的特定组合而偏置所述位线以对于所述第一状态、所述第二状态和所述第三状态的每一个启用或禁止编程。所述行电路在偏置所述位线以分别编程所述第一状态、所述第二状态和所述第三状态时使用与所述第一状态、所述第二状态和所述第三状态相对应的编程配置来驱动所选择的字线。

[0029] 在第四方面,提供了一种用于对闪速存储器装置进行编程的方法。所述方法包括:向所述闪速存储器装置的页缓冲器内加载至少两个数据页;偏置所述闪速存储器装置的位线;以及,驱动所述闪速存储器装置的字线。响应于所述至少两个数据页的数据位的逻辑状态,将所述闪速存储器单元的所述位线偏置到对禁止编程和启用编程之一有效的电压电平,其中所述编程用于对耦合到所述位线的闪速存储器单元编程不同的阈值电压。将所述字线驱动一个时间段,以用于并行地对连接到所述字线的所述闪速存储器单元编程不同的阈值电压。

[0030] 在第五方面,提供了一种用于并行地对多个闪速存储器单元编程不同状态的方法。所述方法包括:针对所述闪速存储器单元的每一个接收 M 比特的数据,其中, M 是至少为 2 的整数;响应于所述 M 比特的数据来确定要在所述多个闪速存储器单元的每一个中存储的 2^M 个可能状态的一个状态;使用预定电压电平来偏置连接到所述多个闪速存储器单元的位线,其中,每一个预定电压电平对应于所述 2^M 个可能状态的每一个;以及,驱动所选择的字线以并行地在所述多个闪速存储器单元中编程多达 2^M 个不同状态。

[0031] 在第六方面,提供了一种位线调制器电路,用于将多个数据页多态编程到闪速存储器单元的物理页。所述位线调制器电路包括编程数据解码器和位线偏置电路。所述编程数据解码器接收 M 比特的数据,并且响应于所述 M 比特的数据的逻辑组合来提供解码后的选择信号,其中, M 是至少为 2 的整数,并且所述 M 比特的数据的每一个与所述数据页之一相关联。所述位线偏置电路响应于所述解码后的选择信号将位线偏置到 2^M 个电压电平之一。

[0032] 在第七方面,提供了一种闪速存储器装置。所述闪速存储器装置包括存储阵列、M 个页缓冲器、位线调制器和行电路。所述存储阵列具有连接到字线并且耦合到位线的闪速存储器单元。所述 M 个页缓冲器存储 M 个数据页,其中, M 是至少为 2 的整数。所述位线调制器耦合到所述位线的每一个并且耦合到所述 M 个页缓冲器。所述位线调制器的每一个响应于来自所述 M 个数据页的每一个的数据位的组合向对应的位线施加 2^M 个电压电平之一。所述行电路使用编程配置来驱动所选择的字线,以将所述 M 个数据页编程到一行闪速存储器单元。

[0033] 在结合附图阅读下面的本发明的特定实施例的说明后,本发明的其他方面和特征对于本领域内的普通技术人员将变得清楚。

附图说明

[0034] 现在参考附图仅通过示例描述本发明的实施例,在附图中:

[0035] 图 1 是 NAND 闪速存储器装置的功能框图;

[0036] 图 2 是用于单比特闪速存储器单元的阈值电压分布图;

[0037] 图 3 是示出 NAND 闪速存储阵列的 NAND 单元串的电路示意图;

[0038] 图 4 是用于两比特闪速存储器单元的阈值电压分布图;

[0039] 图 5 是示出在两比特闪速存储器单元中存储的下数据页和上数据页的存储的图示;

[0040] 图 6A 和 6B 是示出用于在 NAND 闪速存储器中使用两个编程操作来编程每单元两个比特的数据的序列的图示;

[0041] 图 7 示出用于在 NAND 闪速存储器中使用两个编程操作来编程每单元两个比特的数据的替代序列;

[0042] 图 8 是根据本实施例的多页可编程 NAND 闪速存储器的框图;

[0043] 图 9 是根据本实施例的用于操作闪速存储器装置以用于多页编程操作的方法的流程图;

[0044] 图 10 是根据本实施例的、用于操作闪速存储器装置以用于两页多页编程操作的方法的流程图;

[0045] 图 11 是示出根据本实施例的闪速存储器装置的示例两页多页编程操作的序列图;

[0046] 图 12A 是示出根据本实施例的、用于闪速存储器装置的示例命令和数据输入序列的序列图;

[0047] 图 12B 是示出根据本实施例的、用于闪速存储器装置的替代示例命令和数据输入序列的序列图;

[0048] 图 13 是根据本实施例的、用于在单个编程操作周期中对 MLC 闪速存储器单元编程所有逻辑状态的方法的流程图;

[0049] 图 14A 是示出根据本实施例的、用于对连接到 4 个不同位线的闪速存储器单元编程逻辑状态的多页编程操作的序列图;

[0050] 图 14B 是示出在图 14A 中被编程的 4 个闪速存储器单元的阈值电压的阈值电压分布图;

[0051] 图 15A 是示出根据本实施例的、用于对连接到 4 个不同位线的闪速单元编程逻辑状态的替代多页编程操作的序列图;

[0052] 图 15B 是示出在图 15A 中被编程的 4 个闪速单元的阈值电压的阈值电压分布图;

[0053] 图 16 是根据本实施例的、在图 8 中所示的位线访问电路的框图;

[0054] 图 17 是示出根据本实施例的、图 16 的数据解码器的细节的示意图;

[0055] 图 18A 是示出根据本实施例的、闪速存储器装置的两页并行多页编程操作的序列图;

- [0056] 图 18B 是示出在图 18A 中被编程的闪存存储器单元的示例阈值电压的阈值电压分布图；
- [0057] 图 19A 是示出根据本实施例的、闪存存储器装置的替代的两页并行多页编程操作的序列图；
- [0058] 图 19B 是示出在图 19A 中被编程的闪存存储器单元的示例阈值电压的阈值电压分布图；
- [0059] 图 20A 是示出根据本实施例的、闪存存储器装置的替代的两页并行多页编程操作的序列图；
- [0060] 图 20B 是示出在图 20A 中被编程的闪存存储器单元的示例阈值电压的阈值电压分布图；
- [0061] 图 21 是示出用于闪存存储器单元的编程禁止方法的流程图；
- [0062] 图 22A 是 NAND 单元串的截面图；
- [0063] 图 22B 是示出图 21 的编程禁止方法的示例的序列图；
- [0064] 图 23 是示出根据本实施例的、用于并行多态编程的位线的示例电压偏置的电路示意图；
- [0065] 图 24 是根据本实施例的、用于并行多态编程的位线访问电路的框图；
- [0066] 图 25 是示出根据本实施例的、在图 24 中所示的位线调制器的功能块的示意图；
- [0067] 图 26 是根据本实施例的、在图 25 中所示的编程数据解码器的电路示意实施例；
- [0068] 图 27 是在图 25 中所示的位线偏置电路的电路示意实施例；
- [0069] 图 28 是用于图 18A 的多态编程实施例的控制电路的电路示意实施例；以及
- [0070] 图 29 是根据本实施例的、用于概述用于并行多态编程的方法的流程图。

具体实施方式

[0071] 本发明总体上涉及用于在闪存存储器中在单个编程周期中将数据的多个逻辑页编程到单元的一个物理页的方法。具体地说，描述了用于在单个编程操作周期中将多个比特的数据编程到闪存存储器单元的电路和方法。在存储器装置上的页缓冲器或其他存储部件中存储要被编程到闪存存储阵列的一个物理页的多个数据页。在不同的时间间隔使用预定的编程配置来驱动连接到要编程的单元的所选择的字线，其中，每个预定编程配置被配置来用于将擦除阈值电压偏移到与特定的逻辑状态相对应的特定阈值电压。响应于属于与相应的位线相关联的每一个数据页的比特的特定逻辑状态的组合，位线在时间间隔的每一个期间被偏置到特定状态。

[0072] 在 NAND 闪存存储器中实现优选实施例，然而，本发明可以被应用于其他类型的闪存存储器，诸如 NOR 闪存存储器。

[0073] 在实施例的讨论之前，现在描述当前的 MLC 编程方案。通常的 MLC 闪存存储器架构是每单元 2 比特的闪存存储器。因此，描述当前的每单元 2 比特的编程方案。在每单元 2 比特的方案中，NAND 单元串内的每个字线被解码为作为第一页（下页）和第二页（上页）的两个逻辑页。参考图 5，逻辑页 0 可以是下页，而逻辑页 1 可以是上页。在这个示例中，首先编程下页，然后编程上页。在图 6A 和 6B 中示例性地示出了这个序列。

[0074] 图 6A 是示出与要在 2 比特闪存存储器单元中存储的数据的第一比特（1 或 0）相

对应的逻辑状态的阈值电压分布图。在编程之前,擦除存储块的所有单元。图 6A 示出擦除状态是状态 0,其在本示例中为负阈值电压。在第一编程操作周期中,根据要存储的比特的逻辑状态,对应的比特被编程为状态 1,或者它保持在状态 0。图 6B 是示出用于将也被称为上页的比特的数据的第二比特(1 或 0)编程到在图 6A 中编程的单元的可能序列的另一个阈值电压分布图。在随后的编程周期中,编程与上页相对应的该第二比特。该第二比特的逻辑状态确定是否和如何编程状态 2 和状态 3。如果例如状态 0 和状态 1 具有图 4 所示的二进制分配,并且要编程的第二比特(最左边的比特)是逻辑 1,则不需要编程状态 2 或状态 3,因为状态 0 和状态 1 两者具有被分配到逻辑 1 的上比特。另一方面,如果要编程的第二比特是逻辑 0,则需要编程状态 2 或状态 3。

[0075] 在编程本示例的状态 2 或状态 3 时,保留原始编程的下比特的逻辑状态。因此,如图 6B 中所示,将与状态 0 相对应的阈值电压偏移到与状态 3 相对应的阈值电压,或者将与状态 1 相对应的阈值电压偏移到与状态 2 相对应的阈值电压。因此,所示的逻辑状态状态 0 至状态 3 编码了上比特和下比特的不同逻辑组合。图 6A 和 6B 示出用于编程每单元两个比特的数据的双周期编程方案。图 7 示出阈值电压分布图,该图用于示出替代的双周期编程方案,以显示出阈值电压可以根据用于每一个逻辑状态的比特分配被偏移到不同的电平。在两个所示的示例中,需要两个独立的编程周期来编程每单元两个比特。每一个编程周期包含:接收具有地址信息和数据页的命令,并且使得存储器装置编程第一数据页,随后进行编程验证操作,以保证已经正确地编程了数据位。

[0076] 应当注意,在被配置来用于每个物理页两比特数据存储的传统 MLCNAND 闪速存储器中,用于下页和上页的总的页编程时间不同。通常,因为下面的原因,用于下页的页编程时间比用于上页的页编程时间快大约 30%。下页编程周期仅需要从状态 0 向状态 1 的单元编程,但是上页编程周期进一步包括下页数据的读取操作,以确定需要状态 2 或状态 3 编程的哪个。而且,用于状态 2 和状态 3 的编程配置比状态 1 的编程更精细并且需要更多的时间。因此,在 MLC 存储器中将多个页编程到一个物理页需要比在 SLC 存储器中将相同数量的多个页编程到它们自己的物理页长得多的时间量。

[0077] 本领域内的技术人员已知,使闪速存储器单元承受较长的编程时间可以使得单元的栅极氧化物变差,由此降低它们的可靠性。该问题由于工艺技术的不断缩小而变得更为复杂。对于较小的部件尺寸和较薄的栅极氧化物,由于较长的编程时间而导致的编程吞吐量变差成为严重的问题。另外,编程压力变为显著的问题,该显著的问题影响装置可靠性,并且缩短装置使用期限。

[0078] 为了克服当前多页编程方案的缺点,描述用于在单个编程操作周期中将多个比特的数据编程到闪速存储器单元的电路和方法。在存储器装置上的页缓冲器或其他存储部件中存储要被编程到闪速存储阵列的一个物理页内的多个数据页。在不同的时间间隔使用预定的编程配置驱动连接到要编程的单元的所选择的字线,其中,每个预定编程配置被配置来用于将擦除阈值电压偏移到与特定的逻辑状态相对应的特定阈值电压。位线被偏置到以下电压电平,该电压电平对于连接到在每一个时间间隔期间被驱动的所选择的字线的存储器单元的编程的启用或禁止有效。要施加到位线的特定电压电平取决于属于与相应位线相关联的各数据页的比特的特定逻辑状态的组合。实施例可以被应用到每单元 m 个比特的 MLC 存储器,其中, m 可以是任何整数。注意,对于每单元 m 个比特的 MLC 存储器的引用表示

单元的每一个物理页可以存储 m 个逻辑页。

[0079] 图 8 是根据本实施例的闪速存储器装置的框图。图 8 限于示出用于现在所描述的多页编程实施例的电路块。本领域内的技术人员应当明白,省略的电路块可以包括之前在图 1 中示出的那些。图 8 的存储器装置存储多个数据页,并且在单个编程周期中将所述页编程到所选择的物理页。

[0080] 存储器装置 100 包括存储阵列 102、行电路 104、高压发生器 106、控制电路 108 和位线访问电路 110。存储阵列 102 包括耦合到位线和字线的、使用 NAND 单元串配置或 NOR 配置排列的闪速存储器单元。存储阵列 102 具有位线 BL[1] 至 BL[j],其中, BL[j] 是在存储阵列中的最后位线,并且 j 是整数。存储阵列 100 可以被组织为存储体(bank)或子阵列。行电路 104 包括地址解码电路,诸如用于选择字线的存储体、块和行地址解码器和用于在读取、编程和擦除操作期间向字线施加各个电压的驱动器。高压发生器 106 向行电路提供比外部提供的电源电压更大的一个或多个正和 / 或负电压。这些高压可以被提供到存储器装置 100 的其他电路。控制电路 108 负责执行闪速存储器装置 100 的各种算法,并且使用适当的定时来确立(assert)控制信号以控制闪速存储器装置 100 的电路。具体地说,根据本实施例,控制电路 108 包括被配置来协调行电路 104 和位线访问电路 110 的控制以用于多页编程的逻辑。

[0081] 根据本实施例,位线访问电路 110 将各个位线偏置以在不同的时间以预定顺序并且与使用特定编程配置驱动的字线相结合地编程连接到各个位线的存储器单元的所有可能状态。各个位线被偏置以响应于来自在存储器装置 100 中存储的数据页的对应的比特位置的比特的特定组合来在特定时间启用或禁止编程。这些数据页被存储在位线访问电路 110 中。位线访问电路 110 包括页缓冲器 112 和至少一个多页编程缓冲器 114。页缓冲器 112 可以是具有诸如感测放大器、列解码电路和编程验证逻辑电路之类的电路的任何公知的页缓冲器。替代地,数据寄存器可以用于存储要编程的数据页。页缓冲器 112 用于存储要编程的第一数据页。位线访问电路 110 包括多页编程逻辑电路,用于保证连接到各位线的存储器单元已经被编程为与特定逻辑状态相对应的最终阈值电压。

[0082] 在本示例中,存储器装置 100 被配置来用于每单元两个比特的数据存储,因此存在一个多页编程缓冲器 114,用于存储要编程的第二数据页。可以在替代实施例中包括另外的多页编程缓冲器 114,在该替代实施例中超过 2 个的数据页要被存储在存储器装置 100 的物理页中。缓冲器 114 还包括列解码逻辑电路,用于接收开始列地址,在该地址,要存储第二数据页的第一比特。页缓冲器 112 和 114 两者具有相同的大小,其中,页缓冲器 112 存储第一数据页的数据位 DATA1_[1] 至 DATA1_[j],其各自对应于相应的位线 BL[1] 至 BL[j],其中, j 是用于表示页缓冲器 112 的最后比特位置的整数。页缓冲器 114 存储第二数据页的数据位 DATA2_[1] 至 DATA2_[j],其各自对应于相应的位线 BL[1] 至 BL[j]。

[0083] 下面是根据本实施例的、用于控制被配置来用于多页编程的闪速存储器装置的方法的说明。图 9 是用于操作被配置来用于前述的多页编程操作的闪速存储器装置的方法的流程图,该多页编程操作可以被主机系统执行,该主机系统包括但是不限于存储器控制器或具有存储器控制器功能的计算机系统。可以根据当前所述的方法控制的示例闪速存储器装置可以具有在图 8 中所示的配置。

[0084] 图 9 的方法在 300 开始,其中,经由被闪速存储器装置所识别的适当的命令将多

达 m 个数据页迭代地加载到闪速存储器装置的页缓冲器内。例如参考图 8 的电路实施例，将第一数据页存储在页缓冲器 112 中，并且将第二页缓冲器存储在页缓冲器 114 中。在步骤 302，在要编程到一个物理页的所有数据页已经被加载到闪速存储器装置内之后，在闪速存储器装置中启动多页编程操作。可以经由被闪速存储器装置所识别的命令来进行多页编程操作的启动。现在，闪速存储器装置执行内部操作以将逻辑页编程到单个物理页。下面更详细地描述本实施例的存储器装置的内部多页编程操作。主机系统通过状态请求命令在步骤 304 轮询闪速存储器装置，以检查闪速存储器装置的编程状态。响应于状态请求命令，闪速存储器装置从其数据输出端口输出状态位，该状态位指示编程操作是否完成。在步骤 306，主机系统根据所返回的状态信号来判断闪速存储器装置的编程是否完成。如果编程完成，则方法进入步骤 308，否则，主机系统之后重新发送其状态请求，并且该方法返回至步骤 304。

[0085] 在步骤 308 进行闪速存储器装置的多页编程操作的成功或失败的判断，其可以由主机系统通过另一个请求来进行。响应于该请求，闪速存储器装置可以输出用于指示成功或失败的多页编程的另一个信号。如果编程成功，则该方法进入步骤 310，并且主机系统可以使用闪速存储器装置启动另一个操作，诸如另一个多页编程操作。另一方面，如果多页编程失败，则该方法进入步骤 312。此时，可以将数据重新编程到不同的行地址或不同的闪速存储器，或者主机系统仅简单提供出现错误并且数据未被写入到闪速存储器装置的通知。应当注意，由闪速存储器装置提供的状态信号被写入闪速存储器装置的寄存器，该状态信号可以响应于特定命令而被输出。

[0086] 图 9 的方法是用于控制存储器装置以将数据的 m 个逻辑页编程到存储器装置的单个物理页的一般方法。许多当前可获得的闪速存储器装置可以将数据的 2 个逻辑页存储到存储器装置的单个物理页。图 10 示出根据本实施例的、用于操作闪速存储器装置以用于双页编程操作的方法。现在参考图 11 描述在图 10 中所示的方法，图 11 示出本实施例的存储器装置的操作的一般序列图。图 11 示出存储器装置输入输出端口 I/O、存储器装置的所选择的字线 WL 和位线 BL[1:j] 的信号轨迹。图 10 的方法在步骤 350 开始，其中，第一数据页被提供到存储器装置，并且被存储在存储器装置的第一页缓冲器中。如图 11 所示，在 360 处，在 I/O 端口接收数据页 1。短期之后，在步骤 352 向存储器装置提供第二数据页，其被存储在存储器装置的第二页缓冲器中。如图 11 中所示，在 362 处，在 I/O 端口接收数据页 2。从这一点开始，步骤 302 至 312 与对于图 9 所示和所述的那些相同。经由图 11 的多页编程开始命令 364，在步骤 302 处启动多页编程。

[0087] 一旦在存储器装置内启动多页编程，则在 3 个不同的时间段处驱动所选择的字线 WL，其中每一个时间段具有预定的编程配置。应当注意，仅编程每一个单元 4 个可能的状态中的 3 个，因为状态之一是默认的擦除状态。在图 11 中，在 366、368 和 370 处，通过被驱动到高逻辑电平的字线 WL 来示出该 3 个时间段。应当注意，在存储器装置中，可以在每一个时间段期间将字线 WL 驱动到不同的电压电平直到指定的时间量。在一个实施例中，可以在每个时间段后执行编程验证操作，以验证刚刚被编程到单元的阈值电压。图 11 示出在分别用于编程状态 1、2 和 3 的时间段 366、368 和 370 期间偏置位线。在本示例实施例中，在时间段 366 的结尾，特定存储器单元具有与状态 1 (ST1) 相对应的最终阈值电压，在时间段 368 的结尾，特定存储器单元具有与状态 2 (ST2) 相对应的最终阈值电压，并且在时间段 370 的

结尾,特定存储器单元具有与状态 3 (ST3) 相对应的最终阈值电压。在时间段 370 已经过去后,执行步骤 304 至 312,并且编程周期在图 11 中的 372 结束,存储器装置经由其状态寄存器向存储器控制器报告多页编程操作是否成功。

[0088] 图 12A 是示出根据本实施例的用于闪速存储器装置的示例命令和数据输入序列的序列图。新的图 11A 示出存储器装置输入 / 输出端口 I/O 和由存储器装置提供的就绪 / 忙碌信号 R/B# 的信号轨迹。针对双页的多页编程操作示出了本示例命令和数据输入序列。可以被存储器装置理解的任何格式来提供在图 12A 中所示的命令,以具有与前述相同的功能。从新的图 11A 的左侧开始,要接收的第一命令是数据加载命令 380,其后是输入数据,该输入数据包括地址信息 382 和写入数据 384。地址信息 382 可以包括存储体、行和列地址信息,并且写入数据 384 是要被编程到存储器装置的数据页。接收到数据结束命令 386 以指示输入数据的结束。直到这个时间点,R/B# 信号在高逻辑电平,以指示存储器装置处于就绪状态,并且可用于接收在 I/O 端口上出现的信息。在接收到数据结束命令 386 后,存储器装置可以在很短的时间段中处于忙碌状态,其中,R/B# 降低到低逻辑电平。

[0089] 在 R/B# 升高到高逻辑电平后,存储器控制器可以通过发送另一个数据加载命令 388 来恢复多页编程操作。随后是包括地址信息 390 和写入数据 392 的第二组输入数据。地址信息可以包括与在地址信息 382 中相同的列地址,但是具有不同的行地址,以将写入数据指定为第二逻辑页。最后的数据结束和多页编程命令 394 被接收以指示写入数据 392 的结束,并且启动存储器装置的内部多页编程操作。命令 394 可以被提供为独立的数据结束命令和独立的多页编程命令。其后,R/B# 降低到低逻辑电平直到用于将逻辑页编程到存储器装置的物理页所需的时间段过去为止。

[0090] 图 12B 示出图 12A 的命令和数据输入序列的变形。在这个实施例中,存储器装置可以在命令 386 和 388 之间没有短的忙碌时间段,并且不要求数据结束命令 386 的接收。因此,第二数据加载命令 388 也可以用作用于写入数据 384 的数据结束命令。图 12A 和图 12B 是用于控制能够进行多页编程的存储器装置的示例命令和数据输入序列,然而,可以使用导致存储器装置的页缓冲器的加载并且随后进行内部多页编程的任何命令和数据输入序列。

[0091] 如上所述,连接到所有位线的存储器单元在一个编程周期中被编程到它们各自的逻辑状态,由此避免在启动用于将两个数据页编程到同一物理页的随后的编程周期之前读出在前一个编程周期中先前编程的数据页的需要。下面是用于将所有可能的逻辑状态编程到存储器装置的存储器单元的不同技术的讨论。

[0092] 图 13 是概述根据本实施例的、用于将多个数据页编程到存储器的一个物理页的方法的流程图。假定,要在多页编程操作中编程的所有页已经被加载到存储器装置的页缓冲器内。而且,假定所有的存储器单元当前处于擦除状态,并且可以例如具有负阈值电压。这个擦除状态被称为 ST0,如图 14B 中所示。要以增大对应的阈值电压的顺序来编程不同的状态。在本示例中,编程第一编程状态 ST1,随后编程第二编程状态 ST2,然后编程第三编程状态 ST3。参考示例的每单元 2 比特的情况来描述图 13 的多页编程方法。参考图 14A 和图 14B,图 14A 是示出用于所选择的字线 WL 以及四个位线 BL1、BL2、BL3 和 BL4 的信号轨迹的时序图,并且图 14B 示出连接到所选择的字线和位线 BL1、BL2、BL3 和 BL4 中的每个位线的单元的阈值电压的编程。在该情况下,连接到 BL1、BL2、BL3 和 BL4 的存储器单元分别要存

储与 ST2、ST1、ST3 和 ST2 相对应的阈值电压。

[0093] 该方法在步骤 400 开始,其中,针对所有的位线确定用于要编程的第一状态 ST1 (其中, $n = 1$) 的位线偏置。更具体地,位线被偏置到足以用于启用存储器单元的编程的诸如低逻辑电平(VSS)的第一电压或足以禁止存储器单元的编程的诸如高逻辑电平(VDD)的第二电压。在图 14A 中的 t_1 前的时间,所选择的字线 WL 被保持在 VSS 的无效电平。进入步骤 402,位线被偏置以用于启用或禁止第一编程状态(ST1) $n = 1$ 的编程。因为状态 ST2 和 ST3 具有比与状态 ST1 相对应的阈值电压大的最终阈值电压,所以位线 BL1、BL3 和 BL4 与 BL2 一起被偏置到 VSS。随后在步骤 404,将所选择的字线驱动以编程连接到被偏置到低逻辑电平的位线的任何闪速存储器单元。在图 14A 中,在时间 t_1 将所选择的字线 WL 驱动到代表性的高逻辑电平。这个代表性高逻辑电平用于示出对于编程与 $n = 1$ 编程状态相对应的阈值电压有效的编程配置的应用。在实际使用中,可以将字线驱动到高电压电平,或者根据已知的字线编程控制方案步进。用于状态 $n = 1$ 的编程将在由用于状态 $n = 1$ 的特定编程配置所确定的时间结束,此时,在步骤 406 复位所选择的字线。

[0094] 在图 14A 中,在时间 t_2 ,将 WL 复位到 VSS。由 t_1 和 t_2 界定的时间段是编程周期的第一迭代。虽然在图 13 中未示出,但是在时间 t_2 后执行编程验证操作以验证在存储器单元中存储的 ST1 的阈值电压。为了本示例的目的,假定使用适当的阈值电压成功地对所有的存储器单元进行编程。理想地,连接到 BL2 的闪速存储器单元具有与 ST1 相对应的最终阈值电压,如图 14B 中所示。然而,连接到 BL1、BL3 和 BL4 的存储器单元现在应当具有与 ST1 相对应的中间阈值电压,如在图 14B 中的虚线的阈值电压分布曲线所示,因为这些不是连接到这些位线的单元的最终阈值电压。返回图 13,在步骤 408 判断编程状态是否是要编程的最后状态。如果是,则该方法在步骤 410 处结束。否则,存在要编程的更多的状态,并且该方法进入步骤 412,其中, n 被递增($n = 2$)以指示要在随后的迭代中编程的下一个状态。然后,该方法返回到步骤 400,其中,确定用于步骤 ST2 的位线偏置。因为连接到 BL2 的存储器单元应该具有 ST1 的最终状态,所以编程验证电路禁止 BL2 接收 VSS 电压,该 VSS 电压用于在当前的编程周期中对于所有随后的编程迭代启用编程。

[0095] 在第二迭代中重复步骤 402、404 和 406,以编程与 ST2 相对应的第二编程状态 $n = 2$ 。在图 14A 的示例中,该第二迭代在时间 t_3 和 t_4 之间发生,其中,偏置 BL1、BL3 和 BL4 以用于编程。如图 14B 中所示,连接到 BL1 和 BL4 的闪速存储器单元具有与 ST2 相对应的最后阈值电压,而连接到 BL3 的存储器单元现在具有与 ST2 相对应的中间阈值电压。在步骤 406 中复位字线后,禁止 BL1 和 BL4 在任何随后的编程迭代中接收任何另外的 VSS 电压,因为连接到这两个位线的存储器单元应该具有 ST2 的最后状态。

[0096] 因为还要编程与 ST3 相对应的编程状态,所以在第三迭代中重复步骤 400、402、404 和 406 的第三迭代,以编程与 ST3 相对应的第三编程状态 $n = 3$ 。在图 14A 的示例中,该第三迭代发生在时间 t_5 和 t_6 之间,其中, BL3 被偏置到 VSS,以用于编程。如图 14B 中所示,连接到 BL3 的闪速存储器单元具有与 ST3 相对应的最终阈值电压。因为编程状态 $n = 3$ 是要编程的最后状态,所以编程周期将在步骤 410 结束。

[0097] 因此,在上述的多页编程方法中,具有比被编程的当前阈值电压大的最终阈值电压的任何存储器单元同时将其阈值电压增大。因此,基于制造工艺、装置几何形状和每一个状态的期望的阈值电压,用于特定编程迭代的每一个字线编程配置被设计以具有用于将存

存储器单元的阈值电压改变为期望的阈值电压的累积效应。在例如图 14A 的实施例中,在用于 BL1 的时间 t_3 和时间 t_4 之间的编程迭代将存储器单元的阈值电压从中间状态 ST1 偏移到与最终状态 ST2 相对应的阈值电压。

[0098] 根据替代实施例,各字线编程配置可以被设计来将擦除阈值电压直接地偏移或改变为与最终状态相对应的阈值电压。在图 15A 和图 15B 中示出了这个实施例。假定相同的状态被编程到与 BL1、BL2、BL3 和 BL3 连接的存储器,就像在图 14A 和图 14B 中所示的示例中那样。在图 15A 和图 15B 的本实施例中,位线仅在一个编程迭代的持续时间中被偏置到用于启用编程的电压电平,其中,使用特定的编程配置来驱动字线。如图 15A 中所示,刚好在时间 t_3 驱动字线之前将 BL1 驱动到低逻辑电平 VSS,然后刚好在时间 t_4 撤消(deassert)字线后将 BL1 驱动回高逻辑电平 VDD。如图 15B 中所示,连接到位线 BL1 的存储器单元的擦除阈值电压被直接地偏移到与状态 ST2 相对应的阈值电压。在图 15A 中示出用于位线 BL2、BL3 和 BL4 的类似的偏置序列以分别用于编程状态 ST1、ST2 和 ST3。因此,存在三个特定字线编程配置,每一个对于将擦除阈值电压偏移到与 ST1、ST2 和 ST3 相对应的阈值电压的每一个有效,并且,位线被偏置以启用针对那个特定状态的编程,同时被偏置来用于禁止针对所有其他状态的编程。

[0099] 图 8 的控制电路 108 经由高压发生器 106 和行电路 104 来提供用于图 14A 和图 15A 的多页编程实施例的各个编程配置的所选择的字线的定时和电压电平控制。在两个实施例中,通过控制电路 108 并且响应于在页缓冲器中存储的数据页来控制用于偏置位线的定时。

[0100] 可以响应于在图 8 的页缓冲器 112 和 114 中存储的比特的逻辑状态来进行用于在图 14A 和图 15A 所示的多页编程方案中编程数据或禁止数据的编程的位线中的每一个的偏置。例如,在与位线相对应的页缓冲器 112 和 114 中存储的一对比特的特定逻辑状态指示位线要被偏置到对于启用所选择的存储器单元的编程有效的电压电平。例如参考图 4,状态 1、状态 2 和状态 3 各自具有一对比特,其中,该对比特的至少一个是逻辑 0。因此,在本示例中,可以检测与一个位线相对应的一对比特的至少一个逻辑 0 比特的存在,以用于将位线偏置到对于启用编程有效的电压电平。否则,如果两对比特是逻辑 1,则将位线偏置到对于禁止连接到该位线的所选择的存储器单元的编程有效的电压电平。注意,图 4 的状态 0 具有在逻辑 1 的两个比特。因为状态 0 是默认擦除状态,所以将具有与其相关联的二进制值“11”的任何位线偏置以用于禁止编程。

[0101] 因此,分配至图 4 的状态 0、状态 1、状态 2 和状态 3 的示例二进制值可以有益地用于在图 14A 中所示的多页编程方案中控制位线。更具体地,如果已经在特定编程迭代后验证了用于特定位线的最终的期望编程状态,则在每次编程迭代后将该位线偏置到高逻辑电平,以用于在随后的编程迭代中禁止进一步的编程。另一方面,如果该特定编程迭代导致中间状态的成功编程,则该位线应当在随后的编程迭代中保持被偏置以用于进一步的编程。可以通过将与各位线相对应的比特对的逻辑状态切换为不同值来实现这一点。

[0102] 在一个示例中,可以将比特对的逻辑状态切换为二进制值“11”或包括至少一个逻辑 0 的二进制值,二进制值“11”指示已经编程了最终状态,包括至少一个逻辑 0 的二进制值指示已经编程了中间状态。可以实现数据状态过渡规则以保证数据从原始数据状态向一个或多个中间状态过渡,并且最后过渡到最终状态。

[0103] 使用在图 4 中所示的示例二进制值分配和用于启用编程的在比特对中的至少一个逻辑 0 的存在,可以开发下面的示例数据过渡规则:

[0104] (1) “10”至“11”

[0105] (2) “00”至“10”

[0106] (3) “01”至“00”

[0107] 下面的表格 1 示出了用于在图 14A 中所示的位线 BL1、BL2 和 BL3 的这些数据过渡规则的示例应用。在表格 1 中,将 3 个编程迭代示出为“PGM1”、“PGM2”和“PGM3”。每个编程迭代以与相应的位线相对应的在数据缓冲器中存储的初始二进制值“初始”开始,该初始二进制值在成功的编程验证后被切换到新的二进制值“新”。

[0108] 表格 1

[0109]

	PGM 1		PGM 2		PGM 3	
	初始	新	初始	新	初始	新
BL1	“00”	“10”	“10”	“11”	-	-
BL2	“10”	“11”	-	-	-	-
BL3	“01”	“00”	“00”	“10”	“10”	“11”

[0110] 图 16 是根据本实施例的在图 8 中所示的位线访问电路 110 的功能框图。图 16 意欲示出用于响应于页缓冲器中存储的关联于或对应于特定位线的数据而偏置位线以启用或禁止所选择的存储器单元的编程的一般原理,并且假定诸如位线感测放大器和编程验证电路的其他电路存在以启用适当的操作。这样的其他电路未被示出以简化示意图。在图 16 中仅示出与第一位线 BL[1] 和最后位线 BL[j] 相关联的电路。位线访问电路 110 与在不同时间间隔被驱动的字线协同地偏置位线。页缓冲器 112 包括多个数据存储电路 500,每一个用于存储第一数据页的 1 比特。页缓冲器 114 包括多个数据存储电路 502,每一个用于存储第二数据页的 1 比特。连接到“与”逻辑门 504 的每对数据存储电路 500 和 502 与相应的位线相关联。

[0111] 除了存储第一数据页和第二数据页的比特之外,位线访问电路 110 包括数据解码电路,用于响应于来自与各位线相对应的两个数据页的比特的特定逻辑状态而将该位线偏置到编程禁止电压或编程启用电压。用于每个位线的数据解码电路包括“与”逻辑门 504 和数据验证解码器 506。在启动编程迭代之前,“与”逻辑门 504 接收在数据存储电路 500 和 502 中存储的逻辑状态,并且响应于逻辑“与”运算来偏置位线。在当前描述的示例中,当数据存储电路 500 和 502 的至少一个存储逻辑 0 时,将位线偏置到 VSS。

[0112] 数据验证解码器 506 包括逻辑电路,用于解码在数据存储电路 500 和 502 中存储的逻辑状态,并且用于响应于所检测到的用于指示成功编程的位线的状况来选择性地反转任何一个所存储的逻辑状态,以使得数据存储电路 500、502 存储新的二进制值。其可以被认为是编程验证电路。根据预定数据过渡规则来进行在数据存储电路 500 和 502 中的任何一个所存储的逻辑状态的该选择性反转。在本实施例中,响应于完成的编程迭代的成功编程验证来执行数据存储电路 500 和 502 的反转。如果存储器单元还没有被充分地编程为具

有期望的阈值电压,则重复编程。在本示例中,对于任何编程状态的成功编程的存储器单元形成到 VSS 的电流路径,由此在编程验证操作期间将预先被充电到 VDD 的位线放电到 VSS。否则,位线基本保持在预充电的 VDD 电平,以指示连接到该位线的存储器单元需要重复的编程迭代。因此,在本实施例中,当在编程验证操作期间位线在 VSS 时,执行根据数据过渡规则的选择性数据反转。否则,不执行选择性数据反转。

[0113] 图 17 示出根据本实施例的数据验证解码器 506 及其到数据存储电路 500 和 502 的互连的另外的细节。数据存储电路 500 包括逻辑 0 反转电路 600 和逻辑 1 反转电路 602。数据存储电路 502 包括逻辑 0 反转电路 604 和逻辑 1 反转电路 606,这两者分别与电路 600 和 602 相同。逻辑 0 反转电路 600 响应于从数据验证解码器 506 接收到的解码后的数据信号将在数据存储电路 500 中存储的任何逻辑 0 状态反转为逻辑 1 状态。逻辑 1 反转电路 602 响应于从数据验证解码器 506 接收到的解码后的数据信号将在数据存储电路 500 中存储的任何逻辑 1 状态反转为逻辑 0 状态。反转电路 604 和 606 执行与针对反转电路 600 和 602 上述的相同的功能。如图 17 中所示,反转电路 600、602、604 和 606 接收不同的解码后的数据信号。

[0114] 数据验证解码器 506 包括“与”逻辑门 604、606 和 608。每一个“与”逻辑门具有:第一输入端,其从数据存储电路 500 接收所存储的逻辑状态;以及,第二输入端,其从数据存储电路 502 接收所存储的逻辑状态。应当注意,每一个“与”逻辑门具有反相和同相的第一和第二输入端的不同组合,并且向特定反转电路(600、602、604 和 606)提供每一个“与”逻辑门的输出。反相和同相的第一和第二输入端的不同组合的该配置是前述的数据过渡规则的实现方式的具体示例。虽然未示出,但是每个位线包括检测电路,用于在编程验证期间检测位线的低电压电平,并且启用数据验证解码器 506。否则,如果检测到位线的高电压电平,则编程不成功,并且数据验证解码器 506 被禁用以防止反转电路 600、602、604 和 606 运行。

[0115] 图 17 的电路的示例操作现在使用前述的表格 1 的 BL3 的示例情况如下,其中,在数据存储电路 500 和 502 中存储初始二进制值“01”。因此,应当理解,数据存储电路 500 存储逻辑 0,而数据存储电路 502 存储逻辑 1。在第一编程迭代 PGM1 之后并且假定编程验证是成功的,初始二进制值“01”使得“与”门 612 向逻辑 1 反转电路 606 输出逻辑 1。作为响应,逻辑 1 反转电路 606 切换数据存储电路 502,并且初始存储的逻辑 1 被改变为逻辑 0。逻辑 0 反转电路 600 从“与”门 608 接收逻辑 0,并且因此不切换数据存储电路 500。因此,新的二进制值是在数据存储电路 500 和 502 中存储的“00”,并且,对应的位线保持被偏置以对于随后的编程迭代 PGM2 启用编程。在 PGM2 的成功编程验证后,数据验证解码器 506 可以解码来自数据存储电路 500 和 502 的“00”二进制值。现在,“与”逻辑门 608 向逻辑 0 反转电路 600 输出逻辑 1,逻辑 0 反转电路 600 切换数据存储电路以将所存储的逻辑 0 改变为逻辑 1。“与”逻辑门 610 和 612 的输出处于无效的逻辑 0 状态,因此不对数据存储电路 502 进行任何改变。

[0116] 现在,在数据存储电路 500 和 502 中存储新的二进制值“10”,并且,对应的位线保持被偏置以对于随后的编程迭代 PGM3 启用编程。在 PGM3 的成功编程验证后,数据验证解码器 506 可以对来自数据存储电路 500 和 502 的“10”二进制值进行解码。与逻辑门 604 向逻辑 0 反转电路 604 输出逻辑 1,逻辑 0 反转电路 604 切换数据存储电路 502 以将所存储

的逻辑 0 改变为逻辑 1。现在,数据存储电路 500 和 502 都存储逻辑 1,并且位线被偏置到 VDD 以禁止编程。当前示出的示例性数据验证解码器 506 及其到反转电路的连接保证在数据存储电路 500 和 502 中存储的图 4 的任何初始可编程状态将根据数据过渡规则被过渡到下一个中间或最后状态。

[0117] 当前描述的数据过渡规则和在图 16 和 17 中所示的示例电路实施例被提供来示出在基于对于可能的存储状态的特定的二进制值分配(诸如在图 4 中所示的二进制值分配)而偏置位线之后的原理。可以使用这 4 个状态的不同二进制值的分配,并且,可以开发不同的数据过渡规则集合。因此,数据验证解码器 506 及其到反转电路的连接配置可以被适配来适应于这样的替代二进制值分配和数据过渡规则。当前描述的实施例涉及每一个单元 2 比特的配置,但是可以基于当前所述的原理而对于每一个单元 3 比特或更多的配置容易地调整。可以使用其他电路技术来实现同一期望的结果。而且,本示例描述了如何使用逻辑“0”来指示位线的进一步的偏置以启用编程,但是替代实施例可以使用逻辑“1”来代替。

[0118] 因此,在前面的实施例中描述的多页编程电路和方法相对于现有技术来说减少了总的编程时间,在该现有技术中,需要多个编程周期来用于将数据的各逻辑页编程到物理页。这是因为下述情况:在本多页编程实施例中,由于在一个编程周期内编程所有的逻辑页,因此消除了用于在不同的编程周期中编程不同的逻辑页通常所需的定时开销。而且,与使用单独的编程周期来编程每个逻辑页相比,闪速存储器单元的编程压力有所降低。另外,相对于现有的编程技术,使用当前所述的多页编程实施例时的编程需要这个能力。

[0119] 前述的实施例可以被进一步改善以减少在一个编程周期中编程所有的状态所需的整体时间。在例如图 11 的前述实施例中,顺次执行多个编程迭代以将每一个状态编程到用于那个特定迭代的所选择的单元。更具体地,用于将多个状态或页编程到存储器单元的一个编程周期涉及:顺次施加不同的位线电压,并且使用用于要编程的特定状态/阈值电压的特定编程配置来驱动所选择的字线。因此,用于编程周期的总的的时间是各编程迭代所需的时间的和。根据本替代实施例,可以通过在单个编程迭代中将所有的可能状态编程到单元来显著地减小编程周期时间。换句话说,取代用于编程各状态的顺次编程迭代,当使用编程配置驱动所选择的字线时,并行编程所有的状态。这被称为并行多态编程。

[0120] 图 18A 是示出根据本实施例的闪速存储器装置的示例双页并行多态编程操作的序列图。这个序列图在下述方面类似于在图 11 中所示的序列图:在 360 在 I/O 端口接收到数据页 1,并且在 362 在 I/O 端口接收到数据页 2。经由多页编程开始命令 364 来启动多页编程,如在图 11 中那样,在 372,存储器装置经由其状态寄存器向存储器控制器报告多页编程操作是否成功,以向存储器控制器指示编程周期已经结束。然而,取代编程所有的状态所需的 3 个时间段 366、368 和 370,仅需要一个单个时间段 700。在其中使用预定的编程配置来驱动所选择的字线的这个时间段 700 期间,位线 BL[1:j] 被偏置到特定电压电平以编程与不同状态相对应的阈值电压,其中,对于特定的制造工艺校准电压电平。注意,字线的编程配置被优化以针对特定制造工艺来有效编程阈值电压。

[0121] 如图 18A 中所示,在时间段 700 期间,在驱动字线时,响应于特定位线电压电平而同时将状态 ST1、ST2 和 ST3 编程到存储器单元。图 18B 是示出连接到位线 BL1、BL2、BL3 和 BL4 的在图 18A 中被编程的 4 个闪速存储器单元的阈值电压的阈值电压分布图。如图 18B 中所示,存储器单元在时间段 700 期间从 ST0 偏移与 ST1、ST2 和 ST3 相对应的它们各自

的最终编程阈值电压。

[0122] 图 19A 和图 20A 是示出闪速存储器装置的示例双页混和的顺次和并行的多页编程操作的替代序列图。在这些替代实施例中,使用多个编程迭代,其中,在一个迭代中并行地编程要编程的可能状态的子集,而在第二迭代中编程剩余的一个或多个状态。

[0123] 在图 19A 中,附图标记 360、362、364 和 372 与之前对于图 18A 和图 11 所述的相同。在这个替代实施例中,在时间段 702 编程一个状态 ST1,随后顺次在时间段 704 进行状态 ST2 和 ST3 的并行多态编程。注意,在时间段 702 处发生第一编程迭代后,被指定来用于存储状态 ST2 和 ST3 的存储器单元将具有与状态 ST1 相对应的相同阈值电压。在随后的时间段 704 中,具有与 ST1 相对应的阈值电压的单元被进一步调整以对应于状态 ST2 和 ST3。如在图 19B 的阈值电压分布图中所示,以与状态 ST0 相对应的擦除阈值电压开始的存储器单元在时间段 702 中被偏移到与 ST1 相对应的编程阈值电压。在时间段 704 中,连接到位线 BL1、BL3 和 BL4 的存储器单元将与 ST1 相对应的它们的中间阈值电压偏移到与 ST2 和 ST3 相对应的最终阈值电压。

[0124] 在图 20A 的实施例中,状态 ST1 和 ST2 的并行多态编程在时间段 706 发生,随后顺次在时间段 708 进行剩余的状态 ST3 的编程。注意,在时间段 706 发生的第一编程迭代后,被指定用于存储状态 ST3 的存储器单元具有与具有较高阈值电压的状态相对应的相同阈值电压,该状态在这个示例中是状态 ST2。在随后的时间段 708 中,具有与 ST2 相对应的阈值电压的单元被进一步调整为与状态 ST3 相对应。如在图 20B 的阈值电压分布图中所示,连接到 BL2 的存储器单元在时间段 706 中将其擦除阈值电压偏移到与 ST1 相对应的阈值电压。而且,在时间段 706 中,连接到 BL1、BL3 和 BL4 的存储器单元将它们的擦除阈值电压偏移到与 ST2 相对应的阈值电压,包括连接到 BL3 的存储器单元。在时间段 708 中,连接到位线 BL3 的存储器单元将与 ST2 相对应的其中间阈值电压偏移到与 ST3 相对应的最终阈值电压。

[0125] 在图 19A、19B、20A 和 20B 中所示的两个替代实施例中,并行编程至少两个状态。应当注意,这些实施例不是限制性的,因为替代实施例可以包括使得第一编程迭代并行编程至少 2 个状态,随后进行第二编程迭代以并行编程至少另外 2 个状态。而且,可以使用并行多态编程迭代和单态编程迭代的组合。

[0126] 对于在图 18A、图 19A 和图 20A 中所示的实施例,在图 12A 和图 12B 中所示的命令和数据输入序列可以用于向存储器装置加载数据页,并且启动多态并行编程操作、或两个混和顺次和并行多页编程操作之一。

[0127] 通过将位线偏置到特定电压来实现并行多态编程的当前公开的实施例,这被称为位线电压调制。如上所述,根据位线被偏置或被设置为的逻辑电平,连接到位线的存储器单元经由所选择的字线被禁止编程或启用编程。通过引用包含于此的共同拥有的美国专利 No. 7, 511, 996 描述了用于禁止存储器单元的编程的实施例,其中,被选择来用于编程的存储器单元将它们的位线偏置到诸如 0V 的编程电压,并且不被编程的存储器单元将它们的位线偏置到诸如 VCC 的编程禁止电压。首先提供在美国专利 No. 7, 511, 996 中描述的一个这样的实施例的说明,以便于理解向位线施加的编程禁止电压如何可以禁止所选择的存储器单元的编程。

[0128] 图 21 是示出美国专利 No. 7, 511, 996 的编程禁止方法的实施例的流程图。在本讨

论中,上字线指的是在所选择的存储器单元和位线之间的那些字线和存储器单元,而下字线指的是在所选择的存储器单元和源极线之间的那些字线和存储器单元。图 21 的编程禁止方法在步骤 800 开始,其中,使用禁止电压或编程电压来偏置位线。在当前所述的示例中,假定位线被驱动到禁止电压 VCC。在步骤 802,将所选择的字线和上字线驱动到第一通过电压。该第一通过电压足够高以接通与上字线相对应的每一个存储器单元,而与其编程或未编程状态无关。与步骤 802 同时地,下字线在步骤 804 被驱动到第二通过电压。根据本方法的实施例,第一通过电压和第二通过电压可以彼此相同,或者第二通过电压可以小于第一通过电压。然后在步骤 806 将所选择的字线驱动到编程电压,随后在步骤 808 进行隔离操作以将所选择的存储器单元从 NAND 串的未选择的存储器单元去耦。

[0129] 图 22A 是本领域内的技术人员应当公知的 NAND 串 900 的简化截面图。NAND 串 900 包括:具有连接到字线 WL0 至 WL31 的栅极的存储器单元 902;串选择晶体管 904,用于将 NAND 串 900 耦合到位线 906;以及,接地选择晶体管 908,用于将 NAND 串 900 耦合到源极线 910。NAND 串 900 形成在 PP 阱内,PP 阱本身形成在 P 衬底的 N 阱内。NAND 串存储器单元晶体管的源极/漏极区域是 n+ 扩散区域。

[0130] 图 22A 包括根据本发明的一个实施例的、用于示出向特定字线施加的相对电压电平的表示。注意物理 NAND 串相对于现有技术保持不变。在图 22A 中所示的示例中,假定连接到 WL29 的存储器单元被选择来用于编程,并且,要编程的数据是逻辑“1”状态。因为擦除状态是逻辑“1”,所以要禁止编程。图 22B 是进一步示出在图 22A 中施加的电压的序列的序列图。

[0131] 时间段 T1 是初始状态,其中,在应用于编程存储器单元和用于禁止存储器单元的编程的位线电压的同时,所有存储器单元 902 的栅极、信号 SSL 和信号 GSL 被偏置到 0V。CSL 可以被偏置到 0V 或 VCC,但是可以优选地在本示例中被偏置到 VCC 以最小化泄漏。在本示例中,位线 906 被偏置到 VCC 以禁止编程。在时间段 T2,信号 SSL 被偏置到 VCC 以将位线电压耦合到 NAND 串 900。在图 22A 中所示的 NAND 串 900 中,串选择晶体管 904 电截止,因为连接到位线 906 的其漏极端子和其栅极都被配置到 VCC。更具体地,串选择晶体管 904 一旦其源极升高到大约 VCC 减去晶体管的阈值电压将会截止。在时间段 T3 期间,所有的字线(上、下和所选择的字线)被驱动至通过电压 V2,其中,V2 被选择得大于 VCC。这具有将 NAND 串 900 的整个通道升高到主升高电压的效果。在时间段 T4 的开始,所选择的字线 WL29 被升高到编程电压 V1,而相邻的字线 WL28 和 WL30 被降低到去耦电压 V3。这些分别对应于在图 5 中的步骤 104 和 106。电压电平 V1 被设置得大于 V2,并且电压电平 V3 被设置得小于 V1 和 V2。

[0132] 将 WL28 和 WL30 的字线电压降低为 V3 的另一个优点是补偿在连接到 WL29 的控制栅极以及耦合到 WL28 和 WL30 的浮动栅极之间的电容耦合。当 WL29 被驱动至 V1 时,电容耦合效应可以增大浮动栅极电压,由此增大与所选择的存储器单元紧邻的存储器单元中的通过干扰的可能性。因此,将字线电压降低为 V3 将减小浮动栅极电压。

[0133] 在图 22A 中所示的实施例中,可以根据下面的标准来选择 V1、V2 和 V3 的电压电平。V1 应当是足够高以促进耦合到被偏置到 0V 的位线的存储器单元中的 F-N 隧道化的电压电平。电压电平 V2 应当被选择得满足下面三个标准。首先,V2 应当足够高以保证使得未选择的单元的通道被导通。第二,当电压 V2 在时间段 T4 期间降低到用于 WL28 和 WL30 的

电压 V3 时, 连接到 WL28 和 WL30 的存储器单元关断, 并且在时间段 T4 期间保持关断。这是实现的局部去耦效应。第三, 当所选择的字线 WL29 从电压 V2 升高到编程电压 V1 时, 可以将所选择单元的通道升高到期望的电压, 即, 升高到辅助升高电压电平。V3 应当是足够高以将 0V 位线电压通过未选择的单元到达所选择的单元的漏极并且同时足够低以充分关断与所选择的单元 (连接到 WL29) 紧邻的存储器单元 (连接到 WL28 和 WL30) 的通道电压电平。

[0134] 通过在编程迭代期间在驱动字线之前改变向位线施加的电压, 当前描述的编程禁止方案可以被修改以用于当前描述的并行多态编程实施例。在前述的编程禁止实施例中, 当位线被偏置到 VCC 时, 编程禁止电压 VCC 的施加最小化了在被驱动到 V1 的所选择的字线和所选择的单元的通道之间的电场强度。根据本实施例, 通过将位线偏置到在 VCC 和 VSS 之间的电压, 这个电场强度相对于全 VCC 禁止情况增大, 由此允许单元的阈值电压正向偏移。然而, 阈值电压偏移的量小于位线被偏置到 VSS 的情况。因此, 单元被部分地禁止被完全编程。

[0135] 图 23 是四个闪速存储器 NAND 单元串的电示意图, 用于示出用于每单元两个比特的存储方案的并行多态编程实施例的位线的示例电压偏置。已经对于之前在图 22A 中所示的 NAND 单元串描述了在图 23 中所示的 NAND 单元串的元件, 并且因此, 使用相同的附图标记来指定相同的元件。NAND 单元串的每一个连接到相应的位线 BL0、BL1、BL2 和 BL3。在当前所示的示例中, BL0 被偏置到 VCC, BL1 被偏置到 VBL1, BL2 被偏置到 VBL3, 并且 BL3 被偏置到 VSS。根据本实施例, 在驱动字线之前同时地使用这四个电压之一来偏置所有的位线。然后, 例如使用所述电压和在图 22B 中对于所述的编程禁止方案概述的序列来驱动字线以进行编程。在一个电压用于禁止编程并且另一个用于启用编程的原理下运行的其他编程禁止方案可以适于由本实施例教导的并行多态编程。

[0136] 在每单元两个比特的存储方案的一个示例中, 可以将位线偏置到编程禁止电压 VCC、编程电压 VSS、VBL1 和 VBL2 之一, 其中, VBL1 和 VBL2 都大于 VSS 并小于 VCC。例如, VBL1 可以是 1 伏特, 并且 VBL2 可以是 2 伏特。可以基于诸如制造工艺、期望的阈值电压和用于所选择的字线的编程配置之类的因素来确定用于 VBL1 和 VBL2 的特定电压电平。如上所述的相同原理可以应用到其中要在存储器单元中存储超过两个比特的实施例。

[0137] 将位线向特定电压电平的偏置取决于与特定位线相对应的数据页的比特的逻辑状态。图 24 是根据本实施例的用于调制用于并行多态编程的位线电压电平的位线访问电路的框图。位线访问电路 1000 包括页缓冲器 1002 和至少一个多页编程缓冲器 1004。页缓冲器 1002 包括多个数据存储电路 1006, 每一个用于存储第一数据页的 1 比特。页缓冲器 1004 包括多个数据存储电路 1008, 每个用于存储第二数据页的 1 比特。连接到 BL 调制器 1010 的每对数据存储电路 1006 和 1008 与相应的位线相关联。用于每个位线的数据解码电路包括 BL 调制器 1010 和数据验证解码器 1012。元件 1002、1006、1008 和 1012 与在图 16 中所示的元件 112、114、504、502 和 506 基本上相同。在图 24 和图 16 的实施例之间的主要差别是 BL 调制器 1010 和数据验证解码器 1012 的解码逻辑电路, BL 调制器 1010 负责解码与特定位线相关联的两个数据页的数据位, 并且使用预定电压偏置该位线。在存储阵列中的每一个 BL 调制器 1010 在启动编程迭代之前同时偏置其各自的位线。

[0138] 图 25 是示出 BL 调制器 1010 的功能块的示意图。BL 调制器 1010 包括编程数据解码器 1100 和位线偏置电路 1102。编程数据解码器 1100 从第一页缓冲器接收被称为 DATA1_

[1] 的数据位并且从第二页缓冲器接收被称为 DATA2_[1] 的数据位。“DATA1”项指示比特来自第一页缓冲器，而“DATA2”项指示比特来自第二页缓冲器。“[1]”项指示在数据页中的比特位置。编程数据解码器 1100 解码这两个数据位以提供选择信号 S0、S1、S2 和 S3，其中仅一个信号处于用于 DATA1_[1] 和 DATA2_[1] 的任一组合的有效逻辑电平。这些选择信号被位线偏置电路 1102 使用来向位线 BL[1] 传送预定电压 VCC、VSS、VBL1 或 VBL2 之一，其中， $VCC > VBL2 > VBL1 > VSS$ 。在本实施例中，各选择信号被映射到对应的预定电压。当前示出的实施例被配置来用于每单元两个比特的存储方案，该方案可以被调整来用于每单元三个比特或更多的存储方案。因此，对于 DATA1_[1] 和 DATA2_[1] 的每种可能组合，BL 调制器 1010 将其位线选择并偏置到预定电压 VCC、VSS、VBL1 或 VBL2 之一。可以通过电压发生器来产生电压。

[0139] 图 26 是示出在图 25 中所示的编程数据解码器 1100 的实施例的示例电路示意图。编程数据解码器 1100 包括“与”逻辑门 1200、1202、1204 和 1206，每一个具有第一输入端和第二输入端，用于接收页缓冲器数据位 DATA1_[1] 和 DATA2_[1]。每个“与”逻辑门提供选择信号 S0、S1、S2 和 S3 之一。“与”逻辑门的第一和第二输入端被不同地配置以保证针对 DATA1_[1] 和 DATA2_[1] 的任何组合仅选择信号中的一个被驱动到高态有效逻辑电平。在图 26 中所示的示例逻辑门配置是可以使用的一种可能的配置。不同类型的逻辑门和输入端配置可以用于获得由在图 26 中所示的“与”逻辑门配置所示的相同结果，在图 26 中所示的“与”逻辑门配置要选择施加到位线的特定电压。

[0140] 图 27 是示出在图 25 中所示的位线偏置电路 1102 的实施例的示例电路示意图。位线偏置电路 1102 包括 p 沟道晶体管 1300 以及 n 沟道晶体管 1302、1304 和 1306。p 沟道晶体管 1300 响应于在高逻辑电平的 S0 经由逆变器 1308 将位线 BL 连接到 VCC。N 沟道晶体管 1302 响应于在高逻辑电平的 S1 将位线 BL 连接到 VBL1。N 沟道晶体管 1304 响应于在高逻辑电平的 S2 将位线 BL 连接到 VBL2。N 沟道晶体管 1306 响应于在高逻辑电平的 S3 将位线 BL 连接到 VSS。因此，通过由编程数据解码器 1100 提供的选择信号 S0、S1、S2 和 S3，可以向位线 BL 施加预定电压之一。可以使用不同的电路配置来响应于选择信号向位线提供不同的电压。

[0141] 在图 26 中所示的电路实施例可以用于图 18A 的并行多态编程实施例，但是该电路的修改版本将用于根据另一个实施例的图 19A 的混和顺次和并行多页编程实施例。在这样的替代实施例中，编程数据解码器 1100 的逻辑被配置使得在用于编程状态 ST1 的第一编程迭代中，被指定用于存储状态 ST2 或 ST3 的任何存储器单元将其各自的位线偏置到与 ST1 相对应的相同位线电压，该电压在当前示例中为 VBL1。可以通过设计逻辑电路来如此进行以实现该期望的结果，或者替代地，位线偏置电路 1102 的晶体管 1304 和 1306 可以被配置来在第一编程迭代中接收 VBL1，而非分别接收 VBL2 和 VSS。在用于并行编程状态 ST2 和 ST3 的下一个编程迭代中，可以以上述的方式使用图 26 的电路。

[0142] 图 20A 的混和顺次和并行多页编程实施例要求对编程数据解码器 1100 的逻辑的类似修改。在该情况下，在第一编程迭代中编程状态 ST1 和 ST2。然而，被指定来用于存储状态 ST3 的存储器单元应当在第一编程迭代中被编程到中间 ST2 状态。因此，编程数据解码器 1100 的逻辑将被配置以使得在用于编程 ST1 和 ST2 的第一编程迭代中，被指定来用于存储状态 ST3 的任何存储器单元将其相应的位线偏置到与 ST2 相对应的相同位线电压，该

电压在当前示例中为 VBL2。再一次,逻辑解码电路可以被设计来实现该期望的结果,或者替代地,可以使用用于位线偏置电路 1102 的电压电平切换。

[0143] 位线访问电路 1000 的当前描述的实施例可以用于执行在图 18A 中所示的全并行多态编程实施例,其中,同时编程所有的可能状态。针对其中在编程另一个状态的另一个编程迭代之前或之后对于一些状态执行并行多态编程的图 19A 和 20A 的实施例,可以根据图 16 和图 17 的实施例的教导来修改位线访问电路 1000 的前述实施例。这样的修改使得该电路适应于对于在图 18A 中所示的全并行多态编程实施例和在图 19A 和图 20A 中所示的混和顺次和并行多页编程实施例特有的编程验证操作。下面是用于前述的多态编程实施例的编程验证的讨论。

[0144] 在其中在每个编程迭代中编程一个状态的前述实施例中,在每个迭代后执行编程验证操作,以保证期望的阈值电压已经被正确地编程到存储器单元。对于当前所述的多态编程实施例,在一个编程迭代中同时编程至少 2 个状态。因此,新的编程验证方案用于验证与该至少 2 个状态相对应的阈值电压已经被编程到它们各自的单元。

[0145] 对于图 18A 的实施例,虽然并行地编程所有的状态,但是对于各状态顺次执行编程验证。在本实施例中,当连接到位线的存储器单元已经被验证为被成功地编程时,与该位线相对应的 2 个数据缓冲器比特被改变为二进制值“11”,其对应于用于禁止进一步编程的擦除状态。然而,仅用于被验证的编程状态的数据缓冲器比特应当被切换到擦除状态。换句话说,用于未被验证的状态的数据缓冲器比特不应当在用于另一个编程状态的编程验证操作期间被切换到擦除状态。如之前对于图 16 的实施例所述,数据验证解码器 506 在编程验证操作中切换数据缓冲器寄存器比特。因此,应当对于在编程到当前正在执行编程验证操作的状态期间偏置的那些位线启用这个电路。

[0146] 图 28 示出对于图 18A 的多态编程实施例而言的、用于选择性地启用或禁止图 24 的实施例的数据验证解码器 1012 的控制电路的电路示意实施例。编程验证控制电路 1400 包括“与”逻辑门 1402、1404 和 1406、以及“或”逻辑门 1408。每一个“与”逻辑门具有第一输入端,用于接收由在图 26 中所示的编程数据解码器 1100 提供的选择信号 S1、S2 或 S3 之一。每一个“与”逻辑门具有第二输入端,用于接收分别与选择信号 S1、S2 和 S3 相对应的验证控制信号 Verify(验证)S1、VerifyS2 和 VerifyS3。“与”逻辑门 1402、1404 和 1406 的输出被门 1408 一起逻辑“或”,以提供用于启用其对应的数据验证解码器 1012 的输出。注意,对于状态 S0 不需要任何“与”逻辑门,因为状态 S0 是不必被编程验证的擦除状态。假定,数据验证解码器 1012 被配置来响应于来自编程验证控制电路 1400 的启用信号。

[0147] 在运行中,编程验证控制电路 1400 在不同的时间段接收在高态有效逻辑电平的验证控制信号 Verify S1、Verify S2 和 Verify S3,其中,每一个时间段对应于用于特定状态的编程验证操作。例如,可以首先确立 VerifyS1 以用于验证状态 S1,然后确立 Verify S2 以用于验证状态 S2。如上所述,编程数据解码器 1100 对于 DATA1_[1] 和 DATA2_[1] 的任一给定逻辑状态组合仅确立选择信号 S0、S1、S2 和 S3 之一。因此,当确立每个验证控制信号 Verify S1、Verify S2 和 Verify S3 以验证特定状态时,仅启用与被偏置来编程该特定状态的位线相关联的数据验证解码器 1012,并且禁止所有其他数据验证解码器 1012。在图 28 中所示的特定逻辑门配置是一种可能配置的示例。替代的逻辑门配置可以用于使用与其他控制信号组合的选择信号来启用数据解码器 1012 以获得相同的结果。

[0148] 一旦被启用,数据验证解码器 1012 就被配置来在特定状态的编程已经被认为成功的情况下将与该位线相对应的页缓冲器的数据位的对切换到擦除状态。否则,该对数据位不变。因此,未被成功地编程的存储器单元将它们的对应的位线偏置到由选择信号 S1、S2 和 S3 选择的特定电压电平以进行重新编程。在本实施例中,数据验证解码器 1012 包括逻辑电路,该逻辑电路被配置来将与状态 ST1、ST2 和 ST3 相对应的数据位的对切换到与擦除状态相对应的逻辑电平,诸如“11”。

[0149] 既然已经描述了用于并行多态编程实施例的编程验证的技术,应当注意,用于图 19A 和图 20A 的混和顺次和并行多页编程实施例的编程验证技术组合了图 16 的顺次编程实施例中使用的编程验证实施例和用于图 22 的并行多态编程实施例的编程验证实施例。

[0150] 以图 19A 的实施例开始,要编程的第一状态是状态 ST1,然后是状态 ST2 和 ST3 的并行编程。然而,被指定来用于存储状态 ST2 和 ST3 的存储器单元在第一编程迭代中被编程,并且因此将状态 ST1 存储为中间状态。第一编程迭代类似于在图 14B 中所示的那个,其中,所有的单元被编程来将状态 ST1 存储为最终状态或中间状态。因此,可以使用图 14B 的前述的编程验证技术。在下一个编程迭代中,中间状态被偏移到最终状态。因此,作为用于第一编程迭代的成功编程验证操作的一部分,数据验证解码器 1012 的逻辑电路被配置来:i)将与状态 ST1 相对应的数据位切换到擦除状态;或者,ii)将与状态 ST2 相对应的数据位切换到用于选择位线电压 VBL2 的组合;或者,iii)将与状态 ST3 相对应的数据位切换到用于选择位线电压 VSS 的组合。因此,位线被偏置到用于第二编程迭代的新的电压电平,以并行地编程状态 ST2 和 ST3。用于图 18A 的实施例的编程验证技术可以用于第二编程迭代。

[0151] 对于图 20A 的实施例,在第一编程迭代中并行地编程状态 ST1 和 ST2,然后在第二编程迭代中编程状态 ST3。对于第一编程迭代,可以使用针对图 18A 的实施例描述的编程验证技术。作为用于第一编程迭代的成功编程验证操作的一部分,数据验证解码器 1012 的逻辑电路被配置来:i)将与状态 ST1 或 ST2 相对应的数据位切换到擦除状态;或者,ii)将与状态 ST3 相对应的数据位切换到用于选择位线电压 VSS 的组合。因此,位线被偏置到用于第二编程迭代的新的电压电平,以编程状态 ST3。用于图 14B 的实施例的编程验证技术可以用于第二编程迭代。

[0152] 前面公开和描述的电路和逻辑配置被提供为可以用于实现不同的多态编程实施例和对应的编程验证实施例的实施例的示例。可以使用替代电路和逻辑配置来获得相同的期望结果。

[0153] 图 29 是概述根据本实施例的用于并行多态编程的方法的流程图。该方法可以用在图 9 的方法的块 302 中。图 28 的方法通过下述方式在 1500 开始:如在前面公开的电路实施例中所述那样解码与每个位线相对应的数据页的比特,以确定要编程的状态。在 1502,位线被偏置到与要编程的不同状态中的每个状态相对应的电压电平。一旦位线被偏置到它们各自的电压电平,则使用用于将不同状态同时编程到存储器单元的编程配置在 1504 驱动所选择的字线,其中,每一个不同的状态具有对应的阈值电压。一旦编程完成,则可以在 1506 执行根据前述实施例的编程验证操作。元件 1500、1502、1504 和 1506 可以用于所有可能状态的全并行多态编程或用于所有可能状态的子集的并行多态编程。

[0154] 例如,在 1500 处解码数据之前,可以在 1508 执行第一编程迭代来编程一个状态,并在 1510 处使用对应的编程验证操作,由此,留下要在 1500、1502 和 1504 处并行地编程剩

余的可能状态(第二编程迭代)。在 1500、1502、1504 和 1506 处执行的编程操作可以用于编程所有的可能状态,除了作为第一编程迭代的一个状态之外。可以在第二编程迭代 1512 中编程剩余的状态,并在 1514 处使用对应的编程验证操作。当前描述的实施例不限于在第一编程迭代中编程一个状态,然后在第二编程迭代中并行地编程剩余的状态,或者反之亦然。在其中存在许多要编程的状态的情况下,可以使用多个编程迭代,其中,在每一个编程迭代中编程至少 2 个状态。

[0155] 在如上所述的实施例中,为了简化,如在附图中所示,装置元件和电路彼此连接。在本发明的实际应用中,元件、电路等可以彼此直接地连接。元件、电路等也可以通过装置和设备的操作所需的其他元件、电路等间接地彼此连接。因此,在实际配置中,电路元件和电路直接地或间接地彼此耦合或连接。

[0156] 本发明的上述实施例仅是示例。在不偏离由本发明所附的权利要求唯一限定的本发明的范围的情况下,本领域内的技术人员可以对于特定实施例进行修改和改变。

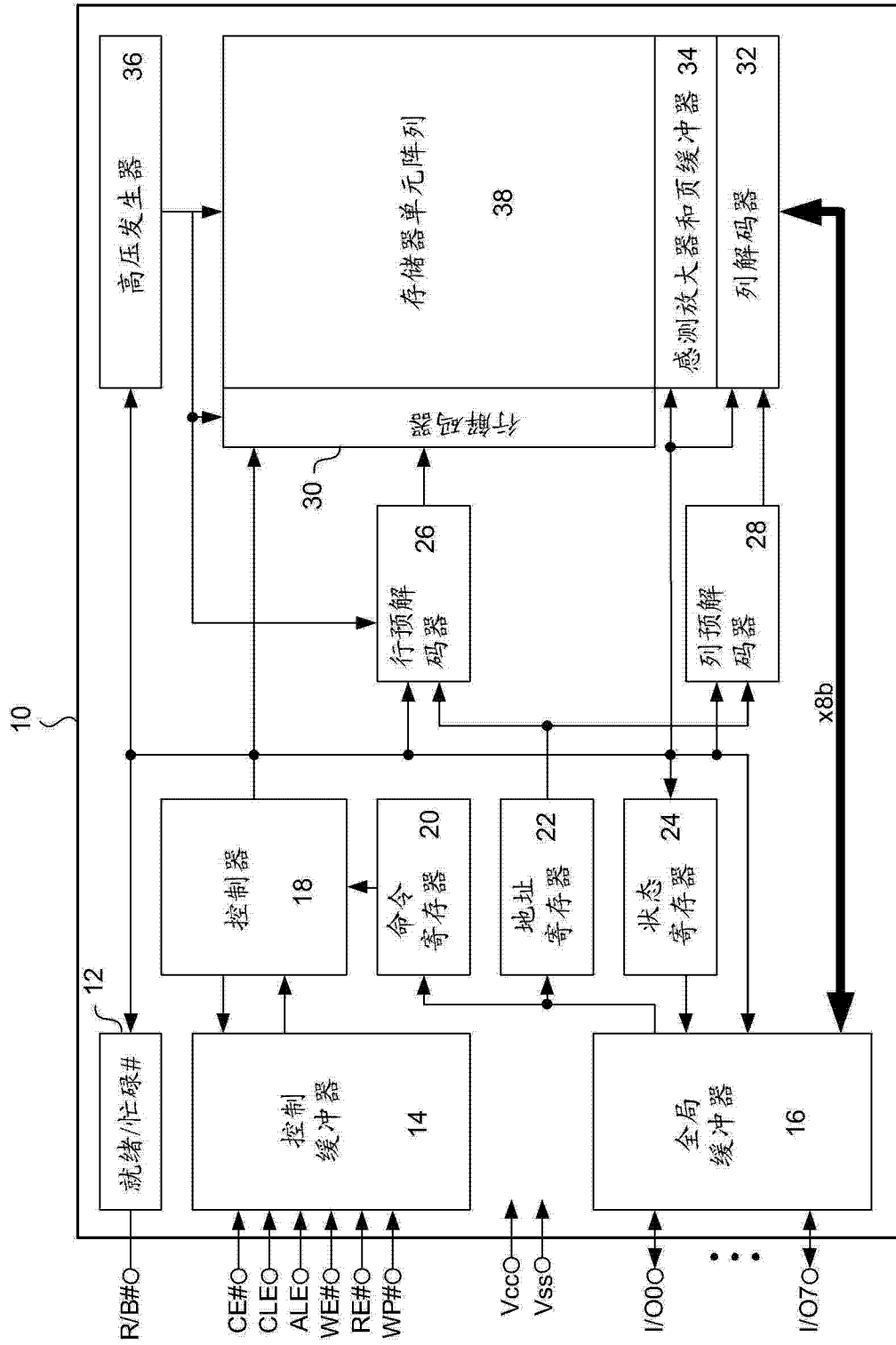


图 1

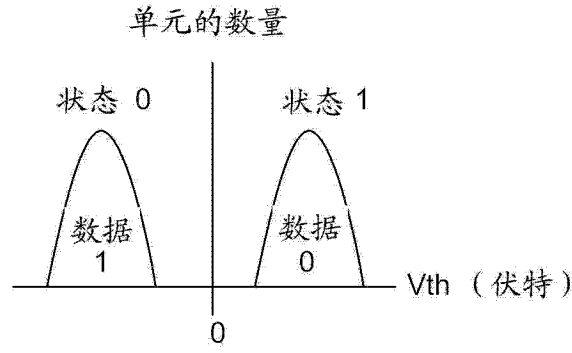


图 2

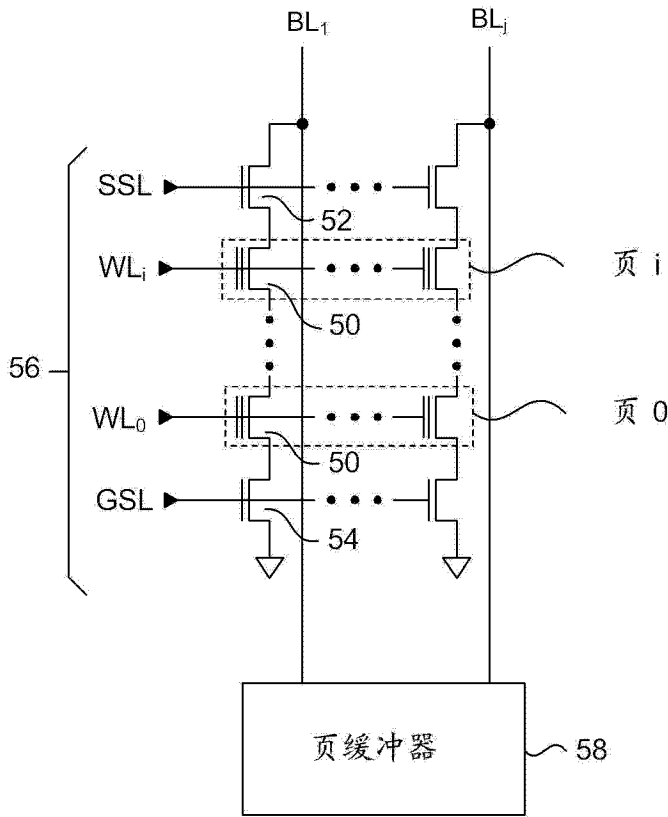


图 3

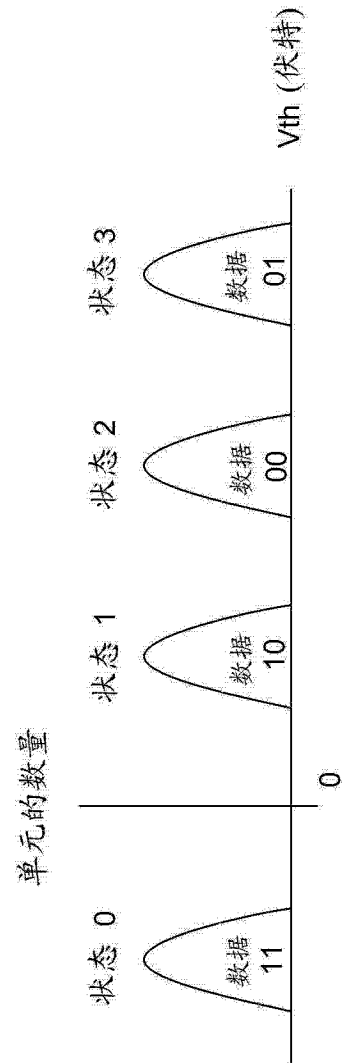


图 4

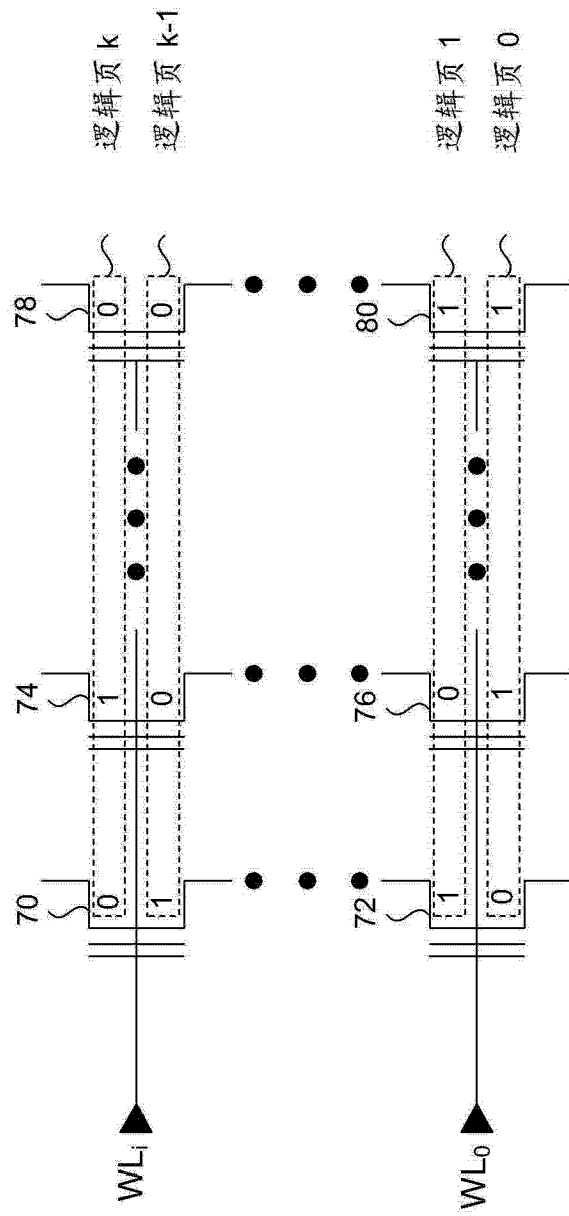


图 5

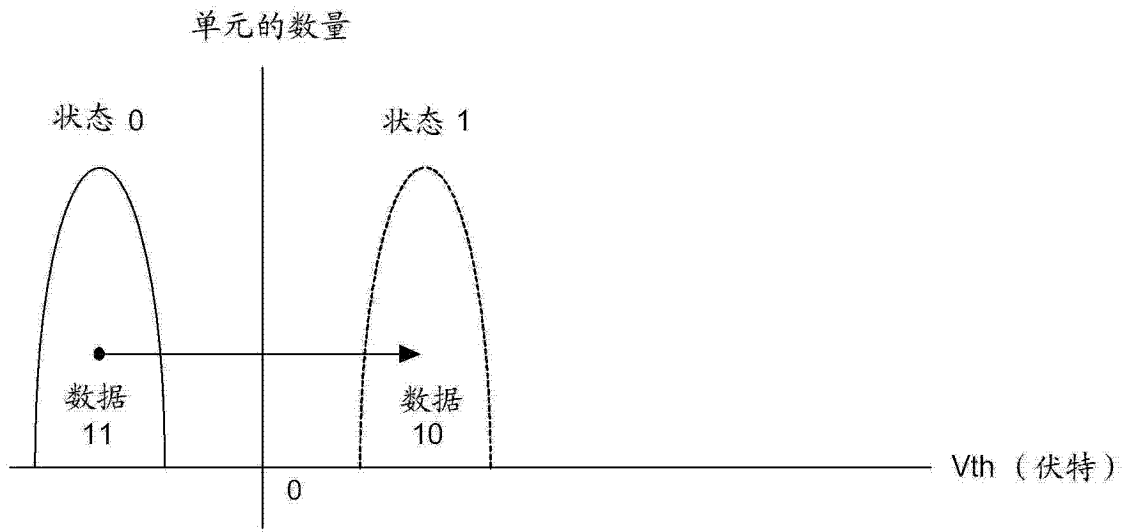


图 6A

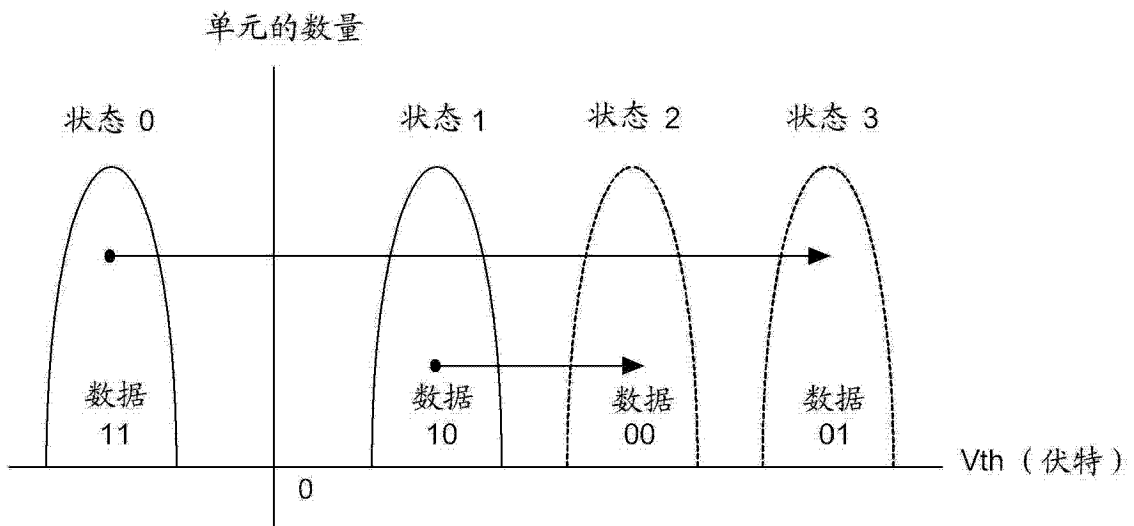


图 6B

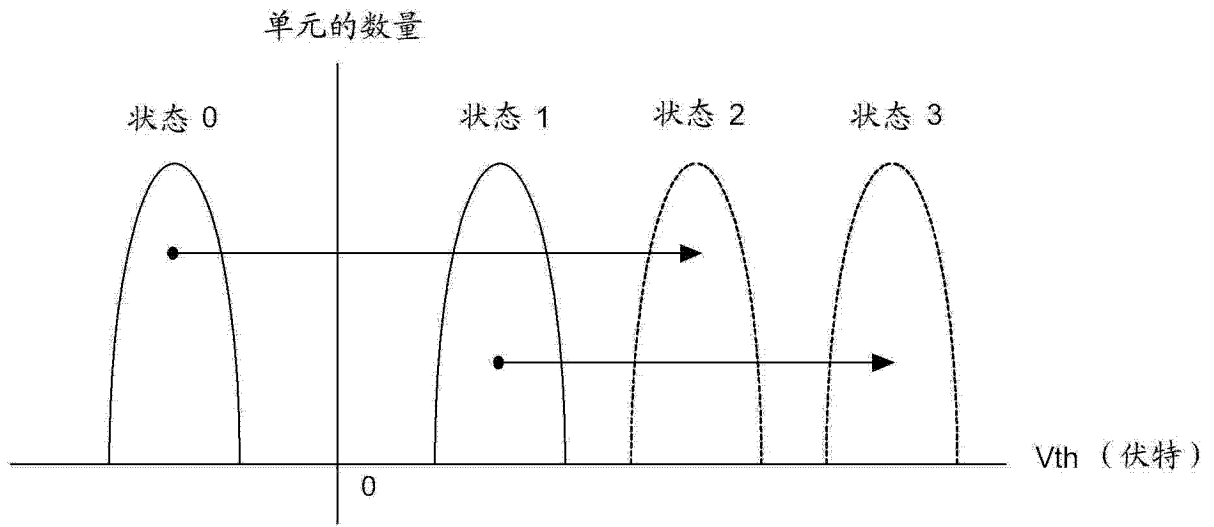


图 7

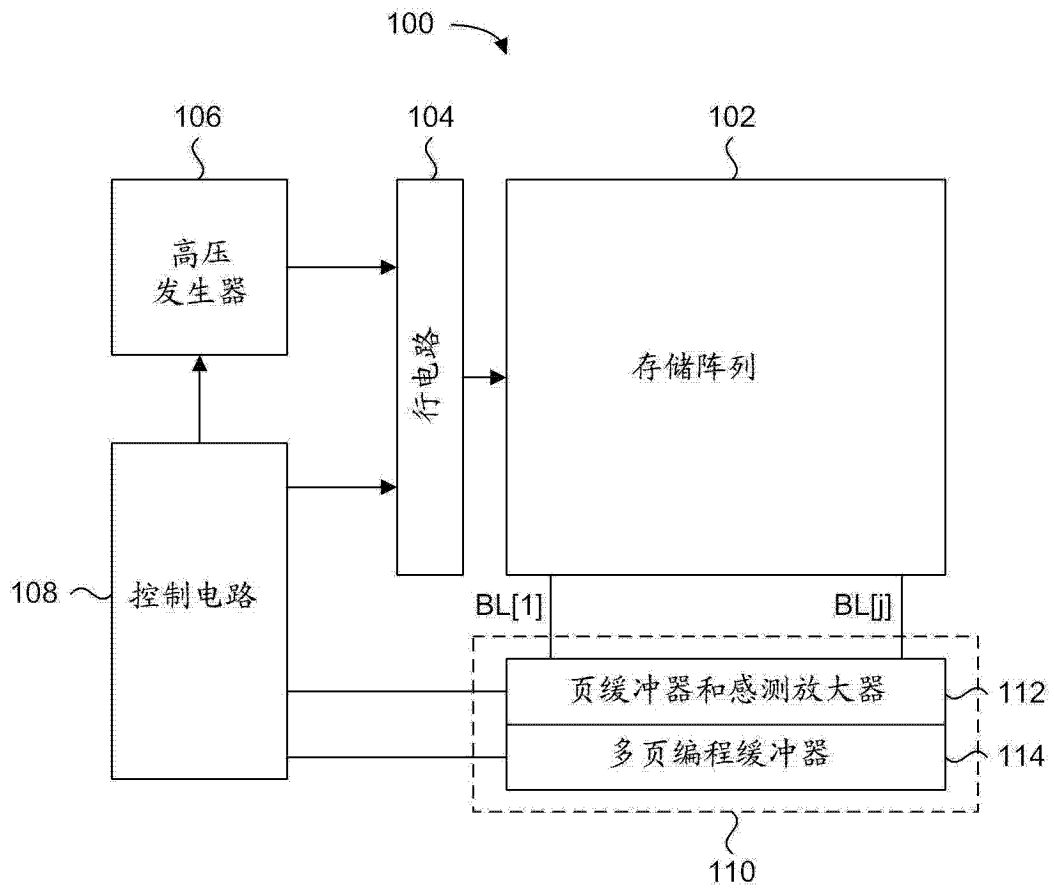


图 8

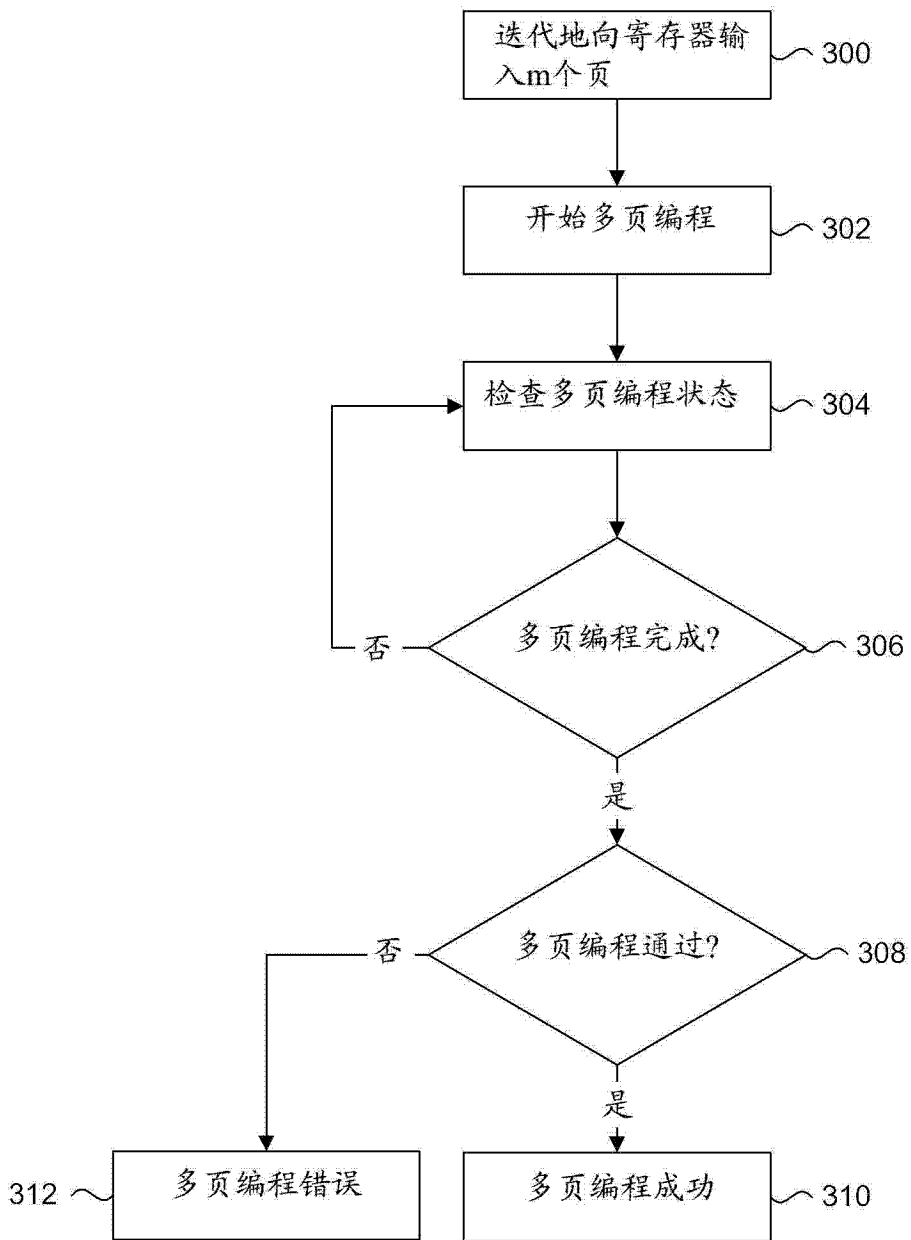


图 9

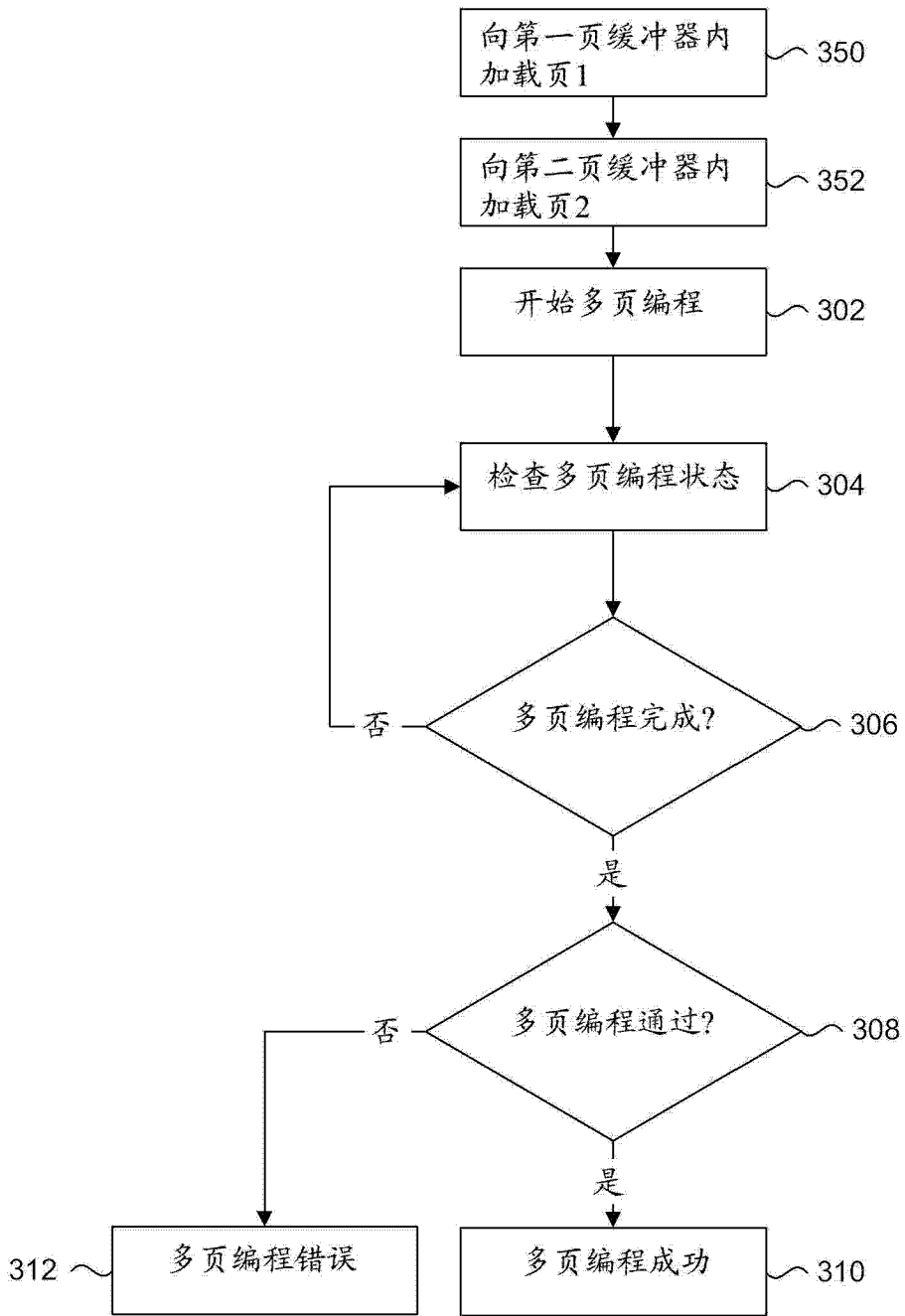


图 10

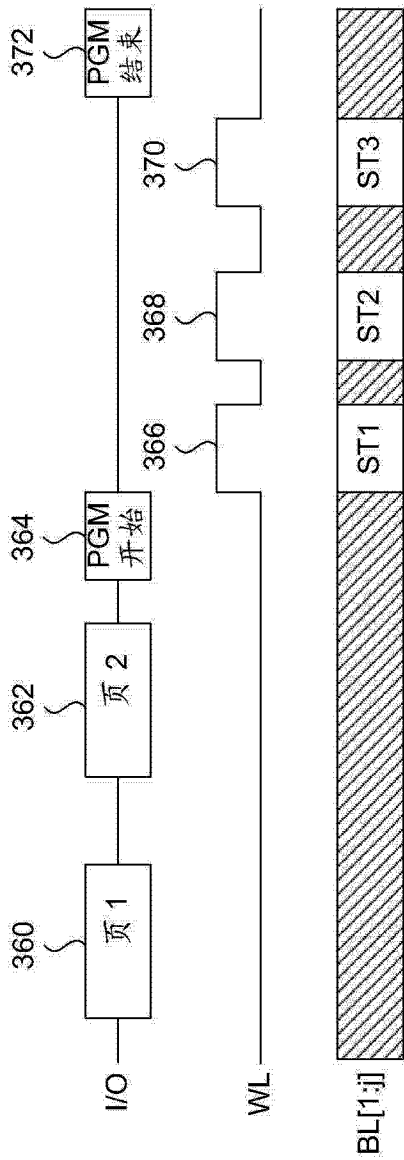


图 11

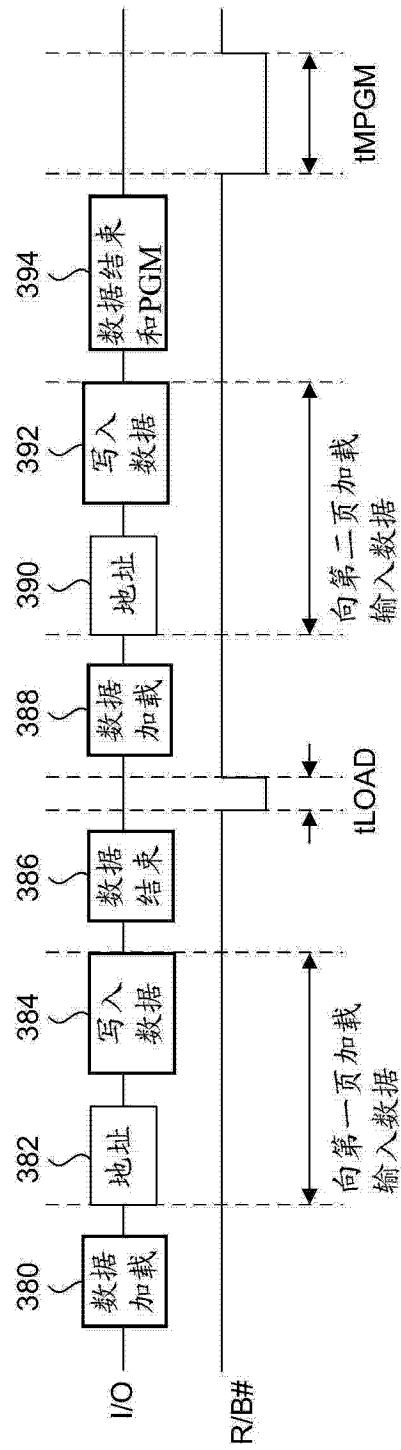


图 12A

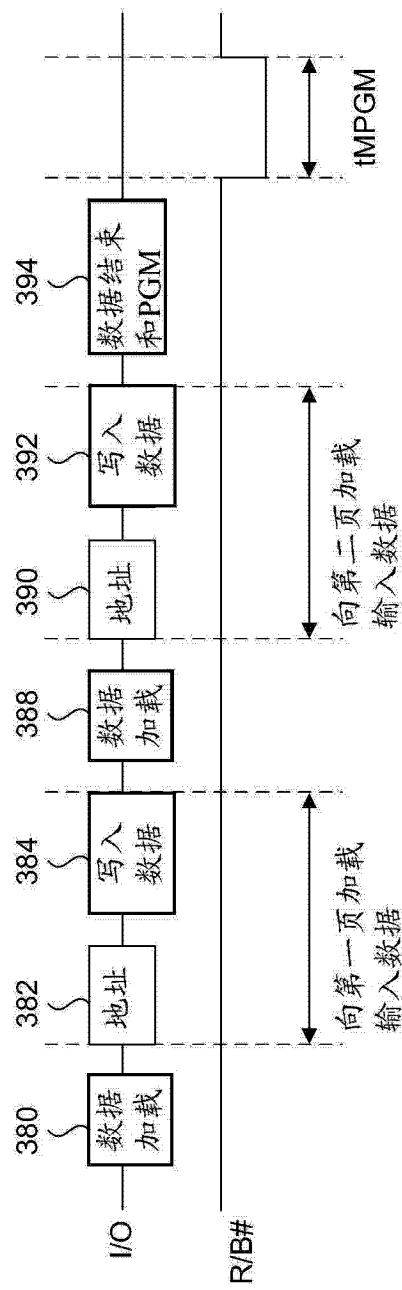


图 12B

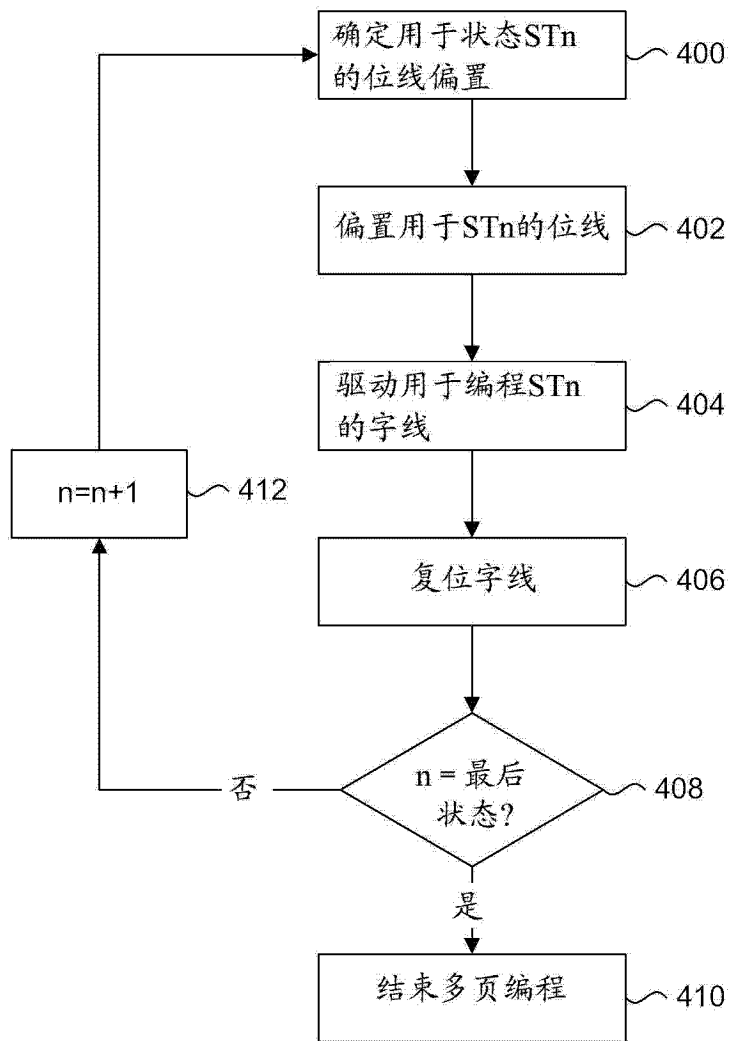


图 13

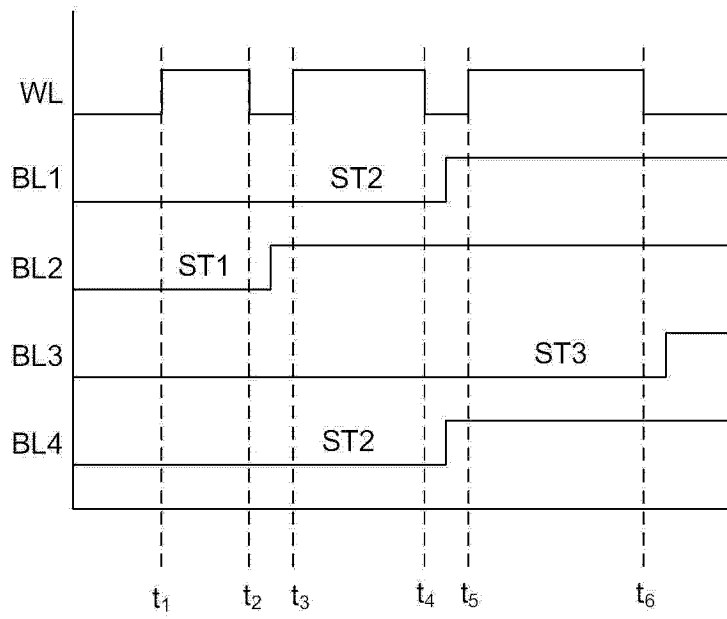


图 14A

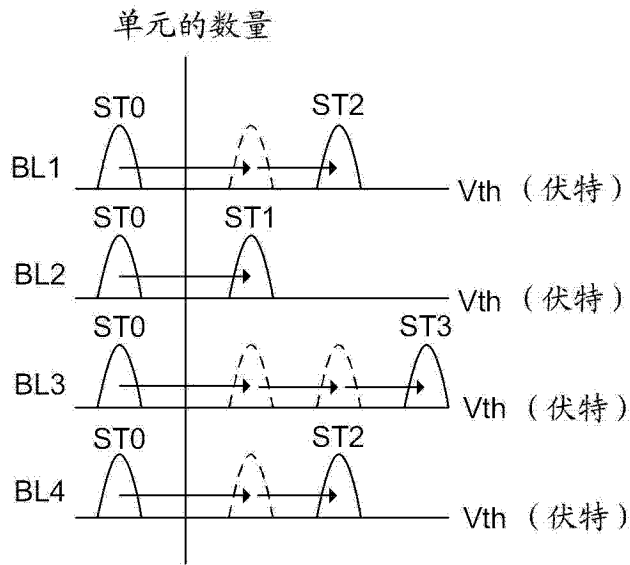


图 14B

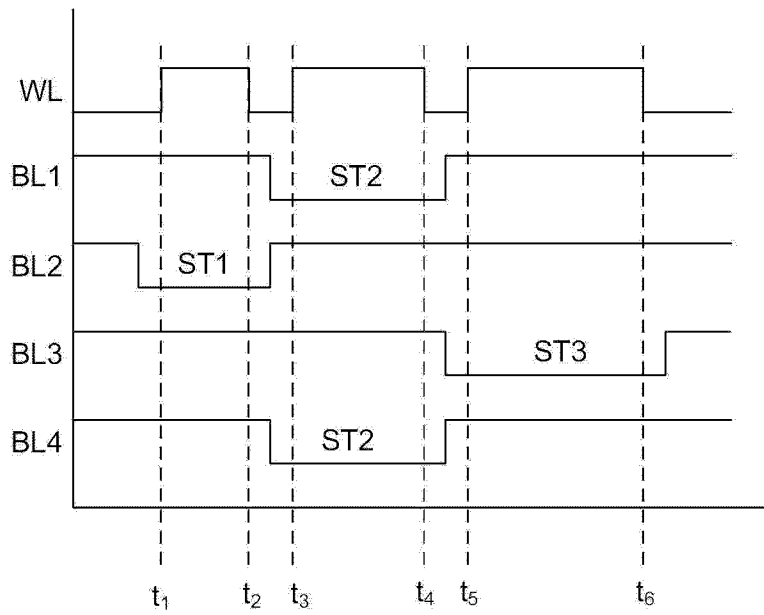


图 15A

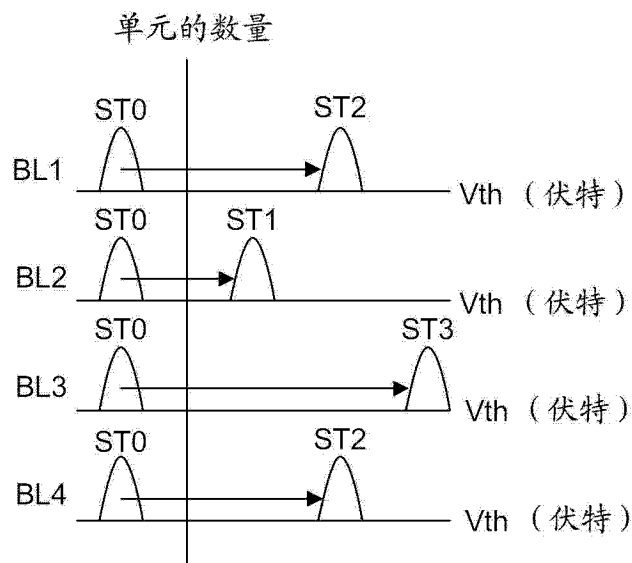


图 15B

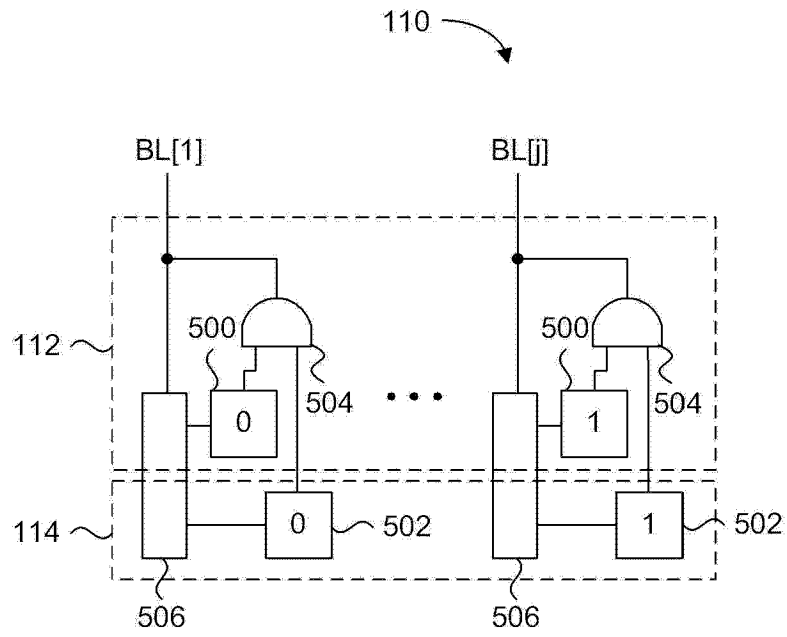


图 16

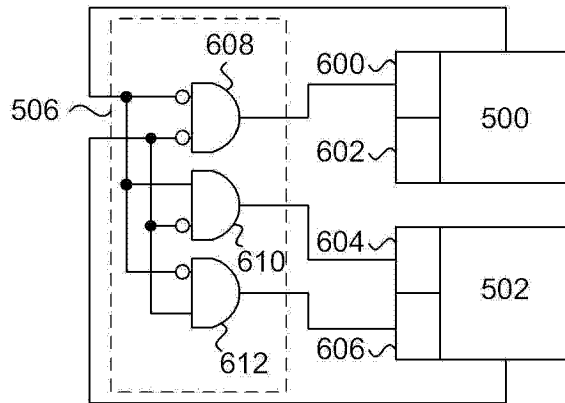


图 17

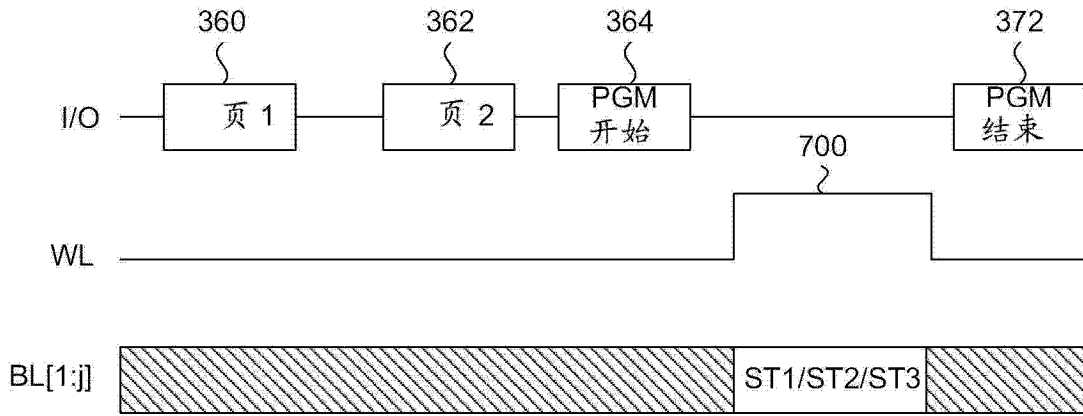


图 18A

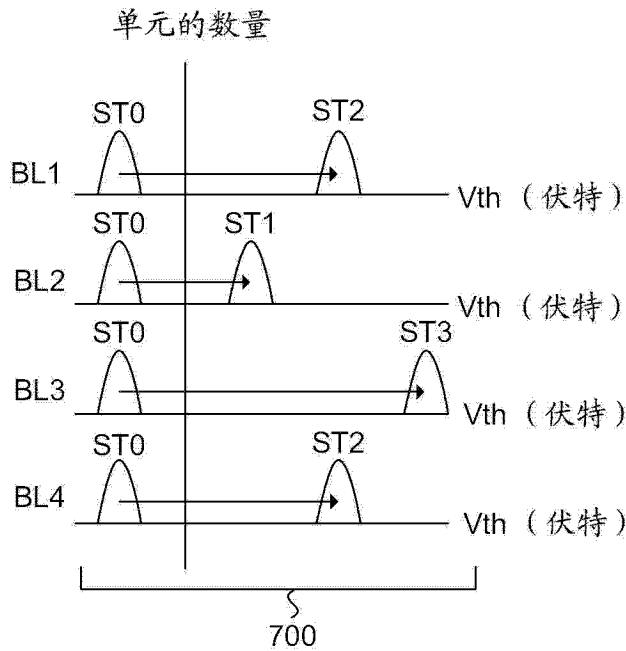


图 18B

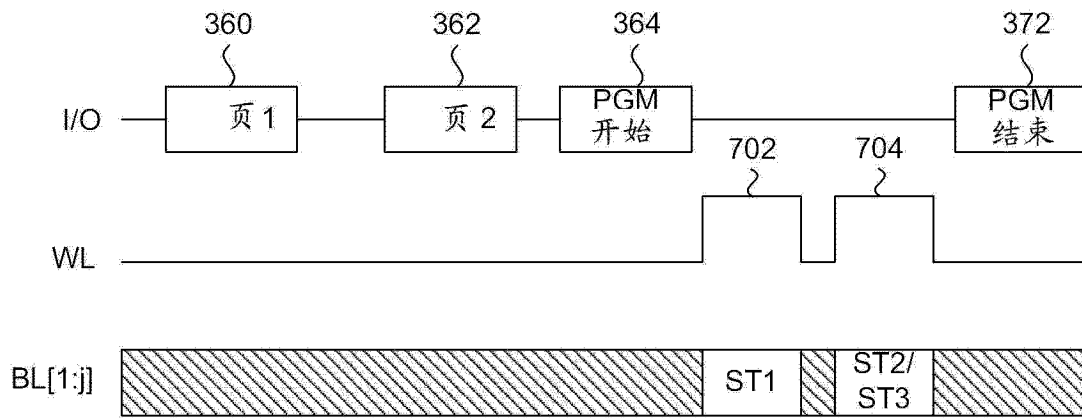


图 19A

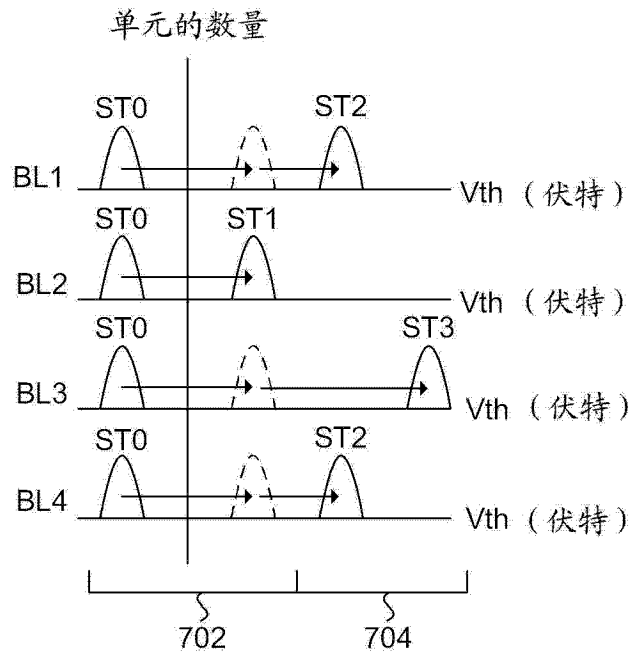


图 19B

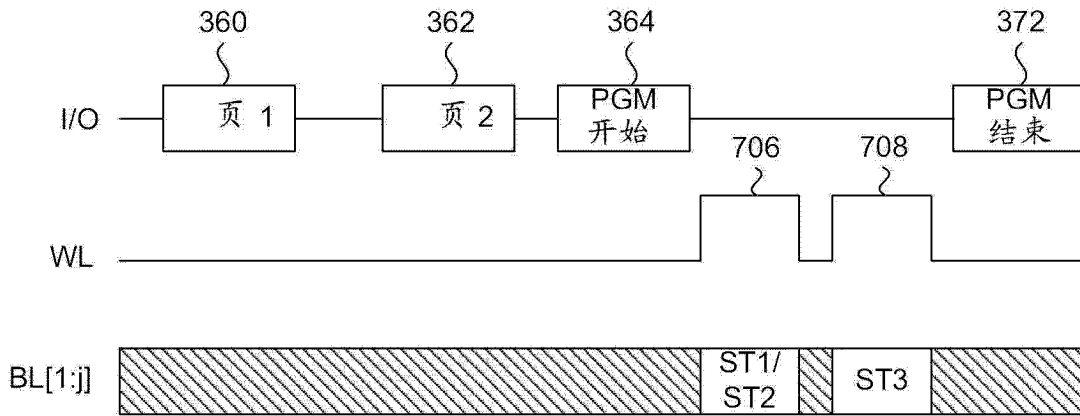


图 20A

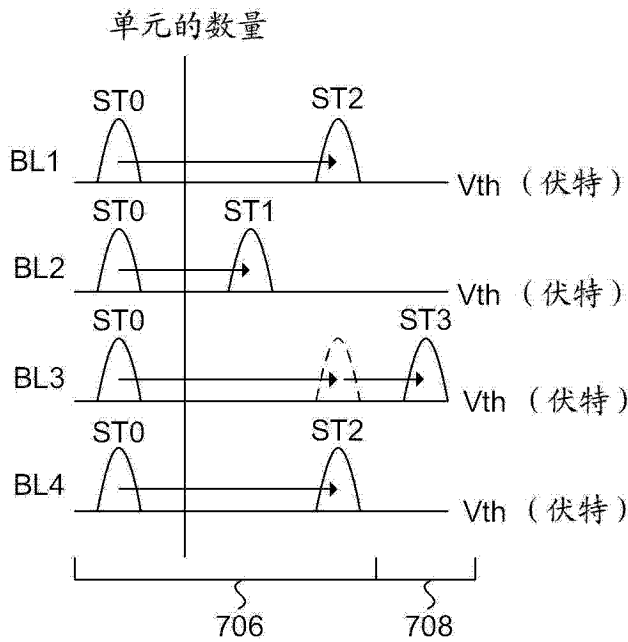


图 20B

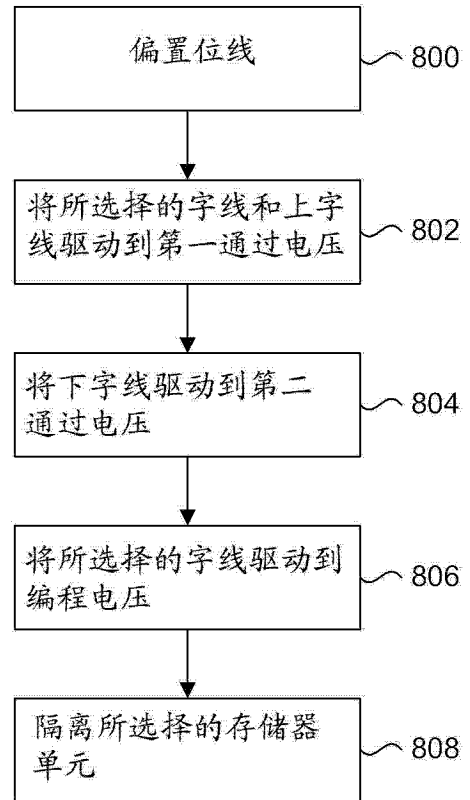


图 21

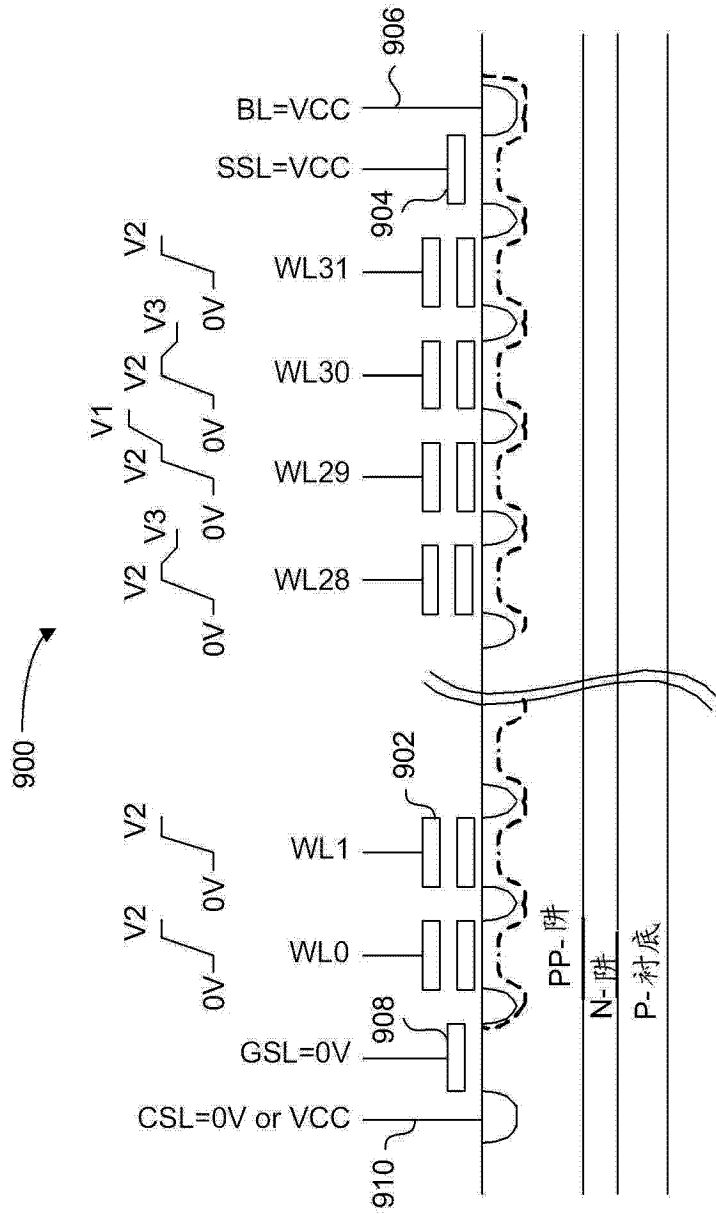


图 22A

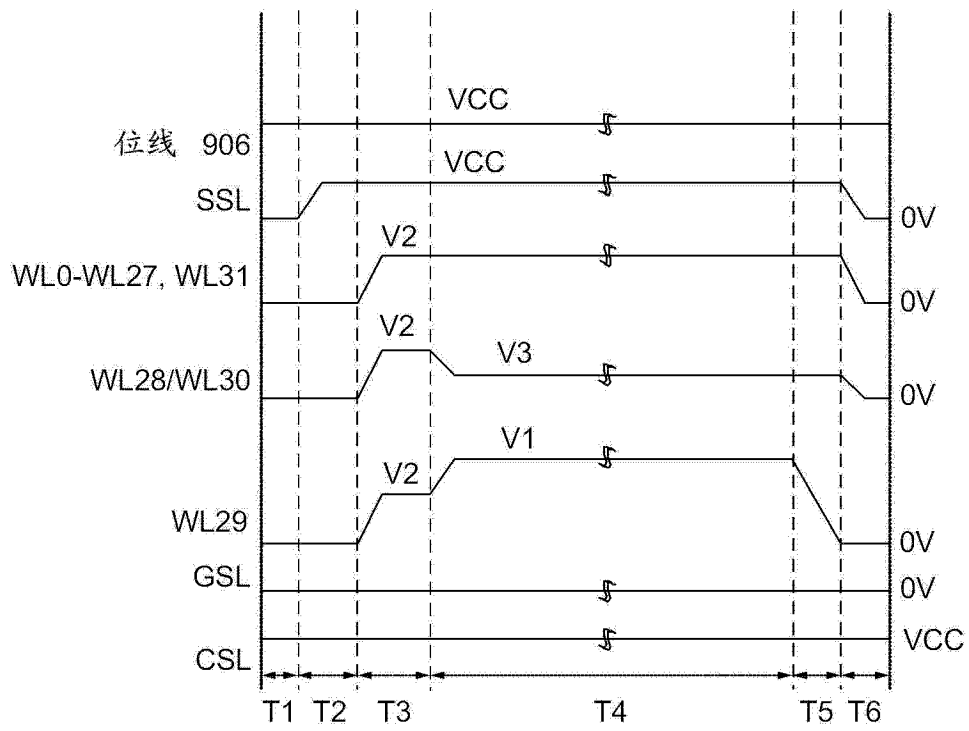


图 22B

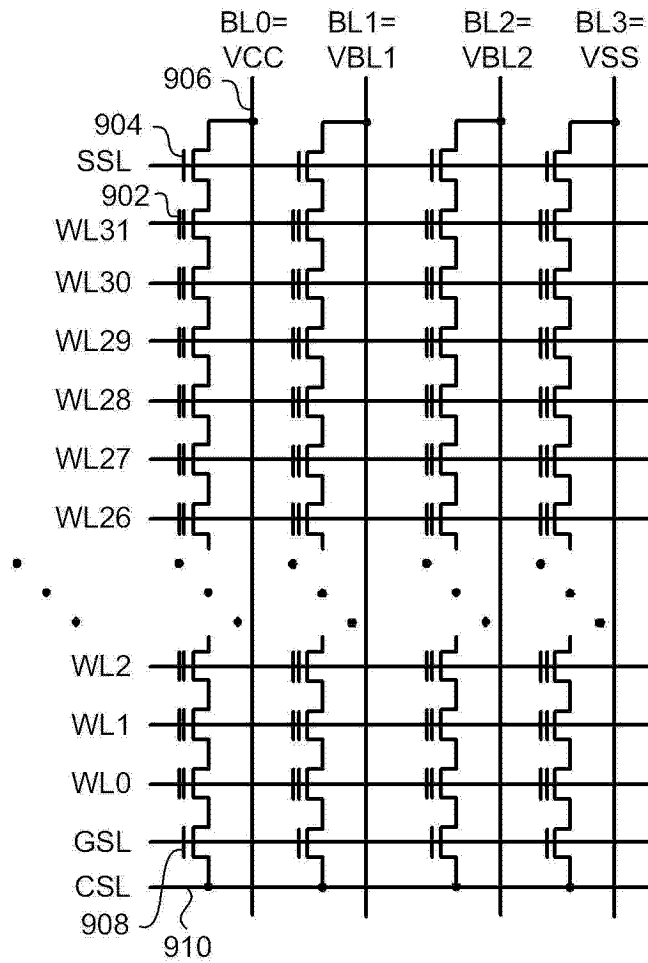


图 23

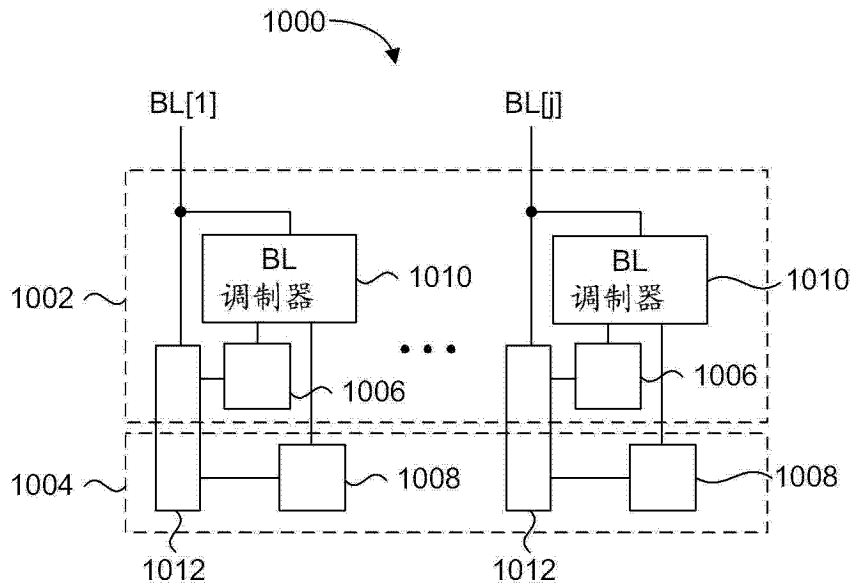


图 24

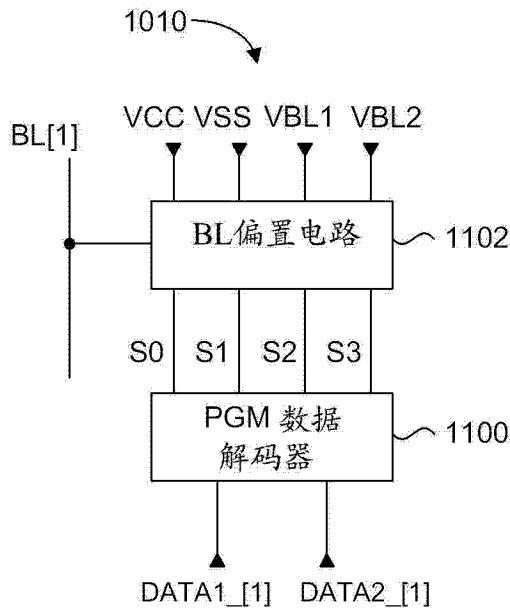


图 25

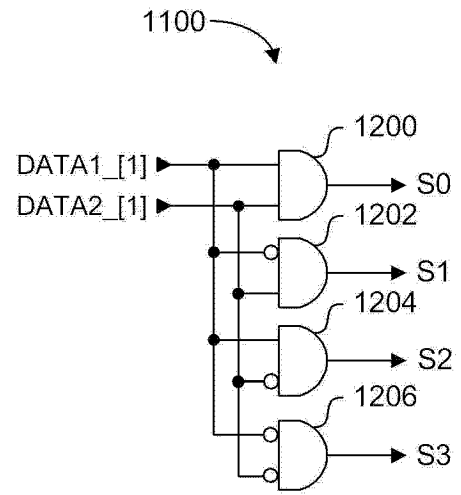


图 26

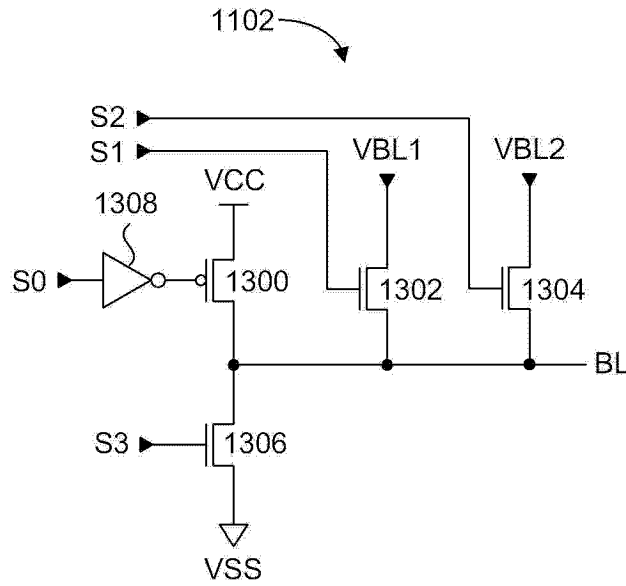


图 27

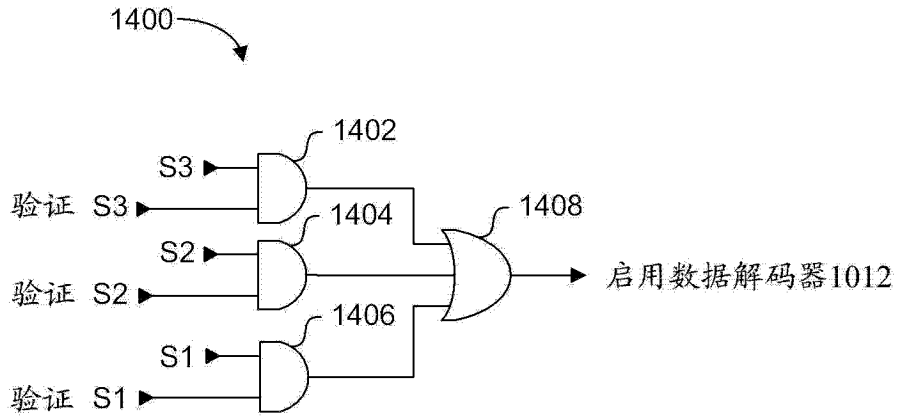


图 28

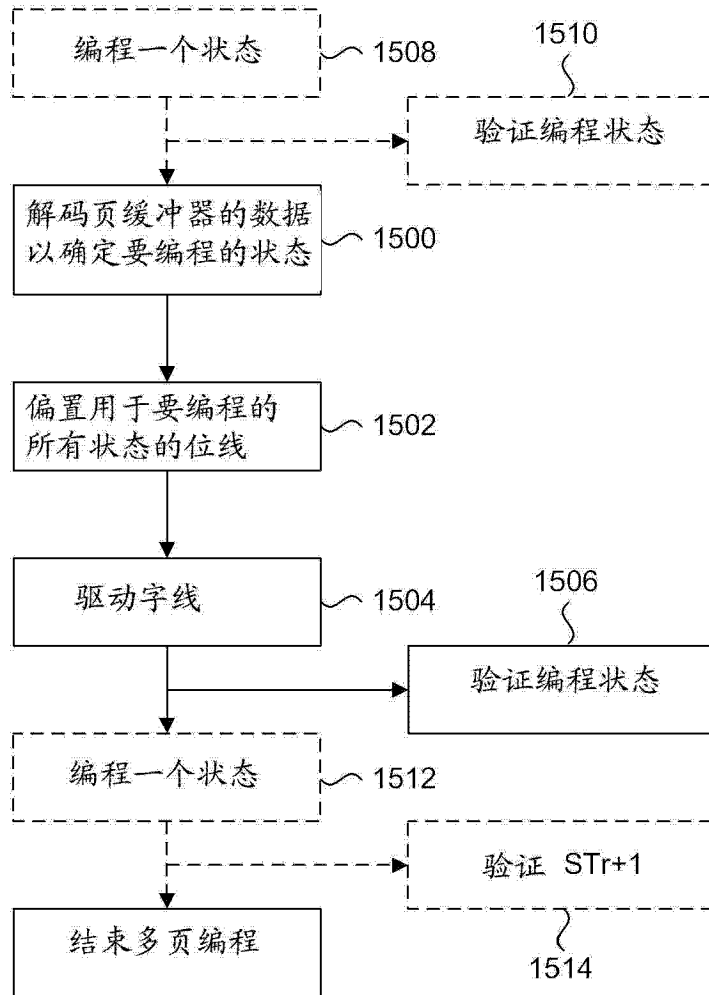


图 29