



(12)发明专利

(10)授权公告号 CN 103473146 B

(45)授权公告日 2017.04.19

(21)申请号 201310223394.2

(51)Int.Cl.

(22)申请日 2013.06.06

G06F 11/10(2006.01)

(65)同一申请的已公布的文献号

审查员 郭全萍

申请公布号 CN 103473146 A

(43)申请公布日 2013.12.25

(30)优先权数据

102117418 2013.05.16 TW

61/656,017 2012.06.06 US

(73)专利权人 慧荣科技股份有限公司

地址 中国台湾新竹县

(72)发明人 杨宗杰

(74)专利代理机构 深圳新创友知识产权代理有

限公司 44223

代理人 江耀纯

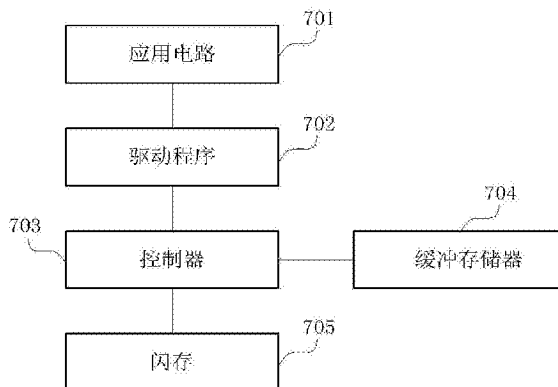
权利要求书3页 说明书11页 附图16页

(54)发明名称

存储器控制方法、存储器控制器以及电子装置

(57)摘要

本发明公开了一种存储器控制方法、存储器控制器和电子装置。所述存储器控制方法用于控制闪存。所述闪存包含第一存储元件与第二存储元件，所述第二存储元件包含多个区块，每一所述区块包含多个数据页。在这个方法中，将原始数据写入到所述第一存储元件。从所述第一存储元件读取所述原始数据得到输入数据，所述输入数据包含多个输入数据列。将所述多个输入数据列分成多个数据组。将每一数据组对应的每一输入数据列写到所述第二存储元件对应的一个数据页。将对每一数据组的一校验列写到第二存储元件对应的一个数据页，每一数据组对应的数据列的数目小于第二存储元件每一区块具有的数据页的数目。所述存储器控制方法可确保数据的正确性与稳定性。



1. 一种存储器控制方法,用于控制一闪存,所述闪存包含一第一存储元件与一第二存储元件,所述第二存储元件包含多个区块,每一所述区块包含多个数据页,所述存储器控制方法的特征在于包含:

将一原始数据写入到所述第一存储元件;

从所述第一存储元件读取所述原始数据得到一输入数据,所述输入数据包含多个输入数据列;

将所述多个输入数据列分成多个数据组;

将每一所述数据组对应的每一所述输入数据列写到所述第二存储元件对应的一个所述数据页;

将对应每一所述数据组的一校验列写到所述第二存储元件对应的一个所述数据页,其中,每一所述数据组对应的所述数据列的数目小于所述第二存储元件每一所述区块具有的所述数据页的数目;以及

每次在写入一个所述输入数据列到所述第二存储元件之前,利用所述一个所述输入数据列的内容累积计算更新所属所述数据组所对应的所述校验列的内容。

2. 如权利要求1所述的存储器控制方法,其特征在于,在对于所述多个输入数据列分成所述多个数据组时,每一所述数据组所对应的所述输入数据列的所述数目是参考所述第二存储元件的一错误发生率而决定。

3. 如权利要求2所述的存储器控制方法,其特征在于,当所述第二存储元件的所述错误发生率越高,每一所述数据组所对应的所述输入数据列数目越少。

4. 如权利要求1所述的存储器控制方法,其特征在于,至少两个所述数据组所对应的所述输入数据列的所述数目不相同,所述至少两个所述数据组存放于不同的所述第二存储元件的两个不同的所述区块。

5. 如权利要求1所述的存储器控制方法,其特征在于,至少两个所述数据组所对应的所述输入数据列的所述数目不相同,所述至少两个所述数据组存放于不同的所述第二存储元件的同一所述区块。

6. 如权利要求1所述的存储器控制方法,其特征在于,还包含提供一设定机制,供设定每一所述数据组所对应的所述输入数据列数目。

7. 如权利要求1所述的存储器控制方法,其特征在于,还包含根据侦测所述第二存储元件在运作过程中的一错误发生率,动态调整每一所述数据组所包含的所述输入数据列数目。

8. 如权利要求6所述的存储器控制方法,其特征在于,还包含计算所述一个所述输入数据列所对应的一列校验码,并且将所述列校验码连同所述一个所述输入数据列写入到所述第二存储元件。

9. 如权利要求1所述的存储器控制方法,其特征在于,还包含在从所述第一存储元件读取所述原始数据时,对所述原始数据执行错误更正以获得所述多个输入数据列。

10. 如权利要求1所述的存储器控制方法,其特征在于,所述第一存储元件比所述第二存储元件发生错误率较低。

11. 如权利要求1所述的存储器控制方法,其特征在于,所述第一存储元件为一单层单元闪存,且所述第二存储元件为一多层单元闪存。

12. 一种存储器控制器,用于控制一闪存,所述闪存包含一第一存储元件与一第二存储元件,所述第二存储元件包含多个区块,每一区块包含多个数据页,所述存储器控制器的特征在于包含:

一第一存储元件写入器,将一原始数据写入到所述第一存储元件;

一读取器,从所述第一存储元件读出所述原始数据并且产生一输入数据,所述输入数据包含多个输入数据列;以及

一第二存储元件写入器,将所述多个输入数据列分成多个数据组,将每一所述数据组对应的所述多个输入数据列写到所述第二存储元件对应的所述多个数据页,以及将对应每一所述数据组对应的一校验列写到所述第二存储元件对应的所述数据页,其中,每一所述数据组对应的所述多个输入数据列的数目小于所述第二存储元件每一所述区块具有的所述数据页的数目。

13. 如权利要求12所述的存储器控制器,其特征在于,还包含一设定器,供设定每一所述数据组所包含的所述输入数据列的数目。

14. 如权利要求12所述的存储器控制器,其特征在于,所述第二存储元件写入器还包含根据侦测所述第二存储元件在运作过程中的错误发生率,动态调整每一所述数据组所包含的所述输入数据列数目。

15. 如权利要求12所述的存储器控制器,其特征在于,还包含一列校验码产生器,每次在写入一个所述输入数据列到所述第二存储元件前,所述列校验码产生器利用所述一个所述输入数据列的内容累积更新所述一个所述输入数据列所属所述数据组所对应的所述校验列的内容。

16. 一种存储器装置,其特征在于,包含:

一闪存,包含一第一存储元件与一第二存储元件,所述第二存储元件包含多个区块,每一区块包含多个数据页;以及

一存储器控制器,用于控制所述闪存,所述存储器控制器包含:

一第一存储元件写入器,将一输入数据写入到所述第一存储元件;

一读取器,从所述第一存储元件读出所述输入数据,所述输入数据包含多个输入数据列;以及

一第二存储元件写入器,将所述多个输入数据列分成多个数据组,将每一数据组对应的所述多个输入数据列写到所述第二存储元件对应的所述多个数据页,以及将对应每一数据组的一校验列写到所述第二存储元件对应的所述数据页,其中,每一所述数据组对应的所述多个输入数据列数目小于所述第二存储元件每一所述区块具有的数据页数目。

17. 如权利要求16所述的存储器装置,其特征在于,所述第一存储元件为一单层单元闪存,且所述第二存储元件为一多层单元闪存。

18. 一种存储器控制方法,用于控制一闪存,所述闪存包含一存储元件,所述存储元件包含多个区块,每一区块包含多个数据页,所述存储器控制方法的其特征在于包含:

读取一数据组设定,所述数据组设定指示一个数据组包含的多个数据列的一数目,每一数据列存储于对应的一个所述数据页,并且所述数据组的所述多个数据列的数目少于每一所述区块所拥有的多个数据页个数;

根据所述数据组设定,从所述存储元件读取对应一个所述数据组的所述多个数据列以

及一校验列;以及

利用所述校验列对读取的所述多个数据列执行错误侦测及更正。

19. 如权利要求18所述的存储器控制方法,其特征在于,还包含:

对于所述数据组的每一所述数据列,利用所述数据列的一列校验码判断是否发生错误;以及

当利用所述列校验码发现错误且错误可更正时,利用所述列校验码更正所述数据列。

20. 如权利要求19所述的存储器控制方法,其特征在于,还包含:

当利用所述列校验码发现错误且错误不可更正时,记录无法通过所述列校验码更正的所述数据列;

针对所述数据组对应的各行,利用校验列计算一对应校正子;以及

假如所述多个对应校正子配合所述多个列检验码找出无法更正错误的所述数据列的记录足以指出错误发生的位置,利用所述校验列对所述数据组执行错误更正。

21. 如权利要求20所述的存储器控制方法,其特征在于,还包含:

在利用所述校验列执行错误更正之后,利用错误更正后的数据重新计算所述数据组中的所述多个列校验码,以再次执行错误侦测与更正。

存储器控制方法、存储器控制器以及电子装置

技术领域

[0001] 本发明涉及一种闪存的错误处理装置与方法,特别涉及一种多维度闪存的错误处理装置与方法。

背景技术

[0002] 闪存等非易失性存储器近年来快速发展,并且出现在各式各样的电子装置。目前看来,接下来不管是容量或是技术的发展,还会继续加强。一旦越来越多的数据被放在这类的非易失性存储器,为了确保储存数据的正确性与安全性,提供有效而且可靠的错误侦测与错误管理也就越来越重要。

发明内容

[0003] 本发明的一实施例公开了一种存储器控制方法,用于控制闪存。所述闪存包含第一存储元件与第二存储元件,所述第二存储元件包含多个区块,每一所述区块包含多个数据页。在这个存储器控制方法中,首先,将一原始数据写入到所述第一存储元件。并且,从所述第一存储元件读取所述原始数据得到一输入数据,所述输入数据包含多个输入数据列。此外,将所述多个输入数据列分成多个数据组。将每一所述数据组对应的每一所述输入数据列写到所述第二存储元件对应的一个所述数据页。并且,将对应每一所述数据组的一校验列写到所述第二存储元件对应的一个所述数据页,其中,每一所述数据组对应的所述数据列的数目小于所述第二存储元件每一所述区块具有的所述数据页的数目。

[0004] 本发明的另一实施例公开了一种存储器控制器,用于控制一闪存。所述闪存包含一第一存储元件与一第二存储元件,所述第二存储元件包含多个区块,每一区块包含多个数据页,所述存储器控制器包含第一存储元件写入器、读取器跟第二存储元件写入器。第一存储元件写入器将一原始数据写入到所述第一存储元件。读取器从所述第一存储元件读出所述原始数据并产生一输入数据,所述输入数据包含多个输入数据列。

[0005] 第二存储元件写入器将所述多个输入数据列分成多个数据组,将每一所述数据组对应的所述多个输入数据列写到所述第二存储元件对应的所述多个数据页,以及将对应每一所述数据组对应的一校验列写到所述第二存储元件对应的所述数据页,其中,每一所述数据组对应的所述多个输入数据列的数目小于所述第二存储元件每一所述区块具有的所述数据页的数目。

[0006] 本发明的另一实施例公开了一种电子装置,具有闪存跟控制器。闪存包含一第一存储元件与一第二存储元件,所述第二存储元件包含多个区块,每一区块包含多个数据页。

[0007] 存储器控制器,用于控制所述闪存。所述存储器控制器具有第一存储元件写入器、一读取器,以及第二存储元件写入器。第一存储元件写入器,将一输入数据写入到所述第一存储元件。读取器从所述第一存储元件读出所述待输入数据,所述输入数据包含多个输入数据列。第二存储元件写入器将所述多个输入数据列分成多个数据组,将每一数据组对应的所述多个输入数据列写到所述第二存储元件对应的所述多个数据页,以及将对应每一数

据组的一校验列写到所述第二存储元件对应的所述数据页,其中,每一所述数据组对应的所述多个输入数据列数目小于所述第二存储元件每一所述区块具有的数据页数目。

[0008] 本发明的另一实施例公开了一种存储器控制方法,用于控制一闪存。所述闪存包含一存储元件,所述存储元件包含多个区块,每一区块包含多个数据页。所述存储器控制方法包含下列步骤。

[0009] 读取一数据组设定,所述数据组设定指示一个数据组包含的多个数据列的一数目,每一数据列存储于对应的一个所述数据页,并且所述数据组的所述多个输入数据列的数目少于每一所述区块所拥有的多个数据页个数。根据所述资料组设定,从所述存储元件读取对应一个所述数据组的所述多个数据列以及一校验列。此外,利用所述校验列对读取的所述多个数据列执行错误侦测及更正。

附图说明

- [0010] 图1绘示了一个与非门结构闪存的存储元件的一个区块。
- [0011] 图2是一个三层单元的闪存存储单元存放电量和操作电压的示意图。
- [0012] 图3绘示了对存储单元施加操作电压所可能发生的情形。
- [0013] 图4绘示了在一次的读取操作时,依序使用七个不同的电压对存储单元执行读取操作。
- [0014] 图5绘示了用来找中间有效位的方法。
- [0015] 图6绘示了用来找最高有效位的方法。
- [0016] 图7绘示了一个使用闪存的电子装置的范例。
- [0017] 图8绘示了一个具有阶层式存储元件结构的闪存结构。
- [0018] 图9绘示了在多层单元闪存内的数据存放格式。
- [0019] 图10绘示了第二种配对数据跟校验数据的方法。
- [0020] 图11绘示了第三种配对数据跟校验码的方法。
- [0021] 图12是图8之中的数据从单层存储单元读出之后写到多层存储单元的流程图。
- [0022] 图13是图8之中的数据从单层存储单元读出之后写到多层存储单元的示意图。
- [0023] 图14是图8之中的数据从单层存储单元读出之后写到多层存储单元的示意图。
- [0024] 图15是图8之中的数据从单层存储单元读出之后写到多层存储单元的示意图。
- [0025] 图16是图8之中的数据从单层存储单元读出之后写到多层存储单元的示意图。
- [0026] 图17绘示了一种存储器控制方法的一实施例的流程图。
- [0027] 图18绘示了闪存模块一个区块的逻辑分配示意图。
- [0028] 图19是闪存模块除错的示意图。
- [0029] 图20是闪存模块除错的示意图。
- [0030] 其中,附图标记说明如下:
- | | | |
|--------|--------|-------|
| [0031] | 701 | 应用电路 |
| [0032] | 702 | 驱动程序 |
| [0033] | 703 | 控制器 |
| [0034] | 704 | 缓冲存储器 |
| [0035] | 705、80 | 闪存 |

[0036]	804	存储器接口电路
[0037]	806、SLC	单层单元闪存模块
[0038]	807、808、MLC	多层单元闪存模块
[0039]	82	控制器
[0040]	820	第一存储元件写入器
[0041]	822	读取器
[0042]	824	第二存储元件写入器
[0043]	M_0、M_1、M_K	存储单元
[0044]	P_0、P_1、P_2、P_N	数据页
[0045]	VG_0、VG_1、VG_2、VG_N、	电压
[0046]	VT_0、VT_1、VT_2、VT_3、	
[0047]	VT_4、VT_5、VT_6、VT_7、V _{LSB} 、	
[0048]	D、V _{CSB1} 、V _{CSB2} 、V _{MSB1} 、V _{MSB2} 、	
[0049]	V _{MSB3} 、V _{MSB4}	
[0050]	L0、L1、L2、L3、L4、L5、L6、L7	区间

具体实施方式

[0051] 本发明的实施例包括一种存储器控制方法,通过软件、硬件或软件加硬件的方式实作,用来控制闪存。在这个存储器控制方法的实施例中,被控制的闪存具有一个或多个的第一类存储元件跟一个或多个的第二类存储元件,并且也是闪存模块,例如以裸芯的方式封装在一起,或是以芯片的方式通过连接线连在一起或其它制作方式产生。第一类存储元件相对于第二类存储元件相对稳定,出错率较低。在原始数据要写入闪存时,先将原始数据写入第一类存储元件。然后,在适当的时间,原始数据再度从第一类存储元件读出,做为写入第二类存储元件的输入数据。

[0052] 第二类存储元件具有多个区块(block),每个区块包括一定数目的数据页(page)。上述的输入数据被分成多个数据组(group),每个数据组具有一定数目的输入数据列(row)。数据组所具有输入数据列的数目小于第二类存储元件每个区块所能承载的数据页数目。也就是说,第二类存储元件的每个区块可存放多个数据组的输入数据列。

[0053] 如上所述,由于第二类存储元件的稳定性可能较差,因此,对要存放到第二类存储元件的每个数据组,可通过低密度校验码(low density parity-check,LDPC)、博斯-乔赫里-霍克文黑姆(Bose-Chaudhuri-Hocquenghem,BCH)等已知的校验码(parity)产生方法来计算对应每个数据组的校验列(parity row)。除此之外,每个输入资料列本身也可以包含对应所述输入资料列的列校验码(row parity code)。

[0054] 相对于以整个区块作为单位,采用上述数据组的方法可减少写入过程断电或其它原因造成写入中断的风险。

[0055] 此外,每个数据组对应的输入数据列数目可设定跟调整,并且每个数据组可设定对应一个或多个校验列。这样的设计可针对第二存储元件的特性来作调整。

[0056] 此外,在不同的区块中,数据组可设定为包含不同数目的输入数据列。事实上,在同一区块中,不同的数据组也可以有不同数目的输入数据列。

[0057] 举例来说,第一类存储元件可以是单层单元闪存(Single Level Cell Flash Memory,SLC Flash Memory),也就是每个存储单元存放的电位代表一个比特,第二类存储元件可为多层单元闪存(Multiple Layer Cell Flash Memory,MLC Flash Memory),例如三层单元闪存(Triple Layer Cell Flash Memory,TLC Flash Memory),在一个存储单元可存放代表多个比特的信息,例如一个存储单元可存放三个比特的信息。在这样的配置下,随着半导体工艺持续往高密度度微小化发展,加上对于成本下降的需求,每个存储单元的稳定性有越来越大的问题。

[0058] 通过上述的设计概念,可以针对不同稳定性的闪存,使用同一套存储器控制电路的设计。例如,在确定在某个设计下,会使用到稳定性较低的第二类存储元件,可增加每个数据组所使用校验列的数目,减少每个数据组所对应的输入数据列数目等,或增加列校验码的占全体数据的比例等。

[0059] 相反的,假如第二类存储元件的错误率相对较少,则可设定增加每个数据组所包含的输入数据列的数目,或是减少校验列与列校验码占全体数据的比例。

[0060] 除了出厂时可进行上述设置外,也可在控制电路中设定动态侦测的机制。假如发现第二类存储元件的稳定性在产品使用的过程中逐渐变差,可动态调整每个数据组对应的输入数据列的比例。因此,即使电子装置用久了,数据的稳定性也不会有急速的降低。

[0061] 必须说明的是,虽然这边举例的第一类存储元件与第二类存储元件为闪存内的电路,但本领域的技术人员通过本说明的介绍,应该可以将同一个概念用在闪存以外的存储装置。

[0062] 本发明的实施例还包括如上说明的存储器控制器,以及配备此存储器控制器与闪存的电子装置。

[0063] 此外,本发明的实施例还包括存储器控制方法,用来读取闪存的数据。其读取数据组设定,以得知闪存每个区块包含的数据组设定方式,例如每个数据组包含多少数目的输入数据列以及多少数目的校验列。

[0064] 在得知闪存的数据组设定之后,以每个数据组作为单位,将数据组的数据列依次读出。在读出数据列之后,使用数据列的列校验码对数据列执行错误的侦测动作。假如发现通过列校验码即可更正错误,则对数据列执行错误更正。

[0065] 假如发现通过列校验码还是无法更正数据,例如列校验码能承受的误差只有两个比特,而实际数据有三个以上的错误时,则继续通过数据组的校验列执行错误的侦测与更正。反之,假如无法通过列校验码执行错误的侦测与更正,则记录下到底是哪几个数据列属于无法更正的情况。

[0066] 此外,如上所述,每一数据组具有多个数据列,这些数据列按序排列时,构成具有列与行的阵列(array)。对于每一行数据,有对应的校验列的校验数据(parity data)。假如使用的是低密度校验码(LDPC)或博斯-乔赫里-霍克文黑姆(BCH)等错误侦测与更正方法来产生校验列,此时可通过校验列跟数据组的数据列进行计算,而得出每行对应的校正子(syndrome)。在这些方法中,校正子可以指出,有可能是那些数据列出现错误。

[0067] 假如校正子指出的出错数据列跟前述记录下来无法更正且出现错误的的数据列记录吻合,则推测校正子指出的错误是可靠的。此时,通过校验列的校验数据对数据列的错误内容进行更正。

[0068] 随着上述方法的逐次进行,可进一步减少错误。此时,可再次执行列校验码的计算。原先可能因为一个数据列有超过三个比特出错而无法更正的情况,由于这三个比特已经更正一个比特,让错误降低到两个比特,因此,已经可以用列校验码来更正错误。

[0069] 随着列校验码进一步减少错误比特,可再次执行上述校正子的计算。如果有需要,可执行多次(iteration)的动作,逐步找出错误,并且在一定的限制条件下,尽可能更正所有的错误,以确保数据的正确性与稳定性。

[0070] 这样的特性,对于保存使用者数据来说,是非常重要的。虽然为了降低成本,可能需要使用稳定性较低的第二存储单元,但通过上述方法,可尽量降低使用者宝贵数据丧失的风险。

[0071] 以下将进一步通过图示,更详细的说明关于本发明实施例的实际制作方法。

[0072] 请参考图1,其绘示了一个与非门(NAND)结构闪存的一个存储元件的一个区块(block)。在这个区块中具有一定数目的数据页(page),即P₀、P₁、P₂到P_N。每一个数据页具有存储单元(cell)M₀、M₁到M_K。通过对每一数据页设定适当的电压VG₀、VG₁、VG₂到VG_N,可读取存储于每个存储单元的浮置栅极(Floating Gate)的电位,进而得到每个存储单元所存放的数据。

[0073] 对于单层单元(SLC)闪存来说,每个存储单元只存放一个比特的数据,也就是0或1。此时,理论上,只要给每一数据页适当的一个设定电压VG₀、VG₁、VG₂到VG_N,就可以侦测出存储单元存放的电量,而得出对应的数据值。

[0074] 相对的,假如是多层单元(MLC)闪存,针对一次的读取动作就需要施加多个不同的设定电压,以判断存储单元内存放的电量到底是多少,而换算出实际存放的数据内容。

[0075] 图2是一个三层单元(TLC)的闪存存储单元存放电量和操作电压的示意图。在这个示意图中可看到,一个存储单元根据储存的电量落在L₀、L₁、L₂...L₇区间,而代表存放的比特数据分别为111,011,001,...110。

[0076] 对于这样的存储单元,理论上,在施加VT₁的操作电压时,侦测电路可以判断到底存储单元存放的电量是属于L₀这一边(也就是数据111)还是L₁、L₂、L₃、L₄、L₅、L₆、L₇那一边(也就是数据为111,011,001,101,100,000或110)。

[0077] 通过有次序的施加多个不同的电压组合,理论上就可以判断出所有三个比特的数据,也就是最高有效位(Most Significant Bit,MSB)、中间有效位(Central Significant Bit,CSB)与最低有效位(Least Significant Bit,LSB)的数据内容。

[0078] 但是,如上所述,随着半导体工艺持续朝密集化以及微小化的方向发展,以及闪存为了降低成本或是使用时间越来越长,相关的电路以及存储器存储单元的稳定性问题越来越大。

[0079] 图3绘示了一种可能的状况,也就是如果对存储单元施加操作电压VT₁,有可能因为比特状态间出现部分重迭或甚至位移,导致解读出的数据发生不正确的情形。在这样的情况下,就需要通过各种不同错误校验方法,或是动态调整操作电压来解决数据判断不准确的问题。

[0080] 图4绘示了在一次的读取操作时,依序使用七个不同的电压对存储单元执行读取操作,侦测存储单元内浮置栅极的电量,以判读存储单元存放的数据的LSB数值到底是0还是1。

[0081] 从图4可以清楚看到,如果存储单元储存的电量分布落在 V_{LSB} 的左侧L0,L1,L2,L3,代表LSB的内容是0。相反的,如果是落在 V_{LSB} 的右侧L4,L5,L6,L7,则代表LSB的内容是1。

[0082] 由于在状态间有重迭的问题,因此,可依序施加不同的电压 $V_{LSB}, V_{LSB}+D, V_{LSB}-D, V_{LSB}+2D, V_{LSB}-2D, V_{LSB}+3D, V_{LSB}-3D$ 。因此,假如存储单元的电量分布刚好落在例如 $V_{LSB}+D$ 与 V_{LSB} 之间,就可以从侦测的结果得到一定的信息。

[0083] 每次施加一个电压可以得到一个比特结果,因此七次电压就可以得到七个比特。这七个比特总共有八种可能的组合。由这七个比特对应的二进制序列(bit sequence),可配合LDPC的译码电路跟方法,用来计算校验码以及用来找出正确的比特数据,也就是利用所取得的软信息(soft information)搭配LDPC与BCH等方法,用来执行错误校验。

[0084] 图5绘示了用来找CSB的方法。由于CSB代表第二个比特,在图5中可看到如果存储单元的电量是落在L2,L3,L4,L5则代表存储单元储存的CSB是0。另一方面,如果存储单元的电量落在L0,L1,L6,L7区间,则代表存储单元储存的CSB是1。在这样的配置下,可以理解的是需要使用 V_{CSB1} 与 V_{CSB2} 两个操作电压来过滤出到底存储单元的电量是落在哪个区间。

[0085] 相似于上述的说明, V_{CSB1} 与 V_{CSB2} 也可以施加多个步进调整量,依序多次用不同的电压执行读取的动作。每次读取的结果产生二进制序列,可搭配LDPC与BCH等方法,用来执行错误校验。

[0086] 图6绘示了用来找MSB的方法。由于MSB代表最高的比特,在图6中可看到如果存储单元的电量是落在L0,L3,L4,L7区间,则代表存储单元存放的MSB比特为1。相对的,假如存储单元的电量是落在L1,L2,L5,L6区间,则代表存储单元存放的MSB比特为0。

[0087] 相似于上述的说明, $V_{MSB1}, V_{MSB2}, V_{MSB3}, V_{MSB4}$ 也可以施加多个步进调整量,依序多次用不同的电压执行读取的动作。每次读取的结果产生二进制序列,可搭配LDPC与BCH等方法,用来执行错误校验。

[0088] 图7绘示了一个使用闪存的电子装置的范例。这个电子装置包括应用电路(Application Circuit)701、驱动程序702、控制器703、缓冲存储器704、闪存705。这样的电子装置可通过不同的硬件跟软件的设计,搭配额外的传感器与输出输入周边装置,而构成各种市面上的电子装置,例如笔记本电脑、照相机、摄像机、SSD硬盘、手机、优盘、多媒体播放器、电视机、机顶盒等等。

[0089] 根据不同的设计需求,应用电路701可以由一个或多个客制化集成电路芯片,或一般处理器、控制器,搭配对应的固件跟软件,配合电路板跟相关的组件,共同构成所需能达成特定应用的应用电路701。

[0090] 在这个例子中,应用电路701在需要存取闪存705时,通过驱动程序702的指令的执行,经由控制器703存取闪存705内部存放的数据。里提到的闪存705可以是上述的闪存结构或是其它的存储硬件。在一实施例中,可将应用电路701与驱动程序702视为一主机(host),但不以此为限。

[0091] 在需要将数据写入到闪存705的时候,控制器703负责协调跟闪存705的沟通,按照闪存705的结构将数据传送给闪存705。通常,控制器705会使用一些缓冲存储器704来做为缓冲或是高速缓存之用。这些缓冲存储器704在设计的时候可以跟控制器703封装在一起、做成同一个模块,或是设计成两个不同的区块。举例来说,有些闪存705每次存取需要有一定的数据量,例如一个区块或是一个数据页等,这时候要写入闪存705的资料就会先准备好

放在缓冲存储器704。此外,为了避免过度频繁使用闪存705,也可以使用不同的算法设计高速缓存机制,以加快数据的整体存取速度。

[0092] 应用电路701可以有許多不同的类型。例如手机或照相机内的电路板,摄像镜头,处理器等。应用电路701也可以是数字电视内的处理器或是计算机内的处理器。控制器可以制作成独立的集成电路芯片,或是嵌入到应用电路701内,也可能将相关控制流程写成程序代码,并且由通用控制器加以执行之后,产生对应控制功能的控制器。有时候,应用电路701跟控制器703之间的沟通,也可能先写成程序代码构成驱动程序,然后由应用电路701执行这个驱动程序,来完成跟控制器之间的沟通动作。

[0093] 到底哪些功能由控制器703执行,哪些功能由驱动程序702,或甚至应用电路701本身来执行,会根据系统跟设计需求的不同,而能够做不同的调整。举例来说,以下所描述跟错误管理有关的功能,可以由狭义的控制器703来执行,也可能部分或全部的错误管理机制由驱动程序702配合应用电路701来执行。此时,完成这些错误管理机制的各个环节,整体构成了广义的控制器概念。

[0094] 图8绘示了一个具有阶层式存储元件结构的闪存80结构。在这个例子中,闪存80包括存储器接口电路804、单层单元闪存模块(以SLC表示)806,以及多层单元闪存模块(以MLC表示)807与多层单元闪存模块(以MLC表示)808。在实际的设计上,可以配置更多数量的多层单元闪存。

[0095] 控制器82包括第一存储元件写入器820、读取器822与第二存储元件写入器824。控制器82用来控制闪存80。控制器82将控制指令跟数据信号传给闪存80的存储器接口电路804,根据不同的设计需求,闪存804的存储器接口电路804可以只是单纯执行控制器82的指令,也可以跟控制器82共同分担以下的数据存取与错误管理的处理工作。

[0096] 单层单元闪存跟多层单元闪存各有不同的优缺点。简单来说,单层单元闪存由于构造单纯,存取速度较快而且数据比较不容易出错,但多层单元闪存由于每个存储单元可以存放两个以上的比特,所以单位成本会比单层单元闪存便宜很多,但其电路却也较为复杂。

[0097] 在图8的结构之中,单层单元闪存可以作为多层单元闪存的一个缓冲机制。也就是说,数据如果要写入闪存80,可以先指定写到单层单元闪存之中。例如使用者正在摄像或录音,因为有时间的限制,数据先写入到单层单元闪存。

[0098] 接着,控制器82利用空档,或是指示闪存80的存储器接口电路804同步或流水线(pipeline)方式,将单层单元闪存的数据写到对应的多层单元闪存。

[0099] 在图8的例子中,单层单元闪存只画了一个,而多层单元闪存只画了两个,但是在实际设计时,单层单元闪存跟多层单元闪存的数量都可以根据实际需求加以调整,而且,单层单元闪存也不一定只能专门作为缓冲高速缓存,本身也可根据需求作为数据长期储存使用。以下的错误管理机制,也可以根据设计的需求,或使用者对闪存80的使用情况执行对应的调整。例如,在数据已经快要满载,而不得不用到单层单元闪存来存放数据时,也可以动态开启,关闭或是调整以下的错误调整机制。

[0100] 此外,图8中的单层单元闪存也可以用质量比较好或是存取速度比较快的多层单元闪存来替代,甚至也可以使用易失性存储器,例如随机存取存储器来替代。例如假如有后备电源,即使突然断电或是主电池用完,后备电源仍然可以维持随机存取存储器一定时间

保存数据的能力,而在此期间,控制器或存储器接口电路尽快将数据存放到非易失性存储器之中。另外,也可以将图8的多层单元(MLC)改成三层单元(TLC)(同一个存储单元存放三个或更多比特信息),或是同时把图8的单层单元(SLC)改成多层单元(MLC)。这一类的变更都可以适用以下的错误管理机制。

[0101] 图9绘示了在多层单元闪存内的数据存放格式。如上所述,闪存通常是进行分组,以执行读取,写入或是删除的动作,而图9显示的就是图8的多层单元闪存之中一种数据排列配置的范例。

[0102] 在图9中绘示了的数据排列配置中,整个阵列构成一个区块,在这个区块中,每一列构成了一个数据页,在每列中的每个方块代表的,就是用来存放一个字节或是一个比特的存储单位,所述数据页可视为逻辑上的数据页,也可视为实体上的数据页,本领域的技术人员应可在本发明的教导之下,自由选择运用。通常,闪存的标准,例如联合电子设备工程委员会(Joint Electron Device Engineering Council,JEDEC),会要求区块的大小是固定值,并且数据页的大小也是固定值,不可以任意改变,因此,让不同的制造商制造的产品之间,具有一定程度的兼容性,以利整体产业的研究跟发展。然而,以下所描述的做法,即使在区块大小固定,以及数据页大小固定下,仍然可以加以运用。当然,本发明的做法也可以用在特殊的区块跟数据页的大小规格。

[0103] 另外,如上所述,为了达成错误侦测,以及错误更正的能力,可以使用目前已知或正在发展的错误处理编码方式。例如LDPC,BCH或里德-所罗门码(Reed Solomon codes)等。由于这类的计算方法会针对所要储存的数据执行运算,以产生校验码数据(parity data),这些校验码数据,可通过图8的控制器82执行对应的计算产生出来之后,一并存放到图8的MLC的存储器单元中。

[0104] 在图9中的这个范例中,每一个数据页,也就是每列的列校验码放在最后的每一列最后头(标示为列校验码的斜线方块)。这里虽然只以一个方块为例,但究竟要保留多少存储器单元来放置校验码,则跟到底要保留多高的容错率,以及使用哪种算法有关,而可以做对应的调整,本领域的技术人员得在本发明的教导之下,搭配使用适当比特数的列校验码。

[0105] 此外,除了对于每列资料执行错误侦测或错误更正的编码,也可以针对每行数据执行错误侦测或错误更正的编码处理。在图9的例子中,针对每行执行错误更正编码产生的一个或多个校验列存放在这个区块最下方(标示为校验列的斜线方块)。在一实施例中,可将一校验列放置在所述区块的最后一页,但不以此为限。同样地,到底要保留多少的空间来放置校验码数据(校验列)跟选择的错误更正算法以及所需要的容错率有关,而可以作对应的调整。

[0106] 在这个例子中,校验列与列校验码的产生,虽然可以用相同的算法,但也不一定要用同样的算法。例如校验列对应的是在垂直方向存储器单元的状况,而垂直方向存储器单元可能有共享信号线,而跟水平方向存储器单元间有不同的关联特性。此时,各自使用不同的算法可以更有效的完成错误管理的处理工作。

[0107] 另外,在图9中可看到标示校验列跟列校验码有交集的地方。这个地方的数据可以用另一种算法或产生校验列跟列校验码的算法,或是全部都采用同样的算法。在一些算法中,不管是先算列校验码再算校验列或是相反,最后得出交集部分的校验数据都会一样。

[0108] 在图9中,在同一个区块中的所有数据页在垂直方向共同算出对应的校验数据(校

验列),但这并不是必要的限制,图10跟图11说明了不同的数据与校验数据之间的配置关系,而这些配置都可以应用在这个实施例的方法。

[0109] 图10绘示了第二种配对数据跟校验数据的方法。在图10中的例子中,每三列(数据页)就提供一个对应的列(数据页)用来存放校验列数据(校验列),整体构成了一个数据组。也就是说,在这样的配置下,同一个区块可以有多个数据组。当然,这边绘示的三列数据配置一列校验列也可以改成例如五十列数据配置三个校验列或各种其它的数字。

[0110] 相似的,在水平方向,数据与列校验码之间的配对关系也可以作动态的调整。到底要有多少的列校验码以及列校验码是放在哪个位置,列校验码跟数据是不是要交错放置,或列校验码是否统一放在最前面或最后面,都可以根据实际的需要作调整。

[0111] 图11绘示了第三种配对数据跟校验码的方法。图11绘示了另一种可能性。在图11中,数据跟校验数据的比例不需要是固定比例,而可以根据需求动态调整。举例来说,控制器可以在确定配合的闪存之后,通过驱动程序或是设定寄存器等设定所要使用的错误管理方式,数据跟校验码之间的比例以及如何安排存放位置等等。

[0112] 事实上,本发明的实施例也可以在产品出厂之后,通过统计实际的错误发生情况,而动态调整或是选择不同的算法,设置不同的数据跟校验码之间的比例以及如何安排存放位置。这些可能的变化方式都被本发明的概念所涵盖。

[0113] 图12的流程图及图13至图16的示意图,说明了图8之中的数据从SLC读出之后写到MLC的处理流程。

[0114] 在这个处理范例中,控制器的第二存储元件写入器824对MLC的区块执行分组。在图13的范例中,为了说明方便,每三个数据列(数据页)就搭配一个校验列(数据页),共同构成一个数据组。当然,在实际的应用中,数据列跟校验列的比例会比较小,例如在一个有100个数据页的区块中,可以分成五组,也就是每组19个数据列搭配一个校验列。

[0115] 接着请结合图12至图16,用来了解这个范例所公开的做法。

[0116] 首先,控制器先从SLC读出一个数据页放到缓冲器中(步骤1202),由于这个数据页本身就有可能出错,控制器一旦发现错误(步骤1204),就用SLC读到的数据之中的校验码对数据执行更正(步骤1206)。更正后的数据,或者说,正确的数据接着被传到闪存的存储器接口电路,将这个数据页的数据写到对应的MLC的数据页位置(步骤1208)。

[0117] 此外,这个经过检查的数据也传给控制器中的错误更正码产生单元(步骤1210),用来产生列校验码(步骤1211)。要产生这样的数据,在这个例子之中,是将本次检查过的数据列跟前次算出来的校验列数据进行累积计算,以更新校验列的内容,其中,本次处理的数据页由图13/图14/图15中MLC下方以阴影线表示的数据列来表示,上次累积的校验列由图13/图14/图15中本次处理的数据页下方的数据列来表示,以及更新的的校验列由图13/图14/图15中上次累积的校验列下方的数据列来表示。

[0118] 举例来说,图14的校验列使用图13算出来的校验列跟图14读到的第二列数据,用例如LDPC,BCH等方法来产生。图15的校验列则是使用图14更新的校验列跟图15读到的第三列数据作运算而执行更新。

[0119] 等到本次分组的数据列都已经处理完毕,也就是对于本数据组的校验列已经完整的得出,这个校验列(以图16中MLC下方的数据列来表示)就被写入到MLC对应的位置中(步骤1212),请一并参考图16。

[0120] 通过这样的做法,即使在操作的过程中发生断电的情况,例如使用者拔插头或是临时把存储卡从计算机拔出,或是把手机的电池取出,受影响的将不会是整个区块的数据,而最多只是受影响的所述组数据需要重新写入到MLC。

[0121] 接下来,请参考图17至图20,其用来说明本发明的另一实施例。

[0122] 图17绘示了一种存储器控制方法的一实施例的流程图。这个控制方法用于控制一闪存,以读取所述闪存内的数据。

[0123] 这个闪存具有一个存储元件,例如一个多层单元闪存(MLC),而这个存储元件具有多个区块。每一区块具有多个数据页,这个存储器控制方法包含下列步骤。

[0124] 首先,读取数据组设定,以得知存储元件的区块跟数据列的配置关系,例如一个数据组包含多少数据列、多少校验列等信息,一个区块内那些数据页存放的是数据列,那些又是校验列。接着读取数据组的一个数据列(步骤170)。利用所述数据列的列校验码判断所述数据列数据是否有错误发生(步骤171)。假如发现有错误发生且所述错误可更正(步骤172),则更正错误(步骤173)。

[0125] 假如通过校验码无法直接更正错误,例如校验码能承受的误差为三个比特,而此时发生超过三个比特的错误,便将这个数据列无法更正错误的状态记录起来,例如,在一错误更正表中记录所述数据列的地址(步骤174)。

[0126] 接着,继续读取数据组的其它数据列以及校验列。利用校验列对应的校验方法,例如LDPC或BCH或其它相关校验方法,计算所述数据组中各行的对应校正子(步骤175)。也就是所述数据组各数据列排列成阵列(array)时,跨数据列同相对位置数据的组合。

[0127] 以图18为例,其绘示了一个存储元件,例如闪存模块一个区块的逻辑分配示意图。在这个区块中,具有两个数据组182跟184。第一个数据组182由七个数据列1821跟两个校验列1822组成。除了校验列,数据列1821中的每个数据列还有自己的列校验码1823。这些数据列以图示阵列方式排列,同一列指的是水平方向数据单元构成的逻辑数据单位,同一行指的是垂直方向数据单元构成的逻辑数据单位,例如图示186代表这个区块的第一行数据。

[0128] 由于LDPC跟BCH等校验方法,在根据校验列跟数据组的数据内容计算出校正子之后,可以指出到底可能在哪儿几列出现错误并相对应的执行更正。例如,根据第一个数据组182的第一行数据1823与第一行数据的校验列部分1824执行错误更正运算(例如计算校正子等),可以得知所述第一行数据是否存在错误,以及判断所述错误状态是否在可更正的范围内(步骤176),例如第一行数据的校验列部分1824可提供3比特的更正能力,而第一行数据1823仅有2比特的错误,所述错误状态即属于可更正。若所述错误状态可更正,则执行更正(步骤178)。反之,若所述错误状态不可更正,则不执行更正。

[0129] 举例来说,图19中的数据组192具有七个数据列以及两列校验列(下方画阴影线的地方)。首先,每个数据列先用自己的列校验码执行校验,假如发现错误就执行更正。假如发现错误又无法更正,则将其记录在表格194中(画阴影线对应的数据列)。在这个例子中,每个列校验码最多可更正两个错误,而每个校验列可更正一个错误。表格194指出第二列跟第四列发现错误,而且这个错误无法直接由其列校验码直接加以更正(错误超过两个)。

[0130] 接着,一行一行的利用校验列来协助更正。首先,利用数据组192的第一行数据1921与其相对应的校验列1922对第一行数据1921执行错误校验,并且发现所述行数据没有发生错误。接着,利用数据组192的第二行数据1923与其相对应的校验列1924对第二行数据

1923执行错误校验,并且发现所述行数据具有一个错误(标示星号的地方),在此实施例中,每个校验列可更正一个错误,故立即对此错误执行更正。以此类推。接下来,利用数据组192的第四行数据1925与其相对应的校验列1926对第四行数据1925执行错误校验,并且发现所述行数据具有两个错误(标示星号的地方),在此实施例中,每个校验列可更正两个错误,故无法对此错误执行更正。

[0131] 以此类推,在利用校验列逐行对所述数据组192执行错误校验之后,可以更正各数据行中具有一个错误的错误状态。从而所述数据组的错误状态可以图20表示,其中,标示星号的数据单元即表示发生错误的数据单元。从图20可以发现,数据组192的每一个数据列中,都不存在超过两个错误的的数据单元,故此时可再利用相对应的列校验码对第二列跟第四列数据执行错误校验,则可得到正确的第二列及第四列。

[0132] 综上,假如原先因为错误太多,无法用列校验码更正,此时可能因为利用校验列执行错误校验之后,使得错误减少而可以利用列校验码更正,例如图20所描绘的,代表错误的星星符号减少。一旦更正之后,错误更少,再反复使用先前垂直方向的校正子则可以继续减少错误。

[0133] 通过多次(iteration)的重复操作,虽然不是所有的错误一定都可以更正回来,但确实有更高比例的错误可以通过这些方法更正回来。

[0134] 以上所述仅为本发明的优选实施例而已,并不用于限制本发明,对于本领域的技术人员来说,本发明可以有各种更改和变化。凡在本发明的精神和原则之内,所作的任何修改、等同替换、改进等,均应包含在本发明的保护范围之内。

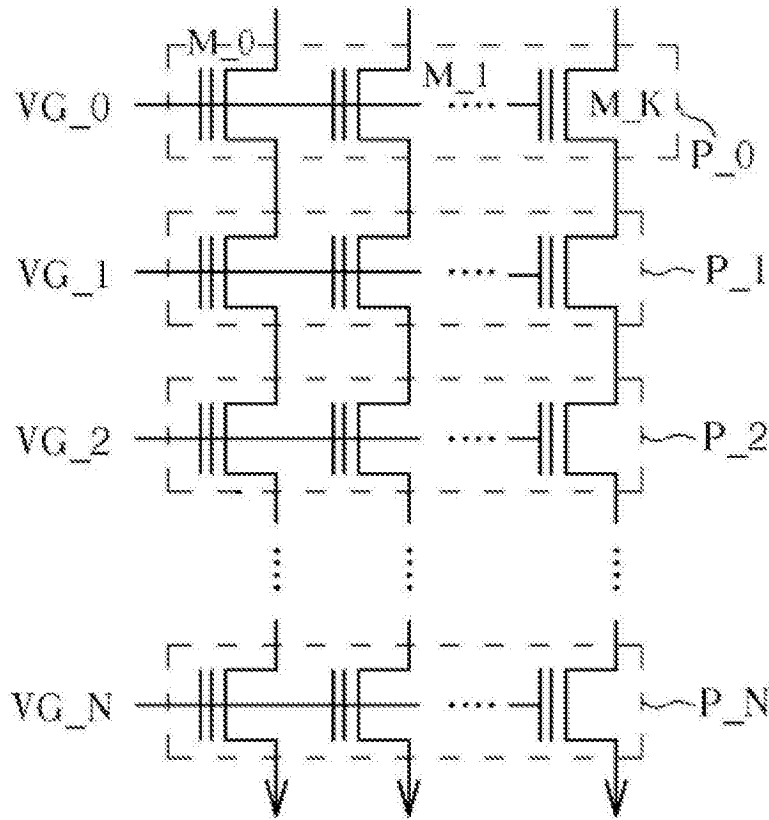


图1

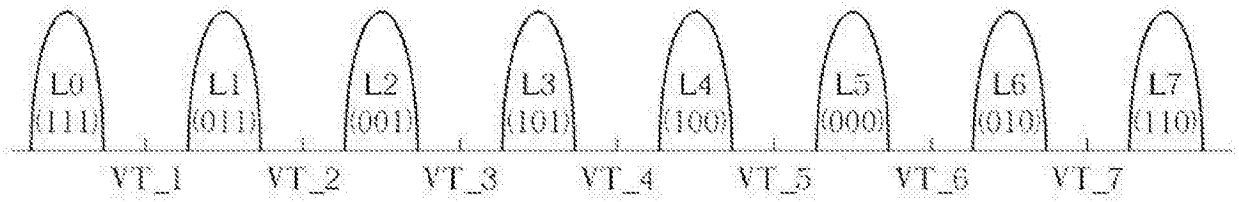


图2

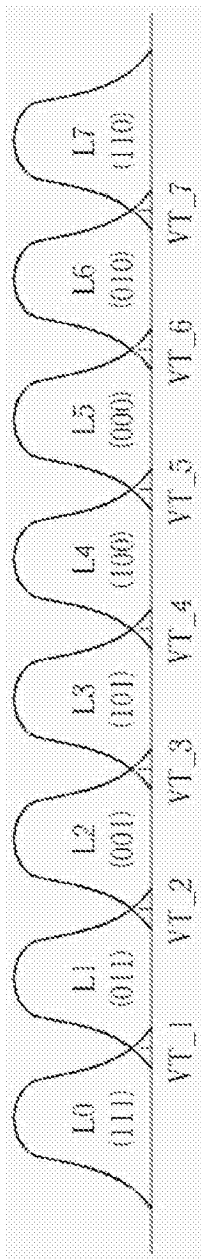


图3

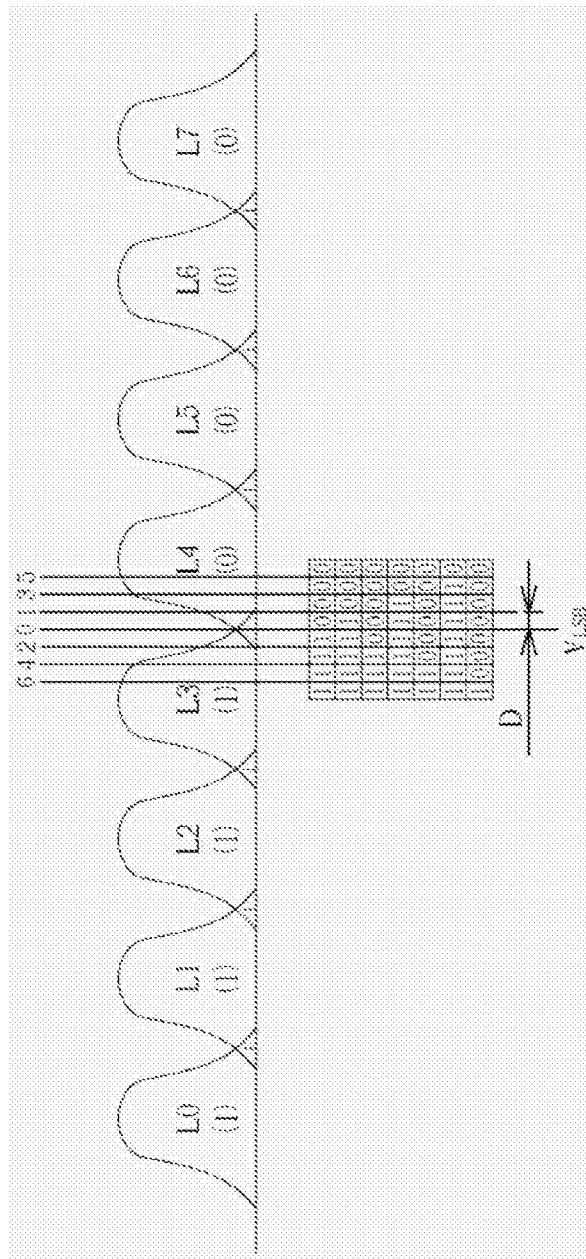


图4

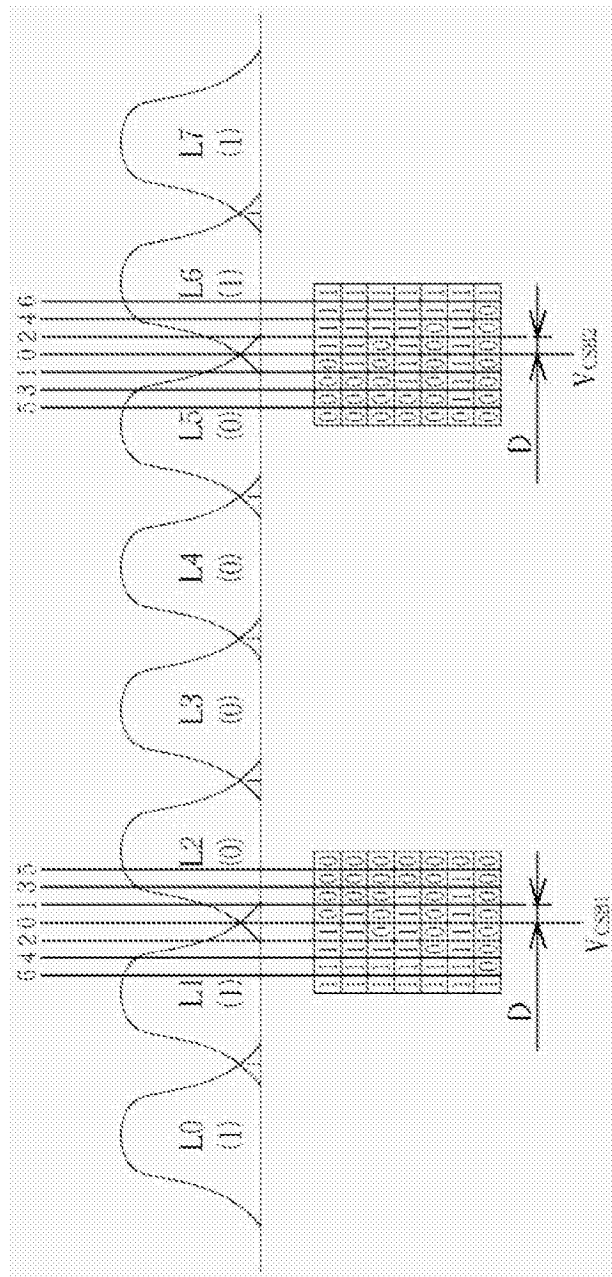


图5

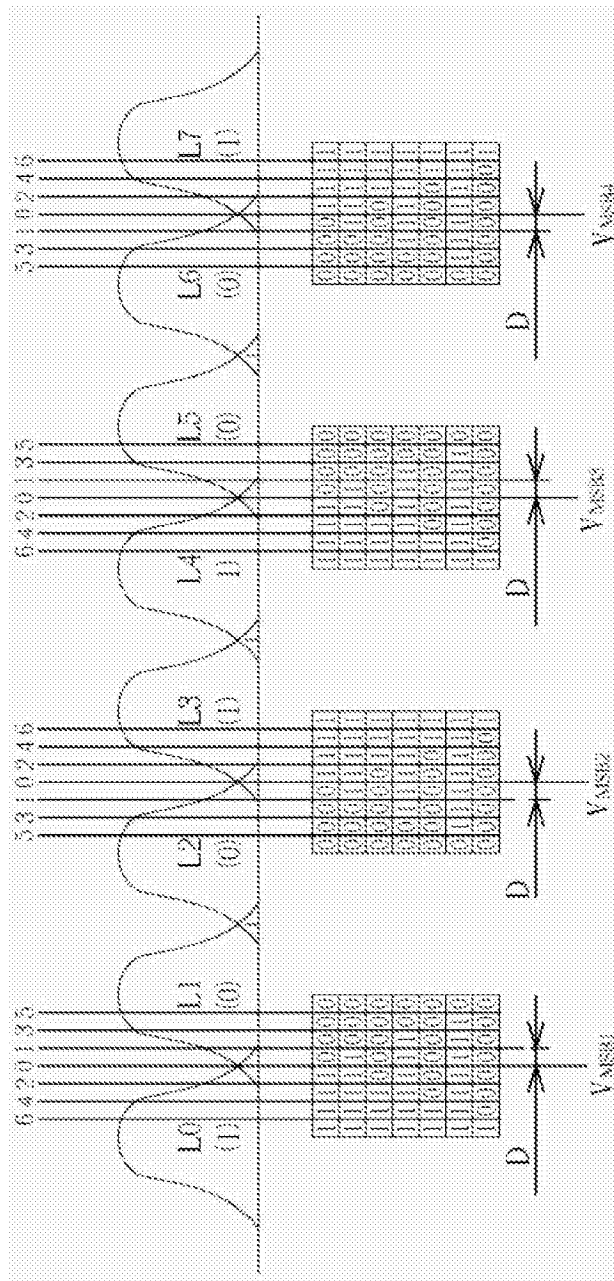


图6

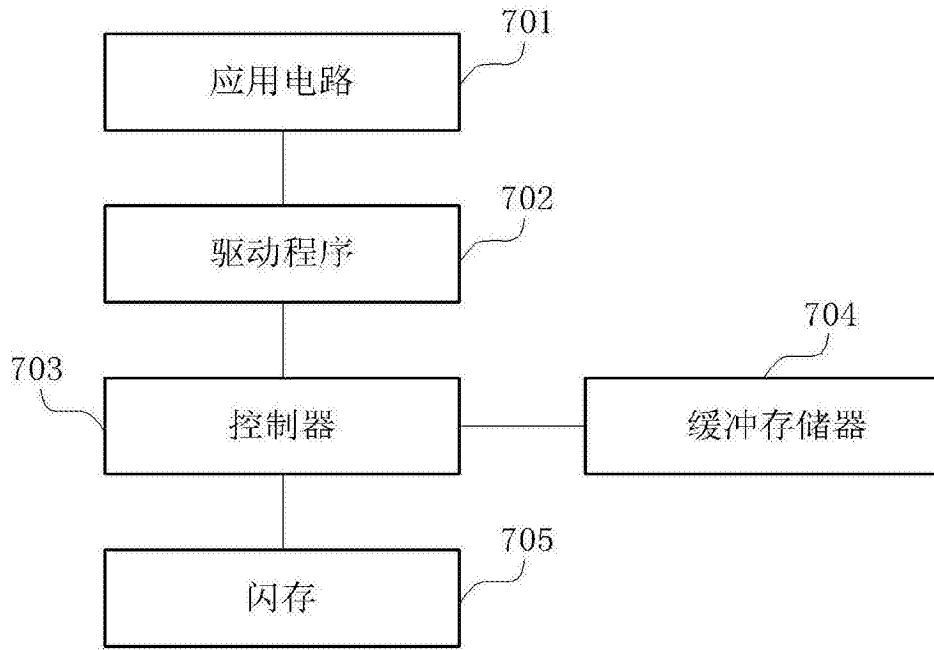


图7

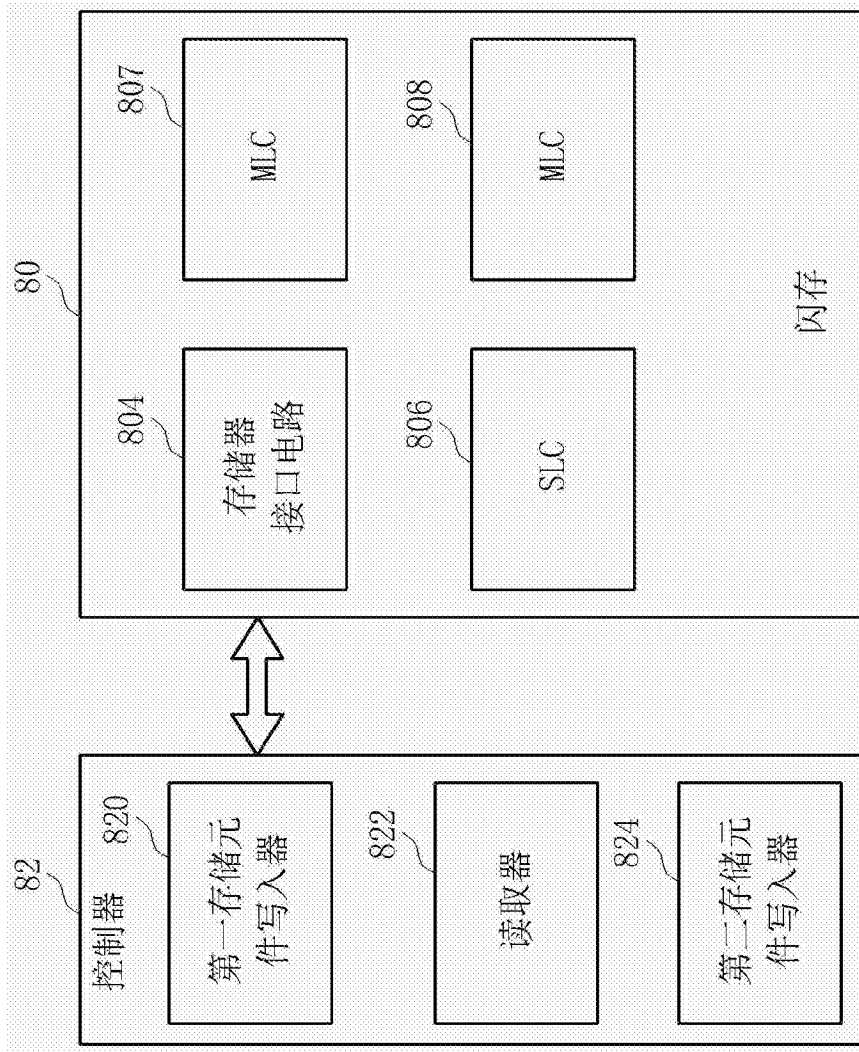


图8

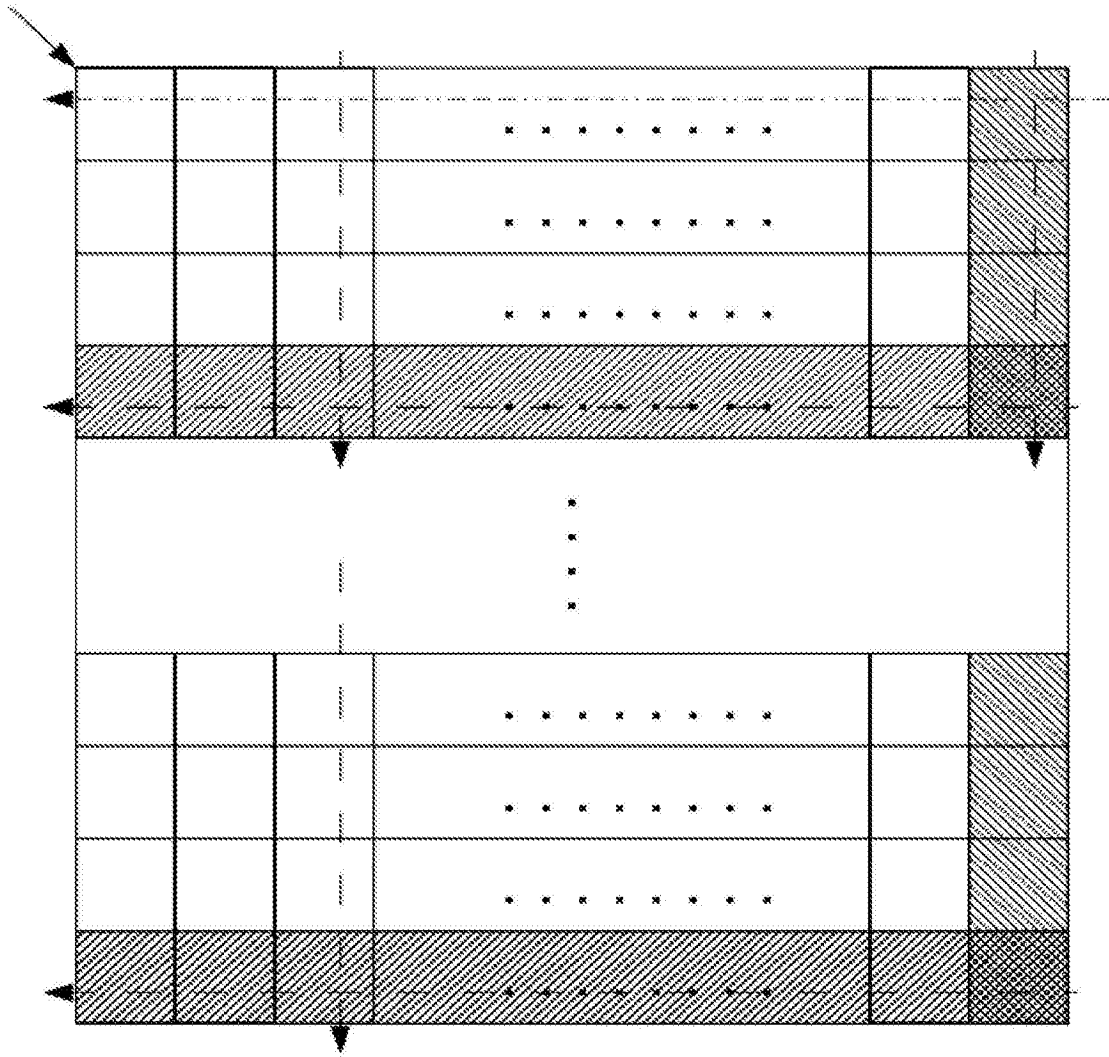


图10

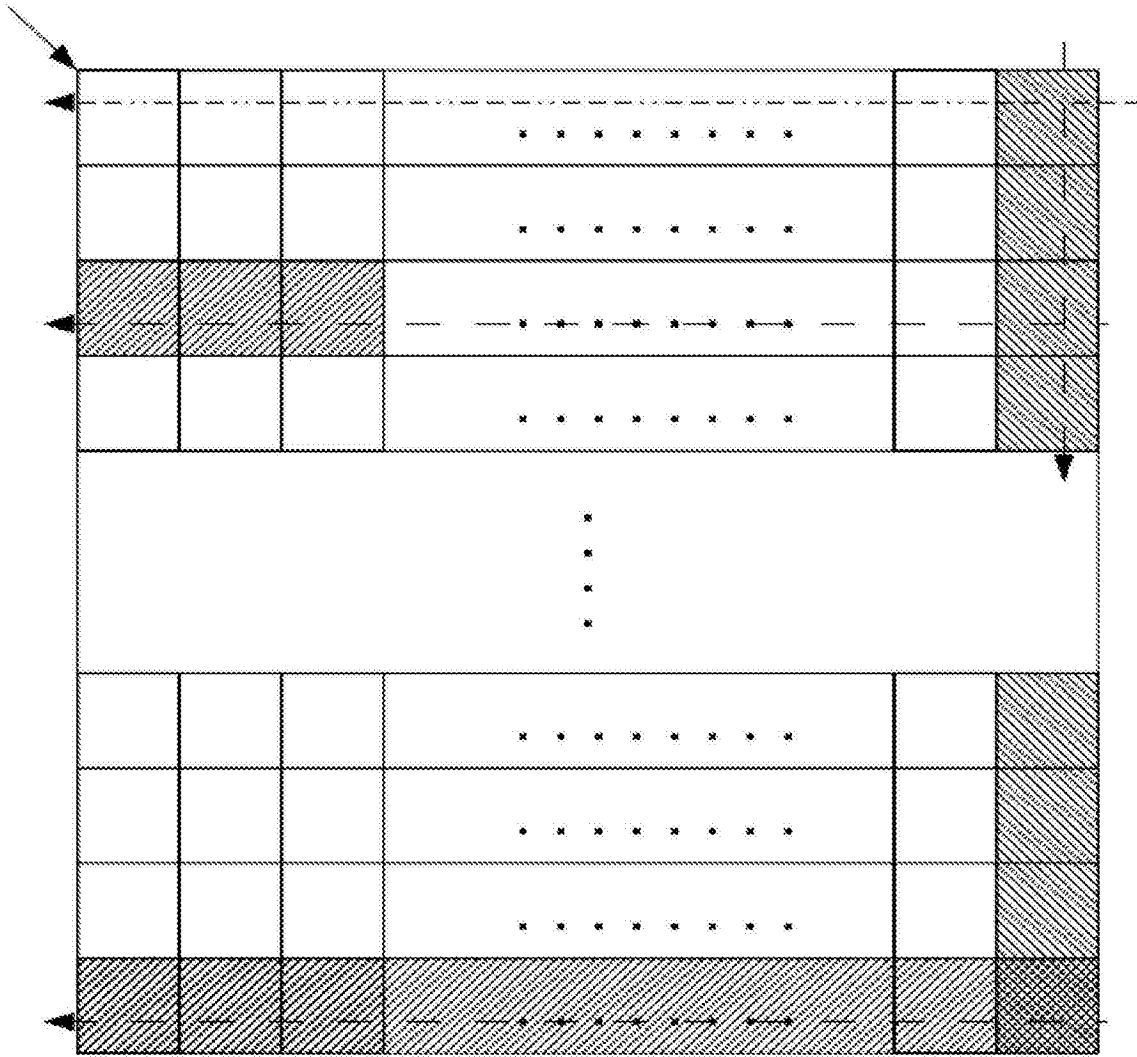


图11

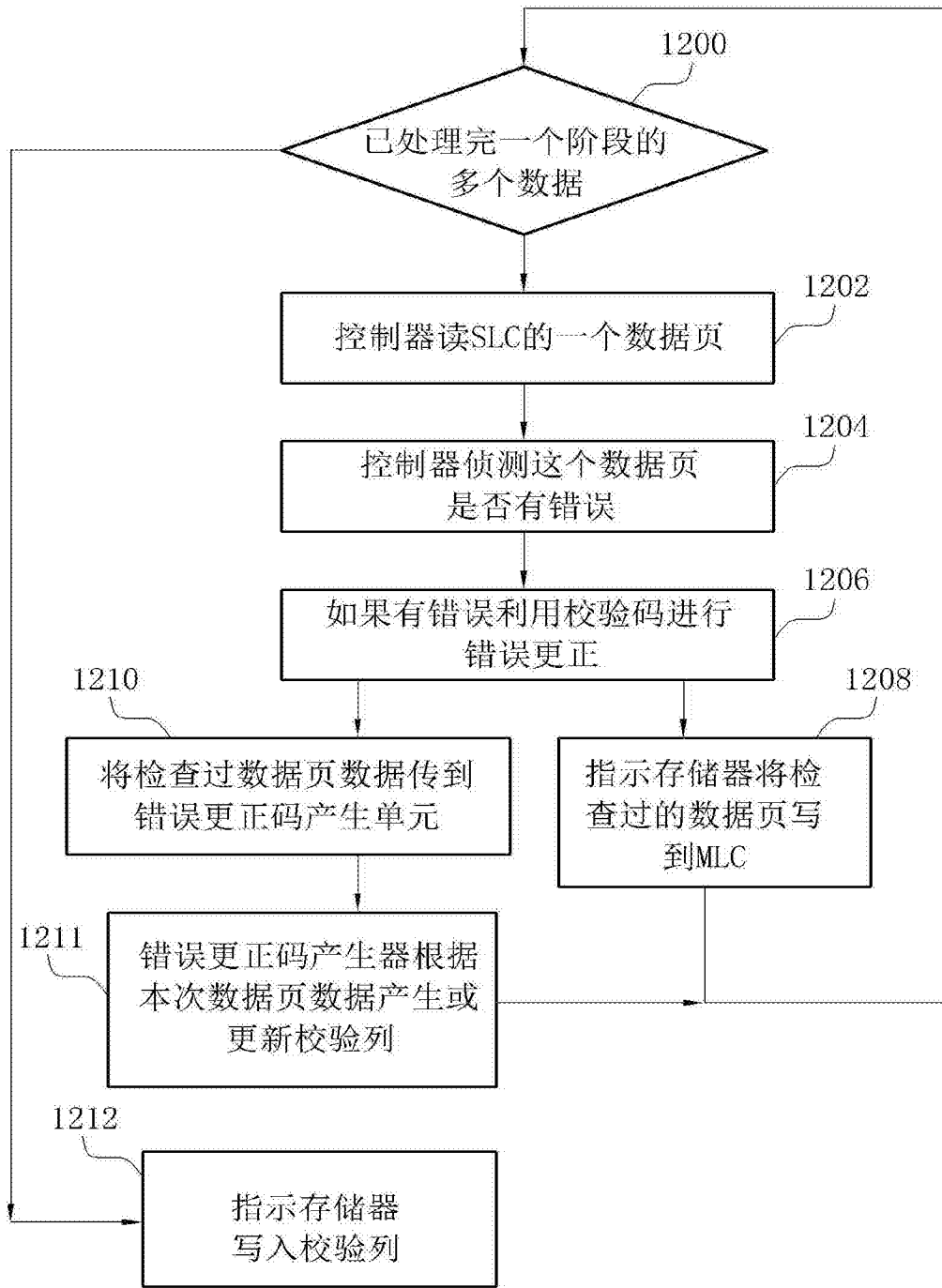


图12

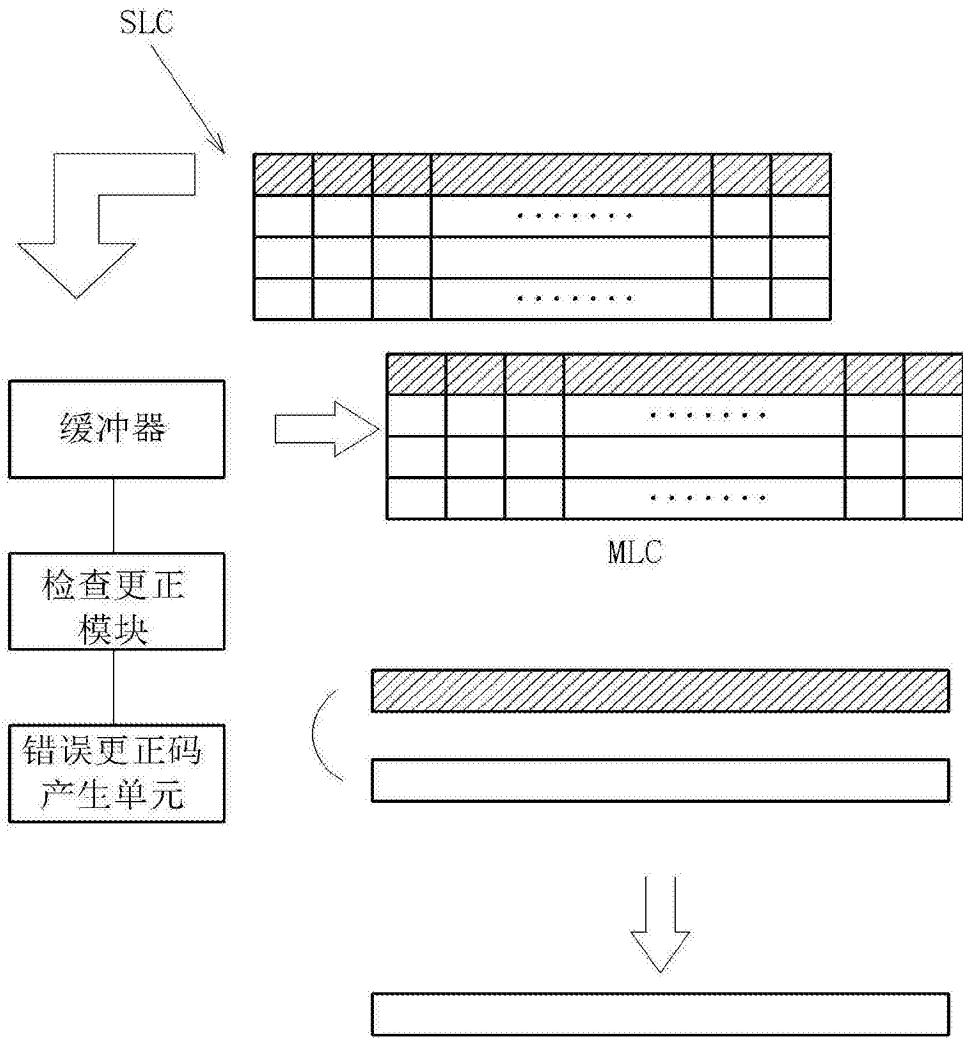


图13

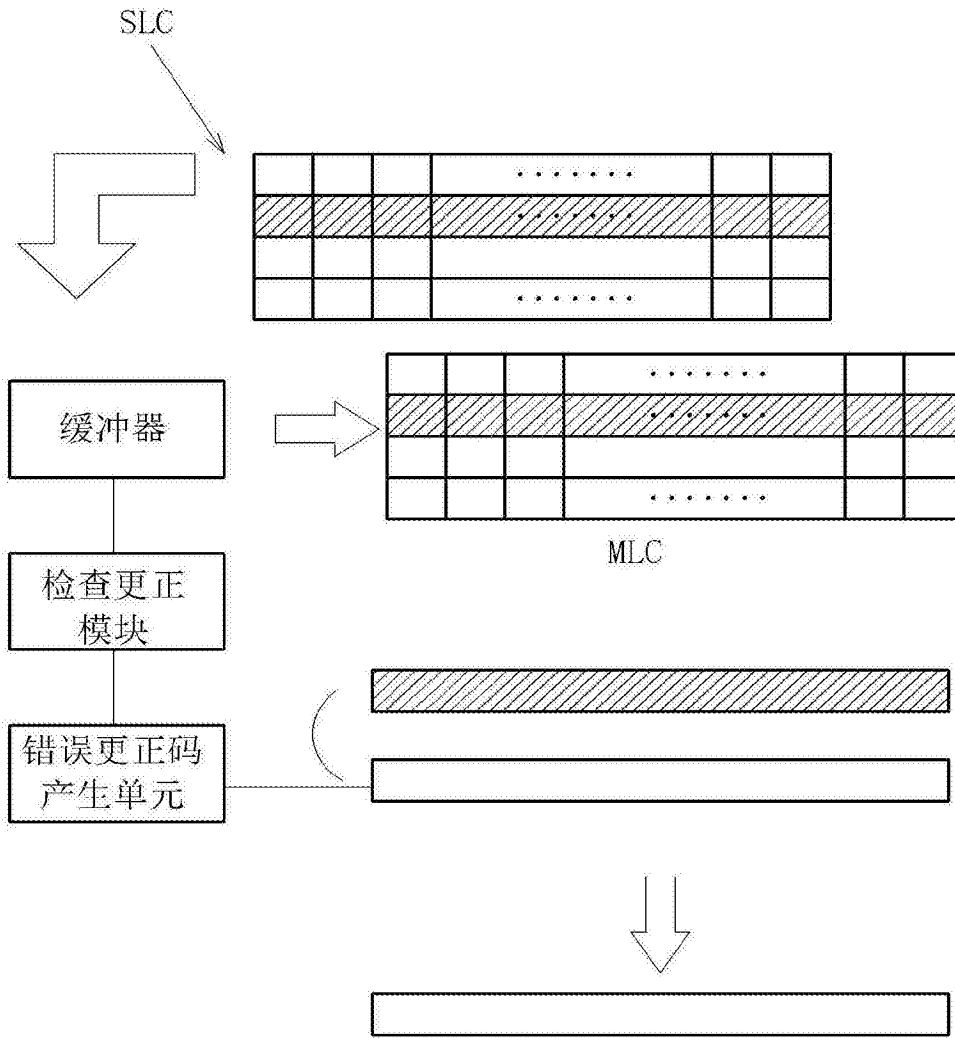


图14

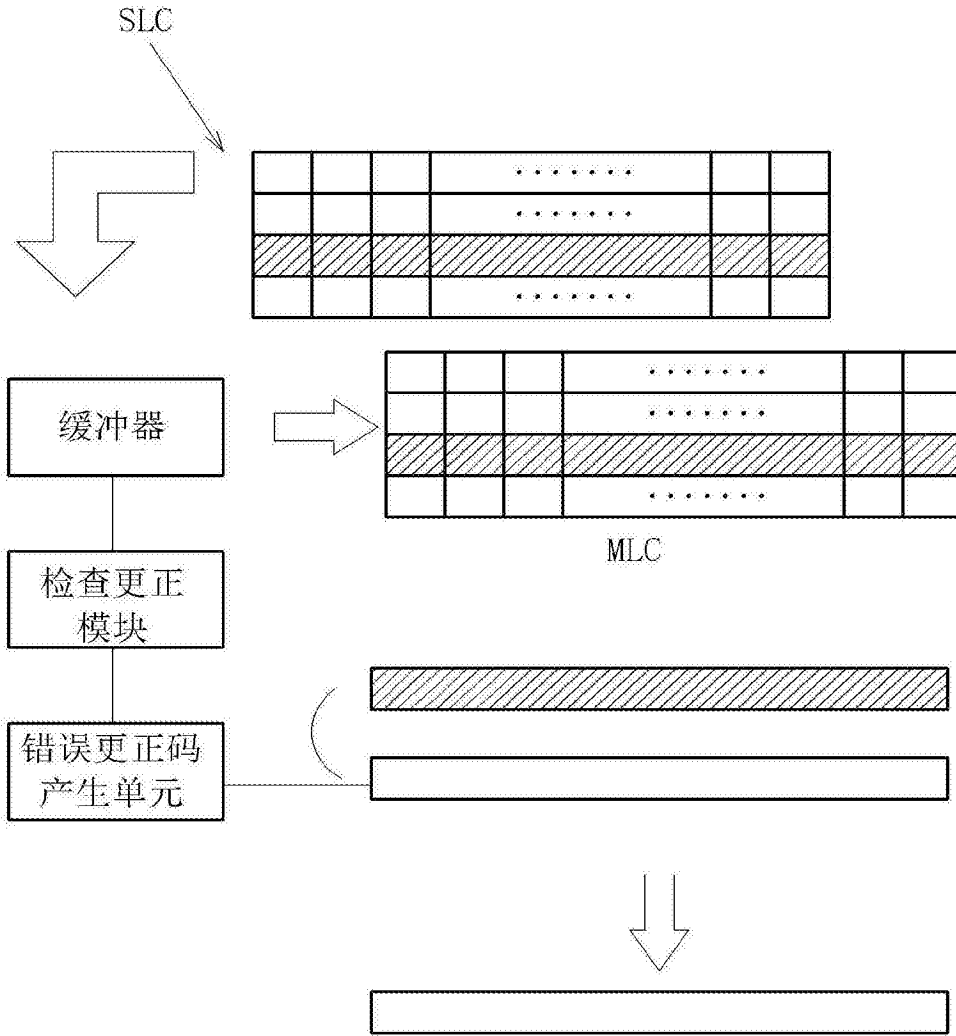


图15

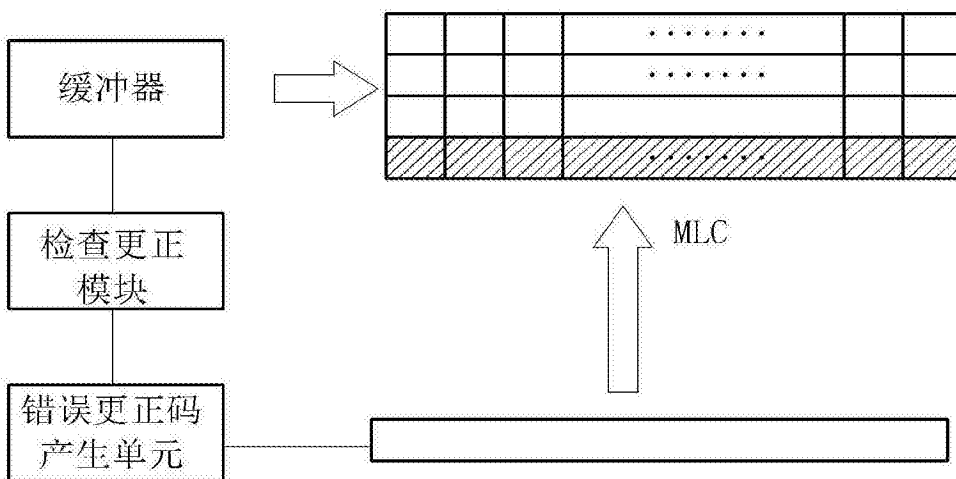


图16

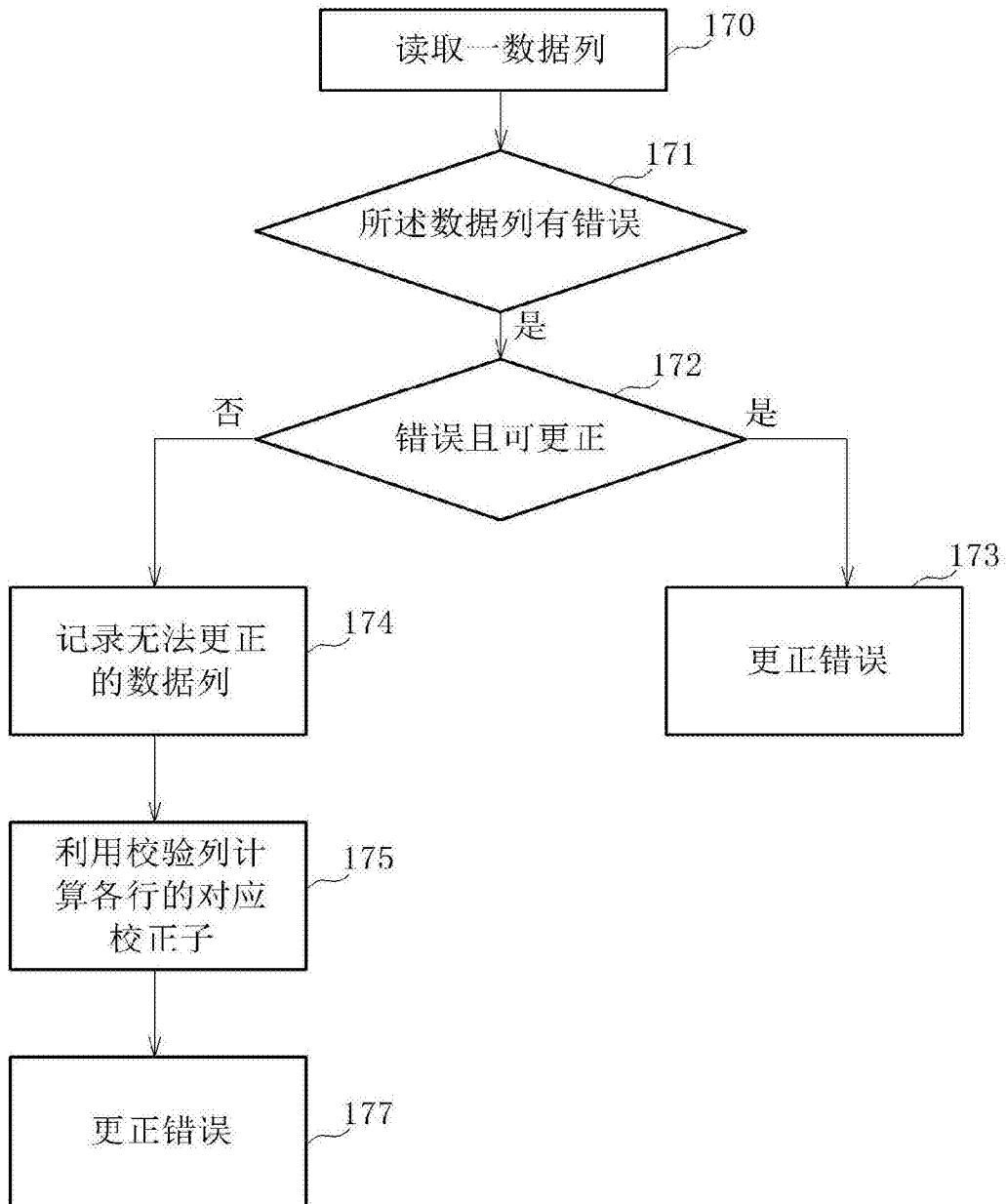


图17

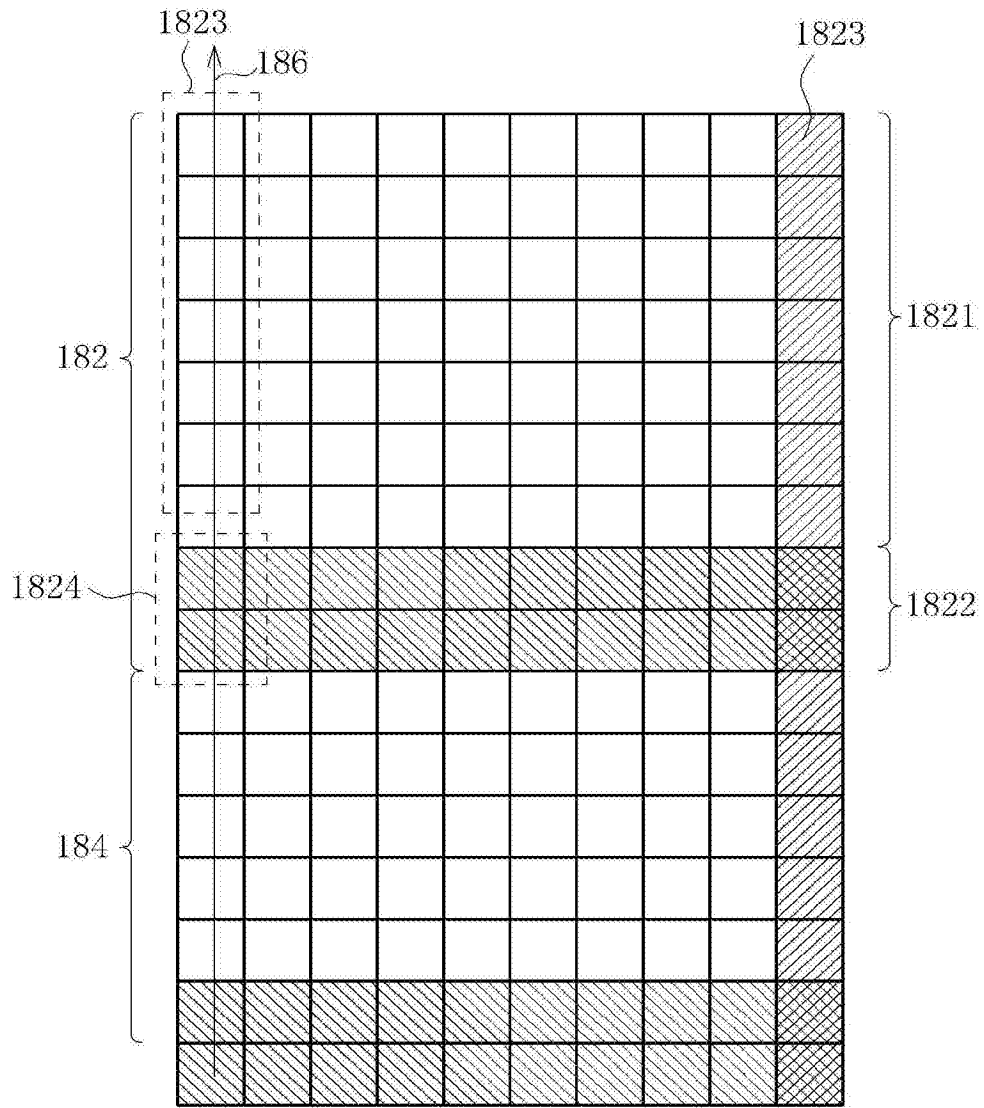


图18

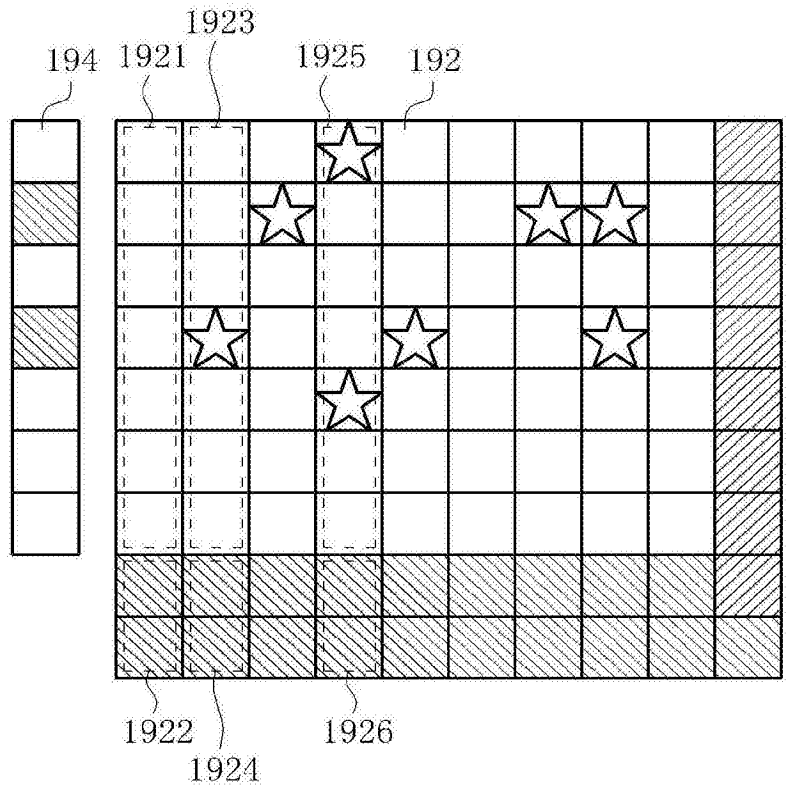


图19

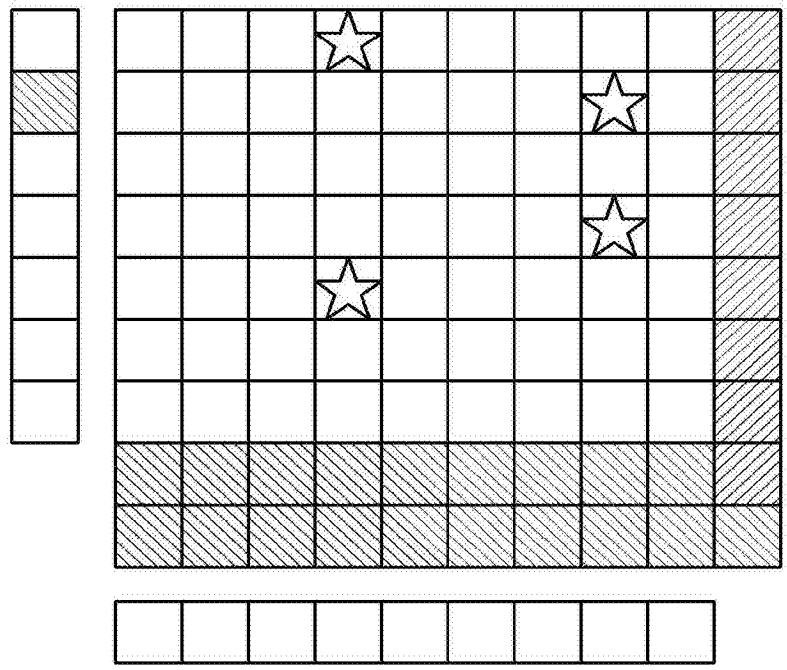


图20