



(12) 发明专利申请

(10) 申请公布号 CN 116615718 A

(43) 申请公布日 2023. 08. 18

(21) 申请号 202180083119.8

(22) 申请日 2021.12.07

(30) 优先权数据

17/116,952 2020.12.09 US

(85) PCT国际申请进入国家阶段日

2023.06.09

(86) PCT国际申请的申请数据

PCT/US2021/062201 2021.12.07

(87) PCT国际申请的公布数据

W02022/125545 EN 2022.06.16

(71) 申请人 超威半导体公司

地址 美国加利福尼亚州

(72) 发明人 罗丝·V·拉费特拉

(74) 专利代理机构 上海胜康律师事务所 31263

专利代理师 樊英如 张静

(51) Int.Cl.

G06F 11/10 (2006.01)

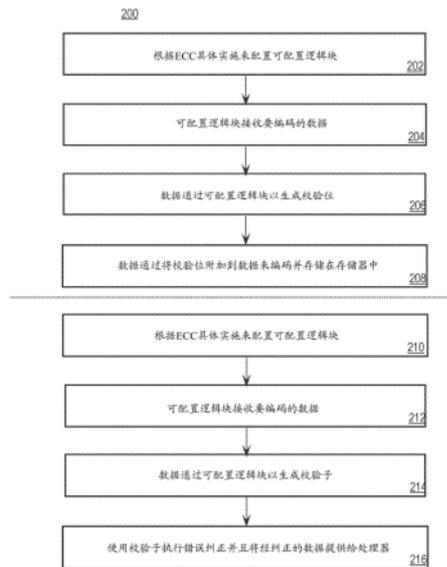
权利要求书4页 说明书9页 附图8页

(54) 发明名称

可编程纠错码编码和解码逻辑

(57) 摘要

一种存储器模块包括可配置用于特定ECC具体实施的逻辑元件。如本文所用,术语“ECC具体实施”是指用于执行错误检测和后续处理的ECC功能性,例如使用该错误检测的结果来执行错误纠正并且编码数据,使得以后可识别并纠正任何错误。该方法允许将存储器模块或计算设备配置用于特定ECC具体实施,而不需要在主机之间来回发送请求。



1. 一种方法,包括:

在可配置逻辑元件集处接收所述配置指令,所述配置指令针对所述纠错码(ECC)具体实施定义所述可配置逻辑元件集中要进行掩码处理的位;

在所述可配置逻辑元件集处接收要根据所述ECC具体实施编码或解码的数据;

使用具有根据所述配置指令进行了掩码处理的数据或码字位的所述可配置逻辑元件集来执行所述ECC具体实施。

2. 根据权利要求1所述的方法,其中:

所述配置指令包括根据对应于所述ECC具体实施的G矩阵定义所述逻辑元件中要进行掩码处理的位;

执行所述ECC具体实施包括编码数据,编码所述数据包括对特定数据值执行以下操作:

使用具有根据所述配置指令进行了掩码处理的位的所述元件来确定所述数据值的多个校验位值;以及

生成并存储所述特定数据值的码字,所述码字包括所述数据值的数据和所述多个校验位。

3. 根据权利要求2所述的方法,其中确定所述多个校验位值包括使用具有根据所述配置指令进行了掩码处理的位的所述逻辑元件,将所述码字与所述G矩阵相乘以生成所述多个校验位。

4. 根据权利要求1所述的方法,其中:

所述配置指令包括根据对应于所述ECC具体实施的H矩阵定义所述逻辑元件中要进行掩码处理的位;

执行所述ECC具体实施包括解码码字,解码所述码字包括:

使用具有根据所述配置指令进行了掩码处理的位的所述逻辑元件,利用所述码字计算校验子列;

识别所述H矩阵的与所述校验子列匹配的特定列,并且作为响应,确定在对应于所述H矩阵的所述特定列的所述码字的值中存在错误。

5. 根据权利要求1所述的方法,其中:

所述配置指令包括根据对应于所述ECC具体实施的H矩阵定义所述逻辑元件中要进行掩码处理的位;

执行所述ECC具体实施包括解码码字,解码所述码字包括:

对于所述H矩阵的每个符号集,存储对应于第一行集的第一值集与对应于第二行集的第二值集的比率集;

使用具有根据所述配置指令进行了掩码处理的位的所述逻辑元件计算校验子;

确定所述校验子的对应于所述第一行集的第一部分等于特定符号集的所述比率与所述校验子的对应于所述第二行集的第二部分的乘积,并且作为响应,确定所述数据的对应于所述特定符号集的部分包含错误。

6. 根据权利要求5所述的方法,其中解码所述码字还包括:

通过计算所述校验子的所述第一部分与对应于所述特定符号的所述第一行集的所述第一值集的商来确定所述错误的量值;

使用所述错误的所述量值和所述码字的被确定为包括所述错误的所述部分来纠正所

述数据。

7. 根据权利要求1所述的方法,其中所述可配置逻辑元件被配置为通过在将码字存储在存储器中之前对由处理器生成的数据进行编码或通过对存储在所述存储器中的所述码字进行解码来为存储器模块提供所述ECC具体实施。

8. 根据权利要求1所述的方法,其中所述可配置逻辑元件被配置成为经由网络发送或接收的数据提供所述ECC具体实施。

9. 一种装置,包括:

可配置逻辑元件集;

其中所述可配置逻辑元件集根据配置指令进行配置,所述配置指令针对纠错码(ECC)具体实施定义所述可配置逻辑元件集中要进行掩码处理的位;

其中所述可配置逻辑元件集被配置成通过根据所述配置指令对所述可配置逻辑元件集中的数据或码字位进行掩码处理来执行所述ECC具体实施。

10. 根据权利要求10所述的装置,其中:

所述配置指令包括根据对应于所述ECC具体实施的G矩阵定义所述逻辑元件中要进行掩码处理的位;

执行所述ECC具体实施包括编码数据,编码所述数据包括对特定数据值执行以下操作:

使用具有根据所述配置指令进行了掩码处理的位的所述元件来确定所述数据值的多个校验位值;以及

生成并存储所述特定数据值的码字,所述码字包括所述数据值的数据和所述多个校验位。

11. 根据权利要求10所述的装置,其中确定所述多个校验位值包括使用具有根据所述配置指令进行了掩码处理的位的所述逻辑元件,将所述码字与所述G矩阵相乘以生成所述多个校验位。

12. 根据权利要求9所述的装置,

所述配置指令包括根据对应于所述ECC具体实施的H矩阵定义所述逻辑元件中要进行掩码处理的位;

执行所述ECC具体实施包括解码码字,解码所述码字包括:

使用具有根据所述配置指令进行了掩码处理的位的所述逻辑元件,利用所述码字计算校验子列;

识别所述H矩阵的与所述校验子列匹配的特定列,并且作为响应,确定在对应于所述H矩阵的所述特定列的所述码字的值中存在错误。

13. 根据权利要求9所述的装置,其中:

所述配置指令包括根据对应于所述ECC具体实施的H矩阵定义所述逻辑元件中要进行掩码处理的位;

执行所述ECC具体实施包括解码码字,解码所述数据包括:

对于所述H矩阵的每个符号集,存储对应于第一行集的第一值集与对应于第二行集的第二值集的比率集;

使用具有根据所述配置指令进行了掩码处理的位的所述逻辑元件计算校验子;

确定所述校验子的对应于所述第一行集的第一部分等于特定符号集的所述比率与所

述校验子的对应于所述第二行集的第二部分的乘积,并且作为响应,确定所述码字的对应于所述特定符号集的部分包含错误。

14. 根据权利要求13所述的装置,其中解码所述码字还包括:

通过计算所述校验子的所述第一部分与对应于所述特定符号的所述第一行集的所述第一值集的商来确定所述错误的量值;

使用所述错误的所述量值和所述码字的被确定为包括所述错误的所述部分来纠正所述数据。

15. 根据权利要求9所述的装置,其中所述可配置逻辑元件被配置为通过在将码字存储在存储器中之前对由处理器生成的数据进行编码或通过存储在所述存储器中的所述码字进行解码来为存储器模块提供所述ECC具体实施。

16. 根据权利要求9所述的装置,其中所述可配置逻辑元件被配置成为经由网络发送或接收的数据提供所述ECC具体实施。

17. 一种存储器模块,包括:

存储器,所述存储器存储根据纠错码(ECC)具体实施编码的数据;

处理器,所述被配置成对存储在所述存储器中的数据执行操作并向所述存储器写入另外数据;

可配置逻辑元件集,所述可配置逻辑元件集根据配置指令进行配置,所述配置指令针对纠错码(ECC)具体实施定义所述可配置逻辑元件集中要进行掩码处理的位;

其中所述可配置逻辑元件集被配置成通过在写入到所述存储器之前或在由所述处理器读取之前根据所述配置指令对所述可配置逻辑元件集中的位进行掩码处理来执行所述ECC具体实施。

18. 根据权利要求17所述的存储器模块,其中:

所述配置指令包括根据对应于所述ECC具体实施的G矩阵定义所述逻辑元件中要进行掩码处理的位;

执行所述ECC具体实施包括编码数据,编码所述数据包括对特定数据值执行以下操作:

使用具有根据所述配置指令进行了掩码处理的位的所述元件来确定所述数据值的多个校验位值;以及

生成并存储所述特定数据值的码字,所述码字包括所述数据值的数据和所述多个校验位。

19. 根据权利要求17所述的存储器模块,其中:

所述配置指令包括根据对应于所述ECC具体实施的H矩阵定义所述逻辑元件中要进行掩码处理的位;

执行所述ECC具体实施包括解码码字,解码所述码字包括:

使用具有根据所述配置指令进行了掩码处理的位的所述逻辑元件,利用所述码字计算校验子列;

识别所述H矩阵的与所述校验子列匹配的特定列,并且作为响应,确定在对应于所述H矩阵的所述特定列的所述码字的值中存在错误。

20. 根据权利要求17所述的存储器模块,其中:

所述配置指令包括根据对应于所述ECC具体实施的H矩阵定义所述逻辑元件中要进行

掩码处理的位；

执行所述ECC具体实施包括解码码字,解码所述码字包括:

对于所述H矩阵的每个符号集,存储对应于第一行集的第一值集与对应于第二行集的第二值集的比率集;

使用具有根据所述配置指令进行了掩码处理的位的所述逻辑元件计算校验子;

确定所述校验子的对应于所述第一行集的第一部分等于特定符号集的所述比率与所述校验子的对应于所述第二行集的第二部分的乘积,并且作为响应,确定所述码字的对应于所述特定符号集的部分包含错误;

通过计算所述校验子的所述第一部分与对应于所述特定符号的所述第一行集的所述第一值集的商来确定所述错误的量值;

使用所述错误的所述量值和所述码字的被确定为包括所述错误的所述部分来纠正所述码字。

## 可编程纠错码编码和解码逻辑

### 背景技术

[0001] 本节所述的方法是可以实行的方法,但不一定是先前已经设想或实行过的方法。因此,除非另有说明,否则不应假设:本节所述的任何方法仅由于其包含在本节中而假定它们是现有技术。另外,不应假设:本节所述的任何方法仅由于其包含在本节中而假定它们是公知的、例行的或常规的。

[0002] 纠错码 (ECC) 存储器模块实现ECC引擎以检测并纠正内部数据损坏。常规ECC存储器模块的限制之一是特定ECC具体实施由制造商配置并且往往是制造商拥有专利的方法。因此,如果制造商用特定ECC具体实施来配置主机,则存取数据的任何存储器控制器必须用相同的ECC具体实施来配置。如本文所用,术语“主机”是指任何类型的处理元件,诸如中央处理单元 (CPU)、图形处理单元 (GPU)、存储器控制器等。例如,如果基于存储器内嵌处理器 (PIM) 的存储器模块将结果位用于内部错误检测和纠正并且识别不可纠正的损坏数据,则基于PIM的存储器模块及主机必须使用相同的ECC具体实施来确保相同等级的错误保护并且避免结果位的不一致使用。这是有问题的,因为基于PIM的存储器模块可被设计成与多个主机一起使用,然而例如芯片上系统 (SoC) 的主机设计者或公司可能不希望基于PIM的存储器供应商知道其正在使用的可能拥有专利的ECC具体实施。类似地,使用基于PIM的存储器模块的益处是数据的处理速度,因为数据不必从基于PIM的存储器模块传递到外部处理器。因此,需要一种可被配置成实施ECC以将存储在存储器模块中的数据解码的可编程存储器模块。

[0003] 另外,可能希望接收编码数据的任何进程能够将数据解码而不将请求发送回到初始编码器。例如,通过网络接收编码数据的计算设备可通过能够对数据执行错误检查而不通过网络向发送设备发送请求得以改进。因此,需要一种可被配置成实现ECC具体实施的系统作为常规ECC存储器模块的替代或补充。

### 附图说明

[0004] 在附图的图中以示例而非限制的方式描绘了实施方案,并且在附图中相同的附图标号是指类似的元件。

[0005] 图1A是描绘存储器布置的框图。

[0006] 图1B描绘了ECC编码器和/或解码器160的可配置逻辑元件的示例。

[0007] 图2是描绘用于编程ECC编码器或解码器的方法的流程图。

[0008] 图3A描绘了使用H矩阵来配置ECC具体实施的示例。

[0009] 图3B描绘了使用G矩阵来配置ECC具体实施的示例。

[0010] 图3C描绘了对特定数据值进行编码的示例。

[0011] 图3D描绘了将所存储的码字解码的示例。

[0012] 图3E描绘了将所存储的码字解码以识别多位错误的示例。

## 具体实施方式

[0013] 在以下描述中,出于解释的目的,阐述了许多具体细节以便提供对实施方案的透彻理解。然而,对于本领域技术人员将显而易见的是,可在没有这些具体细节的情况下实践实施方案。在其他情况下,以框图形式示出了公知的结构和设备,以避免不必要地模糊这些实施方案。

[0014] I. 概述

[0015] II. 架构

[0016] III. 通用编码器/解码器

[0017] IV. 操作概述

[0018] V. 编码

[0019] VI. 解码

[0020] I. 概述

[0021] 存储器模块包括可被编程为匹配特定ECC具体实施的一个或多个通用ECC引擎。如本文所用,术语“ECC具体实施”是指用于执行错误检测和后续处理的ECC功能性,例如使用错误检测的结果来执行错误纠正并且编码不能纠正的损坏数据等。

[0022] 通过提供通用ECC引擎,存储器模块可被配置成无需预编程到任何特定ECC具体实施即可实现ECC具体实施。这在可与支持不同ECC具体实施的不同主机设备配对的基于PIM的存储器模块中可能尤其具有价值。存储器模块被配置成接收定义ECC具体实施(诸如,定义为由主机使用的H矩阵)的输入。当存储器模块接收到对要存储在存储器模块中的数据进行编码或对存储在存储器模块中的数据进行解码的请求时,存储器模块通过选择性地使用存储器模块的逻辑元件来执行ECC具体实施。

[0023] II. 架构

[0024] 图1A是描绘存储器布置100的框图。存储器布置100包括主机110和存储器模块120。主机110和存储器模块120经由存储器总线130通信地联接。如本文所用,术语“总线”是指任何类型的有线或无线联接,诸如线材、导体及/或无线通信链路。另外,尽管本文中在总线的上下文中描述实施方案,但实施方案不限于总线本身并且适用于其他形式的存储器连接,诸如串行链路及电容性/电感性耦合。出于解释的目的,实施方案在附图中描绘并且在本文中在单个存储器模块120的上下文中描述,但实施方案适用于具有任何数目的存储器模块的存储器布置,诸如图1中的M个存储器模块。

[0025] 主机向存储器模块120写入数据并从其接收数据。主机可被配置成直接从存储器获得数据而不执行错误纠正,或从纠错码(ECC)编码器和/或解码器160获得数据。存储器布置100可包括附图中未描绘但出于解释的目的而在本文中描述的其他元件。

[0026] 例如可以是DRAM存储器模块的存储器模块120包括具有N个PIM单元的存储器布置(PIM单元1至PIM单元N)。每个PIM单元包括存储器150、ECC编码器和/或解码器160以及处理器170。存储器模块包括附图中未描绘但出于解释的目的而在本文中描述的其他元件,诸如缓冲器和解码器。PIM单元包括用于执行算术运算的处理逻辑和存储装置,诸如本地寄存器。虽然实施方案在附图中描绘并且在本文中在基于PIM的存储器模块的上下文中描述,但实施方案不限于该上下文并且适用于非基于PIM的存储器模块。

[0027] 存储器模块120可包括多个逻辑元件。逻辑元件包括被配置为执行逻辑操作的硬

件部件。示例性逻辑元件包括与门、或门、或非门、与非门、异或门和异或非门。逻辑元件可在硬件中实现,诸如通过使用充当电子开关的二极管或晶体管。

[0028] 图1B描绘了ECC编码器和/或解码器160的可配置逻辑元件的示例。如本文所用,“可配置逻辑元件”或“可配置逻辑块”是指可通过设定特定门中的特定位置而配置用于不同ECC具体实施的硬件逻辑的固定集。图1B的示例是可用作可配置ECC编码器和/或解码器的可配置逻辑元件的一个具体实施。其他选择可包括在各种层级处使用多路复用器的方案。在图1B的具体实施中,多个与门确定哪些数据位将传递到异或门。图1B的与逻辑元件中的每一者包括通过配置指令设定的位。如本文所用,将数据位“掩码”对应于将该位设定为“0”,其对于与门来说致使输入值不传递到异或门,而未被掩码的数据位对应于被设定为“1”的位,使得输入值通过与门传递到异或门。ECC编码器和/或解码器160可根据奇偶校验矩阵(H矩阵)和/或生成矩阵(G矩阵)进行掩码处理,如本文中进一步描述。

### [0029] III. 通用编码器/解码器

[0030] 根据一个实施方案,存储器模块120配置有用于ECC具体实施的通用编码器和/或解码器。如本文所用,术语“通用编码器及/或解码器”是指将用于ECC具体实施的数据编码及/或将用于ECC具体实施的数据解码的可编程处理逻辑。通用编码器和/或解码器是可编程的,使得其可用多个ECC具体实施中的任一者来实现,只要其用特定于ECC具体实施的信息来编码即可。用于实现编码器和/或解码器的引擎可以可取决于特定具体实施而变化的若干不同方式实现。示例可包括但不限于CPU、GPU、微控制器、现场可编程门阵列(FPGA)、专用集成电路(ASIC)以及其他类型的处理逻辑。ECC引擎可扩充存储器模块120内部的任何ECC能力,诸如高带宽存储器3(HBM3)存储器模块中的管芯上ECC引擎。

[0031] ECC编码器和/或解码器160可在存储器150与处理器170之间实现,使得从存储器150存取的数据可在由处理器170使用之前被解码,并且由处理器170生成的数据可在存储在存储器150中之前被编码。在图1A中,存储器模块中的每一者包括用于通过PIM单元中的ECC编码器和/或解码器实现的逻辑元件的掩码指令。在其他实施方案中,掩码指令可单独存储并且用于将逻辑元件掩码。另外地或另选地,多个PIM单元可共享对单个掩码指令集的访问。又如,多层存储器模块可每层配置有一个掩码指令集。

[0032] 主机处理元件(本文中也称为“主机”)通过提供识别多个逻辑元件以针对特定ECC具体实施进行掩码的掩码指令来对通用编码器/解码器编程。例如,主机可提供对应于在特定ECC具体实施中使用的H矩阵的数据。存储器模块可基于H矩阵来识别待激活的逻辑元件以及不激活或进行掩码处理的逻辑元件。例如,FPGA、EPROM等可用特定H矩阵及关于如何使用H矩阵来编码或解码数据的指令来编程。又如,主机可将定义用于特定实施方案的H矩阵的数据提供给存储器模块以供PIM单元或微控制器执行。掩码指令由PIM单元的用于处理PIM命令的处理逻辑、由PIM单元内的单独处理逻辑或PIM单元内的微控制器来处理。

[0033] 本文所述的掩码指令和任何另外的ECC具体实施指令可存储在如图1所描绘的存储器模块120的存储器阵列(即,库)中。另选地,指令可存储在PIM单元中或存储在单独设备中。掩码指令和任何另外的ECC具体实施指令可以是任何类型或形式的指令。例如,指令可以是源代码(例如,寄存器传输语言(RTL)指令)、解析源代码、编译代码(例如,目标代码)或可执行代码(例如,二进制代码)的形式。指令的另一种示例性形式是与FPGA一起使用的硬件描述语言,诸如Verilog和VHDL。主机可编译源代码以创建可执行代码,该可执行代码被

提供给PIM模块以存储和执行。另选地,主机可向PIM或微控制器提供源代码,该PIM或微控制器编译源代码以创建被存储和执行的可执行代码。因此,可在主机处或在存储器模块处执行从源代码创建可执行代码。

[0034] ECC编码器和/或解码器可被配置成编码从处理器写入到存储器的所有数据并且解码由处理器从存储器读取的所有数据。通过自动编码和解码由处理器从存储器存取和存储到存储器中的数据,存储器模块通过确保存储在存储器中的数据可被成功解码,而不需要专门的命令来导致解码或检测编码数据,从而改进系统的性能。

[0035] 根据一个实施方案,通用ECC编码器/解码器在PIM单元内的处理器与存储器之间执行ECC处理。例如,ECC编码器/解码器可以是配置成从PIM单元的存储器接收数据、对数据执行错误检查并且将结果提供给PIM单元的处理器可配置逻辑块。相反,ECC编码器/解码器可被配置成从PIM单元的处理器接收新数据,用校验位对新数据进行编码,并且使得编码数据存储在存储器中。在另一个具体实施中,ECC编码器/解码器可被配置成经由网络接收数据并且在利用数据或将数据发送到存储装置之前对数据执行错误纠正和/或编码。

[0036] 根据一个实施方案,通用编码器/解码器被配置成用不同ECC具体实施来实现。不同ECC具体实施可包括由主机使用的拥有专利的编码器/解码器。通用编码器/解码器可通过用ECC具体实施使用的特定H矩阵来编程ECC而配置用于特定ECC具体实施。当由主机使用的ECC的确切具体实施变化时,这使得通用编码器/解码器适用于不同系统。

#### [0037] IV. 操作概述

[0038] 本文所述的通用编码器/解码器可通过使用配置信息来识别逻辑元件以针对ECC具体实施进行激活而经由硬件来实现。其他实施方案可通过数字编程计算在软件中实现类似的逻辑处理。

[0039] 图2是描绘用于实现ECC编码器和/或解码器的方法的流程图。步骤202至208包括用于在存储之前使用可配置逻辑元件来编码数据的方法。在步骤202处,根据ECC具体实施来配置可配置逻辑块。例如,可配置逻辑块可包括具有多个异或门的异或树。可配置逻辑块可被配置成根据配置指令集将通向异或树中的异或门的多个与门中的数据位掩码。指令可直接指示与门的要掩码的位。另外地或另选地,指令可包括存储器模块用来确定要掩码与门的哪些位的二进制或符号矩阵。本文还描述了用于将特定掩码以匹配ECC具体实施的具体实施。

[0040] 图3A描绘了使用H矩阵来配置ECC具体实施的示例。H矩阵302包括从主机设备接收的表示从数据生成码字的线性函数的H矩阵。图3A中描绘的H矩阵包括 $GF(2^3)$ 表示、 $\alpha$ 以及二进制表示。虽然本文所述的方法是关于二进制矩阵来描述的,但相同方法可应用于任何H矩阵。列0至17表示数据,而列18至23表示校验位。使用H矩阵,系统生成包括可配置逻辑元件的与门中的未掩码数据位和掩码数据位的H矩阵异或门输入304。H矩阵异或门输入304的形式以与H矩阵等效的形式描绘以提供清楚的示例。在实施方案中,H矩阵异或门输入304的形式包括异或树,其中根据配置指令将通向异或门的与门中的数据位掩码。一旦通过将数据位掩码进行配置,就可接着利用H矩阵异或门输入304来对编码数据执行错误纠正。

[0041] 图3B描绘了使用G矩阵来配置ECC具体实施的示例。G矩阵303包括可使用已知技术从H矩阵导出和/或可由主机设备提供的G矩阵。图3B中描绘的G矩阵包括 $GF(2^3)$ 表示、 $\alpha$ 以及二进制表示。虽然本文所述的方法是关于二进制矩阵来描述的,但相同方法可应用于任何G

矩阵。使用G矩阵,系统生成包括可配置逻辑元件的与门中的未掩码数据位和掩码数据位的G矩阵异或门输入305。G矩阵异或门输入305的形式以与G矩阵等效的形式描绘以提供清楚的示例。在实施方案中,G矩阵异或门输入305的形式包括异或树,其中根据配置指令将通向异或门的与门中的数据位掩码。一旦进行配置,就可接收利用G矩阵异或门输入305来产成待编码的数据的检验位。

[0042] 在步骤204处,可配置块接收待编码的数据。例如,PIM模块包括允许对存储器进行读取和写入而无需连接到主机处理器的处理器。当处理器用所存储的数据完成计算时,处理器将结果存储在存储器中。在存储结果之前,存储器模块可使数据通过ECC编码器以致使ECC编码器使用本文所述的方法对结果进行编码。

[0043] 虽然以对PIM存储和数据检索的请求来描述实施方案,但本文所述的系统和方法可应用于存储或接收由单独设备编码的数据的任何系统。例如,如果计算设备经由网络接收编码信息,则计算设备可使用本文所述的方法在检索期间对数据执行错误纠正,而不需要将另外的请求发送到编码设备。

[0044] 在一个实施方案中,计算设备配置有多个纠错码具体实施。当计算设备经由网络接收编码数据时,计算设备可例如诸如数据的发送方或基于与数据一起发送的另外信息来识别应用于数据的纠错码具体实施。计算设备可使用所识别的纠错码具体实施来解码数据。另外,如果数据要被发送回单独设备,则计算设备可识别对应于该单独设备的纠错码具体实施并使用该纠错码具体实施来编码数据。

[0045] 在步骤206处,数据通过可配置逻辑块以生成校验位。例如,可配置逻辑元件可根据G矩阵而配置有掩码位和未掩码位。可使数据通过可配置逻辑元件,其中对应于未掩码与门的数据前进到异或门,以生成检验位。

[0046] 在步骤208处,数据通过将校验位附加到数据来编码并存储在存储器中。本文还描述了使用可配置逻辑元件的编码方案。

[0047] 步骤210至216包括用于在使用之前使用可配置逻辑元件对数据执行纠错的方法。在步骤210处,根据ECC具体实施来配置可配置逻辑块。例如,可配置逻辑块可包括具有多个异或门的异或树。可配置逻辑块可被配置成根据配置指令集将通向异或树中的异或门的多个与门中的数据位掩码。指令可直接指示与门的要掩码的位。另外地或另选地,指令可包括存储器模块用来确定要掩码与门的哪些位的二进制或符号矩阵。本文还描述了用于将特定掩码以匹配ECC具体实施的具体实施。

[0048] 在步骤212处,可配置逻辑块接收待解码的数据。例如,PIM模块包括允许对存储器进行读取和写入而无需连接到主机处理器的处理器。如果存储在PIM的存储器中的数据是由主机处理器创建并因此用特定ECC具体实施来编码,则PIM的处理器可从存储器请求数据以便对数据执行一个或多个操作。作为实际示例,PIM可被配置成使用由主机写入存储器中的数据来执行模拟并随后将模拟的结果存储在存储器中。在将数据发送到处理器以执行模拟之前,存储器模块可使数据通过ECC解码器以致使ECC解码器使用本文所述的方法对数据进行解码。

[0049] 在步骤214处,数据通过可配置逻辑块以生成校验子。例如,可配置逻辑元件可根据H矩阵而配置有掩码位和未掩码位。可使数据通过可配置逻辑元件,其中对应于未掩码与门的数据前进到异或门,以生成检验子。

[0050] 在步骤216处,使用校验子执行错误纠正并且将经纠正的数据提供给处理器。例如,校验子可用于识别错误位置及量值,从而可纠正错误。本文还描述了使用可配置逻辑元件的解码方案。

#### [0051] V. 编码

[0052] 在一个实施方案中,ECC具体实施包括对要存储在存储器中的新数据进行编码。可响应于来自处理器的将数据存储在存储器中的请求及/或响应于经由网络接收要与其他编码数据一起存储的数据而执行编码。

[0053] 在一个实施方案中,编码数据包括使用根据特定ECC具体实施配置的可配置逻辑元件来确定用于特定数据值的多个校验位。可将检验位附加到特定数据值的位以生成用于特定数据值的码字。可随后将码字存储在存储器中。在一个实施方案中,ECC编码器通过经由可配置逻辑元件发送对应于数据值的位来确定多个校验位值,从而形成矩阵乘法。例如,ECC编码器可包括具有根据H矩阵设定的位的多个与门。由于仅从数据值接收数据位的与门将向异或门发送数据位,所以不对应于数据值的位的列将不向异或门贡献数据位以计算对应于异或门的行的值。因此,如果特定数据值在第0位和第2位中包括“1”,则ECC编码器可仅将这些位从第0列和第2列中的“与”门发送到异或门。

[0054] 图3C描绘了对特定数据值进行编码的示例。在图3C的示例中,系统使用G矩阵来计算数据值的校验位。在其他实施方案中,系统可使用H矩阵并计算校验位列的值,当数据值通过逻辑元件相乘时,校验位列的值将导致计算出的校验子为0。

[0055] 在图3C中,数据值306包括要编码的数据值的二进制表示。数据值306与在逻辑元件中编码的G矩阵相乘。G矩阵异或门输入305的每个循环对应于具有未掩码位的与门。因此,通过没有循环的门的任何位将不被传递到异或门。当数据值306通过G矩阵异或门输入305时,数据值306的每个数据位(1)通过对应列中的与门,从而形成编码异或门输入308。编码异或门输入304包括数据位所通过的每个门。由于数据位仅在第一列和第三列中被设定为“1”,因此在编码异或门输入308时仅描绘G矩阵异或门输入305的第一列和第三列。

[0056] 在编码异或门输入308的剩余位通过异或门之后,它们生成校验位310。例如,第一行包括通过与门的两个位,从而在它们通过异或门时生成0。类似地,行2和5包括两个位,从而在它们通过它们各自的异或门时也生成0。行3和4各自具有通过异或门的单个位,因此生成1,而行6没有通过异或门的位。因此,最终校验位310包括(0 0 0 1 1 0 0)。然后将校验位310附加到数据值306以生成编码码字312。

[0057] 以上描述提供了用于形成校验位值的矩阵乘法的硬件具体实施。矩阵乘法可被计算为:

$$[0058] \quad \sum_{j=1}^K G_{ij}D_j = C_i \quad (\text{对于 } i = 1 \dots R)$$

[0059] 其中 $G_{ij}$ 是G矩阵的第i行和第j列的值,D是第j列的数据值, $C_i$ 是第i行的校验位值,K是列的总数,并且R是G矩阵的行数。以上矩阵乘法可在符号空间或二进制空间中执行,结果相同。

#### [0060] VI. 解码

[0061] 在一个实施方案中,ECC具体实施包括将在ECC解码器处接收的数据解码。可响应

于来自处理器的存取存储在存储器模块中的所存储数据的请求及/或响应于经由网络接收编码数据而执行解码。解码数据可包括使用所存储的码字来识别码字的一个或多个位中的错误并且调整码字的位以修复错误。

[0062] 在一个实施方案中,ECC解码器首先使用根据特定ECC具体实施配置的可配置逻辑元件来计算校验子列。系统随后将校验子列与对应于可配置逻辑元件的每一列进行比较以识别匹配校验子的列。系统确定与校验子匹配的列包含错误。在二进制具体实施中,错误的值始终为1,因此一旦识别出列,即可固定位。

[0063] 图3D描绘了将所存储的码字解码的示例。在图3D中,码字314包括使用包含错误的ECC具体实施编码的数据。类似于编码,码字314与在逻辑元件中编码的H矩阵相乘。H矩阵异或门输入304的每个循环对应于具有未掩码位的与门。由于码字314包括在编码步骤生成的校验位,因此校验位被另外地馈送通过使用包括对应于校验位的列的H矩阵来配置的可配置逻辑元件。因此,解码异或门输入316包括五个列,位通过这些列以计算所得校验子。存储器模块将校验子318与H矩阵异或门输入304的列进行比较。由于校验子318在第二、第四和第五行中包括1,因此存储器模块搜索第二、第四和第五行中的列。校验子318与H矩阵异或门输入304的第五列匹配。因此,码字314的第五位包含错误。存储器模块可通过按错误值改变位来纠正码字。因此,第五列中的1被替换为0。

[0064] 以上描述提供了用于计算校验子的矩阵乘法的硬件具体实施。矩阵乘法可被计算为:

$$[0065] \quad \sum_{j=1}^{KN} H_{ij} D_j = S_i (\text{对 } i = 1 \dots)$$

[0066] 其中 $H_{ij}$ 是H矩阵的第i行和第j列的值,D是码字的第j列的数据值, $S_i$ 是第i行的校验子值,N是列的总数,并且R是H矩阵的行数。以上矩阵乘法可在符号空间或二进制空间中执行,结果相同。

[0067] 图3D的实施方案表示一种用于识别和修复单位错误的方法。因此,将校验子匹配到H矩阵的列可通过识别在相同位置中具有与校验子相同的值的H矩阵的列来执行。如果所计算出的校验子是仅0个值的列,则存储器模块可确定码字中不存在错误。如果所计算出的校验子是与H矩阵中的列中的任一列不匹配的列,则存储器模块可确定码字包含多位或多符号错误。

[0068] 对于多位错误,系统可利用符号记法来识别同一符号内的单位错误或多位错误。如本文所用,符号是指多个位诸如3个位的集。例如,图3A中的H矩阵的第一表示包括由包括1的第一行和包括 $\alpha$ 的幂的第二行表示的两行符号,其中每个值包括3位符号。对应于符号的 $3 \times 3$ 矩阵可包括两个3位符号的乘法的表示,其中一个符号包括被编码到矩阵中的常数。例如,前3行和前3列对应于 $D_0$ 与1的第一次相乘,而后面3行和后面3列对应于 $D_0$ 与 $\alpha^1$ 的第一次相乘,其中 $D_0$ 是第一数据符号(3位)。

[0069] 在一个实施方案中,使用H矩阵来执行解码,其中在符号空间中,第一行的每一列包括符号值1。例如,在图3A的H矩阵中,前3行的3列的每个重复组包括在对角线上为1而在其他地方为0的对角矩阵。当每一列的行的值为1时,校验子包含错误。因此,图3D的校验子318中的前三个值 $S_0 = 010$ 将包含错误。可将错误应用于每个符号列以确定校验子何时匹配

符号。将错误应用于每个符号列可包括使错误通过对应于H矩阵的符号的一组“与”门并且利用“异或”门计算结果。

[0070] 使用图3D作为实际示例,当010的错误与H矩阵异或门输入304的第一组三列相乘时,仅第二列接收通过与门的数据位。由于第二列包括第二行和第六行中的置位,所以异或门计算值(010001)。由于这与校验子(010110)不匹配,所以系统可检查下一组列。当010的错误与H矩阵异或门输入304的第一组三列相乘时,仅第五列接收通过与门的数据位。由于第五列包括第二、第四和第五行中的置位,所以异或门计算与校验子匹配的值(010110)。因此,错误在第五列中。

[0071] 图3E描绘了将所存储的数据解码以识别多位错误的示例。如果错误已包含110,则存储模块将保持前两列并将其与错误进行比较。例如,改变代码字以在第四列和第五列中包含错误(因此码字的前六个值是101110),所计算出的校验子将包含(110111)。如果将错误110应用于第一符号,即H矩阵异或门输入304的前三列,则第一列和第二列保持,从而得到值(110011)。由于此值与校验子不匹配,因此存储器模块移动到下一符号。将110应用于第二符号(即H矩阵异或门输入304的后面三列)留下第四列和第五列,当通过异或门组合时,得到与校验子匹配的值(110111)。因此,系统将确定错误存在于第二符号中,并且将错误值110应用于第二符号指示错误在第四列和第五列中。

[0072] 在一般场景中,系统可被配置成使用H矩阵的第n行与H矩阵的多个其他行之间的比率来计算错误位置。这些比率可使用以下等式预先计算并存储以用于ECC具体实施:

$$[0073] \quad I_{ij} = \frac{H_{ij}}{H_{nj}}$$

[0074] 其中 $I_{ij}$ 是特定行和特定列的比率值, $H_{ij}$ 是H矩阵在特定行和特定列处的值, $H_{nj}$ 是H矩阵在第n行处的值。如上所述,如果 $H_{nj}$ 对于每行均为“1”,则该比率等于H矩阵的特定列和行的值。从H矩阵的多个行中选择作为 $H_{nj}$ 的行可以是任何行。在一个实施方案中,如果 $H_{nj}$ 的所选行在列中包含“0”,则系统选择 $H_{nj}$ 的不同行用于该列,使得比率的计算不是不确定的。

[0075] 为了计算错误的位置,系统可测试每一列(j)以确定以下等式对于列(j)的每一行是否成立:

$$[0076] \quad S_i = I_{ij} * S_n$$

[0077] 其中 $S_i$ 是第i行的校验子值,并且 $S_{1n}$ 是与用于 $H_{nj}$ 的相同的行的校验子值。因此,如果对于第一列, $H_{nj}$ 是第二行,则 $S_n$ 将是第n行的校验子值。另外,如果对于第二列, $H_{nj}$ 是第六行,则 $S_n$ 将是第六行的校验子值。对于特定列,如果校验子值与每一行的比率的乘积等于该行的校验子值,则系统可确定错误存在于该特定列中。

[0078] 系统可进一步使用校验子的值来计算错误量值。该步骤可在计算错误位置之前、之后执行或与计算错误位置并行地执行。错误值可被计算为:

$$[0079] \quad E_j = S_n * H_{nj}^{-1}$$

[0080] 其中 $E_j$ 是错误。用于 $S_n$ 和 $H_{nj}^{-1}$ 的行可以是在错误位置的计算中使用的相同行和/或可以是不同的行,只要将相同行用于错误计算等式中的 $S_n$ 和 $H_{nj}^{-1}$ 即可。可在实现之前对每一列预先计算 $H_{nj}^{-1}$ 的值。该值可另外对每列使用相同的值,或者可在列之间改变行。系统可存储指示对于每一列,哪一行正用于 $H_{nj}^{-1}$ 的数据。

[0081] 一旦错误值和错误位置已知,系统就可通过将错误值应用于错误位置来纠正错误。如果系统执行以上计算以识别错误位置并且没有计算对所有行满足以上等式,则系统可确定码字包含多符号错误。一般来讲,上述方法可检测初始ECC ECC代码可检测的任何错误并且可纠正任何单符号错误。可关于任何线性区块码来执行本文所述的方法。

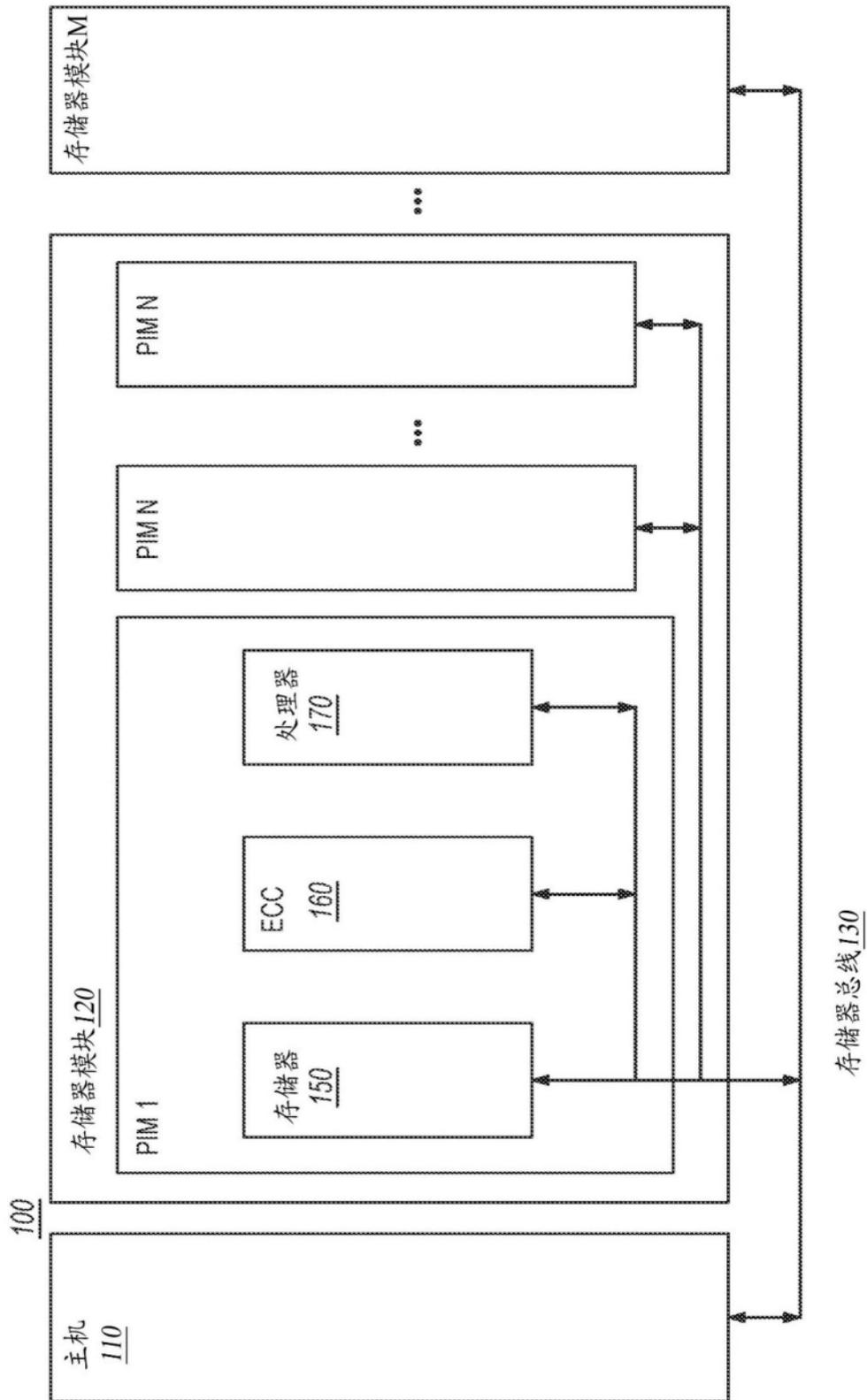


图1A

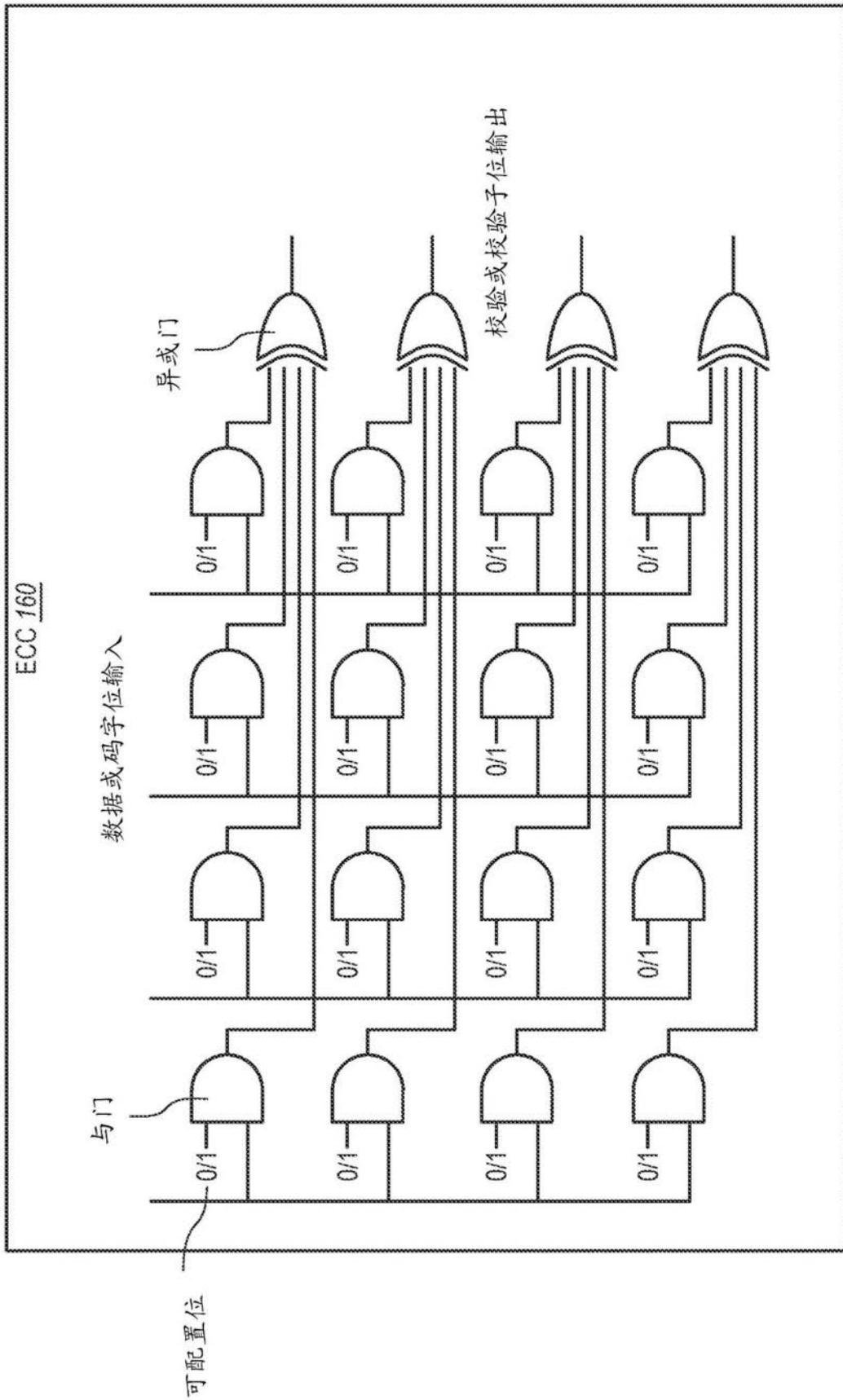


图1B

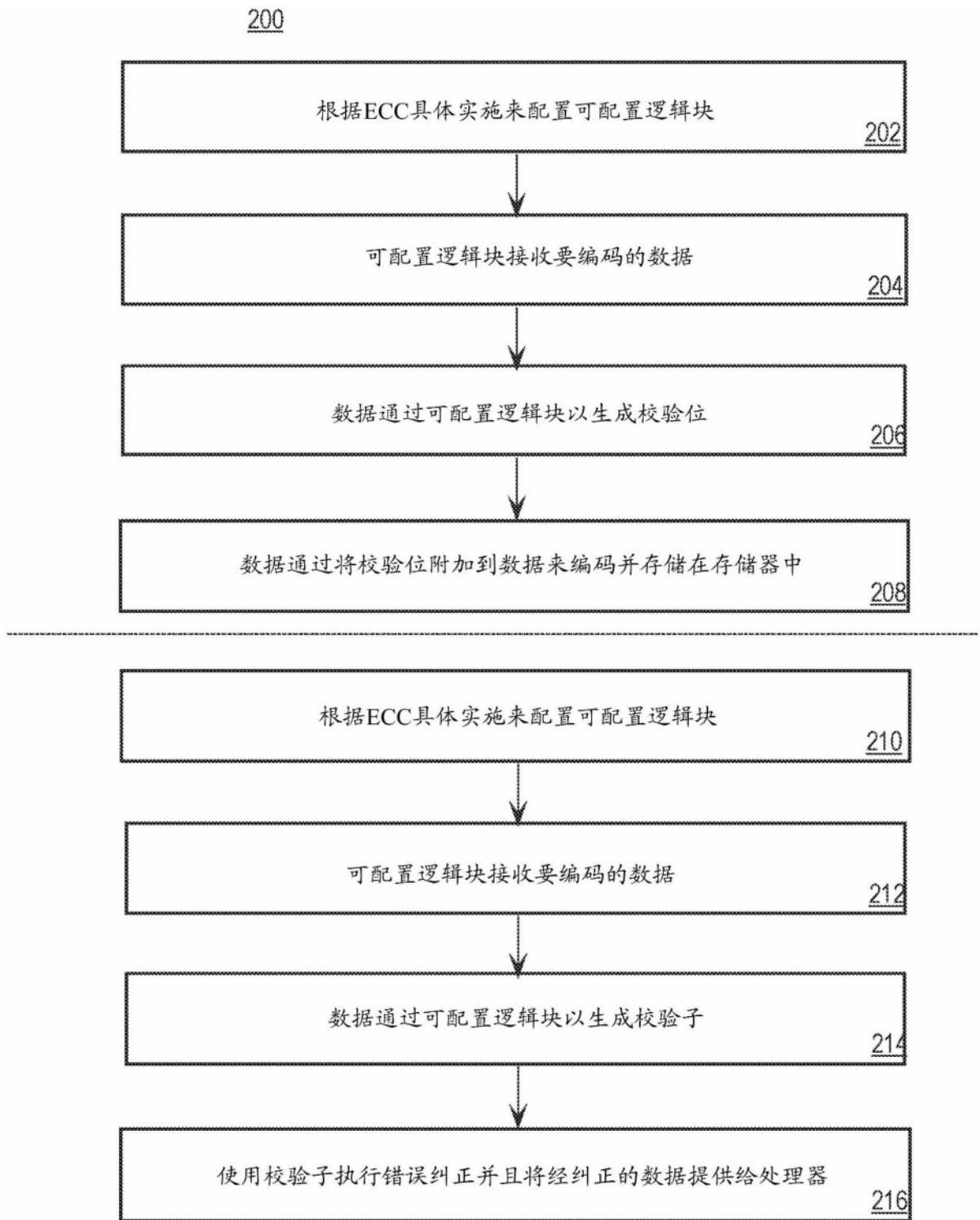


图2

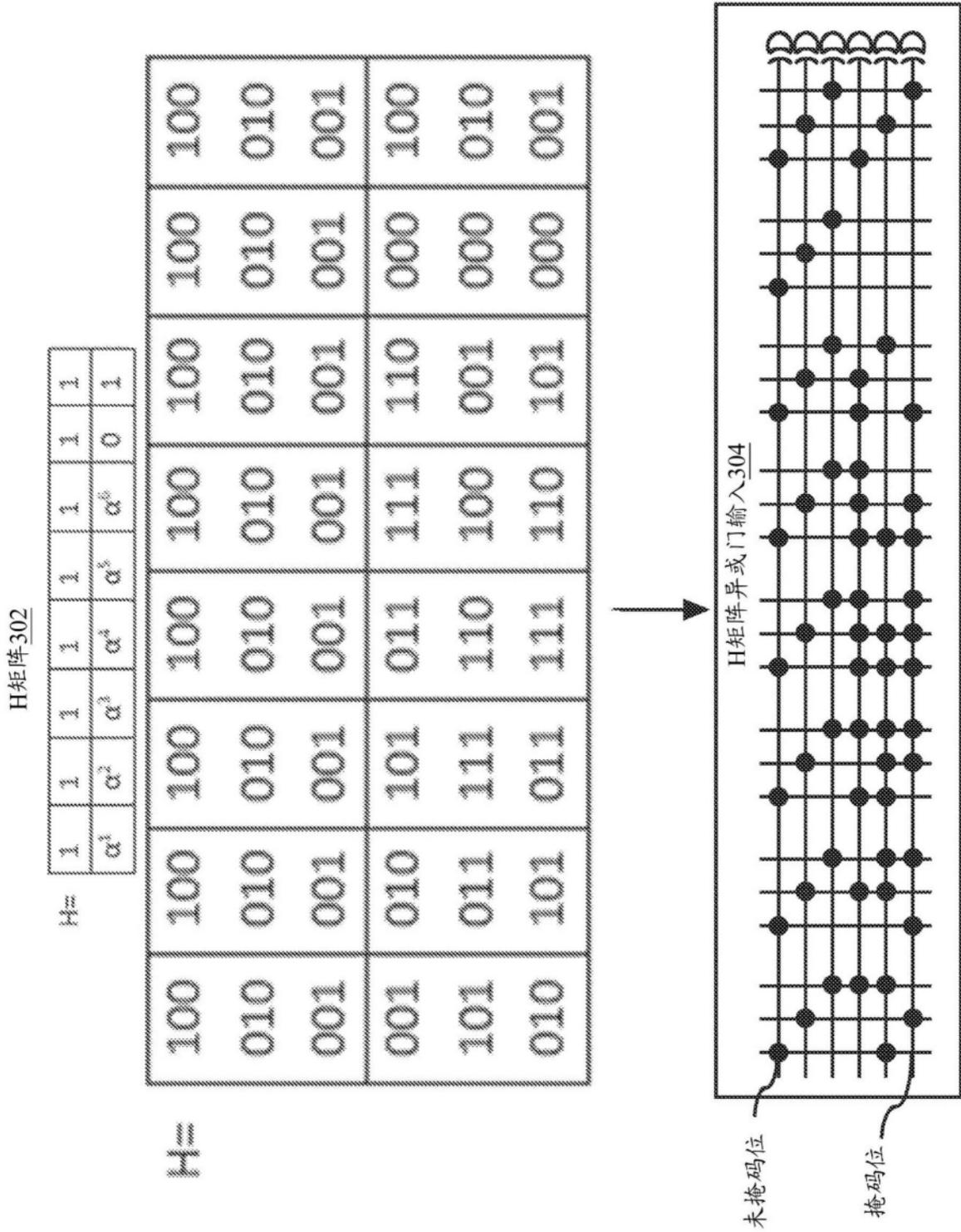


图3A

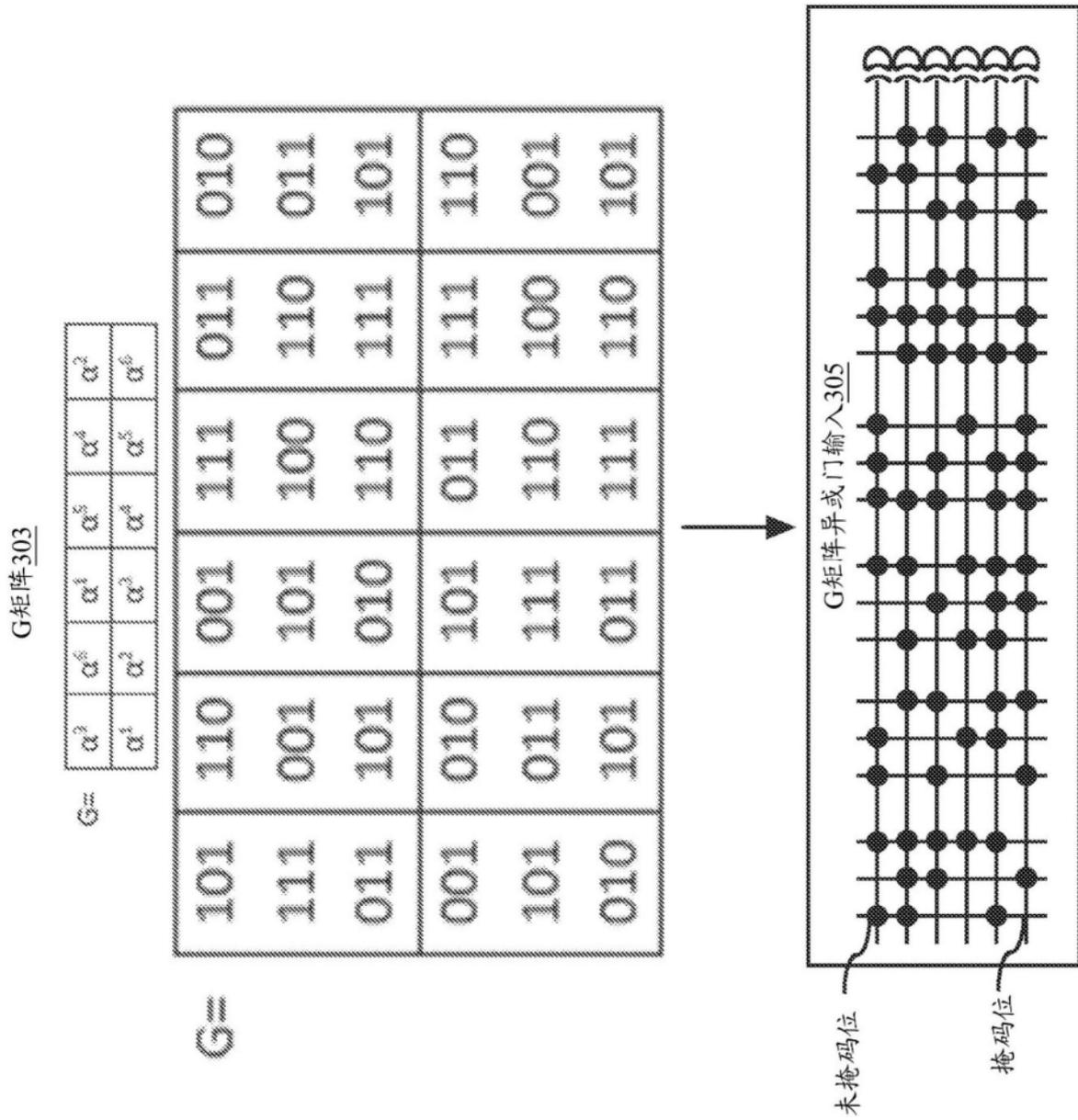


图3B





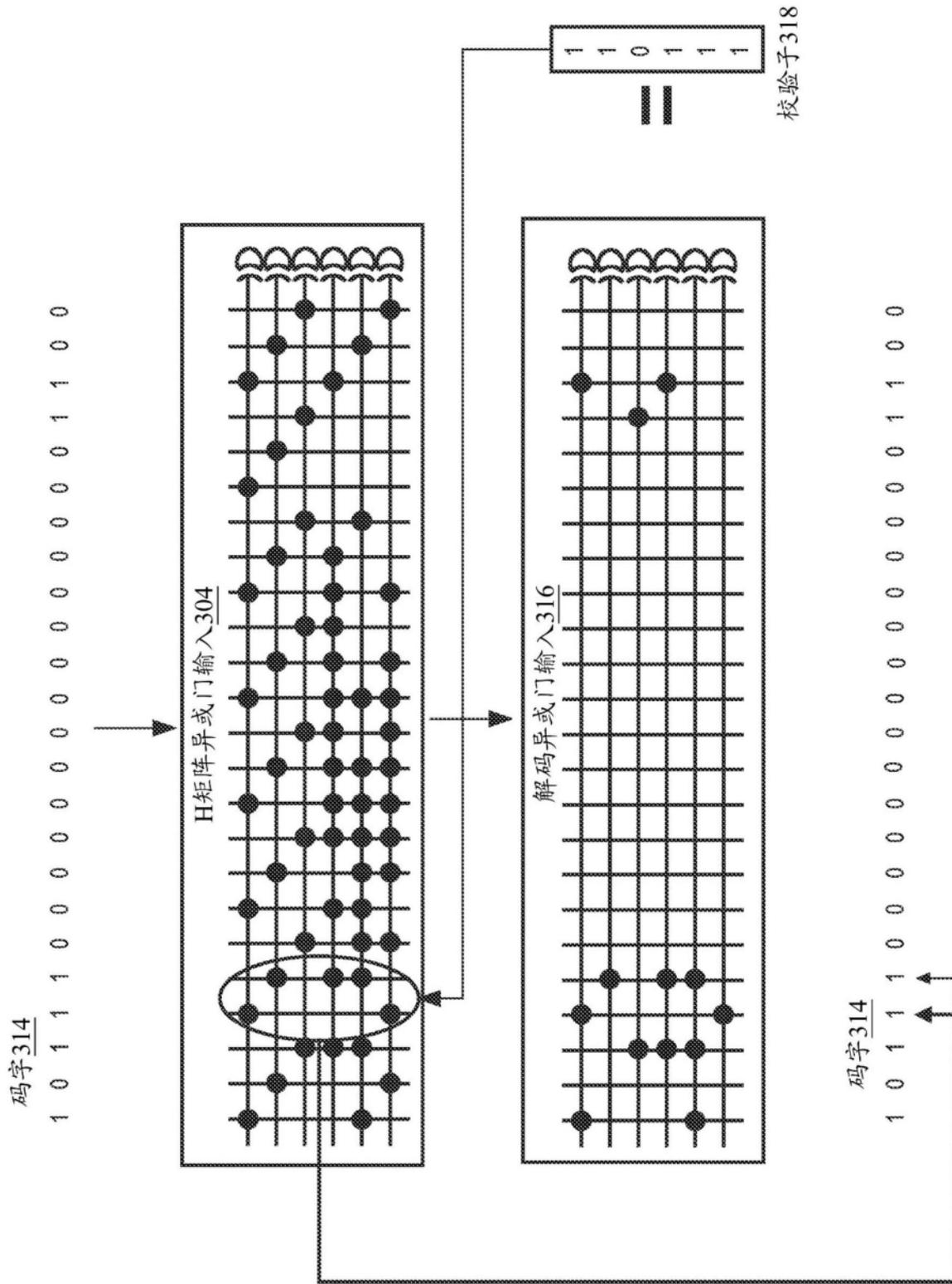


图3E